NIST SPECIAL PUBLICATION 1800-16

Securing Web Transactions

TLS Server Certificate Management

Includes Executive Summary (A); Security Risks and Recommended Best Practices (B); Approach, Architecture, and Security Characteristics (C); and How-To Guides (D)

Mehwish Akram
William C. Barker
Rob Clatterbuck
Brandon Everhart
Jane Gilbert
William Haag
Brian Johnson
Alexandros Kapasouris
Dung Lam
Brett Pleasant
Mary Raguso
Murugiah Souppaya
Susan Symington
Paul Turner
Clint Wilson

DRAFT

This publication is available free of charge from:
https://www.nccoe.nist.gov/projects/building-blocks/tls-server-certificate-management

National Institute of Standards and Technology
U.S. Department of Commerce

NATIONAL CYBERSECURITY CENTER OF EXCELLENCE

NIST SPECIAL PUBLICATION 1800-16

Securing Web Transactions: TLS Server Certificate Management

Includes Executive Summary (A); Security Risks and Recommended Best Practices (B); Approach, Architecture, and Security Characteristics (C); How-To Guides (D)

William Haag
Murugiah Souppaya
NIST

Paul Turner
Venafi

William C. Barker
Dakota Consulting

Mehwish Akram
Brandon Everhart
Brian Johnson
Brett Pleasant
Mary Raguso
Susan Symington
The MITRE Corporation

Clint Wilson
DigiCert

Dung Lam
F5

Alexandros Kapasouris
Symantec

Rob Clatterbuck
Jane Gilbert
SafeNet Assured Technologies

DRAFT

July 2019

U.S. Department of Commerce
Wilbur Ross, Secretary

National Institute of Standards and Technology
Walter Copan, NIST Director and Undersecretary of Commerce for Standards and Technology

NIST SPECIAL PUBLICATION 1800-16A

Securing Web Transactions

TLS Server Certificate Management

Volume A:
Executive Summary

William Haag
Murugiah Souppaya
NIST

Paul Turner
Venafi

William C. Barker
Dakota Consulting

Mary Raguso
Susan Symington
The MITRE Corporation

July 2019

DRAFT

This publication is available free of charge from:
https://www.nccoe.nist.gov/projects/building-blocks/tls-server-certificate-management

National Institute of Standards and Technology
U.S. Department of Commerce

NCCoE
NATIONAL CYBERSECURITY CENTER OF EXCELLENCE

Executive Summary

The internet has enabled rapid, seamless commerce across the globe. Billions of dollars' worth of transactions are performed across the internet every day. This is possible only because connections across the internet are trusted to be secure. Transport Layer Security (TLS), a cryptographic protocol, is fundamental to this trust.

Organizations leverage TLS to provide the connection security that has enabled today's unprecedented levels of commerce across the internet. TLS, in turn, depends on TLS certificates. Organizations must deploy TLS certificates and corresponding private keys to their systems to provide them with unique identities that can be reliably authenticated. The TLS certificate enables anybody connecting to a system to know that they are sending their data to the right place. In addition, it also enables establishment of secure connections so that no one in the middle can eavesdrop on communications.

Many organizations might be surprised to discover how many TLS certificates they have. A large- or medium-scale enterprise may have thousands or even tens of thousands, each identifying a specific server in their environment. This is because organizations use TLS not only to secure external connections between themselves and their customers over the internet but also to establish trust between different machines inside their own organization and thereby secure internal communications.

Even though TLS certificates are critical to the security of both internet-facing and private web services, many organizations do not have the ability to centrally monitor and manage their certificates. Instead, certificate management tends to be spread across each of the different groups responsible for the various servers and systems in an organization. Central security teams struggle to make sure that certificates are being properly managed by each of these disparate groups. This lack of a central certificate management service puts the organization at risk because once certificates are deployed, they require regular monitoring and maintenance. Organizations that improperly manage their certificates risk system outages and security breaches, which can result in revenue loss, harm to reputation, and exposure of confidential data to attackers.

The National Cybersecurity Center of Excellence (NCCoE) at the National Institute of Standards and Technology (NIST) built a laboratory environment to explore and develop guidelines to help large and medium enterprises better manage TLS server certificates by:

- defining operational and security policies and identifying roles and responsibilities
- establishing comprehensive certificate inventories and ownership tracking
- conducting continuous monitoring of certificates' operational and security status
- automating certificate management to minimize human error and maximize efficiency on a large scale
- enabling rapid migration to new certificates and keys when certificate authorities or cryptographic mechanisms are found to be weak, compromised, or vulnerable

The NCCoE has identified as a best practice that all enterprises establish a formal TLS server certificate management program that is consistent with overall organizational security policies and that has executive responsibility, guidance, and support for the following purposes:

- Recognize the harm that improper management of TLS server certificates can cause to business operations, and provide guidance to mitigate risks related to TLS certificates.
- Ensure that the central certificate services team and the local application owners and system administrators understand the risks to the enterprise and are accountable for their roles in managing TLS server certificates.
- Establish an action plan to implement these recommendations and track progress.

CHALLENGE

As the use of web transactions has grown, the number of TLS server certificates has increased to many thousands in some enterprises. Many of these enterprises struggle to effectively manage their certificates and, as a result, face significant risks to their core operations, including:

- application outages caused by expired TLS server certificates
- hidden intrusion, exfiltration, disclosure of sensitive data, or other attacks resulting from encrypted threats or server impersonation
- disaster-recovery risk that requires the rapid replacement of large numbers of certificates and private keys in response to either certificate authority compromise or discovery of vulnerabilities in cryptographic algorithms or libraries

Challenges to TLS server certificate management include the broad distribution of certificates across enterprises, the complexity of certificate management processes, and the multiple roles involved in certificate management and issuance. TLS server certificates are typically issued by a central certificate services team, but the certificates are often installed and managed by the groups (lines of business) and local system administrators responsible for individual web servers, application servers, network devices, and other network components for which certificates are used. Some of these managers and administrators lack awareness of the risks and best practices associated with certificate management. Certificate services teams having this awareness often lack access to systems holding the certificates.

Despite the mission-critical nature of TLS server certificates, many organizations have not defined clear policies, processes, roles, and responsibilities needed for effective certificate management. Moreover, many organizations do not leverage available technology and automation to effectively manage the growing numbers of certificates. The consequence is continuing incidents due to TLS certificate issues.

SOLUTION

Executive leadership should establish formal TLS server certificate management programs across their enterprises and set organization-specific implementation milestones. For example:

- Within 30 days, define the TLS server certificate policies, and communicate the responsibilities.
- Within 90 days, establish the inventory of TLS server certificates, and identify the risks.
- Beyond 90 days, address near-term risks, and establish automated implementation processes.

The NCCoE, in collaboration with industry partners, has developed this practice guide, *Securing Web Transactions: TLS Server Certificate Management*, to help large- and medium-size organizations better manage TLS server certificates. It provides recommended best practices for large-scale TLS server certificate management and describes the automated TLS certificate management example solution that was built to demonstrate how to prevent, detect, and recover from certificate-related incidents.

While the NCCoE used a suite of commercial products to address this challenge, this guide does not endorse these particular products, nor does it guarantee compliance with any regulatory initiatives. Your organization's information security experts should identify the products that will best integrate with your existing tools and IT system infrastructure. Your organization can adopt this solution or one that adheres to these guidelines in whole, or you can use this guide as a starting point for tailoring and implementing parts of a solution.

SHARE YOUR FEEDBACK

You can view or download the guide at https://nccoe.nist.gov/projects/building-blocks/tls-server-certificate-management. Help the NCCoE make this guide better by sharing your thoughts with us as you read the guide. If you adopt this solution for your own organization, please share your experience and advice with us. We recognize that technical solutions alone will not fully enable the benefits of our solution, so we encourage organizations to share lessons learned and best practices for transforming the processes associated with implementing this guide.

To provide comments or to learn more by arranging a demonstration of this example implementation, contact the NCCoE at tls-cert-mgmt-nccoe@nist.gov.

TECHNOLOGY PARTNERS/COLLABORATORS

Organizations participating in this project submitted their capabilities in response to an open call in the Federal Register for all sources of relevant security capabilities from academia and industry (vendors and integrators). The following respondents with relevant capabilities or product components (identified as "Technology Partners/Collaborators" herein) signed a Cooperative Research and Development Agreement (CRADA) to collaborate with NIST in a consortium to build this example solution.

Certain commercial entities, equipment, products, or materials may be identified by name or company logo or other insignia in order to acknowledge their participation in this collaboration or to describe an experimental procedure or concept adequately. Such identification is not intended to imply special status or relationship with NIST or recommendation or endorsement by NIST or NCCoE; neither is it intended to imply that the entities, equipment, products, or materials are necessarily the best available for the purpose.

The National Cybersecurity Center of Excellence (NCCoE), a part of the National Institute of Standards and Technology (NIST), is a collaborative hub where industry organizations, government agencies, and academic institutions work together to address businesses' most pressing cybersecurity challenges. Through this collaboration, the NCCoE develops modular, easily adaptable example cybersecurity solutions demonstrating how to apply standards and best practices using commercially available technology.

LEARN MORE

Visit https://www.nccoe.nist.gov
nccoe@nist.gov
301-975-0200

NIST SPECIAL PUBLICATION 1800-16B

Securing Web Transactions

TLS Server Certificate Management

Volume B:
Security Risks and Recommended Best Practices

William Haag
Murugiah Souppaya
NIST

Paul Turner
Venafi

William C. Barker
Dakota Consulting

Brett Pleasant
Susan Symington
The MITRE Corporation

July 2019

DRAFT

This publication is available free of charge from:
https://www.nccoe.nist.gov/projects/building-blocks/tls-server-certificate-management

National Institute of Standards and Technology
U.S. Department of Commerce

NATIONAL CYBERSECURITY CENTER OF EXCELLENCE

DISCLAIMER

Certain commercial entities, equipment, products, or materials may be identified by name or company logo or other insignia in order to acknowledge their participation in this collaboration or to describe an experimental procedure or concept adequately. Such identification is not intended to imply special status or relationship with NIST or recommendation or endorsement by NIST or NCCoE; neither is it intended to imply that the entities, equipment, products, or materials are necessarily the best available for the purpose.

National Institute of Standards and Technology Special Publication 1800-16B, Natl. Inst. Stand. Technol. Spec. Publ. 1800-16B, 102 pages, (July 2019), CODEN: NSPUE2

FEEDBACK

You can improve this guide by contributing feedback. As you review and adopt this solution for your own organization, we ask you and your colleagues to share your experience and advice with us.

Comments on this publication may be submitted to tls-cert-mgmt-nccoe@nist.gov.

Public comment period: July 17, 2019 through September 13, 2019.

All comments are subject to release under the Freedom of Information Act.

National Cybersecurity Center of Excellence
National Institute of Standards and Technology
100 Bureau Drive
Mailstop 2002
Gaithersburg, MD 20899
Email: nccoe@nist.gov

NATIONAL CYBERSECURITY CENTER OF EXCELLENCE

The National Cybersecurity Center of Excellence (NCCoE), a part of the National Institute of Standards and Technology (NIST), is a collaborative hub where industry organizations, government agencies, and academic institutions work together to address businesses' most pressing cybersecurity issues. This public-private partnership enables the creation of practical cybersecurity solutions for specific industries, as well as for broad, cross-sector technology challenges. Through consortia under Cooperative Research and Development Agreements (CRADAs), including technology partners—from Fortune 50 market leaders to smaller companies specializing in information technology (IT) security—the NCCoE applies standards and best practices to develop modular, easily adaptable example cybersecurity solutions using commercially available technology. The NCCoE documents these example solutions in the NIST Special Publication 1800 series, which maps capabilities to the NIST Cybersecurity Framework and details the steps needed for another entity to recreate the example solution. The NCCoE was established in 2012 by NIST in partnership with the State of Maryland and Montgomery County, Maryland.

To learn more about the NCCoE, visit https://www.nccoe.nist.gov/. To learn more about NIST, visit https://www.nist.gov.

NIST CYBERSECURITY PRACTICE GUIDES

NIST Cybersecurity Practice Guides (Special Publication 1800 series) target specific cybersecurity challenges in the public and private sectors. They are practical, user-friendly guides that facilitate the adoption of standards-based approaches to cybersecurity. They show members of the information security community how to implement example solutions that help them align more easily with relevant standards and best practices, and provide users with the materials lists, configuration files, and other information they need to implement a similar approach.

The documents in this series describe example implementations of cybersecurity practices that businesses and other organizations may voluntarily adopt. These documents do not describe regulations or mandatory practices, nor do they carry statutory authority.

ABSTRACT

Transport Layer Security (TLS) server certificates are critical to the security of both internet-facing and private web services. A large- or medium-scale enterprise may have thousands or even tens of thousands of such certificates, each identifying a specific server in its environment. Despite the critical importance of these certificates, many organizations lack a formal TLS certificate management program and do not have the ability to centrally monitor and manage their certificates. Instead, certificate management tends to be spread across each of the different groups responsible for the various servers and systems in an organization. Central security teams struggle to make sure that certificates are being properly managed by each of these disparate groups. Where there is no central certificate management

service, the organization is at risk because once certificates are deployed, it is necessary to maintain current inventories to support regular monitoring and certificate maintenance. Organizations that do not properly manage their certificates face significant risks to their core operations, including

- application outages caused by expired TLS server certificates
- hidden intrusion, exfiltration, disclosure of sensitive data, or other attacks resulting from encrypted threats or server impersonation
- disaster-recovery risk that requires rapid replacement of large numbers of certificates and private keys in response to either certificate authority compromise or discovery of vulnerabilities in cryptographic algorithms or libraries

Despite the mission-critical nature of TLS server certificates, many organizations have not defined the clear policies, processes, roles, and responsibilities needed for effective certificate management. Moreover, many organizations do not leverage available automation tools to support effective management of the ever growing numbers of certificates. The consequence is continuing susceptibility to security incidents.

This NIST Cybersecurity Practice Guide shows large and medium enterprises how to employ a formal TLS certificate management program to address certificate-based risks and challenges. It describes the TLS certificate management challenges faced by organizations; provides recommended best practices for large-scale TLS server certificate management; describes an automated proof-of-concept implementation that demonstrates how to prevent, detect, and recover from certificate-related incidents; and provides a mapping of the demonstrated capabilities to the recommended best practices and to NIST security guidelines and frameworks.

This NIST Cybersecurity Practice Guide consists of the following volumes:

- **Volume A:** Executive Summary
- **Volume B:** Security Risks and Recommended Best Practices **(you are here)**
- **Volume C:** Approach, Architecture, and Security Characteristics
- **Volume D:** How-To Guides – instructions for building the example solution

KEYWORDS

Authentication; certificate; cryptography; identity; key; key management; PKI; private key; public key; public key infrastructure; server; signature; TLS; Transport Layer Security

ACKNOWLEDGMENTS

We are grateful to the following individuals for their generous contributions of expertise and time.

Name	Organization
Dean Coclin	DigiCert
Tim Hollebeek	DigiCert
Clint Wilson	DigiCert
Dung Lam	F5
Robert Smith	F5
Elaine Barker	NIST
Rob Clatterbuck	SafeNet Assured Technologies (SafeNet AT)
Jane Gilbert	SafeNet AT
Alexandros Kapasouris	Symantec
Mehwish Akram	The MITRE Corporation
Brian Johnson	The MITRE Corporation
Sarah Kinling	The MITRE Corporation
Bob Masucci	The MITRE Corporation
Susan Prince	The MITRE Corporation
Mary Raguso	The MITRE Corporation
Aaron Aubrecht	Venafi
Justin Hansen	Venafi

DOCUMENT CONVENTIONS

The terms “shall” and “shall not” indicate requirements to be followed strictly in order to conform to the publication and from which no deviation is permitted.

The terms “should” and “should not” indicate that among several possibilities one is recommended as particularly suitable, without mentioning or excluding others, or that a certain course of action is preferred but not necessarily required, or that (in the negative form) a certain possibility or course of action is discouraged but not prohibited.

The terms “may” and “need not” indicate a course of action permissible within the limits of the publication.

The terms “can” and “cannot” indicate a possibility and capability, whether material, physical or causal.

CALL FOR PATENT CLAIMS

This public review includes a call for information on essential patent claims (claims whose use would be required for compliance with the guidance or requirements in this Information Technology Laboratory (ITL) draft publication). Such guidance and/or requirements may be directly stated in this ITL Publication or by reference to another publication. This call also includes disclosure, where known, of the existence of pending U.S. or foreign patent applications relating to this ITL draft publication and of any relevant unexpired U.S. or foreign patents.

ITL may require from the patent holder, or a party authorized to make assurances on its behalf, in written or electronic form, either:

a) assurance in the form of a general disclaimer to the effect that such party does not hold and does not currently intend holding any essential patent claim(s); or

b) assurance that a license to such essential patent claim(s) will be made available to applicants desiring to utilize the license for the purpose of complying with the guidance or requirements in this ITL draft publication either:

i) under reasonable terms and conditions that are demonstrably free of any unfair discrimination; or

ii) without compensation and under reasonable terms and conditions that are demonstrably free of any unfair discrimination.

Such assurance shall indicate that the patent holder (or third party authorized to make assurances on its behalf) will include in any documents transferring ownership of patents subject to the assurance, provisions sufficient to ensure that the commitments in the assurance are binding on the transferee, and that the transferee will similarly include appropriate provisions in the event of future transfers with the goal of binding each successor-in-interest.

The assurance shall also indicate that it is intended to be binding on successors-in-interest regardless of whether such provisions are included in the relevant transfer documents.

Such statements should be addressed to: tls-cert-mgmt-nccoe@nist.gov

Contents

1 Introduction 1

1.1 Objective 1

1.2 Scope 1

2 TLS Server Certificate Background 2

2.1 Certificate Authorities 6

2.2 Certificate Request and Installation Process 9

3 TLS Server Certificate Risks 10

3.1 Outages Caused by Expired Certificates 10

3.2 Server Impersonation 12

3.3 Lack of Crypto-Agility 12

3.4 Encrypted Threats 13

4 Organizational Challenges 17

4.1 Certificate Owners 18

4.2 Certificate Services Team 19

5 Recommended Best Practices 19

5.1 Establishing TLS Server Certificate Policies 19

5.1.1 Inventory 20

5.1.2 Ownership 21

5.1.3 Approved CAs 22

5.1.4 Validity Periods 23

5.1.5 Key Length 24

5.1.6 Signing Algorithms 25

5.1.7 Subject DN and SAN Contents 25

5.1.8 Automation 26

5.1.9 Certificate Request Reviews – Registration Authority (RA) 27

5.1.10 Private Key Security 28

5.1.11 Rekey/Rotation upon Reassignment/Terminations 29

5.1.12 Proactive Certificate Renewal 29
5.1.13 Crypto-Agility 30
5.1.14 Revocation 31
5.1.15 Continuous Monitoring 32
5.1.16 Logging TLS Server Certificate Management Operations 32
5.1.17 TLS Traffic Monitoring 33
5.1.18 Certificate Authority Authorization 34
5.1.19 Certificate Transparency 34
5.1.20 CA Trust by Relying Parties 35
5.2 Establish a Certificate Service 35
5.2.1 CAs 36
5.2.2 Inventory 36
5.2.3 Discovery and Import 37
5.2.4 Management Interfaces 38
5.2.5 Automated Enrollment and Installation 39
5.2.6 RA/Approvals 39
5.2.7 Reporting and Analytics 40
5.2.8 Passive Decryption Support 40
5.2.9 Continuous Monitoring 40
5.2.10 Education 41
5.2.11 Help Desk 42
5.3 Terms of Service 43
5.4 Auditing 43
6 Implementing a Successful Program 43
Appendix A List of Acronyms and Abbreviations 46
Appendix B Glossary 49
Appendix C Mapping to the Cybersecurity Framework 59
Appendix D Special Publication 800-53 Controls Applicable to Best Practices for TLS Server Certificate Management 65

Appendix E References 100

List of Figures

Figure 2-1 TLS Certificates Are Broadly Used for Communications in Organizations 3
Figure 2-2 Server Address, Public Key, and Issuer Information on Four of the Organization's TLS Server Certificates 4
Figure 2-3 Upon Connecting to the Server, the Client Receives the Server's TLS Certificate, Which Includes the Server's Public Key 5
Figure 2-4 Browsers and Various Automated Processes (Web Servers, Containers, and IoT Devices) Connect as Clients to TLS Servers 6
Figure 2-5 A Public Root CA's Root Certificate Is Delivered to the User, Installed on a Software Vendor's Software 7
Figure 2-6 A Root CA Issues a Certificate to an Intermediate/Issuing CA, Which Issues TLS Server Certificates 7
Figure 2-7 Upon Connecting to the Server, the Client Receives Both the Server's TLS Certificate and Its CA Certificate Chain 8
Figure 2-8 Certificate Issuance Process 9
Figure 3-1 How an Attacker Leverages Encrypted Connections to Hide Attacks 14
Figure 3-2 Methods for Gaining Visibility into Encrypted Communications 16
Figure 4-1 TLS Certificates Are Distributed Broadly Across Enterprise Environments and Groups 18
Figure 5-1 Various Options for Automated Discovery and the Import of Certificates 38
Figure 5-2 Example Timeline of Processes and Notifications Triggered by Impending Certificate Expiration 41

List of Tables

Table 1 Mapping the Recommended Best Practices for TLS Server Certificate Management to the Cybersecurity Framework 59
Table 2 Application of Specific Controls to TLS Server Certificate Management Recommended Best Practices 65

1 Introduction

Organizations risk losing revenue, customers, and reputation, and exposing internal or customer data to attackers if they do not properly manage Transport Layer Security (TLS) server certificates. TLS is the most widely used security protocol to secure web transactions and other communications on the internet and internal networks. TLS server certificates are central to the security and operation of internet-facing and internal web services. Improper TLS server certificate management results in significant outages to web applications and services—such as government services, online banking, flight operations, and mission-critical services within an organization—and the risk of security breaches. Organizations should ensure that TLS server certificates are properly managed to avoid these issues.

The broad distribution of TLS server certificates across multiple groups and technologies within an enterprise requires that organizations establish formal management programs that include clear policies and responsibilities, a central Certificate Service, automation, and education. Successful implementation of a certificate management program relies on executive sponsorship, clear objectives, an action plan, and regular progress reviews.

1.1 Objective

The objective of this volume is to describe risks and challenges related to TLS server certificates and address those challenges by providing recommended best practices for large-scale TLS server certificate management. This document recommends that organizations establish a formal TLS certificate management program, and it enumerates elements that should be considered for inclusion in such a program. It is important to note that the best practices recommended in this guide are just that—recommendations.

1.2 Scope

The scope of this document is confined to recommendations regarding TLS server certificate management. TLS client certificate management is out of scope. This document is not intended to provide an extensive explanation of what TLS certificates and keys are or how they are used. Also, certificate management policies need to be considered within the context of an organization's overall enterprise security policies.

It is also beyond the scope of this document to discuss the broader aspects of organizational policies and procedures with which TLS server certificate management should be consistent. For example, general recommendations regarding security policy, vulnerability management, incident response, disaster recovery, security testing, etc. that are not specifically related to certificate management are out of scope. Discussion of general security protections for certificate management system components is also beyond the scope of this document. This document assumes the security of these components is protected by recommended security best practices, e.g., patching, strong authentication, and access control that the organization has in place as part of its overall security policy.

An organization's business operations may be internally or externally supported. For those organizations that have third parties supporting key business operations, those third parties may use TLS certificates. If a function is outsourced, the organization should ensure that its requirements are met by the third party performing the function. The TLS certificate management recommendations provided in this document can be applied to these third parties as well as to the organization itself.

In accordance with their security policies, some organizations may choose to perform inspection of internal traffic that has been encrypted using TLS, by intercepting and decrypting TLS traffic at the network edge or by performing passive decryption at locations deeper within the network. The question of whether to perform such inspection is complex, and it involves important tradeoffs between traffic security and traffic visibility that organizations should weigh carefully. It is beyond the scope of this document to advocate for or against TLS traffic inspection. Some organizations have determined that the security risks posed by inspection of internal TLS traffic are not worth the potential benefits of having visibility into the encrypted traffic. Other organizations, however, have determined that it is in their best interests to perform TLS traffic inspection. For those organizations that have a policy of performing TLS traffic inspection, this document provides recommended best practices regarding how to securely manage the TLS private keys required for this purpose.

The security and integrity of TLS relies on secure implementation and configuration of TLS servers and effective TLS server certificate management. Guidance regarding the implementation and configuration of TLS servers is outside the scope of this document. The secure implementation and configuration of TLS servers is addressed in NIST *Special Publication 800-52*. Organizations should provide clear instruction to groups and individuals deploying TLS servers in their environments to read, understand, and follow the guidance provided in 800-52.

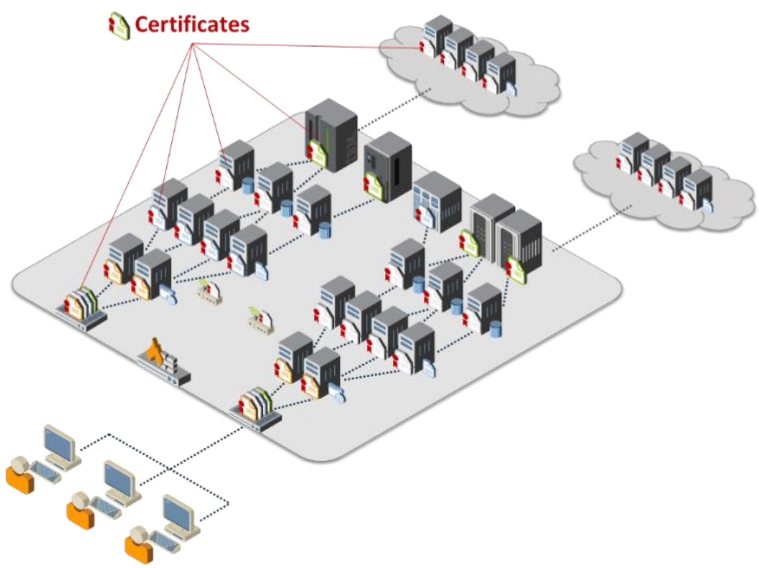
Lastly, the recommendations included in this document are generic. Each organization should determine for itself how to best apply these recommendations to its own enterprise. Volumes C and D of this Practice Guide describe a specific implementation used to demonstrate the application of these recommendations.

2 TLS Server Certificate Background

TLS is the security protocol used to authenticate and protect internet and internal network communications for a broad number of other protocols—including Hypertext Transfer Protocol (http) for web servers; Lightweight Directory Access Protocol (LDAP) for directory servers; and Simple Mail Transfer Protocol, Post Office Protocol, and Internet Message Access Protocol for email.

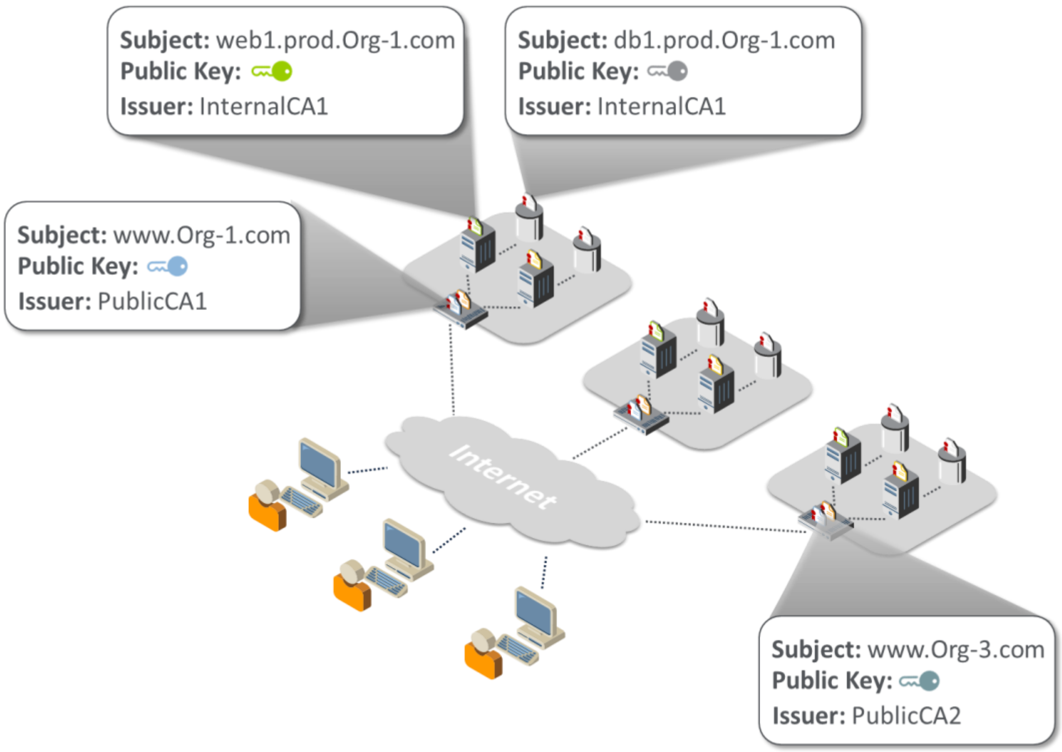
TLS server certificates serve as machine identities that enable clients to authenticate servers via cryptographic means. For example, when a bank customer connects across the internet to an online banking website, the customer's browser (i.e., the TLS client) will present an error message if the server does not provide a valid certificate that matches the address the user entered in the browser. Further, TLS server certificates are used extensively inside corporate and government networks to establish trust between machines — servers, applications, devices, micro-services, etc. Most enterprises have thousands of certificates, each identifying a specific server in their environment. (Note: Web browsers play the role of clients to web servers. As such, they contain functionality to automatically establish TLS connections on behalf of users, evaluate certificates received during the TLS handshake process, and present errors when unexpected certificate issues are encountered.) Figure 2-1 illustrates the pervasive use of certificates within organizations.

Figure 2-1 TLS Certificates Are Broadly Used for Communications in Organizations



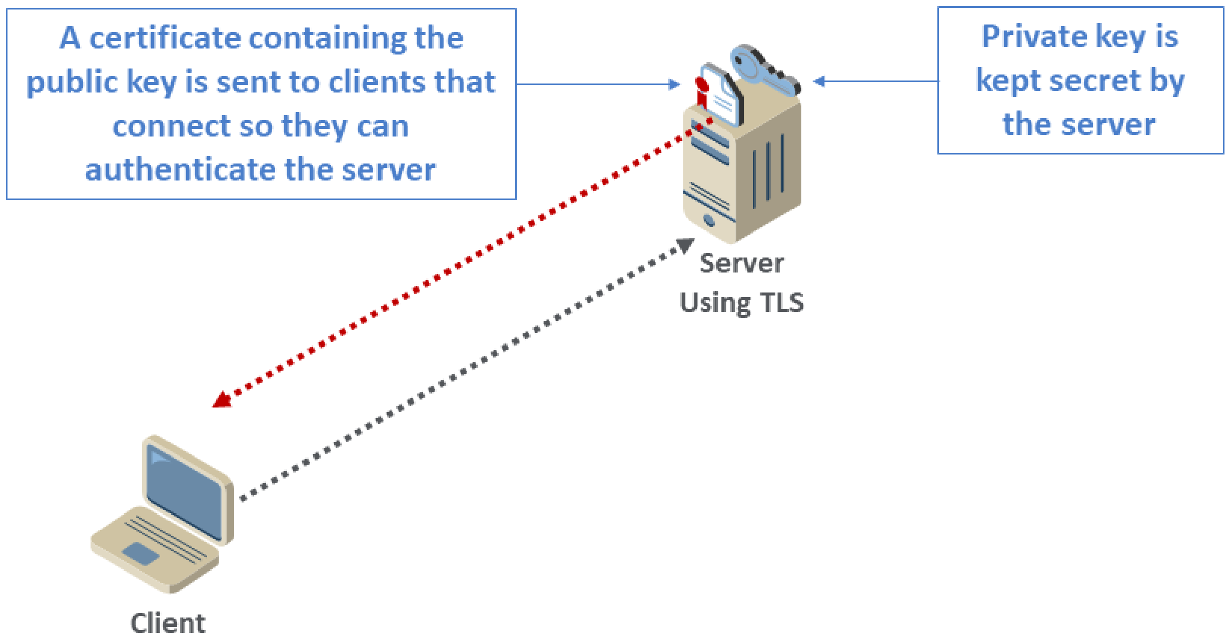
Each TLS server certificate contains the address of the server that it identifies (e.g., *www.organization1.com*) and a cryptographic key, called a public key, which is unique to the server and used by clients to securely authenticate to the server (see Figure 2-2).

Figure 2-2 Server Address, Public Key, and Issuer Information on Four of the Organization's TLS Server Certificates



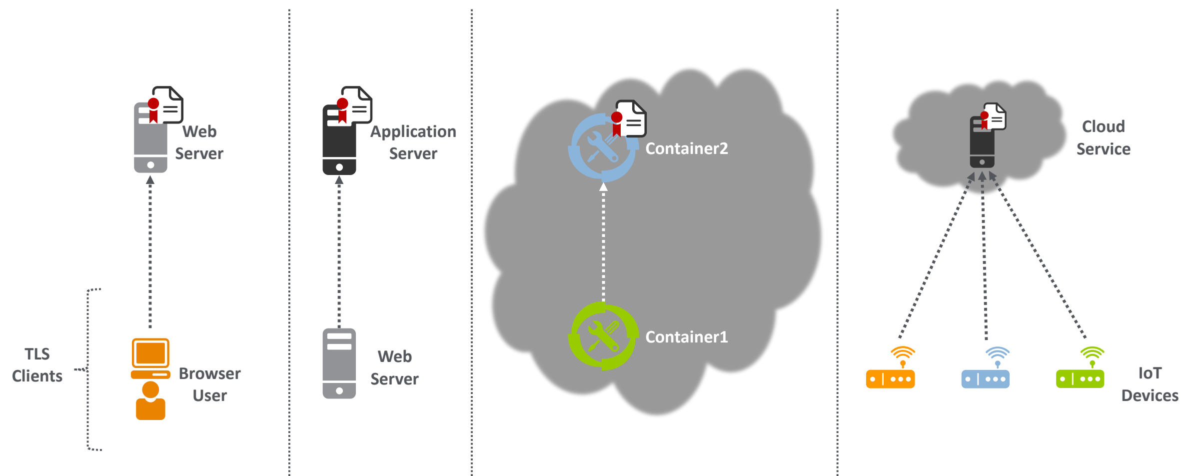
As shown in Figure 2-3, each server holds a private key that corresponds to the public key in the certificate so each server can prove it is the holder of the certificate. While the certificate is shared with any client that connects to the server, the private key must be kept secure and secret so it cannot be obtained by an attacker and used to impersonate the server. Many private keys used with TLS are stored in plaintext files on TLS servers. Alternatively, private keys can be stored in files encrypted with a password; however, the passwords are generally stored in plaintext configuration files so they are accessible by the TLS server software when it is started. These common practices make it possible for private keys to be viewed and copied by system administrators or malicious actors.

Figure 2-3 Upon Connecting to the Server, the Client Receives the Server's TLS Certificate, Which Includes the Server's Public Key



In addition to users with browsers connecting to servers that have TLS server certificates, automated processes also connect as clients to TLS servers and must trust TLS server certificates. Examples of automated processes acting as TLS clients include a web server making requests to an application server, one cloud container connecting to another, or an Internet of Things (IoT) device connecting to a cloud service. (See Figure 2-4.)

Figure 2-4 Browsers and Various Automated Processes (Web Servers, Containers, and IoT Devices) Connect as Clients to TLS Servers

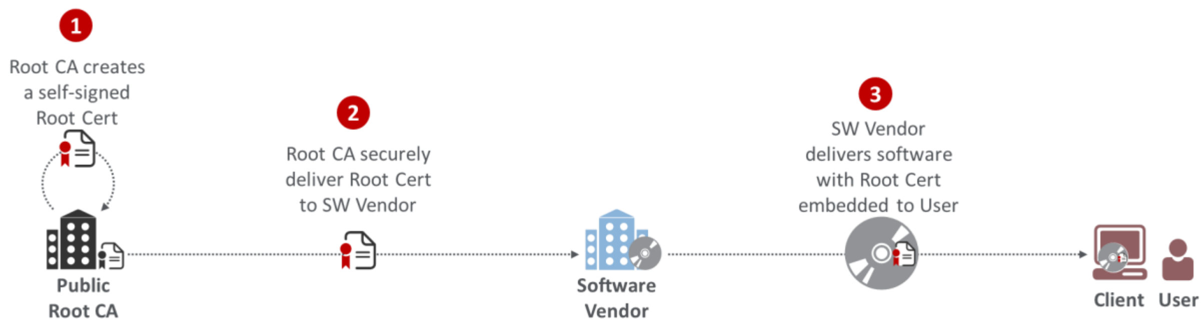


2.1 Certificate Authorities

TLS server certificates are issued by entities called certificate authorities (CAs). CAs digitally sign certificates so that their authenticity can be validated — to prevent attackers from easily impersonating servers. Clients (e.g., browsers, devices, applications, services) validate certificates by using a CA's certificate to verify the signature. Clients, such as browsers, are configured to trust specific CAs (called root CAs). This is done by installing a CA's certificate, commonly called a root certificate, on the client.

Some CAs arrange for their root certificate to get installed by software manufacturers in their software (e.g., browser, application, or operating system) so the certificates issued by the CAs are trusted broadly. These CAs are commonly called public root CAs. (See Figure 2-5.)

Figure 2-5 A Public Root CA's Root Certificate Is Delivered to the User, Installed on a Software Vendor's Software

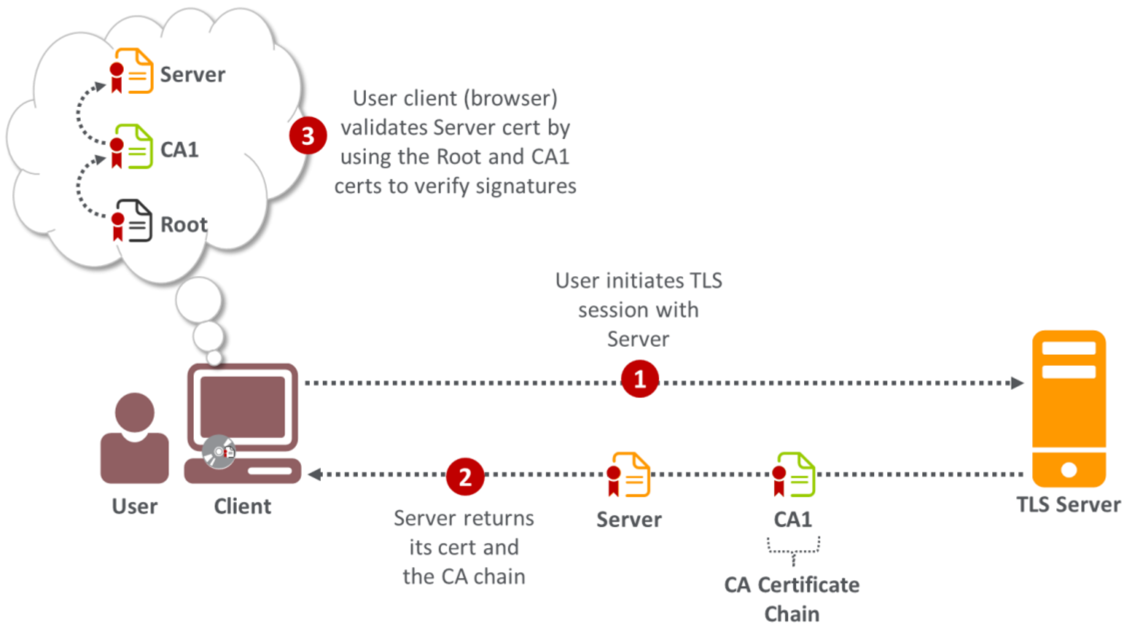


To protect them from attacks, root CAs are generally not connected to the internet and do not issue TLS server certificates directly. Root CAs certify other CAs, generally called intermediate or issuing CAs, which issue TLS server certificates. (See Figure 2-6.)

Figure 2-6 A Root CA Issues a Certificate to an Intermediate/Issuing CA, Which Issues TLS Server Certificates

As shown in Figure 2-7, when a client, such as a browser, connects to a TLS server, the server will return its certificate as well as the certificate for the CA that issued its certificate (called the CA certificate chain).

Figure 2-7 Upon Connecting to the Server, the Client Receives Both the Server's TLS Certificate and Its CA Certificate Chain



Public CAs are regularly audited to ensure they operate in compliance with the [CA/Browser Forum Baseline Requirements](), which are standards intended to minimize the possibility of CA compromises and fraudulent certificates. When CAs have been found to violate the requirements, their root certificates have been removed from and distrusted by browsers, requiring customers of those CAs to rapidly replace their TLS server certificates.

There are three different types of certificates issued by public CAs (as specified by the CA/Browser Forum, which defines standards for public CAs), each with a different level of validation required by the CA to confirm the identity of the requester and its authority to receive a certificate for the domain in question:

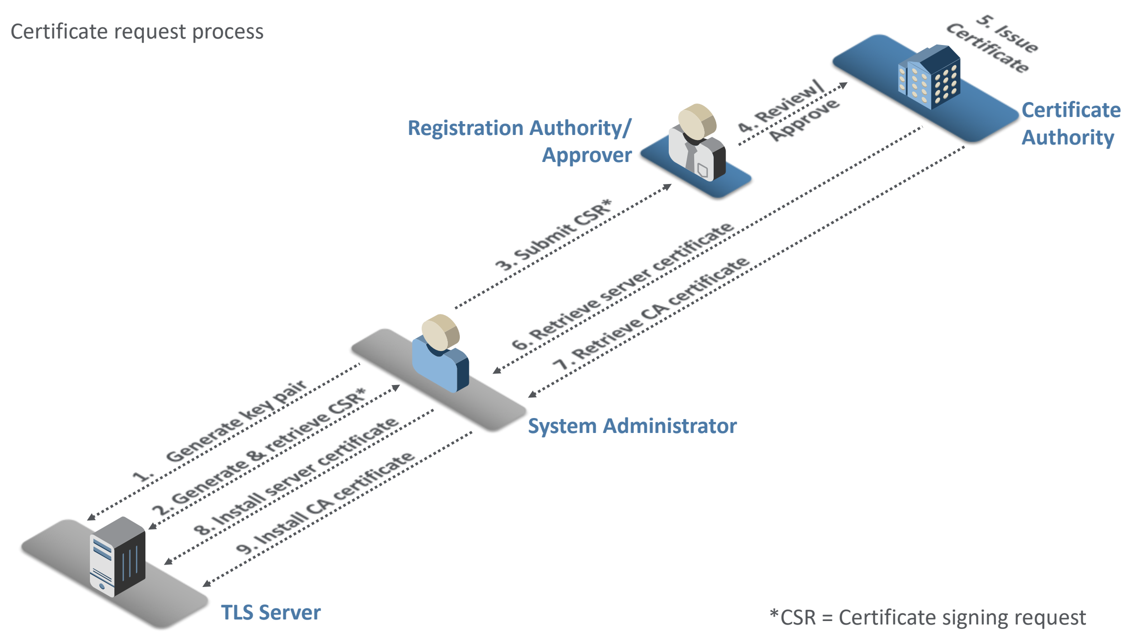
- Domain Validated (DV): The CA validates that the requester is the owner of the domain, by verifying that the requester can reply to an email address associated with the domain, has operational control of the website at the domain address, or is able to make modifications to the Domain Name System (DNS) record for the domain
- Organization Validated (OV): In addition to the checks for DV certificates, the CA conducts additional vetting of the requester's organization
- Extended Validation (EV): EV certificates undergo the most rigorous checks, including verifying the identity and the legal, physical, and operational existence of the entity requesting the certificate, by using official records

Organizations that wish to issue certificates to their internal TLS servers can establish their own CAs, commonly called internal CAs. Organizations using internal CAs must ensure that all clients connecting to their servers trust the internal CAs by installing the internal CAs' root certificates on each system acting as a client (e.g., browsers, operating systems, applications, appliances).

2.2 Certificate Request and Installation Process

The following steps, shown in Figure 2-8 and detailed below, are typically followed by a system administrator to get a TLS certificate for a server that he or she manages.

Figure 2-8 Certificate Issuance Process



1. The system administrator for the TLS server uses utilities on the server to generate a cryptographic key pair (a public key and a private key).

2. The system administrator enters the address of the server (e.g., *www.organization1.com*). The utilities create a request for a certificate, called a certificate signing request (CSR), which contains the address of the server and the public key. The system administrator retrieves a copy of the CSR (which is contained in a file) from the server.

3. The system administrator submits the CSR to the registration authority (RA), who acts as a reviewer and approver of the certificate request.
4. The RA/approver reviews the CSR, performs necessary checks to confirm the validity of the request and the authority of the requester, and then sends an approval to the CA.
5. The CA issues the certificate.
6. The CA notifies the system administrator that the certificate is ready, either by emailing a copy of the certificate or providing a link from which it can be downloaded. The system administrator retrieves the server certificate.
7. The system administrator retrieves the CA certificate chain from the CA.
8. The system administrator installs the server certificate on the server.
9. The system administrator installs the CA certificate chain on the server.

The CA certificate chain is used by TLS clients to validate the signature on the server certificate. When a client connects to a TLS server, the server returns its certificate and the CA certificate chain, which can contain one or more CA certificates. The client starts with one of its locally trusted root CA certificates and successively validates the signatures on certificates in the CA certificate chain until it reaches the server certificate.

The system administrator must note the expiration date in the certificate to ensure that a new certificate is requested and installed before the existing certificate expires.

3 TLS Server Certificate Risks

When TLS server certificates are not properly managed, organizations risk negative impacts to their revenue, customers, and reputation. There are four primary types of negative incidents that result from certificate mismanagement: outages to important business applications, caused by expired certificates; security breaches resulting from server impersonation; outages or security breaches resulting from a lack of crypto-agility; and increased vulnerability to attack via encrypted threats.

3.1 Outages Caused by Expired Certificates

TLS server certificates contain an expiration date to ensure that the cryptographic keys are changed regularly; this reduces the possibility of a security breach caused by a compromised private key. If a server certificate is not changed before its expiration date, then clients should generate an error message and stop the connection process to the server. This causes the application supported by the server with the expired certificate to become unavailable.

Application outages can also be caused by the mismanagement of CA certificate chains that results in expired intermediate CA certificates. The TLS server is responsible for providing the client with the intermediate CA certificates (CA certificate chain) necessary for the client to link the server's end-entity certificate with the root CA certificate trusted by the client. The absence or expiration of an intermediate certificate means the client will not trust the server, even though the server may have a perfectly trustworthy end-entity certificate. Intermediate CA certificates are typically renewed every few years, and it is possible for a TLS server to fail to use the most current version. As a result, although the server certificate has been updated, the installed intermediate CA certificate may expire, resulting in an outage due to expiration. Such outages are often difficult to diagnose because the focus of investigation is typically on the server certificate, which is still valid and not the cause of the outage.

Nearly every enterprise has experienced an application outage due to an expired certificate, including outages to major applications such as online banking, stock trading, health records access, and flight operations. Organizations' increased use of TLS server certificates to secure the organizations' applications increases the likelihood of outages, because there are more certificates to track and more certificates per business application that can impact operations.

Various scenarios result in a certificate expiring while still in use, causing an outage, including these:

- The system administrator forgets about the certificate
- The system administrator ignores notifications that the certificate will soon expire
- The system administrator does not properly install or update the CA certificate chain
- The system administrator is reassigned, and nobody else receives expiry notifications
- The system administrator enrolls for a new certificate but does not install it on the server(s) in time or installs it incorrectly
- The application relies on multiple load-balanced servers, and the certificate is not updated on all of them
- The certificate is installed on a backup system, but the certificate has expired before the backup system is brought online

Troubleshooting an incident where an application is unavailable due to an expired certificate can be complex and often requires hours to discover the source of the problem. If the server on which an expired certificate is deployed is being accessed by people using browsers, then each of those people will receive an error message, making it clear that the cause of the issue is an expired certificate. If, on the other hand, the server with the expired certificate is an application server receiving requests from a web server, then the web server stops its operations and may log a message, but that message may not be immediately discovered in the log file, increasing the amount of time required to identify the root cause of the outage and fix it. If certificates that are deployed on backup systems are not updated when they expire, an outage can occur if operations are shifted to the backup systems.

3.2 Server Impersonation

An attacker may be able to impersonate a legitimate TLS server (e.g., a banking website) if the attacker is able to get a fraudulent certificate containing the address of the server and the attacker's own public key by tricking a trusted CA into issuing the certificate to the attacker or by compromising the CA and issuing the certificate. A client connecting to the attacker's server will accept the certificate because the certificate contains the address to which the client intended to connect and because the certificate has been issued by a trusted CA. Because the certificate contains the attacker's public key (and the attacker also holds the private key corresponding to this public key), the attacker can decrypt the communications from the client (including passwords intended for login to the legitimate server). Alternatively, if the attacker can access a copy of the legitimate server's private key, then the attacker can also impersonate that server by using the legitimate server's certificate. To successfully perform these attacks, the attacker must redirect traffic destined for the legitimate server to a system that the attacker is operating (e.g., using Border Gateway Protocol [BGP] hijacking or DNS compromise). (Note: The BGP is used to communicate optimal routes between internet service providers on the internet. It is possible for an attacker to hijack traffic by falsely advertising that the fastest route to one or more internet protocol [IP] addresses is via systems that the attacker is operating, thereby causing traffic to be rerouted through the attacker's systems. The DNS provides translation between human-readable addresses [e.g., *www.company123.com*] and IP addresses. If an attacker can compromise an organization's DNS account, then the attacker can change the IP address to which traffic intended for that organization will be sent.)

Most private keys used on TLS servers are stored in files. The private keys are directly managed and handled by system administrators, who can make copies of the private keys. In addition, many TLS servers are clustered (for load balancing); in many cases, the same TLS server certificate and the private key will be copied to each server in the cluster. The manual handling and copying of private keys significantly increase the possibility of a key compromise.

3.3 Lack of Crypto-Agility

There are several types of incidents that have required organizations to replace large numbers of TLS certificates and private keys, including the following:

- **CA compromise**: If a CA is breached by an attacker, then the attacker can cause that CA to issue fraudulent certificates. After the CA breach is discovered and forensics are performed, it may be concluded that certificates issued by the CA cannot be trusted and that new certificates must be installed on all servers with certificates from the compromised CA.
- **Vulnerable algorithm**: Cryptographic algorithms are constantly evaluated for vulnerabilities, by parties with both positive and negative intent. When an algorithm is found to be vulnerable (e.g., Secure Hash Algorithm 1 [SHA-1] for signature generation), TLS server certificates that are dependent on the algorithm must be replaced. Ongoing advancements in quantum computing require that organizations establish the ability to rapidly replace all existing certificates and keys and be prepared for implementation of post-quantum algorithms.

- **Cryptographic library bug**: Because cryptographic operations are quite complex, a few groups have specialized in developing cryptographic libraries that are used by TLS servers and other systems. If a bug is found with the key-generation functions of a cryptographic library, then all keys generated since the bug was introduced must be replaced. (Note: In 2008, a key-generation bug in the cryptographic libraries in Debian Linux was discovered. That bug was introduced in 2006. In 2017, a key-generation bug was discovered in the Infineon cryptographic libraries used in smart cards and trusted platform module chips.)

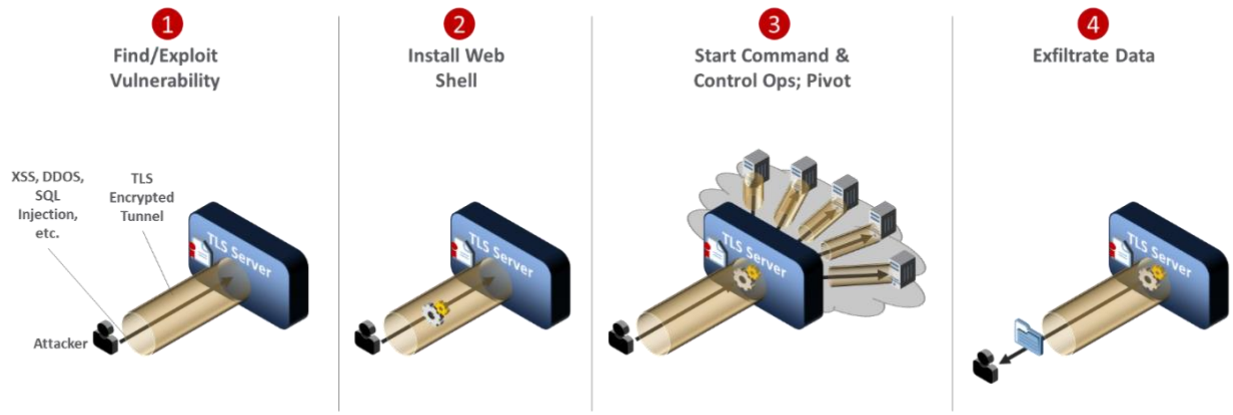
Most enterprises are not prepared to respond to the large-scale cryptographic failure that results from these types of incidents. Many organizations do not have comprehensive inventories of their TLS server certificates. In addition, they cannot contact the certificate owners, because they do not have up-to-date information about the certificate owners responsible for each certificate. Finally, many organizations rely on manual processes to manage certificates and do not have processes for tracking the progress in replacing large numbers of certificates — leaving the organizations to guess how many systems have been updated. All these factors can result in organizations requiring several weeks or months to replace all affected certificates, during which time business applications can be unavailable or vulnerable to security breaches.

3.4 Encrypted Threats

Many organizations are working to encrypt all communications by using TLS server certificates to prevent interception of plaintext credentials and eavesdropping on communications. While TLS server certificates enable confidentiality for legitimate communications, they can also allow attackers to hide their malicious activities within encrypted TLS connections. When a TLS server certificate is installed and enabled on a server, all users who connect (including attackers) can establish an encrypted connection to the server. An attacker who establishes an encrypted connection can then begin to probe the server for vulnerabilities within that encrypted connection.

The following steps, shown in Figure 3- and detailed below, describe how an attacker can leverage encrypted connections in his or her attacks.

Figure 3-1 How an Attacker Leverages Encrypted Connections to Hide Attacks



1. The attacker begins by connecting to a server and establishing an encrypted TLS session. Within that encrypted session, the attacker can probe for vulnerabilities that exist on the server and its software
2. If the attacker discovers a vulnerability and sufficiently elevates his or her privileges, then the attacker can load malware, generally called a “web shell,” onto the server
3. With this web shell loaded, the attacker can send commands over TLS connections (i.e., encrypted connections facilitated by the server’s certificate). The attacker can then work to pivot to other systems by probing for vulnerabilities in servers accessible from the compromised system. The increased use of encryption enables an attacker who has compromised one system to pivot and attack other systems via encrypted connections, without being detected
4. Once the attacker has successfully reached data that he or she desires, the attacker is able to use the web shell to exfiltrate data. Because the attacker is establishing TLS connections by using the server’s certificate to connect to the web shell, all the exfiltrated data is encrypted while in transit

As stated in Section 1.2, in accordance with their security policies, some organizations may choose to perform inspection of internal traffic that has been encrypted using TLS. The question of whether to perform such inspection is complex, and it involves important tradeoffs between traffic security and traffic visibility that each organization should weigh for itself.

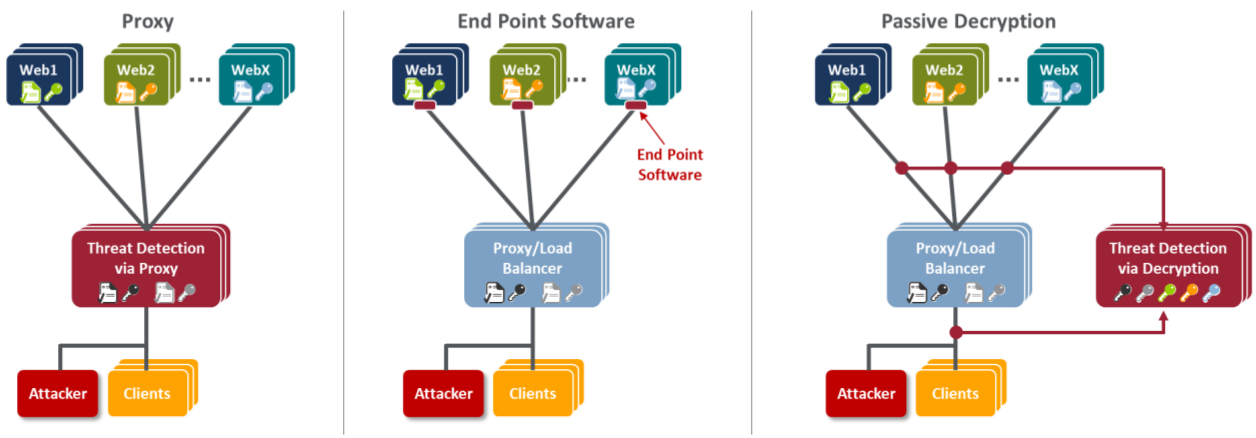
Some organizations are concerned about the risk posed by attackers who leverage encrypted connections to hide their attacks, as illustrated in Figure 3-1 above. If these attackers gain access to trusted internal systems via malware or some other exploit, they may be able to move about the network without being detected by hiding their traffic within TLS connections. Organizations that are concerned about these risks want the option of decrypting internal TLS traffic so it can be inspected. Such inspection may be used not only for intrusion and malware detection, but also for troubleshooting, fraud detection, forensics, and performance monitoring. These organizations have concluded that the visibility into their internal traffic that can be provided by TLS inspection is worth the tradeoff of the weaker encryption and other risks that come with such inspection. For these organization, TLS inspection may be considered standard practice and may represent a critical component of their threat detection and service assurance strategies. Some of these organizations have complex networks that are several tiers deep, so it would not be realistic to expect them to be able to manage the movement of keys required to perform such inspection securely using purely manual processes. For those organizations that have a policy to perform inspection of TLS traffic, this document provides recommendations regarding how to securely move the TLS private keys needed for this inspection.

On the other hand, inspection creates a single location where traffic may be decrypted, creating an attractive target for hackers. It also may have compliance implications if sensitive data is being decrypted. An organization that performs decryption on border devices or that performs passive internal decryption runs the risk of such devices being taken over by a malicious attacker who would then have access to private keys and traffic. In addition, passive decryption requires the use of static key exchange, which results in weaker encryption than can be achieved when using ephemeral key exchange methods. If an attacker captures a server’s private key and that key was negotiated using static key exchange, the attacker will also be able to decrypt traffic that had been captured in the past. If, instead, that key was negotiated using an ephemeral key exchange method, the key will provide forward secrecy, meaning the attacker will not be able to decrypt past traffic. For some organizations, the reduced security of performing inspection or using static keys is unacceptable. These organizations have determined that the security risks posed by inspection of internal TLS traffic are not worth the potential benefits of having visibility into the encrypted traffic. These organizations should have a policy against performing TLS inspection. As an alternative to inspection, they may choose to perform traffic analysis to try to detect illegitimate internal TLS traffic. None of the discussion or recommendations in this document are intended to mandate or encourage an organization to begin performing TLS inspection of its traffic if that organization has determined that the risks of TLS inspection are not worth the benefits.

An organization that has a policy to perform inspection of TLS traffic so it can monitor and detect malicious activity has several methods it can use to gain visibility into encrypted communications. Some examples are listed below and are illustrated in Figure 3-2:

- placing a threat detection system that acts as a reverse proxy in front of servers
- installing end point software on each server to monitor communications
- passively decrypting communications

Figure 3-2 Methods for Gaining Visibility into Encrypted Communications



The use of threat detection proxies is ideal at the perimeters of organizations for monitoring inbound internet communications for attacks. The threat detection proxy is connected in-line, requiring all inbound traffic to pass through it before moving on to the next device. The threat detection proxy terminates the TLS connection. It decrypts and examines incoming traffic. If the traffic is determined to be malicious, the proxy drops it. Because the threat detection proxy is terminating all TLS connections, it must have a certificate for each server to which clients are attempting to connect. After the threat detection proxy decrypts and examines the traffic, it can establish a TLS session with the appropriate server behind it and send the traffic to that server in an encrypted TLS session.

While a threat detection proxy is ideal for use at the perimeter of an organization, many organizations also want to inspect their internal TLS traffic. Many enterprise applications include multiple tiers of servers and services (e.g., load balancers, web servers, application servers, databases, identity services) that communicate with each other internally via encrypted TLS sessions, making it impractical to place threat detection proxies between all systems on internal networks.

End point software can be installed on each server to monitor communications, alleviating the need to install proxies, but may impose additional processing requirements on servers that are already under a high load. In addition, because of the diversity of TLS server systems, it may be difficult to find an end point solution that operates on all platforms and provides comprehensive and consistent visibility and monitoring of all communications.

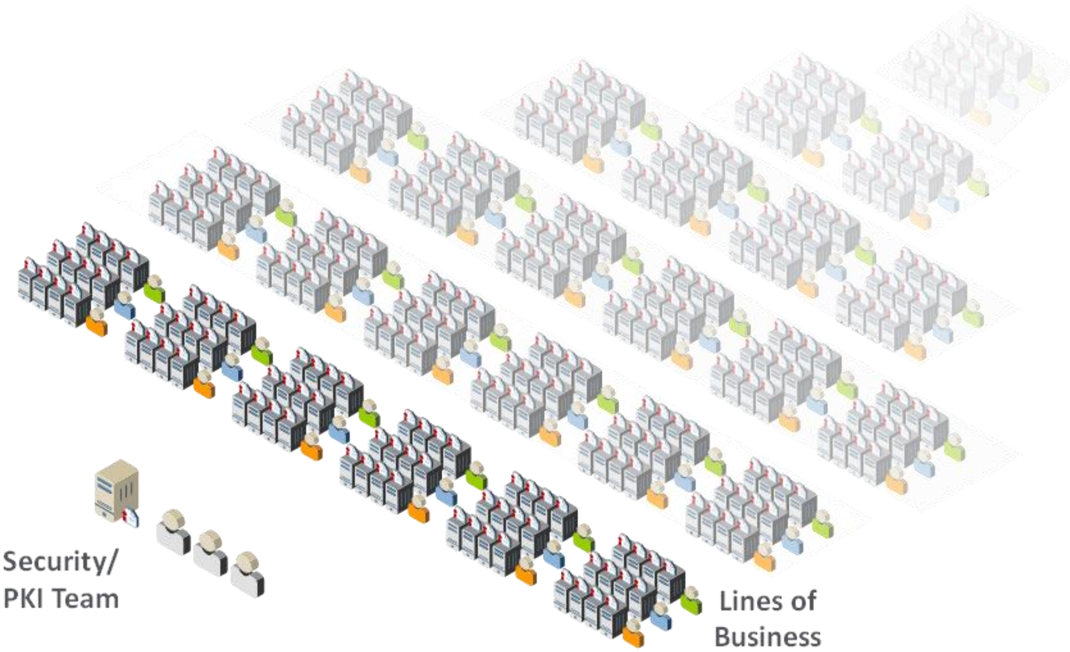
Passive, out-of-band decryption and threat analysis are performed by using devices that decrypt TLS-encrypted communications but that do not terminate TLS connections. The TLS connection is established between the client and the server. The passive decryption device listens to the TLS traffic without affecting it and decrypts it. Threat analysis is performed either by the passive decryption device or via other systems to which decrypted traffic is forwarded. Security-focused passive decryption devices can detect malicious traffic that has been sent on TLS connections, but these devices do not react in real time to block this traffic. Passive decryption does not require a change in network architecture or loading additional software on TLS servers. However, passive decryption poses a TLS server certificate management challenge, because private keys must be copied to decryption devices from each TLS server whose communications will be monitored. The transfer of private keys must be done securely to avoid a key compromise and rapidly to avoid blind spots in monitoring for attacks. Automation can significantly aid in securely transferring private keys from TLS servers to the decryption device and keeping keys up-to-date when certificates are replaced.

4 Organizational Challenges

Despite the mission-critical nature of TLS server certificates, many organizations do not have clear policies, processes, and roles and responsibilities defined to ensure effective certificate management. Moreover, many organizations do not leverage available technology and automation to effectively manage the large and growing number of TLS server certificates. As a result, many organizations continue to experience significant incidents related to TLS server certificates.

As illustrated by Figure 4-1, the management of TLS server certificates is challenging due to the broad distribution of certificates across enterprise environments and groups, the complex processes needed to manage certificates, the multiple roles involved in certificate management and issuance, and the speed at which new TLS servers are being deployed. TLS server certificates are typically issued by a Certificate Services team (often called the public key infrastructure team). However, the certificates are commonly installed and managed by the certificate owners — the groups and the system administrators responsible for individual web servers, application servers, network appliances, and other devices for which certificates are used.

Figure 4-1 TLS Certificates Are Distributed Broadly Across Enterprise Environments and Groups



4.1 Certificate Owners

The term “certificate owner” is used to denote a group responsible for systems where certificates are deployed. Typically, there are several roles within a certificate owner group, including executives who have ultimate accountability for ensuring that certificate-related responsibilities are addressed, system administrators who are responsible for managing individual systems and the certificates on them, and application owners who can review and approve certificate requests from system administrators to ensure that only authorized certificates are issued. The certificate owners typically are not knowledgeable about the risks associated with certificates or the best practices for effectively managing certificates.

With the advent of virtualization, the development and operations (DevOps) teams provision systems and software through programmatic means. This introduces a new type of certificate owner and new TLS server certificate challenges for organizations. As organizations push for more rapid and efficient deployment of business applications, many DevOps teams deploy certificates without coordination with the Certificate Services team. This can result in certificates for mission-critical applications not being tracked. This can be particularly problematic if bugs in DevOps programs/scripts cause certificates to be improperly deployed or updated. In addition, as DevOps teams adopt newer frameworks and tools, it is important to continue to monitor certificates and applications deployed and maintained by older DevOps frameworks and tools.

4.2 Certificate Services Team

The Certificate Services team is typically the group that has been given responsibility for managing relationships with public CAs and for the internal CAs. The Certificate Services team typically comprises one to three people. Though the team members have good knowledge and expertise about TLS server certificates, they do not have the resources or access required to directly manage certificates on the extensive number of systems where certificates are deployed. However, the Certificate Services team is often blamed when TLS certificate incidents, such as outages, occur.

5 Recommended Best Practices

To effectively address the risks and organizational challenges related to TLS server certificates and to ensure that they are a security asset instead of a liability, organizations should establish a formal TLS certificate management program with executive leadership, guidance, and support. The formal TLS certificate management program should include clearly defined policies, processes, and roles and responsibilities for the certificate owners and the Certificate Services team, as well as a central Certificate Service. The program should be driven by the Certificate Services team but should include active participation by the certificate owners — whether the certificate owners are responsible for traditional servers, appliances, virtual machines, cloud-based applications, DevOps, or other systems acting as TLS servers.

5.1 Establishing TLS Server Certificate Policies

As previously mentioned, most certificate owners are typically not knowledgeable about the best practices for effectively managing TLS server certificates. Because certificate owners are responsible for the systems where certificates are deployed, it is imperative that they be provided with clear requirements and that those requirements be enforced as policies. This section provides recommended TLS server certificate policies. It also includes recommended responsibilities for the certificate owners and the Certificate Services team to successfully meet those requirements and policies.

These recommendations are intended to serve as guidance for organizations that do not already have their own TLS server certificate management policies and responsibilities defined, or that are looking to improve existing policies and procedures. They are not intended to override any organization's existing policies. Organizations should feel free to copy, delete, augment, or modify these recommended policies and responsibilities as needed to suit their own requirements. Appendix B contains a table that maps the recommended best practices for TLS server certificate management proposed in this document to the NIST *Framework for Improving Critical Infrastructure Cybersecurity* (Cybersecurity Framework—CSF). Appendix C contains a table that explains how specific controls defined within NIST Special Publication 800-53 should be applied to these TLS server certificate management recommended best practices.

The recommended requirements in the remaining subsections use the word “should” throughout. Based on their own security policies, organizations may choose to make these recommendations mandatory, e.g., by changing “should” to “must.”

5.1.1 Inventory

To address TLS server certificate risks, organizations should establish and maintain clear visibility across all TLS server certificates in their environment so they can perform the following actions:

- detect potential vulnerabilities (e.g., the use of weak algorithms, such as SHA-1)
- identify certificates that are nearing expiration and replace them
- respond to large-scale cryptographic incidents, such as a CA compromise, vulnerable algorithms, and cryptographic library bugs
- ensure compliance with regulatory guidelines and established organizational policy

This visibility is achieved by maintaining an inventory of all TLS server certificates. A single central inventory is recommended, as it minimizes the possibility of overlooking critical TLS server certificates.

Recommended Requirement:

An up-to-date inventory of all deployed certificates (end-entity certificates and CA certificate chain certificates) should be maintained, including certificates on backup systems that may not necessarily be online. For each certificate, the inventory should include the following components:

- Subject Distinguished Name (DN)
- Subject Alternative Names (SANs)
- issue date (i.e., notBefore date)
- expiration date (i.e., notAfter date)
- issuing Certificate Authority (CA)
- key length
- key algorithm (e.g., Rivest, Shamir, & Adleman [RSA]; Elliptic Curve Digital Signature Algorithm [ECDSA])
- signing algorithm
- validity period (i.e., from the notBefore date/time to the notAfter date/time)
- installed location(s) of certificate (e.g., IP or DNS address and file path)
- certificate owner (i.e., the group responsible for the certificate)

- group responsible for the DevOps technology used to deploy the certificate (if the certificate was deployed via DevOps technology)
- contacts (i.e., the group of individuals that should be notified of issues)
- approver(s) (i.e., the parties responsible for reviewing issuance and renewal requests)
- type of system (e.g., web, email, directory server, appliance, virtual machine, container)
- business application (i.e., the application using the certificate)
- applicable regulations (e.g., Payment Card Industry Data Security Standard [PCI-DSS], Health Insurance Portability and Accountability Act [HIPAA])
- key-usage flags
- extended key-usage flags

Recommended Responsibilities:

- Certificate Services team: provide a central system for certificate owners to establish and maintain their inventories
- Certificate owners: establish and maintain an inventory of all certificates and keys on their systems

5.1.2 Ownership

To rapidly respond to issues with TLS server certificates, it is necessary to know who is responsible for each certificate. This information should be kept up-to-date as people are reassigned or terminated. Because reassignments can happen frequently, and because there may be a lag in updating ownership information, it is recommended that ownership be assigned to functional groups (e.g., an Active Directory [AD] group) that contain multiple individuals, instead of assigning ownership to individuals. In cases where DevOps technologies are used to deploy TLS server certificates, the group responsible for the DevOps deployment technology should be tracked, in addition to the certificate owner, so they can both be contacted when incidents arise.

Recommended Requirement:

- Contact information for certificate owners should be assigned to functional groups (e.g., AD groups), and the content of a group should be updated within <30> business days of a role reassignment or termination of an individual member of that group. (Note: Here and elsewhere in this practice guide, when specific time frames, such as “<30> business days” are recommended, these values are often placed within brackets (“<>”) to indicate they are provided only as suggestions. Each organization should determine the time frames to be instituted within its own enterprise, based on its needs. If it is possible for organizations to require compliance within shorter time frames, then that would be preferable.)

- If the certificate was deployed via DevOps technology, contact information should be provided for the group that is responsible for this technology, and the content of this group should be updated within <30> business days of a role reassignment or termination of an individual member of that group

Recommended Responsibilities:

- Certificate Services team: provide a system to track ownership as part of the inventory
- Certificate Owners: keep ownership information up-to-date (i.e., membership information for certificate owner group up-to-date)
- DevOps team: Where DevOps technology is used to deploy the certificate, the DevOps team should keep membership information for DevOps deployment technology group up-to-date

5.1.3 Approved CAs

CAs are trusted issuers of certificates. If organizations do not control the CAs that are used to issue certificates in their environments, then they will face several potential risks:

- **Increased costs**: If multiple groups are individually purchasing certificates from CAs, then the cost per certificate can be significantly higher because organizations are not taking advantage of volume discounts
- **Trust issues**: Each CA used to issue TLS certificates to servers in an organization must be trusted by the clients connecting to those servers via a root certificate. If a large number of CAs (internal and external) is used, then the organization is required to take on the extra burden of maintaining multiple trusted CA certificates on clients to avoid cases in which the necessary CA is not trusted, which can result in outages
- **Security risk**: A certificate owner may decide to set up his or her own CA on a system that does not have the necessary security controls and to configure the system to trust that CA. This increases the possibility of an attacker impersonating a server if the attacker compromises that CA and issues fraudulent certificates
- **Unexpected CA incidents**: If one of the untracked CAs used in the organization's environment encounters an issue, such as a CA compromise or suddenly being untrusted by browser vendors, then the organization may have to scramble to avoid security or operational issues for core applications

To ensure they can rapidly respond to a CA compromise or another incident when using public CAs, organizations should maintain contractual relationships with more than one public CA. By doing this, organizations will not have to scramble to negotiate a contract (which may take days or weeks) while attempting to respond to an urgent situation. Organizations should also maintain at least one backup internal CA so they can respond to an internal CA compromise or incident.

Recommended Requirements:

Certificates should be issued only by the following CAs:

- <External CA1>
- <External CA2>
- <Internal CA1>
- <Internal CA2>
- <…>
- Contractual relationships with at least two public CAs that conform to the CA/Browser Forum Baseline Requirements should be maintained at all times
- Internal CAs should be securely operated. Backup internal CAs should be maintained to support a rapid response to incidents, such as CA compromise

Recommended Responsibilities:

- Certificate Services team: manage business relationships with approved external CAs, and operate or outsource the operation of approved internal CAs
- Certificate owners: ensure that only certificates from approved CAs are used

5.1.4 Validity Periods

The validity period for a certificate defines the time that it is valid, from the first date/time (notBefore) to the last date/time (notAfter) that it can be used. It is important to note that the validity period of a certificate is different than the cryptoperiod of the public key contained in the certificate and the corresponding private key. It is possible to renew a certificate with the same public and private keys (i.e., not rekeying during the renewal process). However, this is only recommended when the private key is contained with a hardware security module (HSM) validated to Federal Information Processing Standards (FIPS) Publication 140-2 Level 2 or above.

One of the greatest risks of private-key compromise is from administrators who have direct access to plaintext private keys (including the ability to make a copy) and who are then reassigned or terminated. Although certificates would ideally be changed (rekeyed) each time an administrator with access to private keys is reassigned, this is often not practical. Therefore, ensuring certificates and their corresponding private keys are changed regularly is important, as shorter validity periods reduce the amount of time that a compromised private key can be used for malicious purposes. However, validity periods that are too short may increase the risk of outages. Organizations should determine the ideal validity period that balances security and operational risks for their organization. In general, due to the regular reassignment of administrative staff, it is recommended that validity periods be one year or less. The automated management of certificates can enable a more frequent renewal of certificates.

Recommended Requirement:

- The maximum validity period (i.e., from the notBefore date to the notAfter date for certificates should be <one year or less>

Recommended Responsibilities:

- Certificate Services team: ensure CAs are available to certificate owners to issue certificates with approved validity periods
- Certificate owners: ensure certificates are renewed and replaced before their expiration

5.1.5 Key Length

Each certificate contains a public key that is mathematically matched to a private key (which should be kept secret). To prevent an attacker from guessing the value of the private key, it is necessary to randomly pick the value of the private key from a large set of possible values. For example, it is more difficult for someone to guess a number selected between zero and 1,000,000 than a number selected between zero and 100. The key length effectively defines the size of the range of numbers from which private and public key values are selected. A longer key length is considered more secure. However, longer key lengths require more processing power and time, as well as more storage. Consequently, a balance must be struck between security risk and resource requirements. NIST monitors the industry to continually assess the potential crypto-analytical capabilities of possible attackers and their ability to guess the values of private keys. Based on this information, it sets recommended minimum key lengths. It is recommended that organizations require the use of keys with key lengths equal to or greater than the NIST recommendations.

Recommended Requirement:

All certificates should use key lengths that comply with NIST Special Publication (SP) 800-131A, which are currently equal to or greater than the following key lengths:

- RSA: <2,048>
- ECDSA: <224>

Recommended Responsibilities:

- Certificate Services team: provide dashboards, reports, and alerts that enable the rapid detection of unauthorized key lengths, and provide automation technologies that enable rapid remediation

- Certificate owners: use only TLS certificate public and private keys whose key lengths meet or exceed the organization's key-length policy, monitor their inventory, and replace certificates that do not comply with the policy

5.1.6 Signing Algorithms

Certificates are digitally signed by CAs so their authenticity can be verified. Signatures are generated by using digital signature algorithms (e.g., RSA, ECDSA) and hash algorithms (e.g., Secure Hash Algorithm 256 [SHA-256]). If certificates are signed by using a signing algorithm with an insufficient key length or by using vulnerable hash algorithms (e.g., SHA-1), then attackers can forge certificates and impersonate TLS servers. Consequently, organizations should ensure that all certificates are signed by using cryptographic algorithms that conform to approved standards.

Recommended Requirement:

- All certificates should be signed with an approved signature algorithm and key length and with an approved hash algorithm (e.g., SHA-256), as defined in NIST SP 800-131A and FIPS Publication 180-4

Recommended Responsibilities:

- Certificate Services team: ensure the availability of CAs that use approved signing algorithms, and provide reporting and alerting tools to enable the rapid identification of noncompliant certificates
- Certificate Owners: use only certificates signed with an approved signature algorithm and key length and with an approved hash algorithm, and identify and replace certificates signed with unapproved algorithms or key lengths

5.1.7 Subject DN and SAN Contents

The combination of Subject DN and SAN are used to identify the TLS server to which the certificate is issued. The Subject DN is in the form of an X.500 DN, which can include information such as the country, state, city/locality, organization, organizational unit (e.g., department), and a common name (CN). The CN, when present, and the SAN field contain the fully-qualified domain name or IP address of the TLS server. For publicly trusted certificates, the contents of the Subject DN are governed by the public CA that issues them. The CA/Browser Forum requires the SAN field to be present, however, the CN is now deprecated and the other fields in the DN are now optional, though in practice they are still present. For internal certificates, the contents of the Subject DN fields, such as the organizational unit, can help identify the group responsible for certificates.

Public CAs will often perform checks to validate that an organization owns a top-level domain (e.g., *www.company123.com*), and will then allow the organization to request a certificate with Subject DNs and with SANs containing domains subordinate to that domain (e.g., *www.company123.com*, *www.server1.company123.com*). Consequently, it is critical that organizations implement approval processes that ensure the Subject DNs and SANs in all certificate requests are thoroughly reviewed and vetted before they are sent to the CA.

Recommended Requirements:

Names used in Subject DNs should conform to the following requirements:

- The Organization (O) attribute in the Subject DN should be one of the following values:
    - <e.g., Company, Inc.>
    - The Organizational Unit attribute in the Subject DN should conform to the following categorization:
        - <specify whether department, location, or another categorization should be used>
    - The Locale (City), State (Province), and Country codes should be set to the following location:
        - <City, State, Country of organization identified in O = headquarters offices>
    - The CNs and SANs should not include wildcards (e.g., *.company123.com).
- The fully-qualified domain names or IP addresses in all Subject DNs and SANs should be reviewed and approved by an individual who is knowledgeable about the application or system for which the certificate is being requested and who can confirm that the requester is authorized to make the request.

Recommended Responsibilities:

- Certificate Services team: provide technology solutions to automatically detect and prevent Subject DN and SAN policy violations
- Certificate owners: ensure the Subject DNs and SANs in all certificates comply with policy

5.1.8 Automation

The broadening use of and reliance on TLS server certificates to secure important applications is rendering manual certificate management impractical. Risks such as certificate-related outages are often the result of errors made while manually managing certificates. Organizations are unable to manually replace large numbers of certificates in response to large-scale cryptographic incidents, such as CA compromises, in a timely manner. Consequently, organizations should work to automate certificate management on as many systems and applications as possible to decrease security and operational risks. Historically, many organizations can find it difficult to induce certificate owners to move from manual to automated methods—though the move to automation can significantly reduce their work and risk. New automation tools (e.g., DevOps) and protocols have increased the methods and options by which automated certificate management can be successfully performed. Consequently, organizations should define clear guidelines and policies for automation and for when continued manual management is justified due to operational or organizational constraints.

Recommended Requirement:

- Automation should be used wherever possible for the enrollment, installation, monitoring, and replacement of certificates, or justification should be provided for continuing to use manual methods that may cause operational security risks.

Recommended Responsibilities:

- Certificate Services team: provide a central system that supports certificate owners in automating the management of their certificates
- Certificate owners: automate the management of their certificates

5.1.9 Certificate Request Reviews – Registration Authority (RA)

To prevent the issuance of rogue certificates that can be used maliciously to impersonate legitimate servers, all certificate requests should be vetted to ensure they are issued only for valid systems and requested only by authorized parties. For certificates requested by individuals, it is important that the reviewer/approver has sufficient knowledge about the need for the certificate and about the personnel authorized to request certificates for the specific DNS address of the servers. It is generally impossible for a central team to be aware of all new applications and the people authorized to request certificates for those applications. Consequently, it is necessary to have certificate requests reviewed by local application owners who have this knowledge. For certificates requested by automated processes, such as DevOps frameworks, the necessary automated controls should be put in place to ensure that requesting applications are authenticated and that the DNS addresses for which they request certificates match specific patterns.

Recommended Requirements:

- All manual certificate requests for first issuance or renewal should be reviewed and approved by the business or application owner, who will confirm the following statements are true:
    - A certificate is required for the application/system. The certificate CN (when included) and SANs of the certificate match the addresses of the application/system in question.
    - The requester is authorized to make the request.
- When certificates are being issued by automated processes, the automated process should be reviewed by the business or application owner prior to implementation, who will confirm the following statements are true:
    - The automated process is capable of requesting certificates for specific CNs and SANs.

- There is consideration for the automation of the entire certificate life cycle, including renewal and revocation, built into the automated processes.
- A system for auditing and reviewing all certificates issued by the automated processes is in place.

Recommended Responsibilities:

- Certificate Services team: provide a central system for assigning approvers, alerting approvers when certificate requests need approval, and enabling approvers to review and approve/reject requests
- Certificate owners: assign review/approval responsibility to individuals who have knowledge of the systems (addresses) required for applications and of the individuals authorized to request certificates for those systems, and approve certificate requests in a timely manner

5.1.10 Private Key Security

Each TLS server certificate has a corresponding private key that must be kept secret to prevent compromise. Often, the private keys used with TLS server certificates are stored in plaintext files, which may be accessible by administrators if not properly secured. Even when the files where private keys are stored are encrypted with passwords, the passwords are stored in plaintext configuration files so that TLS servers can gain access to the private keys when they are started. It is possible to protect TLS private keys in HSMs; however, due to the large number of TLS servers where private keys would be required, many organizations have not used HSMs to protect private keys. Organizations should assess the criticality and risk of each TLS server and determine the appropriate level of protection required for private keys. Further, organizations should ensure that only authorized personnel have access to private keys and that the authorized personnel are trained in the processes necessary to keep the private keys secure.

Recommended Requirements:

- Access to TLS server private keys stored in plaintext files should be limited to authorized personnel. For mission-critical systems, TLS private keys should be stored in an HSM.
- Individuals granted access to private keys should complete training on procedures and practices for keeping private keys secure.

Recommended Responsibilities:

- Certificate Services team: provide training on the proper procedures for keeping private keys secure, and provide automation to simplify the management of TLS private keys stored in HSMs
- Certificate owners: ensure only authorized personnel are granted access to private keys, regularly review who is granted access to private keys, and ensure the authorized personnel receive training on the proper procedures for keeping private keys secure

5.1.11 Rekey/Rotation upon Reassignment/Terminations

Most private keys associated with TLS server certificates are stored in plaintext files. System administrators who manually manage TLS server certificates and associated private keys on their systems can make copies of the private-key files. Consequently, if a system administrator is reassigned or terminated, then the private key and certificate should be replaced (renewed) with a new key pair and certificate, and the previous certificate should be revoked, to prevent any malicious activities with the original private key and certificate. If automation is used for the management of certificates and private keys and if direct access by system administrators is limited (via limited-access controls and audit logging on any access), then certificate owners can avoid replacing certificates when a system administrator is reassigned or terminated.

Recommended Requirement:

- Private keys and the associated certificates that have the capability of being directly accessed by an administrator should be replaced within <30> days of reassignment or <5> days of termination of that administrator.

Recommended Responsibilities:

- Certificate Services team: provide automated certificate and key management services that remove the need for administrators to manually access private keys, alleviating the need to replace certificates and private keys when a system administrator is reassigned or terminated
- Certificate owners: ensure manually managed certificates and private keys are replaced when a system administrator with access is reassigned or terminated

5.1.12 Proactive Certificate Renewal

When a certificate is nearing expiration, it should be replaced. The replacement of certificates involves multiple steps, including reviewing and approving requests and testing the newly installed certificate(s) to ensure the application they secure is operating properly after replacement. If an unexpected issue is encountered with the new certificate and the associated private key, the previous certificate and private key can be restored and used if the certificate has not yet expired. If certificate owners are not proactive and instead wait until the last minute before requesting, obtaining, and installing a new certificate, this procrastination can cause unplanned, urgent work by multiple teams (including the Certificate Services team) and risk unplanned downtime for the application. Certificate owners should plan, initiate, and complete the certificate renewal, installation, and testing process several weeks ahead of certificate expiration to ensure unexpected issues and circumstances can be addressed and to avoid unnecessary “fire drills” for supporting teams (e.g., the Certificate Services team).

Recommended Requirement:

- Certificates should be renewed, installed, and tested at least <30> days prior to expiration of the currently installed certificate.
- If the validity period (total lifetime) of a certificate is shorter than <60> days (e.g., 20-day certificates used in short-lived/automated applications), then the certificate should be renewed before <80 percent> of the total validity period has elapsed.

Recommended Responsibilities:

- Certificate Services team: provide automated services for monitoring certificate expiration dates, send reports to certificate owners showing certificates expiring in the next <60–90> days, send alerts and escalations to certificate owners for certificates expiring in <30> days or fewer, and send alerts to executives for certificates expiring in <30> days or fewer
- Certificate owners: track upcoming expiration dates for their certificates, schedule replacement (in change windows where necessary), and ensure completion of certificate renewal, installation (of the new certificate), and verification of proper operation prior to the minimum renewal windows

5.1.13 Crypto-Agility

There are several incidents that can require organizations to rapidly replace large numbers of certificates and private keys, including CA compromise or distrust, vulnerable algorithms, or bugs in cryptographic libraries. There have been multiple examples of these incidents in recent years, including the CA compromise of DigiNotar, the distrust of Symantec certificates by browser vendors, the deprecation of SHA-1 for signature generation, and cryptographic library bugs in Debian and Infineon. In 2006, NIST first recommended that organizations stop using SHA-1 for signatures. However, many organizations were still struggling to eradicate the use of certificates signed with SHA-1 in 2017, when their use was forcibly stopped by browser vendors.

An unexpected cryptographic incident can require an organization to rapidly respond to ensure that its operations and services to customers are not interrupted for an extended period. In addition, the industry is preparing for a transition to quantum-resistant algorithms, which will require organizations to replace large numbers of certificates and private keys.

Recommended Requirements:

- System owners should maintain the ability to replace all certificates on their systems within <2> days to respond to security incidents such as CA compromise, vulnerable algorithms, or cryptographic library bugs.
- System owners should maintain the ability to track the replacement of certificates so it is clear which systems are updated and which are not.

- Select and establish contracts with backup CAs for public and internal certificates to enable rapid transition in response to a CA compromise.

Recommended Responsibilities:

- Certificate Services team: document effective processes for replacing large numbers of certificates and private keys; train all certificate owners on certificate replacement processes; provide services, such as automation, that enable the rapid replacement of large numbers of certificates and private keys; actively track the occurrence of cryptographic incidents that require replacement of certificates and private keys, and communicate clearly to certificate owners when such an event occurs; and ensure contracts with backup CAs for both public certificates and internal certificates (if applicable) are in place
- Certificate owners: proactively support crypto-agility by maintaining an inventory of all certificates for which they are responsible and corresponding ownership information, making sure that certificate replacement processes are as efficient as possible and that personnel are trained; and appropriately prioritize replacement of certificates and private keys when cryptographic incidents occur

5.1.14 Revocation

If the private key associated with a TLS server certificate is compromised, then the certificate can be revoked by the CA so that potential relying parties are alerted and do not trust the certificate. Certificate owners should understand their responsibility in revoking certificates and should proactively revoke certificates when an incident occurs. Inadvertent or malicious revocation of a certificate can cause downtime for the application that it secures; therefore, organizations should ensure they have processes to prevent unauthorized revocation.

Recommended Requirements:

- TLS server certificates should be revoked if the associated private key has been or is suspected of being compromised.
- Revocation of a TLS server certificate outside the renewal/replacement process can be initiated only by a certificate owner or identified security personnel and should be approved by the Certificate Services team or a designated security approver.

Recommended Responsibilities:

- Certificate Services team: provide the infrastructure and services to ensure that certificates can be rapidly and securely revoked when necessary and that certificates cannot be revoked without proper approval
- Certificate owners: request revocation of old certificates that have been replaced but that are still valid, and request revocation of certificates when a private key is compromised or suspected to be compromised

5.1.15 Continuous Monitoring

Because of the broad use of TLS server certificates in all critical communications, operational or security failures related to TLS server certificates can significantly impact the business operations of organizations. TLS certificates should be continuously monitored to prevent outages and security vulnerabilities. The certificates should be monitored for impending expiration; for situations in which they are not operating, are not configured properly, or are vulnerable; and for situations in which they are not consistent with policy.

Recommended Requirements:

- The expiration dates of certificates should be continuously monitored. Notifications should be automatically sent to certificate contacts <90, 60, and 30> days prior to expiration. If a certificate is not successfully renewed and replaced <30> days prior to expiration, then escalation notifications should be sent to the certificate owner management and incident response teams.
- The operation and configuration of certificates should be periodically checked to identify any issues or vulnerabilities.
- Certificates should be periodically checked to ensure they are consistent with policy.

Recommended Responsibilities:

- Certificate Services team: provide systems and services for continuously monitoring TLS server certificates, and support certificate owners in implementing TLS server certificate continuous monitoring and in keeping it operational
- Certificate owners: ensure continuous monitoring processes are in place and operational for all their TLS server certificates

5.1.16 Logging TLS Server Certificate Management Operations

TLS server certificates serve as trusted credentials that authenticate servers for mission-critical applications. Just as logging data access is required for forensics and other purposes, logging all certificate and private-key management operations is critical. Organizations should ensure they have a complete chain of custody for private keys and certificates that includes a log of all operations, including key-pair generation, certificate requests, request approval, certificate and key installation, the copying of certificates and keys (e.g., for load-balanced applications), certificate and key replacement, and certificate revocation. Logs should be collected and stored in a central location so the complete chain of events for certificates and private keys can be reviewed when necessary.

Recommended Requirement:

- A complete automated log should be maintained of all TLS certificate and private-key management operations (from creation to installation to revocation) that includes a description of the operation performed, any relevant metadata about the event (e.g., the location of files), the identity of the person/application performing the operation, and the date/time it was performed.

Recommended Responsibilities:

- Certificate Services team: provide a system for collecting all logged events, and provide tools that automatically log certificate and private-key management operations
- Certificate owners: ensure all tools used for certificate and private-key management operations log events in a central log

5.1.17 TLS Traffic Monitoring

While providing authentication and confidentiality for legitimate communications and operations, TLS can also be used by attackers to hide their operations, such as scanning for vulnerabilities, leveraging vulnerabilities for privilege escalation, denial-of-service operations, and data exfiltration. Depending on organizational policy, in addition to monitoring the content of TLS communications for external-facing systems, organizations may monitor TLS communications between internal systems to retain the ability to detect attackers who are attempting to pivot between internal systems (to gain access to critical data) or are exfiltrating compromised data. This monitoring may be accomplished in a variety of ways, including via proxy, end point software, or passive decryption. As discussed in Section 3.4, each organization should decide for itself whether the security risks posed by monitoring internal TLS traffic are worth the potential benefits of having visibility into the encrypted traffic. If, on the other hand, the organization determines it is in its best interests to perform TLS traffic monitoring, then the recommended related requirements and responsibilities are as follows.

Recommended Requirement:

- Where TLS monitoring via passive decryption is supported, TLS server private keys should be securely and automatically transferred to TLS decryption devices and updated when TLS certificates are replaced.

Recommended Responsibilities:

- Certificate Services team: provide a secure method for transporting TLS private keys between TLS servers and passive decryption devices when passive decryption is used for TLS traffic monitoring
- Certificate owners: ensure all communications protected by TLS are monitored for unauthorized operations and data exfiltration

5.1.18 Certificate Authority Authorization

An attacker can impersonate a server if the attacker is able to get a certificate issued that includes the name of the server and his or her own public key. To mitigate this type of attack, organizations can populate Certificate Authority Authorization (CAA) records for the DNS domains of their servers with the names of one or more CAs authorized to issue certificates for that server. When a CA receives a certificate request for a domain, it should check the domain in the DNS to see if a CAA record is defined. If a CAA record is defined, then before issuing a certificate, the CA should ensure the CA's name is listed in a CAA record for the domain. CAA records can be specified for second-level domains (e.g., *www.organization1.com*), which will apply to all subordinate domains and to individual domains (e.g., *www.alpha.organization1.com*). Because an attacker can attempt to request a certificate for a domain from one of the CAs listed in the CAA record, the organization should ensure the listed CAs accept certificate requests only from parties authorized by the organization.

Recommended Requirement:

- CAA records should be populated with authorized CAs for all domains for which public certificates may be issued.

Recommended Responsibilities:

- Certificate Services team: ensure CAA records are defined with approved CAs for all second-level domains owned by an organization
- Certificate owners: ensure the Certificate Services team is aware of all second-level domains for which the certificate owner is requesting certificates

5.1.19 Certificate Transparency

Certificate Transparency (CT) provides a publicly searchable log of issued certificates. CT is primarily focused on certificates issued by public CAs. Some browsers require that certificates issued by public CAs be published to a publicly available CT log; otherwise, the browser will display a warning to the user. The availability of CT logs enables organizations to confirm that unauthorized certificates have not been issued for their domains.

Recommended Requirement:

- CT logs should be regularly monitored to ensure unauthorized certificates have not been issued for any domains owned by the organization.

Recommended Responsibility:

- Certificate Services team: establish an automated process for monitoring CT logs

5.1.20 CA Trust by Relying Parties

Clients that connect to TLS servers verify the validity of those servers' certificates by using CA certificates or root certificates that they store locally in their systems. Many operating systems and applications (e.g., browsers) are preloaded with certificates from public CAs that have met the requirements of standards organizations, such as the CA/Browser Forum. Some applications, such as browsers, may include more than 100 trusted CA certificates. To reduce their exposure to CA compromise incidents, organizations should minimize the CAs that their clients trust to only those they are likely to need to trust. For example, if certain systems acting as TLS clients are used only for internal operations, then they should trust only the certificate(s) from the internal CA(s). Furthermore, if certain TLS clients communicate with TLS servers from select partners, then certificates from only the CAs expected to be used by those partners should be trusted. Organizations should maintain an inventory of CA certificates trusted on all their systems, ensure only needed CAs are trusted, and maintain the ability to rapidly remove or replace CA certificates that should no longer be trusted.

Recommended Requirement:

- CA certificates trusted by TLS clients should be limited to only those required to validate TLS certificates of the servers with which the client communicates. All unneeded CA certificates should be removed. The following CAs should never be trusted:
    - <e.g., DigiNotar>
    - <…>

Recommended Responsibilities:

- Certificate Services team: provide the technology and services for discovering and creating inventories of existing CA certificates and for managing (e.g., adding, removing) CA certificates
- Certificate owners: limit CA trust to the minimum needed for each system and ensure all other CAs are removed

5.2 Establish a Certificate Service

Manually managing TLS server certificates is infeasible due to the large number of certificates in most enterprises. It is also not feasible for each certificate owner to create their own certificate management system. The most efficient and effective approach is for the Certificate Services team to provide a central Certificate Service that includes technology-based solutions that provide automation and that support certificate owners in effectively managing their certificates. This service should include the technology/services for CAs, certificate discovery, inventory management, reporting, monitoring, enrollment, installation, renewal, revocation, and other certificate management operations.

The central Certificate Service should also provide self-service access for certificate owners so they are able to configure and operate the services for their areas without requiring significant interaction with the Certificate Services team. Furthermore, the central Certificate Service should be able to integrate with other enterprise systems, including identity and access management systems, ticketing systems, configuration management databases, email, workflow, and logging and auditing.

5.2.1 CAs

Approved CAs should be designated and made available to certificate owners for requesting public and internal certificates. If, as is common, different CAs will be used for issuing public and internal certificates, then instructions should be provided to certificate owners to help them select the correct CA based on the purpose of the server where the certificate will be used. Establish backup CAs for both public and internal certificates, including completing contracts with backup public CAs so an immediate cutover is possible in case of a CA compromise, for business reasons, or because of some other motivation.

5.2.2 Inventory

An up-to-date inventory of deployed TLS server certificates is the foundation of an effective certificate management program. The functionality required by an inventory system generally makes it infeasible for certificate owners to operate and manage their own inventory systems. It is imperative that the Certificate Services team provides a central system that certificate owners can use to maintain an inventory of their certificates. Without a central, up-to-date inventory, the Certificate Services team has no way of proactively monitoring for certificate-related security and operational risks or supporting certificate owners in minimizing such risks.

The central inventory system should provide the following characteristics and functions:

- **Automatic parsing**: certificates contain multiple fields of information (e.g., subject, issuer, expiration date) that should be monitored. The inventory system should provide automatic parsing of the contents of certificates that are loaded into it so searches can be performed on individual fields
- **Additional metadata**: It should be possible to associate additional information/metadata with each certificate (e.g., identifiers of the owners and approvers; installed locations; application identifiers; cost center numbers)
- **Organization**: With hundreds or thousands of certificates spread across many certificate owners and geographic locations, the inventory system should support organizing certificates into distinct groups/folders
- **Access controls**: To prevent unauthorized actions, it should be possible to define and enforce access controls that are assigned to groups or individuals

- **Support certificate management**: As the foundation of a certificate management program, the inventory system should integrate with and support all other certificate management functions (e.g., discovery, enrollment portal, approvals, automation)

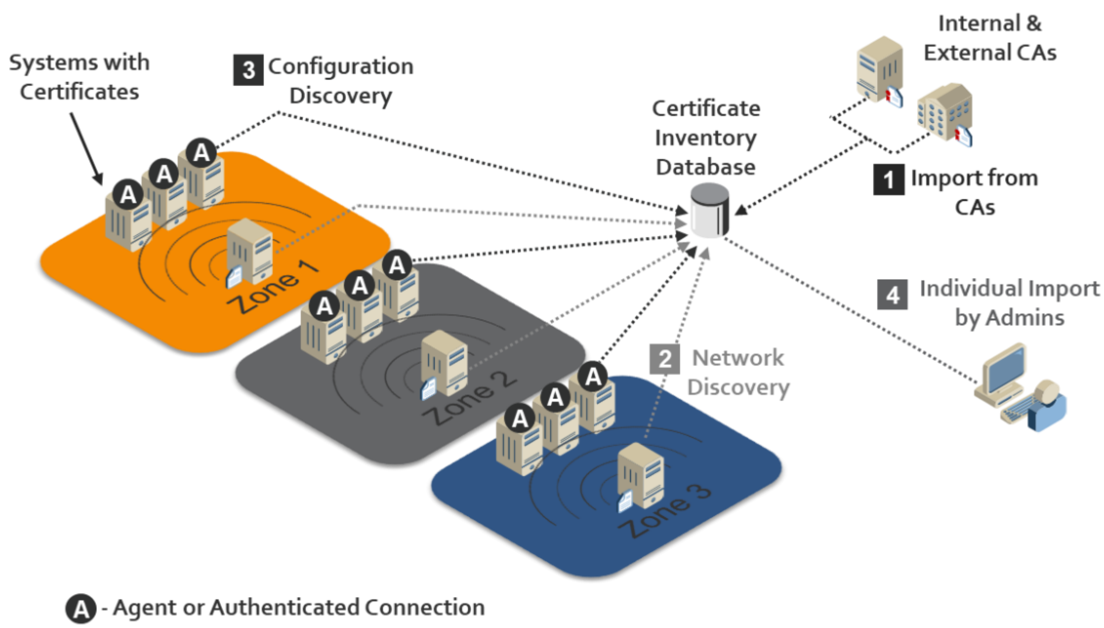
5.2.3 Discovery and Import

Manually establishing and maintaining an up-to-date and comprehensive inventory is difficult, if not impossible. Because of the complexity of most enterprise environments — which contain firewalls, different security/operations restrictions, etc. — it is often not sufficient to have a single method of automatically populating and maintaining an inventory. The central Certificate Service should provide multiple options for automated discovery and the import of certificates, including those listed below:

- **CA import**: automated import of certificates from CAs. This is often the fastest way to initially populate the certificate inventory. However, it will only provide an inventory of certificates from known CAs
- **Network discovery**: automated scanning of one or more configurable sets of IP addresses, IP address ranges, and ports for TLS server certificates. This helps provide a comprehensive view of all certificates and their locations. Organizations typically find certificates from unapproved CAs and self-signed certificates (which should likely be replaced with certificates from approved CAs). The network discovery service should support operation across multiple network zones separated by firewalls
- **Configuration discovery**: Network discovery can find certificates and determine their network location(s); however, it does not allow for collection of configuration information, such as the type of keystore (e.g., Privacy Enhanced Mail, Public Key Cryptography Standards [PKCS] #12, HSM), the storage location on the server, and other information that can be helpful in detecting issues and in setting up automated management for the certificate. The inventory system should provide a means of discovering certificate configuration information via an authenticated connection or agent
- **Bulk import**: In addition to network discovery and CA import, it is beneficial to have the option for administrators to import certificate data. This helps in cases where network discovery and CA import are not possible and in cases where there is additional information/metadata (e.g., contacts, approvers, cost centers) that can be associated with each certificate to help in tracking and management.

Figure 5-1 depicts options for automated discovery and import of certificates.

Figure 5-1 Various Options for Automated Discovery and the Import of Certificates



5.2.4 Management Interfaces

Certificate owners and the Certificate Services team should provide user interfaces to view and manage certificates. The interfaces should be simple enough to support certificate owners who have small numbers of certificates and perform management operations infrequently. The interfaces should also offer more-sophisticated functionality to support the needs of certificate owners with large numbers of certificates and the needs of the Certificate Services team.

The interfaces should provide the following characteristics and functions:

- **Inventory view**: Certificate owners should be able to view their certificates (to which they have been granted access). The Certificate Services team should be able to view the entire inventory
- **Searching and filtering**: Certificate owners with large numbers of certificates, and the Certificate Services team, should be able to search and filter operations so they can quickly find specific certificates
- **Enrollment and renewal**: The portal should provide a simple method to request new certificates and to renew existing certificates. Having a single interface for enrollment and renewal across all CAs reduces the retraining needed when moving CAs, resulting in better crypto-agility
- **Approvals**: If an external system is not used for reviewing certificate requests, then the portal should provide a method for an approver to perform RA functions to review the relevant details of certificate requests and to approve/reject the requests with comments

5.2.5 Automated Enrollment and Installation

Manually requesting, installing, and managing large numbers of certificates is error-prone and resource-intensive; increases security risk; and does not allow for a rapid response to large-scale incidents, such as CA compromises. In cloud environments, the ability to quickly spin up new instances to support increased loads is critical. Because most enterprises have a range of systems from different vendors with diverse management methods, the central Certificate Service should offer multiple options for automation, including those listed below:

- **Programmatic automation**: The central Certificate Service should provide a set of application programming interfaces (APIs) (e.g., Representational State Transfer) that enable enrollment, revocation, reporting, etc. The central Certificate Service should support easy integration with and access from DevOps frameworks and other programming tools
- **Standard protocol support**: The central Certificate Service should support standard protocols for requesting certificates, including the Simple Certificate Enrollment Protocol (SCEP), Automated Certificate Management Environment, and Enrollment over Secure Transport
- **Proprietary automation**: Some systems may not support programmatic or standards-based enrollment and installation but may provide other methods (e.g., APIs, command-line utilities) that can be used to automate certificate enrollment and installation. This may be performed with an agent or via a remote authenticated connection
- **Secure key transport**: Within organizations that, by policy, permit TLS traffic monitoring and enable detection of encrypted threats by using passive decryption devices, the central Certificate Service should provide the ability to securely transport TLS private keys from TLS servers to the decryption devices that enable inspection of encryption communications

Automation should support integration with HSMs when HSMs are used for protection of private keys.

5.2.6 RA/Approvals

Certificate requests should be reviewed and vetted to ensure unauthorized certificates are not issued or used for malicious purposes. Large enterprises generally have hundreds of different departments, business applications, projects, and systems administrators, making it infeasible for a central group to have the relevant knowledge needed to vet requests. The central Certificate Service should provide the ability to assign individuals (e.g., application owners) to review certificate requests for their respective areas. Once approvers are assigned, the central Certificate Service should automatically route certificate requests to assigned reviewers for approval and enable them to review any relevant data needed to properly vet requests.

5.2.7 Reporting and Analytics

To address TLS server certificate-related risks, certificate owners and the Certificate Services team should have visibility across their inventory and be able to quickly identify TLS server certificate issues or vulnerabilities. The most efficient method of addressing risks is proactive notifications sent by the central Certificate Service, based on configured rules. However, reports and dashboards can help in planning (e.g., an unexpectedly large number of certificate expirations coming in the next few weeks) and identifying anomalies that would otherwise not be caught by the automated rules. The central Certificate Service should support the following reporting and analysis tools:

- **Custom reporting**: Users should be able to create customized reports, including the data to be presented, the filtering criteria for the results, the scheduling of execution, and the selection of report recipients
- **Dashboards**: To help in identifying anomalies or unexpected issues, dashboards should proactively highlight risks, such as certificates with weak keys, vulnerable algorithms, impending expirations, operational errors, and other issues
- **Interfaces to monitoring systems**: Many organizations rely upon automated security incident and event monitoring systems that collect, analyze, and correlate information that is subsequently displayed or used to notify humans of events and the actions required. Certificate-related anomalies and issues should be delivered to such systems

5.2.8 Passive Decryption Support

If passive decryption devices are used to monitor TLS-encrypted communications for attacks, then those devices must have copies of the private keys from all monitored TLS servers so the devices are able to decrypt TLS traffic to those servers. Manually transporting private keys from TLS servers to passive decryption devices creates risk of a compromise. Consequently, when passive decryption is used, the central Certificate Service should provide an automated and secure method for transporting private keys from TLS servers to passive decryption devices and for keeping the private keys up-to-date when new keys (and certificates) are deployed.

5.2.9 Continuous Monitoring

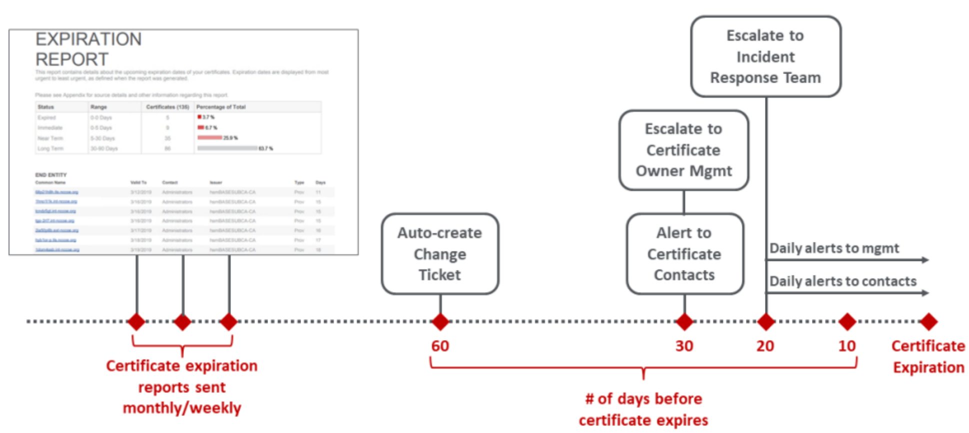
To prevent operational or security incidents, the certificates should be continuously monitored across the enterprise. Continuous monitoring should include the following types of monitoring:

- **Expiration monitoring**: To prevent outages due to expired certificates, the expiration dates for all certificates should be monitored. It should be possible to configure the time periods when notifications will be sent to certificate contacts prior to expiration (e.g., 90 days, 60 days, 30 days). If timely action is not taken, then it should be possible to escalate and send notifications to managers or a central incident response team

- **Operation/configuration monitoring**: Once a known good state is established (e.g., the location and configuration of certificates), the central Certificate Service should monitor and detect situations in which certificates are not operating, are not configured properly, or are vulnerable
- **Policy compliance**: The central Certificate Service should detect and send alerts when deployed certificates are not consistent with policy

Because certificate expirations are a regular occurrence, especially for certificate owners with large numbers of certificates, it is important to not inundate certificate owners with notifications, as they will likely start to ignore them. An effective strategy is to combine the use of reports, change tickets, and alerts. Sending regular (e.g., monthly) reports containing a list of certificates expiring within a certain number of days (e.g., 120 days) helps certificate owners plan for expirations. Automatically creating change tickets in the organization's central ticketing system can ensure certificate renewals and replacements are handled in the same way that other change operations are performed. Sending alerts within 30 days of expiration and escalating to management and incident response teams ensures certificates not replaced in a timely fashion are identified before they expire. Figure 5-2 provides an example schedule for reports, tickets, and alerts.

Figure 5-2 Example Timeline of Processes and Notifications Triggered by Impending Certificate Expiration



5.2.10 Education

Management of TLS server certificates in an enterprise environment is complex, time-consuming, error-prone, and security-sensitive. Most certificate owners are not knowledgeable about TLS server certificates, the processes for effectively managing certificates, or their own certificate-related responsibilities. Consequently, the Certificate Services team should provide readily accessible educational materials, preferably online and available on demand. The TLS server certificate educational materials should include the following items:

- basic introduction to certificates and keys (e.g., when certificates are used, obtaining certificates, protecting keys, certificate changes, revocation)
- risks of improper TLS server certificate management
- explanation of TLS server certificate policies and certificate owner responsibilities
- step-by-step instructions for managing TLS server certificates, including any of the following steps offered via the central Certificate Service:
  - creating an inventory
  - reviewing the inventory and identifying risks/vulnerabilities (e.g., generating reports)
  - manually requesting and installing TLS server certificates on each relevant operating system/application (e.g., Apache)
  - DevOps/API-based request and installation
  - agentless automated installation
  - agent-based automated installation
  - renewing certificates
  - revoking certificates

There are many educational resources available on the internet that can alleviate the need to create new materials. An internal TLS server certificate education website can include links to helpful web pages and websites.

5.2.11 Help Desk

In addition to educational materials, certificate owners should have a central support service that they can contact about questions and that can assist in troubleshooting issues. Many certificate owners may be new to TLS server certificate management or responsible for only a small number of certificates (e.g., one to five certificates) and will likely need assistance in successfully performing necessary operations. Any certificate owner calling the help desk should be required to have completed the educational programs that apply to their use cases so that help-desk personnel do not need to explain basic concepts that can be learned prior to the request for help.

TLS server certificates are typically installed or renewed during scheduled maintenance windows, which are often scheduled on weekends and/or in the middle of the night. Issues related to TLS server certificates can often arise during these scheduled maintenance operations; therefore, help-desk personnel should be made available during all times when certificate issues may arise (e.g., 24 hours a day, seven days a week). Help-desk personnel should be knowledgeable about and experienced in TLS server certificate management. It is possible to have general help-desk personnel answer and address Level One certificate calls and escalate to more-experienced personnel as needed for Level Two and Level Three calls.

5.3 Terms of Service

It is helpful to define the terms of service for the central Certificate Service to avoid confusion by certificate owners about the services they will receive and their responsibilities. The terms of service should include those listed below:

- description of the services provided (e.g., network discovery, monitoring enrollment, automation)
- responsibilities of the certificate owners and the Certificate Services team (e.g., the Certificate Services team will help with network discovery, but a certificate owner is responsible for working with the network team to allow the discovery on their systems)
- expected service levels — stated in service level agreements — with response times

5.4 Auditing

Due to the fundamental role that TLS server certificates play in securing data and systems, periodic reviews of TLS server certificate management practices are essential. Auditors should confirm that TLS server certificate policy requirements are addressed. For example, all certificate owners should be able to demonstrate they have a certificate inventory and to describe the steps they have taken to ensure all certificates are included in the inventory. The Certificate Services team should demonstrate it is providing the services needed for certificate owners to comply with policy.

TLS server certificate risks can lie latent for long periods of time and then can unexpectedly have significant impact to an organization's operations —due to either operational outages or security issues. Consequently, regular audits of certificate management practices performed by compliance auditors are critical to prevent unanticipated issues.

6 Implementing a Successful Program

The broad distribution of TLS server certificates across distinct groups, networks, and systems can present unique challenges in implementing an effective certificate management program across an enterprise environment. The following resources are helpful for successful implementation:

- **Executive owner**: It is essential to have an executive owner for the certificate management program. This executive owner should be prepared to educate the executives of each group of certificate owners on TLS server certificate risks and the executives' responsibilities
- **Prioritization of risks**: Each organization has different challenges and priorities related to TLS server certificates. Although the best practices detailed in this practice guide are intended to help address all the risks related to TLS server certificates, it is helpful to prioritize those risks based on historical certificate issues and business needs. This prioritization can help in communications with certificate owners and with setting objectives and prioritizing tasks
- **Objectives**: Establishing clear and achievable objectives provides targets, helps focus efforts, and improves the likelihood of successful implementation. For example, if an organization finds it does not have an inventory and recognizes there are two groups that may be difficult to inventory in the near term, then one objective may be to create an inventory of all other groups' TLS server certificates in the next 12 months
- **Action plan**: An action plan with specific tasks, responsibilities, and milestones, geared to achieve the objectives, should be created, communicated, and reviewed by all stakeholders (e.g., certificate owners, Certificate Services team, executive owner). The action plan should be prioritized to address the most important objectives first. For example, an action plan might include the following objectives:
    - 30 days from the start of the project:
        - complete certificate imports from CA1, CA2, and CA3
        - require certificate enrollment through the central Certificate Service portal and prevent enrollment directly to CAs
    - 90 days from the start of the project:
        - complete network discovery across all North American and European data centers
        - complete the assignment of certificate owners for all certificates in inventory
    - 180 days from the start of the project:
        - automate certificate enrollment and installation on all load balancers
        - automate certificate enrollment and installation for all e-commerce web servers
        - complete network discovery across all Asia-Pacific data centers
- **Regular executive reviews**: The objectives and action plan should be reviewed with the executive owner at commencement of the project, and regular reviews should be scheduled (e.g., every 90 days) to track progress. During these reviews, the executive owner should note areas where additional action by certificate owners is needed so the executive owner can proactively communicate with peer executives to ensure action is taken

- **Periodic audits**: Due to the critical role that TLS server certificates play in the security and operations of organizations, and the risks resulting from improper management, regular audits should confirm the Certificate Services team and certificate owners are fulfilling their responsibilities in TLS server certificate management.

Security testing should be defined as part of the organization’s policies. Before going live with any recommendations in this document, authorization from the security team should be provided, as specified by security policy.

Appendix A List of Acronyms and Abbreviations

ACME	Automated Certificate Management Environment
AD	Active Directory
API	Application Programming Interface
BGP	Border Gateway Protocol
CA	Certificate Authority
CAA	Certificate Authority Authorization
CAS	Certification Authority System
CAPI	Cryptographic Application Programming Interface (also known variously as CryptoAPI, Microsoft Cryptography API, MS-CAPI or simply CAPI)
CIO	Chief information officer
CN	Common Name
CRL	Certificate Revocation List
CSF	Cybersecurity Framework
CSR	Certificate Signing Request
CT	Certificate Transparency
DevOps	Development Operations
DN	Distinguished Name
DNS	Domain Name System
ECDSA	Elliptic Curve Digital Signature Algorithm
EV	Extended Validation
FIPS	Federal Information Processing Standards
HSM	Hardware Security Module
HTTP	Hypertext Transfer Protocol
HTTPS	Hypertext Transfer Protocol Secure

IETF	Internet Engineering Task Force
IIS	Internet Information Server (Microsoft Windows)
IoT	Internet of Things
IP	Internet Protocol
LDAP	Lightweight Directory Access Protocol
NIST	National Institute of Standards and Technology
NCCoE	National Cybersecurity Center of Excellence
OS	Operating System
OV	Organization Validated
PCI-DSS	Payment Card Industry Data Security Standard
PKCS	Public Key Cryptography Standards
PKI	Public Key Infrastructure
RA	Registration Authority
REST	Representational State Transfer (API)
RMF	Risk Management Framework
RSA	Rivest, Shamir, & Adleman (public key encryption algorithm)
SAN	Subject Alternative Name
SCEP	Simple Certificate Enrollment Protocol
SHA-1	Secure Hash Algorithm 1
SHA-256	Secure Hash Algorithm 256
SP	Special Publication
SSL	Secure Socket Layer (protocol)
SSLV	SSL Visibility (Symantec Appliance)
TLS	Transport Layer Security (protocol)
TPP	Trust Protection Platform (Venafi)

UPN	User Principal Name
URL	Uniform Resource Locator

Appendix B Glossary

Active Directory	A Microsoft directory service for the management of identities in Windows domain networks.
Application	1. The system, functional area, or problem to which information technology is applied. The application includes related manual procedures as well as automated procedures. Payroll, accounting, and management information systems are examples of applications. (NIST SP 800-16) 2. A software program hosted by an information system. (NIST SP 800-137)
Authentication	Verifying the identity of a user, process, or device, often as a prerequisite to allowing access to a system's resources. (NIST SP 800-63-3)
Automated Certificate Management Environment	A protocol defined in IETF RFC 8555 that provides for the automated enrollment of certificates.
Certificate	A set of data that uniquely identifies an entity, contains the entity's public key and possibly other information, and is digitally signed by a trusted party, thereby binding the public key to the entity. Additional information in the certificate could specify how the key is used and its validity period. (NIST SP 800-57 Part 1 Rev. 4 under Public-key certificate) (Certificates in this practice guide are based on IETF RFC 5280.)
Certificate Authority	A trusted entity that issues and revokes public key certificates. (NISTIR 8149)
Certificate Authority Authorization	A record associated with a Domain Name Server (DNS) entry that specifies the CAs that are authorized to issue certificates for that domain.
Certificate Chain	An ordered list of certificates that starts with an end-entity certificate, includes one or more certificate authority (CA) certificates, and ends with the end-entity certificate's root CA certificate, where each certificate in the chain is the certificate of the CA that issued the previous certificate. By checking to see if

	each certificate in the chain was issued by a trusted CA, the receiver of an end-user certificate can determine whether or not it should trust the end-entity certificate by verifying the signatures in the chain of certificates.
Certificate Management	Process whereby certificates (as defined above) are generated, stored, protected, transferred, loaded, used, and destroyed. (CNSSI 4009-2015) (In the context of this practice guide, it also includes inventory, monitoring, enrolling, installing, and revoking.)
Certificate Revocation List	A list of digital certificates that have been revoked by an issuing CA before their scheduled expiration date and should no longer be trusted.
Certificate Signing Request	A request sent from a certificate requester to a certificate authority to apply for a digital identity certificate. The certificate signing request contains the public key as well as other information to be included in the certificate and is signed by the private key corresponding to the public key.
Certificate Transparency	A framework for publicly logging the existence of Transport Layer Security (TLS) certificates as they are issued or observed in a manner that allows anyone to audit CA activity and notice the issuance of suspect certificates as well as to audit the certificate logs themselves. (Experimental RFC 6962)
Chief information officer	Organization's official responsible for: (i) Providing advice and other assistance to the head of the organization and other senior management personnel of the organization to ensure that information technology is acquired and information resources are managed in a manner that is consistent with laws, directives, policies, regulations, and priorities established by the head of the organization; (ii) Developing, maintaining, and facilitating the implementation of a sound and integrated information technology architecture for the [organization]; and (iii) Promoting the effective and efficient design and operation of all major information resources management processes for the organization, including improvements to work processes of the organization. (NIST SP 800-53 Rev. 4 adapted) Note: A subordinate organization may assign a chief information officer to denote an individual filling a position with security

	responsibilities with respect to the subordinate organization that are similar to those that the chief information officers fills for the organization to which they are subordinate.
Client	1. A machine or software application that accesses a cloud over a network connection, perhaps on behalf of a consumer. (NIST SP 800-146) 2. A function that uses the PKI to obtain certificates and validate certificates and signatures. Client functions are present in CAs and end entities. Client functions may also be present in entities that are not certificate holders. That is, a system or user that verifies signatures and validation paths is a client, even if it does not hold a certificate itself. (NIST SP 800-15)
Cloud Computing	A model for enabling ubiquitous, convenient, on-demand network access to a shared pool of configurable computing resources (e.g., networks, servers, storage, applications, and services) that can be rapidly provisioned and released with minimal management effort or service provider interaction. (NIST SP 800-145)
Common Name	An attribute type that is commonly found within a Subject Distinguished Name in an X.500 directory information tree. When identifying machines, it is composed of a fully qualified domain name or IP address.
Configuration Management	A collection of activities focused on establishing and maintaining the integrity of information technology products and information systems, through control of processes for initializing, changing, and monitoring the configurations of those products and systems throughout the system development life cycle. (NIST SP 800-53 Rev. 4)
Container	A method for packaging and securely running an application within an application virtualization environment. Also known as an application container or a server application container. (NIST SP 800-190)
Cryptographic Application Programming Interface	An application programming interface included with Microsoft Windows operating systems that provides services to enable developers to secure Windows-based applications using cryptography. While providing a consistent API for applications,

	CAPI allows for specialized cryptographic modules (cryptographic service providers) to be provided by third parties, such as hardware security module (HSM) manufacturers. This enables applications to leverage the additional security of HSMs while using the same APIs they use to access built-in Windows cryptographic service providers. (Also known variously as CryptoAPI, Microsoft Cryptography API, MS-CAPI or simply CAPI)
Cryptography API: Next Generation	The long-term replacement for the Cryptographic Application Programming Interface (CAPI).
Demilitarized Zone	A perimeter network or screened subnet separating an internal network that is more trusted from an external network that is less trusted.
Development Operations (DevOps)	A set of practices for automating the processes between software development and information technology operations teams so that they can build, test, and release software faster and more reliably. The goal is to shorten the systems development life cycle and improve reliability while delivering features, fixes, and updates frequently in close alignment with business objectives.
Digital Certificate	Certificate (as defined above).
Digital Signature	The result of a cryptographic transformation of data that, when properly implemented, provides origin authentication, assurance of data integrity, and signatory non-repudiation. (NIST SP 800-133)
Digital Signature Algorithm	A Federal Information Processing Standard for digital signatures, based on the mathematical concept of modular exponentiations and the discrete logarithm problem. (FIPS 186-4)
Directory Service	A distributed database service capable of storing information, such as certificates and CRLs, in various nodes or servers distributed across a network. (NIST SP 800-15) (In the context of this practice guide, a directory services stores identity information and enables the authentication and identification of people and machines.)
Distinguished Name	An identifier that uniquely represents an object in the X.500 directory information tree. (RFC 4949 Ver 2)
Domain	A distinct group of computers under a central administration or authority.

Domain Name	A label that identifies a network domain using the Domain Naming System.
Domain Name Server	The internet's equivalent of a phone book. It maintains a directory of domain names, as defined by the Domain Name System, and translates them to Internet Protocol addresses.
Domain Name System	The system by which Internet domain names and addresses are tracked and regulated as defined by IETF RFC 1034 and other related RFCs.
Elliptic Curve Digital Signature Algorithm	A digital signature algorithm that is an analog of DSA using elliptic curve mathematics and specified in ANSI draft standard X9.62. (NIST SP 800-15)
Enrollment	The process that a CA uses to create a certificate for a web server or email user. (NISTIR 7682) (In the context of this practice guide, enrollment applies to the process of a certificate requester requesting a certificate, the CA issuing the certificate, and the requester retrieving the issued certificate.)
Extended Validation Certificate	A certificate used for HTTPS websites and software that includes identity information that has been subjected to an identity verification process standardized by the CA Browser Forum in its Baseline Requirements that verifies that the identified owner of the website for which the certificate has been issued has exclusive rights to use the domain; exists legally, operationally, and physically; and has authorized the issuance of the certificate.
Federal Information Processing Standards (FIPS)	A standard for adoption and use by federal departments and agencies that has been developed within the Information Technology Laboratory and published by the National Institute of Standards and Technology, a part of the U.S. Department of Commerce. A FIPS covers some topic in information technology in order to achieve a common level of quality or some level of interoperability. (NIST SP 800-161)
Hardware Security Module (HSM)	A physical computing device that provides tamper-evident and intrusion-resistant safeguarding and management of digital keys and other secrets, as well as crypto-processing. FIPS 140-2 specifies requirements for HSMs.

Hostname	Hostnames are most commonly defined and used in the context of DNS. The hostname of a system typically refers to the fully qualified DNS domain name of that system.
Hypertext Transfer Protocol	A standard method for communication between clients and Web servers. (NISTIR 7387)
Internet Engineering Task Force (IETF)	The internet standards organization made up of network designers, operators, vendors, and researchers that defines protocol standards (e.g., IP, TCP, DNS) through process of collaboration and consensus.
Internet Message Access Protocol	A method of communication used to read electronic mail stored in a remote server. (NISTIR 7387)
Internet of Things (IoT)	As used in this publication, user or industrial devices that are connected to the internet. IoT devices include sensors, controllers, and household appliances.
Internet Protocol	The Internet Protocol, as defined in IETF RFC 6864, which is the principal communications protocol in the IETF Internet protocol suite for specifying system address information when relaying datagrams across network boundaries.
Lightweight Directory Access Protocol (LDAP)	The Lightweight Directory Access Protocol, or LDAP, is a directory access protocol. In this document, LDAP refers to the protocol defined by RFC 1777, which is also known as LDAP V2. LDAP V2 describes unauthenticated retrieval mechanisms. (NIST SP 800-15)
Microservice	A set of containers that work together to compose an application. (NIST SP 800-190)
Organization	An entity of any size, complexity, or positioning within an organizational structure (e.g., a federal agency or, as appropriate, any of its operational elements). (NIST SP 800-39) This publication is intended to provide recommendations for organizations that manage their own networks (e.g., that have a chief information officer).
Outage	A period when a service or an application is not available or when equipment is not operational.

Payment Card Industry Data Security Standard	An information security standard administered by the Payment Card Industry Security Standards Council that is for organizations that handle branded credit cards from the major card schemes.
Pivoting	A process where an attacker uses one compromised system to move to another system within an organization.
PIN Entry Device	An electronic device used in a debit, credit, or smart card-based transaction to accept and encrypt the cardholder's personal identification number.
Post Office Protocol (POP)	A mailbox access protocol defined by IETF RFC 1939. POP is one of the most commonly used mailbox access protocols. (NIST SP 800-45 Version 2).
Private Key	The secret part of an asymmetric key pair that is used to digitally sign or decrypt data. (NIST SP 800-63-3).
Public CA	A trusted third party that issues certificates as defined in IETF RFC 5280. A CA is considered public if its root certificate is included in browsers and other applications by the developers of those browsers and applications. The CA/Browser Forum defines the requirements public CAs must follow in their operations.
Public Key	The public part of an asymmetric key pair that is used to verify signatures or encrypt data. (NIST SP 800-63-3).
Public Key Cryptography	Cryptography that uses separate keys for encryption and decryption; also known as asymmetric cryptography. (NIST SP 800-77)
Public Key Infrastructure (PKI)	The framework and services that provide for the generation, production, distribution, control, accounting, and destruction of public key certificates. Components include the personnel, policies, processes, server platforms, software, and workstations used for the purpose of administering certificates and public-private key pairs, including the ability to issue, maintain, recover, and revoke public key certificates. (NIST SP 800-53 Rev. 4)
Registration Authority (RA)	An entity authorized by the certification authority system (CAS) to collect, verify, and submit information provided by potential subscribers, which is to be entered into public key certificates. The

	term RA refers to hardware, software, and individuals that collectively perform this function. (CNSSI 4009-2015)
Re-key	To change the value of a cryptographic key that is being used in a cryptographic system application; this normally entails issuing a new certificate on the new public key. NIST SP 800-32 under Re-key (a certificate)
Renew	The act or process of extending the validity of the data binding asserted by a public key certificate by issuing a new certificate. NIST SP 800-32 (The new certificate is typically used to replace the existing certificate, and both certificates typically contain the same Subject DN and SAN information. It is best practice to generate a new key pair and CSR, i.e., re-key, when renewing a certificate, but re-keying is not required by all certificate authorities. Renewal is typically driven by the expiration of the existing certificate but could also be triggered by a suspected private key compromise or other event requiring the existing certificate to be revoked.)
Replace	The process of installing a new certificate and removing an existing one so that the new certificate is used in place of the existing certificate on all systems where the existing certificate is being used.
Representational State Transfer	A software architectural style that defines a common method for defining APIs for Web services.
Risk Management Framework	The Risk Management Framework (RMF), presented in NIST SP 800-37, provides a disciplined and structured process that integrates information security and risk management activities into the system development life cycle.
Rivest, Shamir, & Adleman	An algorithm approved in [FIPS 186] for digital signatures and in [SP 800-56B] for key establishment. (NIST SP 800-57 Part 1 Rev. 4)
Root certificate	A self-signed certificate, as defined by IETF RFC 5280, issued by a root CA. A root certificate is typically securely installed on systems so they can verify end-entity certificates they receive.
Root certificate authority	In a hierarchical public key infrastructure (PKI), the CA whose public key serves as the most trusted datum (i.e., the beginning of trust paths) for a security domain. (NIST SP 800-32)

Rotate	The process of renewing a certificate in conjunction with a rekey, followed by the process of replacing the existing certificate with the new certificate.
Subject Alternative Name	A field in an X.509 certificate that identifies one or more fully qualified domain names, IP addresses, email addresses, URIs, or UPNs to be associated with the public key contained in a certificate.
Simple Certificate Enrollment Protocol	A protocol defined in an IETF internet draft specification that is used by numerous manufacturers of network equipment and software who are developing simplified means of handling certificates for large-scale implementation to everyday users, as well as referenced in other industry standards.
Secure Hash Algorithm 1	A hash function specified in FIPS 180-2, the Secure Hash Standard. (NIST SP 800-89)
Secure Hash Algorithm 256	A hash algorithm that can be used to generate digests of messages. The digests are used to detect whether messages have been changed since the digests were generated. (FIPS 180-4 (March 2012))
Secure Transport	Transfer of information using a transport layer protocol that provides security between applications communicating over an IP network.
Server	A computer or device on a network that manages network resources. Examples include file servers (to store files), print servers (to manage one or more printers), network servers (to manage network traffic), and database servers (to process database queries). (NIST SP 800-47)
Service Provider	A provider of basic services or value-added services for operation of a network; generally refers to public carriers and other commercial enterprises. (NISTIR 4734)
Simple Mail Transfer Protocol	The primary protocol used to transfer electronic mail messages on the internet. (NISTIR 7387)
Special Publication	A type of publication issued by NIST. Specifically, the Special Publication 800-series reports on the Information Technology Laboratory's research, guidelines, and outreach efforts in computer

	security, and its collaborative activities with industry, government, and academic organizations. The 1800 series reports the results of NCCoE demonstration projects.
System Administrator	Individual responsible for the installation and maintenance of an information system, providing effective information system utilization, adequate security parameters, and sound implementation of established Information Assurance policy and procedures. (CNSSI 4009-2015)
Team	A number of persons associated together in work or activity. (Merriam Webster) As used in this publication, a team is a group of individuals that has been assigned by an organization's management the responsibility and capability to carry out a defined function or set of defined functions. Designations for teams as used in this publication are simply descriptive. Different organizations may have different designations for teams that carry out the functions described herein.
Transport Layer Security (TLS)	An authentication and security protocol widely implemented in browsers and web servers. TLS is defined by RFC 5246 and RFC 8446.
Trust Protection Platform	The Venafi Machine Identity Protection platform used in the example implementation described in this practice guide.
User Principal Name	In Windows Active Directory, this is the name of a system user in email address format, i.e., a concatenation of username, the "@" symbol, and domain name.
Validation	The process of determining that an object or process is acceptable according to a pre-defined set of tests and the results of those tests. (NIST SP 800-152)
Web Browser	A software program that allows a user to locate, access, and display *web* pages.

Appendix C Mapping to the Cybersecurity Framework

The following table maps the recommended best practices for TLS server certificate management to the NIST Cybersecurity Framework.

Table 1 Mapping the Recommended Best Practices for TLS Server Certificate Management to the Cybersecurity Framework

CSF Function	CSF Subcategory	Applicability to TLS Server Certificates
Identify	**ID.AM-2**: Software platforms and applications within the organization are inventoried	An inventory of TLS server certificates is established and maintained—including certificate attributes and metadata, such as the certificate owner for each certificate.
	ID.AM-6: Cybersecurity roles and responsibilities for the entire workforce and third-party stakeholders (e.g., suppliers, customers, partners) are established	The responsibilities for complying with TLS Server Certificate policies and maintaining operational integrity and security related to TLS server certificates are clearly defined for certificate owners, the Certificate Services Team, and other relevant stakeholders. (See NIST SP 1800-16b: Security Risks and Recommended Best Practices, Section 5.1)
	ID.GV-1: Organizational cybersecurity policy is established and communicated	TLS server certificate policies are established, communicated to all stakeholders, enforced, and audited. (See NIST SP 1800-16b: Security Risks and Recommended Best Practices, Section 5)
	ID.GV-2: Cybersecurity roles and responsibilities are coordinated and aligned with	certificate owners, the Certificate Services Team, and any other applicable stakeholders are educated on

	internal roles and external partners	and have agreed to their roles and responsibilities for ensuring TLS server certificate policy compliance and maintaining operational integrity and security related to TLS server certificates. (See NIST SP 1800-16b: Security Risks and Recommended Best Practices)
	ID.GV-3: Legal and regulatory requirements regarding cybersecurity, including privacy and civil liberties obligations, are understood and managed	The impact of applicable legal and regulatory requirements on TLS server certificate policies and processes is reviewed. Necessary adjustments to policies and processes are completed and communicated. (See NIST SP 1800-16b: Security Risks and Recommended Best Practices)
	ID.GV-4: Governance and risk management processes address cybersecurity risks	The effectiveness of implementing and complying with TLS server certificate policies to address operational and security risks is regularly reviewed by management and auditors. Adjustments are made to policies and processes when deficiencies are identified. (See NIST SP 1800-16b: Security Risks and Recommended Best Practices)
Protect	**PR.AC-1**: Identities and credentials are issued, managed, verified, revoked, and audited for authorized devices, users and processes	The following are performed for TLS server certificates, which serve as machine identities: Certificates are issued by organizationally-approved certificate authorities Certificate requests are reviewed by knowledgeable persons or via approved automated processes

		An inventory of certificates is maintained Certificate owner information is kept up to date Certificate expiration dates are tracked and new certificates requested/installed prior to expiration Access to TLS private keys is limited to authorized personnel and keys are replaced when personnel with access are reassigned or terminated Certificate operation and configuration is continuously monitored All certificate/key management operations are logged Private keys are securely transferred to TLS inspection devices Certificates are revoked when a private key is suspected to have been compromised or another event occurs that may invalidate the trustworthiness of a certificate Certificate Authority Authorization (CAA) records are populated for public-facing TLS server certificates Certificate Transparency (CT) logs are monitored for fraudulent certificates
	PR.AC-4: Access permissions and authorizations are managed, incorporating the principles of least privilege and separation of duties	Access to private keys associated with TLS server certificates is limited to authorized personnel. Certificates are replaced when personnel with direct access to corresponding private keys are reassigned or terminated. Controls

		are implemented to ensure that access to certificates is only granted to personnel or systems authorized for the corresponding domains.
	PR.AC-6: Identities are proofed and bound to credentials and asserted in interactions	TLS server certificate requests are reviewed by knowledgeable personnel or via approved automated processes.
	PR.AC-7: Users, devices, and other assets are authenticated (e.g., single-factor, multi-factor) commensurate with the risk of the transaction (e.g., individuals' security and privacy risks and other organizational risks)	All servers have TLS server certificates so they can be securely authenticated by clients.
	PR.DS-1: Data-at-rest is protected	Least privileged access is enforced for TLS server private keys or, where possible, hardware security modules are used to generate, store, and protect TLS server private keys.
	PR.DS-2: Data-in-transit is protected	All servers enforce the use of TLS for communications and the corresponding TLS certificates and private keys are properly managed and secure.
	PR.DS-3: Assets are formally managed throughout removal, transfers, and disposition	Private keys associated with TLS server certificates are replaced when people who have had direct access to those keys are reassigned or terminated. Certificates are revoked when a private key is suspected to have been compromised or another event occurs that may invalidate the trustworthiness of a certificate. New certificates are requested/installed prior to expiration.

	PR.IP-2: A System Development Life Cycle to manage systems is implemented	TLS server certificate management processes effectively manage the life cycle of TLS certificates (e.g., inventory, request, replacement, revocation, etc.).
	PR.IP-3: Configuration change control processes are in place	Change control processes are defined and enforced for TLS server certificates, e.g., certificates are replaced during off-hours and are tested before going operational.
	PR.IP-9: Response plans (Incident Response and Business Continuity) and recovery plans (Incident Recovery and Disaster Recovery) are in place and managed	The system supports the replacement of large numbers of TLS server certificates and private keys in response to CA compromises, vulnerable algorithms, or cryptographic library bugs.
	PR.PT-1: Audit/log records are determined, documented, implemented, and reviewed in accordance with policy	All TLS server certificate and private key management/administrative operations can be logged to a central location and reviewed in accordance with policy.
	PR.PT-5: Mechanisms (e.g., failsafe, load balancing, hot swap) are implemented to achieve resilience requirements in normal and adverse situations	Support is provided for managing the copying and transfer of TLS certificates needed to support resilience mechanisms such as load balancing and hot swap.
	DE.AE-5: Incident alert thresholds are established	Clear thresholds are defined for: Notifications and escalations related to certificates nearing expiration (e.g., 60, 30, 15 days prior to expiration) The implementation of large-scale certificate replacement processes (e.g., suspected CA compromise triggers replacement)

	DE.CM-1: The network is monitored to detect potential cybersecurity events	TLS inspection mechanisms are implemented to monitor encrypted traffic within TLS-secured connections to ensure that malicious activity and pivoting between internal systems is detected.
Respond	**RS.AN-5**: Processes are established to receive, analyze and respond to vulnerabilities disclosed to the organization from internal and external sources (e.g. internal testing, security bulletins, or security researchers)	In response to disclosed vulnerabilities such as public certificate authority compromise, cryptographic algorithm vulnerabilities, and cryptographic library bugs and vulnerabilities, the system supports the replacement of large numbers of TLS server certificates and private keys.
	RS.MI-2: Incidents are mitigated	All certificates affected by a certificate authority compromise, algorithm vulnerability, or cryptographic library bug can be rapidly replaced.

1438

Appendix D Special Publication 800-53 Controls Applicable to Best Practices for TLS Server Certificate Management

The following table provides an explanation of how specific controls defined within 800-53 should be applied to TLS server certificate management recommended best practices.

Table 2 Application of Specific Controls to TLS Server Certificate Management Recommended Best Practices

SP 800-53 Control #	SP 800-53 Requirement	Mapping to TLS Server Certificates
AC-1	ACCESS CONTROL POLICY AND PROCEDURES Control: a. Develop, document, and disseminate to [Assignment: organization-defined personnel or roles]: 1. An access control policy that: i. Addresses purpose, scope, roles, responsibilities, management commitment, coordination among organizational entities, and compliance; and ii. Is consistent with applicable laws, Executive Orders, directives, regulations, policies, standards, and guidelines.	An access control policy is defined for TLS private keys. Private keys associated with TLS server certificates must be protected from compromise. Most TLS private keys are stored in files. Access to these files must be limited to authorized personnel. If a person with access to a private key is reassigned or terminated, the private key and certificate should be changed.
AC-5	SEPARATION OF DUTIES Control: a. Separate [Assignment: organization-defined duties of individuals]; b. Document separation of duties of individuals; and c. Define system access authorizations to support	When a certificate is requested, another party (with knowledge of the application and requester) or automated process should review and approve the request prior to certificate issuance.

SP 800-53 Control #	SP 800-53 Requirement	Mapping to TLS Server Certificates
	separation of duties. Separation of duties addresses the potential for abuse of authorized privileges and helps to reduce the risk of malevolent activity without collusion.	
AC-6	LEAST PRIVILEGE Control: Employ the principle of least privilege, allowing only authorized accesses for users (or processes acting on behalf of users) which are necessary to accomplish assigned tasks in accordance with organizational missions and business functions.	Access to private keys should only be assigned to appropriate personnel with a need-to-know. Automation should be used where possible to minimize the need for direct private key access by people.
AC-16	SECURITY AND PRIVACY ATTRIBUTES Control: a. Provide the means to associate [Assignment: organization-defined types of security and privacy attributes] having [Assignment: organization-defined security and privacy attribute values] with information in storage, in process, and/or in transmission; b. Ensure that the security and privacy attribute associations are made and retained with the information; c. Establish the permitted [Assignment: organization-defined security attributes] for [Assignment: organization-defined systems]; and	The TLS server certificate inventory should include metadata fields for all relevant security and privacy attributes for each certificate, including issuer, key length, signing algorithm, validity period, and owner.

SP 800-53 Control #	SP 800-53 Requirement	Mapping to TLS Server Certificates
	d. Determine the permitted [Assignment: organization-defined values or ranges] for each of the established security and privacy attributes.	
AT-2	AWARENESS TRAINING Control: Provide basic security and privacy awareness training to system users (including managers, senior executives, and contractors): a. As part of initial training for new users; b. When required by system changes; and c. [Assignment: organization-defined frequency] thereafter.	All certificate owners should have sufficient training to understand the best practices/policies for TLS server certificate and private key management as well as their role and responsibilities.
AU-1	AUDIT AND ACCOUNTABILITY POLICY AND PROCEDURES Control: a. Develop, document, and disseminate to [Assignment: organization-defined personnel or roles]: 1. An audit and accountability policy that: i. Addresses purpose, scope, roles, responsibilities, management commitment, coordination among organizational entities, and compliance; and ii. Is consistent with applicable laws, Executive Orders, directives, regulations, policies, standards, and guidelines; and 2. Procedures to facilitate the	Develop, document, and disseminate policies and procedures for auditing TLS server certificate management.

SP 800-53 Control #	SP 800-53 Requirement	Mapping to TLS Server Certificates
	implementation of the audit and accountability policy and the associated audit and accountability controls; b. Designate an [Assignment: organization-defined senior management official] to manage the audit and accountability policy and procedures; c. Review and update the current audit and accountability: 1. Policy [Assignment: organization-defined frequency]; and 2. Procedures [Assignment: organization-defined frequency]; d. Ensure that the audit and accountability procedures implement the audit and accountability policy and controls; and e. Develop, document, and implement remediation actions for violations of the audit and accountability policy.	
AU-2	AUDIT EVENTS Control: Verify that the system can audit the following event types: [Assignment: organization-defined auditable event types].	Ensure that all TLS certificate and private key management operations are logged, including key generation, certificate enrollment, copying of keys, and certificate issuance/renewal/replacement/revocation.
AU-3	CONTENT OF AUDIT RECORDS Control: The system generates audit records containing	Ensure that logged TLS server certificate management events contain all relevant data

SP 800-53 Control #	SP 800-53 Requirement	Mapping to TLS Server Certificates
	information that establishes what type of event occurred, when the event occurred, where the event occurred, the source of the event, the outcome of the event, and the identity of any individuals or subjects associated with the event.	needed for audits, including date/time, operation performed, identifiers for the person or system performing the operation, identifiers for the asset (e.g., certificate/key) affected, and any other relevant information.
AU-6	AUDIT REVIEW, ANALYSIS, AND REPORTING Control: Review and analyze system audit records [Assignment: organization-defined frequency] for indications of [Assignment: organization-defined inappropriate or unusual activity].	Implement regular manual and/or automated reviews to detect unauthorized TLS server certificate and private key operations.
AU-12	AUDIT GENERATION Control: a. Provide audit record generation capability for the auditable event types in AU-2 a. at [Assignment: organization-defined system components]; b. Allow [Assignment: organization-defined personnel or roles] to select which auditable event types are to be audited by specific components of the system; and c. Generate audit records for the event types defined in AU-2 d. with the content in AU-3.	Ensure that 1) all components involved in TLS server certificate and private key management generate audit records and that the appropriate information and audit records are collected to a central log.
AU-13	MONITORING FOR INFORMATION DISCLOSURE Control: Monitor [Assignment:	Monitor the internet for rogue installations of TLS certificates

SP 800-53 Control #	SP 800-53 Requirement	Mapping to TLS Server Certificates
	organization-defined open source information and/or information sites] [Assignment: organization-defined frequency] for evidence of unauthorized disclosure of organizational information.	(which can indicate private key compromise).
CA-1	ASSESSMENT, AUTHORIZATION, AND MONITORING POLICY AND PROCEDURES Control: a. Develop, document, and disseminate to [Assignment: organization-defined personnel or roles]: 1. A security and privacy assessment, authorization, and monitoring policy that: i. Addresses purpose, scope, roles, responsibilities, management commitment, coordination among organizational entities, and compliance; and ii. Is consistent with applicable laws, Executive Orders, directives, regulations, policies, standards, and guidelines; and 2. Procedures to facilitate the implementation of the security and privacy assessment, authorization, and monitoring policy and the associated security and privacy assessment, authorization, and monitoring controls; b. Designate an [Assignment: organization-defined senior	Establish clear policies and responsibilities for TLS server certificate management. Ensure that all certificate owners and the certificate services team are educated and understand their responsibilities.

SP 800-53 Control #	SP 800-53 Requirement	Mapping to TLS Server Certificates
	management official] to manage the security and privacy assessment, authorization, and monitoring policy and procedures; c. Review and update the current security and privacy assessment, authorization, and monitoring: 1. Policy [Assignment: organization-defined frequency]; and 2. Procedures [Assignment: organization-defined frequency]; d. Ensure that the security and privacy assessment, authorization, and monitoring procedures implement the security and privacy assessment, authorization, and monitoring policy and controls; and e. Develop, document, and implement remediation actions for violations of security and privacy assessment, authorization, and monitoring policy.	
CA-2	ASSESSMENTS Control: a. Develop a security and privacy assessment plan that describes the scope of the assessment including: 1. Security and privacy controls and control enhancements under assessment; 2. Assessment procedures to be used to determine control effectiveness; and 3. Assessment environment,	Develop a security assessment plan to verify that TLS server certificate policies are followed. Ensure that an executive with sufficient authority is assigned to review and assess the current policy compliance status and posture of the TLS server certificate management program (e.g., do all groups have an up-to-date inventory, is ownership information kept up

SP 800-53 Control #	SP 800-53 Requirement	Mapping to TLS Server Certificates
	assessment team, and assessment roles and responsibilities.	to date, are private keys secured, is automation used wherever possible, etc.).
CA-5	PLAN OF ACTION AND MILESTONES Control: a. Develop a plan of action and milestones for the system to document the planned remedial actions of the organization to correct weaknesses or deficiencies noted during the assessment of the controls and to reduce or eliminate known vulnerabilities in the system; and b. Update existing plan of action and milestones [Assignment: organization-defined frequency] based on the findings from control assessments, impact analyses, and continuous monitoring activities.	Establish a remediation plan to address deficiencies. Ensure executive oversight. Regularly review progress on the achievement of milestones and provide executive support where needed to ensure sufficient resources to meet milestones.
CA-7	CONTINUOUS MONITORING Control: Develop a security and privacy continuous monitoring strategy and implement security and privacy continuous monitoring programs that include: a. Establishing the following security and privacy metrics to be monitored: [Assignment: organization-defined metrics]; b. Establishing [Assignment: organization-defined frequencies] for monitoring and [Assignment: organization-defined frequencies] for ongoing assessment of security	Implement continuous monitoring for all TLS server certificates, including: • Regular automated network discovery scans to detect newly deployed certificates • Monitoring certificate expiration dates • Automated checking that all known certificates are correctly installed and operational • Tracking of CT records for fraudulent certificates.

SP 800-53 Control #	SP 800-53 Requirement	Mapping to TLS Server Certificates
	and privacy control effectiveness; c. Ongoing security and privacy control assessments in accordance with the organizational continuous monitoring strategy; d. Ongoing security and privacy status monitoring of organization-defined metrics in accordance with the organizational continuous monitoring strategy; e. Correlation and analysis of security- and privacy-related information generated by security and privacy control assessments and monitoring; f. Response actions to address results of the analysis of security- and privacy-related information; and g. Reporting the security and privacy status of the organization and organizational systems to [Assignment: organization-defined personnel or roles] [Assignment: organization-defined frequency].	Ensure that encrypted TLS sessions can be monitored for malicious activity via proxy, endpoint agent, or passive decryption.
CM-2	BASELINE CONFIGURATION Control: a. Develop, document, and maintain under configuration control, a current baseline configuration of the system; and b. Review and update the baseline configuration of the system.	Perform automated network discovery scans to establish a comprehensive baseline of the TLS server certificate inventory. Review and update baseline configuration.
CM-3	CONFIGURATION CHANGE CONTROL Control:	Ensure that certificate replacement operations are included in change control

SP 800-53 Control #	SP 800-53 Requirement	Mapping to TLS Server Certificates
	a. Determine the types of changes to the system that are configuration-controlled; b. Review proposed configuration-controlled changes to the system and approve or disapprove such changes with explicit consideration for security impact analyses; c. Document configuration change decisions associated with the system; d. Implement approved configuration-controlled changes to the system; e. Retain records of configuration-controlled changes to the system for [Assignment: organization-defined time-period]; f. Monitor and review activities associated with configuration-controlled changes to the system.	plans. Ensure all certificate management operations are scheduled and reviewed. Retain logs of all certificate management operations.
CM-6	CONFIGURATION SETTINGS Control: Establish and document configuration settings for components employed within the system using [Assignment: organization-defined common secure configurations] that reflect the most restrictive mode consistent with operational requirements.	Establish and document the following for TLS server certificates: - Key lengths - Signing algorithms - Certificate authorities - Validity periods - Private key access control and protection
CM-8	SYSTEM COMPONENT INVENTORY Control: a. Develop and document an inventory of system components	Ensure that a comprehensive TLS server certificate inventory

SP 800-53 Control #	SP 800-53 Requirement	Mapping to TLS Server Certificates
	that: 1. Accurately reflects the current system; 2. Includes all components within the authorization boundary of the system; 3. Is at the level of granularity deemed necessary for tracking and reporting; and 4. Includes [Assignment: organization-defined information deemed necessary to achieve effective system component accountability]; and b. Review and update the system component inventory [Assignment: organization-defined frequency].	is established and maintained, including: • Metadata • Installed locations • Owners
CM-12	INFORMATION LOCATION Control: a. Identify the location of [Assignment: organization-defined information] and the specific system components on which the information resides; b. Identify and document the users who have access to the system and system components where the information resides; and c. Document changes to the location (i.e., system or system components) where the information resides.	Identify the location of all TLS certificates and private keys . Identify and document and keep up to date information about all certificate owners and System Administrators. Identify and document and keep up-to-date-information about the location of private keys.
CP-2	CONTINGENCY PLAN Control: a. Develop a contingency plan for	Establish “crypto-agility” plans for the replacement of TLS server certificates in response

SP 800-53 Control #	SP 800-53 Requirement	Mapping to TLS Server Certificates
	the system that: 1. Identifies essential missions and business functions and associated contingency requirements; 2. Provides recovery objectives, restoration priorities, and metrics; 3. Addresses contingency roles, responsibilities, assigned individuals with contact information; 4. Addresses maintaining essential missions and business functions despite a system disruption, compromise, or failure; 5. Addresses eventual, full system restoration without deterioration of the security and privacy controls originally planned and implemented; and 6. Is reviewed and approved by [Assignment: organization-defined personnel or roles]; b. Distributes copies of the contingency plan to [Assignment: organization-defined key contingency personnel (identified by name and/or by role) and organizational elements]; c. Coordinates contingency planning activities with incident handling activities; d. Reviews the contingency plan for the system [Assignment: organization-defined frequency]; e. Updates the contingency plan to address changes to the organization, system, or	to a CA compromise, discovered algorithm vulnerability, discovered cryptographic bug, or compromised private keys.

SP 800-53 Control #	SP 800-53 Requirement	Mapping to TLS Server Certificates
	environment of operation and problems encountered during contingency plan implementation, execution, or testing; f. Communicates contingency plan changes to [Assignment: organization-defined key contingency personnel (identified by name and/or by role) and organizational elements]; and g. Protects the contingency plan from unauthorized disclosure and modification.	
CP-3	CONTINGENCY TRAINING Control: Provide contingency training to system users consistent with assigned roles and responsibilities: a. Within [Assignment: organization-defined time-period] of assuming a contingency role or responsibility; b. When required by system changes; and c. [Assignment: organization-defined frequency] thereafter.	Ensure all certificate owners are trained and understand their responsibilities in TLS server certificate crypto-agility plans.
CP-4	CONTINGENCY PLAN TESTING Control: a. Test the contingency plan for the system [Assignment: organization-defined frequency] using [Assignment: organization-defined tests] to determine the effectiveness of the plan and the organizational readiness to execute the plan;	Ensure that TLS server certificate crypto-agility plans are regularly tested.

SP 800-53 Control #	SP 800-53 Requirement	Mapping to TLS Server Certificates
	b. Review the contingency plan test results; and c. Initiate corrective actions, if needed.	
CP-13	ALTERNATIVE SECURITY MECHANISMS Control: Employ [Assignment: organization-defined alternative or supplemental security mechanisms] for satisfying [Assignment: organization-defined security functions] when the primary means of implementing the security function is unavailable or compromised.	Ensure that backup certificate authorities (CAs) are maintained, including maintaining contracts with backup public CAs.
IA-3	DEVICE IDENTIFICATION AND AUTHENTICATION Control: Uniquely identify and authenticate [Assignment: organization-defined specific and/or types of devices] before establishing a [Selection (one or more): local; remote; network] connection.	Ensure that all TLS servers have certificates for authentication. Ensure that all TLS clients properly validate TLS server certificates when establishing TLS connections
IA-4	IDENTIFIER MANAGEMENT Control: Manage system identifiers by: a. Receiving authorization from [Assignment: organization-defined personnel or roles] to assign an individual, group, role, or device identifier; b. Selecting an identifier that identifies an individual, group, role, or device;	Ensure that all TLS server certificate requests are reviewed by a person with relevant knowledge of the application in question or via an approved automated process to verify that the common names (CNs) and subject alternative names (SANs) that serve as identifiers in TLS server

SP 800-53 Control #	SP 800-53 Requirement	Mapping to TLS Server Certificates
	c. Assigning the identifier to the intended individual, group, role, or device; and d. Preventing reuse of identifiers for [Assignment: organization-defined time-period].	certificates are vetted before issuance.
IA-5	AUTHENTICATOR MANAGEMENT Control: Manage system authenticators by: a. Verifying, as part of the initial authenticator distribution, the identity of the individual, group, role, or device receiving the authenticator; b. Establishing initial authenticator content for any authenticators issued by the organization; c. Ensuring that authenticators have sufficient strength of mechanism for their intended use; d. Establishing and implementing administrative procedures for initial authenticator distribution, for lost/compromised or damaged authenticators, and for revoking authenticators; e. Establishing minimum and maximum lifetime restrictions and reuse conditions for authenticators; f. Changing/refreshing authenticators [Assignment: organization-defined time-period by authenticator type]; g. Protecting authenticator content from unauthorized	Ensure TLS server certificates, which serve as authenticators for servers, are properly managed, including: - An up to date inventory - Up to date ownership information - Secure private key handling and distribution - Sufficient key length and strong signing algorithms - Appropriate reviews for certificate requests - Replacement of certificates and keys on role changes and termination - Continuous monitoring -

SP 800-53 Control #	SP 800-53 Requirement	Mapping to TLS Server Certificates
	disclosure and modification; h. Requiring individuals to take, and having devices implement, specific security controls to protect authenticators; and i. Changing authenticators for group/role accounts when membership to those accounts' changes.	
IA-9	SERVICE IDENTIFICATION AND AUTHENTICATION Control: Identify and authenticate [Assignment: organization-defined system services and applications] before establishing communications with devices, users, or other services or applications.	Use TLS server certificates for identification and authentication on all servers where TLS is the appropriate security protocol to secure communications (e.g., to secure HTTP, SMTP, LDAP, FTP, etc.).
IR-1	INCIDENT RESPONSE POLICY AND PROCEDURES Control: a. Develop, document, and disseminate to [Assignment: organization-defined personnel or roles]: 1. An incident response policy that: i. Addresses purpose, scope, roles, responsibilities, management commitment, coordination among organizational entities, and compliance; and ii. Is consistent with applicable laws, Executive Orders, directives, regulations, policies, standards, and guidelines; and	Document and disseminate TLS server certificate incident response plans for the following: - Certificate authority compromises - Cryptographic algorithms found to be vulnerable - Cryptographic library bugs that affect cryptographic keys and certificates - Compromise of one or more private keys that are associated with certificates - Compromise of the certificate management system itself

SP 800-53 Control #	SP 800-53 Requirement	Mapping to TLS Server Certificates
	2. Procedures to facilitate the implementation of the incident response policy and the associated incident response controls; b. Designate an [Assignment: organization-defined senior management official] to manage the incident response policy and procedures; c. Review and update the current incident response: 1. Policy [Assignment: organization-defined frequency]; and 2. Procedures [Assignment: organization-defined frequency]; d. Ensure that the incident response procedures implement the incident response policy and controls; and e. Develop, document, and implement remediation actions for violations of the incident response policy.	
IR-2	INCIDENT RESPONSE TRAINING Control: Provide incident response training to system users consistent with assigned roles and responsibilities: a. Within [Assignment: organization-defined time-period] of assuming an incident response role or responsibility.	Ensure all certificate owners are trained and understand their responsibilities in TLS server certificate incident response plans.

SP 800-53 Control #	SP 800-53 Requirement	Mapping to TLS Server Certificates
IR-3	INCIDENT RESPONSE TESTING Control: Test the incident response capability for the system [Assignment: organization-defined frequency] using [Assignment: organization-defined tests] to determine the incident response effectiveness and documents the results.	Ensure that TLS server certificate incident response plans are tested.
IR-4	INCIDENT HANDLING Control: a. Implement an incident handling capability for security and privacy incidents that includes preparation, detection and analysis, containment, eradication, and recovery; b. Coordinate incident handling activities with contingency planning activities; c. Incorporate lessons learned from ongoing incident handling activities into incident response procedures, training, and testing, and implement the resulting changes accordingly; and d. Ensure the rigor, intensity, scope, and results of incident handling activities are comparable and predictable across the organization.	• Document and disseminate TLS server certificate incident response plans for the following: Certificate authority compromises • Cryptographic algorithms found to be vulnerable • Cryptographic library bugs that affect cryptographic keys and certificates • Compromise of one or more private keys that are associated with certificates • Compromise of the certificate management system itself
MA-1	SYSTEM MAINTENANCE POLICY AND PROCEDURES Control: a. Develop, document, and disseminate to [Assignment:	Establish TLS server certificate maintenance policies and procedures, including purpose, scope, roles, responsibilities,

SP 800-53 Control #	SP 800-53 Requirement	Mapping to TLS Server Certificates
	organization-defined personnel or roles]: 1. A system maintenance policy that: i. Addresses purpose, scope, roles, responsibilities, management commitment, coordination among organizational entities, and compliance; and ii. Is consistent with applicable laws, Executive Orders, directives, regulations, policies, standards, and guidelines; and 2. Procedures to facilitate the implementation of the system maintenance policy and the associated system maintenance controls; b. Designate an [Assignment: organization-defined senior management official] to manage the system maintenance policy and procedures; c. Review and update the current system maintenance: 1. Policy [Assignment: organization-defined frequency]; and 2. Procedures [Assignment: organization-defined frequency]; d. Ensure that the system maintenance procedures implement the system maintenance policy and controls; and e. Develop, document, and implement remediation actions for	management commitment, coordination, and compliance.

SP 800-53 Control #	SP 800-53 Requirement	Mapping to TLS Server Certificates
	violations of the maintenance policy.	
MA-6	TIMELY MAINTENANCE Control: Obtain maintenance support and/or spare parts for [Assignment: organization-defined system components] within [Assignment: organization-defined time-period] of failure.	Ensure that certificates are renewed and replaced a sufficient number of days prior to expiration to minimize downtime risk.
PL-2	SECURITY AND PRIVACY PLANS Control: a. Develop security and privacy plans for the system that: 1. Are consistent with the organization's enterprise architecture; 2. Explicitly define the authorization boundary for the system; 3. Describe the operational context of the system in terms of missions and business processes; 4. Provide the security categorization of the system including supporting rationale; 5. Describe the operational environment for the system and relationships with or connections to other systems; 6. Provide an overview of the security and privacy requirements for the system; 7. Identify any relevant overlays, if applicable; 8. Describe the security and privacy controls in place or	Develop security plans for TLS private keys to ensure they are consistent with the security plans for other secrets such as passwords and keys for symmetric-key encryption.

SP 800-53 Control #	SP 800-53 Requirement	Mapping to TLS Server Certificates
	planned for meeting those requirements including a rationale for the tailoring decisions; and 9. Are reviewed and approved by the authorizing official or designated representative prior to plan implementation; b. Distribute copies of the security and privacy plans and communicate subsequent changes to the plans to [Assignment: organization-defined personnel or roles]; c. Review the security and privacy plans [Assignment: organization-defined frequency]; d. Update the security and privacy plans to address changes to the system and environment of operation or problems identified during plan implementation or security and privacy control assessments; and e. Protect the security and privacy plans from unauthorized disclosure and modification.	
PL-9	CENTRAL MANAGEMENT Control: Centrally manage [Assignment: organization-defined security and privacy controls and related processes].	Establish a central certificate service that enables central oversight and monitoring. Define clear TLS server certificate management responsibilities for the certificate services team and certificate owners.
PM-1	INFORMATION SECURITY PROGRAM PLAN	Develop and disseminate an information security program

SP 800-53 Control #	SP 800-53 Requirement	Mapping to TLS Server Certificates
	Control: a. Develop and disseminate an organization-wide information security program plan that: 1. Provides an overview of the requirements for the security program and a description of the security program management controls and common controls in place or planned for meeting those requirements; 2. Includes the identification and assignment of roles, responsibilities, management commitment, coordination among organizational entities, and compliance; 3. Reflects the coordination among organizational entities responsible for information security; and 4. Is approved by a senior official with responsibility and accountability for the risk being incurred to organizational operations (including mission, functions, image, and reputation), organizational assets, individuals, other organizations, and the Nation; b. Review the organization-wide information security program plan [Assignment: organization-defined frequency]; c. Update the information security program plan to address organizational changes and	plan that includes the following for TLS server certificates: - Requirements for proper management - Roles and responsibilities - Coordination between the certificate services team and certificate owners

SP 800-53 Control #	SP 800-53 Requirement	Mapping to TLS Server Certificates
	problems identified during plan implementation or control assessments; and d. Protect the information security program plan from unauthorized disclosure and modification.	
PM-2	INFORMATION SECURITY PROGRAM ROLES Control: a. Appoint a Senior Agency Information Security Officer with the mission and resources to coordinate, develop, implement, and maintain an organization-wide information security program; b. Appoint a Senior Accountable Official for Risk Management to align information security management processes with strategic, operational, and budgetary planning processes; and c. Appoint a Risk Executive (function) to view and analyze risk from an organization-wide perspective and ensure management of risk is consistent across the organization.	Appoint a senior executive with the mission of ensuring TLS server certificates are properly managed to minimize security and operational risks.
PM-4	PLAN OF ACTION AND MILESTONES PROCESS Control: a. Implement a process to ensure that plans of action and milestones for the security and privacy programs and associated organizational systems: 1. Are developed and maintained;	Establish actions and milestones for implementing and deploying the TLS server certificate information security program plan. Ensure regular reviews of progress and status are performed.

SP 800-53 Control #	SP 800-53 Requirement	Mapping to TLS Server Certificates
	2. Document the remedial information security and privacy actions to adequately respond to risk to organizational operations and assets, individuals, other organizations, and the Nation; and 3. Are reported in accordance with established reporting requirements. b. Review plans of action and milestones for consistency with the organizational risk management strategy and organization-wide priorities for risk response actions.	
PM-5	SYSTEM INVENTORY Control: Develop and maintain an inventory of organizational systems.	Ensure that a comprehensive TLS server certificate inventory is established and maintained, including: • Metadata • Installed locations Owners
PM-7	ENTERPRISE ARCHITECTURE Control: Develop an enterprise architecture with consideration for information security, privacy, and the resulting risk to organizational operations and assets, individuals, other organizations, and the Nation.	Establish an enterprise architecture that enables the monitoring of communications within TLS encrypted sessions for attacks (Inspect TLS traffic on sessions between external and internal devices as well as sessions between internal devices).
PM-9	RISK MANAGEMENT STRATEGY Control: a. Develops a comprehensive strategy to manage:	Ensure the following risks are addressed in the Risk

SP 800-53 Control #	SP 800-53 Requirement	Mapping to TLS Server Certificates
	1. Security risk to organizational operations and assets, individuals, other organizations, and the Nation associated with the operation and use of organizational systems; 2. Privacy risk to individuals resulting from the collection, sharing, storing, transmitting, use, and disposal of personally identifiable information; and 3. Supply chain risks associated with the development, acquisition, maintenance, and disposal of systems, system components, and system services; b. Implement the risk management strategy consistently across the organization; and c. Review and update the risk management strategy [Assignment: organization-defined frequency] or as required, to address organizational changes.	Management Strategy for TLS server certificates: • Outages due to certificate expirations • Undetected pivoting between systems within TLS encrypted connections • Outages or disclosure of information that could result from an inability to rapidly change large numbers of certificates and keys in response to a large-scale cryptographic event • Discloser of private keys that could result from manual key transfer • Disclosure of information that could result from an adversary installing a rogue server certificate • Disclosure of information that could result from trusting a bogus certificate or unapproved certificate authority • Disclosure of information that could result from using an improperly configured certificate, a vulnerable cryptographic algorithm or an insufficiently long key
RA-3	RISK ASSESSMENT Control: a. Conduct a risk assessment, including the likelihood and magnitude of harm, from:	Ensure the following TLS server certificates risks are included in the Risk Assessment:

SP 800-53 Control #	SP 800-53 Requirement	Mapping to TLS Server Certificates
	1. The unauthorized access, use, disclosure, disruption, modification, or destruction of the system, the information it processes, stores, or transmits, and any related information; and 2. Privacy-related problems for individuals arising from the intentional processing of personally identifiable information; b. Integrate risk assessment results and risk management decisions from the organization and missions/business process perspectives with system-level risk assessments; c. Document risk assessment results in [Selection: security and privacy plans; risk assessment report; [Assignment: organization-defined document]]; d. Review risk assessment results [Assignment: organization-defined frequency]; e. Disseminate risk assessment results to [Assignment: organization-defined personnel or roles]; and f. Update the risk assessment [Assignment: organization-defined frequency] or when there are significant changes to the system, its environment of operation, or other conditions that may impact	• Outages due to certificate expirations • Undetected pivoting between systems within TLS encrypted connections • Outages or disclosure of information that could result from an inability to rapidly change large numbers of certificates and keys in response to a large-scale cryptographic events. • Discloser of private keys that could result from manual key transfer • Disclosure of information that could result from an adversary installing a rogue server certificate • Disclosure of information that could result from trusting a bogus certificate or unapproved certificate authority • Disclosure of information that could result from using an improperly configured certificate, vulnerable cryptographic algorithm or an insufficiently long key

SP 800-53 Control #	SP 800-53 Requirement	Mapping to TLS Server Certificates
	the security or privacy state of the system.	
RA-5	VULNERABILITY SCANNING Control: a. Scan for vulnerabilities in the system and hosted applications [Assignment: organization-defined frequency and/or randomly in accordance with organization-defined process] and when new vulnerabilities potentially affecting the system are identified and reported; b. Employ vulnerability scanning tools and techniques that facilitate interoperability among tools and automate parts of the vulnerability management process by using standards for: 1. Enumerating platforms, software flaws, and improper configurations; 2. Formatting checklists and test procedures; and 3. Measuring vulnerability impact; c. Analyze vulnerability scan reports and results from control assessments; d. Remediate legitimate vulnerabilities [Assignment: organization-defined response times] in accordance with an organizational assessment of risk; e. Share information obtained from the vulnerability scanning process and control assessments	Scan for vulnerabilities in TLS server certificates, including: • Improperly configured certificates • Weak key lengths • Vulnerable cryptographic algorithms • Unapproved certificate authorities • Validity periods that exceed approved maximums

SP 800-53 Control #	SP 800-53 Requirement	Mapping to TLS Server Certificates
	with [Assignment: organization-defined personnel or roles] to help eliminate similar vulnerabilities in other systems; and f. Employ vulnerability scanning tools that include the capability to readily update the vulnerabilities to be scanned.	
RA-7	RISK RESPONSE Control: Respond to findings from security and privacy assessments, monitoring, and audits.	Respond to findings from security and privacy assessments, monitoring, and audits for TLS server certificates and related system components.
SA-1	SYSTEM AND SERVICES ACQUISITION POLICY AND PROCEDURES Control: a. Develop, document, and disseminate to [Assignment: organization-defined personnel or roles]: 1. A system and services acquisition policy that: i. Addresses purpose, scope, roles, responsibilities, management commitment, coordination among organizational entities, and compliance; and ii. Is consistent with applicable laws, Executive Orders, directives, regulations, policies, standards, and guidelines; and 2. Procedures to facilitate the implementation of the system and services acquisition policy and the	Designate approved public and internal CAs from which TLS server certificates may be acquired and used. Designate approved TLS Server Certificate Management components that can be acquired and used, e.g. central certificate service software, HSMs, TLS inspection appliances.

SP 800-53 Control #	SP 800-53 Requirement	Mapping to TLS Server Certificates
	associated system and services acquisition controls; b. Designate an [Assignment: organization-defined senior management official] to manage the system and services acquisition policy and procedures; c. Review and update the current system and services acquisition: 1. Policy [Assignment: organization-defined frequency]; and 2. Procedures [Assignment: organization-defined frequency]; d. Ensure that the system and services acquisition procedures implement the system and services acquisition policy and controls; and e. Develop, document, and implement remediation actions for violations of the system and services acquisition policy. Designate approved public CAs from which TLS server certificates can be acquired.	
SA-3	SYSTEM DEVELOPMENT LIFE CYCLE Control: a. Manage the system using [Assignment: organization-defined system development life cycle] that incorporates information security and privacy considerations; b. Define and document information security and privacy	Define and document clear lifecycle management processes and responsibilities for TLS server certificates.

SP 800-53 Control #	SP 800-53 Requirement	Mapping to TLS Server Certificates
	roles and responsibilities throughout the system development life cycle; c. Identify individuals having information security and privacy roles and responsibilities; and d. Integrate the organizational information security and privacy risk management process into system development life cycle activities.	
SA-4	ACQUISITION PROCESS Control: Include the following requirements, descriptions, and criteria, explicitly or by reference, in the acquisition contract for the system, system component, or system service: a. Security and privacy functional requirements; b. Strength of mechanism requirements; c. Security and privacy assurance requirements; d. Security and privacy documentation requirements; e. Requirements for protecting security and privacy documentation; f. Description of the system development environment and environment in which the system is intended to operate; g. Allocation of responsibility or identification of parties responsible for information	Enforce the criteria in requirements a. through g. in acquisition contracts with public certificate authorities.

SP 800-53 Control #	SP 800-53 Requirement	Mapping to TLS Server Certificates
	security, privacy, and supply chain risk management; and h. Acceptance criteria.	
SA-10	DEVELOPER CONFIGURATION MANAGEMENT Control: Require the developer of the system, system component, or system service to: a. Perform configuration management during system, component, or service [Selection (one or more): design; development; implementation; operation; disposal]; b. Document, manage, and control the integrity of changes to [Assignment: organization-defined configuration items under configuration management]; c. Implement only organization-approved changes to the system, component, or service; d. Document approved changes to the system, component, or service and the potential security and privacy impacts of such changes; and e. Track security flaws and flaw resolution within the system, component, or service and report findings to [Assignment: organization-defined personnel].	Ensure that developers who leverage TLS server certificates in their developed systems (e.g., DevOps) follow TLS server certificate management policies and procedures. Ensure that system administrators that are responsible for installation and configuration of TLS management components such as the central certificate service software, HSMs, and TLS inspection appliances follow TLS server certificate management policies when initially configuring these components. Ensure that all configuration changes are approved an also conform to policies.
SC-1	SYSTEM AND COMMUNICATIONS PROTECTION POLICY AND PROCEDURES Control:	Ensure that secure management of TLS server certificates and private keys is incorporated into

SP 800-53 Control #	SP 800-53 Requirement	Mapping to TLS Server Certificates
	a. Develop, document, and disseminate to [Assignment: organization-defined personnel or roles]: 1. A system and communications protection policy that: i. Addresses purpose, scope, roles, responsibilities, management commitment, coordination among organizational entities, and compliance; and ii. Is consistent with applicable laws, Executive Orders, directives, regulations, policies, standards, and guidelines; and 2. Procedures to facilitate the implementation of the system and communications protection policy and the associated system and communications protection controls; b. Designate an [Assignment: organization-defined senior management official] to manage the system and communications protection policy and procedures; c. Review and update the current system and communications protection: 1. Policy [Assignment: organization-defined frequency]; and 2. Procedures [Assignment: organization-defined frequency]; d. Ensure that the system and communications protection procedures implement the system	Communications Protection Policy and Procedures. Ensure that protection of TLS server certificate management components, e.g., central certificate management service software, HSMs, TLS inspection appliances, is incorporated into Systems Protection Policy and Procedures.

SP 800-53 Control #	SP 800-53 Requirement	Mapping to TLS Server Certificates
	and communications protection policy and controls; and e. Develop, document, and implement remediation actions for violations of the system and communications protection policy.	
SC-8	TRANSMISSION CONFIDENTIALITY AND INTEGRITY Control: Protect the [Selection (one or more): confidentiality; integrity] of transmitted information.	Leverage TLS in the protecting the integrity and confidentiality of transmitted information. Implement secure management of TLS server certificates and private keys to ensure the secure operation of TLS.
SC-12	CRYPTOGRAPHIC KEY ESTABLISHMENT AND MANAGEMENT Control: Establish and manage cryptographic keys for required cryptography employed within the system in accordance with [Assignment: organization-defined requirements for key generation, distribution, storage, access, and destruction].	Establish and manage TLS private keys in compliance with requirements in NIST SP 800-57 and SP 1800-16B.
SC-17	PUBLIC KEY INFRASTRUCTURE CERTIFICATES Control: Issue public key certificates under an [Assignment: organization-defined certificate policy] or obtain public key certificates from an approved service provider.	Document, publish, communicate, and enforce clear policies for TLS server certificate issuance and management.

SP 800-53 Control #	SP 800-53 Requirement	Mapping to TLS Server Certificates
SC-23	SESSION AUTHENTICITY Control: Protect the authenticity of communications sessions.	Use TLS server certificates to authenticate servers.
SI-4	SYSTEM MONITORING Control: a. Monitor the system to detect: 1. Attacks and indicators of potential attacks in accordance with [Assignment: organization-defined monitoring objectives]; and 2. Unauthorized local, network, and remote connections; b. Identify unauthorized use of the system through [Assignment: organization-defined techniques and methods]; c. Invoke internal monitoring capabilities or deploy monitoring devices: 1. Strategically within the system to collect organization-determined essential information; and 2. At ad hoc locations within the system to track specific types of transactions of interest to the organization; d. Protect information obtained from intrusion-monitoring tools from unauthorized access, modification, and deletion; e. Adjust the level of system monitoring activity when there is a change in risk to organizational operations and assets, individuals, other organizations, or the Nation;	Monitor sessions and operations within TLS encrypted connections to detect attacks and indicators of potential attacks.

SP 800-53 Control #	SP 800-53 Requirement	Mapping to TLS Server Certificates
	f. Obtain legal opinion regarding system monitoring activities; and g. Provide [Assignment: organization-defined system monitoring information] to [Assignment: organization-defined personnel or roles] [Selection (one or more): as needed; [Assignment: organization-defined frequency]].	

Appendix E References

E. Barker, "Guideline for Using Cryptographic Standards in the Federal Government: Cryptographic Mechanisms," NIST SP 800-175B, Gaithersburg, MD, Aug. 2016. Available: https://nvlpubs.nist.gov/nistpubs/SpecialPublications/NIST.SP.800-175B.pdf.

E. Barker and A. Roginsky, "Transitions: Recommendation for Transitioning the Use of Cryptographic Algorithms and Key Lengths," National Institute of Standards and Technology (NIST) Special Publication (SP) 800-131A Revision 1, Gaithersburg, MD, Nov. 2015. Available: https://nvlpubs.nist.gov/nistpubs/SpecialPublications/NIST.SP.800-131Ar1.pdf.

D. Cooper et al., "Internet X.509 Public Key Infrastructure Certificate and Certificate Revocation List (CRL) Profile," RFC 5280, May 2008. Available: https://tools.ietf.org/html/rfc5280.

M. Crispin, "Internet Message Access Protocol – Version 4rev1," RFC 3501, Mar. 2003. Available: https://tools.ietf.org/html/rfc3501.

T. Dierks and E. Rescorla, "The Transport Layer Security (TLS) protocol version 1.2," RFC 5246, Aug. 2008. Available: https://tools.ietf.org/html/rfc5246.

Information Technology Laboratory, "Secure Hash Standard (SHS)," NIST, Federal Information Processing Standards PUB 180-4, Gaithersburg, MD, Aug. 2015. Available: https://nvlpubs.nist.gov/nistpubs/FIPS/NIST.FIPS.180-4.pdf.

J. Klensin, "Simple Mail Transfer Protocol," RFC 5321, Oct. 2008. Available: https://tools.ietf.org/html/rfc5321.

P. Mockapetris, "Domain Names – Concepts and Facilities," RFC 1034, Nov. 1987. Available: https://tools.ietf.org/html/rfc1034.

K. Moriarty et al., "PKCS #12: Personal Information Exchange Syntax v1.1," RFC 7292, July 2014. Available: https://tools.ietf.org/html/rfc7292.

J. Myers and M. Rose, "Post Office Protocol – Version 3," RFC 1725, Nov. 1994. Available: https://tools.ietf.org/html/rfc1725.

NIST Framework for Improving Critical Infrastructure Cybersecurity, Version 1.1, April 16, 2018. See https://nvlpubs.nist.gov/nistpubs/CSWP/NIST.CSWP.04162018.pdf

NIST SP 800-53 Rev. 5 (Draft) Security and Privacy Controls for Information Systems and Organizations. See https://csrc.nist.gov/publications/detail/sp/800-53/rev-5/draft

T. Polk et al., "Guidelines for the Selection, Configuration, and Use of Transport Layer Security (TLS) Implementations," NIST SP 800-52 Revision 1, Gaithersburg, MD, Apr. 2014. Available: https://nvlpubs.nist.gov/nistpubs/SpecialPublications/NIST.SP.800-52r1.pdf.

T. Pornin, "Deterministic Usage of the Digital Signature Algorithm (DSA) and Elliptic Curve Digital Signature Algorithm (ECDSA)," RFC 6979, Aug. 2013. Available: https://tools.ietf.org/html/rfc6979.

M. Pritikin et al., "Simple Certificate Enrollment Protocol draft-nourse-scep-23," Internet Draft, Sept. 7, 2011. Available: https://tools.ietf.org/html/draft-nourse-scep-23.

V. Rekhter et al., "A Border Gateway Protocol 4 (BGP-4)," RFC 4271, Jan. 2006. Available: https://tools.ietf.org/html/rfc4271.

E. Rescorla, "HTTP over TLS," RFC 2818, May 2000. Available: https://tools.ietf.org/html/rfc2818.

J. Sermersheim, "Lightweight Directory Access Protocol (LDAP): The protocol," RFC 4511, June 2006. Available: https://www.ietf.org/rfc/rfc4511.txt.

NIST SPECIAL PUBLICATION 1800-16C

Securing Web Transactions

TLS Server Certificate Management

Volume C:
Approach, Architecture, and Security Characteristics

Murugiah Souppaya
NIST

Mehwish Akram
Brian Johnson
Brett Pleasant
Susan Symington
The MITRE Corporation

Paul Turner
Venafi

William C. Barker
Dakota Consulting

July 2019

DRAFT

This publication is available free of charge from:
https://www.nccoe.nist.gov/projects/building-blocks/tls-server-certificate-management

DISCLAIMER

National Institute of Standards and Technology Special Publication 1800-16C Natl. Inst. Stand. Technol. Spec. Publ. 1800-16C, 63 pages, (July 2019), CODEN: NSPUE2

FEEDBACK

You can improve this guide by contributing feedback. As you review and adopt this solution for your own organization, we ask you and your colleagues to share your experience and advice with us.

Comments on this publication may be submitted to: tls-cert-mgmt-nccoe@nist.gov.

Public comment period: July 17, 2019 through September 13, 2019

All comments are subject to release under the Freedom of Information Act (FOIA).

National Cybersecurity Center of Excellence
National Institute of Standards and Technology
100 Bureau Drive
Mailstop 2002
Gaithersburg, MD 20899
Email: nccoe@nist.gov

NATIONAL CYBERSECURITY CENTER OF EXCELLENCE

The National Cybersecurity Center of Excellence (NCCoE), a part of the National Institute of Standards and Technology (NIST), is a collaborative hub where industry organizations, government agencies, and academic institutions work together to address businesses' most pressing cybersecurity issues. This public-private partnership enables the creation of practical cybersecurity solutions for specific industries, as well as for broad, cross-sector technology challenges. Through consortia under Cooperative Research and Development Agreements (CRADAs), including technology partners—from Fortune 50 market leaders to smaller companies specializing in information technology security—the NCCoE applies standards and best practices to develop modular, easily adaptable example cybersecurity solutions using commercially available technology. The NCCoE documents these example solutions in the NIST Special Publication 1800 series, which maps capabilities to the NIST Cybersecurity Framework and details the steps needed for another entity to re-create the example solution. The NCCoE was established in 2012 by NIST in partnership with the State of Maryland and Montgomery County, Maryland.

To learn more about the NCCoE, visit https://www.nccoe.nist.gov/. To learn more about NIST, visit https://www.nist.gov.

NIST CYBERSECURITY PRACTICE GUIDES

NIST Cybersecurity Practice Guides (Special Publication 1800 series) target specific cybersecurity challenges in the public and private sectors. They are practical, user-friendly guides that facilitate the adoption of standards-based approaches to cybersecurity. They show members of the information security community how to implement example solutions that help them align more easily with relevant standards and best practices, and provide users with the materials lists, configuration files, and other information they need to implement a similar approach.

The documents in this series describe example implementations of cybersecurity practices that businesses and other organizations may voluntarily adopt. These documents do not describe regulations or mandatory practices, nor do they carry statutory authority.

ABSTRACT

Transport Layer Security (TLS) server certificates are critical to the security of both internet-facing and private web services. A large- or medium-scale enterprise may have thousands or even tens of thousands of such certificates, each identifying a specific server in its environment. Despite the critical importance of these certificates, many organizations lack a formal TLS certificate management program, and the ability to centrally monitor and manage their certificates. Instead, certificate management tends to be spread across each of the different groups responsible for the various servers and systems in an organization. Central security teams struggle to ensure certificates are being properly managed by each of these disparate groups. Where there is no central certificate management service, the organization is at risk, because once certificates are deployed, it is necessary to maintain current inventories to support regular monitoring and certificate maintenance. Organizations that do not properly manage their certificates face significant risks to their core operations, including:

- application outages caused by expired TLS server certificates

- hidden intrusion, exfiltration, disclosure of sensitive data, or other attacks resulting from encrypted threats or server impersonation
- disaster-recovery risk that requires rapid replacement of large numbers of certificates and private keys in response to either certificate authority compromise or discovery of vulnerabilities in cryptographic algorithms or libraries

Despite the mission-critical nature of TLS server certificates, many organizations have not defined the clear policies, processes, roles, and responsibilities needed for effective certificate management. Moreover, many organizations do not leverage available automation tools to support effective management of the ever-growing numbers of certificates. The consequence is continuing susceptibility to security incidents.

This NIST Cybersecurity Practice Guide shows large and medium enterprises how to employ a formal TLS certificate management program to address certificate-based risks and challenges. It describes the TLS certificate management challenges faced by organizations; provides recommended best practices for large-scale TLS server certificate management; describes an automated proof-of-concept implementation that demonstrates how to prevent, detect, and recover from certificate-related incidents; and provides a mapping of the demonstrated capabilities to the recommended best practices and to NIST security guidelines and frameworks.

The solutions and architectures presented in this practice guide are built upon standards-based, commercially available, and open-source products. These solutions can be used by any organization managing TLS server certificates. Interoperable solutions are provided that are available from different types of sources (e.g., both commercial and open-source products).

KEYWORDS

Authentication; certificate; cryptography; identity; key; key management; PKI; private key; public key; public key infrastructure; server; signature; TLS; Transport Layer Security

DOCUMENT CONVENTIONS

The terms “shall” and “shall not” indicate requirements to be followed strictly to conform to the publication and from which no deviation is permitted.

The terms “should” and “should not” indicate that among several possibilities, one is recommended as particularly suitable without mentioning or excluding others, or that a certain course of action is preferred but not necessarily required, or that (in the negative form) a certain possibility or course of action is discouraged but not prohibited.

The terms “may” and “need not” indicate a course of action permissible within the limits of the publication.

The terms “can” and “cannot” indicate a possibility and capability, whether material, physical, or causal.

CALL FOR PATENT CLAIMS

This public review includes a call for information on essential patent claims (claims whose use would be required for compliance with the guidance or requirements in this Information Technology Laboratory [ITL] draft publication). Such guidance and/or requirements may be directly stated in this ITL Publication or by reference to another publication. This call also includes disclosure, where known, of the existence of pending U.S. or foreign patent applications relating to this ITL draft publication and of any relevant unexpired U.S. or foreign patents.

ITL may require from the patent holder, or a party authorized to make assurances on its behalf, in written or electronic form, either:

a) assurance in the form of a general disclaimer to the effect that such party does not hold and does not currently intend holding any essential patent claim(s); or

b) assurance that a license to such essential patent claim(s) will be made available to applicants desiring to utilize the license for the purpose of complying with the guidance or requirements in this ITL draft publication either:

i) under reasonable terms and conditions that are demonstrably free of any unfair discrimination; or

ii) without compensation and under reasonable terms and conditions that are demonstrably free of any unfair discrimination.

Such assurance shall indicate that the patent holder (or third party authorized to make assurances on its behalf) will include in any documents transferring ownership of patents subject to the assurance, provisions sufficient to ensure that the commitments in the assurance are binding on the transferee, and that the transferee will similarly include appropriate provisions in the event of future transfers with the goal of binding each successor-in-interest.

The assurance shall also indicate that it is intended to be binding on successors-in-interest regardless of whether such provisions are included in the relevant transfer documents.

Such statements should be addressed to tls-cert-mgmt-nccoe@nist.gov.

ACKNOWLEDGMENTS

We are grateful to the following individuals for their generous contributions of expertise and time.

Name	Organization
Dean Coclin	DigiCert
Tim Hollebeek	DigiCert
Clint Wilson	DigiCert
Dung Lam	F5
Robert Smith	F5
Rob Clatterbuck	SafeNet Assured Technologies (SafeNet AT)

Name	Organization
Jane Gilbert	SafeNet AT
Alexandros Kapasouris	Symantec
Nancy Correll	The MITRE Corporation
Sarah Kinling	The MITRE Corporation
Bob Masucci	The MITRE Corporation
Mary Raguso	The MITRE Corporation
Aaron Aubrecht	Venafi
Justin Hansen	Venafi

The Technology Partners/Collaborators who participated in this build submitted their capabilities in response to a notice in the Federal Register. Respondents with relevant capabilities or product components were invited to sign a Cooperative Research and Development Agreement (CRADA) with NIST, allowing them to participate in a consortium to build this example solution. We worked with:

Technology Partner/Collaborator	Build Involvement
DigiCert	External Certificate Authority and CertCentral console
F5	BIG-IP Local Traffic Manager (load balancer)
SafeNet AT	Luna SA 1700 Hardware Security Module
Symantec	SSL Visibility Appliance for TLS interception and inspection
Venafi	Trust Protection Platform (TLS certificate manager, log server, and scanning tool)

Contents

1 Summary 7

1.1 Challenge 7

1.2 Solution 8

1.3 Benefits 9

2 How to Use This Guide 10

2.1 Typographic Conventions 11

3 Approach 12

3.1 Audience 14

3.2 Scope 14

3.3 Assumptions 14

3.4 Risk Assessment 15

3.4.1 Threats, Vulnerabilities, and Risks 15

3.4.2 Security Categorization and NIST SP 800-53 Controls 17

3.4.3 Security Control Map 17

4 Architecture 23

4.1 Logical Architecture 24

4.1.1 External Systems 24

4.1.2 Internal Systems 24

4.2 Physical Architecture 29

4.3 Technologies 31

4.3.1 Certificate Manager and Internal TLS Certificate Network Scanning Tool 34

4.3.2 Internal TLS Certificate Network Scanning Tool 35

4.3.3 Internal Root CA 37

4.3.4 Internal Issuing CA 37

4.3.5 Certificate Database 37

4.3.6 TLS Inspection Appliance 38

4.3.7 Hardware Security Module 38

4.3.8 External Certificate Authority 39

4.3.9 Load Balancer....40
4.3.10 DevOps Framework....41
4.3.11 Automated Certificate Management Frameworks....42
4.3.12 TLS Servers....42
4.3.13 Application Servers....44
5 Security Characteristic Analysis....46
5.1 Assumptions and Limitations....46
5.2 Functional Capabilities Demonstration....46
5.2.1 Definitions....46
5.2.2 Functional Capabilities....46
5.2.3 Mapping to NIST SP 1800-16B Recommendations....50
5.3 Scenarios and Findings....53
5.3.1 Demonstration Scenario....53
5.3.2 Findings....55
6 Future Build Considerations....56
Appendix A List of Acronyms....57
Appendix B Glossary....59
Appendix C References....68

List of Figures

Figure 4-1 Logical Architecture Components and Roles.... 24
Figure 4-2 TLS Server Certificate Management Example Solution Logical Architecture.... 28
Figure 4-3 Laboratory Configuration of TLS Server Certificate Management Example Implementation.... 29
Figure 4-4 Venafi Scanafi Performing Network Scans and Providing Scan Results to Venafi TPP.... 37
Figure 4-5 Example Implementation’s DevOps Components Requesting and Receiving Certificates.... 41
Figure 4-6 Certbot Fetching and Deploying TLS Certificates via the ACME Protocol.... 42

List of Tables

Table 2-1 Typographic Conventions 12

Table 3-1 Mapping Security Characteristics of the Example Implementation to the Cybersecurity Framework and Informative Security Control References 18

Table 4-1 Products and Technologies 31

Table 5-1 Mapping Between Volume B Policy Recommendations and the Example Implementation . 50

1 Summary

The National Cybersecurity Center of Excellence (NCCoE) at the National Institute of Standards and Technology (NIST) recognizes the need to ensure secure communications between clients and servers. To enhance secure communications, the NCCoE launched a project titled Transport Layer Security (TLS) Server Certificate Management. This project uses commercially available technologies to develop a cybersecurity reference design that can be implemented in enterprise environments to reduce outages, improve security, and enable disaster recovery activities related to TLS certificates.

TLS is a broadly used cryptographic protocol that enables authentication and encryption of communications between clients and servers. TLS requires the use of both a certificate that contains information about the certificate owner, as well as a corresponding private key. A server using TLS must have a certificate (and the corresponding private key) to authenticate itself and to establish symmetric keys for encryption. The ongoing maintenance of TLS certificates is labor-intensive and can produce erroneous conditions if the certificate maintenance is not performed correctly.

This project focuses on management of TLS server certificates in medium and large enterprises that rely on TLS to secure both customer-facing and internal applications. Client certificates may optionally be used in TLS for mutual authentication with a TLS server, but management of client certificates is outside the scope of this project. This project demonstrates how to establish, assign, change, and track an inventory of TLS certificates in a manner designed to reduce outages, improve security, and enable disaster recovery activities. This publicly available NIST Cybersecurity Practice Guide details a set of practical steps for implementing a cybersecurity reference design that addresses this TSL server certificate management challenge.

1.1 Challenge

TLS server certificates and private keys are generally installed and managed by the server's system administrator—others usually do not have the access rights required on the system to manage them. To get a certificate, an administrator executes commands on the system to generate a cryptographic key pair (the public key and the private key), and then requests a certificate from a certificate authority (CA). Because many system administrators are not knowledgeable about certificates and cryptography, this process can be confusing and error prone. Large organizations often have a central group, typically called the public key infrastructure (PKI) team, that manages the CAs, which can include external public CAs and internally operated CAs. Due to its expertise in certificates, the PKI team typically supports the system administrators through the key pair generation and certificate request process. Medium and large organizations have many system administrators but only a handful of people on the PKI team. This distributed management environment for certificates and private keys fosters a variety of risks and challenges:

- **Application Outages:** Nearly every enterprise has experienced application outages due to expired TLS server certificates, causing major disruptions to online banking, reservations systems, and healthcare services, to name a few. The drive to encrypt all communications (internal and external) is expanding the reliance on TLS server certificates, increasing the potential for critical system outages.
- **Security Risks:** TLS server certificates function as trusted machine identities. If an attacker can get a fraudulent certificate or compromise a private key, they can impersonate the server or eavesdrop on communications.
- **Disaster Recovery Risks:** Several certificate-related incidents can require an organization to rapidly change large numbers of TLS server certificates, including a CA compromise, algorithm deprecation, or cryptographic library bug. If an organization is not prepared for rapid replacement, its services could be unavailable for days or weeks.

1.2 Solution

The TLS Server Certificate Management Project addressed the risks and challenges described above by:

- Defining an initial reference design that represents a typical enterprise network and recommended TLS infrastructure.
- Building that reference design by using currently available components. This build is known as an “example solution.” In the course of building the example solution, the reference design was enhanced. The example solution is an instantiation of the final reference design.
- Demonstrating how the example solution addresses these risks.

The approach taken to address these issues with life-cycle management of the certificates includes the following phases:

- **Establish Governance:** The project team defined a set of certificate management policies based on the guidance provided in existing NIST documents to establish consistent governance of TLS certificates.
- **Create and Maintain an Inventory:** A PKI team worked with project staff representing lines of business and system administrators to establish a complete inventory of all TLS server certificates through automated discovery. The team leveraged configurable rules to automatically organize discovered certificates and associate them with owners as required to enable automated notifications.

- **Register for and Install Certificates:** Certificates were requested and installed to address cases where new certificates were needed, or existing certificates were nearing expiration and required renewal and replacement. Because enterprise environments are diverse, with different technical and organizational constraints, possible methods for requesting and installing certificates were demonstrated, including:
    - **Manual:** Security, operational, or technical requirements/constraints mandate that the server's system administrator manually requests a certificate by using command line tools and a certificate management system portal.
    - **Standardized Automated Certificate Installation:** A TLS server is configured to automatically request and install a certificate by using a protocol, such as the Automatic Certificate Management Environment (ACME) protocol, developed by the Internet Engineering Task Force (IETF).
    - **Installation Using a Proprietary Method:** The certificate management system uses a method that is proprietary to the TLS server to install certificates on one or more systems that do not support a standard automated method for requesting and installing certificates.
    - **Development Operations (DevOps)-Based Installation:** A DevOps framework used to install and configure servers/applications also requests and installs certificates. This was done in a cloud environment where DevOps frameworks are commonly used.
    - The majority of private keys used with certificates are stored in files; however, Hardware Security Modules (HSMs) were demonstrated to increase the security of private keys. Where practical, the methods listed above were performed on a system that uses an HSM for private-key protection.
- **Continuously Monitor and Manage:** The inventory of certificates was monitored for expiration, proper operation, and security issues. Notifications and alerts were triggered when anomalies were detected. Management operations were regularly performed to ensure proper operation and security.
- **Detect, Respond, and Recover from Incidents:** Scenarios were demonstrated in which, due to situations such as CA compromise or a broken algorithm (e.g., cryptographic library bug that created weak keys for certificates), a large number of certificates required rapid replacement. The certificate management system orchestrated replacement of all certificates.

1.3 Benefits

The project demonstration and its associated documentation offer the following benefits to organizations that have operational or security requirements to implement TLS:

- **Reduced Overhead and Risks—**Large- and medium-size organizations can reduce labor-intensive overhead and risks associated with TLS certificate maintenance by using an example solution comprising currently available components.

- **Improved Information Technology (IT) Environments—**Descriptions of demonstrated methods for using the example solution can reduce the occurrences of erroneous conditions resulting from improper performance of certificate maintenance.
- **Enhanced Cybersecurity—**The availability of source material that explains how the example solution can satisfy specified security requirements can enhance the maturity of cybersecurity programs throughout systems' life cycles.

2 How to Use This Guide

This NIST Cybersecurity Practice Guide demonstrates a standards-based reference design and provides users with the information they need to replicate security platforms composed of currently available components that can be used by large and medium-size organizations to reduce the labor-intensive overhead associated with maintenance of TLS certificates. This reference design is modular and can be deployed in whole or in part.

This guide contains four volumes:

- NIST SP 1800-16A: *Executive Summary*
- NIST SP 1800-16B: Security Risks and Recommended Best Practices
- NIST SP 1800-16C: *Approach, Architecture, and Security Characteristics*–what we built and why **(you are here)**
- NIST SP 1800-16D: *How-To Guides*–instructions for building the example solution
- Depending on your role in your organization, you might use this guide in different ways:
- **Business decision makers, including chief security and technology officers,** will be interested in the *Executive Summary,* NIST SP 1800-16A, which describes the following topics:
- challenges that enterprises face in managing TLS server certificates
- example solution built at the NCCoE
- benefits of adopting the example solution

Senior information technology and security officers will be informed by NIST SP 1800-16B, *Security Risks and Recommended Best Practices*, which describes the:

- TLS server certificate infrastructure and management processes
- risks associated with mismanagement of certificates
- organizational challenges associated with certificate management
- recommended best practices for server certificate management
- recommendations for implementing a successful certificate management program

- You might share the *Executive Summary,* NIST SP 1800-16A*,* with your leadership team members to help them understand the importance and benefits of adopting standards-based TLS server certificate management.
- **Technology or security program managers** who are concerned with how to identify, understand, assess, and mitigate risk will be interested in the following sections of the guide, NIST SP 1800-16C*,* which describe what we did and why:
- Section 3.4.1, Threats, Vulnerabilities and Risks
- Section 3.4.3, Security Control Map, maps the security characteristics of this example solution to cybersecurity standards and best practices
- You might share *Security Risks and Recommended Best Practices,* NIST SP 1800-16B*,* with your leadership team members to help them understand the security context for adopting the standards-based TLS server certificate management approach described in this volume.
- **IT professionals** who want to implement an approach like this will find the whole practice guide useful. You can use the how-to portion of the guide, NIST SP 1800-16D, to replicate all or parts of the build created in our lab. The how-to guide provides specific product installation, configuration, and integration instructions for implementing the example solution. We do not recreate the product manufacturers' documentation, which is generally widely available. Rather, we show how we incorporated the products together in our environment to create an example solution.
- This guide assumes that IT professionals have experience implementing security products within the enterprise. While we have used a suite of commercial products to address this challenge, this guide does not endorse these particular products. Your organization can adopt this solution or one that adheres to these guidelines in whole, or you can use this guide as a starting point for tailoring and implementing parts of enhanced TLS server certificate management. Your organization's security experts should identify the products that will best integrate with your existing tools and IT system infrastructure. We hope that you will seek products that are congruent with applicable standards and best practices. Section 4.3, Technologies, lists the products we used and maps them to the cybersecurity controls provided by this reference solution.

A NIST Cybersecurity Practice Guide does not describe "the" solution, but a possible solution. This is a draft guide. We seek feedback on its contents and welcome your input. Comments, suggestions, and success stories will improve subsequent versions of this guide. Please contribute your thoughts to tls-cert-mgmt-nccoe@nist.gov.

2.1 Typographic Conventions

The following table presents typographic conventions used in this volume.

Table 2-1 Typographic Conventions

Typeface/Symbol	Meaning	Example
Italics	file names and path names; references to documents that are not hyperlinks; new terms; and placeholders	For detailed definitions of terms, see the *NCCoE Glossary*.
Bold	names of menus, options, command buttons, and fields	Choose **File** > **Edit.**
`Monospace`	command-line input, on-screen computer output, sample code examples, and status codes	`Mkdir`
`Monospace Bold`	command-line user input contrasted with computer output	**`service sshd start`**
blue text	link to other parts of the document, a web URL, or an email address	All publications from NIST's NCCoE are available at https://www.nccoe.nist.gov.

3 Approach

The approach taken to building and demonstrating the TLS server certificate management example solution involved composing demonstration environments that included test, diagnostic, and support elements used in the lab for demonstration and test purposes. The demonstration environment includes 1) components typically residing outside the organizational firewall (e.g., public certificate authorities) and 2) systems typically deployed within organizational network environments (e.g., TLS servers, load balancers, DevOps frameworks, internal certificate authorities, certificate managers, and certificate network scanning tools). The goal of the example solution is to permit stakeholders, such as those in the list that follows, to more effectively manage and maintain TLS server certificates throughout system life cycles:

- people in leadership positions who are responsible for cybersecurity
- people in leadership positions who are responsible for the line of business or application and who will drive the need for certificates to be deployed
- system administrators responsible for managing TLS servers and ensuring the load balancer will be represented
- DevOps developers responsible for programming/configuring and managing the DevOps framework

- individuals responsible for reviewing and approving/rejecting certificate management operations
- individuals responsible for managing certificate management systems and public/internal CAs

The NCCoE team accomplished the project in the following sequence:

- established a set of recommended certificate management policy requirements based on the guidance provided in existing NIST documents to establish consistent governance of TLS certificates
- solicited industry collaborators to provide components, operational experience, and configuration assistance; integrated the components into a demonstration environment; configured the components to provide services
- worked with industry collaborators to refine a notional reference design into a demonstration environment capable of:
    - leveraging configurable rules to establish a complete inventory of all TLS server certificates through automated discovery, and automatically organizing discovered certificates and associate owners to enable automated notifications
    - registering for and installing certificates by using manual and automated methods, including protocols such as ACME, proprietary installation methods, and a DevOps framework
- worked with industry collaborators to integrate HSMs into the demonstration environment for protecting private keys
- documented collaborator contributions
- documented the final architecture of the demonstration environment
- worked with industry collaborators to demonstrate continuous monitoring of the inventory of certificates for expiration, proper operation, and security issues and generation of notifications and alerts when anomalies are detected
- worked with industry collaborators to demonstrate detection, response, and recovery from security incidents
- conducted security and functional testing of the demonstration environment
- conducted and documented the results of a risk assessment and a security characteristics analysis, including mapping the security contributions' demonstrated capabilities to the *Framework for Improving Critical Infrastructure Cybersecurity* (Cybersecurity Framework), NIST Special Publication (SP) 800-53, and the recommended policies in NIST SP 1800-16B
- documented the steps taken to install and configure each component of the demonstration environment
- worked with industry collaborators to suggest future considerations for TLS certificate management in general

3.1 Audience

This guide is intended for individuals responsible for security architecture and strategy, system administration, PKI support, IT systems acquisition, cybersecurity assessments, IT system component development, marketing and support for environments for which TLS is an essential security protocol for providing confidentiality and integrity protection to systems and operations, and implementing security solutions in organizations' IT support activities. The technical components will appeal to system administrators, IT managers, IT security managers, and others directly involved in the secure and safe operation of IT networks.

3.2 Scope

As stated in the Summary above, this project focuses on management of TLS server certificates in medium and large enterprises that rely on TLS to secure both customer-facing and internal applications. This guide shows how to establish and maintain an inventory of TLS certificates; assign and track certificate owners (i.e., custodians), identify issues with and vulnerabilities of the TLS infrastructure, automate enrollment and installation, report, and continuously monitor TLS certificates in the environment described above.

This project limits its scope to TLS server certificates. Client certificates may optionally be used in TLS for mutual authentication, but management of client certificates is outside the scope of this project.

The security and integrity of TLS relies on secure implementation and configuration of TLS servers and effective TLS server certificate management. Guidance regarding the implementation and configuration of TLS servers is outside of the scope of this document. Secure implementation and configuration of TLS servers is addressed in NIST SP 800-52. Organizations should provide clear instruction to groups and individuals deploying TLS servers in their environments, to read, understand, and follow the guidance provided in NIST SP 800-52.

3.3 Assumptions

This project is guided by the following assumptions:

- The processes for obtaining and maintaining TLS server certificates in medium and large IT enterprises is labor-intensive and error prone.
- The drive to encrypt all communications (internal and external) is expanding reliance on TLS server certificates, thereby increasing the potential for critical system outages due to expired certificates.
- TLS server certificates serve as trusted machine identities; if an attacker can get a fraudulent certificate or compromise a private key, they can impersonate the server or eavesdrop on communications.
- Certificate-related incidents (e.g., a CA compromise, algorithm deprecation, or cryptographic library bug) can require an organization to rapidly change large numbers of TLS server certificates.

- If an organization is not prepared for rapid replacement, then its services could be unavailable for days or weeks.

3.4 Risk Assessment

NIST SP 800-30 Revision 1, *Guide for Conducting Risk Assessments* states that risk is "a measure of the extent to which an entity is threatened by a potential circumstance or event, and typically a function of (i) the adverse impacts that would arise if the circumstance or event occurs and (ii) the likelihood of occurrence." The guide further defines risk assessment as "the process of identifying, estimating, and prioritizing risks to organizational operations (including mission, functions, image, reputation), organizational assets, individuals, other organizations, and the Nation, resulting from the operation of an information system. Part of risk management incorporates threat and vulnerability analyses, and considers mitigations provided by security controls planned or in place."

The NCCoE recommends that any discussion of risk management, particularly at the enterprise level, begins with a comprehensive review of NIST SP 800-37 Revision 2, *Risk Management Framework for Information Systems and Organizations: A System Life Cycle Approach for Security and Privacy*—material that is available to the public. The risk management framework (RMF) guidance, as a whole, was invaluable and gave us a baseline to assess risks, from which we developed the project, the security characteristics of the build, and this guide.

3.4.1 Threats, Vulnerabilities, and Risks

NIST SP 1800-16B, *Security Risks and Recommended Best Practices*, describes the risks associated with management of TLS server certificates. It points out that, despite the mission-critical nature of TLS server certificates, many organizations do not have clear policies, processes, roles, and responsibilities defined to ensure effective certificate management. Moreover, many organizations do not leverage available technology and automation to effectively manage the large and growing number of TLS server certificates. As a result, many organizations continue to experience significant incidents related to TLS server certificates. Malicious entities are using encryption to attack organizations at an ever-increasing rate. TLS is being turned against enterprises to:

- deliver malware undetected
- listen in on private conversations
- disrupt secured transactions
- exfiltrate data over encrypted communication channels

Volume B states that certificate owners are typically not knowledgeable about the best practices for effectively managing TLS server certificates. The RMF process described in NIST SP 800-37, together with the Cybersecurity Framework and NIST SP 800-53, informed our risk assessment and subsequent recommendations from which we developed the security characteristics of the build and this guide.

The most serious risks associated with certificate management stem from certificate owners, responsible for the systems where certificates are deployed, not being provided clear certificate management requirements, not understanding their responsibilities in fulfilling those requirements, and those requirements not being enforced as policies. Risks identified in Volume B include:

- outages caused by expired certificates due to:
    - the system administrator forgetting about the certificate
    - the system administrator ignoring notifications that the certificate will soon expire
    - the system administrator not properly installing or updating the CA certificate chain
    - the system administrator being reassigned and nobody else receiving expiry notifications
    - the system administrator enrolling for a new certificate but not installing it on the server(s) in time, installing it incorrectly, or not resetting the application/server, so the newly installed certificate is loaded and used
    - the application relying on multiple load-balanced servers and the certificate not being updated on all of them
- server impersonation (an attacker being able to impersonate a legitimate TLS server)
- the organization not being able to replace certificates and private keys in a timely manner due to inadequate records, knowledge, and processes in instances such as:
    - CA compromise
    - cryptographic algorithm vulnerability
    - cryptographic library bugs
- encrypted threats such as TLS encryption allowing attackers to hide malicious activities within encrypted TLS connections

Also, as pointed out in Volume B, an attacker may be able to masquerade as a server to all clients if:

- the server's private key
    - is weak
    - can be obtained by an attacker
- an attacker can obtain a public key certificate for a public key corresponding to its own private key in the name of the server from a CA trusted by the clients

Aside from the risks of not managing TLS server certificates properly, additional risks often plague TLS implementations themselves. Proper protocol specification does not guarantee the security of implementations. In particular, when integrating into higher level protocols, TLS and its PKI-based authentication are sometimes the source of misunderstandings and implementation shortcuts. An extensive survey of these issues can be found in *Proceedings of the 2012 ACM Conference on Computer and Communications Security*.

3.4.2 Security Categorization and NIST SP 800-53 Controls

Under the RMF, the first step in managing risk is determining the impacts of exploitation of system confidentiality, integrity, and availability vulnerabilities. NIST SP 800-53-controls needed to mitigate system vulnerabilities are keyed to the Federal Information Processing Standards (FIPS) 199 impact levels. Based on the risks identified, and assuming a *Standards for Security Categorization of Federal Information and Information Systems*, FIPS 199 ***moderate*** impact level (exploitation of vulnerabilities would result in serious harm to the system and its mission), a number of NIST SP 800-53 controls are assigned to address TLS server certificate risks: AC-1, AC-5, AC-6, AC-16, AT-2, AU-1, AU-2, AU-3, AU-6, AU-12, AU-13, AU-14, CA-1, CA-2, CA-5, CA-7, CM-2, CM-3, CM-5, CM-6, CM-8, CM-9, CM-12, CP-2, CP-3, CP-4, CP-7, CP-13, IA-3, IA-4, IA-5, IA-9, IR-1, IR-2, IR-3, IR-4, MA-1, MA-6, PL-2, PL-9, PL-10, PM-1, PM-2, PM-4, PM-5, PM-7, PM-9, RA-3, RA-5, RA-7, SA-1, SA-3, SA-4, SA-10, SC-1, SC-6, SC-8, SC-12, SC-17, SC-23, and SI-4. Appendix C of Volume B describes these security controls and their relevance to the best practices identified in Volume B.

3.4.3 Security Control Map

The objective of this project is to demonstrate how the processes for obtaining and maintaining TLS server certificates in medium and large IT enterprises can be made less labor-intensive and error prone, to reduce security and operational risks. This requires adherence to the following principles:

- **Governance and Risk Management:** The project includes clear recommended policies that can be used to educate the lines of business and system administrators to ensure they understand the security risks and their responsibilities in addressing those risks. Organizations are free to copy and use these recommended policies for definition of their own internal TLS certificate management policies.
- **Visibility and Awareness:** Most organizations do not have an inventory of their TLS server certificates and private keys, their installed locations, and their responsible individuals/groups. This project demonstrates how to achieve visibility and awareness of all certificates.
- **Reliable and Efficient Certificate Provisioning:** This project demonstrates effective processes to ensure availability of valid certificates and keys for TLS servers while minimizing overhead and the impact on operations.
- **Certificate Disaster Recovery:** This project demonstrates effective processes for organizations to be prepared for and to respond to large-scale incidents (e.g., CA compromise) that require rapid replacement of large numbers of certificates and keys.
- **Audit Logging:** Many organizations do not generate, store, and review audit logs for their certificates and associated private keys. This project demonstrates how to establish and maintain complete audit trails of certificate and private-key life cycles.
- **Secure Certificate Management Platform:** The certificate management platform in this project is deployed on a hardened system and provides the security attributes required to protect the assets it manages.

- **Private-Key Security:** The project demonstrates automated management, which reduces the requirement for direct administrator access to private keys, and HSM-based private-key protection, which significantly increases private-key security.

Appendix B of Volume B maps the recommended best practices for TLS server certificate management described in volume B to the Cybersecurity Framework Subcategories. The following table lists the security Subcategories of the Cybersecurity Framework that are supported by the example TLS server certificate management example solution described in this volume, and it maps these Cybersecurity Framework Subcategories to other informative security control references.

Table 3-1 Mapping Security Characteristics of the Example Implementation to the Cybersecurity Framework and Informative Security Control References

Cybersecurity Framework Function	Cybersecurity Framework Subcategory	Informative References
Identify	ID.AM-2: Software platforms and applications within the organization are inventoried.	• CCS CSC 2 • COBIT 5 BAI09.01, BAI09.02, BAI09.05 • ISA 62443-2-1:2009 4.2.3.4 • ISA 62443-3-3:2013 SR 7.8 • ISO/IEC 27001:2013 A.8.1.1, A.8.1.2 • NIST SP 800-53 Rev. 4 CM-8
	ID.AM-6: Cybersecurity roles and responsibilities for the entire workforce and third-party stakeholders (e.g., suppliers, customers, partners) are established.	• COBIT 5 APO01.02, DSS06.03 • ISA 62443-2-1:2009 4.3.2.3.3 • ISO/IEC 27001:2013 A.6.1.1 • NIST SP 800-53 Rev. 4 CP-2, PS-7, PM-11
	ID.GV-1: Organizational cybersecurity policy is established and communicated.	• CIS CSC 19 • COBIT 5 APO01.03, APO13.01, EDM01.01, EDM01.02 • ISA 62443-2-1:2009 4.3.2.6 • ISO/IEC 27001:2013 A.5.1.1 • NIST SP 800-53 Rev. 4 -1 controls from all security control families
	ID.GV-2: Cybersecurity roles and responsibilities are coordinated and aligned with internal roles and external partners.	• CIS CSC 19 • COBIT 5 APO01.02, APO10.03, APO13.02, DSS05.04 • ISA 62443-2-1:2009 4.3.2.3.3 • ISO/IEC 27001:2013 A.6.1.1, A.7.2.1, A.15.1.1

Cybersecurity Framework Function	Cybersecurity Framework Subcategory	Informative References
		• NIST SP 800-53 Rev. 4 PS-7, PM-1, PM-2
	ID.GV-3: Legal and regulatory requirements regarding cybersecurity, including privacy and civil liberties obligations, are understood and managed.	• CIS CSC 19 • COBIT 5 BAI02.01, MEA03.01, MEA03.04 • ISA 62443-2-1:2009 4.4.3.7 • ISO/IEC 27001:2013 A.18.1.1, A.18.1.2, A.18.1.3, A.18.1.4, A.18.1.5 • NIST SP 800-53 Rev. 4 -1 controls from all security control families
	ID.GV-4: Governance and risk management processes address cybersecurity risks.	• COBIT 5 EDM03.02, APO12.02, APO12.05, DSS04.02 • ISA 62443-2-1:2009 4.2.3.1, 4.2.3.3, 4.2.3.8, 4.2.3.9, 4.2.3.11, 4.3.2.4.3, 4.3.2.6.3 • ISO/IEC 27001:2013 Clause 6 • NIST SP 800-53 Rev. 4 SA-2, PM-3, PM-7, PM-9, PM-10, PM-11
Protect	PR.AC-1: Identities and credentials are issued, managed, verified, revoked, and audited for authorized devices, users, and processes.	• CCS CSC 16 • COBIT 5 DSS05.04, DSS06.03 • ISA 62443-2-1:2009 4.3.3.5.1 • ISA 62443-3-3:2013 SR 1.1, SR 1.2, SR 1.3, SR 1.4, SR 1.5, SR 1.7, SR 1.8, SR 1.9 • ISO/IEC 27001:2013 A.9.2.1, A.9.2.2, A.9.2.3, A.9.2.4, A.9.2.6, A.9.3.1, A.9.4.2, A.9.4.3 • NIST SP 800-53 Rev. 4 AC-2, IA-1, IA-2, IA-3, IA-4, IA-5, IA-6, IA-7, IA-8, IA-9, IA-10, IA-11
	PR.AC-3: Remote access is managed.	• COBIT 5 APO13.01, DSS01.04, DSS05.03 • ISA 62443-2-1:2009 4.3.3.6.6 • ISA 62443-3-3:2013 SR 1.13, SR 2.6 • ISO/IEC 27001:2013 A.6.2.2, A.13.1.1, A.13.2.1 • NIST SP 800-53 Rev. 4 AC-17, AC-19, AC-20

Cybersecurity Framework Function	Cybersecurity Framework Subcategory	Informative References
	PR.AC-4: Access permissions and authorizations are managed, incorporating the principles of least privilege and separation of duties.	• CIS CSC 3, 5, 12, 14, 15, 16, 18 • COBIT 5 DSS05.04 • ISA 62443-2-1:2009 4.3.3.7.3 • ISA 62443-3-3:2013 SR 2.1 • ISO/IEC 27001:2013 A.6.1.2, A.9.1.2, A.9.2.3, A.9.4.1, A.9.4.4, A.9.4.5 • NIST SP 800-53 Rev. 4 AC-1, AC-2, AC-3, AC-5, AC-6, AC-14, AC-16, AC-24
	PR.AC-6: Identities are proofed and bound to credentials and asserted in interactions.	• CCS CSC 16 • COBIT 5 DSS05.04, DSS05.05, DSS05.07, DSS06.03 • ISA 62443-2-1:2009 4.3.3.2.2, 4.3.3.5.2, 4.3.3.7.2, 4.3.3.7.4 • ISA 62443-3-3:2013 SR 1.1, SR 1.2, SR 1.4, SR 1.5, SR 1.9, SR 2.1 • ISO/IEC 27001:2013 A.7.1.1, A.9.2.1 • NIST SP 800-53 Rev. 4 AC-1, AC-2, AC-3, AC-16, AC-19, AC-24, IA-1, IA-2, IA-4, IA-5, IA-8, PE-2, PS-3
	PR.AC-7: Users, devices, and other assets are authenticated (e.g., single-factor, multi-factor) commensurate with the risk of the transaction (e.g., individuals' security and privacy risks and other organizational risks).	• CCS CSC 1, 12, 15, 16 • COBIT 5 DSS05.04, DSS05.10, DSS06.10 • ISA 62443-2-1:2009 4.3.3.6.1, 4.3.3.6.2, 4.3.3.6.3, 4.3.3.6.4, 4.3.3.6.5, 4.3.3.6.6, 4.3.3.6.7, 4.3.3.6.8, 4.3.3.6.9 • ISA 62443-3-3:2013 SR 1.1, SR 1.2, SR 1.5, SR 1.7, SR 1.8, SR 1.9, SR 1.10 • ISO/IEC 27001:2013 A.9.2.1, A.9.2.4, A.9.3.1, A.9.4.2, A.9.4.3, A.18.1.4 • NIST SP 800-53 Rev. 4 AC-7, AC-8, AC-9, AC-11, AC-12, AC-14, IA-1, IA-2, IA-3, IA-4, IA-5, IA-8, IA-9, IA-10, IA-11
	PR.DS-1: Data at rest is protected.	• CCS CSC 17 • COBIT 5 APO01.06, BAI02.01, BAI06.01, DSS06.06 • ISA 62443-3-3:2013 SR 3.4, SR 4.1

Cybersecurity Framework Function	Cybersecurity Framework Subcategory	Informative References
		• ISO/IEC 27001:2013 A.8.2.3 • NIST SP 800-53 Rev. 4 SC-28
	PR.DS-2: Data in transit is protected.	• CCS CSC 17 • COBIT 5 APO01.06, DSS06.06 • ISA 62443-3-3:2013 SR 3.1, SR 3.8, SR 4.1, SR 4.2 • ISO/IEC 27001:2013 A.8.2.3, A.13.1.1, A.13.2.1, A.13.2.3, A.14.1.2, A.14.1.3 • NIST SP 800-53 Rev. 4 SC-8
	PR.DS-3: Assets are formally managed throughout removal, transfers, and disposition.	• COBIT 5 BAI09.03 • ISA 62443-2-1:2009 4. 4.3.3.3.9, 4.3.4.4.1 • ISA 62443-3-3:2013 SR 4.2 • ISO/IEC 27001:2013 A.8.2.3, A.8.3.1, A.8.3.2, A.8.3.3, A.11.2.7 • NIST SP 800-53 Rev. 4 CM-8, MP-6, PE-16
	PR.DS-6: Integrity-checking mechanisms are used to verify software, firmware, and information integrity.	• ISA 62443-3-3:2013 SR 3.1, SR 3.3, SR 3.4, SR 3.8 • ISO/IEC 27001:2013 A.12.2.1, A.12.5.1, A.14.1.2, A.14.1.3 • NIST SP 800-53 Rev. 4 SC-16, SI-7
	PR.DS-8: Integrity-checking mechanisms are used to verify hardware integrity.	• COBIT 5 BAI03.05 • ISA 62443-2-1:2009 4.3.4.4.4 • ISO/IEC 27001:2013 A.11.2.4 • NIST SP 800-53 Rev. 4 SA-10, SI-7
	PR.IP-2: A system development life cycle to manage systems is implemented.	• COBIT 5 APO13.01 • ISA 62443-2-1:2009 4.3.4.3.3 • ISO/IEC 27001:2013 A.6.1.5, A.14.1.1, A.14.2.1, A.14.2.5 NIST SP 800-53 Rev. 4 SA-3, SA-4, SA-8, SA10, SA-11, SA-12, SA-15, SA-17, PL-8
	PR.IP-3: Configuration change control processes are in place.	• COBIT 5 BAI01.06, BAI06.01 • ISA 62443-2-1:2009 4.3.4.3.2, 4.3.4.3.3

Cybersecurity Framework Function	Cybersecurity Framework Subcategory	Informative References
		• ISA 62443-3-3:2013 SR 7.6 • ISO/IEC 27001:2013 A.12.1.2, A.12.5.1, A.12.6.2, A.14.2.2, A.14.2.3, A.14.2.4 • NIST SP 800-53 Rev. 4 CM-3, CM-4, SA-10
	PR.PT-1: Audit/log records are determined, documented, implemented, and reviewed in accordance with policy.	• CCS CSC 14 • COBIT 5 APO11.04 • ISA 62443-2-1:2009 4.3.3.3.9, 4.3.3.5.8, 4.3.4.4.7, 4.4.2.1, 4.4.2.2, 4.4.2.4 • ISA 62443-3-3:2013 SR 2.8, SR 2.9, SR 2.10, SR 2.11, SR 2.12 • ISO/IEC 27001:2013 A.12.4.1, A.12.4.2, A.12.4.3, A.12.4.4, A.12.7.1 • NIST SP 800-53 Rev. 4 AU Family
	PR.PT-5: Mechanisms (e.g., fail-safe, load balancing, hot swap) are implemented to achieve resilience requirements in normal and adverse situations.	• COBIT 5 BAI04.01, BAI04.02, BAI04.03, BAI04.04, BAI04.05, DSS01.05 • ISA 62443-2-1:2009 4.3.2.5.2 • ISA 62443-3-3:2013 7.1, SR 7.2 • ISO/IEC 27001:2013 A.17.1.2, A.17.2.1 • NIST SP 800-53 Rev. 4 CP-7, CP-8, CP-11, CP-13, PL-8, SA-14, SC-6
	DE.AE-5: Incident alert thresholds are established.	• COBIT 5 APO12.06 • ISA 62443-2-1:2009 4.2.3.10 • NIST SP 800-53 Rev. 4 IR-4, IR-5, IR-8
	DE.CM-1: The network is monitored to detect potential cybersecurity events.	• COBIT 5 APO12.06 • ISA 62443-2-1:2009 4.3.4.5.9 • ISA 62443-3-3:2013 SR 6.1 • ISO/IEC 27001:2013 A.16.1.2 • NIST SP 800-53 Rev. 4 AU-6, CA-2, CA-7, RA-5, SI-4
Respond	RS.AN-5: Processes are established to receive, analyze, and respond to vulnerabilities disclosed to the organization from internal and external sources (e.g.,	• CIS CSC 4, 19 • COBIT 5 EDM03.02, DSS05.07 • NIST SP 800-53 Rev. 4 SI-5, PM-15

Cybersecurity Framework Function	Cybersecurity Framework Subcategory	Informative References
	internal testing, security bulletins, or security researchers).	
	RS.MI-2: Incidents are mitigated.	• ISA 62443-2-1:2009 4.3.4.5.6, 4.3.4.5.10 • ISO/IEC 27001:2013 A.12.2.1, A.16.1.5 • NIST SP 800-53 Rev. 4 IR-4
	RS.MI-3: Newly identified vulnerabilities are mitigated or documented as accepted risks.	• ISO/IEC 27001:2013 A.12.6.1 • NIST SP 800-53 Rev. 4 CA-7, RA-3, RA-5

4 Architecture

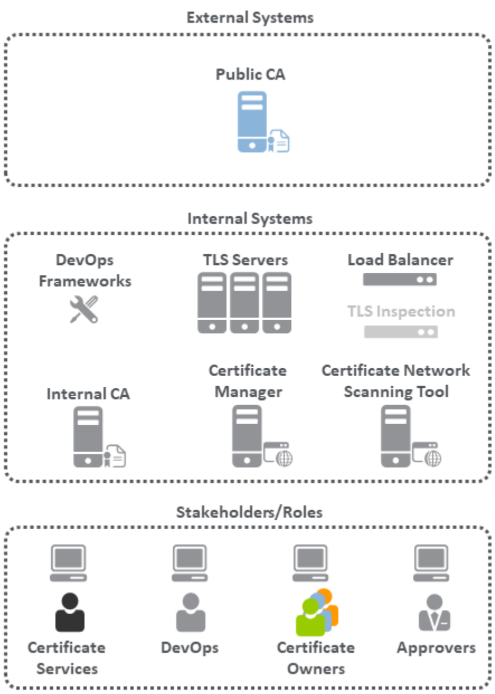
The TLS server certificate management architecture enables medium and large enterprises to manage their TLS server certificates and cryptographic keys efficiently and effectively. The architecture provides the following protections:

- use of a certificate manager and related certificate scanning, monitoring, and storage components to:
    - automate establishment and maintenance of an inventory of TLS server certificates and keys
    - assign and track certificate owners
    - automate enrollment, installation, renewal, and rapid replacement of certificates and keys
    - continuously monitor certificates and keys, report on their status, and automate remediation to enforce compliance with policy and avoid unintended expiration
    - support disaster recovery through rapid, large-scale replacement of certificates
    - log all certificate management operations
- use of a TLS inspection appliance to decrypt network traffic encrypted via TLS, so it can be inspected for malware and other threats
- use of a hardened, tamper-resistant physical appliance that securely generates, stores, manages, and processes cryptographic key pairs for use with TLS certificates; this enables those keys to remain securely within the confines of the secure device while they are used to issue signed TLS certificates

4.1 Logical Architecture

The functions demonstrated in this project require a variety of component systems and configurations. Figure 4-1 depicts the architectural components used in the logical architecture and the roles that support TLS server certificate management.

Figure 4-1 Logical Architecture Components and Roles



4.1.1 External Systems

The architecture includes a CA component that typically resides outside the organizational firewall:

- **Public CA:** A publicly trusted CA issued one or more of the certificates used on the TLS servers in the implementation.

4.1.2 Internal Systems

The architecture includes the following systems that are typically deployed within organizational network environments.

- **TLS Servers:** Multiple systems were configured as TLS servers (e.g., web server, application server, or other service). Certificates are deployed and managed on these systems.
- **Load Balancer:** A load balancer acted as a TLS server with a certificate and facilitated the load balancing of traffic to other TLS servers.

- **DevOps Framework(s):** A DevOps framework (Kubernetes) automated management of containers acting as TLS servers and deployment of certificates on those TLS servers.
- **Internal CA:** An internal CA issued certificates to some TLS servers.
- **Certificate Manager:** A certificate management system was used to inventory and manage TLS server certificates deployed in the environment.
- **Certificate Network Scanning Tool:** A vulnerability scanning tool facilitated discovery of TLS server certificates via network scanning.
- **TLS Inspection Appliance:** This appliance decrypts traffic encrypted via TLS. As a result, traffic is analyzed and inspected for malicious activity, viruses, malware, or other threats. (Figure 4-1 depicts this component by using a faded icon to convey that some organizations, as a matter of policy, may not want to include it in their network architecture.)
- Humans play an important part in the management of TLS server certificates in enterprises. Descriptions of their different roles are explained below:
- **Certificate Owners:** The groups and individuals responsible for the systems where certificates are deployed; they establish and maintain an inventory of all certificates and keys on their systems. Typically, there are several roles within a certificate owner group, including executives who are accountable for ensuring certificate-related responsibilities are addressed; system administrators who manage individual systems and the certificates on them, including requesting and installing certificates; and application owners. The certificate owners typically are not knowledgeable or familiar with the risks associated with certificates or the best practices for effectively managing them. Nonetheless, they must ensure their certificates are compliant by relying on the central certificate service technologies, expertise, and guidance supplied by the Certificate Services team.
- **Certificate Services Team:** This group includes experts that drive and support the organization's formal certificate management program. They manage relationships with public CAs to manage internal CAs, and provide the central certificate service that certificate owners use to establish and maintain their certificate and key inventories. This team is knowledgeable about TLS server certificates but typically lacks sufficient resources or access required to directly manage certificates on the extensive number of systems where certificates are deployed.
- **DevOps:** This group provisions systems and software through automated programmatic processes and tools known collectively as DevOps. It is a common practice to request and deploy TLS server certificates by using DevOps technologies.
- **Approvers:** Approvers serve as registration authorities within organizations. In this role, they review certificate signing requests, and confirm the validity of the request and the authority of the requester. They also send the approval of the certificate signing request to the certificate service or CA.

The internal and external components described above were integrated to create the TLS server certificate management example solution in the TLS lab. Figure 4-2 depicts the logical architecture of the example solution. The logical architecture shows the network structure and components that enable various types of TLS server certificate management operations. For several reasons, it is not intended to serve as a definitive example for an organization to model its own network design. For starters, it lacks a firewall, intrusion detection system, and other components an organization may use to secure its network. Although some IT professionals may consider these components essential to ensuring network security, they were not part of the logical architecture for the example implementation. The TLS team concluded that these components were not relevant in showcasing the TLS server certificate management functionality.

Figure 4-2 shows the logical architecture of the TLS server certificate management example implementation, which comprises an external CA and an internal network logically organized into three zones. These zones roughly model a defense-in-depth strategy of grouping components on subnetworks that require increasing levels of security as one moves inward from the perimeter of the organization: a demilitarized zone (DMZ) between the internet and the rest of the enterprise; a data center hosting applications and services widely used across the enterprise; and a more secure data center hosting critical security and infrastructure components, including certificate management components.

At the ingress from the internet within the DMZ, a load balancer is deployed to act as a TLS proxy— distributing incoming traffic from external users across three TLS servers behind it that are serving the same application: two Apache servers and one Microsoft internet information services (IIS) server. (Note: To simplify the illustration, the connections between individual components are not shown.) TLS certificate management is used to enroll and provision new certificates to the load balancer and servers in the DMZ, and to perform overall certificate management on these devices, including automatically replacing certificates nearing expiration.

Within the data center zone of the logical architecture sit various types of web servers, application servers, and a DevOps framework—all act as TLS servers. These components are used to demonstrate the ability to automatically enroll and provision a new certificate as well as automatically replace a certificate that is nearing expiration on these systems. Various types of certificate management are also demonstrated, including remote agentless management, the ACME protocol, and a DevOps certificate management plug-in.

Within the DMZ and the data center zone, taps (depicted as white dots) are used on the network connections between the load balancer, the servers behind it, and the network connections between the DMZ servers and the second-tier servers in the data center behind them. These taps send traffic on the encrypted TLS connections to a TLS inspection appliance for passive decryption. In Figure 4-2, this TLS inspection appliance is depicted by using a faded icon to convey that some organizations, as a matter of policy, may not want to include it as part of their network architecture. However, for those organizations that consider passive inspection as part of their security assurance strategy, the certificate manager depicted in the architecture can securely copy private keys from several different TLS servers to the TLS inspection appliance. It can also securely replace expiring keys on those servers and immediately copy them to the inspection appliance before expiration.

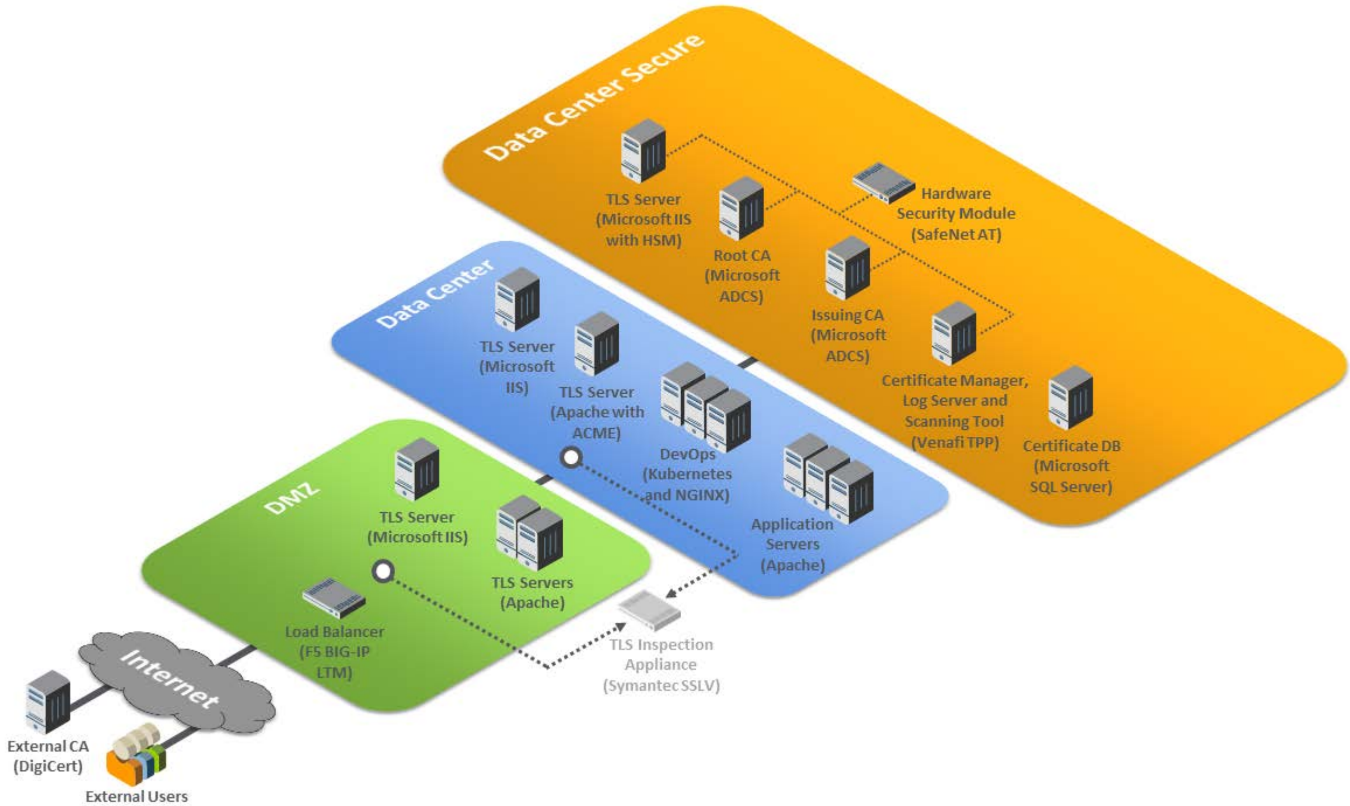
Within the data center secure zone of the logical architecture sit the components that perform TLS server certificate management: internal root and issuing CAs, a certificate manager, a certificate log server, a certificate network scanning tool, a certificate database, and an HSM. For demonstration purposes, a TLS server connected to the HSM is also present in this zone.

The certificate manager, in conjunction with the certificate database and the various types of servers in the rest of the architecture, demonstrates establishment and maintenance of a systematized inventory of certificates (and keys) in use on the network. The certificate manager also monitors the TLS certificates (and keys) managed by the inventory system and responds to any issues. For example, it will send expiration reports and notifications to certificate owners, informing them a certificate is being automatically replaced, is about to expire, or does not conform to policy. It also supports disaster recovery efforts by quickly replacing a large number of certificates located throughout the network architecture.

The certificate manager, in conjunction with the CAs, enrolls and provisions certificates (and keys), stores attributes with those certificates, and discovers the absence of an expected certificate from a machine where it should be installed. The certificate owner or the Certificates Services team can alert a certificate manager when a certificate must be revoked or if the owner associated with a certificate needs to be changed. The certificate scanning tool discovers certificates not currently being managed by the inventory. The certificate log server records all automated certificate and private-key management operations, including certificate creation, installation, and revocation; key pair generation; certificate requests and request approvals; certificate and key copying; and certificate and key replacement.

All components in the data center secure zone, except for the certificate database, are configured to use the HSM to securely generate, store, manage, and process private and symmetric keys. Cryptographic operations are performed within the HSM, ensuring that keys remain safe within its hardened confines rather than risk exposure outside it. The HSM stores and protects the symmetric keys that secure sensitive data in the certificate database. It generates, stores, manages, and performs signing operations with the internal CAs' signing keys and cryptographic operations with the TLS server private key.

676 **Figure 4-2 TLS Server Certificate Management Example Solution Logical Architecture**



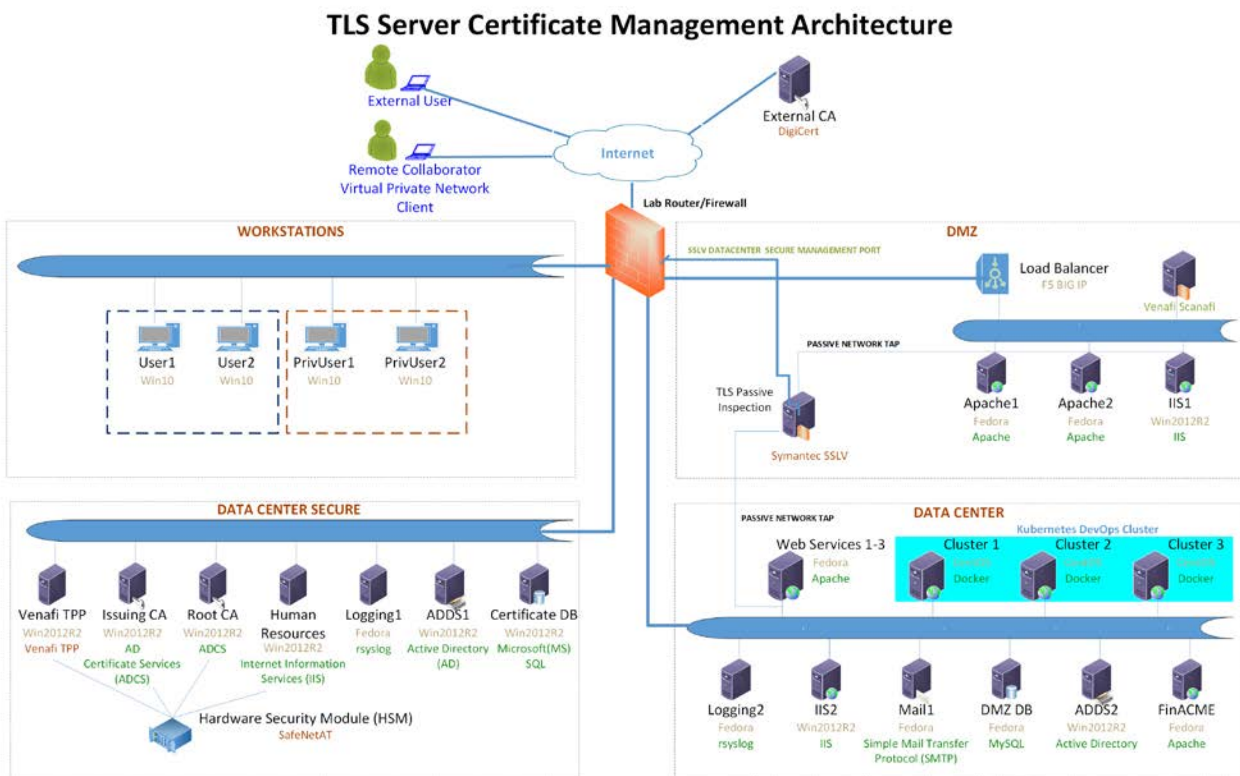
677

678

4.2 Physical Architecture

Figure 4-2 depicts the logical architecture deployed in the TLS lab to yield the TLS server certificate management example implementation. Figure 4-3 illustrates the laboratory configuration of that example implementation.

Figure 4-3 Laboratory Configuration of TLS Server Certificate Management Example Implementation



The NCCoE lab provides the following supporting infrastructure for the example implementation:

- firewall-protected connection to the internet, where an external CA resides
- Windows 2012 server with remote desktop manager that acts as a jump box to facilitate installation, deployment, and management of server software for collaborative projects
- segmented laboratory network backbone that models the separation that typically exists between subnetworks belonging to different parts of a medium-to-large-scale enterprise, such as a DMZ, data center hosting widely used applications and services, and a more secure data center hosting critical security infrastructure components
- virtual machine and network infrastructure
- Windows 2012 servers running Active Directory (AD) Certificate Services, including:
    - internal root CA that can issue and self-sign its own TLS certificate
    - internal issuing CA that:
        - issues TLS certificates to the servers that request them (issue CAs are subordinate to and certified by the root CA)
        - manages the life cycle of certificates (including request, issuance, enrollment, publication, maintenance, revocation, and expiration)
- Microsoft structured query language (SQL) Server hosting the database of TLS certificates and keys and corresponding configuration data
- DevOps automation framework, including Kubernetes, Docker, and Jetstack, that demonstrates automated certificate management when performing open-source container orchestration
- Apache, Microsoft IIS, and NGINX servers used to demonstrate various ways of managing TLS server certificates, including remote agentless certificate management, management via the ACME protocol (via the Certbot utility), and management via DevOps
- Apache servers used to demonstrate certificate management on second-tier internal application servers

The following collaborator-supplied components were integrated into the above supporting infrastructure to yield the TLS server certificate management example implementation:

- Venafi Trust Protection Platform (TPP), which performs automated TLS server certificate and private-key management, including monitoring, remediation, and rapid replacement of TLS certificates and keys; TLS certificate and key policy enforcement; automated certificate requests and renewals; automated network scanning for TLS certificates; and logging of certificate and private-key management operations
- SafeNet Assured Technologies (SafeNet AT) Luna SA 1700 hardware security module used to securely generate, store, manage, and process the cryptographic key pair and uses it to sign TLS certificates within a hardened, tamper-resistant physical appliance. It is also used to store other

keys, such as the database encryption key and the TLS certificate keys for the key manager component (Venafi TPP) and the CAs

- DigiCert external CA, which issues and renews TLS certificates
- F5 Networks BIG-IP Local Traffic Manager load balancer, which acts as a TLS proxy and distributes received traffic across a number of other TLS servers
- Symantec SSL Visibility, a visibility appliance used to inspect intercepted traffic on encrypted TLS connections

The supporting infrastructure components and the TLS-server-specific collaborator-supplied components are discussed further in the technologies section below. Installation, configuration, and integration of these components are described in detail in Volume D.

4.3 Technologies

Table 4-1 lists the technologies used in this project, and provides a mapping among the generic application term, the specific product used, and the security control(s) the product provides. Refer to Table 3-1 for an explanation of the NIST Cybersecurity Framework Subcategory codes.

Table 4-1 Products and Technologies

Component	Product	Functionality	Cybersecurity Framework Subcategories
Certificate manager	Venafi Trust Protection Platform	Automated monitoring, remediation, and rapid replacement of TLS certificates and keys; TLS certificate and key policy enforcement; automated certificate requests and renewals; workflow for required approvals.	PR.AC-4, ID.AM-2, PR.AC-1, PR.DS-2, PR.DS-3, PR.DS-6, PR.IP-2, PR.IP-3, PR.PT-1, DE.AE-5, RS.MI-2, RS.MI-3: Newly identified vulnerabilities are mitigated or documented as accepted risks.
Internal TLS certificate network scanning tool	Venafi TPP	Automated discovery of TLS certificates via network scanning.	PR.AC-1, PR.AC-4, DE.AE-5, DE.CM-1
Certificate log server	Venafi TPP	Used to log all certificate and private-key management operations.	PR.PT-1

Component	Product	Functionality	Cybersecurity Framework Subcategories
Internal root CA	Windows 2012 server running AD Certificate Services	Issues and self-signs its own TLS certificate.	PR.AC-1, PR.AC-4, PR.DS-2, PR.DS-3, PR.DS-6, PR.PT-1
Internal issuing CA	Windows 2012 server running AD Certificate Services	Issues TLS certificates to the servers that request them; issuing CAs are subordinate to and certified by the root CA. Manages the life cycle of certificates, including request, issuance, enrollment, publication, maintenance, revocation, and expiration.	PR.AC-1, PR.AC-4, PR.DS-2, PR.DS-3, PR.DS-6, PR.PT-1
Certificate database	Microsoft SQL Server	Database of TLS certificates and keys; for confidentiality, this database is encrypted, and the encryption key is stored in the hardware security module.	PR.AC-4, PR.DS-1
TLS inspection appliance	Symantec SSLV Appliance	Intercepts and inspects network traffic encrypted via TLS.	PR.AC-4, DE.CM-1
HSM	SafeNet AT Luna SA 1700	Securely generates, stores, manages, and processes the cryptographic key pair and uses it to sign TLS certificates within a hardened, tamper-resistant physical appliance. Also stores other keys, such as the database encryption key and the TLS certificate keys for the key manager component (Venafi) and the CAs. Can issue signed certificates in response to certificate signing requests (CSRs). Administrative access to this component may be supported by using either password-based or secure shell-based public key authentication.	PR.AC-1, PR.AC-3, PR.AC-4, PR.DS-1, PR.DS-2, PR.DS-3, PR.DS-6, PR.PT-1
External certificate authority	DigiCert External CA	Issues, discovers, installs, inspects, remediates, and renews TLS certificates.	PR.AC-1, PR.AC-4, PR.DS-2, PR.DS-3, PR.DS-6

Component	Product	Functionality	Cybersecurity Framework Subcategories
Load balancer	F5 Networks BIG-IP Local Traffic Manager	Acts as a TLS server and distributes received traffic across a number of other TLS servers.	PR.AC-7, PR.DS-2, PR.PT-5
DevOps framework	Kubernetes	Open-source container orchestration system for automating application deployment, scaling, and management.	PR.PT-5
Automated certificate management frameworks	Jetstack Cert-Manager Certbot	Jetstack Cert-Manager provides automated certificate management for Kubernetes. Certbot is an automated client that enrolls and deploys TLS certificates for web servers by using the ACME protocol.	PR.AC-1, PR.AC-4
TLS servers	Apache Microsoft IIS NGINX	The following TLS server configurations were deployed with a TLS server certificate managed as follows: Microsoft IIS: remote agentless certificate management Microsoft IIS attached to the SafeNet AT HSM: remote agentless certificate management Apache: remote agentless certificate management Apache: certificate management via the ACME protocol and certbot client NGINX on Kubernetes: Cert-Manager plug-in for automated certificate management of ingresses.	PR.AC-7, PR.DS-2, PR.PT-5
Application servers	Apache	These systems represented a second tier of internal application servers that were also deployed with TLS server certificates.	PR.AC-7, PR.DS-2, PR.PT-5

4.3.1 Certificate Manager and Internal TLS Certificate Network Scanning Tool

The certificate manager is a key element of the architecture, acting as the primary technology component of an organization's central certificate service. It creates and maintains an inventory of certificates and keys; provides a self-service portal for certificate owners; automates monitoring and remediation; rapidly replaces TLS certificates and keys; enforces TLS certificate and key policy; and enables central oversight, reporting, and auditing.

4.3.1.1 Venafi Trust Protection Platform

Venafi TPP serves as the certificate manager and provides the following certificate management functions:

- establishment and enforcement of TLS server certificate policies
- central inventory of TLS server certificates and private keys
- customer creation of custom metadata fields (e.g., Cost Center, Application ID) associated with certificates and other assets for reporting and accounting
- hierarchical organization of assets (e.g., certificates, applications, devices)
- certificate network scanning (discussed below)
- automated import of certificates from CAs
- onboard discovery of certificates and associated configuration parameters (specifically on F5 BIG-IP Local Traffic Manager [LTM] and Microsoft IIS in the lab)
- separation of duties and least-privilege access through granular access controls—assignable to groups or individuals
- self-service portal for onboarding and certificate management by certificate owners
- automated identification of TLS server certificate vulnerabilities, providing visibility through dashboards, reports, and alerts
- automated monitoring of certificate expiration dates, with configurable time frames for alerts sent prior to expiration
- automated monitoring of certificate operation status
- automated integration with internal and public CAs for certificate enrollment
- automated certificate life-cycle management via remote management connections
- agent-based automated certificate life-cycle management
- standard protocol support, including simple certificate enrollment protocol (SCEP) and ACME
- DevOps framework integration
- cloud platform integration, including Amazon Web Services and Azure

- Representational state transfer (REST)-based application programming interfaces (APIs)
- dual-control enforcement through workflow gates that can be applied at specific steps in the certificate life cycle, and can be assigned to groups and individuals with sufficient knowledge of application context to review and approve certificate requests
- integration with HSMs for private-key security
- integration with identity systems (e.g., Microsoft Active Directory, Lightweight Directory Access Protocol [LDAP] directories)
- central logging of all certificate management operations
- configurable event-based alerts, including delivery via simple mail transfer protocol, syslog, security incident and event management systems, ticketing systems, file, or database
- certificate revocation list (CRL) expiration monitoring to prevent outages caused by expired CRLs
- trust anchor management (e.g., root certificates) on TLS clients that act as relying parties for TLS server certificates
- load balanced architecture to support scalability, fault tolerance, and geographic distribution to support enterprise certificate operations
- Common Criteria certified

4.3.2 Internal TLS Certificate Network Scanning Tool

The internal TLS certificate network scanning tool provides automated discovery of TLS server certificates. It integrates with the certificate manager and enables the Certificate Services team and certificate owners to scrutinize newly discovered certificates for policy compliance and inclusion in the certificated inventory, if desired. An effective strategy for certificate network scanning is to use existing vulnerability scanning tools to pass discovered certificate information to the Certificate Services team. In some cases, organizational or technical constraints require that the Certificate Services team performs network scanning. Because a vulnerability scanning tool was not deployed in the lab, the team used Venafi TPP for certificate network scanning.

4.3.2.1 Venafi TPP for Certificate Network Scanning

Venafi TPP provides two different methods for certificate network scanning: scanning from a Venafi TPP server, and scanning from a command line utility called Scanafi. Both methods were used in the lab: the Venafi TPP server for scanning the data center network zones and Scanafi for scanning the DMZ. The Venafi TPP server provides the following functions for discovering TLS server certificates:

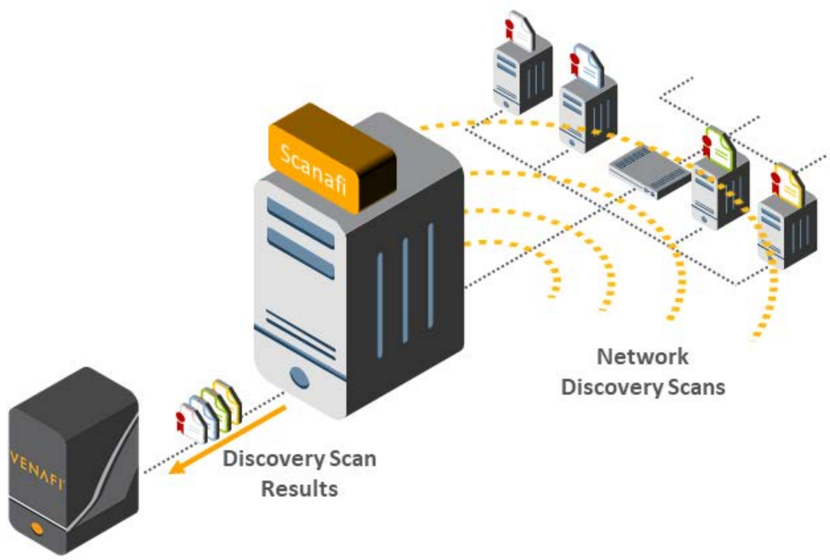
- support for the following as scanning targets:
  - multiple individual internet protocol (IP) addresses or IP ranges
  - multiple host/domain names

    - multiple ports or port ranges
- manual triggering of scans
- scheduled execution of scans, including daily, weekly, monthly, annually
- configuration of blackout periods for scanning
- support for multiple scanning agents
- support for placing scanning agents in distinct network zones (separated by firewalls)
- support for discovering TLS and SSL, including hypertext transfer protocol secure (https), the command STARTTLS, secure lightweight directory access protocol (LDAPS), file transfer protocol secure (FTPS), and server name indication (SNI)
- rules-based, automated processing of discovered certificates for placement into the certificate inventory hierarchy to automatically assign to the appropriate certificate owner(s)

Venafi Scanafi provides the following certificate network scanning functionality:

- support for the following as scanning targets:
    - multiple individual IP addresses or IP ranges
    - multiple host/domain names
    - multiple ports or port ranges
- manual triggering of scans (or triggering from a scheduling tool such as cron)
- support for multiple Scanafi agents (e.g., in different network zones)
- REST-based communications to the Venafi TPP server(s) to report scanning results
- support for discovery of TLS and SSL, including https, STARTTLS, LDAPS, FTPS, and SNI
- discovery of enabled TLS/SSL versions and ciphers for vulnerability identification

Figure 4-4 Venafi Scanafi Performing Network Scans and Providing Scan Results to Venafi TPP



4.3.3 Internal Root CA

The architecture includes an internal root CA that issues and self-signs its own TLS certificates for use in the demonstration. The NCCoE built its internal root CA by using a Windows 2012 server running Active Directory Certificate Services (ADCS).

4.3.4 Internal Issuing CA

The architecture also includes an internal issuing CA that issues TLS certificates to the servers that request them. The internal issuing CA is subordinate to and certified by the root CA. It manages the life cycle of certificates, including request, issuance, enrollment, publication, maintenance, revocation, and expiration. Similar to the internal root CA, the TLS team built its internal-issuing CA by using a Windows 2012 server running ADCS.

4.3.5 Certificate Database

The certificate database stores all TLS certificates and keys and associated metadata inventoried by the certificate manager. For confidentiality, private keys and credentials are encrypted in this database, and the encryption key is stored in the HSM.

4.3.5.1 Venafi TPP Database

The Venafi TPP database stores and provides access to the certificate inventory and product configuration data. The functions provided/supported by the Venafi TPP database include:

- storage of TLS server certificates, with the certificate fields' contents (e.g., key length, expiration date, common name) parsed and stored in separate database fields for rapid search
- storage of TLS private keys, encrypted by using an advanced encryption standard symmetric key stored in an HSM (or soft key if preferred)
- storage of TPP configuration data
- support for the following database versions:
    - Microsoft SQL Server 2012 SP2
    - Microsoft SQL Server 2014 SP2
    - Microsoft SQL Server 2016
- support for disaster recovery and high availability across multiple database instances through Microsoft SQL Server AlwaysON Availability Groups

4.3.6 TLS Inspection Appliance

Whether to perform TLS inspection is a policy decision left to each organization. For those organizations that require inspection, a TLS inspection appliance has been demonstrated with traffic that has been encrypted with TLS. The TLS inspection appliance decrypts this traffic, so it can be analyzed and inspected for viruses, malware, or other threats.

4.3.6.1 Symantec SSL Visibility Appliance

The SSLV Appliance inspects encrypted traffic to detect possible attacks. The Symantec device identifies and decrypts all TLS connections and applications across all network ports (even irregular ports). Existing and new security infrastructure can use the decrypted feeds to strengthen detection of and protection against advanced threats. By off-loading process-intensive decryption, the SSL Visibility Appliance also helps improve the overall performance of the organization's network and security infrastructure.

4.3.7 Hardware Security Module

HSMs are specialized devices dedicated to maintaining security of sensitive data throughout its life cycle. They provide tamper-evident and intrusion-resistant protection of critical keys and other secrets and can off-load processing-intensive cryptographic operations. By performing cryptographic operations within the HSM, sensitive data never leaves the secure confines of the hardened device. An HSM can securely generate, store, manage, and process cryptographic key pairs for use with TLS certificates. A CA leverages an HSM to issue signed certificates in response to certificate signing requests, while ensuring the CA signing keys remain safe within the confines of the HSM. In the build architecture, the HSM also stores other keys, such as the certificate database encryption key for the certificate manager component (Venafi).

4.3.7.1 SafeNet AT Luna SA 1700 HSM

SafeNet AT is a U.S.-based provider of high-assurance data security solutions with a stated mission to provide innovative solutions to protect the most vital data from the core to the cloud to the field. The company focuses on U.S. government defense, intelligence, and civilian agencies.

The SafeNet AT Luna SA for Government is a network-attached HSM with multiple partitions that provide a "many in one" solution to multiple tenants, each with its own security officer management credentials. Depending on security needs, the Luna SA works with or without a secure personal identification number entry device (PED) for controlling management access to the HSM partitions. Utilizing the PED takes the HSM from a FIPS 140-2 Level 2 certified device to Level 3. The Luna SA also comes in two performance models: the lower performance 1700 and the high-performance 7000 for transaction-intensive use cases.

In addition to the Luna SA, SafeNet AT offers Luna G5 for Government, which is a Universal Serial Bus-attached, small form-factor HSM. It is ideal for storing root cryptographic keys in an offline device. The Luna PCI-E for Government is an embedded HSM that can be installed in a server to protect cryptographic keys and accelerate cryptographic operations.

In the TLS Server Certificate Management Project, the Luna SA 1700 for Government was configured with two partitions to protect the keys that secure the Venafi Trust Protection Platform database and the Microsoft IIS root CA private key.

4.3.8 External Certificate Authority

The architecture also includes an external CA.

4.3.8.1 DigiCert External CA

DigiCert is a U.S.-based CA that provides a portfolio of PKI products, including digital certificates (SSL/TLS, Code Signing, Internet of Things [IoT], and more), CA deployment and operation, and tools for CA/PKI management.

DigiCert offers an external CA and management console to operate a deployed CA that is on site or cloud based. This full-service PKI management solution includes configuration of the CA (such as PKI hierarchy, certificate profiles, and revocation checking), certificate life-cycle management, network discovery of certificates, audit logs, and user roles. DigiCert's external CA is operated by the user through the CertCentral console.

CertCentral is a flexible web-based platform for enterprise and small business PKI management. CertCentral supports public and private PKI, and can manage and issue a wide variety of certificate types, including TLS (SSL), Code Signing, Client, Secure/Multipurpose Internet Mail Extensions, and Community standards (including Wi-Fi Alliance and Grid computing). CertCentral also offers a fully functioning API.

Through CertCentral, users can perform all certificate life-cycle operations, including certificate requests, approval/rejection of requests, certificate reissuance, and revocation. Because CertCentral is a centralized tool for certificate issuance and management, organizations can enforce their internal certificate policies and maintain certificates deployed across their networks.

CertCentral includes network scanning tools for identifying certificates installed on a network, regardless of the issuing CA. All discovered certificates are inventoried, and CertCentral will send an alert for expiring certificates and scan for common misconfigurations or security vulnerabilities in the web server and certificate (such as deprecated SSL protocol support or weak encryption ciphers/private keys). By using one tool, network administrators can monitor their PKI operation and receive alerts if problems emerge that can potentially cause network downtime or security risks.

CertCentral supports components of the ACME protocol—an IETF standard for automating issuance, installation, and renewal of SSL/TLS certificates. ACME enables web servers to automatically request and install their certificates, eliminating time-intensive replacement procedures and human error. This facilitates industry best practices such as short-lived certificates (usually 90-day validity or less) and regular key rotation.

An organization's CertCentral account can have as many users as needed, with each one having assigned preset or customizable roles. A user can be limited to what certificates they can request (by certificate type/identity), for which legal organizations/divisions they can make requests, and whether they can approve requests on their own or require an administrator/other approval. This gives users control to issue and manage their own certificates without affecting operations of other divisions within the organization. CertCentral supports two-factor authentication and single sign-on, which are potential requirements for specific roles or users.

Further capabilities and settings of CertCentral are described in the DigiCert Getting Started guide.

4.3.9 Load Balancer

The architecture includes a load balancer that acts as a reverse proxy. It receives client requests at its front end and evenly distributes these requests across a group of back-end TLS servers, which all use the same TLS server certificate and private key.

4.3.9.1 F5 Networks BIG-IP Local Traffic Manager

Businesses depend on applications. Whether the applications help connect businesses to their customers or help employees do their jobs, making these applications available and secure is the main goal. F5 BIG-IP LTM helps enterprises deliver their applications to users in a reliable, secure, and optimized way. It provides the extensibility and flexibility of application services, with the programmability enterprises need to manage their physical, virtual, and cloud infrastructure. With BIG-IP LTM, enterprises can simplify, automate, and customize applications quickly and predictably.

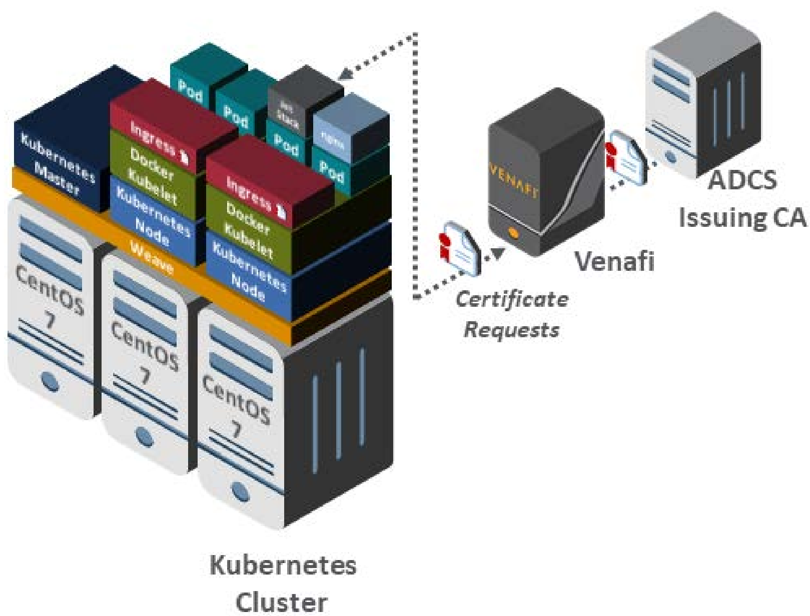
In the example solution architecture, the F5 BIG-IP LTM serves as a load balancer; it acts as a TLS proxy and distributes traffic it receives from external users across a cluster of TLS servers that sit behind it and are serving the same application. To handle traffic securely, each server in the cluster uses the same TLS server certificate and private key. Ideally, copying the keys to each of the servers is not performed manually; rather, automatic copying of private keys can reduce the possibility of a key compromise.

The example solution used in the Venafi TPP certificate manager automatically enrolls and provisions a new certificate to the F5 BIG-IP LTM to automatically replace a certificate on the BIG-IP LTM that was nearing its expiration. It can also configure the LTM's association with the servers behind it. The Venafi TPP certificate manager was also configured to automatically run a certificate discovery service on the F5 BIG-IP LTM, to identify new certificates and associated configuration parameters.

4.3.10 DevOps Framework

In this phase, the example solution architecture includes basic DevOps functionality for automated system and application deployment.

Figure 4-5 Example Implementation's DevOps Components Requesting and Receiving Certificates



4.3.10.1 Kubernetes

Kubernetes is an open-source container orchestration system for automating application deployment, scaling, and management. Kubernetes was deployed on three CentOS Linux systems: one acting as the master, and two nodes.

4.3.11 Automated Certificate Management Frameworks

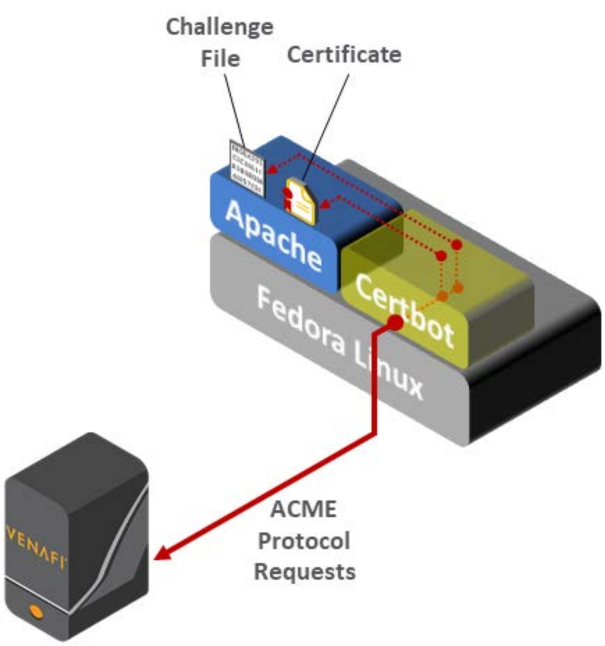
4.3.11.1 Jetstack Cert-Manager

As shown in Figure 4-5, Jetstack Cert-Manager was deployed and configured to automatically manage certificates for ingresses created on the Kubernetes cluster. A Cert-Manager issuer was defined to automatically request certificates from Venafi TPP, so ingress certificates on the Kubernetes cluster were automatically included in the central inventory and tracked (e.g., for expiration).

4.3.11.2 Certbot

Certbot is an open-source automatic client that fetches and deploys TLS certificates for web servers by using the ACME protocol. As shown in Figure 4-6, Certbot was deployed to automate management of certificates on an Apache system in the lab environment.

Figure 4-6 Certbot Fetching and Deploying TLS Certificates via the ACME Protocol



4.3.12 TLS Servers

The architecture included several TLS servers to demonstrate different methods of certificate management. The certificate management methods used in the example implementation included:

- **Remote Agentless Management:** Many existing “legacy” systems do not support standard protocols for certificate management. Consequently, it is necessary to remotely leverage available interfaces to perform certificate management operations. In this case, the certificate manager must authenticate itself to the system where a certificate is deployed, managed, and used. Once authenticated, it must then execute the necessary operations based on the semantics and syntax required by the system in question. Advantages of this approach include support for automated certificate management when built-in automation is not available, and the ability to centrally and rapidly respond to cryptographic events (e.g., CA compromise), because the certificate manager can proactively connect to each system and manage replacement of affected certificates. Some disadvantages to this approach include that the credentials and access must be granted to the certificate manager system, and integrations must be developed for each distinct type of system.

- **ACME Protocol:** The ACME protocol provides an efficient method for validating that a certificate requester is authorized for the requested domain and to automatically install certificates. This validation is performed by requiring the requester to place a random string (provided by the CA or certificate manager) on the server for verification via http or in a text record of the server's Domain Name System (DNS) entry. Client programs such as Certbot can automatically perform all of the operations needed to request a certificate—minimizing the manual work. Let's Encrypt and several other public CAs support the automated management of public-facing certificates by using the ACME protocol. However, public CAs cannot perform ACME validation for certificates installed on systems inside organizational networks. External entities cannot make http or DNS connections to internal systems. The certificate manager is able to make internal http and DNS connections and can be used for ACME-based certificate management on internal systems. A variety of CAs, certificate managers, and clients across a broad set of TLS servers and operating systems support the ACME protocol, which gives it an advantage. A disadvantage of ACME is that there is no central method for triggering a certificate replacement in response to a certificate event (e.g., CA compromise).
- **DevOps Plug-In:** DevOps frameworks can streamline development and deployment processes through add-on libraries and plug-ins that simplify specific programming tasks. Because certificate management is complex and error prone at times, leveraging certificate management plug-ins in DevOps frameworks increases security while minimizing risk. In this phase of the project, certificate management was implemented by using a plug-in for a single DevOps framework. In future phases, certificate management will be investigated more broadly for DevOps.

4.3.12.1 Microsoft IIS–Remote Agentless Management

Microsoft IIS was deployed on a Windows Server 2012 in the data center network zone. A certificate was manually deployed on IIS to simulate a scenario where existing certificates were deployed. The onboard discovery functionality in Venafi TPP was used to automatically discover the certificate and associated configuration (binding) information. This populated the necessary information for automated certificate management to occur. The certificate was automatically replaced by using Venafi TPP, which used Windows Remote Management to perform the remote certificate management operations.

4.3.12.2 Microsoft IIS with SafeNet AT HSM–Remote Agentless Management

Microsoft IIS was deployed on a Windows Server 2012 in the data center secure network zone. The SafeNet AT HSM client was installed on the Windows server to make the SafeNet AT HSM accessible for cryptographic operations through Windows Cryptographic Application Programming Interface (CAPI) or the next generation Cryptographic API. Configuration information for this IIS system was entered into Venafi TPP, including the address of the Windows system, credentials for authenticating to the Windows system, and information for the certificate needed for the IIS system. Venafi TPP automatically connected to the Windows system, instructed the HSM to generate a new key pair (for which the private key never left the HSM) and CSR, retrieved the CSR, enrolled for a certificate with the issuing CA, and installed the certificate with the necessary binding information for IIS. The https (TLS) connections were confirmed to use the issued certificate, and the corresponding private key was stored in the SafeNet AT HSM.

4.3.12.3 Apache–Remote Agentless Management

Apache was deployed on a Fedora Linux system in the DMZ. Configuration information for this Apache system was entered into Venafi TPP, including the address of the Fedora Linux system, credentials for authenticating to the Fedora Linux system, information for the certificate needed for the Apache system, and the location of the privacy enhanced mail files where the certificate and CA chain should be installed. Venafi TPP automatically enrolled for and deployed a certificate to the configured location, so the Apache server could use TLS-secured communications.

4.3.12.4 Apache–ACME Protocol

Apache was deployed on a Fedora Linux system in the DMZ. Certbot was installed on the Fedora Linux system and configured for use with Apache. The ACME server was enabled and configured on Venafi TPP, so Venafi TPP could service ACME protocol requests. Certbot was used to automatically request a certificate from Venafi TPP and install it for use by the Apache web server.

4.3.12.5 NGINX on Kubernetes–DevOps Plug-In

An NGINX deployment and corresponding service were created on the Kubernetes cluster. An ingress was defined to make the NGINX service accessible from outside the Kubernetes cluster. The needed annotation was included in the ingress definition to instruct Cert-Manager to automatically request and install a certificate from Venafi TPP. Once the ingress was enabled, a connection was made to the appropriate address to confirm the certificate from Venafi TPP was successfully installed to secure communications to the NGINX web server.

4.3.13 Application Servers

Most web-based applications include multiple tiers. For example, users of a web-based application may initially connect to a load balancer. The load balancer (tier 1) passes the requests to a web server (tier 2). The web server processes the requests and subsequently makes requests to one or more application servers (tier 3). The application servers process the requests and may read or write to/from a database server (tier 4). Credentials and other confidential information are often passed among adjacent tiers, so each system is typically configured for TLS, including a TLS certificate. The example solution implementation included a load balancer and two web servers in the DMZ. To simulate the existence of application servers, Apache systems were deployed in the data center network zone. NOTE: Apache is not normally used as an application server. However, it was used to minimize complexity of the example implementation. Venafi TPP was used to automatically deploy certificates to the Apache systems acting as application servers.

5 Security Characteristic Analysis

The purpose of the security characteristic analysis is to gauge the extent to which the project meets its objective of demonstrating how the processes for obtaining and maintaining TLS cryptographic certificates can be made less labor-intensive and error prone in medium and large IT enterprises. In addition, it seeks to understand the security benefits and drawbacks of the reference design.

5.1 Assumptions and Limitations

The security characteristic analysis has the following limitations:

- It is neither a comprehensive test of all security components nor a red-team exercise.
- It cannot identify all weaknesses.
- It does not include the lab infrastructure. It is assumed that devices are hardened. Testing these devices would reveal only weaknesses in implementation that would not be relevant to those adopting this reference architecture.

5.2 Functional Capabilities Demonstration

The demonstration shows the extent to which the example solution meets its design goals and stated security requirements.

5.2.1 Definitions

The following definitions apply to terms used in the description of functional capabilities demonstrated.

- discovery–finding new certificates that are not yet known or managed by the certificate management system
- monitoring–maintaining awareness about the status and characteristics of known certificates being managed by the certificate management system, including a determination of whether the certificates conform to policy
- sanctioned certificates–certificates issued by approved CAs
- unsanctioned certificates–certificates issued by CAs that are not approved
- enrolling–creating/issuing a certificate and storing it in the certificate management system inventory
- provisioning–deploying a certificate to a machine; also called *installing*

5.2.2 Functional Capabilities

The following functional TLS server certificate management capabilities were successfully demonstrated in the build phase.

Capability 1: The TLS example implementation demonstrates the ability to **establish a systematized inventory** of certificates (and keys) in use on the network. It enables a user to:

- efficiently **enroll and provision** certificates (and keys) by using:
  - public CA
  - internal CA
  - private key stored in file
  - private key stored in HSM
- store the following **attributes** with certificates in the inventory:
  - subject distinguished name (DN)
  - subject alternative name (SAN)
  - issue date (i.e., notBefore date)
  - expiration date (i.e., notAfter date)
  - issuing CA
  - key length
  - key algorithm (e.g., Rivest, Shamir, and Adleman [RSA], Elliptic Curve Digital Signature Algorithm)
  - signing algorithm
  - validity period (e.g., difference between notBefore and notAfter)
  - key usage flags
  - extended key usage flags
  - installed location(s) of certificate (e.g., IP or DNS address and file path)
  - certificate owner (group responsible for certificate)
  - contacts (the group of individuals that should be notified of issues)
  - approver(s) (parties responsible for reviewing issuance and renewal requests)
  - type of system (e.g., F5 LTM, Microsoft IIS, Apache)
- custom metadata field definition by organizations to associate organizationally relevant information with certificates, such as application identification, cost center, applicable regulations
- use network scanning to **discover certificates** not currently being managed by the inventory, including the ability to:

- discover TLS server certificates **across different network zones and on a variety of TLS server types** (e.g., load balancer, web server, application server, database, identity services, etc.)
- **discover and flag unsanctioned certificates** (i.e., certificates not from an approved CA)
    - enroll a new (sanctioned) certificate and provision it to replace the discovered unsanctioned certificate
- discover and enroll sanctioned certificates
    - end entity (e.g., the TLS server)
    - CA certificate chain certificates (root and intermediate CA certificates)
- discover the **absence of an expected certificate** from a machine where it should be installed
    - **reprovision** that certificate to that machine from the inventory

Capability 2: The TLS example implementation demonstrates the capability to **maintain the inventory** of TLS certificates (and keys). It enables a user to:

- **enroll (add) new certificates** (and keys) to the inventory and provision them to a network device
- **revoke certificates** that are suspected to be compromised or are no longer needed
- delete certificates and private keys from the machine/HSM where they had been installed
    - private key stored in file
    - private key stored in HSM
- **replace** a given **owner** associated with all certificates when that **person resigns or changes roles**
    - This is ideally handled by associating certificates with groups, so that users can join or leave the group without leaving certificates “orphaned” without an owner. In cases where there is an individual owner for a certificate, the individual’s management chain should be included in the group, or Certificate Services or an incident response team should be included to ensure that expiration and other alerts do not go unaddressed.

Capability 3: The TLS example implementation demonstrates the capability to **automatically enroll and provision** a new certificate and **automatically replace a certificate** that is **nearing expiration** on the following systems:

- F5 BIG-IP LTM: The TLS example implementation demonstrates the capability to install and replace a TLS certificate on a load balancer and configure the association with the applicable virtual server.
- Apache with Agentless Management: The implementation demonstrates automated management of certificates on an Apache web server by using a remotely initiated connection.

- Microsoft IIS with Agentless Management: The implementation demonstrates automated management of certificates on a Microsoft IIS web server by using a remotely initiated connection.
- Apache with ACME Protocol: The implementation demonstrates automated certificate management on an Apache web server by using the ACME protocol.
- Kubernetes: The implementation demonstrates automated installation and replacement before expiration of certificates on ingresses defined to allow access to services within Kubernetes.

Capability 4: The TLS example implementation demonstrates the capability to **continuously monitor** the TLS certificates (and keys) managed by the inventory system and to act upon the status of any certificate (e.g., report the status or replace a certificate as needed). The implementation should support these capabilities:

- Enroll and provision a new certificate to **replace** one that is found to **not conform to policy.**
- **Send weekly or monthly expiration reports** to certificate owners showing all of their certificates that are set to expire (e.g., within the next 90 or 120 days).
- Send **notifications** to owners regarding certificates that are **due to expire** within a near term (e.g., 30 days).
- **Send escalation notifications** to managers or incident response if a certificate has not been replaced within a short time of expiration (e.g., 15 days).
- **Enroll and provision new certificates** as existing certificates approach expiration.
    - manual request
    - standardized automated certificate installation

Capability 5: The TLS example implementation demonstrates the disaster recovery capability to **quickly replace a large number of certificates** located across multiple networks and on a variety of server types, because the certificates are no longer trusted. It is able to replace:

- all certificates issued by a given CA
    - This mimics the situation in which a large number of certificates are no longer trusted, because the CA that issued them has been compromised or become untrusted.
- all certificates with associated keys that are dependent on a specific cryptographic algorithm
    - This mimics the situation in which a large number of certificates are no longer trusted, because the algorithm on which they depend is no longer considered secure.
- all certificates with associated keys generated by the faulty cryptographic library after a specific date
    - This mimics the situation where large numbers of certificates are no longer trusted, because the keys associated with them were generated by a faulty cryptographic library after a bug was introduced into that library.

- the ability to track and report on replacement of large numbers of certificates, to monitor the progress of replacement and risk reduction

Capability 6: The TLS example implementation demonstrates the capability to perform **passive, out-of-line decryption** on TLS communications. The demonstration includes the following capabilities:

- verification the decrypted data matches the tapped, TLS-encrypted data
- ability to use the certificate management system to securely transfer private keys from several different TLS servers to the TLS inspection appliance
- ability to use the certificate management system to securely replace expiring keys on servers and immediately copy these to the inspection appliance before expiration
    - manually
    - via standardized automated certificate installation

Capability 7: The TLS example implementation demonstrates the capability to **log all certificate and private-key management operations**, including logging:

- certificate creation
- certificate installation
- certificate revocation
- key pair generation
- certificate requests
- certificate request approvals
- copying certificates and keys
- certificate and key replacement

5.2.3 Mapping to NIST SP 1800-16B Recommendations

The following table provides a mapping between the recommended policy requirements in Volume B of this practice guide (NIST SP 1800-16B) and the example implementation in the TLS Certificate Management lab.

Table 5-1 Mapping Between Volume B Policy Recommendations and the Example Implementation

1800-16B Recommended Requirement	Implementation in TLS Certificate Management Lab
Inventory	Venafi TPP was used to maintain an inventory of all certificates, including metadata fields associated with each certificate for tracking relevant infor-

1800-16B Recommended Requirement	Implementation in TLS Certificate Management Lab
	mation such as key length, signing algorithm, and installed locations. To create a comprehensive inventory of existing certificates, two Venafi TPP functions were used: 1) CA import, to retrieve all issued certificates from the Microsoft CA, and 2) network discovery, to discover all deployed certificates, including certificates that may have been issued by other CAs. Network discovery added location information for each certificate previously imported from the CA.
Ownership	Venafi TPP was used to track owners for certificates. In Venafi TPP, it is possible to assign individuals or groups as owners of each certificate. It is also possible to assign (individual or group) owners to groups of certificates by associating the owner to a folder, which applies the ownership to all certificates within the folder.
Approved CAs	The Venafi TPP dashboard was used to identify discovered certificates issued from unapproved CAs. These certificates were replaced with certificates from approved CAs by using Venafi TPP.
Validity Periods	The Venafi TPP dashboard was used to identify discovered certificates with a validity period longer than allowed (e.g., a three-year versus one-year validity period). These certificates were replaced with certificates with shorter, allowed validity periods by using Venafi TPP.
Key Length	The Venafi TPP dashboard was used to identify discovered certificates that contained keys smaller than allowed (e.g., 1024 bits versus 2048 bits). These certificates were replaced with certificates containing longer, allowed key lengths by using Venafi TPP.
Signing Algorithms	The Venafi TPP dashboard was used to identify discovered certificates signed with noncompliant algorithms (e.g., secure hash algorithm 1 [SHA-1]). These certificates were replaced with certificates that had been signed with compliant algorithms by using Venafi TPP.
Subject DN and SAN	Venafi TPP was configured to allow only certain domain names through domain white-listing. Workflow gates were implemented in Venafi TPP to ensure that Subject DNs and SANs in all certificate requests were reviewed and approved prior to issuance by the CA.
Certificate Request Reviews (Registration Authority)	Workflow gates were configured in Venafi TPP, requiring that certificates be reviewed prior to new issuance or renewal. Individuals/groups were assigned as approvers for groups of certificates via Venafi TPP folders.
Private-Key Security	The SafeNet AT HSM and Venafi TPP were used to secure private keys.

1800-16B Recommended Requirement	Implementation in TLS Certificate Management Lab
	SafeNet AT HSM and Venafi TPP: A Microsoft IIS server was connected to the SafeNet AT HSM across the network, so the private key used with the TLS server certificate on the IIS server could be stored and used within the HSM for a high level of security. Venafi TPP was used to manage generation of the key pair on the HSM. Venafi TPP: Automated management was used on several systems to remove the need for people to access private keys (which they do when manually managing TLS certificates).
Rotation upon Reassignment/ Termination	Venafi TPP was used create an up-to-date inventory, including tracking owners for all certificates. In case a certificate owner were reassigned or terminated, all certificates to which the person had management responsibility could be quickly identified. In addition to the ability to identify the certificates impacted by a reassignment or termination so they could be rotated, Venafi TPP and the SafeNet AT HSM were leveraged to minimize the need to rotate on reassignment. Venafi TPP was used to automate management of certificates and private keys, so that certificate owners did not require direct access to private keys, thereby removing the need to rotate certificates and private keys on reassignment or termination. On one system, additional steps were taken to protect private keys by leveraging the SafeNet AT HSM for protection of the private keys. The HSM prevents direct access to private keys, thereby removing the need to replace on reassignment.
Proactive Certificate Renewal	Venafi TPP was leveraged to monitor expiration dates of all certificates and send reports and alerts to certificate owners prior to expiration. Venafi TPP sent certificate expiration reports weekly showing all certificates expiring within the next 60 days, so certificate owners could proactively plan required replacements. Notification rules were configured in Venafi TPP, so alerts would be sent out if a certificate were within 20 days of expiring.
Crypto-Agility	Venafi TPP was used to establish an inventory of all certificates, so that in case of a large-scale cryptographic event (e.g., CA compromise, vulnerable cryptographic algorithm, or cryptographic library bug), all affected certificates and private keys could be quickly identified and replaced. Automation was configured on multiple systems to enable replacement of certificates and private keys to be completed quickly. In addition, Venafi TPP network validation was configured to automatically confirm the current status of all certificates, so the progress of replacement could be tracked.
Revocation	A workflow gate was configured in Venafi TPP to require review of revocation requests, so a certificate was not accidentally or maliciously revoked, which

1800-16B Recommended Requirement	Implementation in TLS Certificate Management Lab
	would cause an outage to the application dependent on the certificate. Permissions to request revocation were limited to certificate owners (for their own certificates) and administrative staff.
Continuous Monitoring	Venafi TPP was leveraged to perform the following to continuously monitor certificates: Network discovery scans were automatically performed on a periodic basis. Alerts were sent when new (previously unknown) certificates were detected. Venafi TPP network validation was configured to automatically check the operational status of all certificates. Onboard discovery was configured to automatically run periodically on the F5 LTM to discover new certificates.
Logging of Certificate Management Operations	Venafi TPP automatically logged all 1) administrative operations performed within the Aperture and WebAdmin consoles (e.g., new certificates, approvals, revocation requests), 2) API operations that made changes to configuration or data, 3) automated certificate management operations performed by Venafi TPP.
TLS Traffic Monitoring	The Symantec SSLV was deployed and configured to monitor all traffic on the data center and internal DMZ network zones. Private keys used for TLS certificates from the several TLS servers in those zones were automatically provisioned by Venafi TPP to the Symantec SSLV. When certificates on those servers were renewed, the new private keys were automatically provisioned to the SSLV.

5.3 Scenarios and Findings

One aspect of our security evaluation involved assessing how well the reference design addresses the security characteristics it was intended to support. The Cybersecurity Framework Subcategories were used to provide structure to the security assessment by consulting the specific sections of each standard cited in reference to a Subcategory. The cited sections provide validation points that the example solution would be expected to exhibit. Using the Cybersecurity Framework Subcategories as a basis for organizing our analysis allowed us to systematically consider how well the reference design supports the intended security characteristics.

5.3.1 Demonstration Scenario

The demonstration scenario starts with an organization that has deployed and currently uses TLS certificates across multiple groups and applications. In the scenario, an organization encounters the challenges described in Section 3. The approach followed to address the issues associated with life-cycle management of the certificates included the following phases:

- **Establish Governance:** The project team defined a set of certificate management policies based NIST guidance documents regarding how to establish consistent governance of TLS certificates.
- **Create and Maintain an Inventory:** A central team provided automated discovery services to certificate owners to establish a complete inventory of all TLS server certificates. The organization leveraged configurable rules to automatically organize discovered certificates and associate owners to enable automated notifications.
- **Register for and Install Certificates:** As new certificates were needed or existing certificates approached expiration, certificates were requested and installed. Because enterprise environments are diverse and have varying technical and organizational constraints, several methods for requesting and installing certificates were demonstrated. These included:
    - *Manual:* Security, operational, or technical requirements/constraints mandate that the server's system administrator manually requests a certificate by using command line tools and a certificate management system portal.
    - *Standardized Automated Certificate Installation:* A TLS server is configured to automatically request and install a certificate by using a protocol, such as IETF's ACME protocol.
    - *Installation Using Proprietary Method:* The certificate management system uses a method that is proprietary to the TLS server, to perform the operations needed to install certificates on one or more systems that do not support a standard automated method for requesting and installing certificates.
    - *DevOps-Based Installation:* A DevOps framework used to install and configure servers/applications is also used to request and install certificates. This was done in a cloud environment—where DevOps frameworks are most commonly used.
    - *Management of Private Keys Stored in an HSM:* The majority of private keys used with certificates are stored in files; however, HSMs increase the security of private keys. One or more of the methods listed above was performed on a system that uses an HSM for private-key protection.
- **Continuously Monitor and Manage:** The inventory of certificates was monitored for expiration, proper operation, and security issues. Notifications and alerts were triggered when certificates were nearing expiration or anomalies were detected. Management operations were performed to ensure proper operation and security.
- **Detect, Respond, and Recover from Incidents:** Simulated situations, such as a CA compromise and broken algorithms, were demonstrated (i.e., cryptographic library bug that created weak keys for certificates). A large number of organizational certificates needed to be rapidly replaced. The certificate management system orchestrated replacement of all certificates.

5.3.2 Findings

It is possible to deploy and configure a certificate management service and integrate it with ancillary components and services in such a way that the system

- establishes a TLS server certificate inventory by supporting functions such as certificate (and key) discovery, enrollment, provisioning, and revocation
- supports automatic enrollment and provisioning of new certificates
- supports automatic replacement of certificates nearing expiration
- discovers and monitors certificates and sends alerts as required to help avoid having certificates expire while they are still in use
- continuously monitors certificates to ensure their validity
- can quickly identify and replace a large number of certificates that share a common characteristic (e.g., they were all generated by a faulty cryptographic library) that may cause them to become untrusted
- can enroll and provision new certificates as well as automatically replace certificates that are nearing expiration on various types of systems, including Microsoft IIS and Apache web servers, application servers, load balancers, TLS proxies, and DevOps frameworks
- can perform certificate management via various types of mechanisms, including remote agentless management, the ACME protocol, and a DevOps certificate management plug-in
- can use an HSM to generate, store, manage, and process cryptographic key pairs for use with TLS server certificates and use these keys within the HSM to issue signed certificates in response to certificate signing requests
- can use an HSM to store and protect additional keys, such as the symmetric keys that secure sensitive data in the certificate database
- can efficiently and automatically copy private keys from servers to inspection appliances to enable inspection of traffic within encrypted TLS connections if desired
- can log all certificate and private-key management operations

Passive inspection of VMware vSphere workloads by using a remote physical monitoring appliance is challenging. Within the TLS lab deployment, passive decryption monitoring was deployed. This required that network packets captured within VMware vSphere workloads be forwarded to a physical remote monitoring appliance. The packet had to traverse the switch fabric between the VMware ESXi cluster and the physical remote monitoring appliance. VMware standard switches will monitor only east–west traffic locally in a standard switched port analyzer (SPAN) port configuration. VMware needs additional configuration to its virtual distributed switch configurations to support SPAN or mirroring ports. This method is discussed in more detail in Appendix A of Volume D.

There is an additional challenge with passive decryption of TLS traffic. TLS 1.3 prohibits use of the RSA algorithm, requiring use of ephemeral Diffie-Hellman instead. TLS passive inspection is not possible when ephemeral Diffie-Hellman is used. As a result, organizations must continue to use TLS 1.2 or earlier versions to perform TLS passive inspection of traffic on their internal networks. TLS passive inspection is possible with TLS 1.2 and earlier versions because the RSA algorithm is supported for key exchange.

6 Future Build Considerations

The expanding use of cloud environments and DevOps methodologies/tools, and reliance on TLS to secure communications necessitates implementation of sound TLS server certificate management methodologies. Future builds will focus on strategies for effectively managing TLS server certificates for cloud and DevOps, including strategies for adapting management methodologies as cloud environment and DevOps methodologies/tools continue to rapidly evolve and change. Future builds will look at strategies for managing TLS server certificates in individual cloud implementations, as well as implementations where multiple cloud environments are used or those requiring the ability to move implementation between clouds. For DevOps, we will investigate commonalities and differences for TLS server certificate management between the various types of DevOps methodologies and tools.

We have also received suggestions that we should investigate TLS server certificate management recommended best practices in the context of company acquisitions and divestitures, as well as investigate providing more detail regarding what certificate management aspects to audit against.

Appendix A List of Acronyms

ACME	Automated Certificate Management Environment
AD	Active Directory
ADCS	Active Directory Certificate Services
API	Application Programming Interface
CA	Certificate Authority
CAPI	Cryptographic Application Programming Interface (also known variously as CryptoAPI, Microsoft Cryptography API, MS-CAPI, or simply CAPI)
CRL	Certificate Revocation List
CSR	Certificate Signing Request
DevOps	Development Operations
DMZ	Demilitarized Zone
DN	Distinguished Name
DNS	Domain Name System
FIPS	Federal Information Processing Standards
FTPS	File Transfer Protocol Secure
HSM	Hardware Security Module
HTTP	Hypertext Transfer Protocol
HTTPS	Hypertext Transfer Protocol Secure
IETF	Internet Engineering Task Force
IIS	Internet Information Server (Microsoft Windows)
IoT	Internet of Things
IP	Internet Protocol
LDAP	Lightweight Directory Access Protocol
LTM	Local Traffic Manager (F5)
NCCoE	National Cybersecurity Center of Excellence
NIST	National Institute of Standards and Technology
PED	Personal Information Number Entry Device
PKI	Public Key Infrastructure
POP	Post Office Protocol
REST	Representational State Transfer (API)
RMF	Risk Management Framework

RSA	Rivest, Shamir, and Adleman (public key encryption algorithm)
SafeNet AT	SafeNet Assured Technologies
SAN	Subject Alternative Name
SCEP	Simple Certificate Enrollment Protocol
SHA-1	Secure Hash Algorithm 1
SNI	Server Name Indication
SP	Special Publication
SPAN	Switched Port Analyzer
SQL	Structured Query Language
SSL	Secure Socket Layer (protocol)
TLS	Transport Layer Security (protocol)
TPP	Trust Protection Platform (Venafi)
URL	Uniform Resource Locator

Appendix B Glossary

Active Directory	A Microsoft directory service for management of identities in Windows domain networks.
Application	1. The system, functional area, or problem to which information technology is applied. The application includes related manual procedures as well as automated procedures. Payroll, accounting, and management information systems are examples of applications. (National Institute of Standards and Technology [NIST] Special Publication [SP] 800-16). 2. A software program hosted by an information system (NIST SP 800-137).
Application Programming Interface (API)	A system access point or library function that has a well-defined syntax and is accessible from application programs or user code to provide well-defined functionality. (NIST Interagency/Internal Report [IR] 5153)
Authentication	Verifying the identity of a user, process, or device, often as a prerequisite to allowing access to a system's resources. (NIST SP 800-63-3)
Automated Certificate Management Environment	A protocol defined in Internet Engineering Task Force (IETF) Request for Comments (RFC) 8555 that provides automated enrollment of certificates.
Certificate	A set of data that uniquely identifies an entity, contains the entity's public key and possibly other information, and is digitally signed by a trusted party, thereby binding the public key to the entity. Additional information in the certificate could specify how the key is used and its validity period. (NIST SP 800-57 Part 1 Revision 4 under Public-Key Certificate) (Certificates in this practice guide are based on IETF RFC 5280).
Certificate Authority (CA)	A trusted entity that issues and revokes public key certificates. (NISTIR 8149)
Certificate Authority Authorization	A record associated with a Domain Name Server (DNS) entry that specifies the CAs authorized to issue certificates for that domain.
Certificate Chain	An ordered list of certificates that starts with an end-entity certificate, includes one or more CA certificates, and ends with the end-entity certificate's root CA certificate, where each certificate in the chain is the certificate of the CA that issued the previous certificate. By ascertaining whether each certificate in the chain was issued by a trusted CA, the receiver of an end-user certificate can determine if it

	should trust the end-entity certificate, by verifying the signatures in the chain of certificates.
Certificate Management	Process whereby certificates (as defined above) are generated, stored, protected, transferred, loaded, used, and destroyed (Committee on National Security Systems Instruction [CNSSI] 4009-2015) (In the context of this practice guide, it also includes inventory, monitoring, enrolling, installing, and revoking).
Certificate Revocation List	A list of digital certificates revoked by an issuing CA before their scheduled expiration date and should no longer be trusted.
Certificate Signing Request (CSR)	A request sent from a certificate requester to a CA to apply for a digital identity certificate. The certificate signing request contains the public key as well as other information to be included in the certificate and is signed by the private key corresponding to the public key.
Certificate Transparency	A framework for publicly logging the existence of Transport Layer Security (TLS) certificates as they are issued or observed, in a manner that allows anyone to audit CA activity and notice the issuance of suspect certificates, as well as to audit the certificate logs themselves (experimental RFC 6962).
Chief Information Officer	An organization's official who is responsible for (i) providing advice and other assistance to the head of the organization and to other senior management personnel to ensure that information technology (IT) is acquired and that information resources are managed in a manner consistent with laws, directives, policies, regulations, and priorities established by the head of the organization, (ii) developing, maintaining, and facilitating implementation of a sound and integrated IT architecture for the organization, and (iii) promoting the effective and efficient design and operation of all major information resources management processes for the organization, including improvements to work processes of the organization (NIST SP 800-53 Revision 4 adapted). Note: A subordinate organization may assign a chief information officer to denote an individual filling a position with security responsibilities with respect to the subordinate organization that are similar to those the chief information officer fills for the organization to which they are subordinate.
Client	1. A machine or software application that accesses a cloud over a network connection, perhaps on behalf of a consumer. (NIST SP 800-146) 2. A function that uses the public key infrastructure (PKI) to obtain certificates and validate certificates and signatures. Client functions

	are present in CAs and end entities. Client functions may also be present in entities that are not certificate holders. That is, a system or user that verifies signatures and validation paths is a client, even if it does not hold a certificate itself. (NIST SP 800-15)
Cloud Computing	A model for enabling ubiquitous, convenient, on-demand network access to a shared pool of configurable computing resources (e.g., networks, servers, storage, applications, and services) that can be rapidly provisioned and released with minimal management effort or service provider interaction. (NIST SP 800-145)
Common Name	An attribute type commonly found within a subject distinguished name in an X.500 directory information tree. When identifying machines, it is composed of a fully qualified domain name or internet protocol (IP) address.
Configuration Management	A collection of activities focused on establishing and maintaining the integrity of IT products and information systems through control of processes for initializing, changing, and monitoring the configurations of those products and systems throughout the system development life cycle. (NIST SP 800-53 Revision 4)
Container	A method for packaging and securely running an application within an application virtualization environment. Also known as an application container or a server application container. (NIST SP 800-190)
Cryptographic Application Programming Interface (CAPI)	An API included with Microsoft Windows operating systems that provides services to enable developers to secure Windows-based applications by using cryptography. While providing a consistent API for applications, CAPI allows specialized cryptographic modules (cryptographic service providers) to be provided by third parties, such as hardware security module (HSM) manufacturers. This enables applications to leverage the additional security of HSMs while using the same APIs they use to access built-in Windows cryptographic service providers (also known variously as CryptoAPI, Microsoft Cryptography API, MS-CAPI, or simply CAPI).
Cryptography API: Next Generation	The long-term replacement for CAPI.
Demilitarized Zone	A perimeter network or screened subnet separating a more-trusted internal network from a less-trusted external network.
Development Operations (DevOps)	A set of practices for automating the processes between software development and IT operations teams so that they can build, test, and release software faster and more reliably. The goal is to shorten

	the systems development life cycle and improve reliability while delivering features, fixes, and updates frequently in close alignment with business objectives.
Digital Certificate	Certificate (as defined above).
Digital Signature	The result of a cryptographic transformation of data that, when properly implemented, provides origin authentication, assurance of data integrity, and signatory nonrepudiation. (NIST SP 800-133)
Digital Signature Algorithm	One of the Federal Information Processing Standards (FIPS) for digital signatures based on the mathematical concept of modular exponentiations and the discrete logarithm problem. (FIPS 186-4)
Directory Service	A distributed database service capable of storing information, such as certificates and certificate revocation lists, in various nodes or servers distributed across a network (NIST SP 800-15) (In the context of this practice guide, a directory services stores identity information and enables authentication and identification of people and machines.)
Distinguished Name	An identifier that uniquely represents an object in the X.500 directory information tree. (RFC 4949 Version 2)
Domain	A distinct group of computers under a central administration or authority.
Domain Name	A name owned by a person or organization and consisting of an alphabetical or alphanumeric sequence, followed by a suffix indicating a top-level domain; used as an internet address to identify the location of web pages.
Domain Name Server	The internet's equivalent of a phone book. It maintains a directory of domain names, as defined by the DNS, and translates them to IP addresses.
Domain Name System (DNS)	The system by which internet domain names and addresses are tracked and regulated as defined by IETF RFC 1034 and other related RFCs.
Elliptic Curve Digital Signature Algorithm	Elliptic Curve Digital Signature Algorithm specified in ANSI X9.62 and approved in FIPS 186.
Enrollment	The process a CA uses to create a certificate for a web server or email user (NISTIR 7682) (In the context of this practice guide, enrollment applies to the process of a certificate requester requesting a certificate, the CA issuing the certificate, and the requester retrieving the issued certificate).

Extended Validation Certificate	A certificate used for https websites and software that includes identity information subjected to an identity verification process standardized by the CA Browser Forum in its Baseline Requirements that verifies the identified owner of the website for which the certificate has been issued has exclusive rights to use the domain; exists legally, operationally, and physically; and has authorized issuance of the certificate.
Federal Information Processing Standards	A standard for adoption and use by federal departments and agencies that has been developed within the Information Technology Laboratory and published by the National Institute of Standards and Technology, a part of the U.S. Department of Commerce. A FIPS covers some topic in IT to achieve a common level of quality or some level of interoperability. (NIST SP 800-161)
Hardware Security Module	A physical computing device that provides tamper-evident and intrusion-resistant safeguarding and management of digital keys and other secrets, as well as crypto-processing. FIPS 140-2 specifies requirements for HSMs.
Host Name	Host names are most commonly defined and used in the context of DNS. The host name of a system typically refers to the fully qualified DNS domain name of that system.
Hypertext Transfer Protocol (HTTP)	A standard method for communication between clients and web servers. (NISTIR 7387)
Internet Engineering Task Force	The internet standards organization made up of network designers, operators, vendors, and researchers that defines protocol standards (e.g., IP, transmission control protocol, DNS) through processes of collaboration and consensus.
Internet Message Access Protocol	A method of communication used to read electronic mail stored in a remote server. (NISTIR 7387)
Internet of Things (IoT)	As used in this publication, user or industrial devices connected to the internet. IoT devices include sensors, controllers, and household appliances.
Internet Protocol	The internet protocol, as defined in IETF RFC 6864, is the principal communications protocol in the IETF internet protocol suite for specifying system address information when relaying datagrams across network boundaries.
Lightweight Directory Access Protocol (LDAP)	In this document, LDAP refers to the protocol defined by RFC 1777, which is also known as LDAP V2. LDAP V2 describes unauthenticated retrieval mechanisms. (NIST SP 800-15)

Microservice	A set of containers that work together to compose an application. (NIST SP 800-190)
Organization	An entity of any size, complexity, or positioning within an organizational structure (e.g., a federal agency or, as appropriate, any of its operational elements). (NIST SP 800-39) This publication is intended to provide recommendations for organizations that manage their own networks (e.g., that have a chief information officer).
Outage	A period when a service or an application is not available or when equipment is not operational.
Payment Card Industry Data Security Standard	An information security standard, administered by the Payment Card Industry Security Standards Council, for organizations that handle branded credit cards from the major card schemes.
Personal Information Number Entry Device	An electronic device used in a debit-, credit-, or smart card-based transaction to accept and encrypt the cardholder's personal identification number.
Pivoting	A process where an attacker uses one compromised system to move to another system within an organization.
Post Office Protocol (POP)	A mailbox access protocol defined by IETF RFC 1939. POP is one of the most commonly used mailbox access protocols. (NIST SP 800-45 Version 2)
Private Key	The secret part of an asymmetric key pair that is used to digitally sign or decrypt data. (NIST SP 800-63-3)
Public CA	A trusted third party that issues certificates as defined in IETF RFC 5280. A CA is considered public if its root certificate is included in browsers and other applications by the developers of those browsers and applications. The CA/Browser Forum defines the requirements that public CAs must follow in their operations.
Public Key	The public part of an asymmetric key pair that is used to verify signatures or encrypt data. (NIST SP 800-63-3)
Public Key Cryptography	Cryptography that uses separate keys for encryption and decryption; also known as asymmetric cryptography. (NIST SP 800-77)
Public Key Infrastructure (PKI)	The framework and services that provide generation, production, distribution, control, accounting, and destruction of public key certificates. Components include the personnel, policies, processes, server platforms, software, and workstations used for administering certificates and public-private key pairs, including the ability to issue, maintain, recover, and revoke public key certificates. (NIST SP 800-53 Revision 4)

Registration Authority (RA)	An entity authorized by the CA system to collect, verify, and submit information provided by potential subscribers that is to be entered into public key certificates. The term RA refers to hardware, software, and individuals that collectively perform this function. (CNSSI 4009-2015)
Rekey	To change the value of a cryptographic key being used in a cryptographic system application; this normally entails issuing a new certificate on the new public key. (NIST SP 800-32 under Rekey) (a certificate)
Renew	The act or process of extending the validity of the data binding asserted by a public key certificate by issuing a new certificate (NIST SP 800-32). (The new certificate is typically used to replace the existing certificate, and both certificates typically contain the same subject domain name and subject alternative name information. It is a best practice to generate a new key pair and CSR, i.e., rekey, when renewing a certificate, but re-keying is not required by all CAs. Renewal is typically driven by expiration of the existing certificate but could also be triggered by a suspected private-key compromise or other event requiring the existing certificate to be revoked.)
Replace	The process of installing a new certificate and removing an existing one, so that the new certificate is used in place of the existing certificate on all systems where the existing certificate is being used.
Representational State Transfer	A software architectural style that defines a common method for defining APIs for web services.
Risk Management Framework	The Risk Management Framework, presented in NIST SP 800-37, provides a disciplined and structured process that integrates information security and risk management activities into the system development life cycle. (NIST SP 800-82 Revision 2)
Rivest, Shamir, and Adleman	An algorithm approved in FIPS 186 for digital signatures and in NIST SP 800-56B for key establishment. (NIST SP 800-57 Part 1 Revision 4)
Root Certificate	A self-signed certificate, as defined by IETF RFC 5280, issued by a root CA. A root certificate is typically securely installed on systems, so they can verify end-entity certificates they receive.
Root Certificate Authority	In a hierarchical PKI, the CA whose public key serves as the most trusted datum (i.e., the beginning of trust paths) for a security domain. (NIST SP 800-32)
Rotate	The process of renewing a certificate in conjunction with a rekey, followed by the process of replacing the existing certificate with the new certificate.

Secure Hash Algorithm 1	A hash function specified in FIPS 180-2, the Secure Hash Standard. (NIST SP 800-89)
Secure Hash Algorithm 256	A hash algorithm that can be used to generate digests of messages. The digests are used to detect whether messages have been changed since the digests were generated. (FIPS 180-4)
Secure Transport	Transfer of information by using a transport layer protocol that provides security between applications communicating over an IP network.
Server	A computer or device on a network that manages network resources. Examples include file servers (to store files), print servers (to manage one or more printers), network servers (to manage network traffic), and database servers (to process database queries). (NIST SP 800-47)
Service Provider	A provider of basic services or value-added services for operation of a network; generally refers to public carriers and other commercial enterprises. (NISTIR 4734)
Simple Certificate Enrollment Protocol (SCEP)	A protocol defined in an IETF internet draft specification that is used by numerous manufacturers of network equipment and software that are developing simplified means of handling certificates for large-scale implementation to everyday users, as well as referenced in other industry standards.
Simple Mail Transfer Protocol	The primary protocol used to transfer electronic mail messages on the internet. (NISTIR 7387)
Special Publication	A type of publication issued by NIST. Specifically, the Special Publication 800 series reports on the Information Technology Laboratory's research, guidelines, and outreach efforts in computer security and its collaborative activities with industry, government, and academic organizations. The 1800 series reports the results of National Cybersecurity Center of Excellence demonstration projects.
Subject Alternative Name	A field in an X.509 certificate that identifies one or more fully qualified domain names, IP addresses, email addresses, uniform resource identifiers, or user principal names to be associated with the public key contained in a certificate.
System Administrator	Individual responsible for installation and maintenance of an information system, providing effective information system utilization, adequate security parameters, and sound implementation of established information assurance policy and procedures. (CNSSI 4009-2015)

Team	A number of persons associated together in work or activity (Merriam-Webster). As used in this publication, a team is a group of individuals that has been assigned by an organization's management the responsibility to carry out a defined function or set of defined functions. Designations for teams as used in this publication are simply descriptive. Different organizations may have different designations for teams that carry out the functions described herein.
Transport Layer Security (TLS)	An authentication and security protocol widely implemented in browsers and web servers. TLS is defined by RFC 5246 and RFC 8446.
Trust Protection Platform	The Venafi Machine Identity Protection platform used in the example implementation described in this practice guide.
User Principal Name	In Windows Active Directory, this is the name of a system user in email address format, i.e., a concatenation of user name, the "@" symbol, and domain name.
Validation	The process of determining that an object or process is acceptable according to a predefined set of tests and the results of those tests. (NIST SP 800-152)
Web Browser	A software program that allows a user to locate, access, and display web pages.

Appendix C References

[1] E. Barker, *Recommendation for Key Management: Part 1: General*, NIST SP 800-57 Part 1, Revision 4, Gaithersburg, Md., Jan. 2016. Available: https://nvlpubs.nist.gov/nistpubs/SpecialPublications/NIST.SP.800-57pt1r4.pdf.

[2] E. Rescorla, *The Transport Layer Security (TLS) Protocol Version 1*.3, Internet Engineering Task Force, Apr. 2006. Available: https://www.ietf.org/rfc/rfc4346.txt.

[3] Executive Office of the President, Office of Management and Budget (OMB), *Managing Federal Information as a Strategic Resource*, OMB Circular A-130, July 28, 2016. Available: https://www.federalregister.gov/documents/2016/07/28/2016-17872/revision-of-omb-circular-no-a-130-managing-information-as-a-strategic-resource.

[4] Framework for Improving Critical Infrastructure Cybersecurity, Version 1.1, NIST, Gaithersburg, Md., Apr. 16, 2018. Available: https://nvlpubs.nist.gov/nistpubs/CSWP/NIST.CSWP.04162018.pdf.

[5] Joint Task Force Transformation Initiative, *Guide for Conducting Risk Assessments*, National Institute of Standards and Technology (NIST) Special Publication (SP) 800-30 Revision 1, Gaithersburg, Md., Sept. 2012. Available: https://csrc.nist.gov/publications/detail/sp/800-30/rev-1/final.

[6] Joint Task Force Transformation Initiative, *Risk Management Framework for Information Systems and Organizations: A System Life Cycle Approach for Security and Privacy*, NIST SP 800-37 Revision 2, Gaithersburg, Md., Dec. 2018. Available: https://nvlpubs.nist.gov/nistpubs/SpecialPublications/NIST.SP.800-37r2.pdf.

[7] Joint Task Force Transformation Initiative, *Security and Privacy Controls for Information Systems and Organizations*, Draft NIST SP 800-53 Revision 5, Gaithersburg, Md., Aug. 2017. Available: https://csrc.nist.gov/CSRC/media//Publications/sp/800-53/rev-5/draft/documents/sp800-53r5-draft.pdf.

[8] M. Georgiev et al., "The most dangerous code in the world: validating SSL certificates in non-browser software," *Proceedings of the 2012 ACM conference on Computer and Communications Security*, 2012, pp. 38–49. Available: http://doi.acm.org/10.1145/2382196.2382204.

[9] NIST Computer Security Resource Center Risk Management Framework guidance [Website]. Available: https://csrc.nist.gov/projects/risk-management/risk-management-framework-quick-start-guides.

[10] P. Grassi et al., *Digital Identity Guidelines*, NIST SP 800-63-3, Gaithersburg, Md., June 2017. Available: https://csrc.nist.gov/publications/detail/sp/800-63/3/final.

[11] S. Frankel et al., *Guide to IPsec VPNs*, NIST SP 800-77, Gaithersburg, Md., Dec. 2005. Available: http://nvlpubs.nist.gov/nistpubs/Legacy/SP/nistspecialpublication800-77.pdf.

[12] T. Dierks and E. Rescorla, *The Transport Layer Security (TLS) Protocol Version 1.2*, Request for Comments 5246, Internet Engineering Task Force, Aug. 2008. Available: https://www.ietf.org/rfc/rfc5246.txt.

[13] U.S. Department of Commerce, *Minimum Security Requirements for Federal Information and Information Systems*, Federal Information Processing Standard (FIPS 200), Mar. 2006. Available: http://nvlpubs.nist.gov/nistpubs/FIPS/NIST.FIPS.200.pdf.

[14] U.S. Department of Commerce, *Security Requirements for Cryptographic Modules*, FIPS Publication 140-2, (including change notices as of Dec. 3, 2002), May 2001. Available: http://nvlpubs.nist.gov/nistpubs/FIPS/NIST.FIPS.140-2.pdf.

[15] U.S. Department of Commerce*, Standards for Security Categorization of Federal Information and Information Systems*, FIPS Publication 199, Feb. 2004. Available: https://csrc.nist.gov/publications/detail/fips/199/final.

[16] W. Polk. et al, *Guidelines for the Selection, Configuration, and Use of Transport Layer Security (TLS) Implementations*, NIST SP 800-52 Revision 1, Gaithersburg, Md., Apr. 2014. Available: http://nvlpubs.nist.gov/nistpubs/SpecialPublications/NIST.SP.800-52r1.pdf.

NIST SPECIAL PUBLICATION 1800-16D

Securing Web Transactions

TLS Server Certificate Management

Volume D:
How-To Guides

Murugiah Souppaya
NIST

Mehwish Akram
Brandon Everhart
Brian Johnson
Brett Pleasant
Susan Symington
The MITRE Corporation

William C. Barker
Dakota Consulting

Paul Turner
Venafi

Clint Wilson
DigiCert

Dung Lam
F5

Alexandros Kapasouris
Symantec

Rob Clatterbuck
Jane Gilbert
SafeNet Assured Technologies

July 2019

DRAFT

This publication is available free of charge from:
https://www.nccoe.nist.gov/projects/building-blocks/tls-server-certificate-management

National Institute of Standards and Technology
U.S. Department of Commerce

NATIONAL CYBERSECURITY CENTER OF EXCELLENCE

DISCLAIMER

National Institute of Standards and Technology Special Publication 1800-16D Natl. Inst. Stand. Technol. Spec. Publ. 1800-16D, 217 pages, (July 2019), CODEN: NSPUE2

FEEDBACK

You can improve this guide by contributing feedback. As you review and adopt this solution for your own organization, we ask you and your colleagues to share your experience and advice with us.

Comments on this publication may be submitted to: tls-cert-mgmt-nccoe@nist.gov.

Public comment period: July 17, 2019 through September 13, 2019

All comments are subject to release under the Freedom of Information Act.

National Cybersecurity Center of Excellence
National Institute of Standards and Technology
100 Bureau Drive
Mailstop 2002
Gaithersburg, MD 20899
Email: nccoe@nist.gov

NATIONAL CYBERSECURITY CENTER OF EXCELLENCE

The National Cybersecurity Center of Excellence (NCCoE), a part of the National Institute of Standards and Technology (NIST), is a collaborative hub where industry organizations, government agencies, and academic institutions work together to address businesses' most pressing cybersecurity issues. This public-private partnership enables the creation of practical cybersecurity solutions for specific industries, as well as for broad, cross-sector technology challenges. Through consortia under Cooperative Research and Development Agreements (CRADAs), including technology partners—from Fortune 50 market leaders to smaller companies specializing in information technology security—the NCCoE applies standards and best practices to develop modular, easily adaptable example cybersecurity solutions using commercially available technology. The NCCoE documents these example solutions in the NIST Special Publication 1800 series, which maps capabilities to the NIST Cybersecurity Framework and details the steps needed for another entity to re-create the example solution. The NCCoE was established in 2012 by NIST in partnership with the State of Maryland and Montgomery County, Maryland.

To learn more about the NCCoE, visit https://www.nccoe.nist.gov/. To learn more about NIST, visit https://www.nist.gov.

NIST CYBERSECURITY PRACTICE GUIDES

NIST Cybersecurity Practice Guides (Special Publication 1800 series) target specific cybersecurity challenges in the public and private sectors. They are practical, user-friendly guides that facilitate the adoption of standards-based approaches to cybersecurity. They show members of the information security community how to implement example solutions that help them align more easily with relevant standards and best practices, and provide users with the materials lists, configuration files, and other information they need to implement a similar approach.

The documents in this series describe example implementations of cybersecurity practices that businesses and other organizations may voluntarily adopt. These documents do not describe regulations or mandatory practices, nor do they carry statutory authority.

ABSTRACT

Transport Layer Security (TLS) server certificates are critical to the security of both internet-facing and private web services. A large- or medium-scale enterprise may have thousands or even tens of thousands of such certificates, each identifying a specific server in its environment. Despite the critical importance of these certificates, many organizations lack a formal TLS certificate management program, and the ability to centrally monitor and manage their certificates. Instead, certificate management tends to be spread across each of the different groups responsible for the various servers and systems in an organization. Central security teams struggle to ensure certificates are being properly managed by each of these disparate groups. Where there is no central certificate management service, the organization is

at risk, because once certificates are deployed, current inventories must be maintained to support regular monitoring and certificate maintenance. Organizations that do not properly manage their certificates face significant risks to their core operations, including:

- application outages caused by expired TLS server certificates
- hidden intrusion, exfiltration, disclosure of sensitive data, or other attacks resulting from encrypted threats or server impersonation
- disaster-recovery risk that requires rapid replacement of large numbers of certificates and private keys in response to either certificate authority compromise or discovery of vulnerabilities in cryptographic algorithms or libraries

Despite the mission-critical nature of TLS server certificates, many organizations have not defined the clear policies, processes, roles, and responsibilities needed for effective certificate management. Moreover, many organizations do not leverage available automation tools to support effective management of the ever-growing numbers of certificates. The consequence is continuing susceptibility to security incidents.

This NIST Cybersecurity Practice Guide shows large and medium enterprises how to employ a formal TLS certificate management program to address certificate-based risks and challenges. It describes the TLS certificate management challenges faced by organizations; provides recommended best practices for large-scale TLS server certificate management; describes an automated proof-of-concept implementation that demonstrates how to prevent, detect, and recover from certificate-related incidents; and provides a mapping of the demonstrated capabilities to the recommended best practices and to NIST security guidelines and frameworks.

The solutions and architectures presented in this practice guide are built upon standards-based, commercially available, and open-source products. These solutions can be used by any organization managing TLS server certificates. Interoperable solutions are provided that are available from different types of sources (e.g., both commercial and open-source products).

KEYWORDS

Authentication; certificate; cryptography; identity; key; key management; PKI; private key; public key; public key infrastructure; server; signature; TLS; Transport Layer Security

DOCUMENT CONVENTIONS

The terms “shall” and “shall not” indicate requirements to be followed strictly in order to conform to the publication and from which no deviation is permitted.

The terms “should” and “should not” indicate that among several possibilities, one is recommended as particularly suitable, without mentioning or excluding others, or that a certain course of action is preferred but not necessarily required, or that (in the negative form) a certain possibility or course of action is discouraged but not prohibited.

The terms “may” and “need not” indicate a course of action permissible within the limits of the publication.

The terms “can” and “cannot” indicate a possibility and capability, whether material, physical, or causal.

CALL FOR PATENT CLAIMS

This public review includes a call for information on essential patent claims (claims whose use would be required for compliance with the guidance or requirements in this Information Technology Laboratory [ITL] draft publication). Such guidance and/or requirements may be directly stated in this ITL Publication or by reference to another publication. This call also includes disclosure, where known, of the existence of pending U.S. or foreign patent applications relating to this ITL draft publication and of any relevant unexpired U.S. or foreign patents.

ITL may require from the patent holder, or a party authorized to make assurances on its behalf, in written or electronic form, either:

a) assurance in the form of a general disclaimer to the effect that such party does not hold and does not currently intend holding any essential patent claim(s); or

b) assurance that a license to such essential patent claim(s) will be made available to applicants desiring to utilize the license for the purpose of complying with the guidance or requirements in this ITL draft publication either:

i) under reasonable terms and conditions that are demonstrably free of any unfair discrimination; or

ii) without compensation and under reasonable terms and conditions that are demonstrably free of any unfair discrimination.

Such assurance shall indicate that the patent holder (or third party authorized to make assurances on its behalf) will include in any documents transferring ownership of patents subject to the assurance, provisions sufficient to ensure that the commitments in the assurance are binding on the transferee, and that the transferee will similarly include appropriate provisions in the event of future transfers with the goal of binding each successor-in-interest.

The assurance shall also indicate that it is intended to be binding on successors-in-interest regardless of whether such provisions are included in the relevant transfer documents.

Such statements should be addressed to tls-cert-mgmt-nccoe@nist.gov.

ACKNOWLEDGMENTS

We are grateful to the following individuals for their generous contributions of expertise and time.

Name	Organization
Dean Coclin	DigiCert
Tim Hollebeek	DigiCert
Robert Smith	F5
Nancy Correll	The MITRE Corporation
Mary Raguso	The MITRE Corporation
Aaron Aubrecht	Venafi
Justin Hansen	Venafi

The Technology Partners/Collaborators who participated in this build submitted their capabilities in response to a notice in the Federal Register. Respondents with relevant capabilities or product components were invited to sign a Cooperative Research and Development Agreement (CRADA) with NIST, allowing them to participate in a consortium to build this example solution. We worked with:

Technology Partner/Collaborator	Build Involvement
DigiCert	External Certificate Authority and CertCentral console
F5	BIG-IP Local Traffic Manager load balancer
SafeNet Assured Technologies	Luna SA 1700 Hardware Security Module
Symantec	SSL Visibility Appliance for TLS interception and inspection

Technology Partner/Collaborator	Build Involvement
Venafi	Trust Protection Platform (TLS certificate manager, log server, and scanning tool)

Contents

1 Introduction 1
1.1 Practice Guide Structure 1
1.2 Build Overview 3
1.2.1 Usage Scenarios 3
1.2.2 Logical Architecture 5
1.3 Build Architecture Summary 8
1.4 Typographic Conventions 11
1.5 Supporting Infrastructure 11
1.5.1 Lab Backbone 12
1.5.2 Supporting Infrastructure Operating Systems 13
1.5.3 Supporting Infrastructure Component Services 17
1.5.4 Database Services 27
1.5.5 TLS Web Services 29
1.5.6 DevOps Services 39
2 Product Installation and Configuration Guides 40
2.1 Product Installation Sequence (Example Build) 41
2.2 SafeNet AT Luna SA 1700 Hardware Security Module 42
2.2.1 Day 0: Product Installation and Standard Configuration 42
2.2.2 Day 1: Product Integration Configuration 54
2.2.3 Day N: Ongoing Security Management and Maintenance 88
2.3 DigiCert Certificate Authority 90
2.3.1 Day 0: Installation and Standard Configuration 90
2.3.2 Day 1: Integration Configuration 96
2.3.3 Day N: Ongoing Security Management and Maintenance 100
2.4 F5 BIG-IP Local Traffic Manager (LTM) 106
2.4.1 Day 0: Installation and Standard Configuration 106
2.4.2 Day 1: Product Integration Configuration 118
2.4.3 Day N: Ongoing Security Management and Maintenance 122

2.5 Symantec SSL Visibility Appliance 131
2.5.1 Day-0: Install and Standard Configuration 131
2.5.2 Day 1: Product Integration Configuration 141
2.5.3 Day N: Ongoing Security Management and Maintenance 149
2.6 Venafi Trust Protection Platform (TPP) 150
2.6.1 Prerequisites 150
2.6.2 Installation 150
2.6.3 CA Integration 158
2.6.4 Folder Creation 159
2.6.5 Custom Fields 160
2.6.6 Assigning Certificate Owners 161
2.6.7 Setting Policies 162
2.6.8 Domain Whitelisting 164
2.6.9 Workflow – RA Reviews 165
2.6.10 CA Import 166
2.6.11 Network Discovery 168
2.6.12 Identify Certificate Risks/Vulnerabilities 168
2.6.13 Automate Management 169
2.6.14 Continuous Monitoring 187
Appendix A Passive Inspection 192
Appendix B Hardening Guidance 194
Appendix C Venafi Underlying Concepts 196
C.1 Venafi TPP Object Model 198
C.2 Certificate Metadata in Venafi TPP 199
C.3 Custom Fields 201
C.3.1 Organizing Certificate Inventory 201
C.3.2 Policy Enforcement 202
C.4 Domain Whitelisting 202
C.4.1 Certificate Owner Assignment 202
C.4.2 Permissions 202

C.4.3 Contacts 203
Appendix D List of Acronyms 204
Appendix E Glossary 208
Appendix F References 216
Appendix G Supplemental Architecture Configurations........................ 217
G.1 Mail Server Configuration Files 217

List of Figures

Figure 1-1 TLS Server Certificate Management Example Implementation: Logical Architecture............6
Figure 1-2 TLS Server Certificate Management Example Implementation: Laboratory Configuration 9
Figure 1-3 TLS Lab Logging Infrastructure 38
Figure 2-1 Overview of Dependencies Among Components Deployed for the Example Build.............41
Figure 2-2 Venafi Dashboard Expiration Widget showing the Certificate Expiration Profile..............169

List of Tables

Table 1-1 Naming and Addressing Information for all Microsoft Windows Servers 14
Table 1-2 Naming and Addressing Information for all Microsoft Windows 10 Workstations 15
Table 1-3 Naming and Addressing Information for All Fedora-Based Systems 16
Table 1-4 Naming and Addressing Information for All CentOS Servers 17

1 Introduction

Organizations that improperly manage their Transport Layer Security (TLS) server certificates risk system outages and security breaches, which can result in revenue loss, harm to reputation, and exposure of confidential data to attackers. TLS is the most widely used protocol for securing web transactions and other communications on internal networks and the internet. TLS certificates are central to the operation and security of internet-facing and private web services. Some organizations have tens of thousands of TLS certificates and keys requiring ongoing maintenance and management.

The National Cybersecurity Center of Excellence (NCCoE) at the National Institute of Standards and Technology (NIST) built a laboratory environment to demonstrate how large and medium enterprises can better manage TLS server certificates in the following ways:

- defining operational and security policies and identifying roles and responsibilities
- establishing comprehensive certificate inventories and ownership tracking
- conducting continuous monitoring of the certificate operation and security status
- automating certificate management to minimize human error and maximize efficiency on a large scale
- enabling rapid migration to new certificates and keys as needed in response to certificate authority (CA) compromise or discovery of vulnerabilities in cryptographic algorithms or libraries

The following volumes of this guide show information technology (IT) professionals and security engineers how we implemented this example solution. We cover all the products employed in this reference design. We do not re-create the product manufacturers' documentation, which is presumed to be widely available. Rather, these volumes show how we incorporated the products together in our environment.

Note: These are not comprehensive tutorials. There are many possible service and security configurations for these products that are out of scope for this reference design.

1.1 Practice Guide Structure

This National Institute of Standards and Technology (NIST) Cybersecurity Practice Guide demonstrates a standards-based reference design and provides users with the information they need to replicate automated management of TLS server certificates. This reference design is modular and can be deployed in whole or in part.

This guide contains four volumes:

- NIST SP 1800-16A: *Executive Summary*
- NIST SP 1800-16B: *Security Risks and Recommended Best Practices*
- NIST SP 1800-16C: *Approach, Architecture, and Security Characteristics*–what we built and why

- NIST SP 1800-16D: *How-To Guides*—instructions for building the example solution **(you are here)**

Depending on your role in your organization, you might use this guide in different ways:

Business decision makers, including chief security and technology officers, will be interested in the *Executive Summary,* NIST SP 1800-16A, which describes the following topics:

- recommendations for TLS server certificate management
- challenges that enterprises face in proper deployment, management, and use of TLS
- example solution built at the NCCoE

You might share the *Executive Summary*, NIST SP 1800-16A, with your leadership team members to help them understand the importance of adopting standards-based TLS server certificate management.

Senior information technology and security officers will be informed by NIST SP 1800-16B, which describes the:

- TLS server certificate infrastructure and management processes
- risks associated with mismanagement of certificates
- organizational challenges associated with server certificate management
- recommended best practices for server certificate management
- recommendations for implementing a successful certificate management program
- mapping of best practices for TLS server certificate management to the NIST Framework for Improving Critical Infrastructure Cybersecurity (Cybersecurity Framework)
- application of specific controls defined within NIST Special Publication (SP) 800-53 to the TLS server certificate management recommended best practices

Technology or security program managers who are concerned with how to identify, understand, assess, and mitigate risk will be interested in NIST SP 1800-16C, which describes what we did and why. The following sections will be of particular interest:

- Section 3.4.1, Threats, Vulnerabilities and Risks, provides a description of the risk analysis we performed.
- Section 3.4.2, Security Categorization and SP 800-53 Controls, lists the security controls assigned to address TLS server certificate risks.
- Section 3.4.3, Security Control Map, maps the security characteristics of this example solution to cybersecurity standards and best practices.

IT professionals who want to implement such an approach will find this whole practice guide useful. You can use this How-To portion of the guide, NIST SP 1800-16D, to replicate all or parts of the build created in our lab. This How-To portion of the guide provides specific product installation, configuration, and integration instructions for implementing the example solution. We do not re-create the product manufacturers' documentation, which is generally widely available. Rather, we show how we incorporated the products together in our environment to create an example solution.

This guide assumes that IT professionals have experience implementing security products within the enterprise. While we have used a suite of commercial and open source products to address this challenge, this guide does not endorse these particular products. Your organization can adopt this solution or one that adheres to these guidelines in whole, or you can use this guide as a starting point for tailoring and implementing parts of providing automation support for TLS server certificate management. Your organization's security experts should identify the products that will best integrate with your existing tools and IT system infrastructure. We hope that you will seek products that are congruent with applicable standards and best practices. Section 1.4.2, Technologies, lists the products that we used and maps them to the cybersecurity controls provided by this reference solution.

A NIST Cybersecurity Practice Guide does not describe "the" solution, but a possible solution. This is a draft guide. We seek feedback on its contents and welcome your input. Comments, suggestions, and success stories will improve subsequent versions of this guide. Please contribute your thoughts to tls-cert-mgmt-nccoe@nist.gov.

1.2 Build Overview

This NIST Cybersecurity Practice Guide addresses the use of commercially available technologies to develop an example implementation for managing TLS server certificates. This project focuses on certificate management in medium and large enterprises that rely on TLS to secure customer-facing and internal applications. The example implementation developed in this project demonstrates how to manage TLS server certificates to reduce outages, improve security, and enable disaster recovery activities. It shows how to establish, assign, change, and track an inventory of TLS certificates; automate management of TLS certificates; perform continuous monitoring of TLS certificates; perform large-scale replacement of certificates that are not trusted; log all certificate and private-key management operations; manage certificates and keys on proxy servers, load balancers, and inspection appliances; and use a Hardware Security Module (HSM). The HSM can securely generate, store, manage, and use private keys corresponding to TLS server certificates, the signing keys of internal certificate authorities (CAs), and symmetric keys that must be kept secret.

1.2.1 Usage Scenarios

The example implementation fulfills the following use cases:

- building and maintaining inventory of the enterprise's deployed TLS server certificates
- automating management of those certificates, including use of an external CA and protection of private keys and other secrets by using an HSM

- continuously monitoring the certificates for validity
- supporting disaster recovery by quickly replacing a large number of certificates
- logging all certificate and private-key management operations
- for those enterprises with a policy to perform passive inspection, copying private keys from several different TLS servers to the TLS inspection appliance

1.2.1.1 Building the Inventory

The example implementation demonstrates the ability to establish and maintain a systematized inventory of certificates (and keys) in use on the network. It enables a user to discover certificates not currently being managed by the inventory, efficiently enroll and provision new certificates (and keys), store relevant information with those certificates, and discover the absence of an expected certificate from a machine where it should be installed. It also enables certificates to be revoked and to change the owner associated with a certificate, as needed.

1.2.1.2 Automation

The example implementation demonstrates the ability to automatically enroll and provision a new certificate and can replace a certificate approaching expiration. Automated certificate management is demonstrated on various enterprise systems, including load balancers acting as TLS proxies that use remote agentless management, web servers with remote agentless management, web servers using the Automatic Certificate Management Environment (ACME) protocol, and servers that are deployed via development operations (DevOps) technologies by using a certificate management plug-in to the DevOps framework. In conjunction with the demonstration of ACME, HSM is used to securely generate, store, manage, and process the cryptographic key pairs for one TLS server. Remote agentless management was used to automate management of the certificates and keys for this system.

1.2.1.3 Continuous Monitoring

The example implementation demonstrates the ability to continuously monitor TLS certificates (and keys) managed by the inventory system and can act upon the status of any certificate (e.g., report the status of or replace a certificate that has expired, is about to expire, or does not conform to policy). It can send periodic expiration reports to certificate owners to show which of their certificates are nearing expiration, and a variety of notifications and escalating alerts if a certificate's expiration date approaches. Continuous monitoring also includes periodic network scans to ensure any unaccounted-for certificates are discovered and added to the inventory.

1.2.1.4 Disaster Recovery

The example implementation demonstrates how to quickly replace large numbers of certificates that are located across multiple networks and that are on a variety of server types, because the certificates are no longer trusted. It can replace certificates that:

- were issued by a given CA (which would require replacement if the issuing-CA were either compromised or untrusted)
- have associated keys dependent on a specific cryptographic algorithm (which would need replacement, e.g., if the algorithm they depend on is no longer considered secure)
- have associated keys generated by a specific cryptographic library after a specific date (which would need replacement, e.g., if a bug invaded a library on that date)

The example implementation can also track and report on replacement of large numbers of certificates, so the progress of the large-scale certificate replacement effort can be monitored.

1.2.1.5 Logging

The example implementation demonstrates how to log all certificate and private-key management operations, including certificate creation, installation and revocation key pair generation, certificate requests and request approvals, certificate and key copying, and certificate and key replacement.

1.2.1.6 Passive Inspection

The example implementation demonstrates how to perform passive inspection of encrypted TLS connections. The decision to perform this inspection is complex, because it involves important trade-offs between traffic security and traffic visibility that each organization should weigh for itself. Some organizations have determined that the security risks posed by inspection of internal TLS traffic are not worth the potential benefits of visibility into the encrypted traffic. Other organizations have concluded that the visibility into their internal traffic provided by TLS inspection is worth the trade-off of the weaker encryption and other risks that come with such inspection. For these organizations, TLS inspection may be considered standard practice and may represent a critical component of their threat detection and service assurance strategies.

Organizations that perform TLS traffic inspections can use the example implementation to securely copy private keys from several different TLS servers to the TLS inspection appliance, securely replace expiring keys on servers, and immediately copy those keys to the inspection appliance before expiration—manually and via standardized automated certificate installation. See Appendix A for more detail on passive inspection, including a scenario.

1.2.2 Logical Architecture

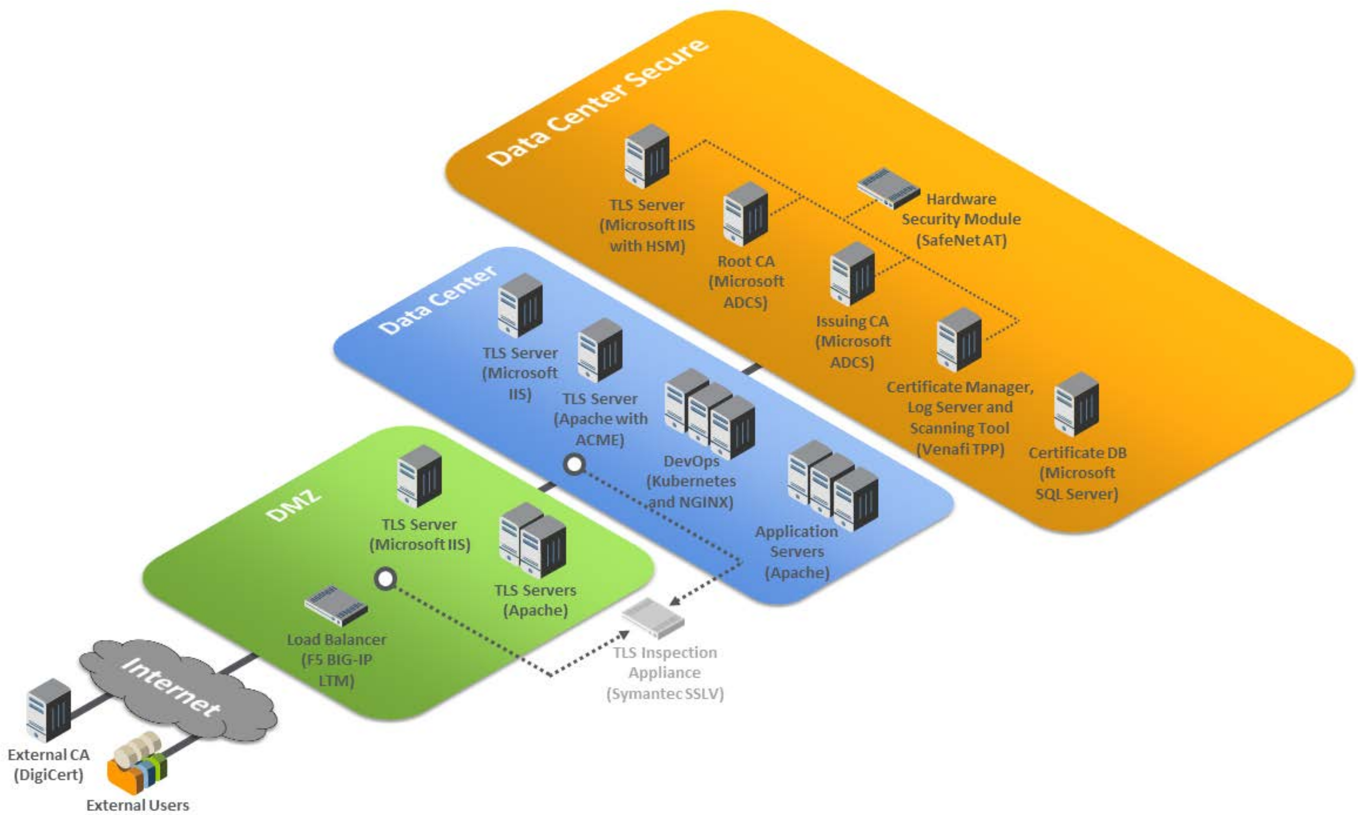
Figure 1-1 depicts the example implementation’s logical architecture, which provides a network structure and components that enable various types of TLS server certificate management operations to function. Figure 1-1 illustrates the logical architecture of the TLS server certificate management example implementation—consisting of an external and an internal portion. The external portion contains an external CA that is used to issue TLS certificates for some TLS servers in the example implementation. The internal portion of the network is logically organized into three zones that roughly model a defense-in-depth strategy of grouping components on subnetworks that require increasing levels of security as one moves inward from the perimeter of the organization. The zones comprise a demilitarized zone (DMZ) that sits between the internet and the rest of the enterprise; a data center hosting applications and services widely used across the enterprise; and a more secure data center hosting critical security and infrastructure components, including certificate management components.

At the ingress from the internet within the DMZ, a load balancer acts as a TLS proxy and distributes the traffic it receives from external users across three TLS servers behind it—all serving up the same application: two Apache servers and one Microsoft Internet Information Services (IIS) server. (Note: To maintain the diagram's simplicity in depicting this network, the connections between individual components are not shown. In the actual network architecture, the load balancer's network connection to all three TLS servers is shown behind it.) TLS certificate management demonstrates how to enroll and provision new certificates to the load balancer and servers in the DMZ and how to perform overall certificate management on these devices, including automatically replacing a certificate that is nearing expiration.

Within the data center zone of the logical architecture sit various types of web servers, application servers, and a DevOps framework—all act as TLS servers. These components demonstrate the ability to automatically enroll and provision a new certificate and can automatically replace a certificate that is nearing expiration on these different systems. Various types of certificate management are also demonstrated, including remote agentless management, the ACME protocol, and the DevOps certificate management plug-in.

Within the DMZ and the data center zones, taps (depicted as white dots) are used on the network connections between the load balancer and the servers behind it, and on the network connections between the DMZ servers and the second-tier servers in the data center behind them. Taps enable all traffic on the encrypted TLS connections to travel to a TLS inspection appliance for passive decryption. Figure 1-1 depicts this TLS inspection appliance as a faded icon to convey that some organizations, as a matter of policy, may not want to include it as part of their network architecture. However, organizations that consider passive inspection as part of their security assurance strategy can use the certificate manager depicted in the architecture to securely copy private keys from several different TLS servers to the TLS inspection appliance, and to securely replace expiring keys on those servers and immediately copy those keys to the decryption device before expiration—manually and via standardized automated certificate installation.

Figure 1-1 TLS Server Certificate Management Example Implementation: Logical Architecture


Data Center Secure
TLS Server (Microsoft IIS with HSM)
Root CA (Microsoft ADCS)
Hardware Security Module (SafeNet AT)
Issuing CA (Microsoft ADCS)
Certificate Manager, Log Server and Scanning Tool (Venafi TPP)
Certificate DB (Microsoft SQL Server)
Data Center
TLS Server (Microsoft IIS)
TLS Server (Apache with ACME)
DevOps (Kubernetes and NGINX)
Application Servers (Apache)
DMZ
TLS Server (Microsoft IIS)
TLS Servers (Apache)
Load Balancer (F5 BIG-IP LTM)
TLS Inspection Appliance (Symantec SSLV)
Internet
External CA (DigiCert)
External Users

382

Within the data center secure zone of the logical architecture sit the components that perform TLS server certificate management. These components include internal root and issuing CAs, a certificate manager, a certificate log server, a certificate network scanning tool, a certificate database, and an HSM. For demonstration purposes, a TLS server connected to an HSM is also present in this zone.

The certificate manager can be used in conjunction with the certificate database and the various types of servers in the architecture to demonstrate how to establish and maintain a systematized inventory of certificates (and keys) used on the network. The certificate manager can also continuously monitor TLS certificates (and keys) managed by the inventory system and act upon the status of any certificate (e.g., report a certificate that is expired, about to expire, or does not conform to policy, or it can replace an expired certificate). It can also send expiration reports and notifications to certificate owners and can support disaster recovery by quickly replacing a large number of certificates located throughout the network architecture.

The certificate manager can be used in conjunction with the CAs to enroll and provision certificates (and keys), store attributes with those certificates, and discover the absence of an expected certificate from a machine where it should be installed. The certificate manager can revoke certificates and change the owner associated with that certificate.

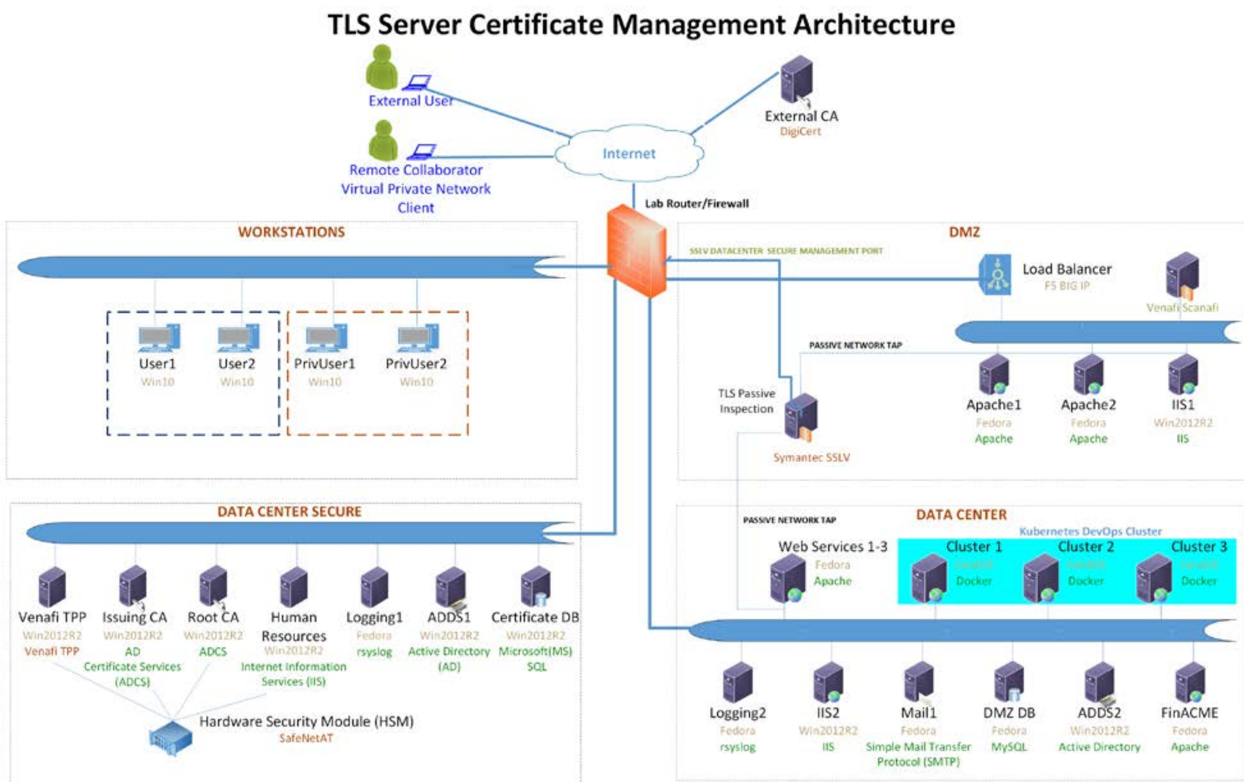
The certificate network scanning tool can discover certificates not being managed by the inventory. The certificate log server can record all certificate and private-key management operations, including certificate creation, installation, and revocation; key pair generation; certificate requests and request approvals; certificate and key copying; and certificate and key replacement.

All components in this portion of the architecture—except for the certificate database—are configured to use the HSM, which can securely generate, store, manage, and process the private key corresponding to the TLS server's certificate. The HSM is capable of storing and protecting the symmetric keys that secure sensitive data in the certificate database, and can generate, store, manage, and process internal CAs' signing keys.

1.3 Build Architecture Summary

Figure 1-2 depicts the physical architecture of the example implementation deployed in the NCCoE laboratory.

Figure 1-2 TLS Server Certificate Management Example Implementation: Laboratory Configuration



The NCCoE laboratory environment provided the following supporting infrastructure for the example implementation:

- firewall-protected connection to the internet where an external CA resides
- Windows 2012 server with remote desktop manager, which acts as a jump box to facilitate installation, deployment, and management of server software for collaborative projects
- segmented laboratory network backbone that models the separation typically existent between subnetworks belonging to different parts of a medium-to-large-scale enterprise—for example, a DMZ, a data center hosting widely used applications and services, a more secure data center hosting critical security infrastructure components, and a segment containing user workstations
- virtual machine and network infrastructure
- Windows 2012 server serving as a Microsoft Active Directory (AD) primary domain controller
- the Windows 2012 server running AD Certificate Services, including
    - an internal Root CA that can issue and self-sign its own TLS certificate

- an internal issuing CA that:
    - issues TLS certificates to servers that request them (issue CAs are subordinate to and certified by the root CA)
    - manages the life cycle of certificates (including request, issuance, enrollment, publication, maintenance, revocation, and expiration)
- Microsoft structured query language (SQL) Server hosting the database of TLS certificates and keys, and corresponding configuration data
- DevOps automation framework, including Kubernetes, Docker, and Jetstack, that demonstrates automated certificate management when performing open-source container orchestration
- Apache, Microsoft IIS, and NGINX servers, which demonstrate various ways of managing TLS server certificates, including remote agentless certificate management, management via the ACME protocol (via the Certbot utility), and management via DevOps
- Apache servers used to demonstrate certificate management on second-tier internal application servers

The following collaborator-supplied components were integrated into the above supporting infrastructure to yield the TLS server certificate management example implementation:

- Venafi Trust Protection Platform (TPP), which maintains the certificate inventory, performs automated TLS server certificate and private-key management, including monitoring, remediation, and rapid replacement of TLS certificates and keys; TLS certificate and key policy enforcement; automated certificate requests and renewals; automated network scanning for TLS certificates; and logging of certificate and private-key management operations
- Symantec SSL Visibility (SSLV), a visibility appliance used to inspect intercepted traffic on encrypted TLS connections
- SafeNet Assured Technologies (SafeNet AT) Luna SA 1700 HSM, used to securely generate, store, manage, and process the cryptographic key pair; also uses it to sign TLS certificates within a hardened, tamper-resistant physical appliance. It is also used to store other keys, such as the database encryption key and the TLS certificate keys for the key manager component (Venafi TPP) and the CAs
- DigiCert external CA, which issues and renews TLS certificates
- F5 Networks BIG-IP Local Traffic Manager load balancer, which acts as a TLS proxy and distributes received traffic across a number of other TLS servers

The remainder of this volume describes in detail the installation, configuration, and integration of the above supporting infrastructure and collaborator components.

1.4 Typographic Conventions

The following table presents typographic conventions used in this volume.

Typeface/Symbol	Meaning	Example
Italics	file names and path names; references to documents that are not hyperlinks; new terms; and placeholders	For detailed definitions of terms, see the *NCCoE Glossary.*
Bold	names of menus, options, command buttons, and fields	Choose **File > Edit.**
`Monospace`	command-line input, onscreen computer output, sample code examples, and status codes	`mkdir`
`Monospace Bold`	command-line user input contrasted with computer output	**`service sshd start`**
blue text	link to other parts of the document, a web URL, or an email address	All publications from NIST's NCCoE are available at https://www.nccoe.nist.gov.

1.5 Supporting Infrastructure

This section is the first in a series of how-to guidance offered in this guide. It contains step-by-step instructions and points to specific, well-known, and trusted information for installing, configuring, and securely maintaining the supporting infrastructure components outlined in previous sections of this document.

All supporting infrastructure components in the following how-to subsections are high-level examples of services and functions that may reside on any network. For example, the Microsoft suite of AD, CA services, domain name server (DNS), web, and database services would typically reside on most organizational networks. Each section follows the other in building the prerequisites. This section on supporting infrastructure is the basis for the subsequent how-to sections on collaborator capabilities.

The lab backbone is the fundamental component of the architecture and forms the basis to develop the implementers' understanding of the simulated build experience. Guidance is provided for each operating system (OS) installation, with specific instructions on the necessary security and system configurations. Finally, specific ancillary services, installation and security configurations for database services, web services, etc. are provided.

1.5.1 Lab Backbone

The NCCoE has a specific implementation of its supporting lab network infrastructure or lab backbone. Although implementors using this document may possess some or most of the components in the TLS lab backbone, they may encounter slight but significant differences in their lab build. These differences are attributed to how we configured our lab backbone to suit the needs of the TLS lab and the larger multitiered lab community within the NCCoE.

The components and configuration approaches listed below may help clarify what basic capabilities are needed at a minimum to simulate the TLS lab infrastructure backbone.

- network topology–designed to provide strict separation of system and workstation duties:
    - Data Center Secure Network–provides physical and logically secure separation of critical security services from nonprivileged or privileged users without specific security responsibilities
    - Data Center Network–provides less privileged users with access to security maintenance services that do not require special access to critical security management services
    - Workstations Network–provides secure, controlled, and monitored access to nonprivileged authorized users to perform organizational business
    - DMZ–provides secure separation and mitigation of risk to the rest of the critical network services from public access to public-facing services
- multiple virtual local area networks (VLANs) and separate subnets–customized naming convention for VLAN names and subnets can be used, or follow the TLS lab approach below:
    - VLAN 2198 services the Data Center Secure Network 192.168.1.0/24
    - VLAN 2199 services the Data Center Network 192.168.3.0/24
    - VLAN 2200 services the Workstations Network 192.168.2.0/24
    - VLAN 2197 services the DMZ Network 192.168.4.0/24
    - VLAN 2196 services connections between the F5 load balancer and lab firewall 192.168.5.0/24
    - VLAN 2202 services wide area network connections between the internet and the firewall; the address used here should mirror whatever is currently used for what the internet provider gave in a subnet address
- One or more managed layer three switches must be capable of:

- traffic separation for six VLANs with multiple devices on each VLAN (see the architecture diagram for more)
- switched port analyzer (SPAN) or port mirroring functions
- VLAN trunk ports when using multiple switches

- One or more manageable advanced firewalls:
  - must be capable of accepting at least six Ethernet port connections for all VLANs if using one firewall
  - must be capable of network address translation (NAT) (port forwarding, hide NAT, and static NAT)
  - should at least be stateful
  - should support deep packet inspection for every possible subnet where feasible and financially practical

1.5.2 Supporting Infrastructure Operating Systems

1.5.2.1 Microsoft Windows

Microsoft Windows and Windows Server are within a group of OSs designed by Microsoft to efficiently manage enterprise needs for data storage, applications, networking, and communications. In addition to the standard OSs used, additional ancillary Microsoft services were installed. These are native components of the OS and critical to the TLS lab design. Guidance on configuration of these ancillary services will be discussed later in this document in the Supporting Infrastructure Component Services section.

- AD Services
- DNS Services
- CA Services

1.5.2.1.1 Microsoft Windows and Server Prerequisites

Both Microsoft Windows servers and workstations have minimal hardware prerequisites, listed directly below this paragraph. In addition, TLS lab host configuration information is provided in Table 1-1 and Table 1-2 below. While it is not imperative that an implementer uses the TLS lab host naming convention and internet protocol (IP) addressing schemes, the tables below may prove useful with informing an organization of the servers and workstations needed should there be customizations to the TLS lab approach.

While the hardware requirements listed below represent the minimum, most business applications of this effort may have higher but differing requirements. All the applications in this TLS build will greatly benefit from adding more than the minimum resources that Microsoft requires, as shown below, in a production environment.

Microsoft's Minimum Hardware Requirements:

- Microsoft Windows Servers 2012
    - 1 gigahertz (GHz) 64-bit processor
    - 512 megabyte (MB) random access memory (RAM)
    - 32 gigabytes (GB) disk space
- Microsoft Windows Workstations 2010
    - 1 GHz 64-bit processor
    - 2 GB RAM
    - 20 GB disk space

1.5.2.1.2 Microsoft Windows Server 2012 Installation

- For instructions regarding downloading the Microsoft Windows Server 2012, refer to the download and deployment guidance at: https://www.microsoft.com/en-us/evalcenter/evaluate-windows-server-2012-r2.

Given that AD and domain services are critical to the adds1 and adds2 installation process, refer to the **Microsoft Active Directory and Domain Services Installation and Configuration** section, 1.5.3.1, of this document for full instructions after initial basic installation of the OS.

Please use the table below to name and assign IP addresses to all Microsoft Windows Servers used in the TLS lab build. The Windows Server version used in most cases is Windows 2012 version R2.

Table 1-1 Naming and Addressing Information for all Microsoft Windows Servers

Host Name	IP Address	Subnet	Gateway	Software Selection
iis1.ext-nccoe.org	192.168.4.4	255.255.255.0	192.168.4.1	Win2012 R2
adds1.int-nccoe.org	192.168.1.6	255.255.255.0	192.168.1.1	Win2012 R2
HSMrootca.int-nccoe.org	192.168.1.10	255.255.255.0	192.168.1.1	Win2012 R2
BaseSubCA.int-nccoe.org	192.168.1.41	255.255.255.0	192.168.1.1	Win2012 R2
HRhsm	192.168.1.16	255.255.255.0	192.168.1.1	Win2012 R2
Venafi1	192.168.1.81	255.255.255.0	192.168.1.1	Win2012 R2
VTPPTrustDB	192.168.1.89	255.255.255.0	192.168.1.1	Win2012 R2
iis2.int-nccoe.org	192.168.3.5	255.255.255.0	192.168.3.1	Win2012 R2

Host Name	IP Address	Subnet	Gateway	Software Selection
adds2.int-nccoe.org	192.168.3.7	255.255.255.0	192.168.3.1	Win2012 R2
dmzdc.ext-nccoe.org	192.168.3.8	255.255.255.0	192.168.3.1	Win2012 R2

1.5.2.1.3 Microsoft Windows 10 Workstations Installation

- For instructions regarding download of the Microsoft Windows 10 workstation used in this TLS lab build, refer to the guidance at https://www.microsoft.com/en-us/software-download/windows10.

Please use the table below to name and assign IP addresses to all Microsoft Windows 10 workstations used in the TLS lab build. The Windows 10 version used in most cases is Windows 10 Pro.

Table 1-2 Naming and Addressing Information for all Microsoft Windows 10 Workstations

Host Name	IP Address	Subnet	Gateway	Software Selection
win10-1.int-nccoe.org	192.168.2.11	255.255.255.0	192.168.2.1	Win10_Pro
win10-2.int-nccoe.org	192.168.2.2	255.255.255.0	192.168.2.1	Win10_Pro
privuser1.int-nccoe.org	192.168.2.3	255.255.255.0	192.168.2.1	Win10_Pro
privuser2.int-nccoe.org	192.168.2.4	255.255.255.0	192.168.2.1	Win10_Pro

1.5.2.2 Linux

Linux is a family of free and open-source OSs based on the Linux kernel, an OS kernel first released on September 17, 1991, by Linus Torvalds. Fedora Server is a Red Hat Corporation-supported, short life-cycle, and fully community-supported server OS. Fedora enables system administrators of any skill to freely (in most cases) make use of the very latest technologies available in the open-source community.

The CentOS Linux distribution is no different in its ability to allow mostly free use of world-class security and general IT capabilities. CentOS is a manageable and reproducible platform derived from the sources of Red Hat Enterprise Linux (RHEL) by an open-source community of volunteers.

1.5.2.2.1 Linux Prerequisites

Table 1-3 and Table 1-4 include the host names and IPs used in the TLS lab for all Linux machines. The recommended minimum hardware requirements for the default installations of Fedora and CentOS have been noted below. An organization’s requirements may differ. However, it is highly recommended that the maximum optimal configuration (in accordance with the organization’s available resources) for each system be applied, as all the applications used in this TLS lab build will benefit from more than the minimum resources in a production environment.

- 1 GHz or faster processor
- 1 GB system memory
- 10 GB unallocated drive space
- 1 VMXNET 3 network adapter

1.5.2.2.2 Fedora and CentOS Installation

The OS installation process for the TLS lab Linux machines did not deviate from the standard installation instructions that exist for each Linux distributor. The links below provide standard guidance for the Fedora and CentOS installations.

When running through the installation process, in some cases, a standard Fedora installation for software selection will not suffice. Should this occur, use Table 1-3. If the Software Selection column includes Fedora Server/Basic Web Server, select Fedora Server for Base Environment, then select Basic Web Server installation for add-ons, and when prompted, select software packages during the installation.

The CentOS Software Selection column includes Basic Web Server—select this as the software package to install when prompted during the installation process for CentOS.

- https://docs.fedoraproject.org/en-US/fedora/f28/install-guide/
- https://docs.centos.org/en-US/centos/install-guide/

Please use Table 1-3 for IP, host name, and other installation-specific options for all Fedora-based systems in the TLS lab build.

Table 1-3 Naming and Addressing Information for All Fedora-Based Systems

Host Name	IP Address	Subnet	Gateway	Software Selection
syslog2.int-nccoe.org	192.168.3.12	255.255.255.0	192.168.3.1	Fedora Server
finacme.int-nccoe.org	192.168.3.61	255.255.255.0	192.168.3.1	Fedora Server/ Basic Web Server
mail1.int-nccoe.org	192.168.3.25	255.255.255.0	192.168.3.1	Fedora Server
dmzdb.ext-nccoe.org	192.168.3.6	255.255.255.0	192.168.3.1	Fedora Server
syslog1.int-nccoe.org	192.168.1.12	255.255.255.0	192.168.1.1	Fedora Server
apache1.ext-ncccoe.org	192.168.4.2	255.255.255.0	192.168.4.1	Fedora Server/ Basic Web Server
apache2.ext-nccoe.org	192.168.4.3	255.255.255.0	192.168.4.1	Fedora Server/ Basic Web Server

Host Name	IP Address	Subnet	Gateway	Software Selection
ws1.int-nccoe.org	192.168.3.87	255.255.255.0	192.168.3.1	Fedora Server/ Basic Web Server
ws2.int-nccoe.org	192.168.3.88	255.255.255.0	192.168.3.1	Fedora Server/ Basic Web Server
ws3.int-nccoe.org	192.168.3.89	255.255.255.0	192.168.3.1	Fedora Server/ Basic Web Server

Please use Table 1-4 for IP, host name, and other installation-specific options for all CentOS servers used in the TLS lab build.

Table 1-4 Naming and Addressing Information for All CentOS Servers

Host Name	IP Address	Netmask	Gateway	Software Selection
scanafi.ext-nccoe.org	192.168.4.107	255.255.255.0	192.168.4.1	Infrastructure Server
cluster1.int-nccoe.org	192.168.3.103	255.255.255.0	192.168.3.1	Basic Web Server
cluster2.int-nccoe.org	192.168.3.104	255.255.255.0	192.168.3.1	Basic Web Server
cluster3.int-nccoe.org	192.168.3.105	255.255.255.0	192.168.3.1	Basic Web Server

1.5.3 Supporting Infrastructure Component Services

1.5.3.1 Microsoft Active Directory and Domain Services Installation and Configuration

Active Directory Services (ADS) and DNS work together to store directory data and make those resources available to administrators and users. For example, ADS stores information about user accounts such as names and passwords. Security is integrated with ADS through log-on authentication and enforced access control for user, file, directory, and other system objects in the directory of services. Administrators are able to manage directory data and organization roles across the enterprise. They can assign permissions to users, which allows users to access resources anywhere on the network. ADS authenticates and authorizes all users and computers in a Windows domain network. ADS works in conjunction with Group Policies Objects (GPOs) in assigning and enforcing security policies for all computers.

A DNS is a protocol for how computers translate domain names. It manages a database used to resolve domain names to IP addresses, allowing computers to identify each other on the network. DNS is the primary locator service for AD. ADS is highly dependent on the DNS in most cases, and as a result, most implementations—including the TLS lab—opt to install the DNS service on the same server as the ADS.

1.5.3.1.1 ADS and DNS Prerequisites

Below are the minimum recommended tools, services, and configurations needed to install ADS and DNS.

- The adds1 and adds2 hosts should be built with the Windows Server 20012 OS installed. As described in Section 1.5.2.1.2 of this document, there are two ADS and DNS servers. The TLS lab ADS and DNS server names used are adds1.int-nccoe.org and adds2.int-nccoe.org. (Note: The DNS server may be run locally on the same Active Directory Domain Services [ADDS] server.)
- local network configurations–all of the local network VLANs, IP addresses, and proper routes
- familiarity with Server Manager

Server Manager is a Windows Server management console that allows administrators to install, configure, and manage server roles and features. Administrators can manage local and remote servers without having physical access to them. The ADS and DNS installation process is integrated with Server Manager, which can be used when installing other server roles.

1.5.3.2 ADS and DNS Installation

For instructions on deploying ADS and DNS on a Windows 2012 server, refer to the guidance at one of the links below:

- **Graphical User Interface (GUI)-Based Installation:** https://docs.microsoft.com/en-us/windows-server/identity/ad-ds/deploy/ad-ds-installation-and-removal-wizard-page-descriptions
- **Command Line-Based Installation:** https://docs.microsoft.com/en-us/windows-server/identity/ad-ds/deploy/install-active-directory-domain-services--level-100-

1.5.3.3 Certificate Authority Services

In an organization where public key infrastructure (PKI) has been implemented, a CA is responsible for validating the identity of users and computers. The CA assigns a trusted credential for use in authenticating user and system identities, by issuing a digitally signed and trusted certificate. The CA can also assist in managing revocation and renewal of its signed certificates.

The first CA built and implemented in a PKI environment is often referred to as the root CA. As the originator and root of trust, the root CA authorizes all subsequent CAs, called subordinates or issuing CAs. Subordinate CAs can also designate their own subsidiaries as defined by the root CA, which results in a certificate hierarchy. The metadata supplied in all certificates issued to CAs lower in the hierarchy from the root CA contain a trace path back to the root.

A compromised root CA will cripple any organization that depends on the integrity of its issued PKI certificates, even in lightweight transactions. With full control or significant unauthorized access to the root CA, a malicious actor may fully infiltrate any transaction that relies on the integrity of the trust chain where that root CA presides as the anchor. It is recommended all organizations—size notwithstanding—implement an enterprise stand-alone offline root CA and separate issuing subordinate

CA(s) topology wherever possible. Doing so mitigates many of the risks associated with compromised root CAs.

The TLS lab followed Microsoft's guidance to develop a highly secure offline stand-alone root CA coupled with an enterprise online issuing CA. The following CA installation and configuration how-to guidance aligns with that goal.

1.5.3.3.1 CA Prerequisites

The prerequisite steps to configure the CA(s) include:

- Build HSMrootca.int-nccoe.org and BaseSubCA.int-nccoe.org in accordance with the OS installation and configuration instructions in Section 1.5.2.1.2.
- Join BaseSubCA.int-nccoe.org to the already created int-nccoe.org domain.
- HSMrootca.int-nccoe.org and BaseSubCA.int-nccoe.org should have network connections to all the TLS lab subnets needed for CA certificate issuance.

1.5.3.3.2 Installation of Offline Root and Issuing CA

In this implementation scenario, the offline root CA is built, configured, and established as the root of the trust chain. The root CA is then configured to securely sign and issue certificates for all of its subordinates. Afterward, it is taken completely offline. Being taken offline includes complete power-down and highly secures physical storage of the root CA device (specifically the hard drive if possible).

Installation of the root CA through the Server Manager console can be done by installing Active Directory Certificate Services (ADCS). ADCS is used to create CAs and configure their role to issue and manage certificates. For instructions on installing ADCS on the root CA and issuing CA server, refer to the steps below:

1. In the **Server Manager,** select **Manage** > click on **Add Roles and Features.**
2. Follow the Add Roles and Features wizard > in **Select Installation Types,** select **Role-Based or feature installation.**
3. In **Select destination server,** confirm **Select a server from the server pool** is selected > select your local computer.
4. In **Select server roles** > under **Roles,** select **Active Directory Certificate Services >** click **Add Features.**
5. In **Select features** > click **Next.**
6. In **Active Directory Certificate Services** > click **Next.**
7. In **Select role services** > in **Roles,** select **Certification Authority.**
8. In **Confirm installation records** > click **Install.**
9. When installation is complete, click **Close.**

1.5.3.3.3 Offline Root CA Configuration

After installing ADCS, refer to the steps below to configure and specify cryptographic options for the root CA:

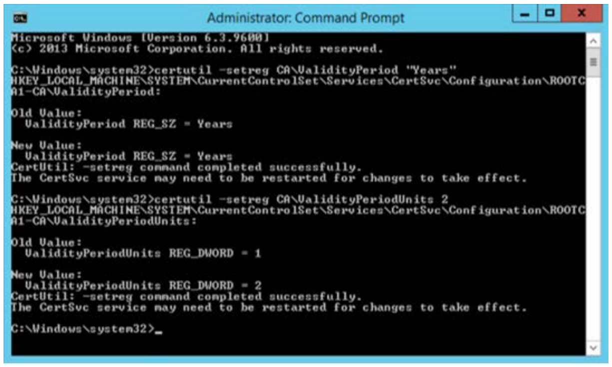
1. Run **Post-deployment Configuration** wizard > click on **Configure Active Directory Services** link.
2. In **Credentials,** read the credentials information. If needed, provide administrator credentials.
3. In **Role Services** > select **Certification Authority.**
4. In **Setup Type** > select **Standalone CA.**
5. In **CA Type** > select **Root CA.**
6. In **Private Key** > select **Create a new private key** to specify type of private key.
7. In **Cryptography for CA**:
    - Select a cryptographic provider: **RSA#SafeNet Key Storage Provider.**
    - Key Length = **2048**
    - Select the hash algorithm for signing certificates issued by this CA: **SHA256.**
8. In **CA Name** > specify the name of CA > **RootCA.**
9. For **Validity Period** > select **2 Years.**
10. Specify the database location > *C:\Window\system32\CertLog.*
11. Review the CA configuration and click **Configure.**
12. Click **Close** when the confirmation message appears.

To configure the CRL Distribution Point (CDP) and Authority Information Access (AIA) extensions on the root CA, follow the steps below:

1. In **Server Manager,** go to **Tools** > select **Certification Authority.**
2. Right-click **RootCA** > click **Properties.**
3. Click the **Extensions** tab. Ensure **Select Extension** is set to **CDP.**
4. In the **Specify locations from which users can obtain a certificate revocation list (CRL),** do the following:
    a. Select the entry ***file://<ServerDNSName>/CertEnroll/<CaName><CRLNameSuffix><DeltaCRLAllowed>.crl*** and then click **Remove.** In **Confirm removal,** click **Yes.**
    b. Select the entry ***http://<ServerDNSName>/CertEnroll/<CaName><CRLNameSuffix><DeltaCRLAllowed>.crl*** and then click **Remove.** In **Confirm removal**, click **Yes.**
5. In **Specify locations from which users can obtain a certificate revocation list (CRL),** click **Add.**
6. In **Add Location,** in **Location,** type *http://BaseSubCA/CertEnroll/<CaName><CRLNameSuffix><DeltaCRLAllowed>.crl* and then click **OK.** This returns to the CA properties dialogue box.
7. On the **Extensions tab,** select the following checkboxes:
    - **Include in CRLs. Clients use this to find the Delta CRL locations.**
    - **Include in the CDP extension of issued certificates.**

8. In **Specify locations from which users can obtain a certificate revocation list (CRL),** select the entry that starts with **ldap://CN=CATruncatedName>,CRLNameSuffix>,CN=<ServerShortName>**.
9. On the **Extensions** tab, select the following checkbox:
    - **Include in all CRLs. Specifies where to publish in the Active Directory when publishing manually.**
    - In **Specify locations, users can obtain a certificate revocation list (CRL).** Select the entry **C:\\Windows\system32\CertSrv\CertEnroll\<CaName><CRLNameSuffix><DeltaCRLAllowed>.crl.**
10. On the **Extensions** tab, select the following checkboxes:
    - **Publish CRLs to this location.**
    - **Publish Delta CRLs to this location.**
11. Change **Select extension** to **Authority Information Access (AIA).**
12. In the **Specify locations, users can obtain a certificate revocation list (CRL)** do the following:
    a. Select the entry ***http://<ServerDNSName>/CertEnroll/<ServerDNSName>_<CaName><CertificateName>.crt*** and then click **Remove.** In **Confirm removal,** click **Yes.**
    b. Select the entry ***file://<ServerDNSName>/CertEnroll/<ServerDNSName>_<CaName><CertificateName>.crt*** and then click **Remove.** In **Confirm removal,** click **Yes.**
13. In **Specify locations, users can obtain a CRL,** click **Add.**
14. In **Add Location,** in **Location**, type ***http://BaseSubCA/CertEnroll/<ServerDNSName>_<CaName><CertificateName>.crt*** and then click **OK.** This returns to the CA properties dialogue box.
15. On the **Extensions** tab, select the following checkbox:
    - **Include in the AIA of issued certificates.**
16. In **Specify locations from which users can obtain a certificate revocation list (CRL),** select the entry that starts with **ldap://CN=CATruncatedName>,CN=AIA,CN=PublicKeyServices.**
17. On the **Extensions** tab, select the following checkbox:
    - **Include in the AIA extension of issued certificates.**
18. In **Specify locations, users can obtain a certificate revocation list CRL.** Select the entry **C:\\Windows\system32\CertSrv\CertEnroll\<ServerDNSName>_<CaName><CertificateName>.crt.**
19. On the **Extensions** tab, ensure **AIA extension of issued certificates** is not selected.
20. When prompted to restart Active Directory Certificate Services, click **No.** Restart that service later.
21. Go back to **RootCA** and expand folders to right-click on **Revoked Certificates >** select **All Tasks >** click **Publish.**
22. When prompted to Publish CRL, select **New CRL >** click **OK.**
23. To configure the Registry Settings, run cmd as an administrator and type the following commands:

```
certutil -setreg CA\ValidityPeriod “Years”
certutil –setreg CA\ValidityPeriodUnits 2
```




```
certutil -setreg CA\DSConfigDN “CN=Configuration,DC=int-nccoe,DC=org”
```

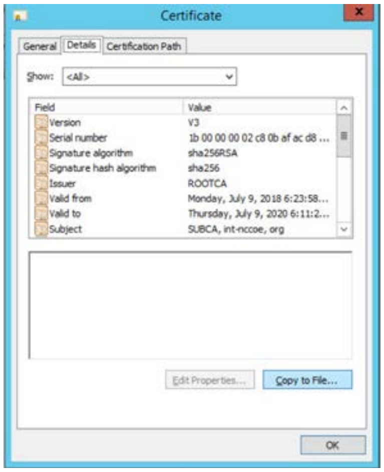

```
cerutil -setreg CA\DSDomainDN “DC=int-nccoe,DC=org”
```

24. For it to accept the new values, restart services > go to **Administrative Tools >** double-click **Certification Authority.**
25. Select the **RootCA** > right-click to select **All Tasks** > click **Start Service.**
26. Go back to **RootCA** to expand folders > right-click on **Revoked Certificates >** select **All Tasks** > click **Publish** to publish revoked certificates.

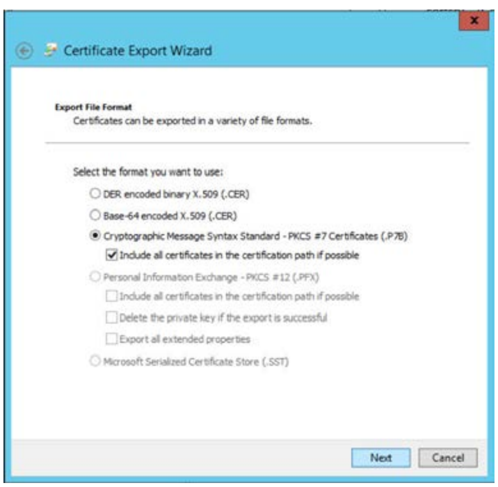
1.5.3.3.4 Enterprise Subordinate/Issuing CA Configuration

After installing ADCS, follow the steps below to configure and specify cryptographic options for the issuing CA:

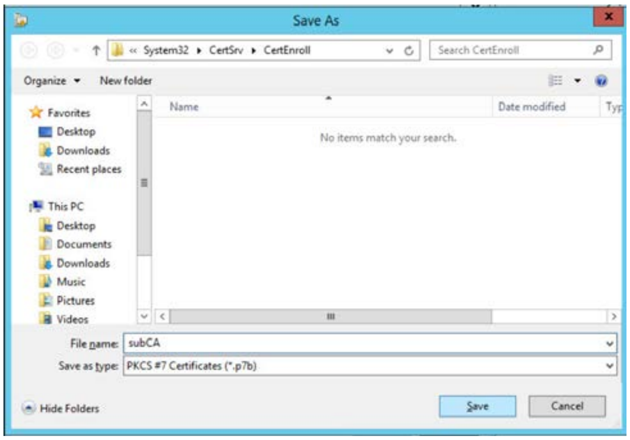
1. Run **Post-deployment Configuration** wizard > click on **Configure Active Directory Services** link.
2. In **Credentials,** read the credentials information. If needed, provide administrator credentials.
3. In **Role Services** > select **Certification Authority.**
4. In **Setup Type** > select **Enterprise CA.**
5. In **CA Type** > select **Subordinate CA.**
6. In **Private Key** > select **Create a new private key** to specify type of private key.
7. In **Cryptography for CA:**
    - Select a cryptographic provider: **RSA#SafeNet Key Storage Provider.**
    - Key Length = **2048**
    - Select the hash algorithm for signing certificates issued by this CA: **SHA256.**
8. In **CA Name** > specify the name of the CA > **BaseSubCA.**
9. In **Certificate Request >** select **Save a certificate request to file on the target machine** > specify folder location > *C:\BaseSubCA.int-nccoe.org_int-nccoe-BASESUBCA-CA.req.*
10. In **CA Database** > specify the folder location for the certification database > **C:\Windows\system32\CertLog.**
11. In **Confirmation >** confirm configurations and select **Configure** > click **Close.**
12. Copy the BaseSubCA request file from the BaseSubCA server to the RootCA server at **C:\Windows\System32\CertServ\CertEnroll.**
13. Copy ***rootCA.crl*** and ***rootCA.crt*** to the BaseSubCA server at **C:\Windows\System32\CertServ\CertEnroll.**
14. To issue a certificate to the BaseSubCA server from the RootCA server, go to **Administrative Tools >** double-click **Certification Authority.**
15. Select **BaseSubCA >** right-click to select **All Tasks >** click **Submit new request.**
16. Select and open the request file in the dialogue box.
17. Go back to the **Certification Authority >** select **BaseSubCA** and expand folders > click on **Pending Requests.**
18. Right-click the pending certificate > right-click to select **All Tasks >** click **Issue.**
19. Go to **Issued Certificates** to view the issued certificate.
20. Double-click on the issued certificate.
21. Go to the **Details** tab > click **Copy to File.**



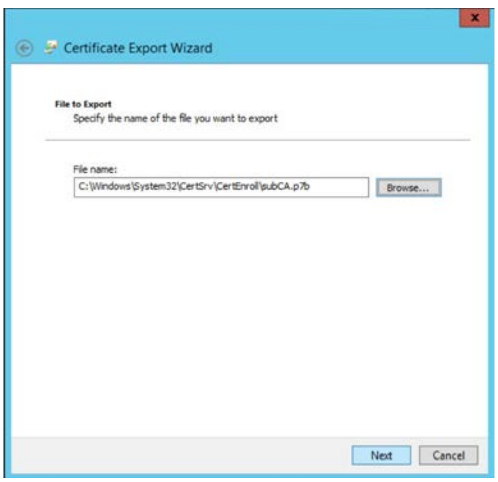
22. Follow the Certificate Export wizard and select the desired format:



23. Save the file as **subCA >** file type is **PKCS #7 Certificates (*.p7b).**

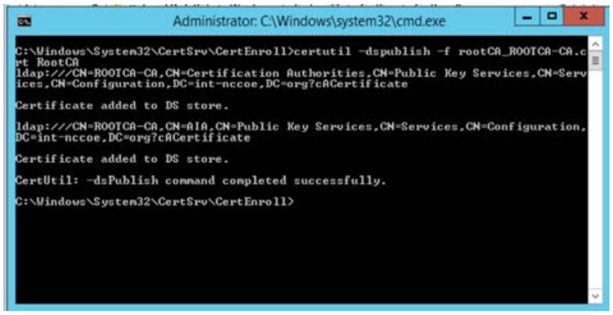


24. Specify the file name to export:

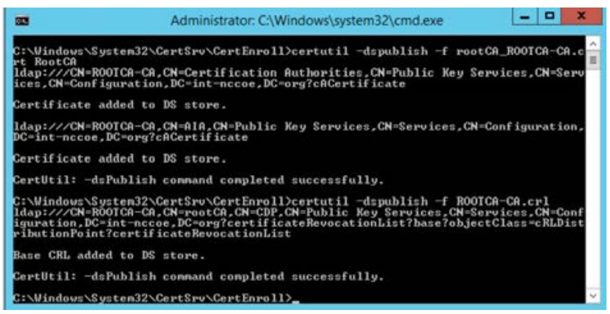


25. Complete the Certificate Export Wizard by confirming settings > click **Finish.**
26. In **Export was successful** > click **OK.**
27. Copy **subCA.p7b** from the RootCA server at **C:\WindowSystem32\CerServ\CertEnroll** to the BaseSubCA server at **C:\WindowSystem32\CerServ\CertEnroll.**
28. On the BaseSubCA server > shift right-click > open the command prompt.
29. Publish the CA Root certificate into Directory Services with the following command:

```
certutil -dspublish -f (tab to rootCA.crt file) RootCA
```



30. To publish the crl file, type the following command:
    `certutil -dspublish -f (tab to .crl file)`



31. Set the **Domain Policy** to make the RootCA trusted by all domain computers.
32. Install the certificate in the subCA server > go to **Administrative Tools** > double-click **Certification Authority.**
33. Select the CA > right-click to select **All Tasks >** click **Install CA Certificate.**
34. Select the *.p7b* file to complete the CA installation.
35. A warning message will be received that the revocation server is offline > click **OK** to ignore the message.
36. Power down the RootCA server.
37. Go to **Administrative Tools** > right-click the CA > select **All Tasks** > click **Start Service** to start services.
38. Install *.crt* files on the Default Domain Policy.
39. Go to the domain controller (DC).
40. Go to **Administrative Tools** > open **Group Policy Management** console.
41. Go to the organization's domain > right-click the **Default Domain Policy** folder > select **Edit.**
42. Navigate to **Computer Configuration,** go to **Policies > Window Settings > Security Settings > Public Key Policies** > right-click **Intermediate Certification Authorities** > select **Import.**
43. Follow the **Certificate Import Wizard** > click **Next.**
44. Select the ***subCA.crt*** file to import > click **Next** to import file.
45. Confirm details > click **Finish.**
46. A dialogue box will pop up to confirm **The import was successful.**
47. Go to **Trusted Root Certification Authority** folder and right-click> select **Import.**

48. Follow the **Certificate Import Wizard** > click **Next.**
49. Select the ***rootCA.crt*** file to import > click **Next** to import file.
50. Confirm details > click **Finish.**
51. A dialogue box will appear to confirm **The import was successful.**

1.5.4 Database Services

1.5.4.1 Microsoft SQL Database Services

Microsoft SQL (MSQL) Server is a relational database management system developed by Microsoft. As a database server and a software product, its primary function is to store and retrieve data as requested by other software applications. MSQL can operate on the same or another computer across a network.

1.5.4.1.1 Prerequisites for MSQL Database Services

The information below is Microsoft's recommended minimum for default installation of MSQL. An organization's requirements may differ. However, all applications can benefit from more than the minimum resources in a production environment.

- 1.4 GHz 64-bit processor
- 1 GB RAM
- 6 GB disk space
- administration privileges (local installations must run Setup as an administrator)

One MSQL database was used for the TLS lab build to support the Venafi TPP server. This guide installs only the basic MSQL application on a server. This prepares the specific configurations that are discussed in the Venafi TPP How -To guidance section. As a prerequisite, see the OS installation instructions in Section 1.5.2.1.2 to build the VTPPTrustDB.int-nccoe.org server.

1.5.4.1.2 Installation of MSQL Database Services

To install MSQL on a Windows 2016 Server, follow the Microsoft steps in the link below:

- Download here: https://www.microsoft.com/en-us/sql-server/sql-server-downloads?&OCID=AID739534_SEM_at7DarBF&MarinID=sat7DarBF_340829462634_microsoft%20sql%20download_e_c__68045082145_kwd-343189224165_
- Install and configure here: https://docs.microsoft.com/en-us/sql/database-engine/install-windows/install-sql-server-from-the-installation-wizard-setup?view=sql-server-2017
- Install MSQL as a stand-alone server.
- Specify the Database Engineer Configuration in step 15 by selecting SQL Server Administrators.

1.5.4.2 MariaDB Database Services

The original inventors of MySQL developed the MariaDB server, which is highly compatible with MySQL. This allows a drop-in replacement capability with library binary parity and exact matching with MySQL's application programming interfaces and commands.

Like MySQL, the open-source version of MariaDB can scale and performs as well as most enterprise database servers. The TLS lab uses the MariaDB to serve its public-facing (DMZ) web-based TLS services described in this document.

1.5.4.2.1 Prerequisites for MariaDB Database Services

The host named dmzdb.ext-nccoe.org should have already been set up within the Fedora OS how-to guidance of Section 1.5.2.2.2. Complete this setup prior to installing the MariaDB server.

1.5.4.2.2 Installation of MariaDB Database Services

- To download and install MariaDB, please refer to the fedoraproject.org guidance at https://fedoraproject.org/wiki/MariaDB

1.5.4.2.3 Configuration of MariaDB Database Services

MariaDB is used to serve dynamic web content with the Drupal application. All three web servers used in the DMZ must be configured via Drupal to point to one database. As a result, the database must be configured to accept connections from the Drupal web servers. MariaDB can be configured by using the Fedora Linux command line. To start, first set up a secure password for the root and any other administrative accounts (see the MariaDB setup instructions on how to specify other accounts). Log in to the dmzdb.int-nccoe.org by using the local command line shell or secure remote administration client (ssh, putty, openssh). Once logged into the system, use the following command to launch MariaDB from the Fedora Linux:

```
[root@dmzdb ~]# mysql -p
```

Note: Although the root account is displayed here as the login account, configuring MariaDB with the root user in a production environment is not recommended.

Configure the database to allow remote connections from either the IP addresses or host names used in the TLS lab. If the IP addresses and host names were customized (apache1: 192.168.4.2, apache2: 192.168.4.3, iis1: 192.168.4.4), please double-check and change the IP addresses in the database by using the commands below. If custom host names were used in place of the IP addresses, the database DNS or host resolution is set to properly resolve to the right IP addresses.

```
[root@dmzdb ~]# mysql -p

Enter password:

Welcome to the MariaDB monitor. Commands end with ; or \g.
Your MariaDB connection id is 1012018
```

```
Server version: 10.2.16-MariaDB MariaDB Server

Copyright (c) 2000, 2018, Oracle, MariaDB Corporation Ab and others.

Type 'help;' or '\h' for help. Type '\c' to clear the current input statement.

MariaDB [(none)]>  create database EXT_NCCOE_DB;

MariaDB [(none)]>  grant all privileges on EXT_NCCOE_DB.* to
'EXTADMIN'@'192.168.4.2'  IDENTIFIED BY 'YOUR PASSWORD';

MariaDB [(none)]>  grant all privileges on EXT_NCCOE_DB.* to
'EXTADMIN'@'192.168.4.3'  IDENTIFIED BY 'YOUR PASSWORD';

MariaDB [(none)]>  grant all privileges on EXT_NCCOE_DB.* to
'EXTADMIN'@'192.168.4.4'  IDENTIFIED BY 'YOUR PASSWORD';

MariaDB [(none)]> quit;
```

Add rules to the local Linux firewall to allow database traffic inbound. Please use the following commands to allow database traffic to inbound ports on the MariaDB server:

- Type the following command to allow database connections to Apache:

```
iptables-I INPUT -p tcp –dport 3306 -mstate --state related, ESTABLISHED, new -
j ACCEPT
```

1.5.5 TLS Web Services

1.5.5.1 Microsoft Internet Information Services

The web server (IIS) role in Windows Server 2012 provides a means for hosting websites, services, and applications. IIS information can be shared with users on the internet, an intranet, or an extranet. IIS is a unified web platform that integrates IIS, ASP.NET, File Transfer Protocol services, Personal Home Page (PHP), and Windows Communication Foundation.

The TLS lab utilized the IIS server as a public-facing member of a load balance web cluster for public-facing internet services. It was also used as an intranet server to simulate an employee web-based knowledge management system that is internal to an organization.

1.5.5.1.1 IIS Prerequisites

Complete the following prerequisite steps prior to installing and configuring IIS:

- Server iis2.int-nccoe.org should ideally be a member of the domain for more streamlined TLS certificate management.
- The IIS administrator must have Request Certificates permission on the issuing CA.
- The iis1.int-nccoe.org and iss2.int-nccoe.org servers should be set up per Section 1.5.2.1.2.
- Server iis1.int-nccoe.org should be used for the public-facing web-based cluster.

- Server iis2.int-nccoe.org should be used as the internal intranet server.

1.5.5.2 IIS Installation

IIS is the topic of this section, however, the PHP is a key component of the IIS installation for the TLS lab implementation of the iis1.int-nccoe.org internet-facing server. PHP is a script language and interpreter and a server-side language that assists IIS and Drupal in serving dynamic web content.

Please follow the instructions in the link below to install IIS and PHP. The iis2.int-nccoe.org server can be set up without PHP installed. Please follow the same instructions below for the iis2 server—skip the PHP part of the installation process.

- https://docs.microsoft.com/en-us/iis/application-frameworks/scenario-build-a-php-website-on-iis/configuring-step-1-install-iis-and-php

Windows 2012 Server provides several methods for enrolling certificates: two of these are the Certificate Enrollment Policy (CEP) and Certificate Enrollment Service (CES). The CEP web service enables users and computers to obtain certificate enrollment policy information. This information includes what types of certificates can be requested and what CAs can issue them. CES provides another web service that allows users and computers to perform certificate enrollment by using the hypertext transfer protocol secure (https). To separate traffic, the CES can be installed on a computer that is separate from the CA. Together with the CEP web service, CES enables policy-based certificate enrollment when the client computer is not a member of a domain or when a domain member is not connected to the domain. CEP/CES also enables cross-forest, policy-based certificate enrollment.

For the purpose of the lab, the IIS configuration option selected for authentication type for the CES is **Windows integrated authentication.** This option provides Kerberos authentication for devices connected to the internal network and joined to a domain. The service account selected is the **Use the built-in application pool identity.**

To configure the SSL protocol to encrypt network traffic, obtain a certificate for IIS, and configure https on the default website, please refer to the link below.

- https://social.technet.microsoft.com/wiki/contents/articles/12485.configure-ssltls-on-a-web-site-in-the-domain-with-an-enterprise-ca.aspx

1.5.5.3 Apache Web Services

The Apache HTTP Server is a free and open-source cross-platform web server software, released under the terms of Apache License 2.0. Apache is developed and maintained by an open community of developers under the Apache Software Foundation.

1.5.5.3.1 Apache Web Services Prerequisites

The Apache web server was used extensively throughout the TLS lab architecture to demonstrate the various means of automated and manual management of TLS certificates. The following servers should be built in accordance with the instructions in Section 1.5.2.2.2.

- *apache1.ext-ncccoe.org*
- *apache2.ext-nccoe.org*
- *ws1.int-nccoe.org*
- *ws2.int-nccoe.org*
- *ws3.int-nccoe.org*

1.5.5.3.2 Apache Installation

PHP is a key component of the Apache installation for the TLS lab implementation of all of the above web servers. PHP assists Apache and Drupal in serving dynamic web content. Please follow the instructions below for installing Apache and PHP.

For the Apache web server installation, please refer to this guidance: https://docs.fedoraproject.org/en-US/fedora/f28/system-administrators-guide/servers/Web_Servers/

All Drupal installations have dependencies on the base PHP application and its supplemental modules. In addition to the base PHP installation, also install the additional modules by using the following command.

- ```
  dnf install drush php php-mysqli php-json php-mbstring php-gd php-dom php-xml
  php-simplexml php-cli php-fpm php-mysqlnd php-pdop-gd php-dom php-xml php-
  simplexml php
  ```

1.5.5.3.3 Apache Web Services Configuration

The TLS lab enabled https on the Apache web servers. For instructions on setting up OpenSSL, refer to the “Using mod_ssl” section from the following link: https://docs.fedoraproject.org/en-US/quick-docs/getting-started-with-apache-http-server/

To allow http and https connections through the local Fedora firewall to Apache, perform the following steps:

- Type the following command to allow http connections to Apache:

  ```
  iptables-I INPUT -p tcp –dport 80 -mstate --state related, ESTABLISHED, new -j
  ACCEPT
  ```

- Type the following command to allow https connections to apache:

  ```
  iptables-I INPUT -p tcp –dport 443 -mstate --state related, ESTABLISHED, new -j
  ACCEPT
  ```

Save the newly created firewall rules with the following command: `iptables-save`
  ```

1.5.5.4 Drupal Web Content Management Services

Drupal is a scalable, open platform for web content management. Drupal can be installed on multiple OSs, including, Fedora, CentOS, and IIS. The TLS lab utilized Drupal to serve web pages on all three of the load balanced web servers in the public-facing DMZ.

1.5.5.4.1 Drupal Prerequisites

- PHP 5.5.9 or higher
- MySQL 5.5.3 or MariaDB 5.5.20
- Apache or IIS web server

1.5.5.4.2 Drupal Web Content Management System Download and Installation

One server should run throughout the setup process, including the database setup. The remaining two servers should be set up to point to the existing database once the first server has been set up. All web servers should be set up to use MariaDB, **not MSQL.** Use the guidance below for download, installation, and configuration of Drupal to simulate the TLS lab architecture:

- download: https://www.drupal.org/download
- Apache installation and configuration: https://www.drupal.org/docs/7/install
- IIS installation and configuration: https://www.drupal.org/docs/develop/local-server-setup/windows-development-environment/installing-on-windows-server

1.5.5.4.3 Web Services Drupal Configuration

A web service is a software system designed to support machine-to-machine interaction over a network. A web service is normally accessed over a network and then executed on a remote system hosting the requested services. Web services protocols normally use application programming interfaces (APIs) based on RESTful, simple object access protocol (SOAP), and extensible markup language (XML) protocols. It is a best practice to execute web services that carry critical personally identifiable information and other sensitive information by using TLS-based encrypted communication channels.

The TLS lab tested implementation of passive monitoring for TLS-enabled web services traffic. The rationale behind this approach is covered in the Symantec How-To guide section of this document. In Appendix A, Passive Inspection, see the full description of how the passive monitoring network was configured.

The web services servers are configured to test the basic passive TLS monitoring capability and are not typical of a fully operational web services implementation. The RESTful, SOAP, and XML protocols are not used in the TLS Lab. Rudimentary machine-to-machine communication over a secured TLS network is configured within each DMZ web server by using JavaScript, PHP, and Drupal's in-line What-You-See-Is-What-You-Get (also known as WYSIWYG) hypertext markup language (HTML) content creation editor.

A simple PHP script that was created for each web service prompted each of the three web services servers to retrieve and push its current times to the main web server. The JavaScript included in the Drupal-based DMZ servers was set to grab updates of the time each second by using https connectivity. Use the steps below to re-create this setup.

Part 1: Drupal DMZ Servers Configuration

1. Log in to Drupal by using the content administrator with enough rights to create a basic page.
2. Navigate to the following administrative menu item (top of the page on the left side, then use the links within the Content administration page itself to navigate to the remaining sections): **Content > Add Content > Basic Page**
3. Verify that a page is displayed that allows entry of data by using a **Title** and **Body** HTML form.
4. Give this page any title.
5. Before populating the body section of the page, ensure that the **Text Format** is set to **Full Html and PHP.** If that selection is not present, enable the **PHP Filter** module in the Drupal **Modules** section of Drupal, and try again.
6. Upon completing step 5, paste the following code into the body of the new document:

```
<div id="timeid"></div>

<?php

$serveraddress = $_SERVER['SERVER_ADDR'];

$javagettime = <<<EOFF
<script>
mydata = "TEST";
function ExportValues(mydata) {
         var xhttp;
        if (window.XMLHttpRequest) {
                // code for modern browsers
                xhttp = new XMLHttpRequest();
        } else {
                // code for IE6, IE5
                xhttp = new ActiveXObject("Microsoft.XMLHTTP");
        }
        xhttp.onreadystatechange = function() {
                if (this.readyState == 4 && this.status == 200) {
                        document.getElementById("timeid").innerHTML =
this.responseText;
                }
        };

        xhttp.open("GET", "https://$serveraddress/PHPTIME.php", true);
        xhttp.send();
```

```
}

ExportValues(mydata);
setInterval(function(){ ExportValues(mydata); }, 1000);
</script>

EOFF;
echo $javagettime;

?>
```

7. Click on the **Publishing options** tab below, then make sure that **Published** and **Promoted to front page** are selected as options.
8. **Save** the page.
9. Repeat these steps for each web services server.

Part II: Drupal DMZ Servers Configuration

The code above in Part I instructs the DMZ web server to connect to itself and execute the script *PHPTIME.php* within its own Drupal directory. This file will be created here in Part II. The *PHPTIME.php* file uses a curl script to simulate secure TLS server-to-server communication between the DMZ web server and its designated web services server. Follow the steps below to create this file on *all* the DMZ web servers.

1. Log in to the local web administration account for each of the three DMZ-based web servers. Navigate to the local Drupal stored file system where Drupal is served to the public. On Apache servers, this will be /var/www/html/<DRUPAL DIRECTORY NAME USED>. On IIS servers, this will be the Drupal document root for the website instantiation.
2. Launch a text editor (notepad++ or notepad for Windows or VIM or VI editor for Linux), then paste the following into that file:

```
<?php
      header("Access-Control-Allow-Origin: *");
      $ch = curl_init();

      curl_setopt($ch, CURLOPT_URL, 'https://ws2.int-nccoe.org');
      curl_setopt($ch, CURLOPT_RETURNTRANSFER, 1);
      curl_setopt($ch, CURLOPT_SSL_VERIFYHOST, false);
      curl_setopt($ch, CURLOPT_SSL_VERIFYPEER, false);

      $result = curl_exec($ch);
      if (curl_errno($ch)) {
            echo 'Error:' . curl_error($ch);
      }
      curl_close ($ch);

```

```
        echo $result;
?>
```

3. The following line will need to be changed on each DMZ web server and customized with the individual host name for the web services server assigned to the specific DMZ web server. Each DMZ web server should have its own individual web services server:

   curl_setopt($ch, CURLOPT_URL,'https://CHANGE TO YOUR MACHINE NAME');

4. Save this file with a .php extension into the root base directory of the Drupal site created for this demonstration.

Web Services Server Configuration

The web services server must be configured to check its own time and send the results back to the requesting DMZ web server via secure communication. Use the following guidance to set up the web services server.

1. Log in to the command line for each web services server, and navigate to the Apache document root configured in the *httpd.conf* file for Apache. In most cases it is */var/www/html*.
2. Open a VIM/VI editor and paste the following into that file:

```
<?php

$sourceip = $_SERVER['HTTP_ORIGIN'];

if (isset($_SERVER["HTTP_ORIGIN"]) === true) {
        $origin = $_SERVER["HTTP_ORIGIN"];
        $allowed_origins = array(

                // ANY
                $_SERVER['HTTP_ORIGIN']

                // SPECIFIC
                 "https://192.168.4.2",
                 "https://apache1.ext-nccoe.org",
                 "https://tls.nccoe.org",
                 "https://apache2.ext-nccoe.org",
                 "https://192.168.4.3",
                 "https://iis1.ext-nccoe.org",
                 "https://192.168.4.4"
        );
        if (in_array($origin, $allowed_origins, true) === true) {
                header('Access-Control-Allow-Origin: ' . $origin);
                header('Access-Control-Allow-Credentials: true');
                header('Access-Control-Allow-Methods: POST');
                header('Access-Control-Allow-Headers: Content-Type');
        }
        if ($_SERVER["REQUEST_METHOD"] === "OPTIONS") {
```

```
                exit; // OPTIONS request wants only the policy, we can stop
here
        }
}

$timetime = exec('date');

echo "WEB SERVICES SERVER2's TIME AN DATE IS: ". $timetime;

?>
```

3. Remember to save the file in the document root directory under the same name used in the previous section with the .php extension.
4. Ensure the Apache service is running: `service httpd restart`

Web Services Testing Process

1. Navigate to the public IP of the Drupal web servers (should be the F5 virtual ip or if behind a firewall, the IP address of the firewall used to NAT to the web server cluster behind the F5).
2. There should be at least three Basic Pages listed on the main site landing page. These should be the pages created in this section to point to the web services server.
3. Choose one by clicking on its title or **Read more** link beside the title.
4. The time should be automatically updating each second to indicate the web server is using its designated web services server to check time via TLS connection (indicated by the https).
5. If the time updates are not being seen, there could be an issue with the browser application accepting the valid certificate. If self-signed untrusted certificates instead of a trusted certificate are being used on the DMZ web servers, then the web client used (Chrome, Internet Explorer, or Edge) may not trust the individual server being accessed. To discover the issue, press the F12 key on the keyboard, then select the **Console** tab. If there is an error stating Net::ERR_CERT_AUTHORITY_INVALID or any other certificate validation error with an associated IP address, open a new tab and navigate directly to the IP address listed by using 192.168.3.85. If there is the standard certificate error for an untrusted site, then accept the risk if this is a laboratory environment. The time should pop up afterward, and the other tabs with the Drupal time connection will also work now. If this is production system, then a valid certificate will need to be placed on the machine with the IP listed. The client that browses that machine should trust the certificate.

1.5.5.5 Mail Services

The TLS lab utilizes a Simple Mail Transfer Protocol (SMTP) service to accept alerts from all the configured components on the network. The SMTP service was created on a Linux server running Fedora. The mail system was composed of a Dovecot Mail Transfer Agent (MTA) and a Postfix Mail User

Agent (MUA). The following section provides guidance on download, installation, and configuration of each service.

1.5.5.5.1 Mail Services Prerequisites

Before installing Dovecot and Postfix, set up the mail1.int-nccoe.org server by using the guidance in Section 1.5.2.2.2.

1.5.5.5.2 Installation and Configuration of Mail Services Postfix Mail Transfer Agent

Postfix is a free and open-source mail transfer agent that routes and delivers electronic mail. To download and install the Postfix MTA, follow the instructions in the following link:

- https://docs.fedoraproject.org/en-US/Fedora/12/html/Deployment_Guide/s3-email-mta-postfix-conf.html

  Note: The actual *main.cf* file used in the TLS lab build is in Appendix F.

1.5.5.5.3 Installation and Configuration of Mail Services Dovecot Mail Transfer Agent

Dovecot is an open-source Internet Message Access Protocol (IMAP) and Post Office Protocol 3 Mail User Agent server for Linux systems. It allows TLS administrators to manage and view email received by the Postfix server. To download and install the Dovecot MUA, please refer to the instructions in the following link:

- https://wiki.dovecot.org/BasicConfiguration

  Note: The actual *dovecot.conf* file used in the TLS lab build is in Appendix F.

1.5.5.6 Log Aggregation and Correlation Services

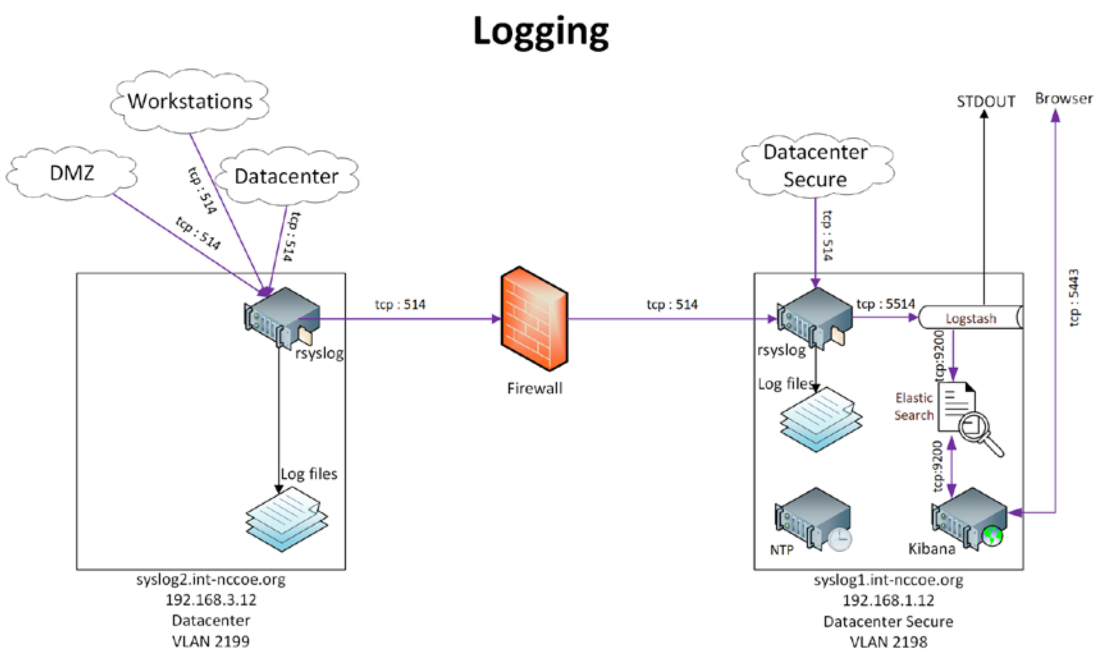
"ELK" stands for three open-source projects:

- Elasticsearch–a search and analytics engine
- Logstash–a server-side data processing pipeline that ingests data from multiple sources simultaneously, transforms it, and then sends it to a "stash" like Elasticsearch
- Kibana–lets users visualize data with charts and graphs in Elasticsearch

The TLS lab utilized the ELK stack log aggregation and correlation services to manage and visualize the remote logging services for all capable supplemental and collaborator products.

The following diagram depicts a view of the TLS lab logging infrastructure.

Figure 1-3 TLS Lab Logging Infrastructure



1.5.5.6.1 Prerequisites for Log Aggregation and Correlation Services

In accordance with the logging architecture above, the TLS lab utilized the hosts below. Both hosts must be configured with Fedora, based on the OS configuration guidance in Section 1.5.2.2.2. Configure both servers with rsyslog.

- syslog1.int-nccoe.org
- syslog2.int-nccoe.org
- Logstash requires Java 8 or Java 11.

1.5.5.6.2 Remote System Logging Services

Rsyslog is an open-source software utility used on UNIX and UNIX-like computer systems for forwarding log messages in an IP network.

- To install rsyslog use the command `dnf install rsyslog`

For more information on configuring rsyslog, refer to the following link:

- https://docs.fedoraproject.org/en-US/fedora/rawhide/system-administrators-guide/monitoring-and-automation/Viewing_and_Managing_Log_Files/#

1.5.5.6.3 Elasticsearch Installation and Configuration

Elasticsearch is a search engine based on the Lucene library. It provides a distributed, multitenant-capable full-text search engine with an http web interface and schema-free JavaScript Object Notation documents. Elasticsearch is developed in Java.

To install and configure Elasticsearch, please refer to the following link:

- https://www.elastic.co/guide/en/elasticsearch/reference/current/rpm.html

1.5.5.6.4 Kibana Installation and Configuration

Kibana is an open-source data visualization plug-in for Elasticsearch and provides visualization capabilities on top of the content indexed on an Elasticsearch cluster. Users can create bar, line, and scatter plots (or pie charts) and maps on top of large volumes of data.

To install and configure Kibana, please refer to the following link:

- https://www.elastic.co/guide/en/kibana/current/rpm.html

1.5.5.6.5 Logstash Installation and Configuration

Logstash is an open-source, server-side data processing pipeline that ingests data from a multitude of sources simultaneously, transforms it, and then sends it to the user's favorite stash.

To install and configure Logstash, please refer to the following link:

- https://www.elastic.co/guide/en/logstash/current/installing-logstash.html#package-repositories

1.5.6 DevOps Services

To show the automated management of TLS server certificates in a container-based environment, we used Kubernetes with Docker, NGINX, and Jetstack Cert-Manager.

1.5.6.1.1 Kubernetes Installation and Configuration

Instructions for installing Kubernetes are available at the following link:

- https://kubernetes.io/docs/setup/

We installed Kubernetes on three CentOS Linux systems (cluster1, cluster2, cluster3.int-nccoe.org).

1.5.6.1.2 Weave

We used Weave as the virtual network to facilitate communications between the Kubernetes master and nodes. Instructions for installing Weave can be found at the following link:

- https://www.weave.works/docs/net/latest/install/

1.5.6.1.3 Docker Installation and Configuration

We used the community edition of Docker with Kubernetes. Instructions for installing Docker on CentOS are found at the following link:

- https://docs.docker.com/install/linux/docker-ce/centos/

1.5.6.1.4 Jetstack Cert-Manager Installation and Configuration

We installed Jetstack Cert-Manager on Kubernetes with the necessary components to request certificates from Venafi TPP by using the following command:

```
kubectl apply -f https://raw.githubusercontent.com/jetstack \
/cert-manager/venafi/contrib/manifests/cert-manager/with-rbac.yaml
```

This automatically created a namespace named “cert-manager,” which we used for the rest of our configuration.

1.5.6.1.5 NGINX Installation and Configuration

NGINX was used as the web server and ingress on Kubernetes. Certificates were associated with the NGINX ingress. Instructions for installing and configuring NGINX on Kubernetes are found at the following link:

- https://www.nginx.com/

In our implementation, we installed NGINX on Kubernetes with the following command into the cert-manager namespace.

```
kubectl create deployment nginx –image=nginx -n cert-manager
```

We then created a service for NGINX by using the following command:

```
kubectl create service nodeport nginx –tcp=80:80 -n cert-manager
```

2 Product Installation and Configuration Guides

This section of the practice guide contains detailed instructions for installing and configuring all of the TLS collaborator products used to build an instance of the example solution. Each major subsection (2.1, 2.2, 2.x) is dedicated to a collaborator’s product capability. Within each product capability section, descriptions of each product capability align with a Day 0, Day 1, and Day N concept. It is important to note that each day builds on the previous day(s) for prerequisites, and each collaborator capability does the same. So, if the implementer’s intent is to fully replicate the TLS lab environment, then following the order of days and component installations will help make that endeavor more successful.

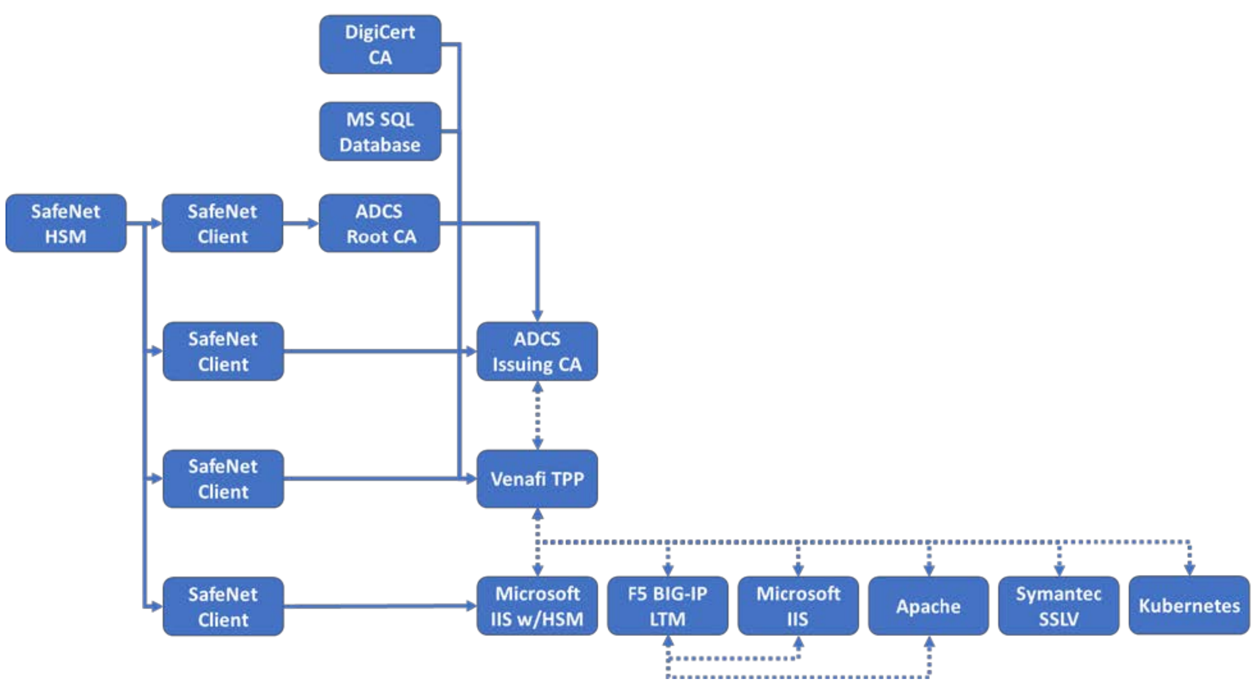
- **Day 0** provides how-to guidance from a first-day installation perspective. It is assumed the implementer is getting acclimated with the collaborator product. The implementer should complete all prerequisites, which include complete installations of other collaborator products in some instances or the Supporting Architecture described in Section 1.3. The expectation is for only basic crucial configuration functions to get the system up and running. Otherwise, other configurations should be executed on Day 1, or there may be issues with prerequisites that have not been executed.

- **Day 1** assumes all Day 0 activities have been completed, including all prerequisites. Expected activities include how-to guidance on more advanced security configuration of functioning in the TLS environment. Day 1 also assists the implementer with configuration guidance for integration with any other collaborator product capabilities.
- **Day N** assists the implementer with all necessary configurations and integrations of systems that help facilitate ongoing security management and maintenance. In most cases, the minimum Day N configuration and integration include security event audit and event logging for TLS systems. In all cases, there are variations of services and offerings, which each collaborator describes in their respective sections.

2.1 Product Installation Sequence (Example Build)

Figure 2-1 shows the dependencies among components deployed for the example build. A solid line with a single arrow signifies hard dependencies. The component from which the arrow points should be installed before the component to which the arrow points. This facilitates phased and secure deployment. A dashed line with a double arrow indicates that integration between the components is not dependent on the installation sequence (i.e., either component can be installed first).

Figure 2-1 Overview of Dependencies Among Components Deployed for the Example Build



2.2 SafeNet AT Luna SA 1700 Hardware Security Module

HSMs are specialized hardware devices dedicated to maintaining the security of sensitive data throughout its life cycle. HSMs provide tamper-evident and intrusion-resistant protection of critical keys and other secrets, and off-loading of processing-intensive cryptographic operations. By performing cryptographic operations within the HSM, sensitive data never leaves the secure confines of the hardened device.

The SafeNet AT Luna SA for Government is a network-attached HSM with multiple partitions to effectively provide a many-in-one solution to multiple tenants—each with its own security officer management credentials. Depending on security needs, the Luna SA can be used with or without a secure personal identification number entry device (PED) for controlling management access to the HSM partitions. Utilizing the PED takes the HSM from a Federal Information Processing Standards (FIPS) 140-2 Level 2 certified device to Level 3. The Luna SA also comes in two performance models: the lower performance 1700, and the high-performance 7000 for transaction-intensive use cases.

2.2.1 Day 0: Product Installation and Standard Configuration

2.2.1.1 Prerequisites

2.2.1.1.1 Rack Space

Installation of the HSM requires rack space with the following characteristics:

- standard 1u 1 gin rack mount chassis
- dimensions: 19" x 21" x 1.725" (482.6 millimeters [mm] x 533.4 mm x 43.815 mm)
- weight capacity: 28 pounds (lb) (12.7 kilograms [kg])
- input voltage: 100-240 V.50-60 hertz
- power consumption: 180 watts (W) maximum, 155 W typical
- temperature: operating 0 degrees Celsius (C)–35 degrees C, storage 20 degrees C–60 degrees C
- relative humidity: 5% to 95% (38 degrees C) noncondensing

2.2.1.1.2 Networking

One of two approaches to networking may be used. The steps for the commands in this document assume the NCCoE’s laboratory networking environment will be replicated. An organization may also opt to use its own network settings. In either case, the following Luna SA HSM appliance parameters information will be needed:

- IP address that will be assigned to this device (Static IP is recommended)
- Host name for the HSM appliance (registered with network DNS)

- a domain name where the device will reside
- default gateway IP address
- DNS Name Server IP address(es)
- Search Domain name(s)
- device subnet mask
- Ethernet device (use eth0, which is the uppermost network jack on the HSM appliance back panel, closest to the power supply, and labeled **1**)

The network must be configured for optimal use of Luna appliances. The following bandwidth and latency recommendations are optimal for performance settings:

- bandwidth
  - minimum supported: 10 megabit (Mb) half-duplex
  - recommended: at least 100 Mb full duplex—full gigabit Ethernet is supported

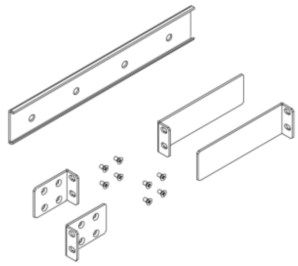
    Note: Ensure the network switch is set to AUTO negotiation, as the Luna appliance negotiates at AUTO. If the network switch is set to use other than automatic negotiation, there is a risk that the switch and the Luna appliance will settle on a much slower speed than is actually possible in the organization’s network conditions.
- network latency
  - maximum supported: 500 milliseconds (ms)
  - recommended: 0.5 ms

2.2.1.1.3 Unpacking the Appliance

Follow this checklist to verify that all of items required for the installation are in hand.

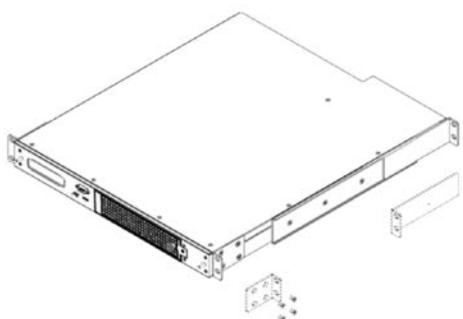
Qty	Item
1	 Luna SA HSM appliance

Qty	Item
2	 power supply cord (one for each power supply; style to suit country for which was ordered)
1	 null modem serial cable
1	 Universal Serial Bus 2.0 to RS232 serial adapter

Qty	Item
1	 Set of: - 2 front mounting brackets with screws - 2 side bracket guides - 2 sliding rear brackets (Fit into the guides for rear support adjustable positioning.)
1	 client/software development kit (SDK) software

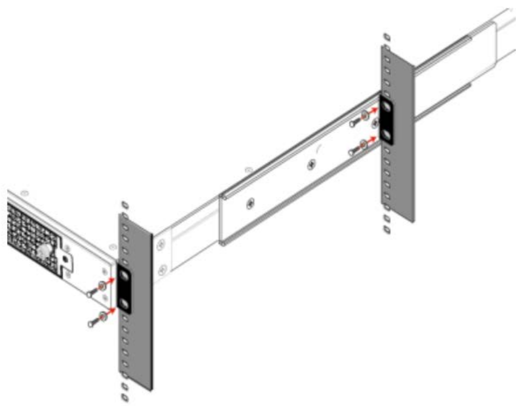
2.2.1.2 Rack-Mount the Appliance

1. Install and adjust rails and brackets to suit the equipment rack.

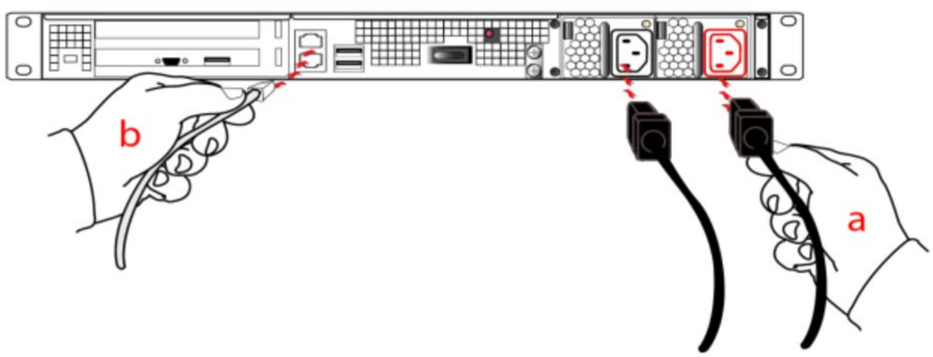


2. Mount the appliance in the equipment rack. Alternatively, ignore the rails and mounting tabs, and rest the Luna SA appliance on a mounting tray or shelf suitable for the organization’s specific style and brand of equipment rack.

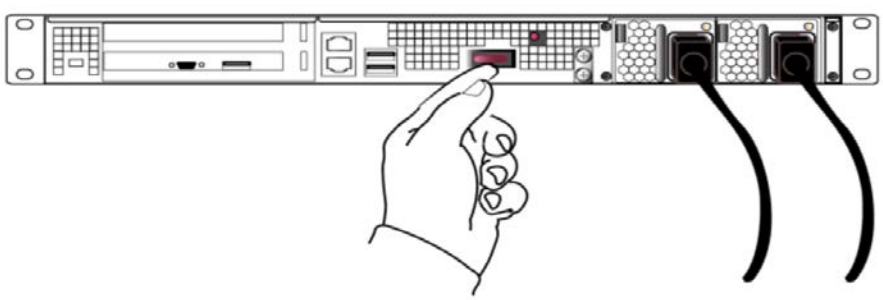
CAUTION: Support the weight of the appliance until all four brackets are secured.



3. Insert the power (a) and network (b) cables at the rear panel. For proper redundancy and best reliability, the power cables should connect to two completely independent power sources.



4. Press and release the Start/Stop switch, on the rear panel.



2.2.1.3 Initial Appliance Configuration

This section describes the process to prepare the new HSM Server and one client system for operation with the application. It includes the following steps:

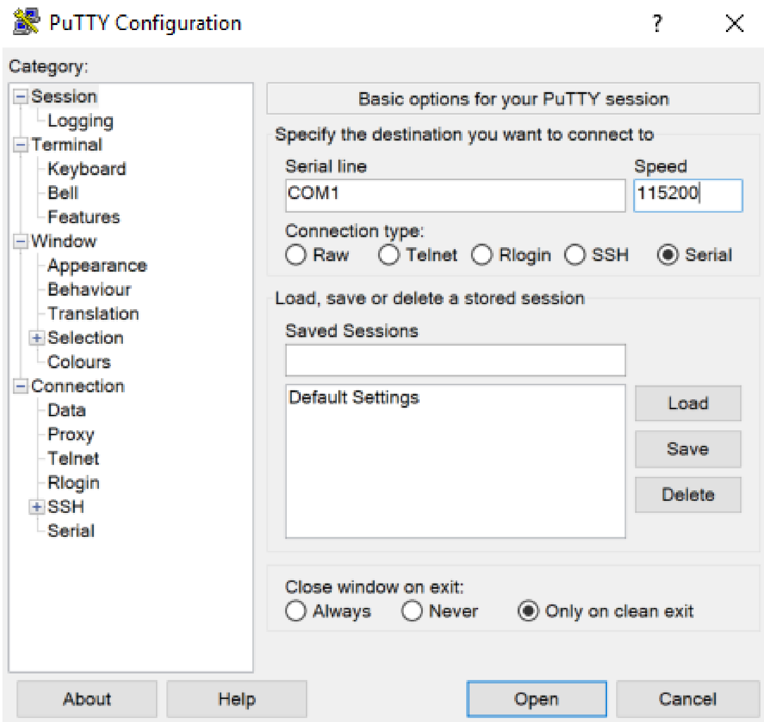
- process for first-time login and changing passwords

- verify and set the date and time
- configure HSM appliance’s IP and network parameters (using static or Dynamic Host Configuration Protocol [DHCP]. In general, we strongly recommend against using DHCP for HSM appliances.)
- make network connections (To make a network connection, refer to Section 1.1.1.3.)
- HSM initialization process
- restart services so configuration changes can take effect

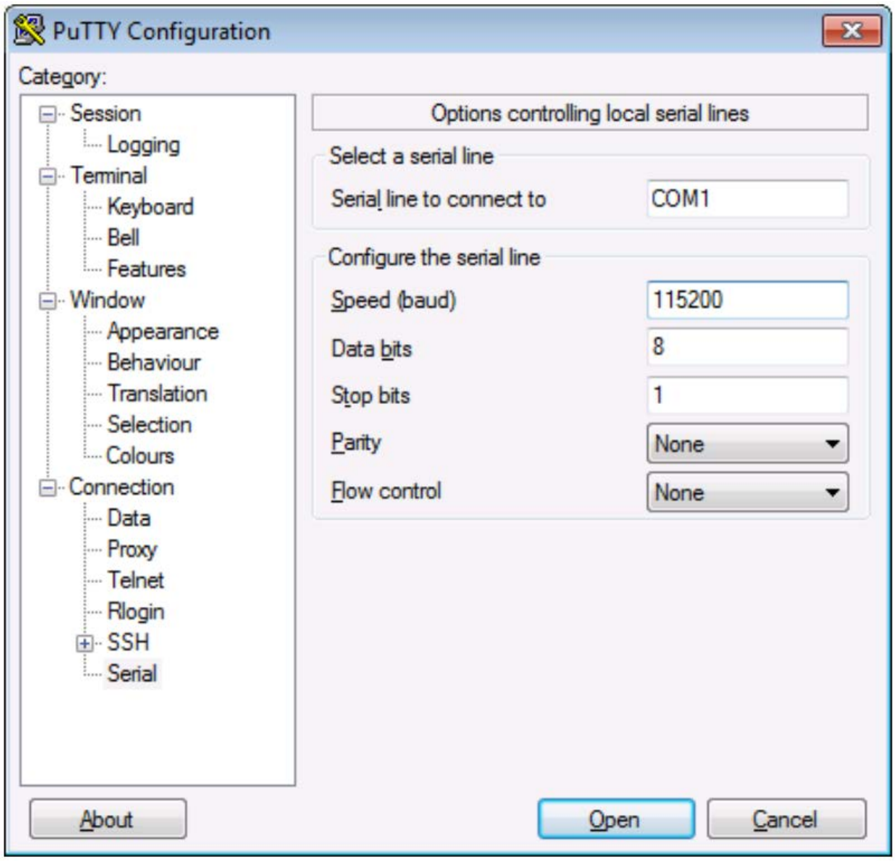
2.2.1.3.1 Process for First-Time Login and Changing Passwords

1. To perform initial login to the HSM appliance, connect a serial cable to serial port on the front of the appliance.

2. On the management laptop, open the PuTTY application and select a **Connection type** of **Serial** with a **Speed** of **115200.**



3. Navigate to the **Serial** Category on the bottom left side of the window.
4. Configure the serial connection to support the SSL Visibility Appliance's console speeds by selecting the following options:
   - **Speed (baud):** 115200
   - **Data bits:** 8
   - **Stop bits:** 1
   - **Parity:** None
   - **Flow control:** None



5. Log in to the appliance by using the default credentials of:
   - **username:** bootstrap
   - **password:** bootstrap
6. For security purposes, the user is immediately prompted to change the factory-default password for the admin account.

[localhost] ttyS0 login: admin
Password:
You are required to change your password immediately (root enforced)
Changing password for admin
(current) UNIX password:

```
A valid password should be a mix of upper and lower case letters, digits, and
other characters. You can use an 8 character long
password with characters from at least 3 of these 4 classes.
An upper case letter that begins the password and a digit that
ends it do not count towards the number of character classes used.
```

```
Enter new password:
   Re-type new password:
Luna SA 5.4.0-14 Command Line Shell - Copyright (c) 2001-2013 SafeNet, Inc. All
   rights reserved.
Command Result: 0 (Success)
   lunash:>
```

The above represents a local serial connection; text will differ slightly for a Secure Shell (SSH) connection.

Note: The username and passwords are case-sensitive.

Note: To protect the HSM appliance and its HSM from vulnerabilities due to weak passwords, new passwords must be at least eight characters in length and must include characters from at least three of the following four groups:

– lowercase alphabetic (abcd...xyz)

– uppercase alphabetic (ABCD...XYZ)

– numeric (0123456789)

– special (nonalphanumeric, #*@#$%&...)

Note: Login must occur within two minutes of opening an administration session, or the connection will time out.

2.2.1.3.2 Date and Time

To configure the HSM's date and time, perform the following steps:

1. Verify the current date and time on the HSM Server.
2. At the lunash prompt, type the command:

   `lunash:>` **`status date`**

3. If the date, time, or time zone is incorrect for the location, change them by using the `lunash sysconf` command. For example: `lunash:>` **`sysconf timezone set Canada/Eastern`**
   `Timezone set to Canada/Eastern`
4. Use sysconf time to set the system time and date <HH:MM YYYYMMDD> in the format shown. Note that the time is set on a 24-hour clock (00:00 to 23:59).
   `lunash`**`:> sysconf time 12:55 20190410 Sun April 10 12:55:00 EDT 2019`**
5. Optionally to configure Network Time Protocol (NTP), use the following command:

   `lunash:>` **`sysconf ntp addserver 192.168.1.12`**

6. Activate the NTP service with the following command:

   `sysconf ntp enable`

2.2.1.3.3 Network Configuration

1. Use the `network show` command to display the current settings and to see how they need to be modified for the network.

```
lunash:>net show
    Hostname:          HSM
    Domain:            int-nccoe.org

    IP Address (eth0): 192.168.1.13
    HW Address (eth0): 00:15:B2:AB:D6:D6
    Mask (eth0): 255.255.255.0
    Gateway (eth0):    192.168.1.1


Name Servers: 192.168.1.6
    Search Domain(s): <not set>

   Kernel IP routing table
   Destination Gateway Genmask Flags Metric Ref Use Iface
   Link status
     eth0: Configured
           Link detected: yes
     eth1: Configured
           Link detected: no

   Command Result : 0 (Success)
   lunash:>
```

2. Use `network hostname` to set the host name of the HSM appliance (use lowercase characters).
   `lunash:> network hostname HSM`

3. Use `network domain` to set the name of the network domain in which the HSM Server (appliance) is to operate.
   `lunash:> net domain int-nccoe.org`

4. Use `network dns add nameserver` to set the Nameserver IP Address (address for the local name server).
   `lunash:> net dns add nameserver 192.168.1.6`

5. Use `net dns add searchdomain` to set the DNS Search Domain (the search list to be used for host name lookups).
   `lunash:> net dns add searchdomain int-nccoe.org`

6. Use `network interface` to change network configuration settings.

   All of the `network interface` parameters are required for the IP setup of the Ethernet device and must be set at the same time for the HSM appliance to connect with the network.

```
[HSM] lunash:>net interface -device eth0 -ip 192.168.1.13 -netmask 255.255.255.0 -
gateway 192.168.1.1
```

7. View the new network settings with `network show`.
   `lunash:> network show`

2.2.1.3.4 Generate a New HSM Server Certificate

Although the HSM appliance came with a server certificate, good security practice dictates that a new one be generated.

1. Use `sysconf regenCert` to generate a new server certificate:

```
lunash:> sysconf regenCert 192.168.1.13
WARNING !! This command will overwrite the current server certificate and private
key.
All clients will have to add this server again with this new certificate.
If you are sure that you wish to proceed, then type 'proceed', otherwise type
'quit'
> proceed
Proceeding...
'sysconf regenCert' successful. NTLS must be (re)started before clients can
connect.
Please use the 'ntls show' command to ensure that NTLS is bound to an appropriate
network device or IP address/hostname for the network device(s) NTLS should be
active on. Use 'ntls bind' to change this binding if necessary.

Command Result: 0 (Success)
lunash:>
```

2.2.1.3.5 Bind the Network Trust Link Service

From the factory, the network trust link service (NTLS) is bound to the loop-back device by default. To use the appliance on the network, bind the NTLS to one of the two Ethernet ports— ETH0 or ETH1—or to a host name or IP address. Use the `ntls show` command to see current status.

1. Use `ntls bind` to bind the service:

```
lunash:>ntls bind eth0 -bind 192.168.1.13
Success: NTLS binding hostname or IP Address 192.168.1.13 set.
NOTICE: The NTLS service must be restarted for new settings to take effect.
If you are sure that you wish to restart NTLS, then type 'proceed', otherwise
type 'quit'
> proceed
Proceeding...
Restarting NTLS service...
Stopping ntls:                                           [   OK  ]
Starting ntls:                                           [   OK  ]
Command Result : 0 (Success)
[myluna] lunash:>ntls show
NTLS bound to network device: eth0  IP Address: "192.168.1.13" (eth0)
Command Result : 0 (Success)
```

NOTE: The “Stopping ntls” operation might fail in the above example, because NTLS is not yet running on a new HSM appliance—ignore this message. The service restarts regardless if the stop was needed.

2.2.1.3.6 Enabling Federal Information Processing Standards 140-2 Mode

In many areas of the information security industry, validations against independent or government standards are considered a desirable or essential attribute of a product. NIST's FIPS 140 is the pre-eminent standard in the field of cryptography. Enabling FIPS 140-2 ensures the HSM uses strong cryptographic modules in its operations.

1. Log in to the APPLIANCE management console (LunaSH) as admin.
    a. SSH into the APPLIANCE
    b. Use these credentials: Username: admin Password: ****YOUR admin PASSWORD****
2. Check if FIPS 140 mode is enabled.
    a. Command: `hsm show`
    b. In the results, look for "The HSM is in FIPS 140-2 approved operation mode." If this is seen, then stop: FIPS 140-2 mode is already enabled on the HSM. Otherwise, continue.
3. Log in to the admin role.
    a. Command: `hsm login`
    b. Password: ****YOUR admin PASSWORD****
4. View HSM Capabilities and Policies.
    a. Command: `hsm showPolicies`
    b. In the results, look for "Allow non-FIPS algorithms" and record its value and code.
5. Edit HSM Capabilities and Policies.
    a. Command: `hsm changePolicy -policy <code>  -value <desired_value>`
        i. `hsm changePolicy -policy 12 -value 1`
        ii. When prompted type: `proceed`
6. Confirm FIPS 140 mode is enabled.
    a. Command: `hsm show`
    b. In the results, look for "The HSM is in FIPS 140-2 approved operation mode." If this is seen, then stop: FIPS 140-2 mode is already enabled on the HSM. Otherwise, further investigation is required.

2.2.1.4 HSM Initialization

In this section, initialize the HSM portion of the Luna appliance and set any required policies. In normal operations, these actions are performed when first commissioning the Luna appliance.

2.2.1.4.1 Initialize a Password-Authenticated HSM

1. To initialize the HSM, type the following command:
   `hsm -init -label HSM`

```
 [HSM] lunash:> hsm -init -label HSM
> Please enter a password for the security officer
> ********
Please re-enter password to confirm:
> ********
Please enter the cloning domain to use for initializing this
HSM (press <enter> to use the default domain):
```

```
> ********
Please re-enter domain to confirm:
> ********
CAUTION:  Are you sure you wish to re-initialize this HSM?
All partitions and data will be erased.
Type 'proceed' to initialize the HSM, or 'quit'
to quit now.
>proceed
'hsm - init' successful.
```

2. When activity is complete, lunash displays a "success" message.

2.2.2 Day 1: Product Integration Configuration

2.2.2.1 Prerequisites

- NTL–This step will need to be completed for each system; refer to Section 2.2.2.2.
- ADCS–Windows server needs to be running; refer to guide.
- IIS–Windows server needs to be running; refer to guide.
- Venafi–must be installed and configured; refer to Section 2.2.2.2.

2.2.2.2 Network Trust Link

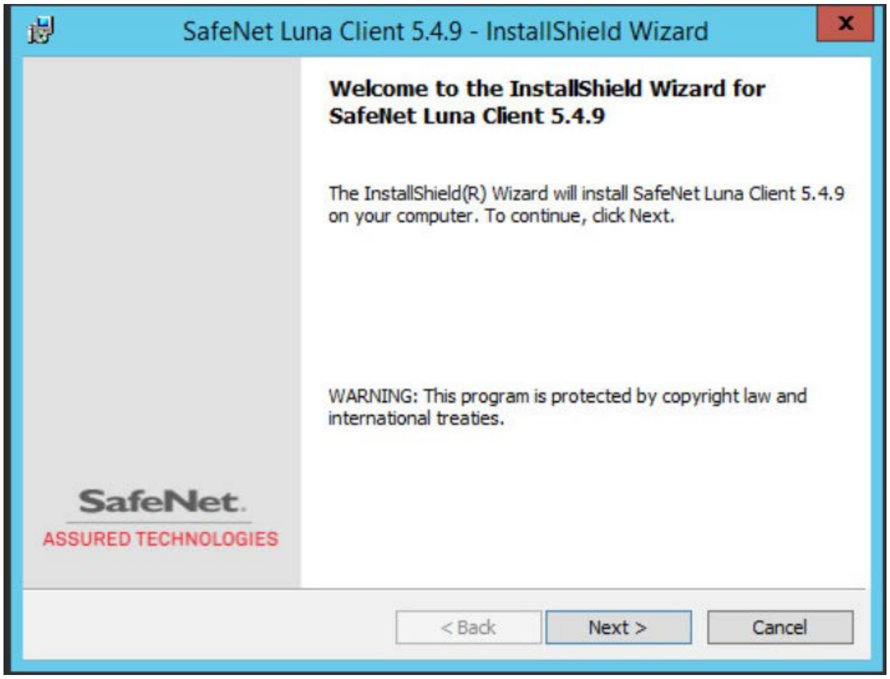
This section provides directions to configure a Luna Client to communicate with the network-attached Luna SA HSM. A client may have multiple Luna SA HSMs connected—using a slot designation when referencing an assigned Luna SA. The client also assumes the Luna SA is installed and operational but without a partition created for the new client.

The Luna Client is available in Windows and Linux. For Linux systems, refer to SafeNet AT's Configuring a Network Trust Link documentation. In this document, the necessary commands and screenshots are listed for Windows-based systems.

2.2.2.2.1 Install the Luna Client Software

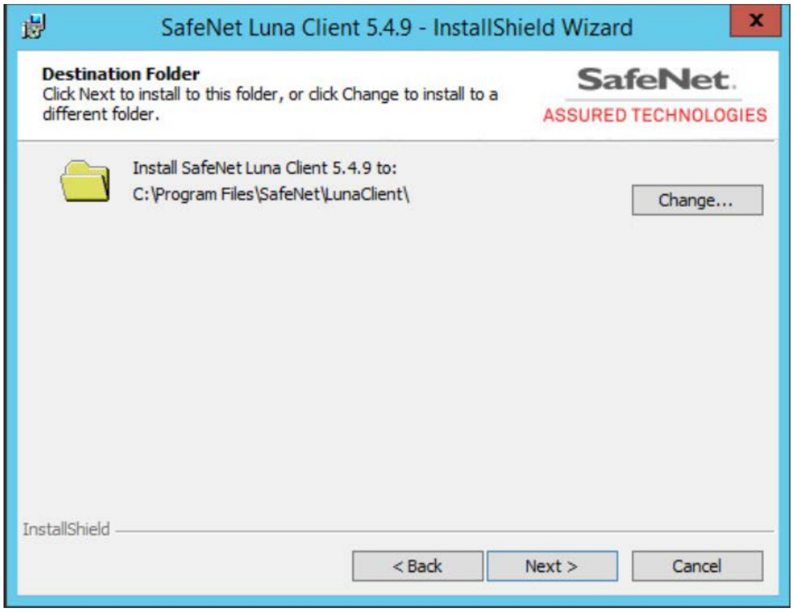
To install the Luna Client software, perform the following steps:

1. Log in to Windows as Administrator or as a user with administrator privileges.
2. Insert the Luna Client Software DVD into the optical drive.
3. Open a file explorer and navigate to **D:\windows\64\.**
4. Double-click **Luna Client.msi.**
5. Click **Next** at the welcome screen.

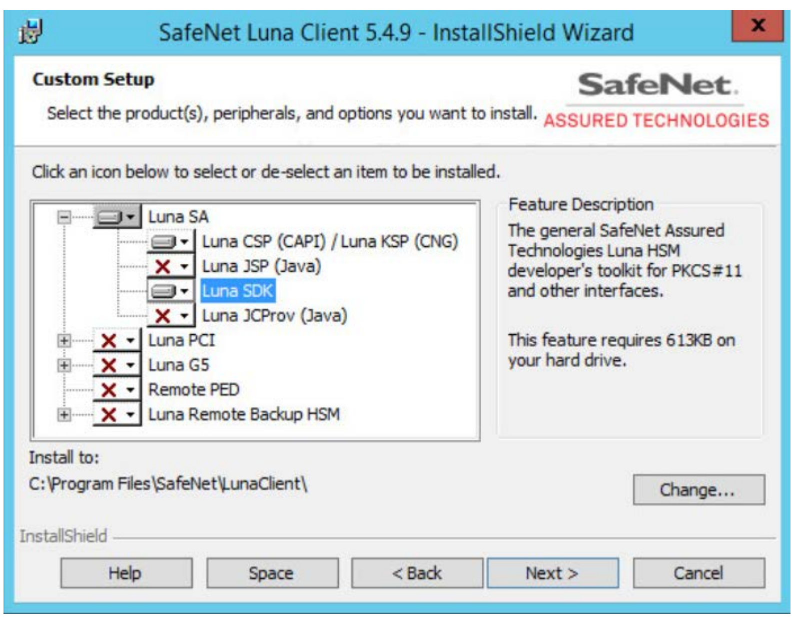


6. Accept the software license agreement by clicking "**I accept the terms in the license agreement**" and clicking **Next.**

7. In the Choose Destination Location dialogue, accept the default offered and click **Next.**



8. Ensure the following options are selected and click **Next:**
    - **Luna CSP (CAPI)/Luna KSP (CNG)**
    - **Luna SDK**



9. On the **Ready to Install** page, click **Install.**

10. If Windows presents a security notice asking if the user wishes to install the device driver from SafeNet AT, click **Install** to accept.

11. When the installation completes, click **Finish.**

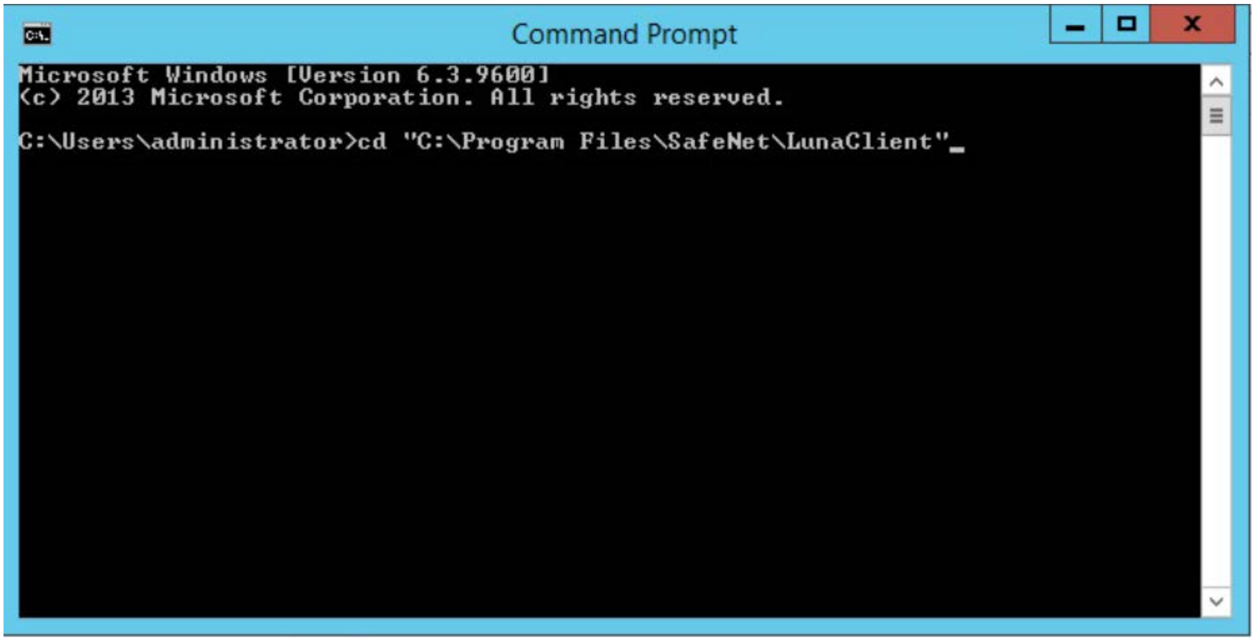
2.2.2.2.2 Configure the Luna Client

To establish the NTL, first create a client certificate, and then the client and server certificates are exchanged. The Luna SA appliance is then added as a trusted server in the client.

2.2.2.2.3 Create the Client Certificate

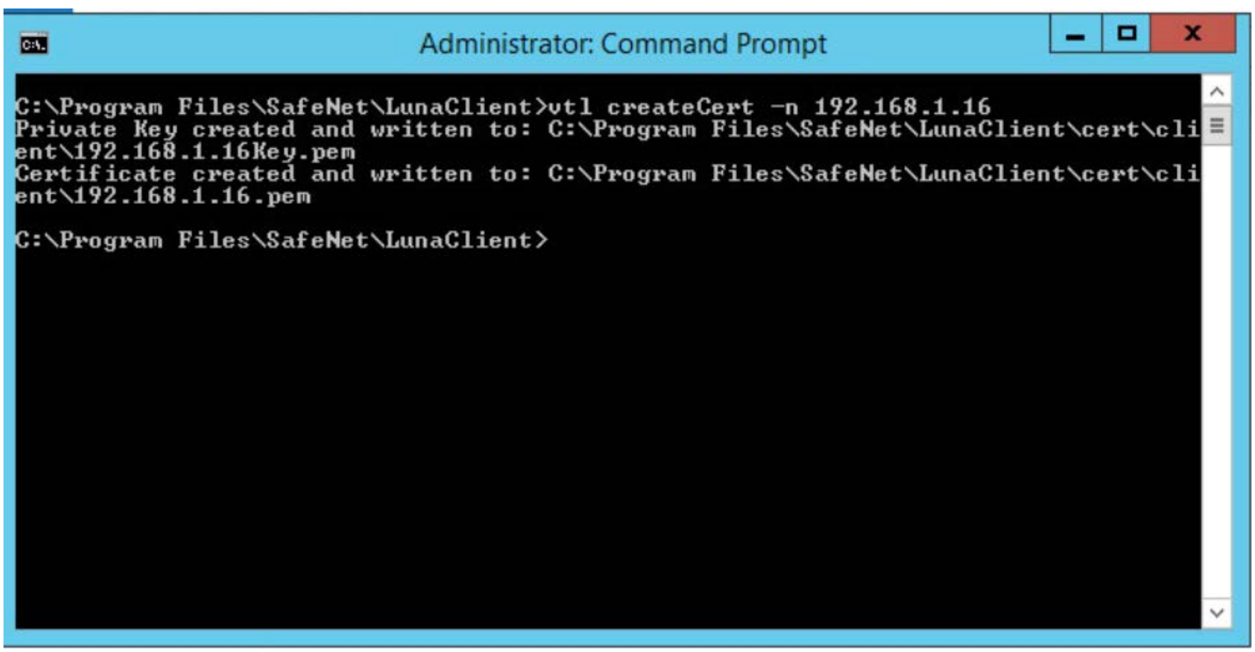
First, create the client certificate by using the SafeNet AT VTL command line. This results in a *.pem* certificate file being created in a \cert\client subfolder.

1. On the client system, from the Windows command environment, run as administrator and navigate to the folder *C:\Program Files\Safenet\LunaClient* **.**



2. Enter the following command:

```
vtl createcert –n <client IP address>
```

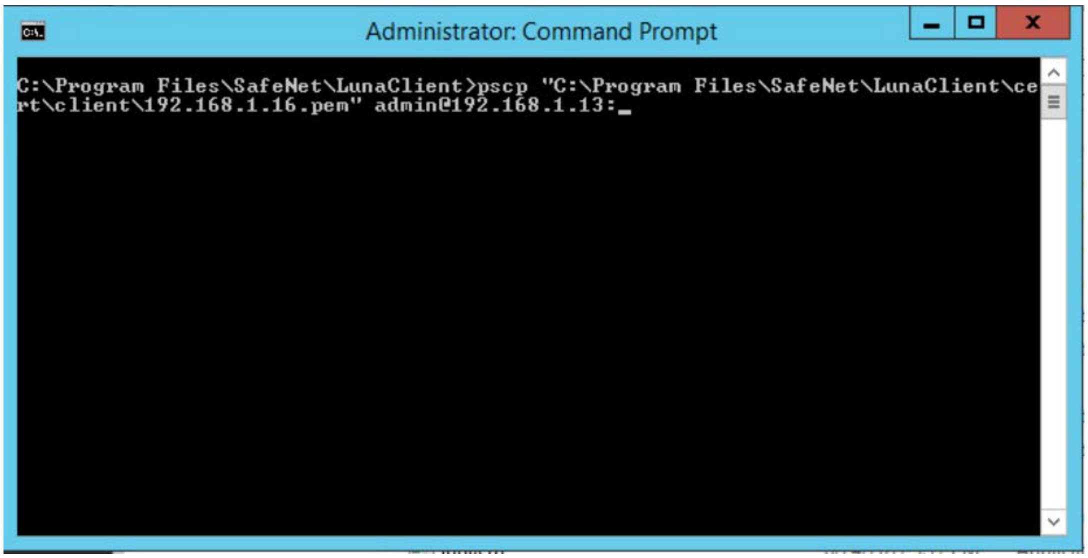


2.2.2.2.4 Transfer the Client Certificate to the Luna SA

Now, transfer the newly created client certificate to the Luna SA by using the PuTTY Secure Copy Protocol (PSCP) or Secure Copy Protocol (SCP) tool.

1. On the client system using Windows, enter the following command:

```
pscp “C:\Program Files\SafeNet\LunaClient\cert\client\192.168.1.16.pem”
admin@192.168.1.13:
```



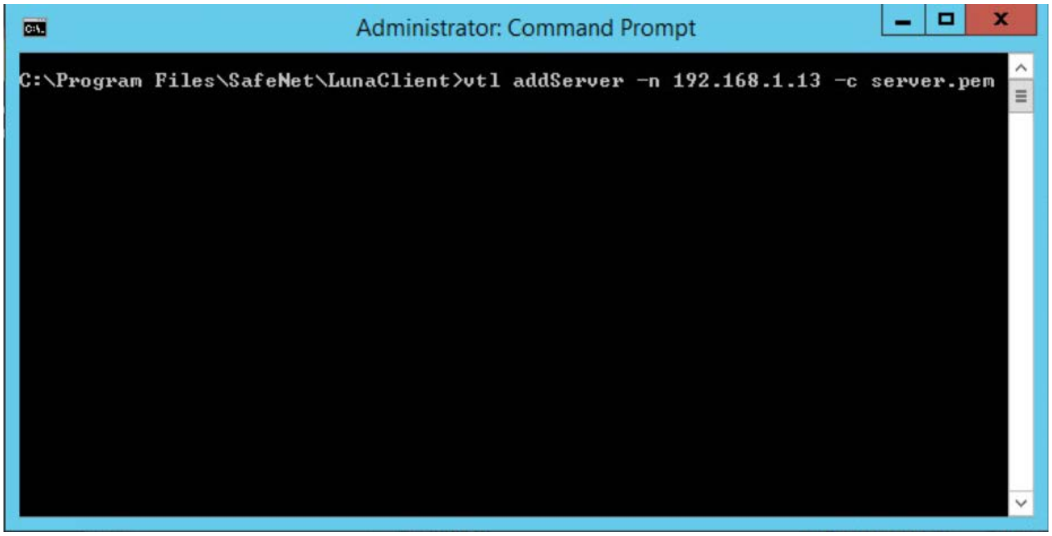
2. When prompted, enter the appliance administrative password for the Luna SA. The transfer automatically takes place.

2.2.2.2.5 Transfer the Server Certificate from the Luna SA

Using PSCP or SCP, transfer the Luna SA’s server certificate to the client.

1. On a client system using Windows, enter the following command:

```
pscp admin@192.168.1.13:server.pem
```



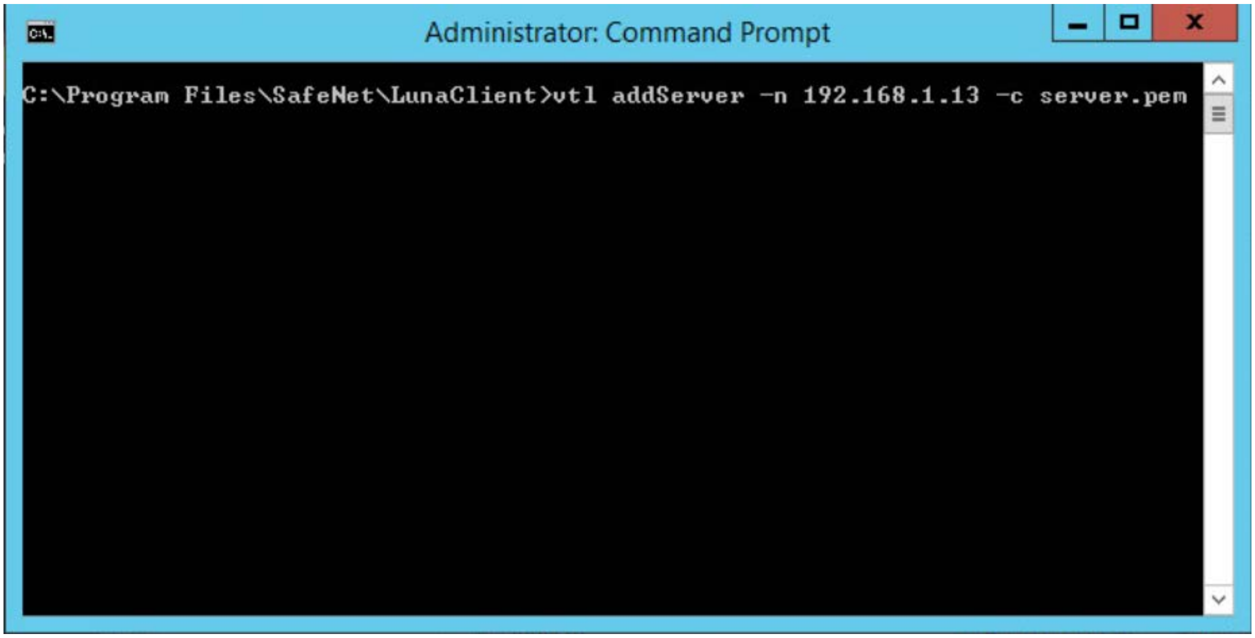
2. When prompted, enter the administrative password for the Luna SA. The transfer will automatically take place.

2.2.2.2.6 Register the HSM on the Client

The final step in configuring the client is to register the Luna SA's certificate with the client.

1. On a client system, enter the following command:

```
vtl addServer -n <HSM IP Address> -c server.pem
```



At this point, the client is fully configured and ready to establish a secure link with the HSM.

2.2.2.2.7 Create a Partition (Password Authentication)

1. Connect into the HSM via SSH or Serial.
2. At the `lunash:>` prompt on the Luna SA, enter the following command:

```
partition create –partition <partition name> -domain <domain name>
```

```
[HSM] lunash:>partition create -partition HRhsmiis

Please ensure that you have purchased licenses for at least this number of partitions: 5

  Please enter a password for the partition:
  > ************

  Please re-enter password to confirm:
  > ************

  Please enter a cloning domain to use when creating this partition:
  > *************

  Please re-enter cloning domain to confirm:
  > *************

If you are sure to continue then type 'proceed', otherwise type 'quit'
> proceed
Proceeding...

'partition create' successful.
```

3. When prompted, enter and re-enter to confirm the partition password.
4. Enter `proceed` when prompted.

2.2.2.2.8 Register the Client on the HSM and Assign It to a Partition

Register the client on the HSM and assign it to a partition. Because the HSM was previously created and the client certificate was transferred to it, the HSM can find the certificate file based on the IP address. Assign a name for the client for easy recognition.

1. On the Luna SA, enter the following command to register the client:

   ```
   client register –client HRhmsiis -ip 192.168.1.16
   ```

   ```
   [HSM] lunash:>client register -client HRhsmiis -ip 192.168.1.16
   ```

2. On the Luna SA, enter the following command to assign the client to the previously created partition.

   ```
   client assignPartition -client <client name> -partition <partition name>
   ```

   ```
   [HSM] lunash:>client assignPartition -client HRhsmiis -partition HRhsmiis_
   ```

3. On the Luna SA, enter the following command to verify the client is assigned to the proper partition.

   ```
   client show -client <client name>
   ```

```
[HSM] lunash:>client show -client HRhsmiis


ClientID:      HRhsmiis
IPAddress:     192.168.1.16
HTL Required:  no
OTT Expiry:    n/a
Partitions:    "HRhsmiis"


Command Result : 0 (Success)
```

At this point, the HSM is configured, and in the next section, the user will return to the client to verify connectivity and the ability to request cryptographic operations from the client.

2.2.2.2.9 Verify the Network Trust Link

Return to the client and verify it can view the Luna SA and its associated slot and partition. Run the Multitoken2 utility to verify the client can request cryptographic operations from the HSM.

2.2.2.2.10 Verify the Luna SA in Client Server Lists

Verify the Luna SA is in the client's server lists.

1. On the client system, from the Windows command environment run as administrator, navigate to the folder *C:\Program Files\Safenet\LunaClient.*

2. On the client system, enter the following command and verify the Luna SA is in the list of servers:

```
vtl listservers
```

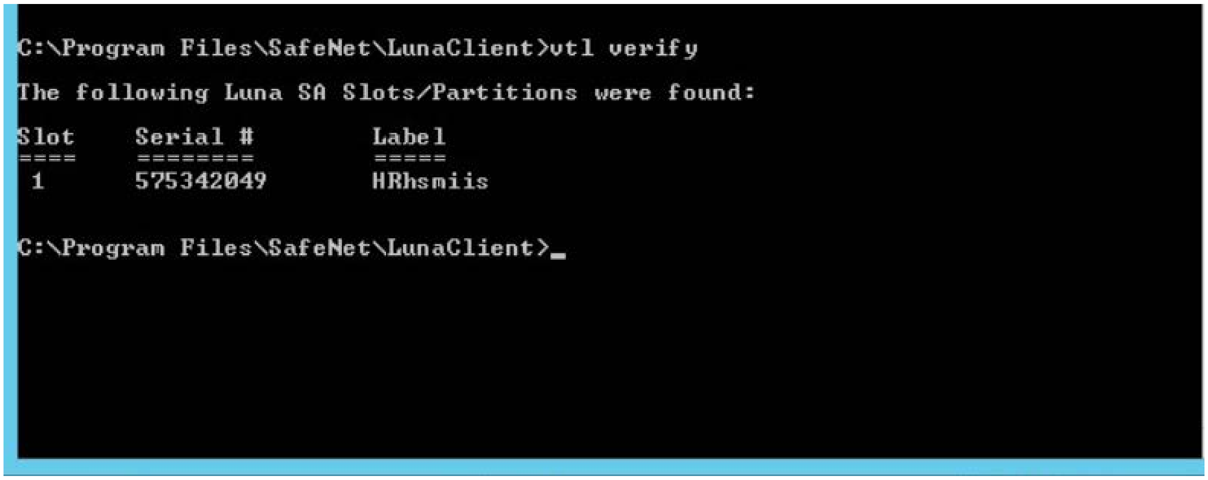
```
C:\Program Files\SafeNet\LunaClient>vtl listservers
Server: 192.168.1.13     HTL required: no
```

2.2.2.2.11 Verify the Slot and Partition

Verify the slot and the assigned HSM partition can be seen.

1. On the client system using either Windows and Linux, enter the following command to verify the Luna SA slot and partition are known to the client:

```
vtl verify
```



Should this verification fail, check the times on the client and HSM to ensure they are set properly.

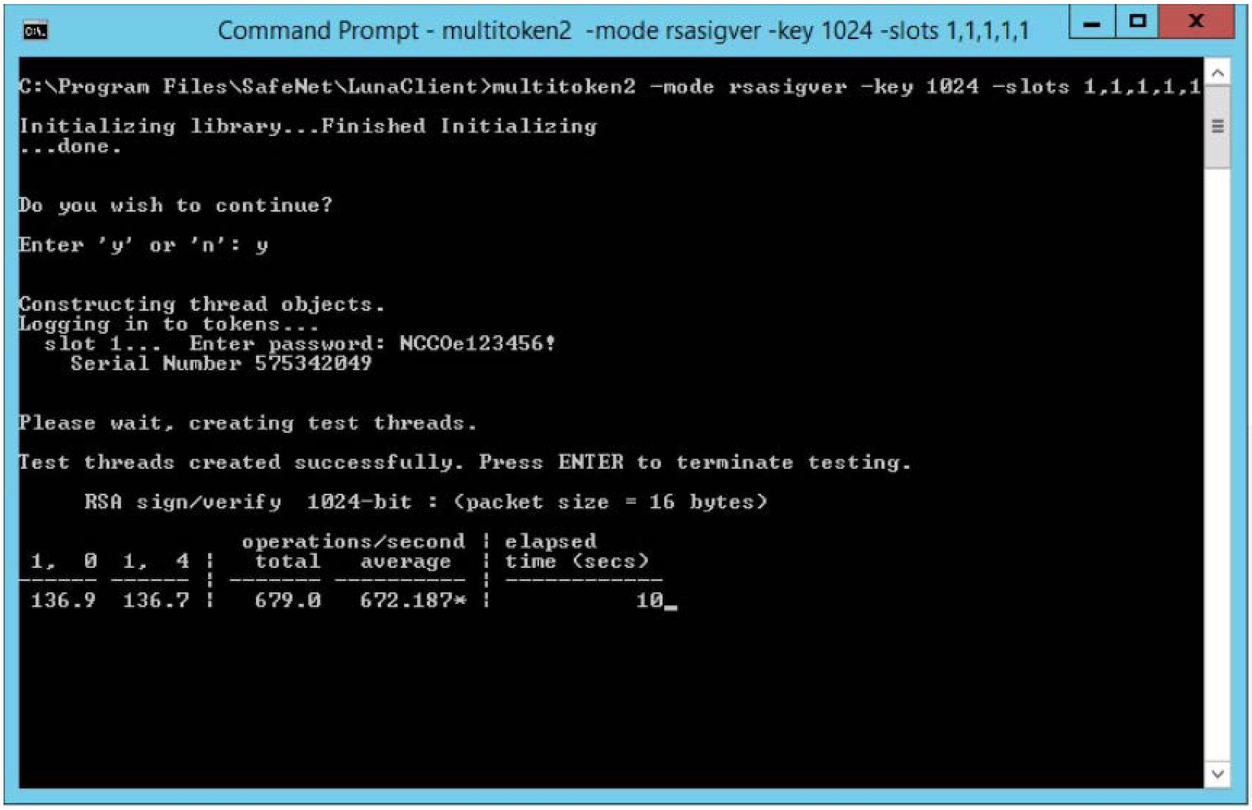
2.2.2.2.12 Request Cryptographic Operations on the HSM

Request an actual crypto operation on the HSM to verify full functionality. The Multitoken utility to use is described in the Luna SA product documentation.

1. On the client system, enter the following command:

```
multitoken2 -mode rsasigver -key 1024 -slots 1,1,1,1,1
```

2. When prompted, if continuing, enter **y.**
3. Enter the partition password when prompted. The test will begin.
4. Press the **Enter** key to terminate the test after verifying that RSA signatures were successfully performed in the statistics table.



2.2.2.3 ADCS Integration Configuration

This section provides the necessary steps for configuring an ADCS CA to use the SafeNet AT Luna SA 1700 HSM for Government, to secure the CA's private key. This section assumes the Luna HSM client has been installed and configured, as detailed in Section 2.2.1.

Perform the following steps:

- Verify the Network Trust Link (NTL) between the Windows Server and the HSM.
- Register the Key Storage Provider (KSP) on the Windows Server.
- Add the CA role.
- Verify the private key for the CA was created on the HSM.

2.2.2.3.1 Prerequisites

To configure Microsoft CA to use the Luna HSM, the following prerequisites must be met:

- The SafeNet AT Luna HSM is installed and operational.
- The SafeNet AT Luna Client is installed on the Windows Server where the CA is being added.

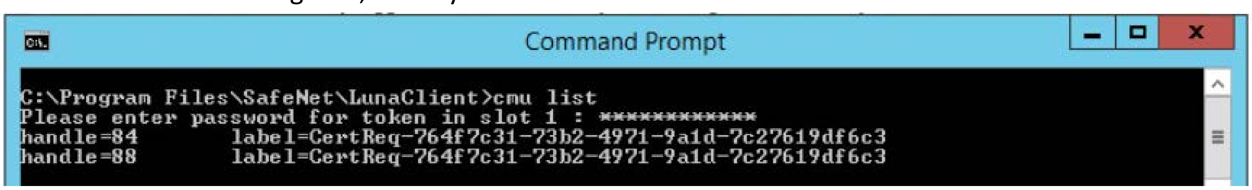
- The NTL is established between the Luna Client and the Luna HSM. If not,see Section 2.2.2.2.

2.2.2.3.2 Verify the HSM Configuration

Verify the HSM client configuration prior to proceeding by following the steps below:

1. Open a Command Prompt as Administrator, and change into the Luna Client directory, typically *C:\Program Files\SafeNet\LunaClient*.
2. Execute the command `VTL.exe verify` to check that the client is configured correctly and the partition is visible. Slot/Partition information should be displayed in response.

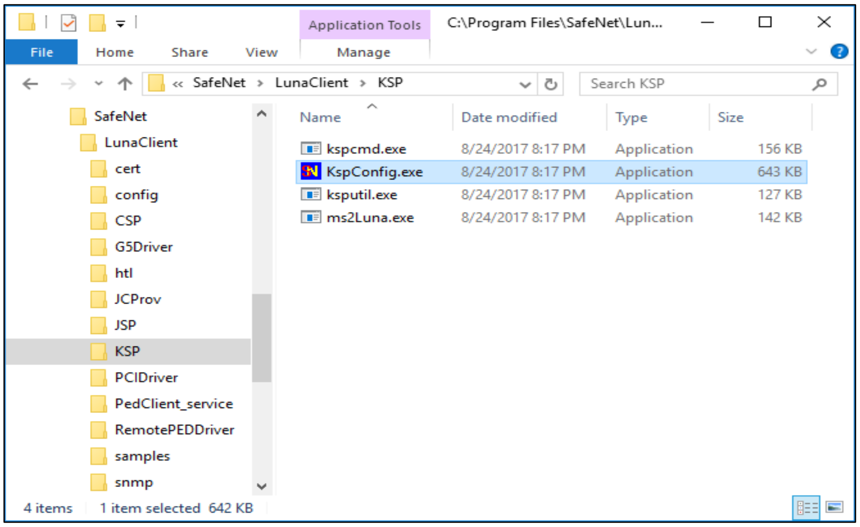
3. Execute the command `cmu list` to see the list of current objects on the HSM, and enter the password when prompted. If nothing has been created on the partition, this list will be blank. Once the CA is configured, the keys created on the HSM are listed.



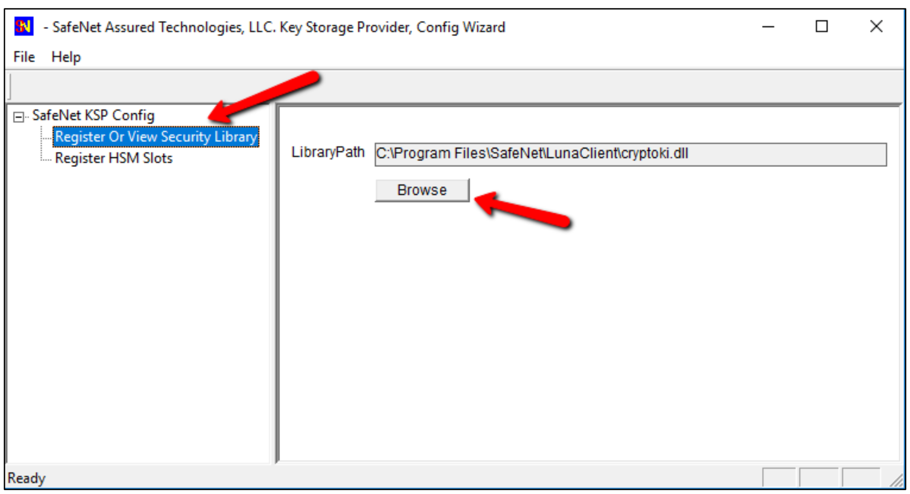
2.2.2.3.3 Register the Key Storage Provider

Beginning with Windows Server 2008, the older CryptoAPI CSP has been superseded by the newer CNGKSP. The Luna Client installation includes a utility to register the SafeNet AT HSM for Government as a KSP for use in Windows applications. To register, follow these instructions:

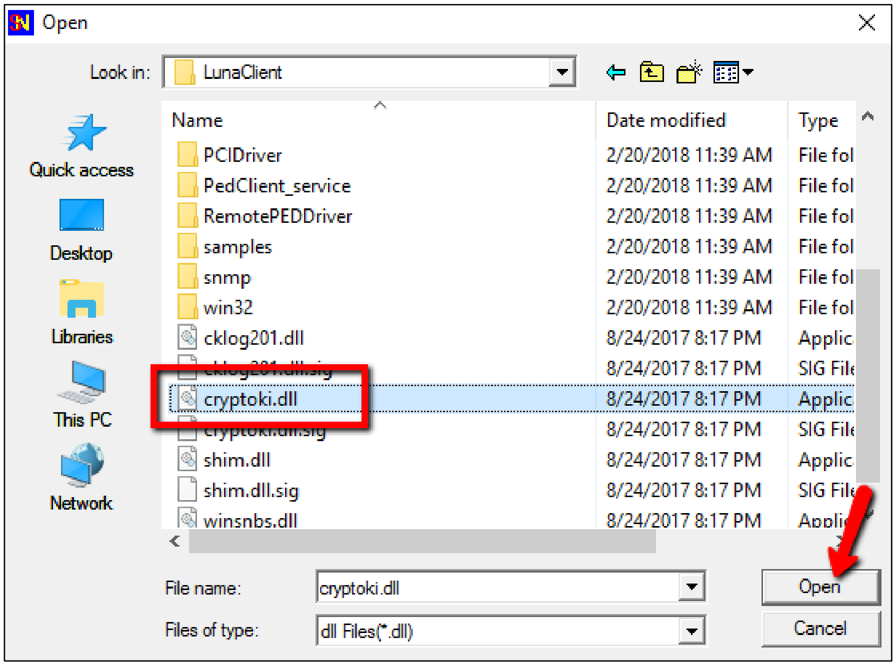
1. Open Windows Explorer, browse to the KSP folder in the Luna Client installation folder, and double-click on the **KSPConfig.exe** utility.



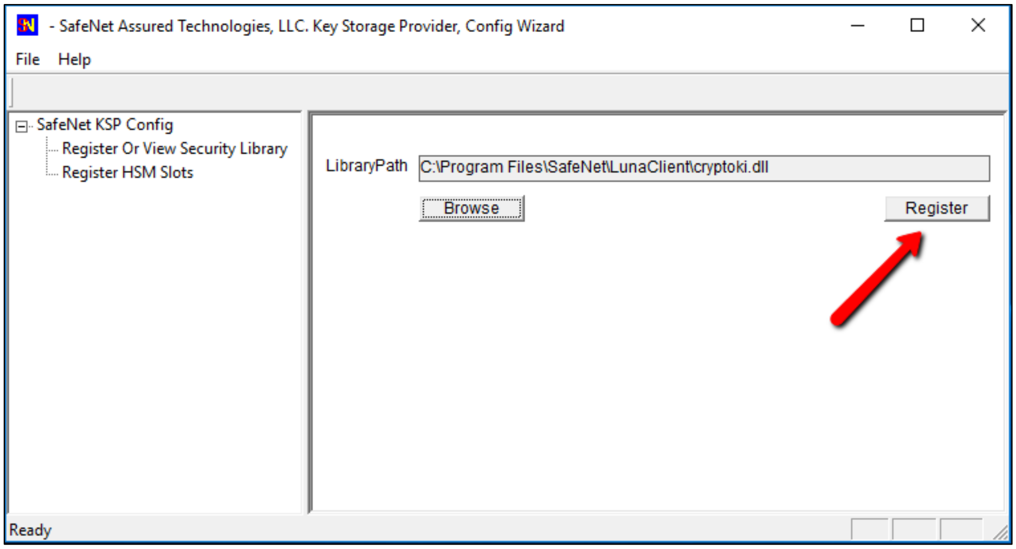
2. Double-click on **Register Or View Security Library**, then click **Browse.**



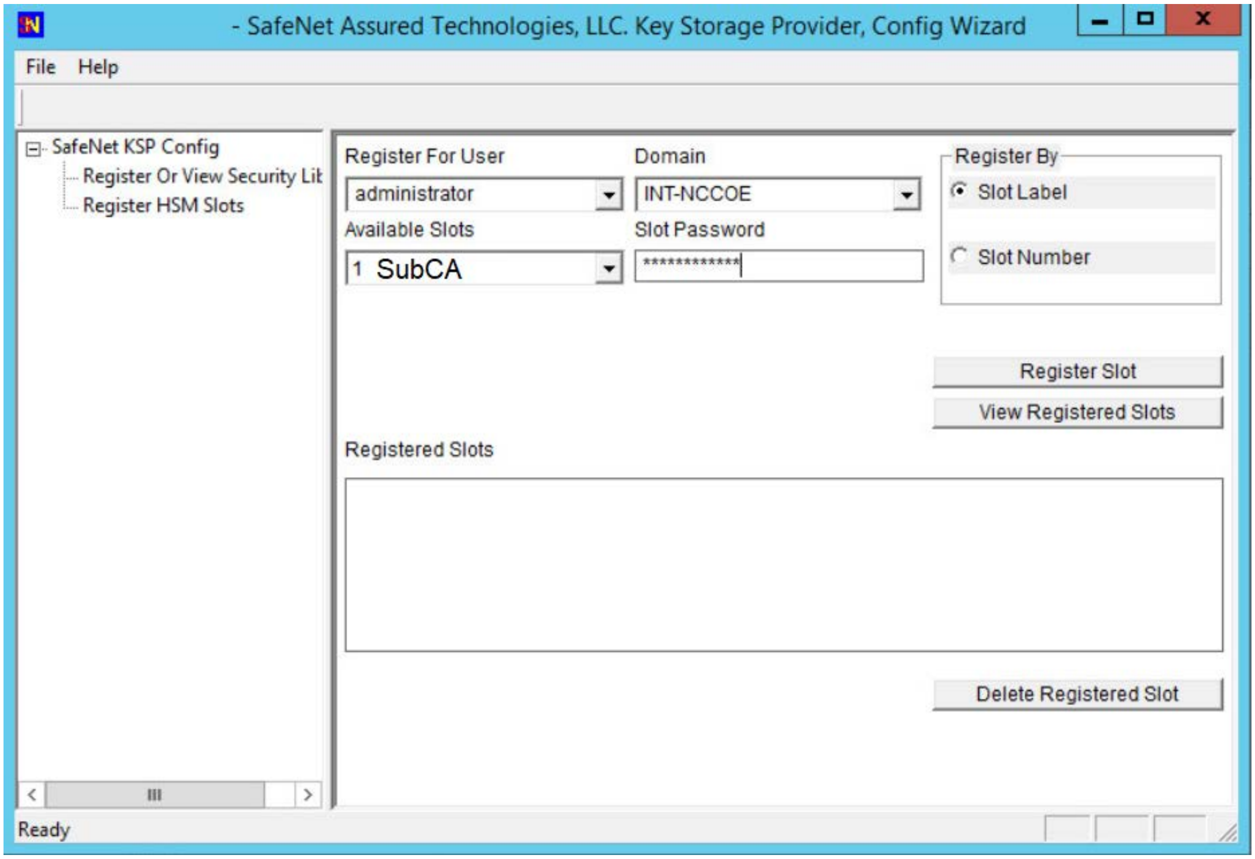
3. Browse to the Luna Client folder, select **cryptoki.dll,** and click **Open.**


Open
Look in:
LunaClient
Name
Date modified
Type
Quick access
Desktop
Libraries
This PC
Network
PCIDriver
PedClient_service
RemotePEDDriver
samples
snmp
win32
cklog201.dll
cryptoki.dll
shim.dll
shim.dll.sig
winsnbs.dll
2/20/2018 11:39 AM
8/24/2017 8:17 PM
File fol
Applic
SIG File
File name:
cryptoki.dll
Files of type:
dll Files(*.dll)
Open
Cancel

4. Click on **Register** to complete the library registration.



5. Double-click **Register HSM Slots** on the left to open the slot registration page. Select the **Administrator** account and the Domain for the user that will be configuring the CA role. For a server joined to a domain, this should be a Domain or Enterprise Admin account rather than the local machine Administrator. Select the slot for the HSM, enter the **Slot Password,** and click **Register Slot.**



6. Repeat the slot registration for the user **SYSTEM** with Domain **NT AUTHORITY,** and click **Register.** This is the account used for the CA service—it must also have access to the HSM. Verify the registration by selecting user and domain and clicking **View Registered Slots.**

2.2.2.3.4 Add CA Role

For instructions on CA installation and configuration, refer to Section 1.5.3.3.2 on root CAs.

2.2.2.3.5 Verify the Successful Integration on the HSM

As a final step, verify the private key and the public key are stored on the HSM.

1. Open a command prompt and change to the Luna Client directory, typically C:\Program Files\SafeNet\LunaClient\.
2. Run **cmu list** to verify the private and public keys for the CA are present on the HSM. They are represented by two "handles."

The screenshot below shows running the `cmu list` command before configuring the CA and then after the configuration has been completed.

This completes integration of the SafeNet AT Luna SA 1700 HSM for Government with Microsoft Active Directory Certificate Services.

2.2.2.4 IIS Integration Configuration

This section provides the steps necessary to integrate the Microsoft IIS web server and the SafeNet AT Luna SA 1700 HSM. The benefit of the integration is that the root private key for IIS is stored in a hardened, FIPS 140-2-certified device.

The following steps explain how to register the SafeNet AT Luna SA 1700 HSM as a KSP to store the root certificate’s private key in the HSM.

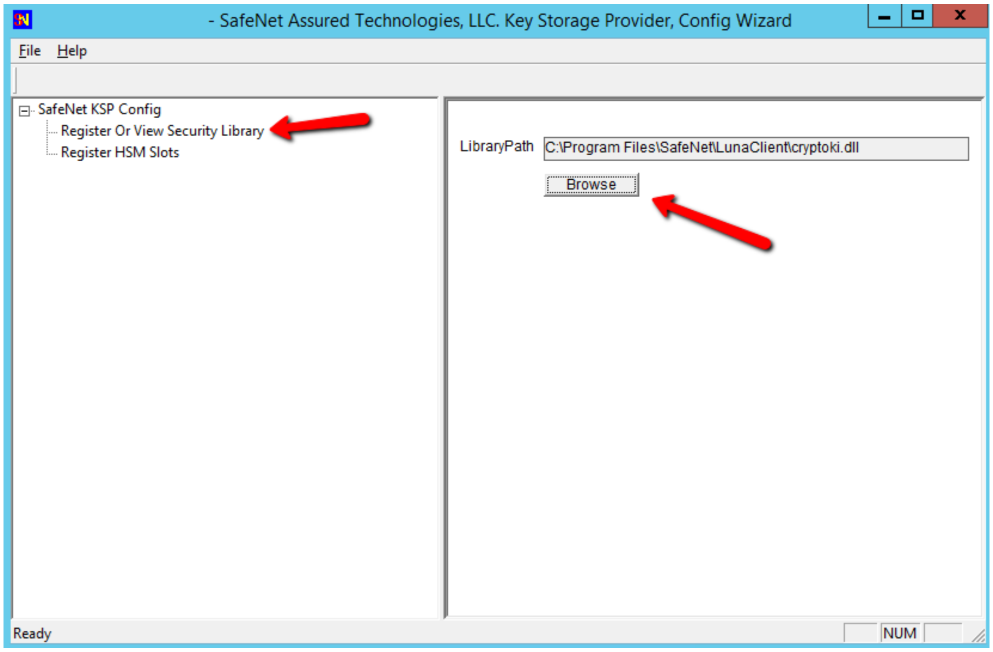
2.2.2.4.1 Prerequisites

- IIS is installed or ready to be installed. The firewall rules may need to be edited to allow https access (typically port 443) and optionally block http (port 80).
- If mutual authentication is being performed, the trusted CA’s certificate has been installed.

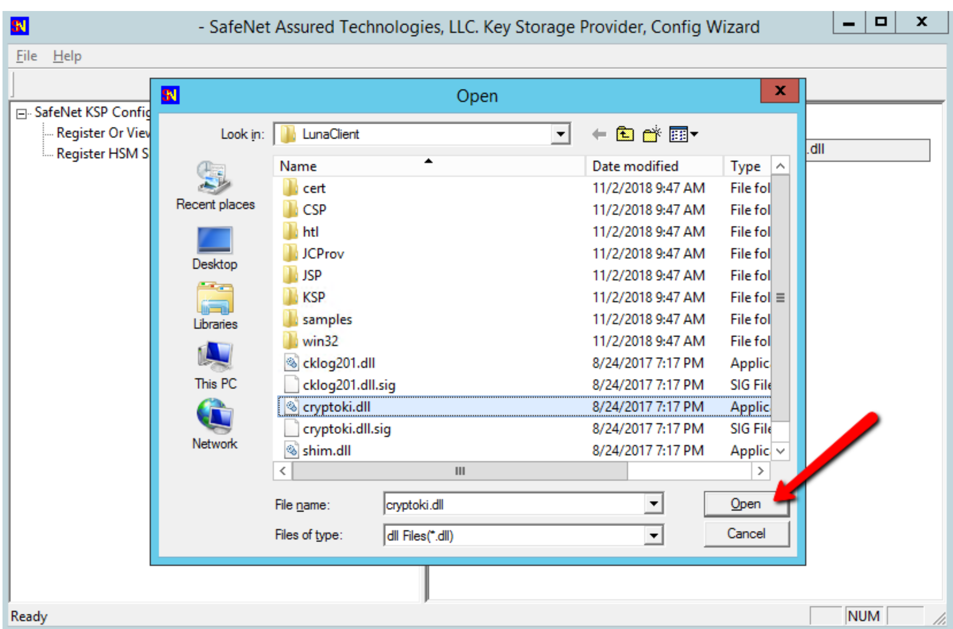
2.2.2.4.2 Register the Luna KSP

For IIS integration, two accounts need access to the HSM. First, the DOMAIN\Administrator account is used for setting up the server—creating the certificate request and installing the certificate. Second, the NT Authority\System account is used by the server to start the IIS service. The **KSPConfig** utility is used to register the HSM as a KSP for these accounts.

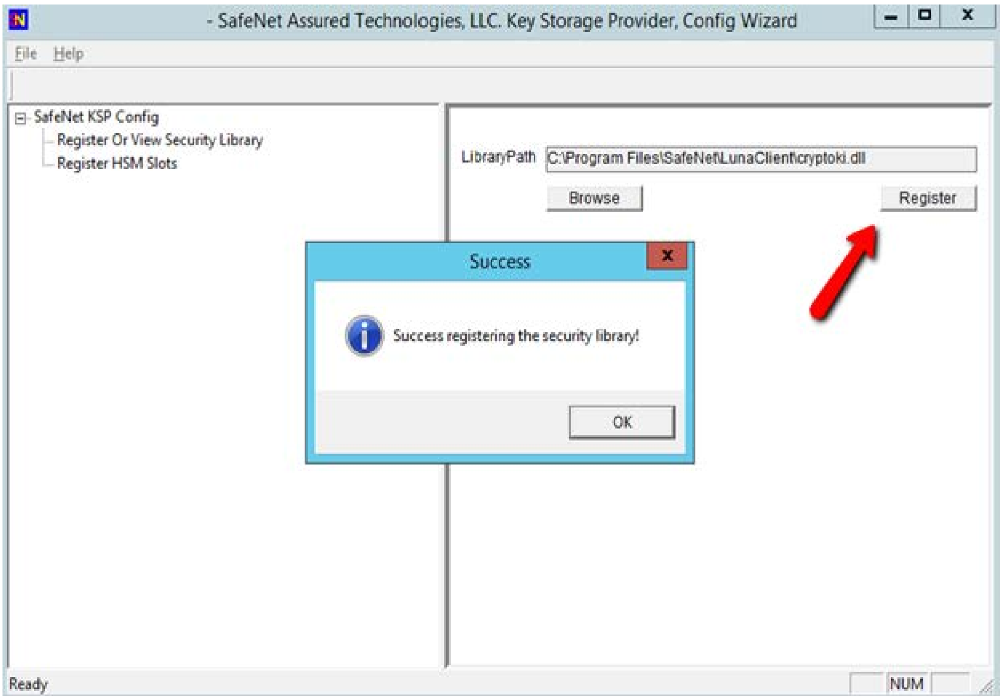
1. Navigate to the **KSP** directory under the Luna installation directory, which is typically *C:\ProgramFiles\SafeNet\LunaClient.*
2. Run **KspConfig.exe** to launch the wizard.
3. When the wizard launches, double-click **Register Or View Security Library** on the left side of the pane, and then click the **Browse** button on the right.



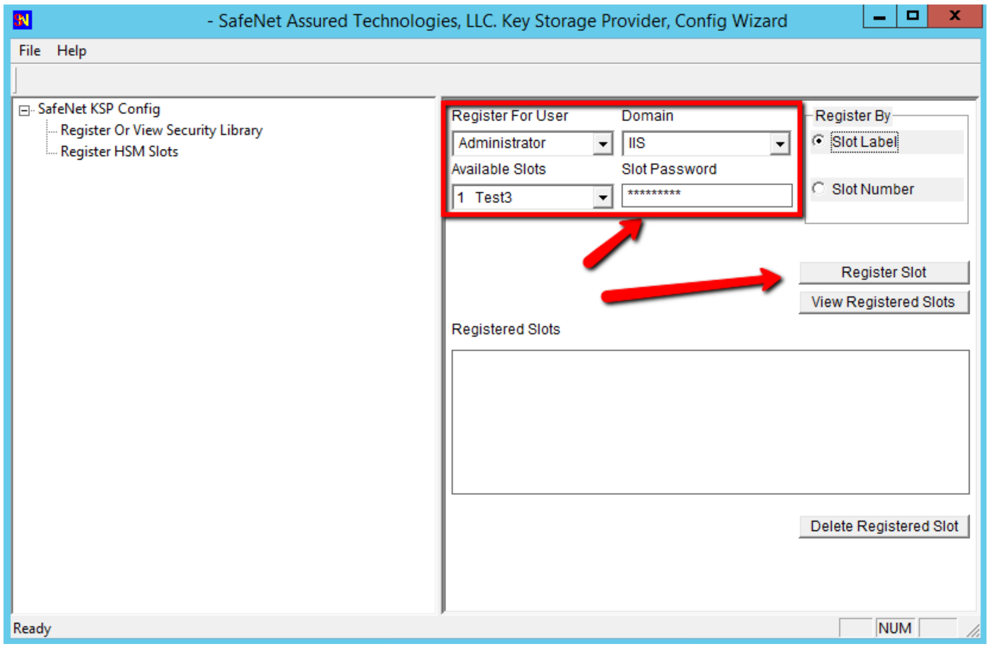
4. Browse to and select the **cryptoki.dll** library in the Luna Client directory.



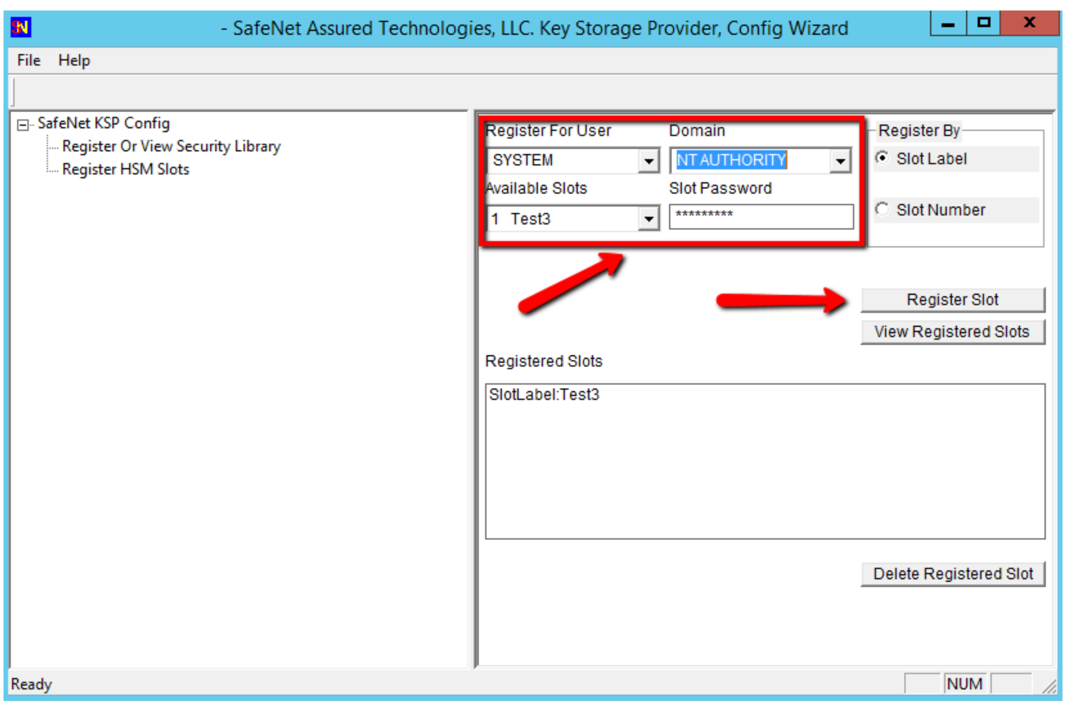
5. Having selected the dll, click the **Register** button. The message **"Success registering the security library!"** displays.



6. Double-click **Register HSM Slots** on the left side of the pane.
7. Verify the correct **User** and **Domain** are selected (the Administrator account on the server) and slot is selected (can be registered by slot label or slot number), and enter the **Slot Password** (HSM partition password).
8. Click **Register Slot** to register the slot for that User/Domain. Upon successful registration, a message **"The slot was successfully and securely registered"** displays.



9. Repeat the steps above to register the slot for the **User SYSTEM** and **Domain NT AUTHORITY.**



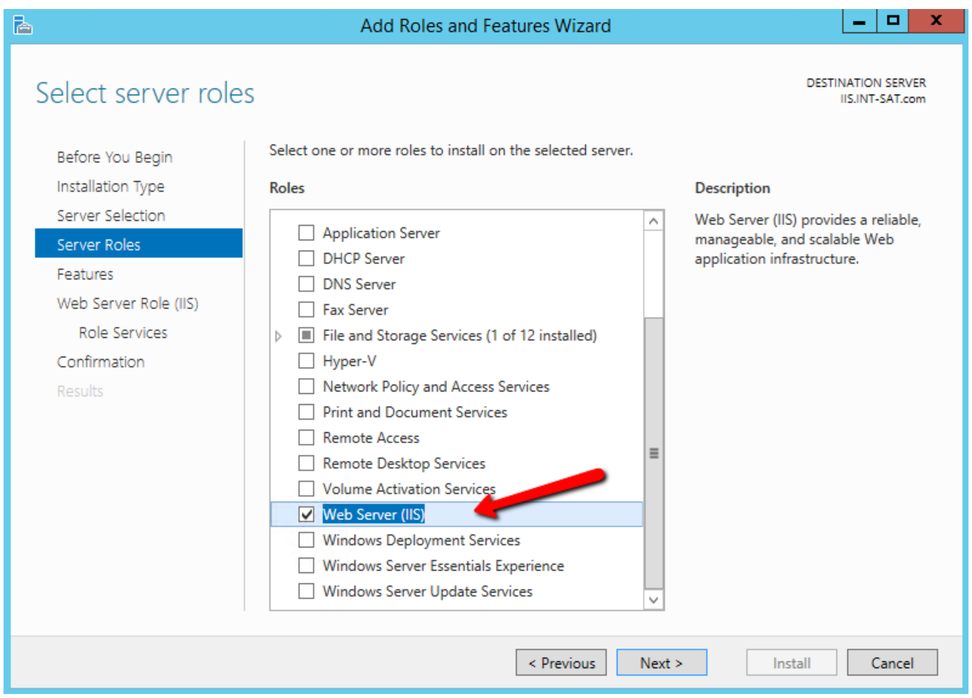
To verify the registered slot, select a **User/Domain,** and click the **View Registered Slots** button.

2.2.2.4.3 Setup Synopsis

- Verify the NTL between the server and the HSM.
- Register the HSM as a KSP.
- Install IIS and configure it to use an HSM.
- Create a certificate request for IIS, and get it signed.
- Install the signed certificate.
- Bind the certificate to the web server.

2.2.2.4.4 Install Microsoft IIS

The next step is to install the **Web Server (IIS)** role by using **Server Manager.** There are no special considerations surrounding the IIS integration with an HSM. Please follow the installation and configuration steps in Section 1.5.5.2.



2.2.2.4.5 Create and Install a Certificate for IIS

IIS will need a certificate installed that has been signed by a trusted CA. This involves creating a certification signing request (CSR), then the CA signs it and installs it back in the server. **IIS Manager** provides an easy way for creating a CSR, but it cannot be used when a key is generated on an external HSM. Instead, use a Microsoft command line utility.

Clients attempting to securely connect to the web server will see an alert if the fully qualified domain name (FQDN) in the Common Name (CN) field (or on more recent browsers, the FQDN in the Subject Alternate Name field) does not match the uniform resource locator (URL) they are accessing. An alert also occurs if the certificate was not issued by a trusted root CA. For this integration, use the FQDN in the CN and Subject Alternative Name (SAN) fields.

2.2.2.4.6 Create a Certificate Signing Request and Private Key

Instructions follow for using the **certreq.exe** utility to create the CSR and private key in the HSM.

1. Create a file called ***request.inf*** that will contain the necessary information for the utility to create the CSR. The contents of the file are as follows—only those items in blue italics will vary per the organization's environment and requirements. The **CN** in the subject and the **dns** name in the **SAN** extension must match the full host name that clients enter as the URL in a web browser.

Copying and pasting the text may insert line breaks or change quotation marks to smart (curly) quotation marks. Ensure that each entry is on a single line and that all quotation marks are standard, straight, and double.

In this document, some entries may appear with line breaks such as the **Subject=…** and **%szOID_ENHANCED_KEY_USAGE…** lines, but they must be on a single line. In addition, if using Notepad, change the file type to "all files" so it does not create the file with an extension of .txt. The "hide extensions for known file types" option may need to be disabled in Windows Explorer to verify the file is an *.inf* file rather than a *.txt* file. The text of the *.inf* file follows, as well as an image of the how the file should look.

```
 [Version]
      Signature= "$Windows NT$"

      [NewRequest]
      Subject = "C=US,CN=HRhsm.int-
      nccoe.org,O=SafeNetAT,OU=TLSLAB,L=Gaithersburg,S=Maryland"
      HashAlgorithm = SHA256
      KeyAlgorithm = RSA
      KeyLength = 2048
      ProviderName = "Safenet Key Storage Provider"
      KeyUsage = 0xf0
      MachineKeySet = True
      [EnhancedKeyUsageExtension]
      OID=1.3.6.1.5.5.7.3.1

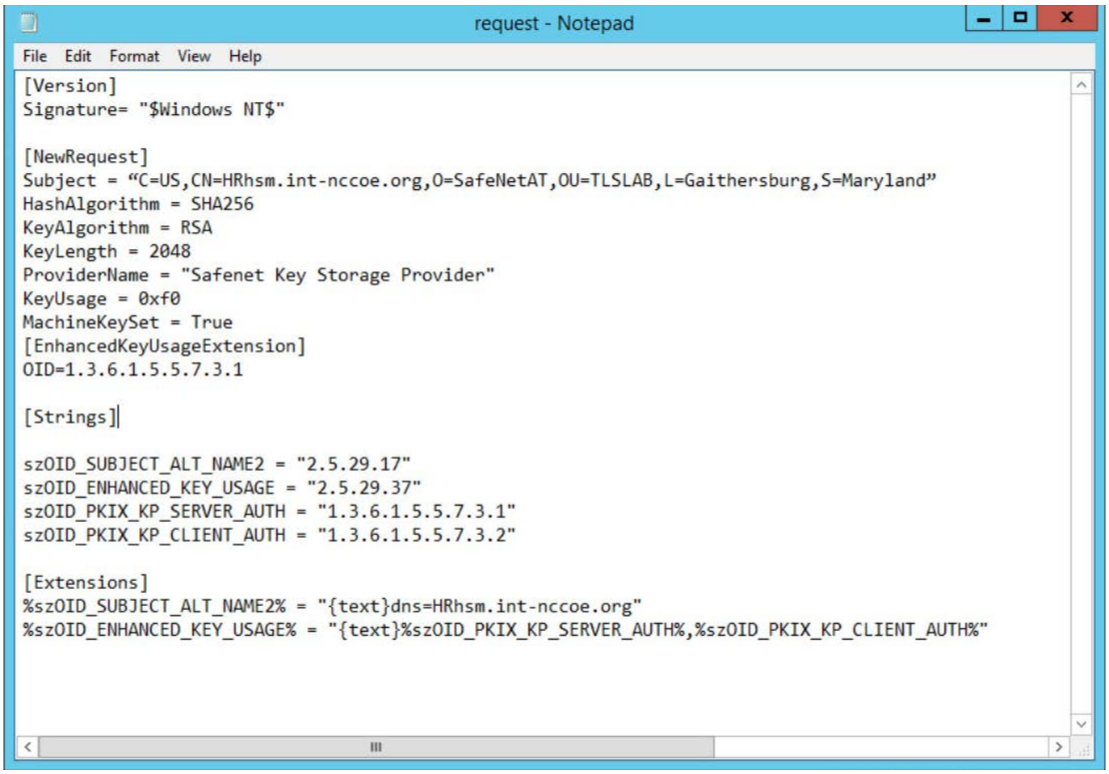
[Strings]

szOID_SUBJECT_ALT_NAME2 = "2.5.29.17"
      szOID_ENHANCED_KEY_USAGE = "2.5.29.37"
```

```
        szOID_PKIX_KP_SERVER_AUTH = "1.3.6.1.5.5.7.3.1" szOID_PKIX_KP_CLIENT_AUTH =
        "1.3.6.1.5.5.7.3.2"

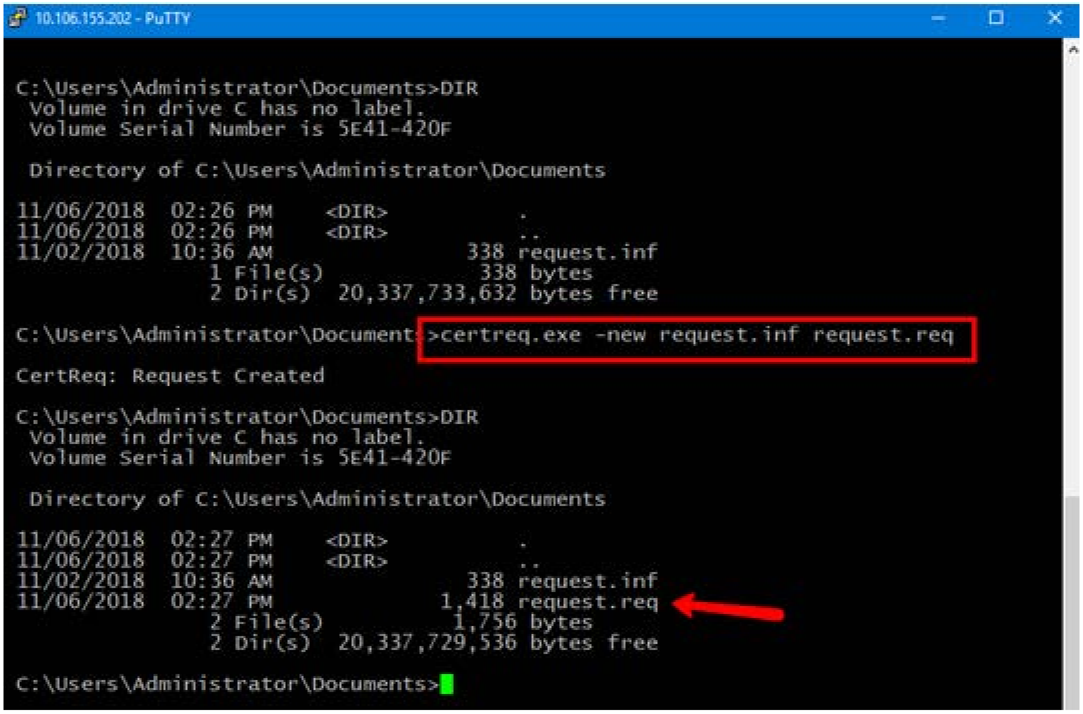
[Extensions]
        %szOID_SUBJECT_ALT_NAME2% = "{text}dns=HRhsm.int-nccoe.org"
        %szOID_ENHANCED_KEY_USAGE% =
        "{text}%szOID_PKIX_KP_SERVER_AUTH%,%szOID_PKIX_KP_CLIENT_AUTH%"
```

Example image of file with correct line breaks:



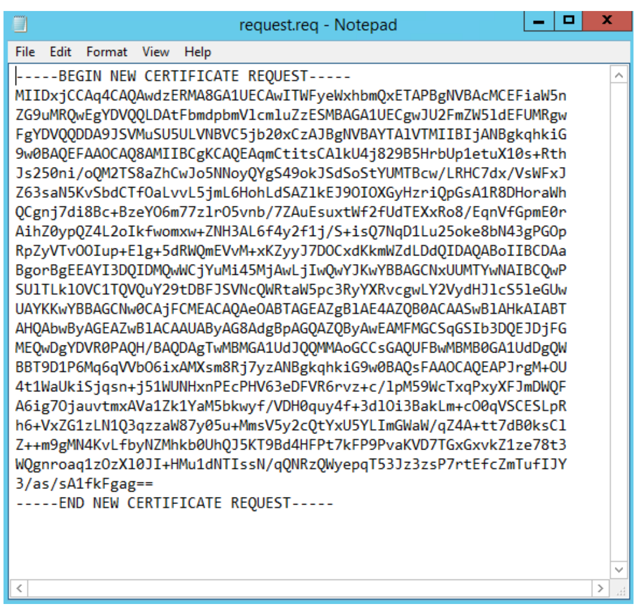
2. With the information file created, execute the **certreq** utility to generate a key on the HSM, and the certificate request. The CSR will output to the file name that the user provides.

```
certreq.exe –new request.inf <CSR_filename>
```



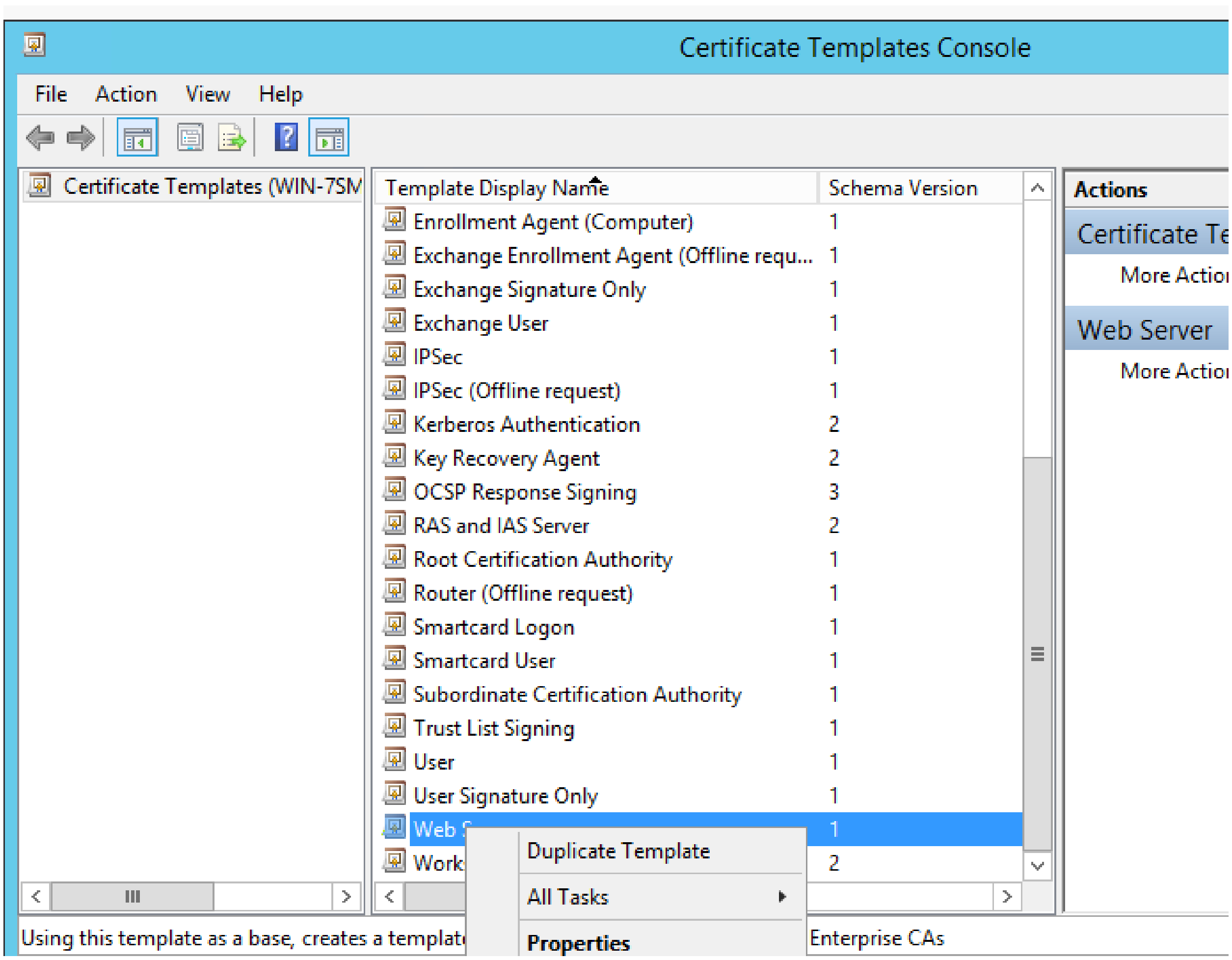
2.2.2.4.7 Get the CSR Signed by a Trusted CA

A trusted CA must sign the generated CSR (example below). The CA authenticates the request and returns a signed certificate or a certificate chain. When the certificate file is received back, save it in the current working directory.

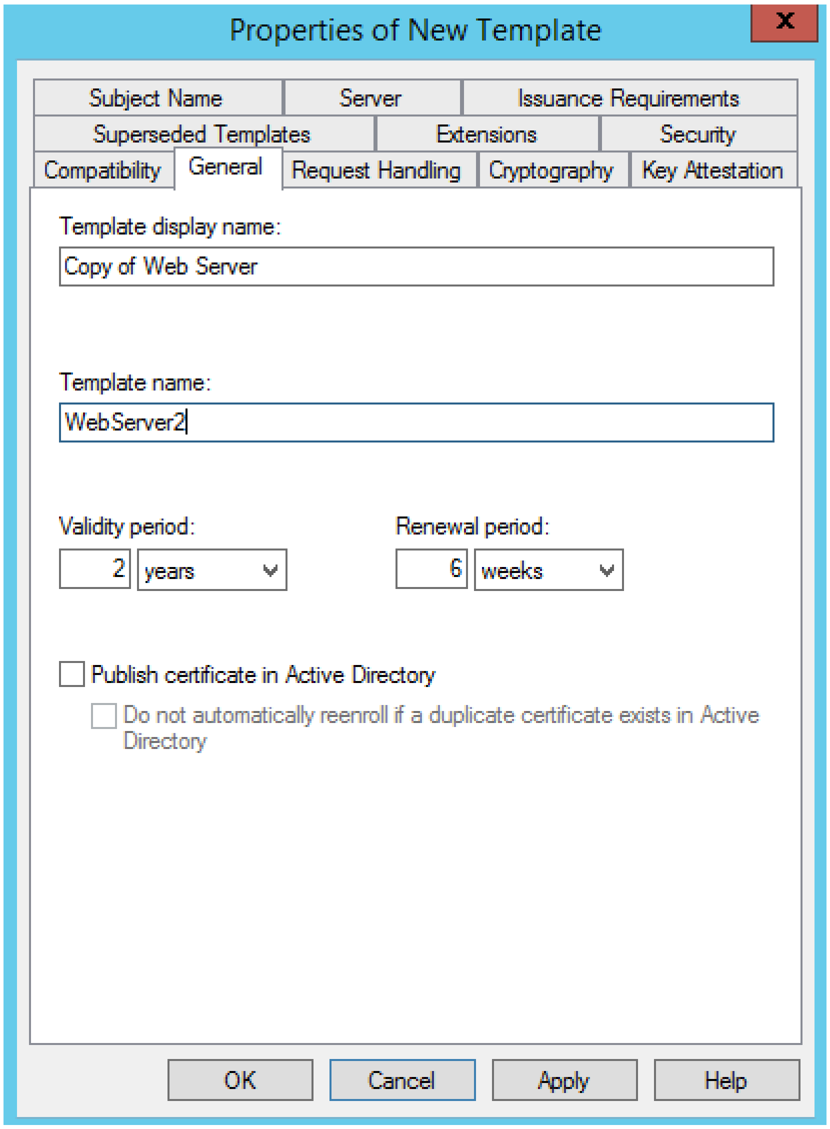


The CSR was signed by using an Enterprise CA. Follow the steps below to create a new template and to sign the certificate request:

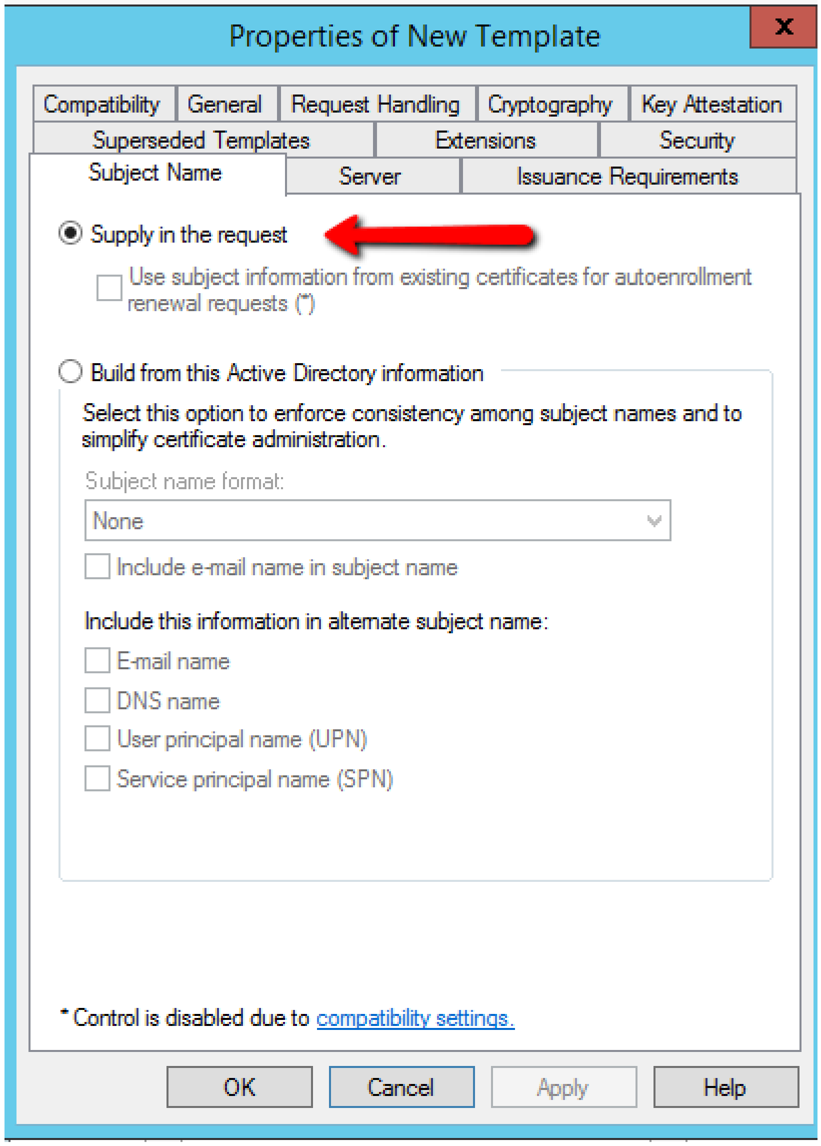
1. Search for and run **certsrv.msc,** or from Server Manager select **Tools > Certification Authority** to view the CA. Expand the CA > right-click **Certificate Templates** > select **Manage.**
2. In the **Certificate Templates Console,** scroll down to find the **Web Server** template and right-click > select **Duplicate Template.**



3. Fill out the various sections of the properties with settings that adhere to the company's security policies. For this guide, the only thing altered is the **Template name** in the **General** tab. This will be the name used when signing the request on the command line.

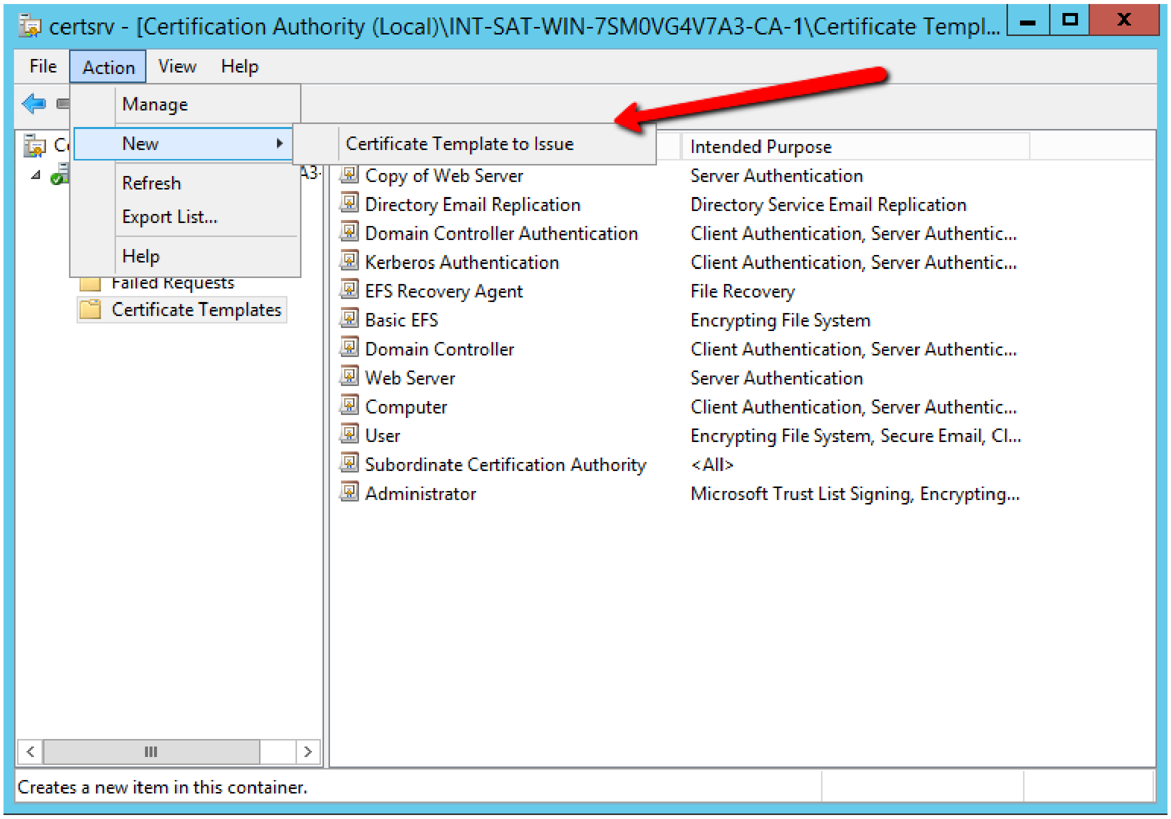


4. Select the **Subject Name** tab, and verify that **Supply in the request** is selected. The FQDN is specified in both the CN and SAN fields in the request file created, and the certificate will use these values.

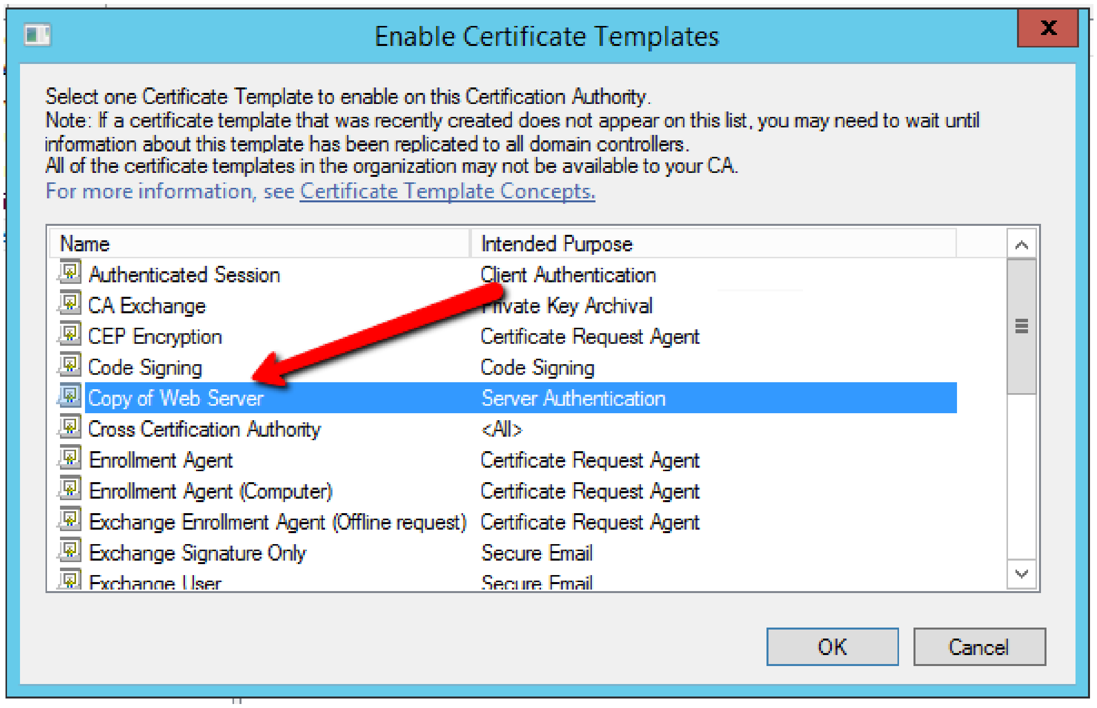


5. Click **OK** to finish creating the new template.
6. Close the **Certificate Templates Console >** return to the **Certificate Authority window.**

7. Click on **Action > New > Certificate Template to Issue**

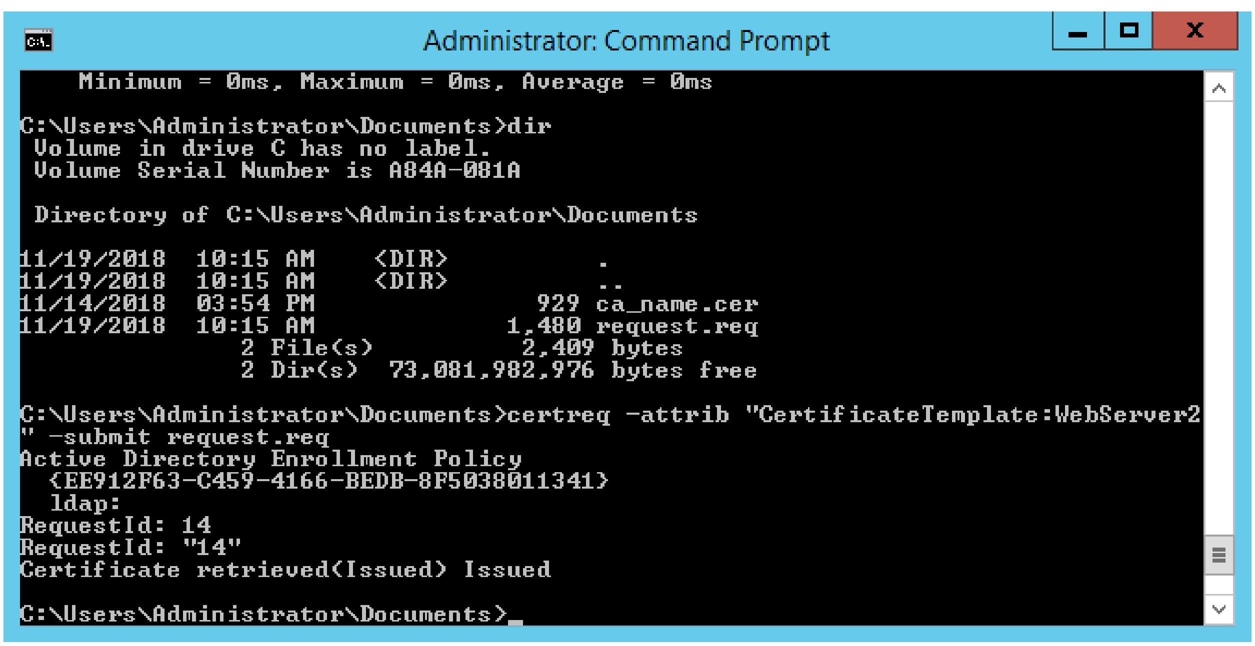


8. Select the certificate template created > click **OK.**



9. Generate a certificate from the certificate request:

```
certreq -attrib "CertificateTemplate:<TemplateName>" -submit <certificate
request filename>
```

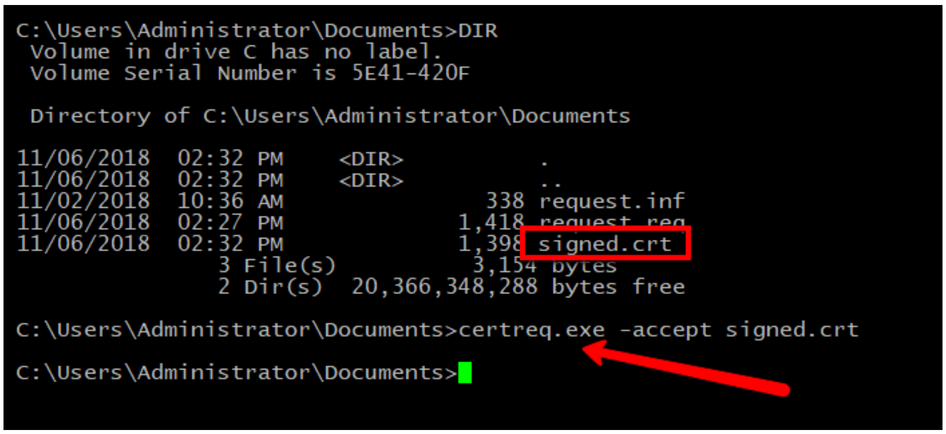


The user will be prompted to select the CA to use for signing, and a location and file name to save the signed certificate. Once the signed certificate file is created, it can be copied to the IIS server to continue with the integration.

2.2.2.4.8 Install the Signed Certificate

Once the CSR is signed and the signed certificate file is received back, accept and install it by using the **certreq** utility.

```
certreq.exe –accept <newcert.crt>
```

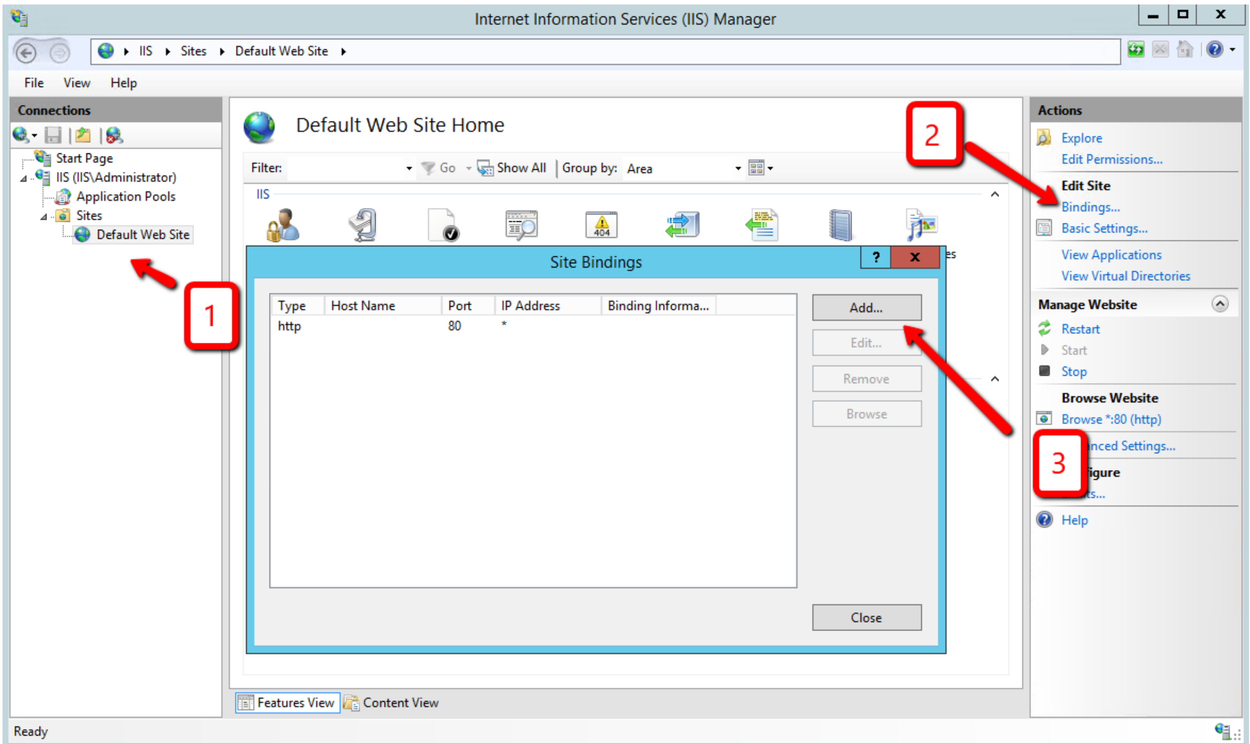


If this step fails, the most common cause is that the issuing CA root certificate is not installed in the server's certificate store. Verify the issuing CA is trusted, or install the CA certificate into the Local Machine—Trusted Root CA certificate store.

2.2.2.4.9 Bind the Certificate to the IIS Web Server

The final step is to bind the certificate to the IIS web server:

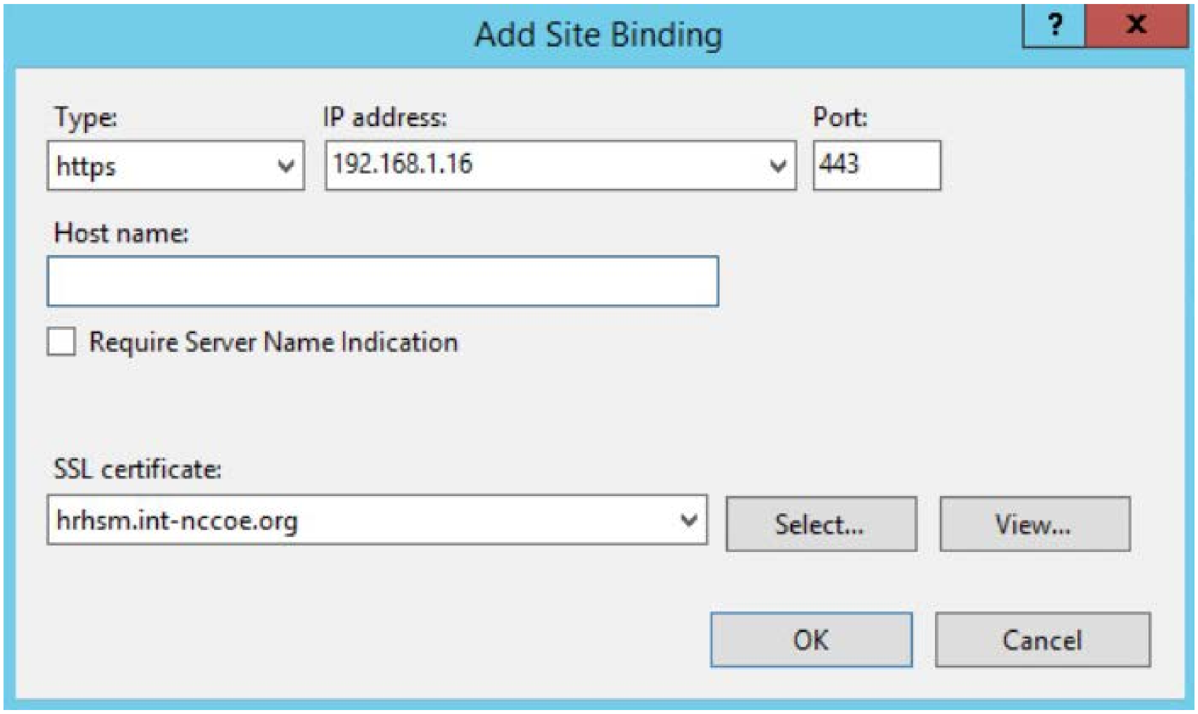
1. Open the **IIS Manager** from **Start > Administrative Tools > Internet Information Services (IIS) Manager.**
2. Under **Sites** on the left side of the IIS Manager window, select the desired website.
3. On the right side of the IIS Manager, click **Bindings.**
4. In the **Site Bindings** window, click **Add.**



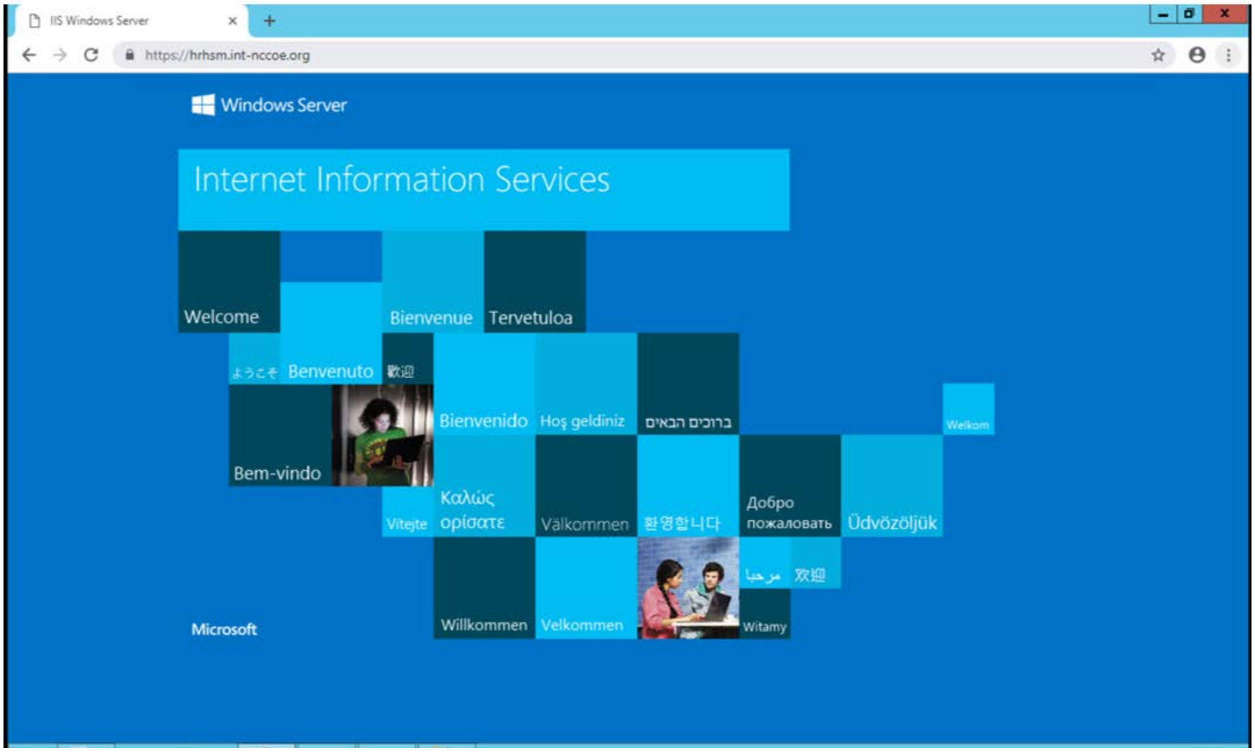
5. Select the protocol as **https.**

6. Select the IP address of the machine running IIS from the **IP Address** drop-down list, or leave blank to use all available network interfaces.

7. Enter port **443.**



8. In the **SSL certificate:** drop-down, select the certificate that was just installed.
9. Complete the certificate binding in support of SSL/TLS, then click **OK.**
10. Verify the connection is working, open a browser, and enter your URL (e.g., *https://hrhsm.int-nccoe.org:443*). There may be a prompt to accept the certificate for the site. The host name must match the name used in the certificate request and must be registered with the DNS server to resolve the host name to the IP address of the IIS server.



2.2.2.5 Venafi Integration Configuration

This section covers the necessary information to integrate Venafi with the SafeNet AT Luna SA 1700 for Government HSM. When integrated with the Luna, Venafi can create and store the master encryption key used to encrypt and decrypt the Venafi database. In this configuration, the Venafi TPP services will not start unless the key stored in the HSM is accessible. This provides an additional hardened layer of security to protect data in the database.

2.2.2.5.1 Prerequisites

To integrate Venafi with the Luna SA HSM, the following prerequisites must be met:

- The SafeNet AT Luna HSM is installed and operational.
- The SafeNet AT Luna Client is installed on the Venafi server.
- The NTL is established between the Luna Client and the Luna HSM as described in Section 2.2.2.2.9.
- The NTL between the Venafi server and the HSM has been verified.
- Venafi has been configured to use the Luna SA HSM.
- The master encryption key was created on the Luna SA HSM and has been verified.

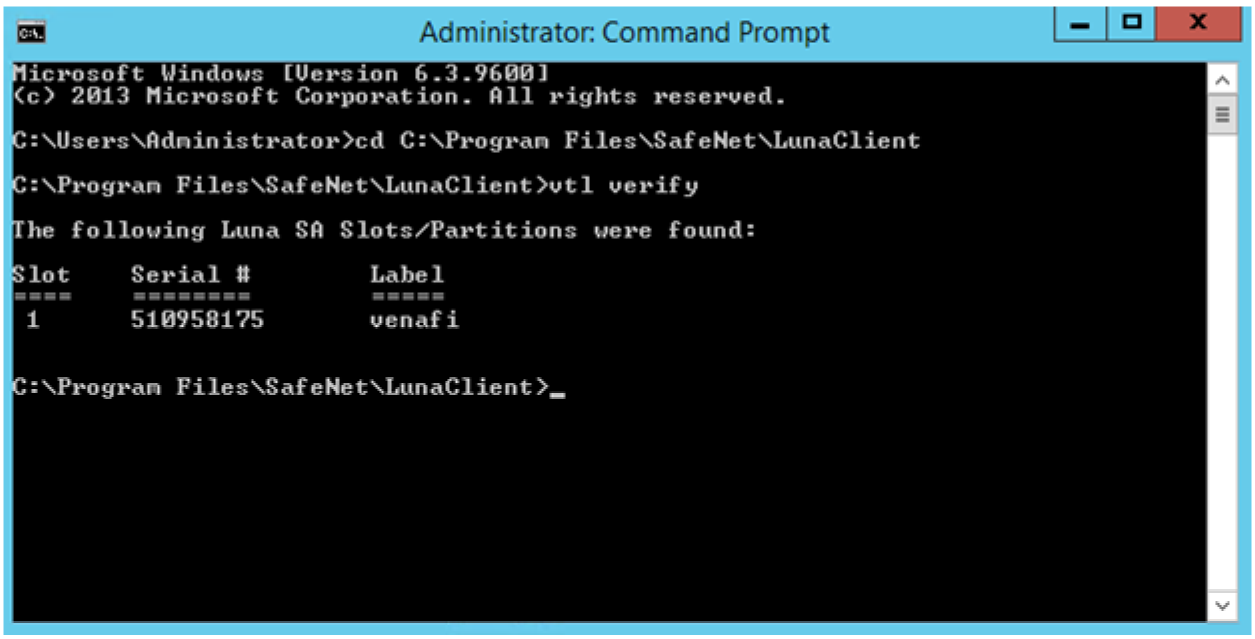
2.2.2.5.2 Verify the Network Trust Link Between Venafi and the HSM

The Luna Client installed on the server enables communication between Venafi and the HSM via a secure connection or an NTL. If the NTL has not been set up during HSM/client installation, reference Section 2.2.2.2 of this guide.

Use the `vtl verify` command in the installed client directory (typically *C:\Program Files\SafeNet\LunaClient*) to determine if the connection was established and that a partition exists on the HSM that the client can access. If no slot and partition are found, the NTL is not established.

The slot number and partition password will be needed when configuring Venafi to use the HSM.

```
vtl verify
```



For further configuration between the HSM and Venafi TPP, please reference Section 2.6.13.3.

2.2.3 Day N: Ongoing Security Management and Maintenance

2.2.3.1 Prerequisites

- remote system logging server

2.2.3.2 Remote System Logging

Refer to the Luna SA syslog commands to use the remote system logging on any UNIX/Linux system that supports the standard syslog service. Refer to the Luna SA syslog commands under “syslog remotehost” (subcommands “add,” “delete,” and “list”) for more information. The remote host must have User

Datagram Protocol (UDP) port 514 open to receive the logging. Refer to the host's OS and firewall documentation for more information.

1. Type the command below on the Luna SA appliance:

   ```
   lunash:>syslog remotehost add 192.168.1.12
   ```

2. Start syslog with the "-r" option on the receiving or target system to allow it to receive the logs from the Luna SA appliance(s).

2.2.3.3 Audit Logging

With Luna SA, the audit logs can be sent to one or more remote logging servers. Either UDP or Transmission Control Protocol (TCP) protocol can be specified. The default is UDP and port 514.

2.2.3.3.1 UDP Logging

If using UDP protocol for logging:

- The following is required in /etc/rsyslog.conf

  $ModLoad imudp

  $InputUDPServerRun (PORT)

- Possible approaches include:

  1. With templates:

     $template AuditFile,"/var/log/luna/audit_remote.log"

     $syslogfacility-text == 'local3' then ?AuditFile;AuditFormat

  2. Without templates:

     local3.* /var/log/audit.log;AuditFormat

  3. Dynamic file name:

     $template DynFile,"/var/log/luna/%HOSTNAME%.log"

     if $syslogfacility-text == 'local3' then ?DynFile;AuditFormat

- The important thing to remember is that the incoming logs go to local3, and the Port/Protocol that is set on the Luna appliance must be the same that is set on the server running rsyslog.

2.2.3.3.2 TCP Logging

Here is an example to set up a remote Linux system to receive the audit logs by using TCP.

- Register the remote Linux system IP address or host name with the Luna SA:

```
lunash:> audit remotehost add -host 172.20.9.160 -protocol tcp -port 1660
```

2.3 DigiCert Certificate Authority

2.3.1 Day 0: Installation and Standard Configuration

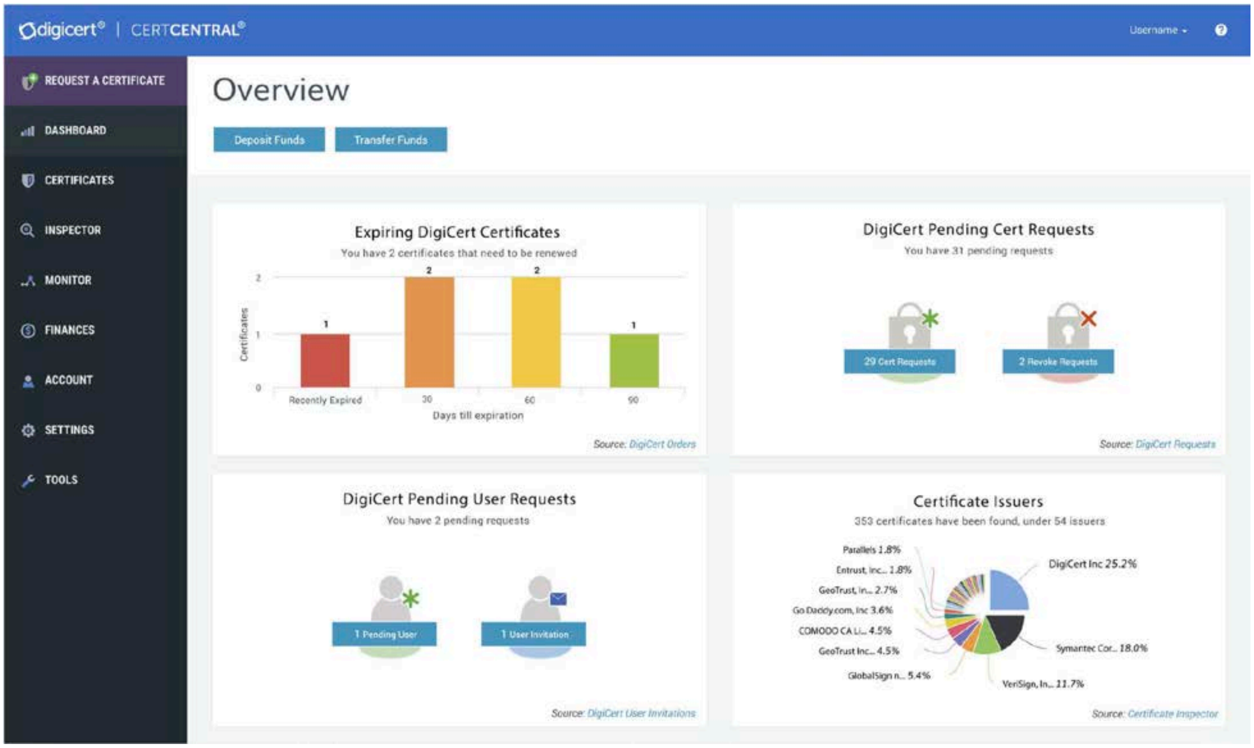
2.3.1.1 Certificate Prerequisites for Domain Validation and Organization Validation

- organization validation–can be an individual or group/team
- domain validation process–DNS text (TXT) record validation
- must have resolvable FQDN entered in zone file *(tls.nccoe.org, app1.tls.nccoe.org)*
- access to DigiCert’s web-based registration system
- account sign-up

2.3.1.2 Standard Configuration

2.3.1.2.1 Account Sign-Up

1. Start the account sign-up process at https://www.digicert.com/account/signup/.
2. Complete the **Your information, Organization information,** and **Account information** sections.
3. Read and accept the terms of the Certificate Services Agreement. Check the box to acknowledge acceptance of the terms.
4. Click the **Sign Up** button to create a CertCentral account.



2.3.1.2.2 Language Preferences

Currently, CertCentral supports the following languages:

- Deutsch
- English
- Español
- Français
- Italiano
- Português
- 한국어
- 日本語
- 简体中文
- 繁體中文

1. To change the language in the CertCentral account, click the account name at the upper-right side of the screen and select **My Profile** from the drop-down list.

2. On the Profile Settings page in the **Language** drop-down list, select the language preference for the account.
3. Click **Save Changes.** The language in CertCentral should now be the same as the one selected.

2.3.1.2.3 Billing Contact

To edit the assigned Billing Contact in the CertCentral account:

1. In the sidebar menu, click **Finances > Settings.**
2. On the Finance Settings page, click **Edit** under **Billing Contact** in the right column.
3. In the **Edit Billing Contact** window, set or change the contact information.
4. Click **Update Billing Contact** to save the change.

2.3.1.2.4 Authentication Settings

Authentication settings allow control over the user login options for the CertCentral account and to set security standards for password requirements and alternative authentication methods.

To access the CertCentral authentication options:

1. In the CertCentral account in the sidebar menu, click **Settings > Authentication Settings.** On this page, the following settings can be changed:
    - Minimum Length: Change the minimum allowed password character length.
    - Minimum Categories: Change the variety of characters allowed (uppercase, lowercase, numbers, and symbols).
    - Expires After: Change the password expiration policy.
    - Two-Factor Authentication: Enable or disable onetime password two-factor authentication for CertCentral users.
2. Configure the authentication settings as desired, then click **Save Settings.**

2.3.1.2.5 Security Assertion Markup Language (SAML) Single Sign-On Prerequisites

SAML is a highly recommended DigiCert feature for secure user authentication. However, it is not required to duplicate the TLS lab setup. For more information on SAML, please refer to guidance at:

- https://pages.nist.gov/800-63-3/sp800-63-3.html

Before beginning, make sure the following prerequisites are met:

- Have a CertCentral account.
- Have SAML enabled on the CertCentral account. (To get the SAML features turned on for the CertCentral account, contact the DigiCert account representative or the DigiCert support team. Once activated, in the sidebar menu, under Settings, see the Single Sign-On and SAML Certificate Request menu options.)

- Have an identity provider (IdP).
- Have the IdP metadata (dynamic or static).
- Have admin privileges on the CertCentral account (or have manager privileges on the CertCentral account with the Allow access to SAML settings permission).

2.3.1.2.6 Organization Validation

To validate an organization, DigiCert firsts verifies the organization requesting a certificate is in good standing. This may include confirming good standing and active registration in corporate registries. It may also include verifying the organization is not listed in any fraud, phishing, or government-restricted entities and anti-terrorism databases. Additionally, DigiCert verifies the organization requesting a certificate is, in fact, the organization to which the certificate will be issued. DigiCert also verifies the organization contact.

1. In the CertCentral account, using the sidebar menu, click **Certificates > Organizations.**
2. On the **Organizations** page, click **New Organization.**
3. On the **New Organization** page, under **Organization Details,** enter the specified organization information:

Legal Name	Enter the organization's legally registered name.
Assumed Name	If the organization has a doing-business-as name and the name should appear on the certificates, enter the name here. If not, leave this box blank.
Organization Phone Number	Enter a phone number at which the organization can be contacted.
Country	In the drop-down list, select the country where the organization is legally located.
Address 1	Enter the address where the organization is legally located.
Address 2	Enter a second address, if applicable.
City	Enter the city where the organization is legally located.
State/Province/ Territory/Region/ County	Enter the state, province, territory, region, or county where the organization is legally located.
Zip Code/Postal Code	Enter the zip or postal code for the organization's location.

4. Under **Validation Contact,** provide the contact's information:

First Name	Enter the contact's first name.
Last Name	Enter the contact's last name.
Job Title	Enter the contact's job title.
Email	Enter an email address at which the contact can be reached.
Phone Number	Enter a phone number at which the contact can be reached.
Phone Extension	Enter the contact's extension, if applicable.

5. When finished, click **Save Organization.**
   Submit an organization for validation.
6. In the CertCentral account, using the sidebar menu, click **Certificates > Organizations.**
7. On the **Organizations** page, use the drop-down list, search box, and column headers to filter the list of organizations.
8. Click the link for the organization being submitted for validation and authorization for certificates.
9. On the organization's information page in the **Submit Organization for Validation** section, select the validation types (certificates) needed for DigiCert to validate the organization's information below:
   - OV—Normal Organization Validation (Recommended)
   - EV—Extended Organization Validation (EV)
   - Private SSL—DigiCert Private SSL Certificate
   - CS—Code Signing Organization Validation
   - EV CS—Code Signing Organization Extended Validation (EV CS)
   - DS–Document Signing Validation
   - Add verified contact (EV/EV CS, and CS).

   If the organization validation chosen is not OV, refer to https://docs.digicert.com/manage-certificates/organization-domain-management/managing-domains-cc-guide/ for additional details.
10. When finished, click **Submit for Validation.**

2.3.1.2.7 Domain Validation

DigiCert's domain validation process ensures the organization requesting a certificate is authorized to request a certificate for the domain in question. Domain validation can include emails or phone calls to the contacts listed in a domain's WHOIS record as well as emails to default administrative addresses at the domain. For example, DigiCert may send an authorization email to the administrator@domain.com or webmaster@domain.com but would not send an authorization email to tech@domain.com.

Note: To validate a domain by using DNS TXT, see the steps below. To use an alternative method, refer to **Error! Hyperlink reference not valid.**https://docs.digicert.com/manage-certificates/organization-domain-management/managing-domains-cc-guide/.

Step I: Add and Authorize a Domain for TLS/SSL Certificates

1. In the CertCentral account in the sidebar menu, click **Certificates > Domains.**
2. On the **Domains** page, click **New Domain.**
3. On the **New Domain** page, under **Domain Details,** enter the following domain information:
    a. **Domain Name**
    In the box, enter the domain name that the certificates will secure (for example, *yourdomain.com*).
    b. **Organization**
    In the drop-down list, select the organization to assign to the domain.
4. Under **Validate This Domain For,** check the validation types needed for the domain to be validated:
    - **OV—Normal Organization Validation (Recommended)**
    Use this option to order Standard SSL, Secure Site SSL, Wildcard SSL, Secure Site Wildcard SSL, Multi-Domain SSL, and Secure Site Multi-Domain SSL certificates for this domain.
5. Under **Domain Control Validation (DCV) Method,** select **DNS TXT Record.**
Note: The default DCV method is by verification email.
6. When finished, click **Submit for Validation.**

Step II: Use DNS TXT Record to Demonstrate Control Over the Domain

1. **Create the DNS TXT record:**
    a. Under **User Actions** in the **Your unique verification token** box, copy the verification token.
    To copy the value to the clipboard, click in the text field.
    Note: The unique verification token expires after 30 days. To generate a new token, click the **Generate New Token** link.
    b. Go to the organization's DNS provider's site and create a new TXT record.
    c. In the **TXT Value** field, paste the verification code copied from the CertCentral account.
    d. Host field
        i. **Base Domain**
        If validating the base domain, leave the **Host** field blank, or use the @ symbol (dependent on the DNS provider requirements).

ii. **Subdomain**
In the **Host** field, enter the subdomain being validated.

e. In the record type field (or equivalent), select **TXT.**
f. Select a Time-to-Live value, or use the organization's DNS provider's default value.
g. Save the record.

2. **Verify the DNS TXT record:**
   a. In the CertCentral account, using the sidebar menu, click **Certificates > Domains.**
   b. On the **Domains** page in the **Domain Name** column, click the link for the domain.
   c. On the domain information page (e.g., *example.com*) at the bottom of the page, click **Check TXT.**

2.3.2 Day 1: Integration Configuration

2.3.2.1 Generate API Key

DigiCert Services API provides the foundation for the CertCentral web portal. Because DigiCert developed CertCentral as an API-first web application, the DigiCert Services API allows one to automate CertCentral web application workflows and typical certificate processes and to streamline certificate management. To access DigiCert Services API documentation, see the DigiCert Developers Portal. The services API uses RESTful conventions. The DigiCert Services API requires a DigiCert Developer API key, which is included in the header as part of each request.

Generate API Key

1. In the CertCentral account, using the side bar menu, click **Account > Account Access.**
2. On the **Account Access** page in the **API Key** section, click **Add API Key.**
3. In the **Add API Key** window, in the **Description** box, enter a description/name for the API key.
4. In the **User** drop-down, select the user to whom they key should be assigned/linked.
   Note: When linking a key to a user, link that user's permissions to the key. The API key has the same permissions as the user and can perform any action that the user can.
5. Click **Add API Key.**
6. In the **New API Key** window, click on the generated key to copy it.
7. Save the key in a secure location.
   Note: The API keys will be displayed only one time. If the window is closed without recording the new API key, the key cannot be recorded again.
8. When done, click **I understand I will not see this again.**

2.3.2.2 Venafi Integration (Automated)

Venafi integrates with the DigiCert Services API. The integrated solution leverages DigiCert's Online Certificate Status Protocol (OCSP) infrastructure and API integration with Venafi's machine identity protection platform. Customers can customize specific features, from fully automating certificate provisioning to enforcing internal policies, allowing them to address industry regulations such as Payment Card Industry Data Security Standard, Health Insurance Portability and Accountability Act of 1996, and General Data Protection Regulation. The integrated solution also simplifies integration of machine identity protection across a wide variety of systems and allows customers to fulfill certificate requests.

2.3.2.3 Order Certificate Directly Through CertCentral (Manual Process)

The TLS certificate life cycle begins when a TLS certificate is ordered. The process for requesting any of the available certificates is the same:

- Create a CSR.
- Fill out the order form by clicking the **Request a Certificate** button from the left navigation bar.
- Complete domain control validation for the domains on the order (in other words, demonstrate control over the domains).
- Complete organization validation for the organization on the certificate order.

2.3.2.4 Order an OV Single- or Multi-Domain TLS Certificate

When ordering Multi-Domain SSL certificates, add **Other Hostnames (SANs)** to the certificate order. This option is not available for the single-domain certificates.

1. **Create the CSR.**
2. **Select the OV Single- or Multi-Domain SSL/TLS certificate.**
    a. In the CertCentral account in the sidebar menu, click **Request a Certificate,** and then under All Products, click **Product Summary.**
    b. On the Request a Certificate page, look over the certificate options and select the certificate.
3. **Add the CSR.**
   On the Request page, under Certificate Settings, upload the CSR to or paste it in the **Add Your CSR** box.
   When copying the text from the CSR file, make sure to include the **-----BEGIN NEW CERTIFICATE REQUEST-----** and **-----END NEW CERTIFICATE REQUEST-----** tags.
4. **Common Name**
   Type the common name in the box, or under Common Name, expand **Show Recently Created Domains,** and select the domain from the list.
5. **Other Hostnames (SANs)**
   In the **Other Hostnames (SANs)** field, enter the additional host names needed for the certificate to be secure.
   For Multi-Domain certificates, four SANs are included in the base price of each certificate.
   Additional SANs (over those included in the base price) increase the cost of the certificate.
6. **Validity Period**

Select a validity period for the certificate: one year, two years, custom expiration date, or custom length.

Custom Validity Periods

- Certificate pricing is prorated to match the custom certificate length.
- Certificate validity cannot exceed the industry-allowed maximum life-cycle period for the certificate.
  For example, a 900-day validity period cannot be set for a certificate.

7. **Additional Certificate Options**
   The information requested in this section is optional.
   Expand **Additional Certificate Options** and provide information as needed.
   a. **Signature Hash**
      Unless there is a specific reason for choosing a different signature hash, DigiCert recommends using the default signature hash: Secure Hash Algorithm 256.
   b. **Server Platform**
      Select the server or system generated on the CSR.
   c. **Organization Unit(s)**
      Adding organization units is optional. This field can be left blank. If the CSR includes an organization unit, we use it to populate the Organization Unit(s) box.
      Note: If an organization's units are included in the order, DigiCert will need to validate them before issuing a certificate.
   d. **Auto-Renew**
      To set up automatic renewal for this certificate, check **Auto-renew order 30 days before expiration.**
      With auto-renew enabled, a new certificate order will be automatically submitted when this certificate nears its expiration date. If the certificate still has time remaining before it expires, DigiCert adds the remaining time from the current certificate to the new certificate (as long as 825 days or approximately 27 months).
      Note: Auto-renew cannot be used with credit card payments. To automatically renew a certificate, the order must be charged to an account balance.
8. To add an organization, click **Add Organization.** Add a new organization or an existing organization in the account.
   Note: When adding a new organization, DigiCert will need to validate the organization before issuing a certificate.
9. **Add Contacts**
   Two different contacts can be added to the order: Organization and Technical.
   Organization Contact (required)
   The **Organization Contact** is someone who works for the organization included in the certificate order. DigiCert will contact the **Organization Contact** to validate the organization and verify the

request for OV TLS/SSL certificates. DigiCert also sends this person an order confirmation and renewal emails.

Technical Contact (optional)

In addition to the **Organization Contact,** the **Technical Contact** will receive order emails, including the one with the certificate attached, as well as renewal notifications.

10. **Additional Order Options**

    The information asked for in this section is optional.

    Expand **Additional Order Options** and add information as needed.

    a. **Comments to Administrator**

       Enter any information the administrator might need for approving the request, such as the purpose of the certificate.

    b. **Order Specific Renewal Message**

       To create a renewal message for this certificate right now, type a renewal message with information possibly relevant to the certificate's renewal.

    Note: Comments and renewal messages are not included in the certificate.

11. **Additional Emails**

    Enter the email addresses (comma separated) for the people who want to receive the certificate notification emails, such as certificate issuance, duplicate certificate, and certificate renewals.

    Note: These recipients cannot manage the order; however, they will receive all the certificate-related emails.

12. **Select Payment Method**

    Under **Payment Information,** select a payment method to pay for the certificate.

13. **Certificate Services Agreement**

    Read the agreement and check **I agree to the Certificate Services Agreement.**

14. Click **Submit Certificate Request.**

2.3.2.5 Manage Order Within CertCentral (Manual)

After submitting the TLS certificate order, DCV and organization validation must be completed before DigiCert can issue the certificate.

If the certificate does not immediately issue, please ensure all Day 0 activities have been completed (Organization Validation and Domain Validation).

2.3.2.6 Download a Certificate from the CertCentral Account

After DigiCert issues the certificate, access it from inside the CertCentral account.

1. In the CertCentral account, go to the **Orders** page.

   In the sidebar menu, click **Certificates > Orders.**

2. On the **Orders** page, use the filters and advanced search features to locate the certificate to be downloaded.
3. In the **Order #** column of the certificate to be downloaded, click the **Quick View** link.

4. In the **Order #** details pane (on the right), using the **Download Certificate As** drop-down, select the certificate format to be used.
    - **.crt (best for Apache/Linux)**
      Download the certificate in a .crt format, best for Apache/Linux platforms.
    - **.pb7 (best for Microsoft and Java)**
      Download the certificate in a .pb7 format, best for Microsoft and Java platforms.
5. (OPTIONAL) In the **Download Certificate As** drop-down, click **More Options** to see more **Server Platform** options and **File Type** options or to download only the **Certificate,** the **Intermediate Certificate,** or the **Root Certificate.**
6. **Download a Combined Certificate File**
   In the **Download Certificate** window, under **Combined Certificate Files,** use any of these options to download the combined SSL certificate file.
    a. **Platform specific**
       In the **Server Platform** drop-down, select the server where the SSL/TLS certificate will be installed, and then click **Download.**
    b. **File type specific**
       In the **File Type** drop-down, select the SSL/TLS file format to be downloaded, and then click **Download.**
7. In the **Download Certificate** window, under **Individual Certificate Files,** use one of these options to download an individual certificate file.
    a. **Server certificate file**
       Under **Certificate,** click the **Download** link. Save the server certificate file to the server or workstation, making sure to note the location.
    b. **Intermediate certificate file**
       Under **Intermediate Certificate,** click the **Download** link. Save the intermediate certificate file to the server or workstation, making sure to note the location.
    c. **Root certificate file**
       Under **Root Certificate,** click the **Download** link. Save the root certificate file to the server or workstation, making sure to note the location.

2.3.3 Day N: Ongoing Security Management and Maintenance

2.3.3.1 Ongoing Auditing

Once the users, divisions, domains, and organizations have been added, an account audit may need to be executed to highlight areas where training is required, reconstruct events, detect intrusions, and discover problem areas.

2.3.3.2 Run an Audit

1. In the CertCentral account, using the sidebar menu, click **Account > Audit Logs.**
2. On the **Audit Logs** page, use the filters to filter the results of the audit.
    a. Choose a filter (for example, User).
    b. In the filter drop-down, select an option (for example, select a user).
    c. Wait for the filter to modify the audit log before using another filter.

2.3.3.3 Set Up Audit Log Notifications

To be of help to the organization, log data must be reviewed. The audit log notifications feature can be used to keep aware of certain activities as well as make log review more meaningful.

1. In the CertCentral account, using the sidebar menu, click **Account > Audit Logs.**
2. On the **Audit Logs** page, click **Audit Log Notifications.**
3. On the **Audit Log Notifications** page, under **Create a New Notification,** take the following steps:

Email Address	Enter the email address of the person to whom the audit log notifications are to be sent.
Division	In the drop-down, select the divisions whose account activity needs to be monitored.
Notify me about	Check any of the following options: • **Order Changes** Alerts if any changes are made to certificate orders. • **User Changes** Alerts if any edits are made to any user accounts. • **User Logins** Alerts of all account logins. • **Logins from Invalid IP Addresses** Alerts if any account logins are made from invalid IP addresses. • **Certificate Revocations** Alerts to all certificates are revocations.

4. When finished, click **Save Changes.**

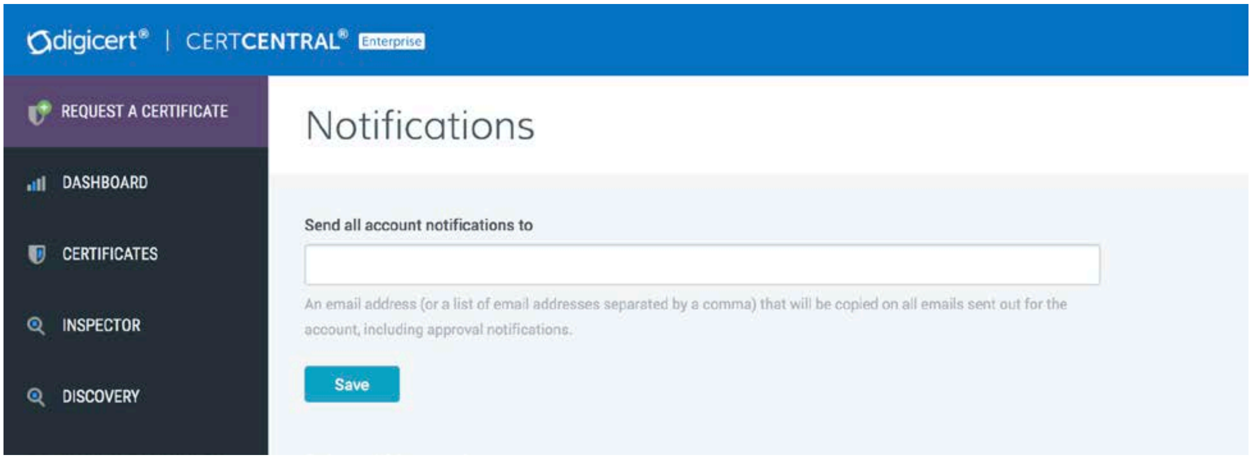
The designated individual should start receiving the selected audit log notifications.

2.3.3.4 Notification Management

Typically, notifications are not strictly required when utilizing Venafi to manage certificates, as expiring certificates are renewed automatically (or not) based on configured policy within Venafi. However, it is beneficial to configure renewal notifications within CertCentral.

2.3.3.4.1 Account Notifications

Before sending email from an account, assign an email address to receive a copy of any message sent (e.g., approval notifications). Configure renewal notifications and add default renewal messages that include renewal notifications.



2.3.3.4.2 Set Up Email Notification Accounts

1. In the CertCentral account's sidebar menu, click **Settings > Notifications.**
2. On the **Notifications** page in the **Send all account notifications to** box, add the email addresses that should be copied on all emails sent from the account.
   Note: When setting up multiple notification accounts, use commas to separate the email addresses.
3. When finished, click **Save.**

2.3.3.4.3 Certificate Renewal Notifications

After DigiCert has issued the first certificate, configure the **Certificate Renewal Settings** (such as when renewal notifications are sent and to whom notifications are sent) to help prevent unexpected certificate expirations.

When configuring the certificate renewal settings, there are two options:

1. **Nonescalation Certificate Renewals**
   This option sends renewal notifications to the same email addresses at every stage as certificates get closer to expiration or after they have expired.
2. **Escalation Certificate Renewals**
   This option configures email escalation settings in which additional email addresses can receive renewal notifications at critical stages as certificates get closer to expiring or after they have expired. This allows additional oversight of certificate expiration.

2.3.3.4.4 Configure Nonescalation Renewal Notifications

Use the steps below to send all renewal notifications to the same email addresses at every stage as certificates get closer to expiring or after they have expired.

1. In the CertCentral account's sidebar menu, click **Settings > Preferences.**
2. On the **Division Preferences** page, scroll down to the **Certificate Renewal Settings,** and uncheck **Enable Escalation.**
3. In the **Send request renewal notifications to** box, enter the email addresses for the people who should receive the renewal notifications (comma separated).
4. Under **When certificates are scheduled to expire in,** check the boxes to indicate when to send renewal notices.

   Note: These options determine when email notifications are sent. For example, if only **30 days, 7 days,** and **3 days** are checked, no email notifications will be sent **90 days** or **60 days** before certificates expire.
5. In the **Default Renewal Message** box, type an optional renewal message for inclusion in all the renewal notification emails.
6. Click **Save Settings** when finished.

2.3.3.4.5 Configure Escalation Renewal Notifications

Email escalation settings allow control over what email addresses will receive renewal notifications at each stage as certificates approach or reach expiration.

1. In the CertCentral account's sidebar menu, click **Settings > Preferences.**
2. On the **Division Preferences** page, scroll down to **Certificate Renewal Settings,** and check **Enable Escalation.**
3. Under **Days before expiration,** check the boxes for when renewal notices should be sent.
4. Under **Additional email addresses or distribution lists**, enter the email addresses for the people who should receive each renewal notification (comma separated).
5. In the **Default Renewal Message** box, type an optional renewal message for inclusion in all renewal notification emails.
6. Click **Save Settings** when finished.

2.3.3.5 Managing Custom Order Fields

CertCentral allows users to add custom fields to certificate order forms. Use the custom field metadata to search or sort a set of certificate orders that match the metadata search criteria.

Note: The **Custom Fields** feature is off by default. To enable this feature for a CertCentral account, please contact a DigiCert account representative.

Once enabled for a CertCentral account, the **Custom Order Fields** menu option is added to the sidebar menu under **Settings (Settings > Custom Order Fields).**

2.3.3.5.1 Custom order form field features

- Apply to Future and Present Requests–When a custom order form field is added, the field is also added to pending requests. If the field is required, the pending requests cannot be approved until the field is completed.
- Apply to Entire Account–When custom order form fields are added, the fields are applied to the order forms for the entire account. Custom order form fields cannot be set per division.
- Apply to All Certificate Types–When custom order form fields are created, the fields are added to the order forms for all certificate types (SSL, Client, Code Signing, etc.). A custom order form field cannot be added to the order forms for only SSL certificate types.
- Apply to Guest URLs–When custom order form fields are added, these fields are added to the certificates ordered from directly inside the CertCentral account as well as from any guest URLs that have been sent.
- Different Types to Choose From–When custom order form fields are created, different types of fields can be added such as single-line and multiple-line text boxes and email address and email address list boxes.
- Required or Optional–When custom order form fields are added, they can be required or optional. Required fields must be completed before the order can be approved. Optional fields can be left blank.
- Deactivated or Activated–After a custom order form field has been added, the field can be deactivated (removed) and activated (added back) as needed. Deactivated fields are removed from pending requests but not from issued orders. Activated fields are added to pending requests. If the field is required, it must be completed before the request can be approved.

2.3.3.5.2 Add a Custom Field to Request Forms

1. In the CertCentral account in the sidebar menu, click **Settings > Custom Order Fields.**
2. On the **Custom Order Form Fields** page, click the **Add Custom Order Form Field** link.
3. In the **Add Custom Order Form Field** window, configure the custom field:

Label	In the box, type a name/label for the field (e.g., Direct Report's Email Address).
Input Type	In the drop-down list, select an input type for the field (i.e., email address). Input Types: ▪ **Anything:** Single-line text box ▪ **Text:** Multiline text box ▪ **Integer:** Number box (limited to nondecimal whole numbers) ▪ **Email Address:** Single email address box

	▪ **Email Address List:** Multiple email address box
This field should be required for all new requests	If the field needs to be completed before the request can be submitted (or approved for pending requests), check this box. Note: If this box is not checked, the field appears on the order form with the word “optional” in the box. The requester does not need to complete the box for the request to be submitted (or approved for pending requests).

4. When finished, click **Add Custom Form Field.**

2.3.3.6 User Management

Add a user to the CertCentral account.

1. In the CertCentral account in the sidebar menu, click **Account > Users.**
2. On the **Users** page, click **Add User.**
3. On the **Add User** page in the **User Details** section, enter the new user’s information.
4. In the **User Access** section, assign the user a role, and configure their division access if applicable:

Username	We recommend using the user’s email address.
Restrict this user to specific divisions	Check this box if the role should be restricted to specific divisions. Note: This option appears only if divisions within the CertCentral account are being used.
User is restricted to the following divisions	Select the divisions to which the role is restricted. Note: This drop-down appears only if “Restrict this user to specific divisions” is checked.
Allow this user to log in only through SAML Single Sign-On SSO	Check this box if this user should be restricted from being able to log in with username and password. Note: SAML SSO must be configured in the account and the IdP must be configured with this user’s information.
Role	Select a role for the new user: Administrator, Standard User, Finance Manager, or Manager.
Limit to placing and managing their own orders	To create a Limited User role, select Standard User, and check this box.

5. When finished, click **Add User.**

What’s next

The newly added user will receive an email with instructions for setting up their account credentials and can use them to sign in to their CertCentral account.

2.3.3.7 Revalidation Processes

Organization and domain validation typically expire in two years. When the validation status nears expiration, CertCentral sends a notification and automatically initiates a revalidation process. The user should complete the steps outlined in Day 0 Organization Validation and Domain Validation. The standards governing the requirements surrounding (re)validation processes are encapsulated in the CA/Browser Forum's Baseline Requirements (https://cabforum.org/baseline-requirements-documents/). The specific allowed methods of validation will change over time.

Note: This revalidation process is outside the Venafi certificate management processes.

- OV validation and revalidation: two years
- DV validation and revalidation: two years
- EV validation and revalidation: one year

Note: Extended Validation provides additional levels of vetting surrounding the legal entity represented in a certificate. Vetting ensures that a complete picture of the identity, which has proven control over the domain in the certificate, is available to user agents verifying the certificate.

2.4 F5 BIG-IP Local Traffic Manager (LTM)

BIG-IP Virtual Edition (VE) is a version of the BIG-IP system that runs as a virtual machine in specifically supported hypervisors. BIG-IP VE emulates a hardware-based BIG-IP system running a VE-compatible version of BIG-IP software.

2.4.1 Day 0: Installation and Standard Configuration

2.4.1.1 Prerequisites

- VMware ESX 6.5
- 2 virtual Central Processing Units (CPUs)
- 4 GB RAM
- 1 x VMXNET3 virtual network adapter or Flexible virtual network adapter (for management)
- x virtual VMXNET3 virtual network adapter
- 1 x 100 GB Small Computer System Interface disk, by default
- connection to a common NTP source
- SMTP for BIG-IP to send email alerts

- a computer with internet (browser) access to activate license
- license key for F5 BIG-IP
- F5 Support ID account

2.4.1.2 Download the Virtual Appliance

To deploy BIG-IP VE, download the open virtualization appliance (OVA) file to your local system.

1. Open the F5 Downloads page at https://downloads.f5.com.
2. Log in with an F5 Support ID.
3. In the Downloads Overview page, click **Find a Download** button.
4. In the Select a Product Line page, click the **BIG-IP v13.x / Virtual Edition…** link.
5. In the Select a Product Version… page, click the **13.1.1.4_Virtual-Edition** link.
6. In the Software Terms… page, review, then click **I Accept** button to agree to terms and conditions.
7. In the Select a Download page, click the **BIGIP-13.1.1.4-0.0.4.ALL-scsi.ova** link.
8. In the Download Locations page, click the link nearest to the correct region.
9. Save the OVA file to the local computer.

2.4.1.3 Deploying the BIG-IP OVA

Use the Deploy Open Virtualization Format (OVF) Template wizard from within the VMware vSphere client. Follow the steps in this procedure to create an instance of the BIG-IP system that runs as a virtual machine on the host system.

1. Start the vSphere Client and log in.
2. Launch the **Deploy OVF Template** wizard.
3. Select an OVF template from Local file. Select the previously downloaded OVA file.
4. In the Virtual machine name field, type in `F5lb1.ext-nccoe.org`. Then select the location for this virtual machine. Click **Next.**
5. Select the compute resource and click **Next.**
6. Verify that the OVF template details are correct, then click **Next.**
7. Review the template details, then click **Next.**
8. Review License agreements. Select “I accept…” and click **Next.**
9. Read and accept the license agreement, and click **Next.**
10. Accept the default value **2 CPUs** and click **Next.**
11. Accept the default value **Thick Provision Lazy Zeroed** and click **Next.**

12. Assign the networks to the network interface cards (NICs) and click **Next.**
    - NIC 1: VLAN 2199 (Datacenter Secure)
    - NIC 2: VLAN 2201
    - NIC 3: VLAN 2197 (DMZ)
13. Review information and click **Finish.**

2.4.1.4 Assigning a Management IP Address to a BIG-IP VE Virtual Machine

The BIG-IP VE virtual machine needs an IP address assigned to its virtual management port.

1. In the main vSphere client window, **Power On** the BIG-IP.
2. Launch a Console session for the BIG-IP.
3. At the login prompt, log in as `root / default`.
4. At the config # prompt, type `config`.

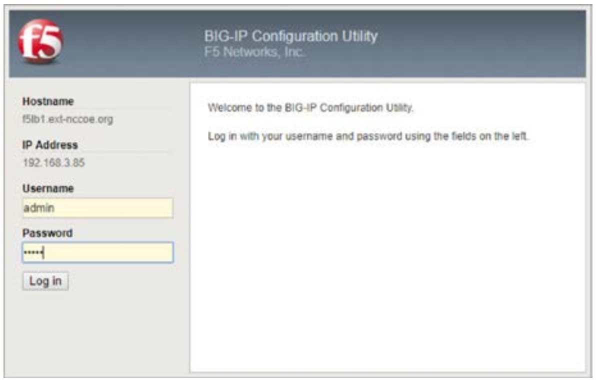
   The Configure Utility panel appears.
5. Press **Enter** for **OK.**

   The Configure IP Address panel appears.
6. For "Automatic configuration...", choose **No.**
7. For IP Address, type `192.168.3.85` Choose **OK.**
8. For Netmask, type `255.255.255.0`. Choose **OK.**
9. For Management Route, choose **Yes.**
10. For Management Route, type `192.168.3.1` Choose **OK.** The Confirm Configuration panel appears. (This Gateway address is used for management traffic.)
11. Review the IP information, and choose **Yes.** Return to the config # prompt.

2.4.1.5 Log in to BIG-IP for the First Time

After the initial login to the BIG-IP, the Setup Utility will guide through the initial setup process.

1. Open the browser and navigate to the BIG-IP address *https://192.168.3.85*.
2. Log in as the default admin/admin.



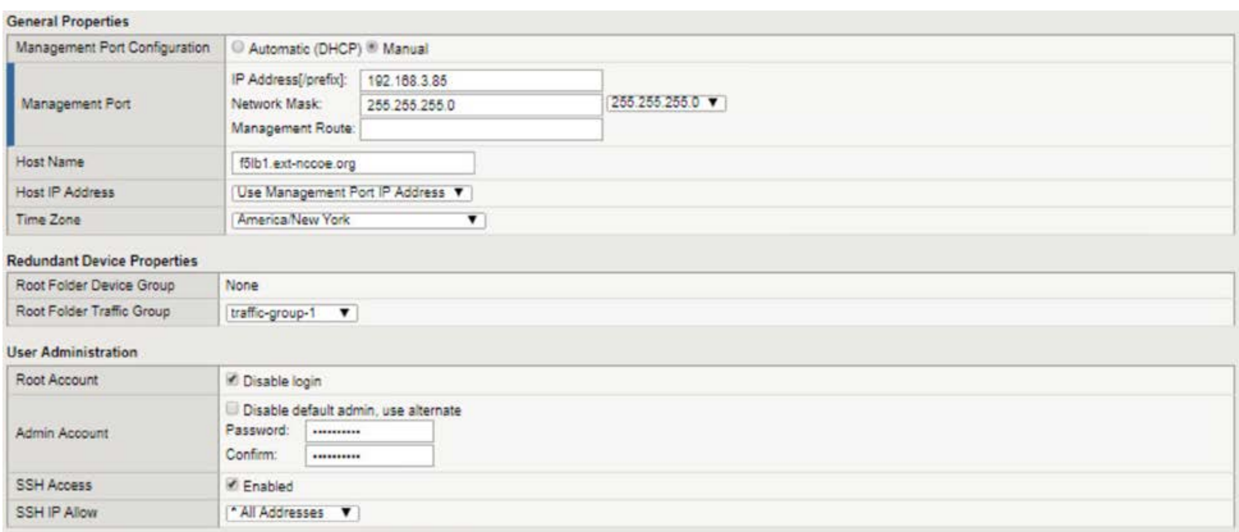
3. The Setup Utility panel appears, then click **Next.**
4. For License, click **Activate.**
5. As a prerequisite, the user should already have a BIG-IP VE license key. Copy the key and paste in the Base Registration Key field.
6. This step is dependent on internet access for the BIG-IP.
    a. If the management route configured in the previous section has a path to internet, select **Automatic.** Click **Next**. Review the End User License Agreement (EULA) and click **Agree.** Then go to step 7.
    b. Otherwise, select **Manual.** Click **Next.**
    c. **Left-click** in the Dossier field, and select all the encrypted text with **Ctrl-A.** Copy the selected text with **Ctrl-C.**
    d. Assuming the administration computer has internet access, click the "Click here to access F5…" link. A new browser tab appears.
    e. In the Enter Your Dossier field, paste in the copied text. Click **Next.**
    f. Review the EULA, and select "I have read and agree… ." Click **Next.**
    g. Left-click the license text field, and select all text with **Ctrl-A.** Copy selected text with **Ctrl-C.**
    h. Return to the BIG-IP Setup Utility. In the License field, paste in the copied text. Click **Next.**
7. Some BIG-IP services will restart and log the user off the BIG-IP. It will automatically resume. Click **Continue.**
8. Review the License page. Click **Next.**

9. On the Resource Provisioning page, verify that the only default value, **Local Traffic (LTM),** is selected and set to **Nominal.** Click **Next.**

10. On the Device Certificates page, leave the default as self-sign device Certificate. Click **Next.**

11. On the Platform page, fill these values. Then click **Next.**

Field	Value	Comments
Management Port Configuration	`443`	
IP Address	`192.168.3.85`	
Network Mask	`255.255.255.0`	
Management Route	`192.168.3.1`	
Host Name	`f5lb1.ext-nccoe.org`	
Time Zone	**EST**	
Root Account	**<your password>**	Refer to NIST SP 800-63B for password guidance.
Admin Account	**<your password>**	Refer to NIST SP 800-63B for password guidance.



12. System logs off the user with password change. Log back in with the new admin password.

13. In the Standard Network Configuration page, click **Next.**

14. In the Redundant Device Wizard Options page, **Un-Select** Display configuration synchronization options.

15. In the Internal Network Configuration page, fill in these values.

Address	*192.168.4.85*
Netmask	*255.255.255.0*
VLAN Interfaces	*internal*
Tagging	*untagged*

16. Click **Add*,*** then click **Next.**

17. In the External Network Configuration page, fill in these values.

Address	*192.168.5.86*
Netmask	*255.255.255.0*
VLAN Interfaces	*external*
Tagging	*untagged*

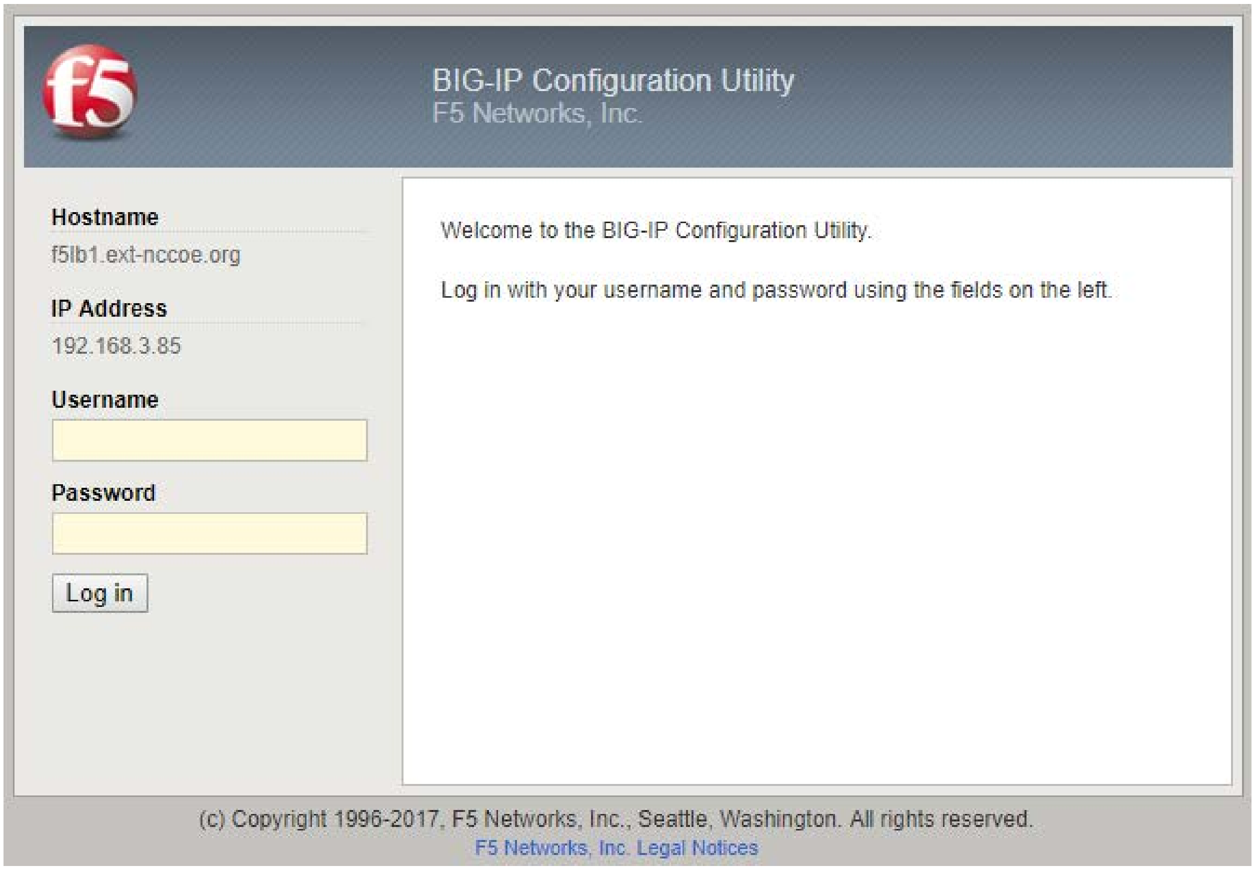
18. Click **Add*,*** then click ***Finished.***

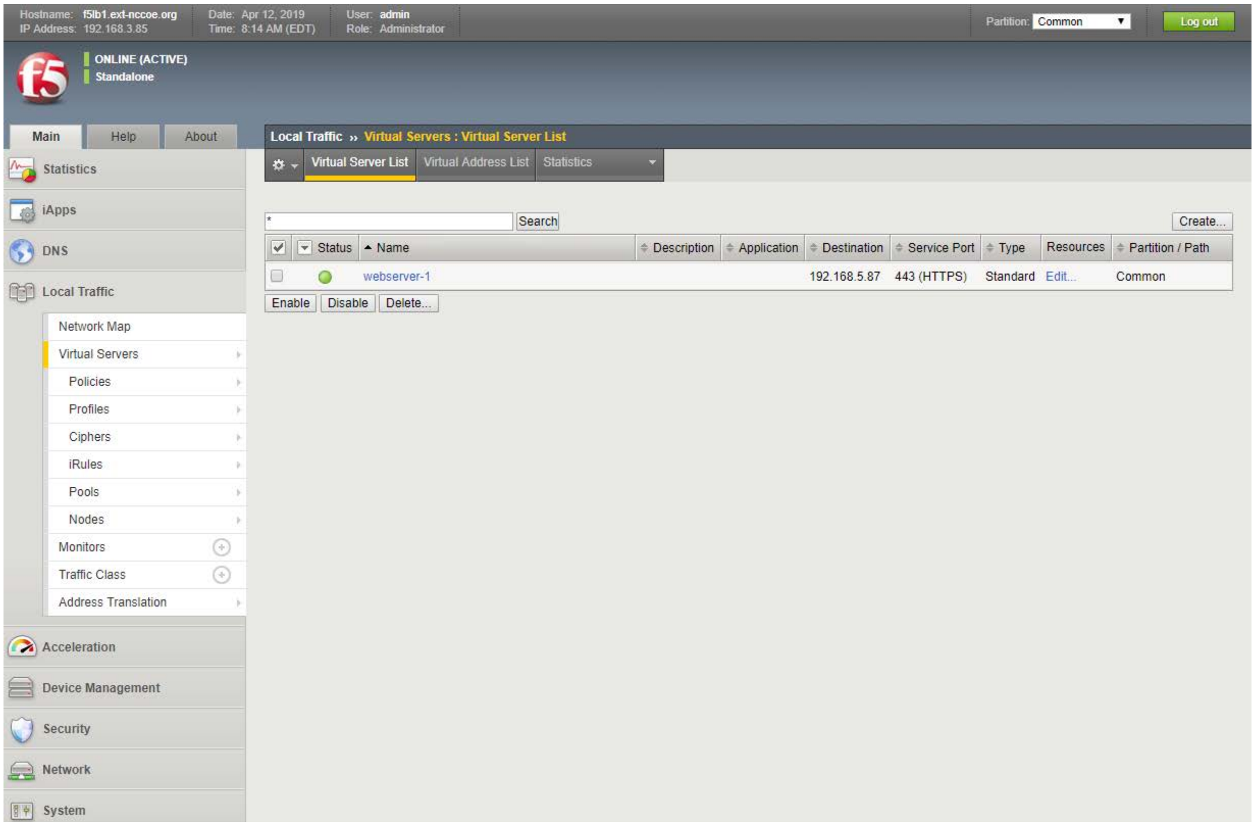
2.4.1.6 BIG-IP Configuration Utility

There are at least two ways to administer the BIG-IP.

- Use SSH to connect to the BIG-IP to access the command line interface, referred to as traffic management shell (TMSH).
- With a web browser, navigate to the management URL—referred to as Configuration utility and mainly used in this guide.

1. Open browser and navigate to the BIG-IP address *https://192.168.3.85*.
2. Log in as admin, and use the password modified from the default during Setup wizard.

2582 
BIG-IP Configuration Utility
F5 Networks, Inc.
Hostname
f5lb1.ext-nccoe.org
IP Address
192.168.3.85
Username
Password
Log in
Welcome to the BIG-IP Configuration Utility.
Log in with your username and password using the fields on the left.
(c) Copyright 1996-2017, F5 Networks, Inc., Seattle, Washington. All rights reserved.
F5 Networks, Inc. Legal Notices

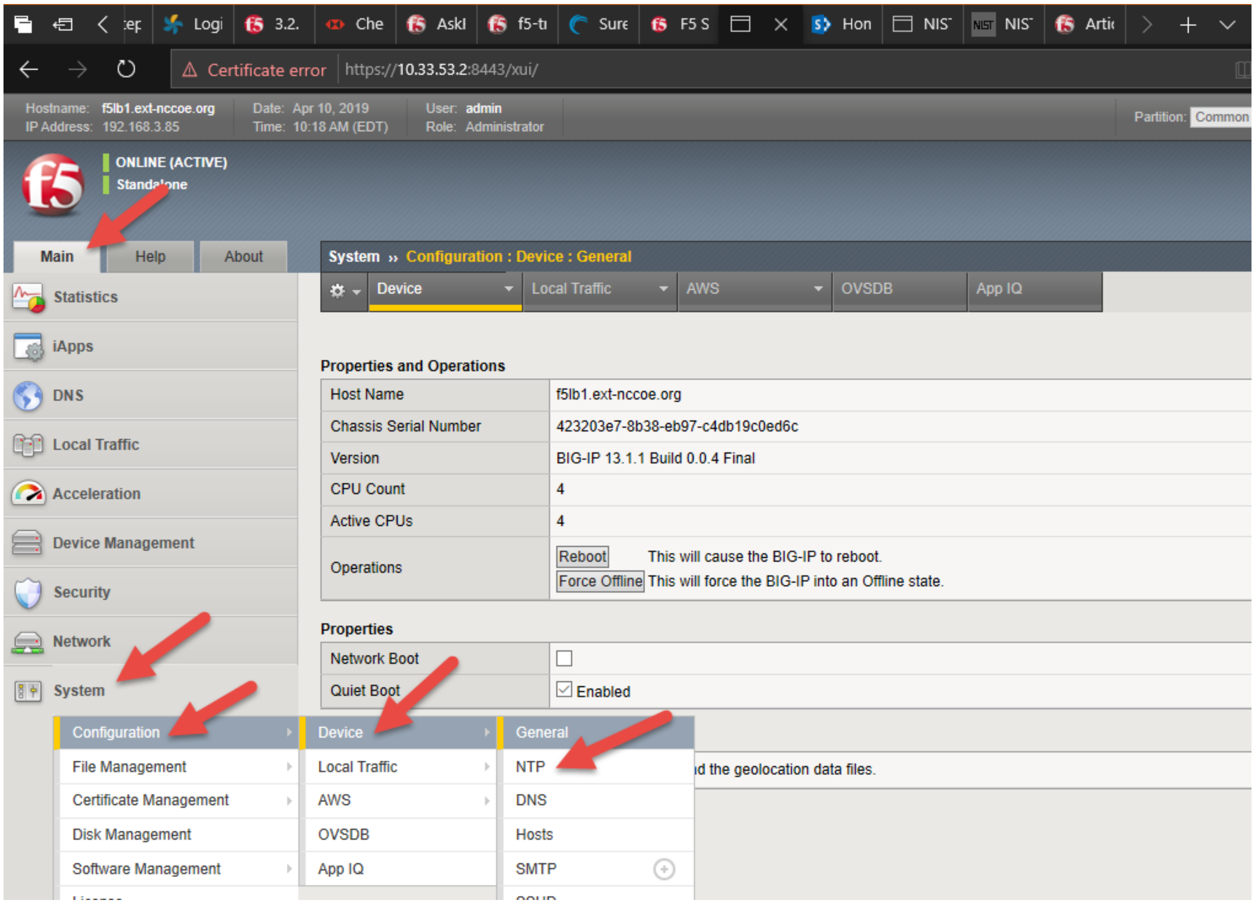


2.4.1.7 Configure NTP

Time synchronization is crucial when multiple BIG-IPs are in a cluster (not covered in this guide). It is also necessary for accuracy of logging information.

1. Log on to the Configuration utility.
2. Navigate to **Main > System.** Then click **Configuration > Device > NTP.**

   The NTP panel appears.



3. In the Address field, type `time-a-g.nist.gov`. Click **Add**.
4. In the Address field, type `time-b-g.nist.gov`. Click **Add.**
5. Click **Update.**

2.4.1.8 Configure SMTP

BIG-IP can be configured to send email alerts.

1. Navigate to **Main > System.** Then click **Configuration > Device > SMTP.**

   The SMTP panel appears.

2. In the upper right corner, click the **Create** button.

   The New SMTP Configuration panel appears.

3. Fill in these values.

Name	`mail1`
SMTP Server Host Name	`mail1.int-nccoe.org`
Local Host Name	`f5lb1-ext-nccoe.org`
From Address	`f5-big-ip@nccoe.org`

4. Click **Finish.**

2.4.1.9 Configure Syslog

Log events either locally on the BIG-IP system or remotely by configuring a remote syslog server.

1. Log on to the Configuration utility.
2. Navigate to **System > Logs > Configuration > Remote Logging.**
3. In Remote IP field, type `192.168.3.12`.
4. Click **Add.**
5. Click **Update.**

2.4.1.10 Secure BIG-IP to NIST SP 800-53

This section provides guidance on using the F5 iApp for NIST SP 800-53 (Revision 5) to configure a BIG-IP device to support security controls according to NIST SP 800-53 (Revision 4): *Security and Privacy Controls for Federal Information Systems and Organizations* (updated January 2, 2015).

Some controls (policies plus supporting technical measures) that organizations adopt by complying with NIST SP 800-53 (Revision 5) relate to the BIG-IP configuration.

This practice guide discusses the security controls in Appendix F of NIST SP 800-53 (Revision 5) that apply to BIG-IP configuration and shows how to support them. It also focuses on configuring the management features of the BIG-IP system rather than the network-traffic-processing modules of a system such as BIG-IP Local Traffic Manager. This approach helps the user manage the BIG-IP system as an entity responsive to NIST SP 800-53 (Revision 5) controls. Using BIG-IP as a tool to help control other entities, such as network-based applications, is beyond the scope of this project.

2.4.1.10.1 F5 iApp

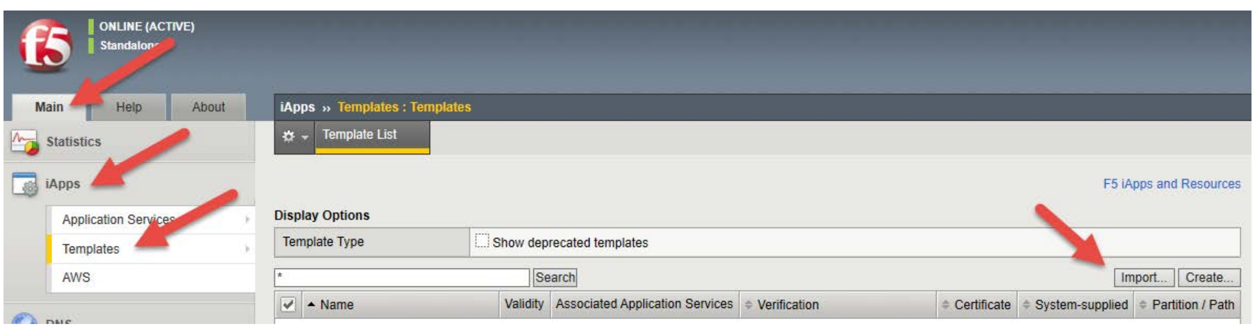
F5 iApp is a feature in the BIG-IP system that provides a way to simplify BIG-IP configurations. An iApp template brings together configuration elements, architectural rules, and a management view to deliver an application reliably and efficiently.

2.4.1.10.2 Download the iApp for NIST SP 800-53 (Revision 5) Compliance

1. In a browser, open the F5 Downloads page at https://downloads.f5.com.
2. Log in with an F5 Support ID.
3. In the Downloads Overview page, click **Find a Download** button.
4. In the Select a Product Line page, under Product Line column, click **iApp Templates.**
5. In the Select a Product Version… page, click **iApp-Templates.**
6. Review the EULA, then click **I Accept.**
7. In the Select a Download page, click **iapps-1.0.0.546.0.zip.**
8. In the Download Locations page, click on the link nearest to the user's region.
9. Save the zip file to the local computer.

2.4.1.10.3 Import iApp to BIG-IP

1. Unzip the downloaded file.
2. Open browser and navigate to the BIG-IP address *https://192.168.3.85*.
3. Log in as admin/admin.
4. On the left menu, click **Main > iApps > Templates.** Then on the right side, click **Import** button.



5. Browse to the file unzip location and to the subfolder **\iapps-1.0.0.546.0\Security\NIST\Release_Candidates.** Select the file ***f5.nist_sp800-53.v1.0.1rc5.tmpl,*** then click **Open.**
6. Click **Upload.**
7. On page 2 of the Template List, verify that the **f5.nist_sp800-53.v1.0.1rc5** template has been uploaded.

2.4.1.10.4 Deploy the NIST iApp

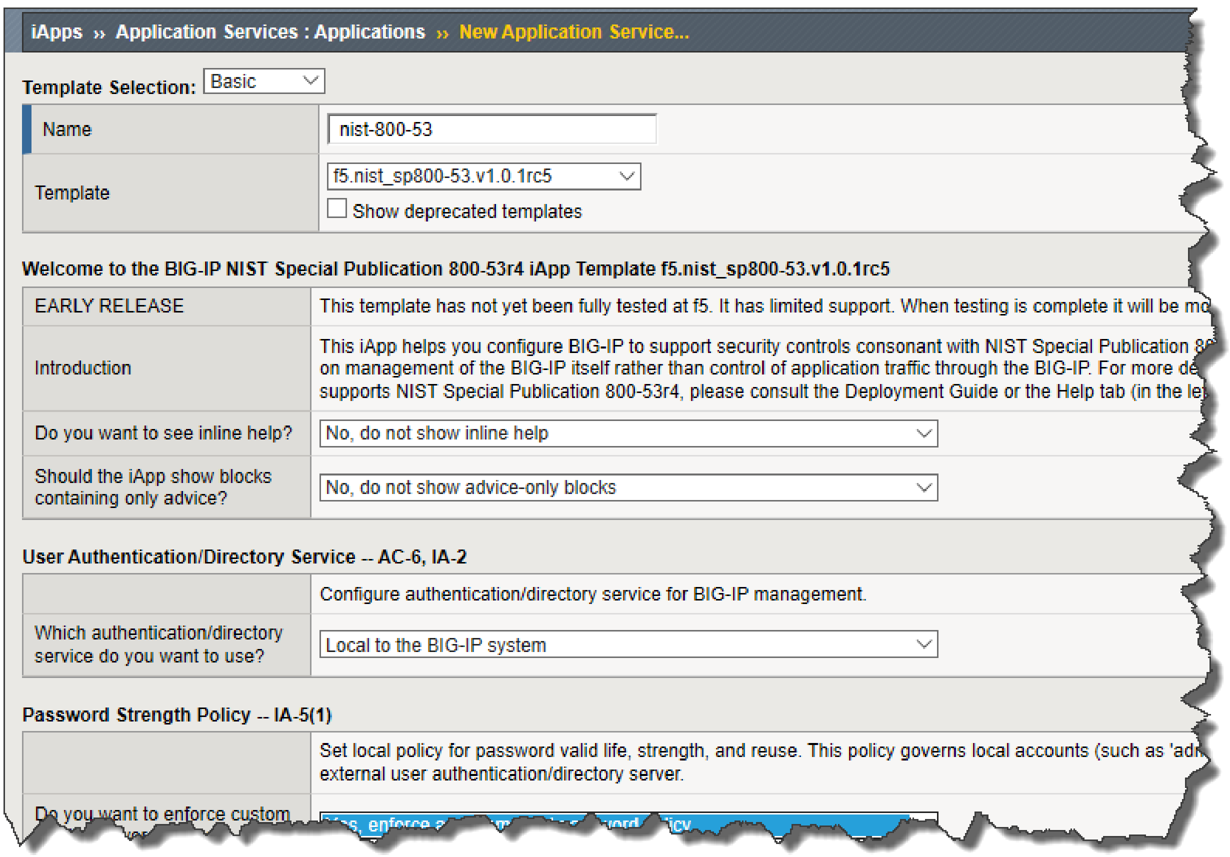
1. On the left menu, click **Main > iApps > Application Services.** Then on the right side, click **Create** button.

   The Template Selection panel appears.

2. In the Name field, type `nist-800-53`.

3. In the Template pull-down, select **f5.nist_sp800-53.v1.0.1rc5.**

   The New Application Service panel appears.



4. Fill in the iApps with parameters in the following table. Leave everything else as default values.

Password Strength Policy—IA-5(1)	
Do you want to enforce custom local password policy?	`“Yes, enforce a custom…”`

How many days should pass before the password expires?	`0`
How many changes before reuse?	`0`
How many characters should be the minimum for each setting?	`Length = 8`
Maximum Failed Login Attempts—AC-7	
Disable account after several failed login attempts?	`“Yes, limit fail…”`
Allow how many consecutive login failures before disabling the account?	`9`
NTP Configuration—AU-8(1,2)	
What is the IP address or FQDN of the primary NTP server?	`time-a-g.nist.gov`
What is the IP address or FQDN of the first alternate NTP server?	`time-b-g.nist.gov`
Syslog Configuration—AU-8, AU-9(2), AU-12(2)	
Should log messages use International Standards Organization (ISO) date format?	`“Yes, log messages…”`
Do you want to add syslog servers?	`“Yes, use this iApp…”`
Which syslog servers do you want to add?	`Server: syslog2.int-nccoe.org`

5. Click **Finished.**

2.4.2 Day 1: Product Integration Configuration

2.4.2.1 Prerequisites

- Venafi installed
- web servers for load balance

2.4.2.2 Venafi Integration

For information on integration with Venafi TPP, see Section 2.6.13.1.

2.4.2.3 Load Balance Web Servers

2.4.2.3.1 Create a Pool to Manage https Traffic

A pool (a logical set of devices, such as web servers, that are grouped together to receive and process https traffic) can be created to efficiently distribute the load on the server resources.

1. On the Main tab, click **Local Traffic > Pools.**

   The Pool List screen opens.

2. Click **Create.**

   The New Pool screen opens.

3. In the Name field, type `app1_pool`.

4. For the Health Monitors setting, assign https by moving it from the Available list to the Active list.

5. Use the New Members setting to add each resource to include in the pool:

   a. In the Address field, type `192.168.4.2`.

   b. In the Service Port field type `443`.

   c. Click **Add.**

6. Repeat step 5 for these three IP addresses.

   a. `192.168.4.3`

   b. `192.168.4.4`

   c. `192.168.4.7`

7. Click **Finished.**

The https load balancing pool appears in the Pool List screen.

2.4.2.3.2 Create Client SSL Profile

Profile for BIG-IP to decrypt traffic from browser

1. On the Main tab, click **Local Traffic > Profiles > SSL > Client.**

   The SSL Client List screen opens.

2. Click **Create.**

   The New Client SSL Profile screen opens.

3. In the Name field, type `app1_client-ssl`.
4. In the Certificate Key Chain setting, select the checkbox on the right. Then click **Add.**

   The Add SSL Certificate to Key Chain screen opens.

5. For **Certificate** pull-down, select app1.tls.nccoe.org-<value>.
6. For **Key** pull-down, select app1.tls.nccoe.org-<value>.
7. Click **Add.**
8. Click **Finished.**

2.4.2.3.3 Create Server SSL Profile

Profile for BIG-IP to encrypt traffic to web servers:

1. On the Main tab, click **Local Traffic > Profiles > SSL > Server.**

   The SSL Server List screen opens.

2. Click **Create.**

   The New Server SSL Profile screen opens.

3. In the Name field, type `app1_server-ssl`.
4. In the Certificate setting, select the checkbox on the right. Then select app1.tls.nccoe.org-<value> in the pull-down.
5. In the Key setting, select the checkbox on the right. Then select app1.tls.nccoe.org-<value> in the pull-down.

   The Add SSL Certificate to Key Chain screen opens.

6. For **Certificate** pull-down, select app1.tls.nccoe.org-<value>.
7. For **Key** pull-down, select app1.tls.nccoe.org-<value>.
8. Click **Finished.**

2.4.2.3.4 Create a Virtual Server to Manage https Traffic

A virtual server can be specified to be either a host virtual server or a network virtual server to manage https traffic.

1. On the Main tab, click **Local Traffic > Virtual Servers.**

   The Virtual Server List screen opens.

2. Click the **Create** button.

   The New Virtual Server screen opens.

3. In the Name field, type `app1_vs`.
4. In the Destination Address field, type `192.168.5.85`.
5. In the Service Port field, type `443`.
6. In the HTTP Profile setting, select **http** in the pull-down.
7. In the SSL Profile (Client) setting, from the Available list, select **app1_client-ssl**, and click the << button to move over to the Selected list.
8. In the SSL Profile (Server) setting, from the Available list, select **app1_server-ssl**, and click the << button to move over to the Selected list.
9. In the Source Address Translation setting, select **Auto Map** in the pull-down.
10. In the Default Pool setting, select **app1_pool** in the pull-down.
11. In the Default Persistence Profile setting, select **cookie** in the pull-down.
12. Click **Finished.**

The https virtual server appears in the Virtual Server List screen.

2.4.2.3.5 Create Redirect Virtual Server from http to https

When a user types *http://<virtual server>* in the browser, this virtual server redirects the user to the secure site *https://<virtual server>.*

1. On the Main tab, click **Local Traffic > Virtual Servers.**

   The Virtual Server List screen opens.

2. Click the **Create** button.

   The New Virtual Server screen opens.

3. In the Name field, type `app1_redir_vs`.
4. In the Destination Address field, type `192.168.5.85`.

5. In the Service Port field, type `80`.
6. In the HTTP Profile setting, select **http** in the pull-down.
7. In the iRules setting, select **_sys_https_redirect** in Available, and click the << button to move over to the Enabled list.
8. Click **Finished.**

The http redirect virtual server appears in the Virtual Server List screen.

2.4.3 Day N: Ongoing Security Management and Maintenance

2.4.3.1 Software Updates

BIG-IP VE updates in the same major version are installed in a similar manner as updates to BIG-IP software already installed on BIG-IP hardware. There is no need to reinstall BIG-IP VE in the hypervisor guest environment to upgrade the system. To update a BIG-IP VE virtual machine, use the Software Management tool in the Configuration utility, or upgrade the software from the command line. The update procedure described in this guide uses the Software Management tool.

2.4.3.1.1 Download the Latest Software

Software release notes contain instructions for that specific installation.

To find the latest software version for an F5 product:

1. Navigate to F5 Downloads (downloads.f5.com).
2. Click **Find a Download.**
3. Find the product desired for download, and click the link for the appropriate version.
4. Find and click the link for the update to download.
5. Read and accept the End User Software license agreement.
6. Click the file name, choose a download location, and save the file to the computer.

2.4.3.1.2 Upgrading BIG-IP Software

Before upgrading the BIG-IP software, we recommend reviewing the release notes on AskF5 (support.f5.com) in the Documentation section of the product and version. In particular, verify the new version supports the hardware, and carefully review these items:

- known issues list
- behavior change section(s)

- upgrading from earlier versions section
- upgrading from earlier configurations section
- installation checklist

2.4.3.1.3 Import a BIG-IP VE Software Update

To install an update, BIG-IP software needs access to the ISO file previously downloaded.

1. Open browser, and navigate to the BIG-IP address *https://192.168.3.85*
2. Log in as an admin.
3. On the **Main** tab, click **System > Software Management**.

   The *Software Management Image List* screen opens.

4. At the right side of the screen, click **Import**.

   The *New Image* screen opens.

5. Click **Browse** to navigate to the downloaded installation file.
6. When the image name appears in the Software Image field, click **Import** to begin the operation.

   The system presents a progress indicator during the operation.

2.4.3.1.4 Installing a BIG-IP VE update

After import the software image, initiate the installation operation.

1. On the **Main** tab of the navigation pane, click **System > Software Management**.

   The *Software Management Image List* screen opens.

2. From the *Available Images* table, select the software image you want to install.

   The image properties screen opens.

3. Click **Install**.

   The *Install Software* screen opens.

4. Select the disk you want to install the image on, and type or select a volume name, and click **Install**.

   The upgrade process installs the software on the inactive disk location that you specify. This process usually takes between three and ten minutes.

   Tip: If a problem arises during installation, use log messages to troubleshoot a solution. The system stores the installation log file as */var/log/liveinstall.log*.

5. The software image is installed.

2.4.3.1.5 Reboot BIG-IP VE to update

When the installation operation is complete, you can safely reboot into the newly installed volume or partition.

1. On the **Main** tab of the navigation pane, click **System > Software Management**.

   The *Software Management Image List* screen opens.

2. On the menu bar, click **Boot Locations**.

   The *Boot Locations* screen opens.

3. In the *Boot Location* column, click the link representing the boot location you want to activate.

   The properties screen for the boot location opens.

4. Click **Activate**.

   A confirmation screen opens.

5. Click **OK** to initiate the reboot operation.

   The system presents progress messages during the restart operation.

When the BIG-IP VE system reboot is complete, the system presents the login screen. To configure the system, log in using an account that has administrative permissions.

2.4.3.2 License and Entitlement

If support is purchased from F5, it is associated with a particular BIG-IP system. A system with an active support contract is considered entitled until the contract expires. To continue receiving support, the contact must be renewed.

Licenses are also associated with modules purchased to run a specific system. Model licenses are considered add-ons to the main license for a system, and are automatically linked to the main BIG-IP system license and eligible for technical support if that system is entitled.

Major software upgrades are only supported for entitled systems and require relicensing of the BIG-IP system. Minor upgrades do not require relicensing.

2.4.3.2.1 Viewing and verifying a BIG-IP system license

Test the validity of the BIG-IP software license by obtaining license information in any of the following ways:

- view license information at the command line
- request a product license profile from F5

- view license profile in BIG-IP iHealth®
- view license profile in the Configuration utility
- At the command line, type the following command: `tmsh show /sys license`

Output displays licensing information for the BIG-IP system should include a list of active modules. For a system with a valid license, output appears similar to the following example:

2.4.3.2.2 Provisioning licenses

If a license is installed for an add-on module on a BIG-IP system, you must provision resources for the module.

Until provisioned, module function is limited in the following ways:

- the system does not perform the functions of the licensed module
- items related to the module do not appear in Configuration utility menus
- the TMOS Shell (tmsh) does not present or permit configuration of objects related to the module.
- the bigstart status command returns output similar to the following example for daemons related to the unprovisioned module: <daemon_name> down, Not provisioned For information on provisioning modules, refer to "Modules."

When you upgrade a BIG-IP system, the install script verifies the Service Check Date with the license check date of the version being installed. If the service check date is missing or the verification process finds your license pre-dates the software's release date, a line displays in the */var/log/liveinstall.log* with a note about the service check date verification, and the installation of the software may continue.

2.4.3.2.3 Reactivating a BIG-IP System License

F5 recommends reactivating the BIG-IP system license before conducting a software upgrade.

Follow these steps to reactivate a BIG-IP system license using the Configuration utility:

1. Navigate to System > License.
2. Click **Re-activate**.
3. In the Activation Method area, select **Automatic** (requires outbound connectivity).
4. Click **Next**.

2.4.3.2.4 Moving a BIG-IP VE license

BIG-IP VE licenses are permanently associated with the virtual instance. To move a license, contact F5 Technical Support for assistance. However, with BIG-IP 12.1.3.3 and BIG-IP 13.1 and later, you can move the RegKey without contacting support by revoking the instance's license from tmsh, the Configuration utility, and iControl/REST by using the 'tmsh revoke sys license' command on that virtual instance. This action revokes the license and unlocks the RegKey—enabling the user to activate a new virtual machine.

Call F5 Technical Support for assistance if the connection is lost and you want to move the license to the current VE, if hypervisor crashes, or if you can't access the password or network address.

2.4.3.3 Backup and Data Recovery

BIG-IP software offers two supported methods for backing up and restoring the configuration: user configuration set (UCS) archives and single configuration files. This guide focuses on using the UCS archive only. To create, delete, upload, or download an archive, you must have either administrator or resource administrator role privileges.

2.4.3.3.1 Backup Configuration Data to a UCS Archive

A UCS archive contains BIG-IP configuration data that can fully restore a BIG-IP system in the event of a failure or return material authorization.

Each time you back up the configuration data, the BIG-IP system creates a new UCS archive file in the */var/local/ucs* directory. In addition to configuration data, each UCS file contains various configuration files necessary for the BIG-IP system to operate correctly.

A UCS archive contains the following types of BIG-IP system configuration data:

- system-specific configuration files (traffic management elements, system and network definitions, and others)
- product licenses
- user accounts and password information
- DNS
- zone files
- installed SSL keys and certificates

To easily identify the file, include the BIG-IP host name and current time stamp as part of the file name.

F5 recommends keeping a backup copy of the UCS archives on a secure remote server. To restore the BIG-IP system if you can't access the */var /loca/ucs* directory on the BIG-IP system, upload the backup file from the remote server, and use it to restore your system.

2.4.3.3.2 To create a UCS archive using the Configuration utility

When creating a new archive, unless otherwise directed, the BIG-IP system automatically stores it in */var/local/ucs* directory—a default location. You can create as many archives as you want, but each archive must have a unique file name.

All boot locations on a BIG-IP system use the same /shared directory, making it a good choice for a UCS save location. Saving an archive to the /shared directory allows you to boot to another boot location and access the archive, and can greatly simplify the recovery from a variety of issues.

1. Navigate to **System > Archives**.
2. Click **Create**.
3. Type a unique file name.
4. To encrypt the archive for Encryption, click **Enabled**.
5. To include private keys in the BIG-IP system, for Private Keys*,* click **Include**. If you choose to include private keys, store the archive file in a secure environment.
6. Click **Finished**.
7. Click **OK** after the data is backed up and the file is created.

2.4.3.3.3 To download and copy an archive to another system using the Configuration utility

1. Navigate to **System > Archives**.
2. Click the UCS file name you want to download.
3. In Archive File, click Download <filename>.ucs.
4. Save the file.
5. Find the file in your computer's Downloads folder and copy it.

2.4.3.3.4 Restoring Configuration Data from a UCS Archive

If the BIG-IP System configuration data becomes corrupted, you can restore the data from the archive currently stored in the directory */var/local/ucs*.

When restoring configuration data, F5 recommends running the same version of the BIG-IP software on the BIG-IP system from which it was backed up.

F5 also recommends restoring a UCS file to another platform of the same model where the UCS file was created. Certain core hardware changes can cause a UCS to load properly on dissimilar hardware, requiring manual intervention to correct.

2.4.3.3.5 To restore a configuration in a UCS archive using the Configuration utility

1. Navigate to **System > Archives.**
2. Click the name of the UCS archive you want to restore.
3. To initiate the UCS archive restore process, click **Restore**.

   When the restoration process is completed, examine the status page for any reported errors before proceeding to the next step.
4. To return to the Archive List page, click **OK**.

If you receive activation errors after restoring a UCS archive on a different device, you must reactivate the BIG-IP system license. Restarting the system ensures that the configuration is fully loaded after relicensing,

2.4.3.3.6 Downloading a UCS Archive to a Remote System

Downloading a copy of an existing archive to a remote system protects the configuration data should you need to restore your BIG-IP system and be unable to access the /var/local/ucs directory on the BIG-IP system.

To download an existing archive, first display the properties of the archive to specify the complete path name of the location where you want to save the archive copy.

1. Navigate to **System > Archives**.
2. Click the name of the archive that you want to view.

   The General Properties for that archive display.
3. Click **Download**: <ucs filename>.
4. Click **Save**.

The BIG-IP system downloads a copy of the UCS file to the system from which you initiated the download.

2.4.3.3.7 Uploading a UCS Archive from a Remote System

If a UCS archive on your BIG-IP system is unavailable or corrupted, upload a previously created archive copy from a remote or backup system to replace it.

1. Navigate to **System > Archives**.
2. Click **Upload**.
3. Type the complete path and file name of the archive that you want to upload onto the BIG-IP system.

   If you do not know the path or file name, click **Browse** and navigate to the location.
4. Click **Upload**.

The specified archive uploads to the */var/local/ucs* directory on the BIG-IP system.

2.4.3.3.8 Deleting a UCS Archive

Use the Configuration utility to delete any archive on the BIG-IP system that is stored in the directory */var/ local/ucs*.

1. Navigate to **System > Archives**.

2. Select the check box next to the name of the file you want to delete.
3. Click **Delete**.
4. Click **Delete** again.

The archive is deleted from the */var/local/ucs* directory on the BIG-IP system.

2.4.3.4 Log Files and Alerts

This section provides context for our recommended procedures in the form of overviews and supplemental information, including the following topics:

- Config for Syslog
- Set up SMTP for email alerts

2.4.3.4.1 Managing Log files on a BIG-IP System

Log files track usage or troubleshoot issues—if left unmanaged, they can grow to an unwieldy size. The BIG-IP system uses a utility called logrotate to manage local log files. The logrotate script deletes log files older than the number of days specified by the Logrotate.LogAge database variable. By default, the variable is set to eight. Therefore, the system is configured to delete archive copies that are older than eight days.

To modify the Logrotate.LogAge database variable:

1. Log in to tmsh at the command line by typing the following command: `tmsh`
2. Modify the age at which log files are eligible for deletion by using the following command syntax: `modify /sys db logrotate.logage value <value 0 - 100>`
3. Save the change by typing the following command: `save /sys config`

2.4.3.4.2 Audit Logging

Audit logging is an optional way to log messages pertaining to configuration changes that users or services make to the BIG-IP system configuration. Audit logging is also known as master control program.

LOG FILES AND ALERTS—PROCEDURES

(MCP) Audit Logging. As an option, you set up audit logging for any tmsh commands that users type on the command line.

For MCP and tmsh audit logging, select a log level. The log levels will not affect the severity of the log messages but may affect the initiator of the audit event.

2.4.3.5 Technical Support

In addition to Support Centers around the world, there are many technical resources available to customers.

2.4.3.5.1 Phone Support

Open a Case at any of the Network Support Centers:

- 1-888-882-7535 or (206) 272-6500
- International contact numbers: http://www.f5.com/training-support/customer-support/contact/

2.4.3.5.2 AskF5 - Web Support

F5 self-support portal: http://www.askf5.com

2.4.3.5.3 DevCentral - F5 User Community

More than 360,000 members—including F5 engineering resources—are actively contributing, sharing and assisting our peers.

http://devcentral.f5.com

2.4.3.5.4 BIG-IP iHealth

BIG-IP iHealth comprises BIG-IP iHealth Diagnostics and BIG-IP iHealth Viewer. BIG-IP iHealth Diagnostics identifies common configuration problems and known software issues. It also provides solutions and links to more information. With BIG-IP iHealth Viewer, you can see the status of your system at-a-glance, drill down for details, and view your network configuration.

https://ihealth.f5.com/

2.4.3.5.5 Subscribing to TechNews

AskF5 Publications Preference Center provides email publications to help keep administrators up-to-date on various F5 updates and other offerings:

- TechNews Weekly eNewsletter Up-to-date information about product and hotfix releases, new and updated articles, and new feature notices.
- TechNews Notifications Do you want to get release information, but not a weekly eNewsletter? Sign up to get an HTML notification email any time F5 releases a product or hotfix.
- Security Alerts Receive timely security updates and ASM attack signature updates from F5.

To subscribe to these updates:

1. Go to the Communications Preference Center (https://interact.f5.com/F5-Preference-Center.html).

2. Under My preferences click **Show**.

3. Select the updates you want to receive.

4. Click **Submit**.

2.4.3.5.6 AskF5 recent additions and updates

You can subscribe to F5 RSS feeds to stay informed about new documents pertaining to your installed products or products of interest. The Recent additions and updates page on AskF5 provides an overview of all the documents recently added to AskF5.

New and updated articles are published over RSS. You can configure feeds that pertain to specific products, product versions, and/or document sets. You can also aggregate multiple feeds into your RSS reader to display one unified list of all selected document.

2.5 Symantec SSL Visibility Appliance

The Symantec SSL Visibility appliance is a high-performance transparent proxy for SSL network communications. It enables a variety of applications to access the plaintext (that is, the original unencrypted data) in SSL encrypted connections, and is designed for security and network appliance manufacturers, enterprise IT organizations, and system integrators. Without compromising any aspect of enterprise policies or government compliance, the SSL Visibility appliance permits network appliances to deploy with highly granular flow analysis while maintaining line rate performance.

2.5.1 Day-0: Install and Standard Configuration

2.5.1.1 Prerequisites

- 120V or 220V Power Source
- computer with browser access to activate license and configure appliance
- putty or a terminal emulator
- four-post equipment rack with a depth of 27.75” to 37.00” with square mounting holes
- category 5E network cables or better (Category 6 or 6A)
- license key for SSL Visibility appliance
- MySymantec account
- DNS Server
- SSL VISIBILITY running version 3.X

2.5.1.2 Unpacking the Appliance

Before racking and configuring the SSL Visibility Appliance, ensure the following contents are included in the SSL Visibility shipping package:

	SV800	SV1800	SV2800	SV3800
External power supply with AC power cord	√			
Two AC power cords		√	√	√
Rack-mount rail kit		√	√	√
Rack-mount ears with fasteners		√	√	√
Safety and Regulatory Compliance Guide	√	√	√	√
Quick Start Guide (this document)	√	√	√	√
Software License Agreement	√	√	√	√
Hardware Warranty	√	√	√	√

2.5.1.3 Rack-Mount the Appliance

The list below shows the requirements to install the SSL Visibility Appliance.

- At least 1U rack space (deep enough for a 27” device)–power and management ports at rear
- Phillips (cross head) screwdriver
- Weight Capacity: 28lb (12.7kg)
- Dimensions: 17.5” (W) x 19.5” (D) x 1.75” (H) (444.5mm x495.3mm x 44.5mm)
- Two available power outlets (110 VAC or 220-240 VAC)
- Two IEC-320 power cords (normal server/PC power cords) should the supplied power cords not be suitable for your environment
- Cooling for an appliance with two 450W power supply units

To see detailed instructions for installing the SSL Visibility in a rack, please refer to Symantec’s Quick Start guide located at the below link:

https://symwisedownload.symantec.com//resources/sites/SYMWISE/content/live/DOCUMENTATION/10000/DOC10294/en_US/SSL_VISIBILITY_Quick_Start_Guide.pdf?__gda__=1556050986_e4bd9c26d33192a730d884f8137ce9e6

2.5.1.4 Connect Cables

To connect the appliance’s cables:

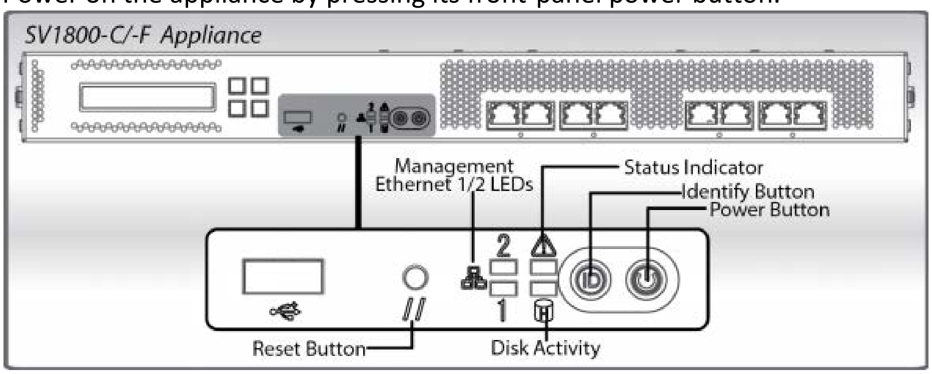
1. Connect a network cable between the **Management Ethernet 1** port, on the rear of the SSL VISIBILITY appliance, and Datacenter Secure network.

   Warning: When deploying the SV1800, SV2800, and SV3800 appliances, do not connect to the Management Ethernet 2 port. This port is not functional.

2. Connect the two AC power cords to the appliance's AC power inlets on the rear panel. Two power supplies are provided for redundant operation.
3. Connect the other ends of the power cords to a 120V or 220V power source.

2.5.1.5 Power on the Appliance and Verify LEDs

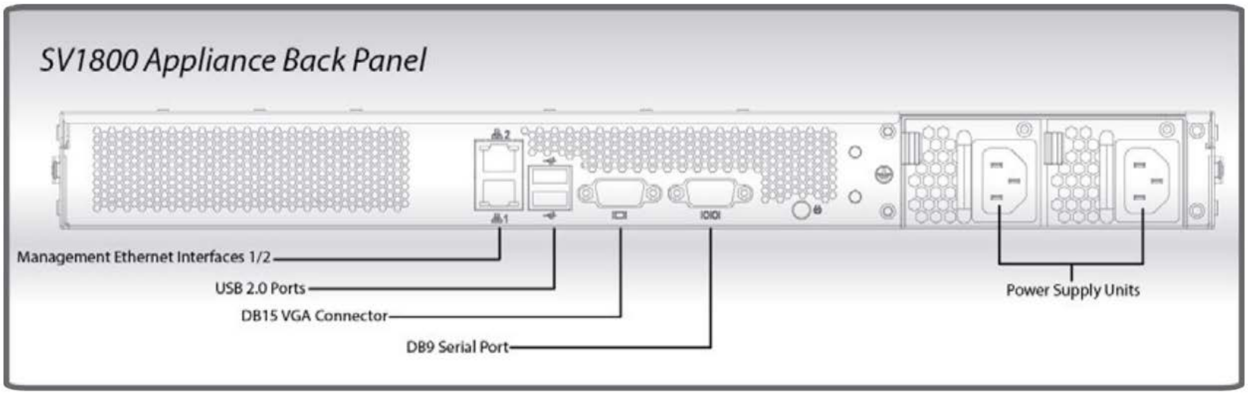
1. Confirm the appliance's power cord or power cords are securely connected to a 120V or 220V power source.
2. Power on the appliance by pressing its front-panel power button.



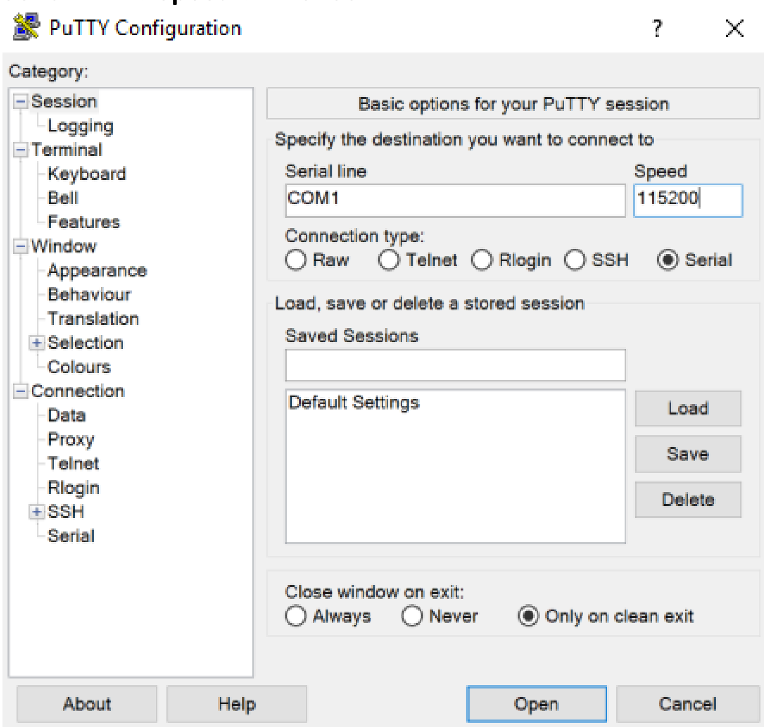
3. As the appliance boots verify the following:
   - The LCD displays startup messages while the appliance boots (Appliance Startup, Validating Firmware, Appliance Boot, etc.).
   - The System Status indicator for the SV1800 changes from red to off.
   - The LEDs for the Management Ethernet port (connected to a management workstation) light up.
   - When the boot process is complete, the LCD displays the appliance's model, software version, and the Up/Down arrows.

2.5.1.6 Initial Appliance Configuration

1. To perform initial configuration of the SSL Visibility Appliance, connect a serial cable to the **DB9 Serial port** on the rear of the Appliance.

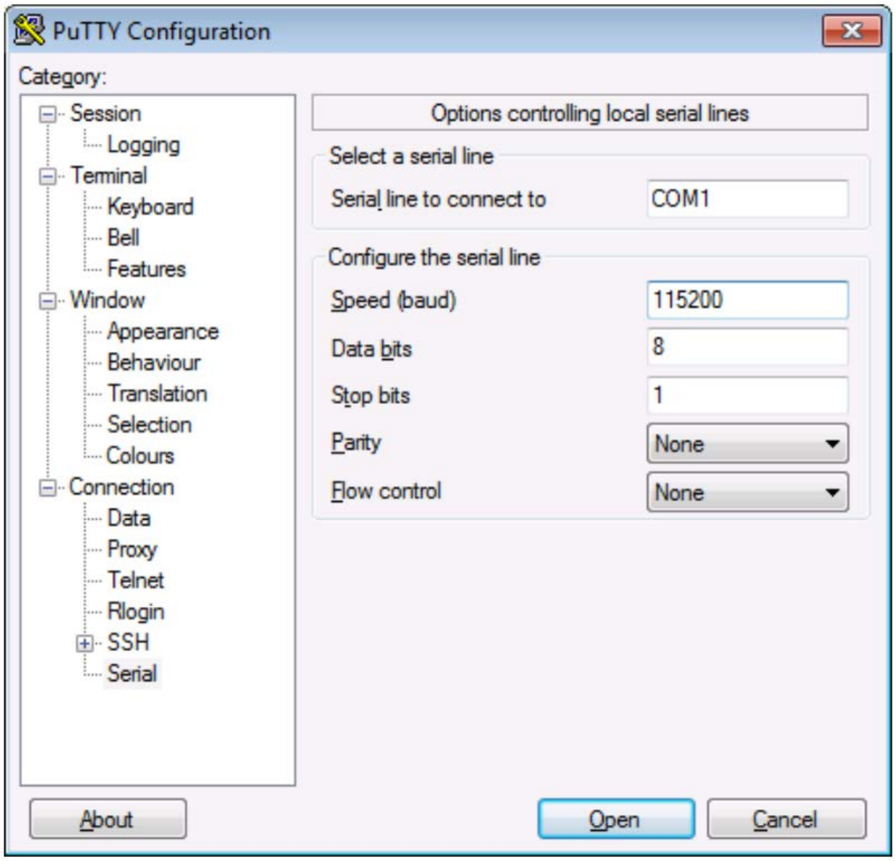


2. On the management laptop, open up the Putty Application and select a **Connection type** of **Serial** with a **Speed** of **115200.**

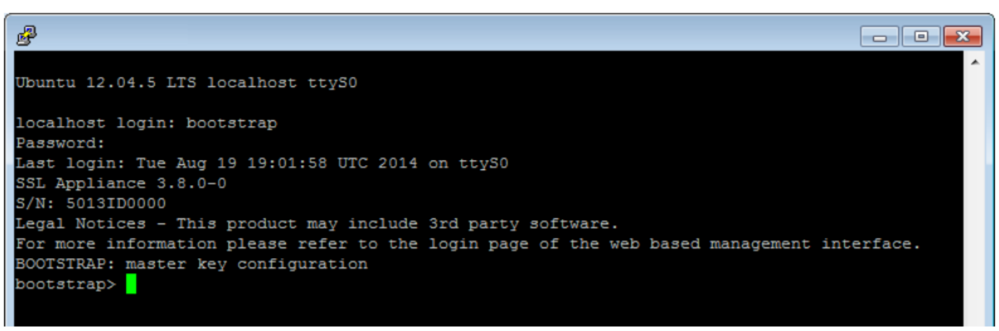


3. Navigate to the **Serial** Category on the bottom left side of the window.
4. Configure the serial connection to support the SSL Visibility Appliance's console speeds by selecting the following options:
   - **Speed (baud): 115200**
   - **Data bits: 8**
   - **Stop bits: 1**

- **Parity: None**
- **Flow Control: None**

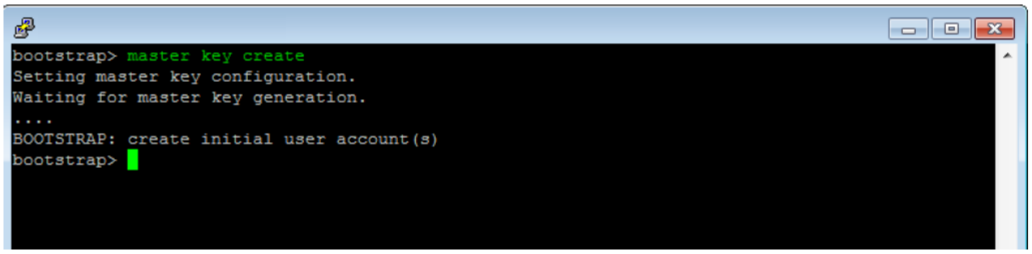


5. Login into the appliance by using the default credentials of:
   - **Username: bootstrap**
   - **Password: bootstrap**

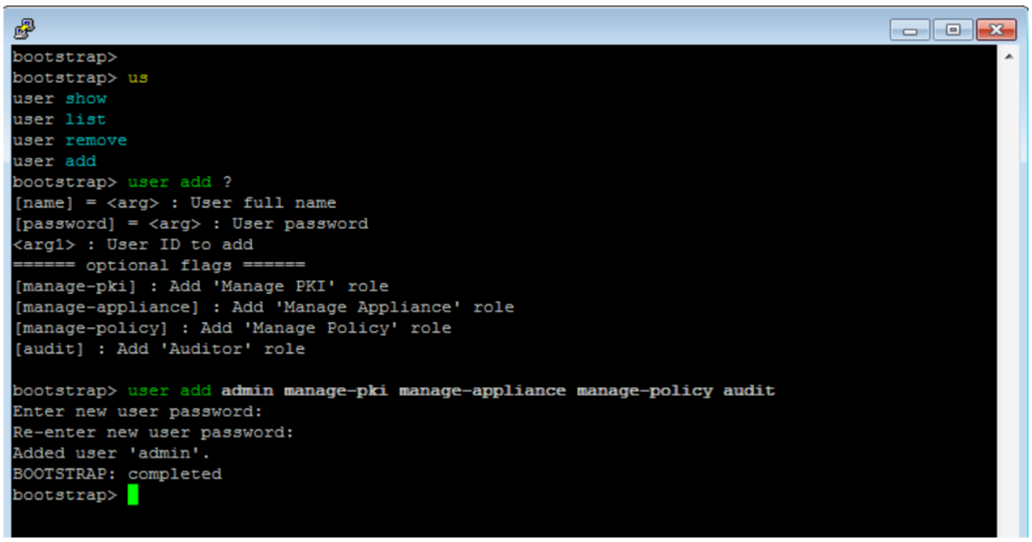


6. Next, create the master key by running the command:

```
master key create
```

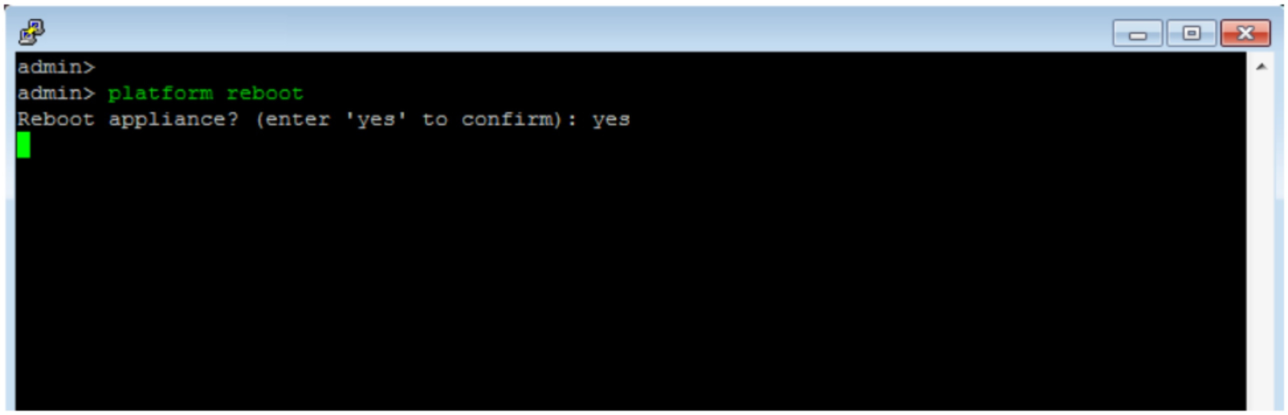


7. Create a new user by running the command:
`user add admin manage-pki manage-appliance manage-policy audit`

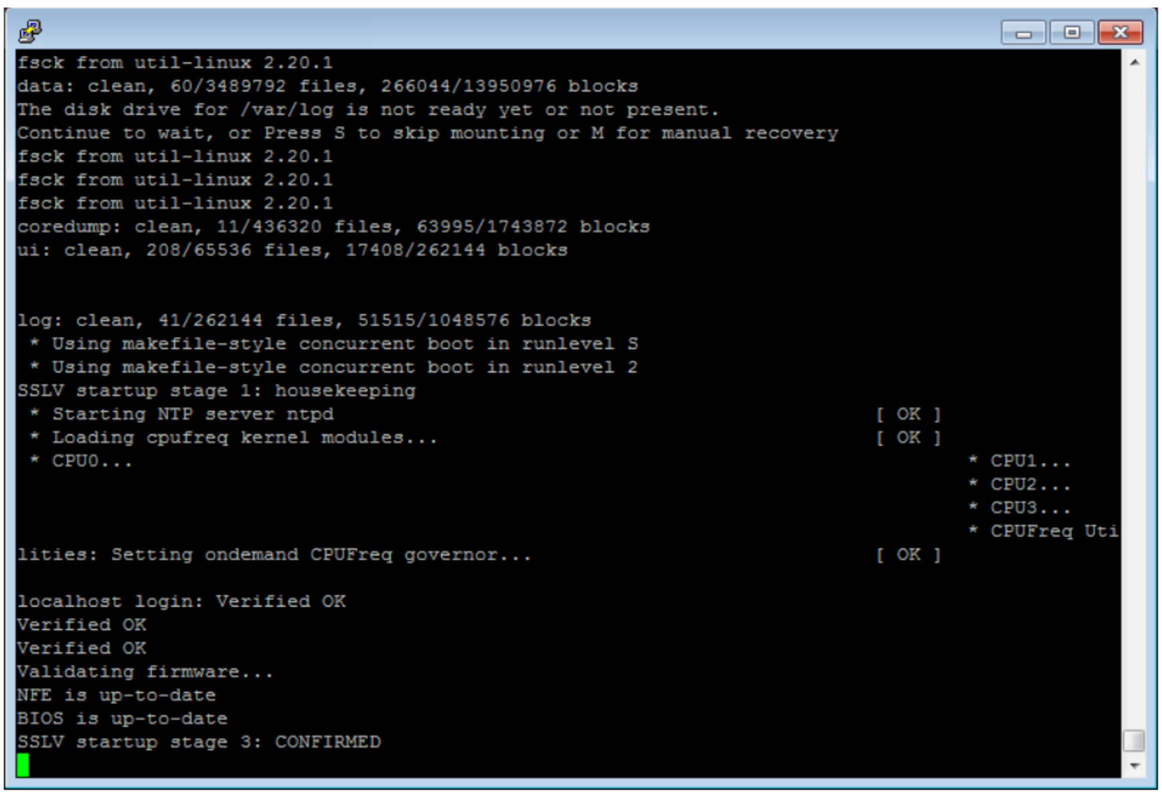


Tip: This step created a single admin user account with all four roles allocated to it. The only requirements for completing the bootstrap phase are that there is a user account with the Manage Appliance role and a user account with the Manage PKI role. These may be the same or different accounts. In most cases, creating a single account with all four roles is the simplest approach.

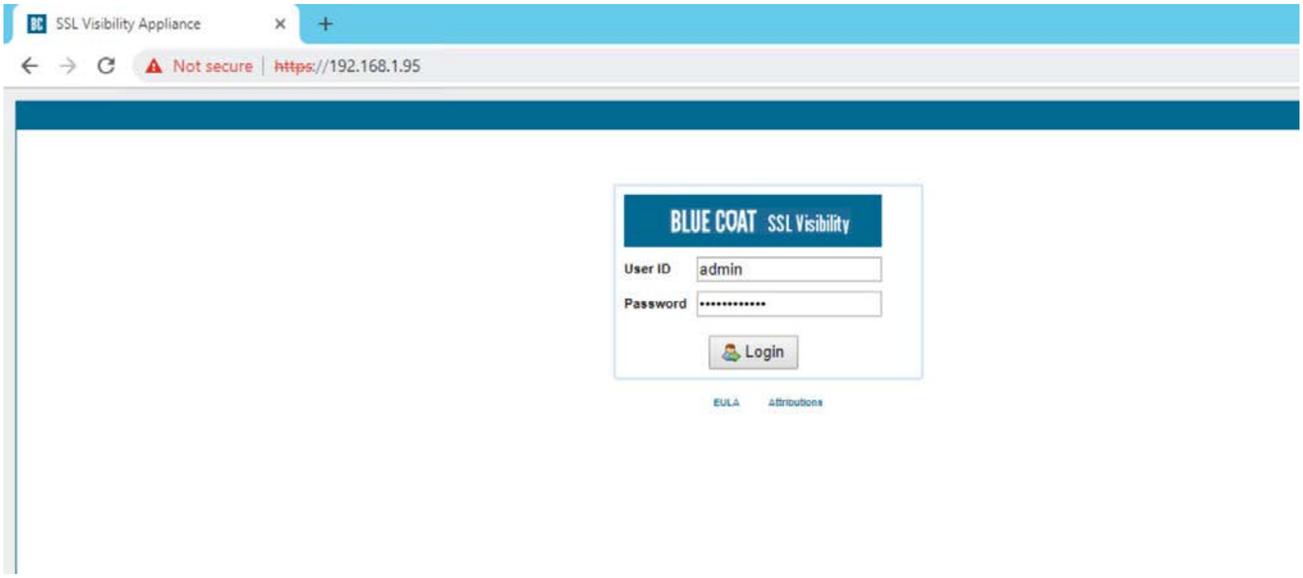
8. Run the following command to configure the management network interface with a static IP address:
`network set ip 192.168.1.95 netmask 255.255.255.0 gateway 192.68.1.1`
9. Reboot the system for the changes to take effect (confirm that you wish to reboot) with the following command: `platform reboot`



10. On reboot, confirm that the **"SSL Visibility startup stage 3: CONFIRMED"** is displayed as shown below.

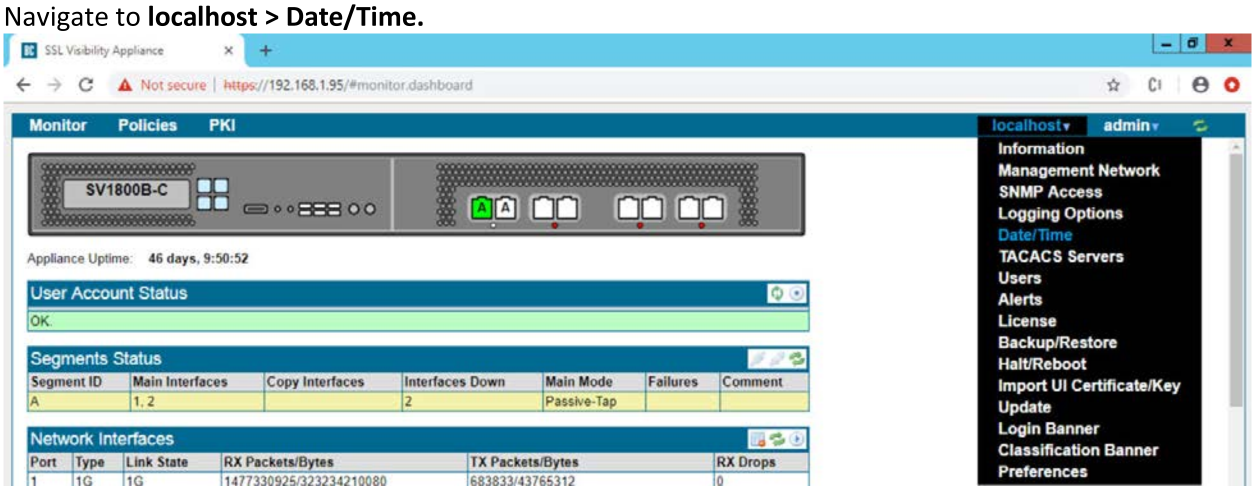


11. Confirm you can log in to the appliance via your browser. Log in via a web browser, using the format *https://192.168.1.95*. Log in with the username and password you created.

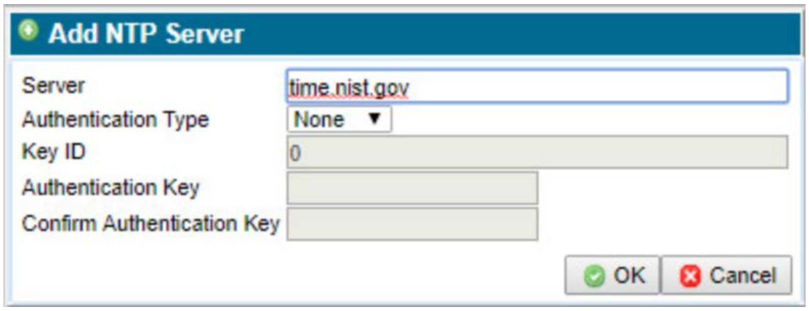


2.5.1.7 Date and Time (NTP)

1. To configure Date and Time, login into the WebUI by browsing to *https://192.168.1.95.*
2. Navigate to **localhost > Date/Time.**



3. Click on the Add button under NTP Servers.
4. In the server field type time.nist.gov and click **OK.**

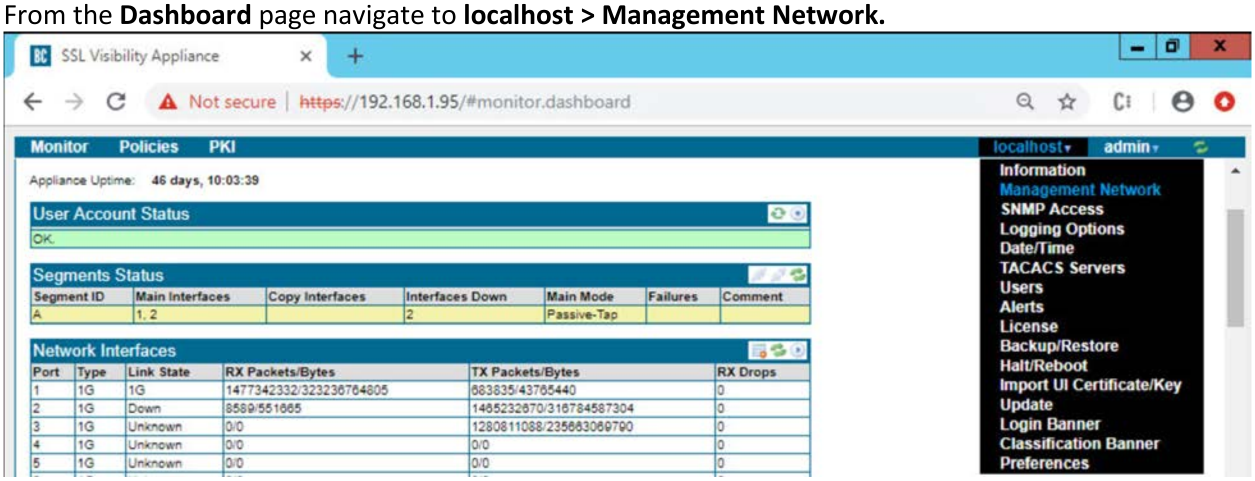


5. Click **Apply Changes** to save the new NTP server.

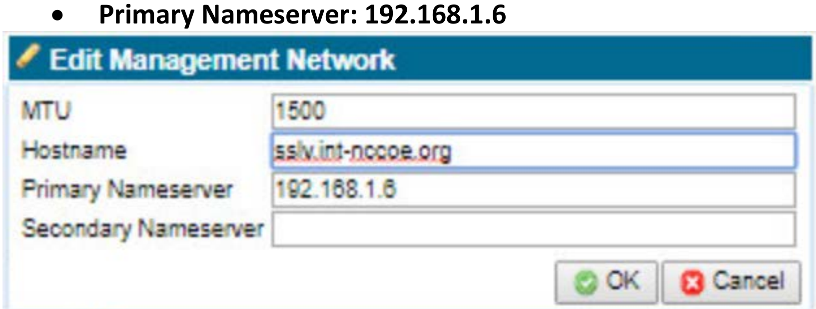
2.5.1.8 Additional Configuration

To add a host name and DNS for the SSL Visibility Appliance, perform the following steps:

1. Log in to the SSL Visibility by opening a web browser and navigating to *https://192.168.1.95.*
2. From the **Dashboard** page navigate to **localhost > Management Network.**



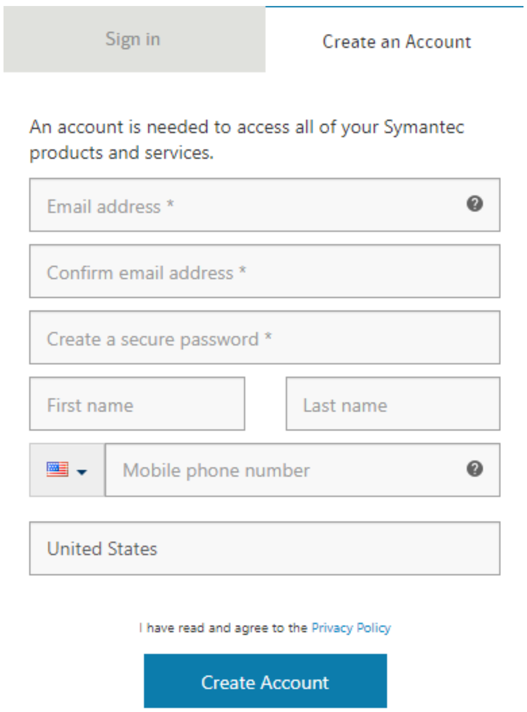
3. Click the **Edit** button  under the **Management Network** Field.
4. Enter the following information into the fields:
    - **MTU: 1500**
    - **Host Name: SSL Visibility.int-nccoe.org**
    - **Primary Nameserver: 192.168.1.6**



5. Click **Apply Changes.**
6. Click **Reboot** to restart the system and apply changes (required).

2.5.1.9 MySymantec Account Creation

1. To create a MySymantec Account, navigate to the following link: https://login.symantec.com/sso/idp/SAML2
2. Click the **Create an Account** tab.



3. Enter the requested information and click **Create Account.**

2.5.1.10 License the SSL Visibility Appliance

2.5.1.10.1 Download a Blue Coat License

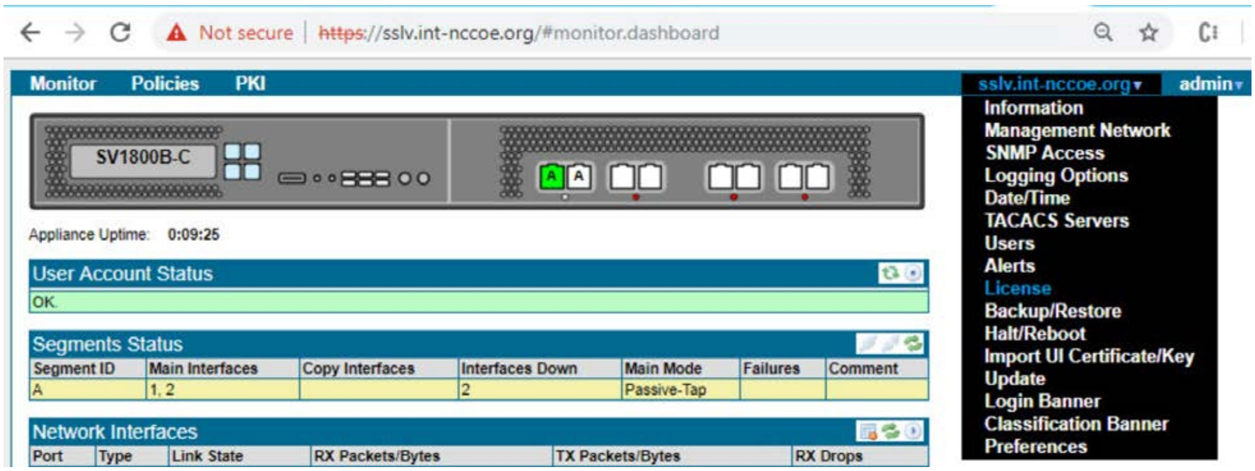
1. Using your BlueTouch Online account, log in to the Blue Coat Licensing Portal.

   (https://services.bluecoat.com/eservice_enu/licensing/register.cgi).

2. From the menu on the left side, select **SSL Visibility**, then select **License Download**.
3. When prompted, enter the serial number of your appliance, then press **Submit**.
4. Once the license is generated, press **Download License File** for the required SSL Visibility Appliance.

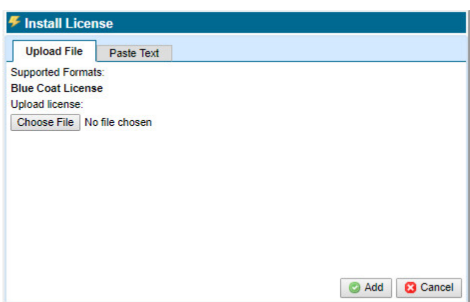
2.5.1.10.2 Install a Blue Coat License

1. Select **SSL Visibility.int-nccoe.org > License.**



2. Click the **Add** button  in the **License** field.

3. On the **Upload File** tab, use the **Choose File** button to browse to the license file location.



4. Click **Add**. You will see a confirmation message and the specific appliance platform model. The license is now installed, and all standard SSL Visibility Appliance features are operational.

2.5.2 Day 1: Product Integration Configuration

2.5.2.1 Prerequisites

1. Install version 3.x on the SSL Visibility Appliance.
2. Complete initial configuration as outlined in the Day 0 Section 2.5.1 above.
3. Required Ports, Protocols and Services:
   SSL Visibility 3.x uses the following ports while operating—allow these ports when setting up SSL Visibility:
   Inbound Connection to SSL Visibility Appliance

Table 18

Service	Port	Protocol	Configurable	Source	Description
WebUI Admin GUI	443	TCP	No	User client	Management Interface WebUI service
SSH Admin CLI	22	TCP	No	User client	SSH Admin CLI service
Symantec/ Blue Coat License	443	HTTPS	No	License server	Symantec/Blue Coat license service
SNMP management	161	UDP	No	User client	SNMP agent for SNMP management access
NTP	123	UDP	No	NTP server	NTP time synchronization service
DHCP	68	UDP	No	DHCP server	DHCP service
Remote Diagnostics Facility (RDF)	2024	TCP	No	RDF	Can be opened for support requests; normally closed

Outbound Connections from SSL Visibility Appliance

Table 19

Service	Port	Protocol	Configurable	Destination	Description
SMTP/Secure SMTP	25, 465, 587, 525, 2526 *	TCP	Yes	SMTP server	SMTP alerts
Syslog	514, 601 * 6514 * 514 *	TCP TLS UDP	Yes	Syslog server	Remote syslog server
DNS	53	TCP UDP	No	DNS server	Domain Name System service
SNMP Trap	162	UDP	No	SNMP Trap receiver	SNMP traps
Host Categorization (BCWF)	443	HTTPS	No	Symantec	Host categorization database
HSM	443	HTTPS	No	HSM appliance	HSM authentication and requests
TACACS+	49	TCP	Yes	TACACS server	TACACS+ authentication
NTP	123	UDP	No	NTP server list	Synchronization to customer-configured NTP server
DHCP	67	UDP	No	DHCP server	DHCP service
Diagnostics Upload	443	HTTPS	No	Symantec	Diagnostics upload service

*Common Values For this Port

Required URLs

Ensure connectivity from SSL Visibility to the following URLs:

Table 20

URL	Port	Protocol	Description
abrca.bluecoat.com	443	HTTPS TCP	Symantec CA
*.es.bluecoat.com	443	HTTPS TCP	License, validation, and subscription services
appliance.bluecoat.com	443	HTTPS TCP	Trust package downloads
upload.bluecoat.com	443	HTTPS TCP	Upload diagnostic reports to Symantec support

2.5.2.2 Venafi Integration

Venafi TPP was used to copy known server key and certificates to the SSL Visibility appliance for TLS decryption.

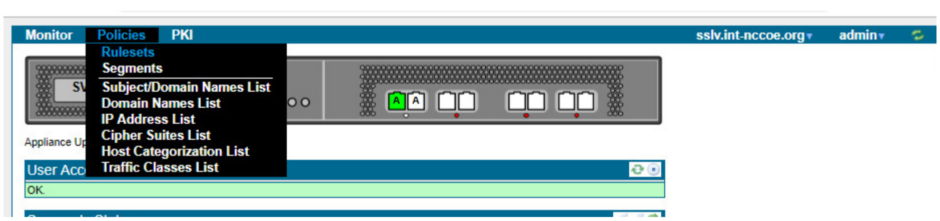
For information on integration with Venafi TPP, see Section: 2.6.13.9.

2.5.2.3 Ruleset Creation

To ensure your SSL Visibility Appliance is connected and configured properly, create a basic ruleset to test that traffic isn't getting blocked. To perform this test, create a ruleset with a Catch All Action of Cut Through.

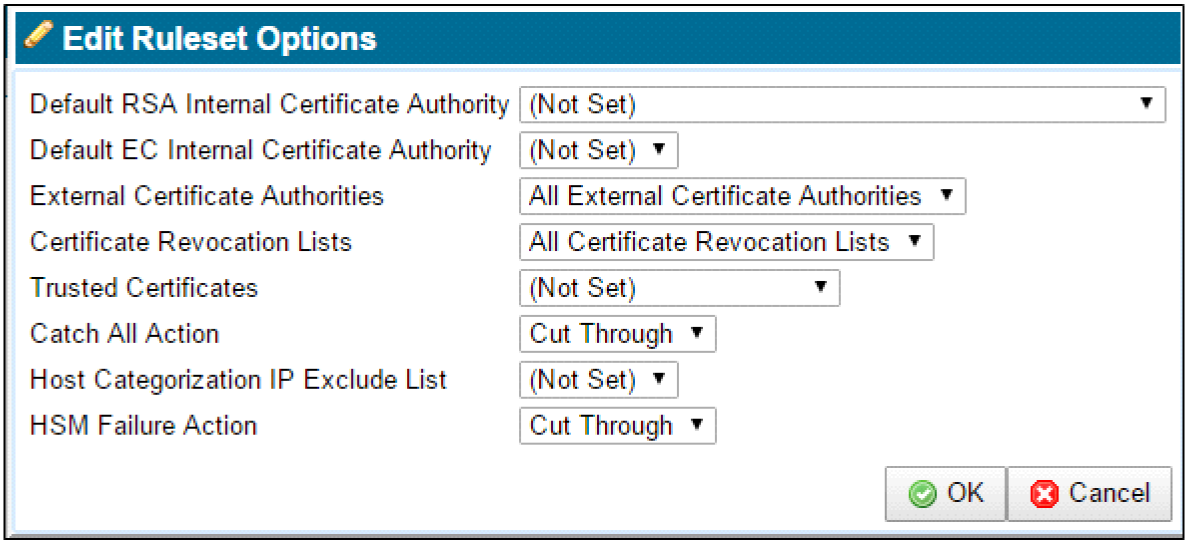
Note: At least one rule must be added to the ruleset for SSL Visibility Appliance to start processing SSL traffic.

1. Select **Policies > Rulesets.**



2. In the **Rulesets** panel, click the **Add**  icon.
3. In the **Add Ruleset** window, enter a name for the ruleset and click **OK.**

4. In the **Ruleset Options** panel, click the **Edit** icon.

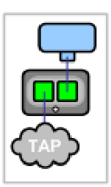


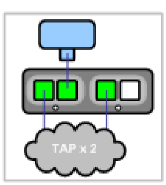
5. Confirm the **Catch All Action** is **Cut Through**.
6. **Apply** the Policy Changes.

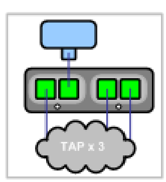
2.5.2.4 Segment Creation

Note: Before creating the segment, determine your deployment mode and create a ruleset for the segment.

The following pictures demonstrate various passive tap deployment types:

(i). 

(ii). 

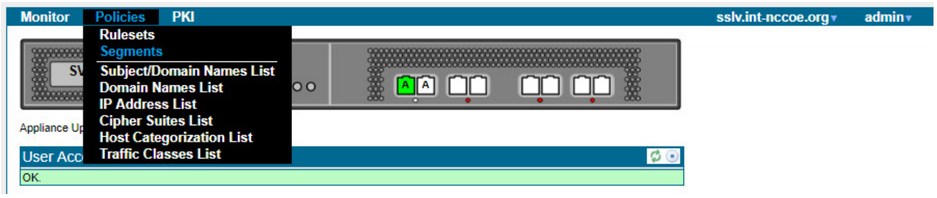
(iii). 

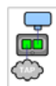
For purpose of this document we used (i).

Note: The latter two tap modes combine traffic from two or three network taps onto a single SSL Visibility Appliance segment. These ports are called *aggregation ports*.

2.5.2.4.1 Add a Segment

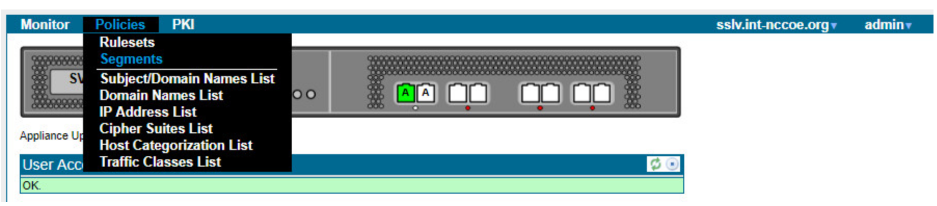
1. Select **Policies > Segments**.



2. Click the **Add**  icon in the **Segments** field.
3. Click **Edit** to select the Mode of Operation.
4. For Mode of Operation, choose  **Passive Tap** mode.
5. Click **OK**.
6. Select the **Ruleset** you previously created.
7. Choose the desired **Session Log Mode**.
8. Enter a brief description of the segment in the **Comments** box.
9. Click **OK**. The new segment appears in the *Segments* panel.
10. **Apply** the Policy Changes.

2.5.2.4.2 Activate a Segment

1. Select **Policies > Segments**.



2. In the **Segments** panel, select the segment to activate.
3. Click the **Activate**  icon. The Segment Activation window displays.

   Note: During segment activation, a series of screens appear that allow you to select the ports the segment will use, and any copy ports and modes where the copy ports will operate. Connect any copy ports to your passive security devices (for example, Symantec DLP Network Monitor, Security Analytics, or an IDS).

4. Follow the prompts. Once the segment is active, the system dashboard displays a green background for the segment, and there are entries under Main Interfaces and Copy Interfaces (if applicable to your deployment).

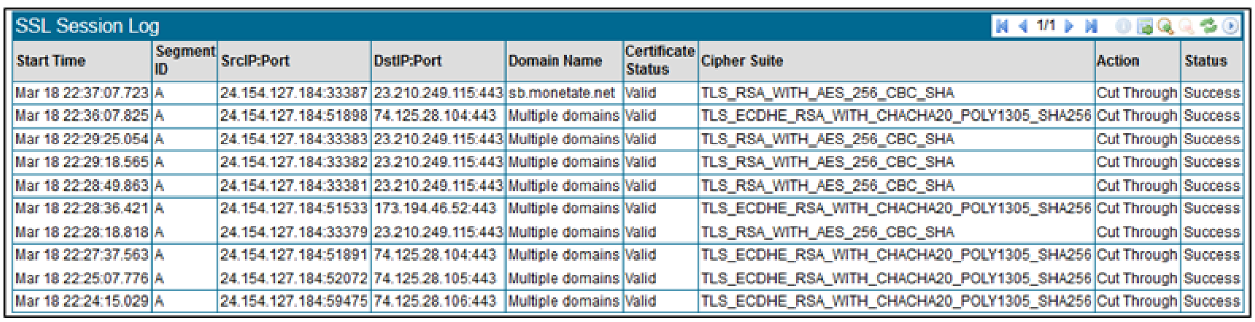
5. **Apply** the Policy Changes.

2.5.2.5 Verification

This section walks through verifying that the SSL Visibility is seeing SSL traffic without blocking it (cut through).

1. To see a list of recent SSL sessions, select **Monitor > SSL Session Log**.

2. Look for the domains of the servers that were accessed, and observe the value in the Action column. Since the initial rule you created cuts through all traffic, the Action should say **Cut Through** for all sessions.


SSL Session Log

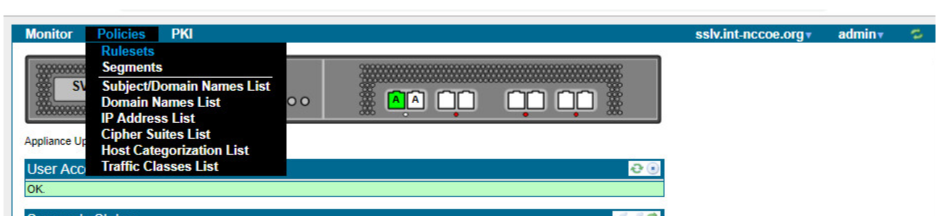
Start Time	Segment ID	SrcIP:Port	DstIP:Port	Domain Name	Certificate Status	Cipher Suite	Action	Status
Mar 18 22:37:07.723	A	24.154.127.184:33387	23.210.249.115:443	sb.monetate.net	Valid	TLS_RSA_WITH_AES_256_CBC_SHA	Cut Through	Success
Mar 18 22:36:07.825	A	24.154.127.184:51898	74.125.28.104:443	Multiple domains	Valid	TLS_ECDHE_RSA_WITH_CHACHA20_POLY1305_SHA256	Cut Through	Success
Mar 18 22:29:25.054	A	24.154.127.184:33383	23.210.249.115:443	Multiple domains	Valid	TLS_RSA_WITH_AES_256_CBC_SHA	Cut Through	Success
Mar 18 22:29:18.565	A	24.154.127.184:33382	23.210.249.115:443	Multiple domains	Valid	TLS_RSA_WITH_AES_256_CBC_SHA	Cut Through	Success
Mar 18 22:28:49.863	A	24.154.127.184:33381	23.210.249.115:443	Multiple domains	Valid	TLS_RSA_WITH_AES_256_CBC_SHA	Cut Through	Success
Mar 18 22:28:36.421	A	24.154.127.184:51533	173.194.46.52:443	Multiple domains	Valid	TLS_ECDHE_RSA_WITH_CHACHA20_POLY1305_SHA256	Cut Through	Success
Mar 18 22:28:18.818	A	24.154.127.184:33379	23.210.249.115:443	Multiple domains	Valid	TLS_RSA_WITH_AES_256_CBC_SHA	Cut Through	Success
Mar 18 22:27:37.563	A	24.154.127.184:51891	74.125.28.104:443	Multiple domains	Valid	TLS_ECDHE_RSA_WITH_CHACHA20_POLY1305_SHA256	Cut Through	Success
Mar 18 22:25:07.776	A	24.154.127.184:52072	74.125.28.105:443	Multiple domains	Valid	TLS_ECDHE_RSA_WITH_CHACHA20_POLY1305_SHA256	Cut Through	Success
Mar 18 22:24:15.029	A	24.154.127.184:59475	74.125.28.106:443	Multiple domains	Valid	TLS_ECDHE_RSA_WITH_CHACHA20_POLY1305_SHA256	Cut Through	Success

2.5.2.5.1 Create a Rule to Test Decryption

To test the SSL Visibility Appliance is decrypting SSL traffic, add a rule that decrypts everything from a specific source IP (e.g., your laptop).

Note: At least one rule must be added to the ruleset for SSL Visibility Appliance to start processing SSL traffic.

1. Select **Policies > Rulesets**.



2. In the **Rulesets** panel, select the ruleset that was previously created.

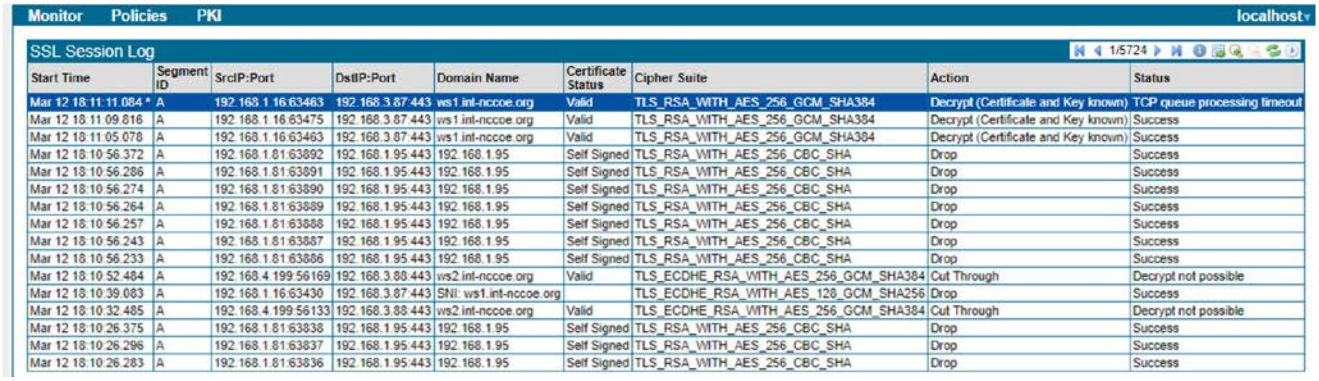
3. In the **Rules** panel, click the **Insert** icon to add a new rule. The **Insert Rule** dialog displays.
4. For Action, select **Decrypt (Certificate and Key Known)**.
5. Select one of the following:
    - If you imported one certificate, select **Known Certificate with Key,** and choose the certificate you imported.
    - If you imported multiple certificates, select **Known Certificates with Keys and All Known Certificates with Keys.**
6. For **Source IP**, enter the IP address of your computer.
7. Click **OK**.
8. **Apply** the Policy Changes.
9. Next Step: Use the SSL Session Log to verify that the SSL Visibility Appliance is decrypting properly.

2.5.2.5.2 Verify Decryption

View the SSL Session log to test, and verify the SSL Visibility Appliance is decrypting traffic according to the rules you created.

1. Access a variety of websites or internal SSL servers. If you have created policies for specific host categories, domains, IP addresses, etc., visit websites that test these policies.
2. To see a list of recent SSL sessions, select **Monitor > SSL Session Log**.
3. Look for the domains of the websites/servers you visited, and observe the value in the Action column. Is the value you expected listed? For example, if you wanted the SSL Visibility Appliance *not* to decrypt a particular type of traffic, does the Action say Cut Through? For sessions designated as decrypted, does the Action say Decrypt? If unexpected values appear, review your policies.

Note: When a session is decrypted, the Action column will show either *Resign Certificate* (if the deployment is using the certificate resigning method) or *Certificate and Key Known* (if you have imported known certificates and keys).



Start Time	Segment ID	SrcIP:Port	DstIP:Port	Domain Name	Certificate Status	Cipher Suite	Action	Status
Mar 12 18:11:11.084 *	A	192.168.1.16:63463	192.168.3.87:443	ws1.int-nccoe.org	Valid	TLS_RSA_WITH_AES_256_GCM_SHA384	Decrypt (Certificate and Key known)	TCP queue processing timeout
Mar 12 18:11:09.816	A	192.168.1.16:63475	192.168.3.87:443	ws1.int-nccoe.org	Valid	TLS_RSA_WITH_AES_256_GCM_SHA384	Decrypt (Certificate and Key known)	Success
Mar 12 18:11:05.078	A	192.168.1.16:63463	192.168.3.87:443	ws1.int-nccoe.org	Valid	TLS_RSA_WITH_AES_256_GCM_SHA384	Decrypt (Certificate and Key known)	Success
Mar 12 18:10:56.372	A	192.168.1.81:63892	192.168.1.95:443	192.168.1.95	Self Signed	TLS_RSA_WITH_AES_256_CBC_SHA	Drop	Success
Mar 12 18:10:56.286	A	192.168.1.81:63891	192.168.1.95:443	192.168.1.95	Self Signed	TLS_RSA_WITH_AES_256_CBC_SHA	Drop	Success
Mar 12 18:10:56.274	A	192.168.1.81:63890	192.168.1.95:443	192.168.1.95	Self Signed	TLS_RSA_WITH_AES_256_CBC_SHA	Drop	Success
Mar 12 18:10:56.264	A	192.168.1.81:63889	192.168.1.95:443	192.168.1.95	Self Signed	TLS_RSA_WITH_AES_256_CBC_SHA	Drop	Success
Mar 12 18:10:56.257	A	192.168.1.81:63888	192.168.1.95:443	192.168.1.95	Self Signed	TLS_RSA_WITH_AES_256_CBC_SHA	Drop	Success
Mar 12 18:10:56.243	A	192.168.1.81:63887	192.168.1.95:443	192.168.1.95	Self Signed	TLS_RSA_WITH_AES_256_CBC_SHA	Drop	Success
Mar 12 18:10:56.233	A	192.168.1.81:63886	192.168.1.95:443	192.168.1.95	Self Signed	TLS_RSA_WITH_AES_256_CBC_SHA	Drop	Success
Mar 12 18:10:52.484	A	192.168.4.199:56169	192.168.3.88:443	ws2.int-nccoe.org	Valid	TLS_ECDHE_RSA_WITH_AES_256_GCM_SHA384	Cut Through	Decrypt not possible
Mar 12 18:10:39.083	A	192.168.1.16:63430	192.168.3.87:443	SNI: ws1.int-nccoe.org		TLS_ECDHE_RSA_WITH_AES_128_GCM_SHA256	Drop	Success
Mar 12 18:10:32.485	A	192.168.4.199:56133	192.168.3.88:443	ws2.int-nccoe.org	Valid	TLS_ECDHE_RSA_WITH_AES_256_GCM_SHA384	Cut Through	Decrypt not possible
Mar 12 18:10:26.375	A	192.168.1.81:63838	192.168.1.95:443	192.168.1.95	Self Signed	TLS_RSA_WITH_AES_256_CBC_SHA	Drop	Success
Mar 12 18:10:26.296	A	192.168.1.81:63837	192.168.1.95:443	192.168.1.95	Self Signed	TLS_RSA_WITH_AES_256_CBC_SHA	Drop	Success
Mar 12 18:10:26.283	A	192.168.1.81:63836	192.168.1.95:443	192.168.1.95	Self Signed	TLS_RSA_WITH_AES_256_CBC_SHA	Drop	Success

2.5.2.5.3 Other Ways to Learn About this Deployment Method

Download a PDF (https://origin-symwisedownload.symantec.com/resources/webguides/SSL Visibility/SSL Visibilitya_first_steps/Content/PDFs/Deployment6.pdf)

View a video tutorial (https://www.youtube.com/watch?v=qxSDDXhE_B8&feature=youtu.be)

2.5.3 Day N: Ongoing Security Management and Maintenance

2.5.3.1 Alerting & Monitoring

2.5.3.1.1 Alerts

Use the Alerts panels to configure the email details the system will use to send out alerts, monitor events, and assess the conditions where an alert is generated. Click **Edit** to bring up the upper Edit Alert Mail Configuration window to construct details of the email system.

2.5.3.1.2 SNMP Support

The SSL Visibility Appliance supports the more secure SNMP version 3, which maintains authentication and encryption for SNMP monitoring. Symantec recommends disabling SNMP versions 1 and 2c, and the default options of using AES for encryption, and SHA for authentication for SNMP version 3.

For more details, see the SSL Visibility Appliance 3.x Administration & Deployment Guide https://symwisedownload.symantec.com//resources/sites/SYMWISE/content/live/DOCUMENTATION/11000/DOC11119/en_US/SSL VISIBILITY_Admin_31231.pdf?__gda__=1556286966_fb942bb8532ca7c1a67d0e2720faa76d

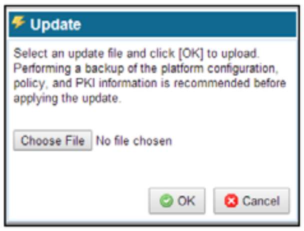
2.5.3.1.3 Logging Options

Use **Platform Management (SSL Visibility-int.nccoe.org) > Logging Options** to enable or disable WebUI TLS logging and to configure remote syslog servers.

Use Logging Options to include Web UI TLS trusted channel establishment and termination logs in the System Log. These events are not included in the System Log by default.

2.5.3.2 Software Update

Use the **Update** menu item to load and apply a file that will update the system software. Update files are digitally signed and checked before being applied to the system. An invalid update file will not be applied.



Click **Choose File** to open a window where you browse the system and select the update file to use. Click **OK**, and the file is checked; if valid, it is copied to the system and applied.

2.6 Venafi Trust Protection Platform (TPP)

2.6.1 Prerequisites

Venafi TPP requires the following in order to be installed:

- Windows Server
- Microsoft SQL Server Database
- Hardware Security Module (if one will be used)
- Microsoft .NET Framework

2.6.2 Installation

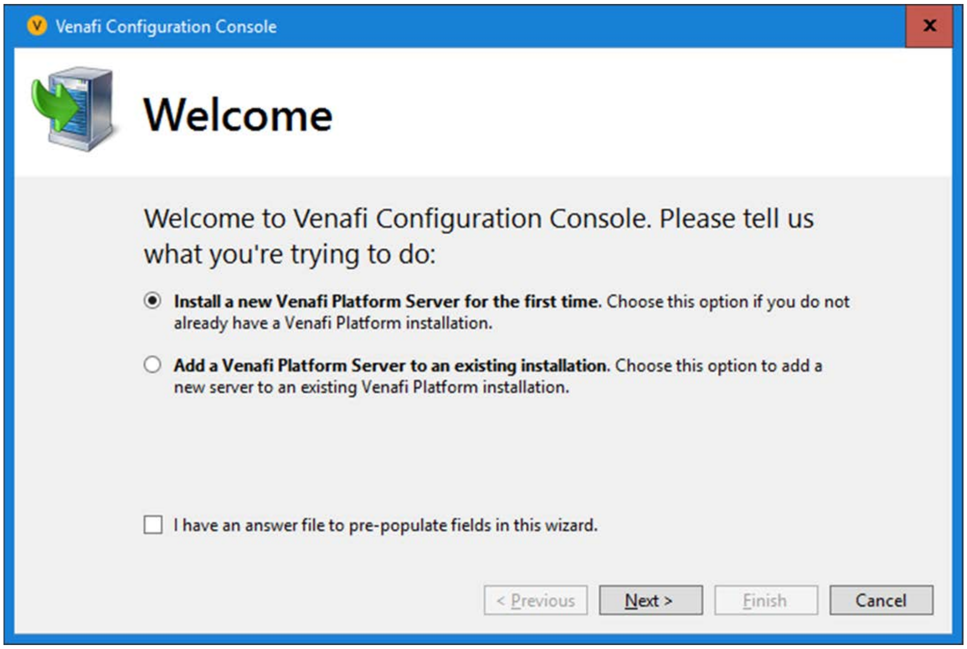
We installed Venafi TPP on Microsoft Windows Server 2012. Before starting the Venafi TPP installation, make sure you have configured your database and HSM.

The installation can be automated via a configuration file or manually performed with an installation wizard. The automated installation configuration file for installation into the production environment is typically created based on the Venafi TPP deployment in the DEV testing environment and placed in the user acceptance environment to formally test it. We recommend using the automated installation to reduce the possibility of errors during the installation into the production environment.

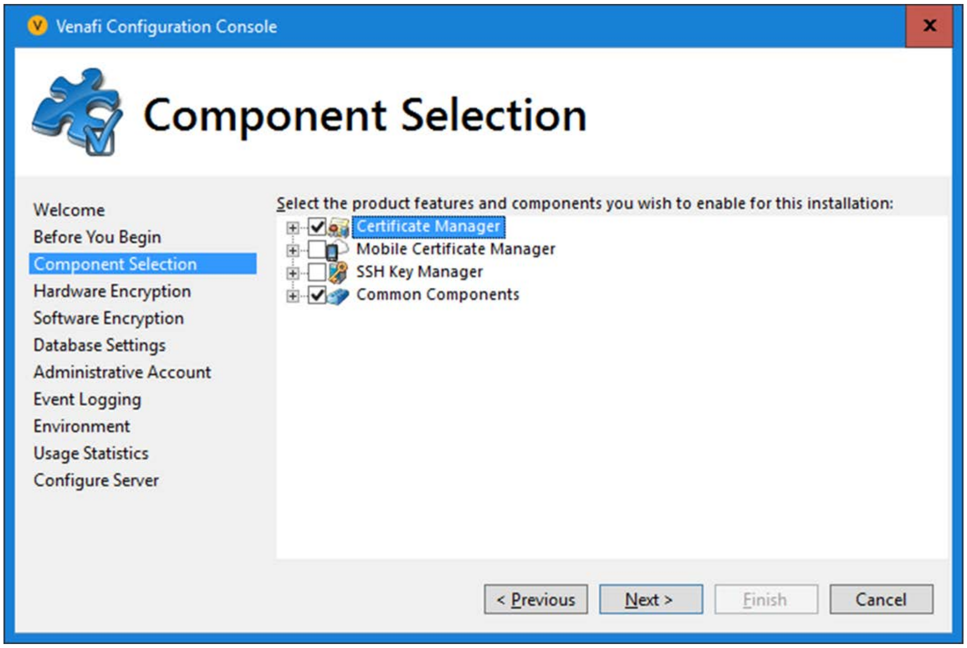
Because we were only configuring a single server in our lab environment, we manually installed and configured the product using the wizard. To install the Venafi TPP binaries and supporting files using the wizard, follow steps 1-7 in the *Venafi Trust Protection Platform Installation Guide* chapter titled "Installing using the Venafi Configuration Console wizard."

Following step 7, the Venafi Configuration Console is automatically launched and is explained in steps 8-22 where specific integrations with the HSM and database are performed. We performed the following steps in our implementation:

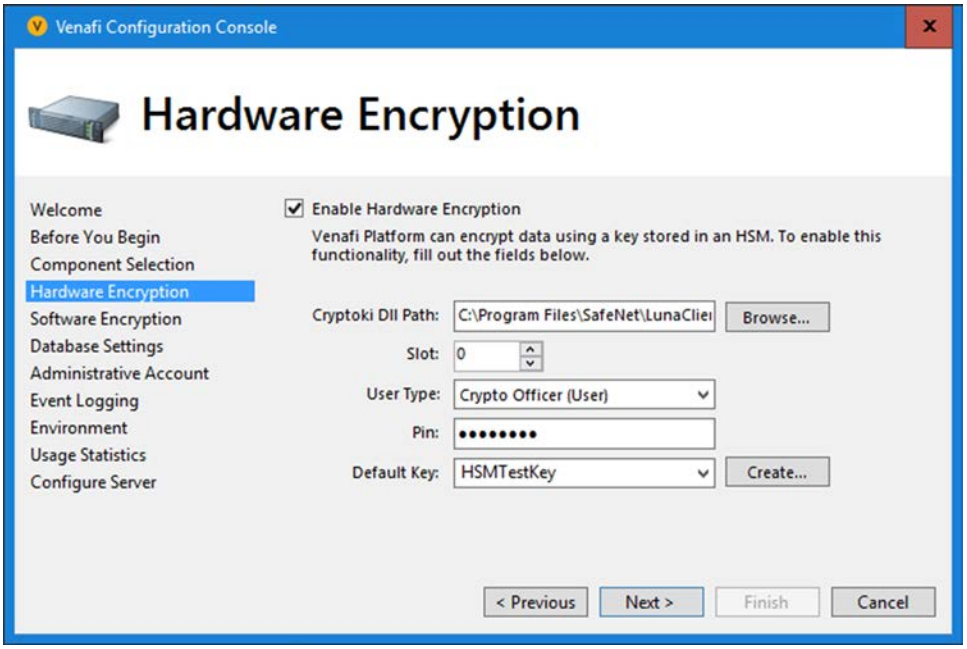
1. At the prompt for first time or existing installation, select “first-time installation.”



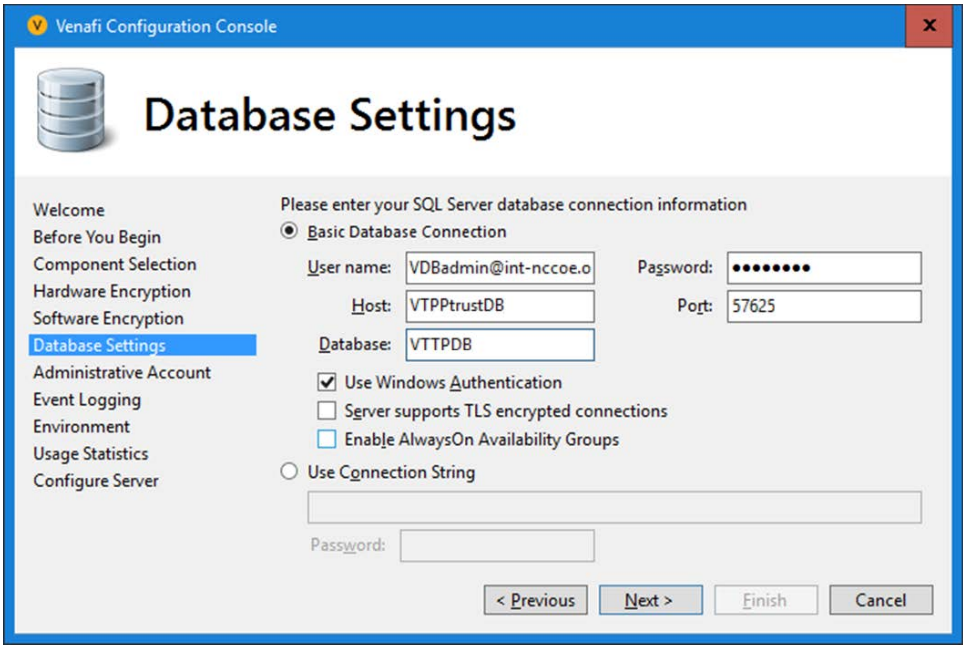
2. The Venafi Certificate Manager manages TLS server certificates, so it was selected. The Mobile Certificate and SSH Key Managers were not enabled.



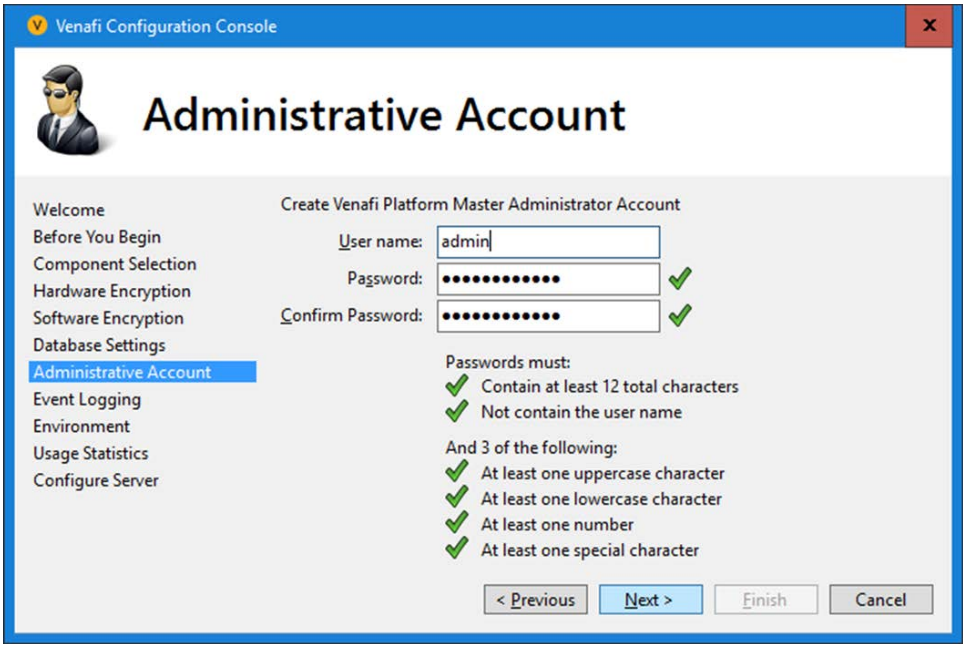
3. We recommend using an HSM with Venafi TPP to protect the symmetric key that encrypts private keys and credentials in the Venafi TPP database. In our implementation, we integrated with the SafeNet AT HSM. We entered the following configuration:



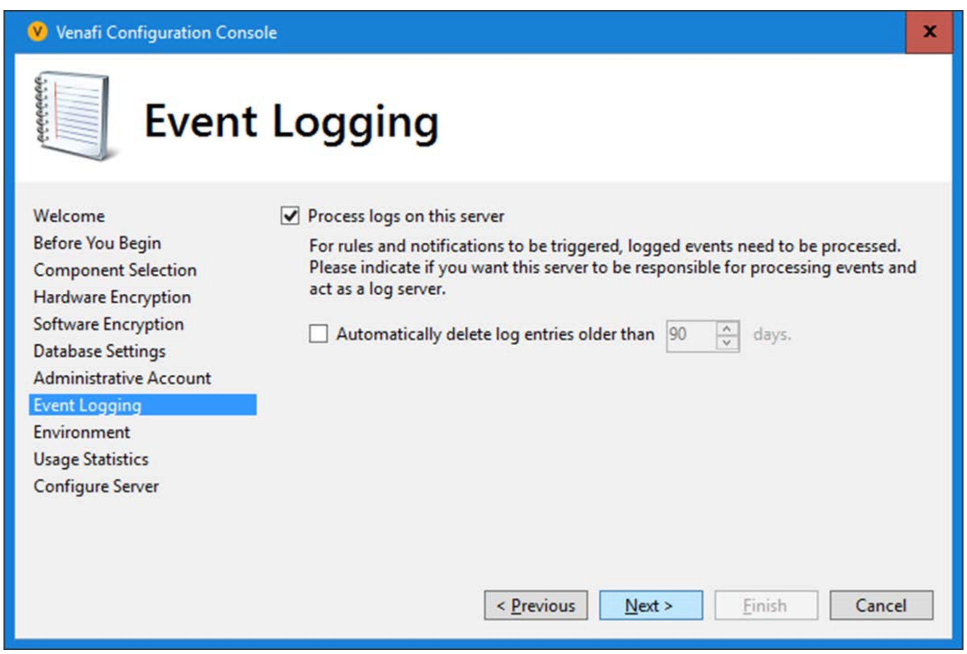
4. Windows authentication was used to authenticate to Microsoft SQL Server from Venafi TPP. Windows authentication is recommended, because it consolidates user account management, including control of password rules, failed logins, etc.



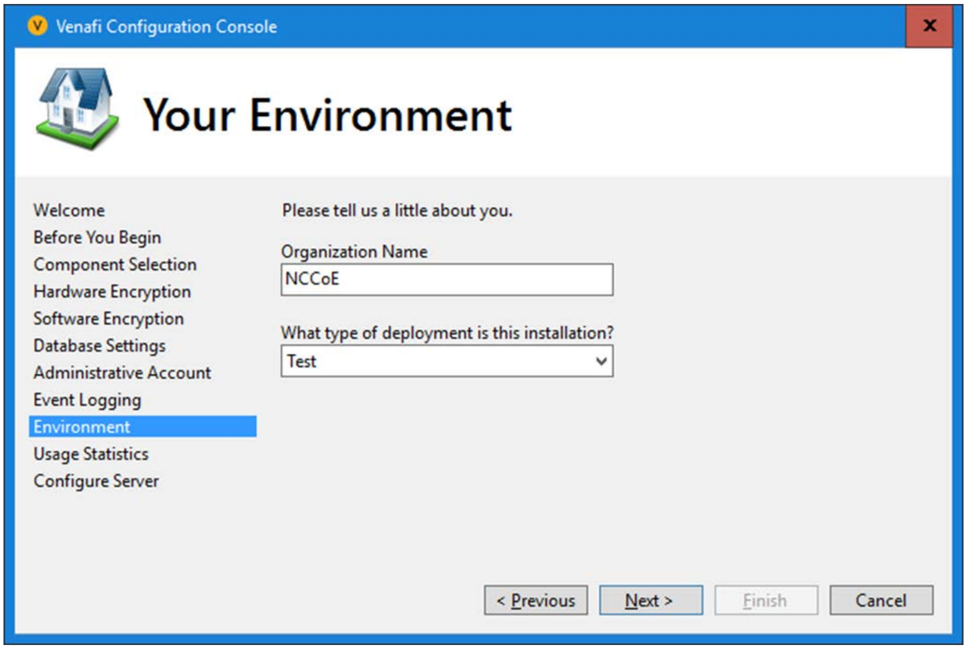
5. The initial Master Administrator account username was set to “admin,” and the password was also set.



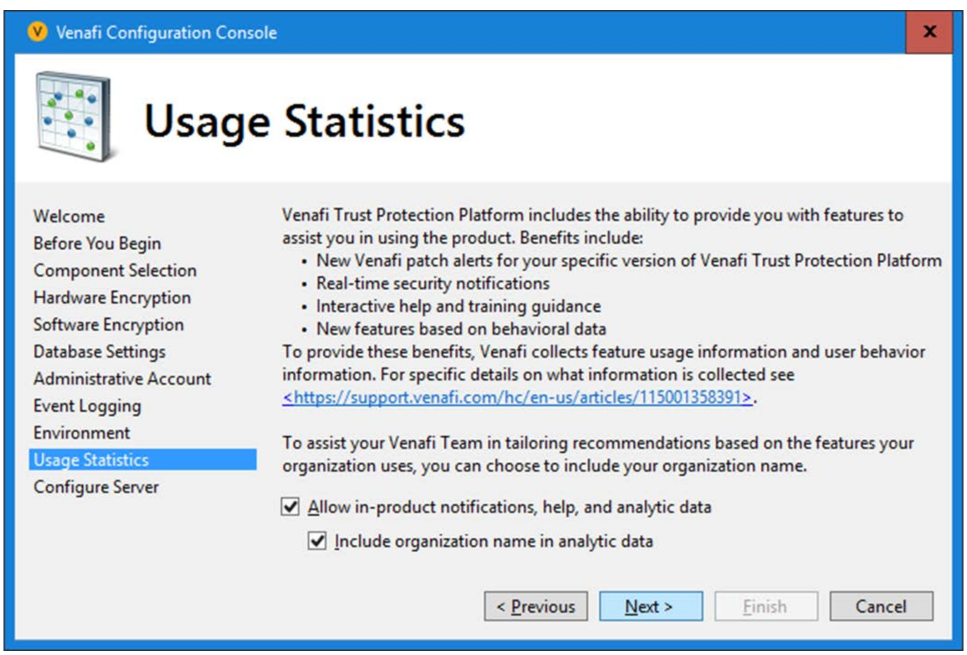
6. The Venafi TPP server was configured to process logs, as it was the only server in the environment.



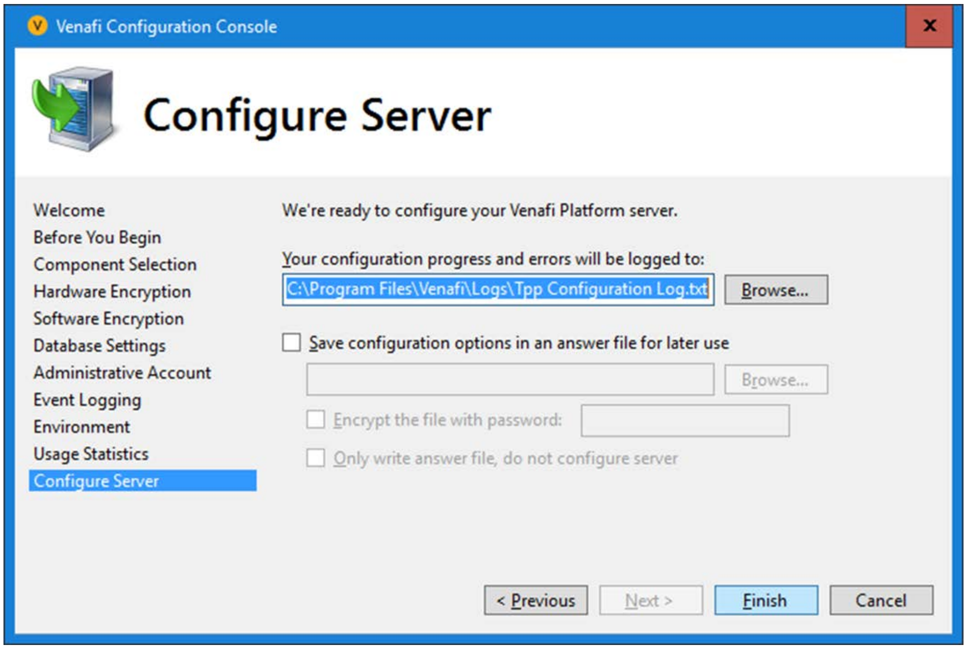
7. The organization name was set to “NCCoE”; the environment was set to “Test.”



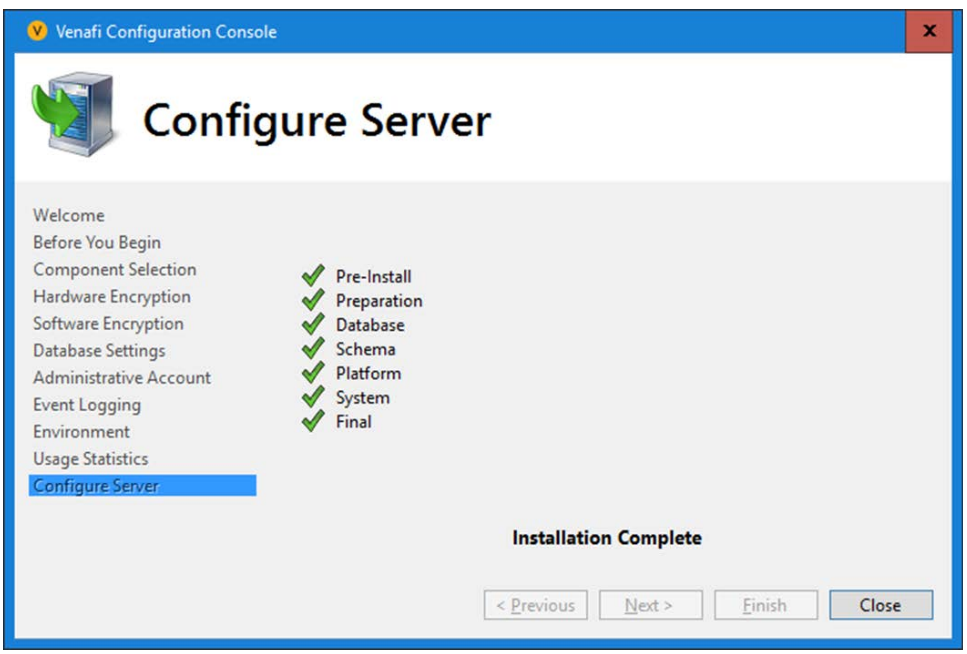
8. The collection of usage statistics was enabled.



9. The default log file location was used.



10. The Finish button was selected, and the configuration of the Venafi TPP server was completed successfully.



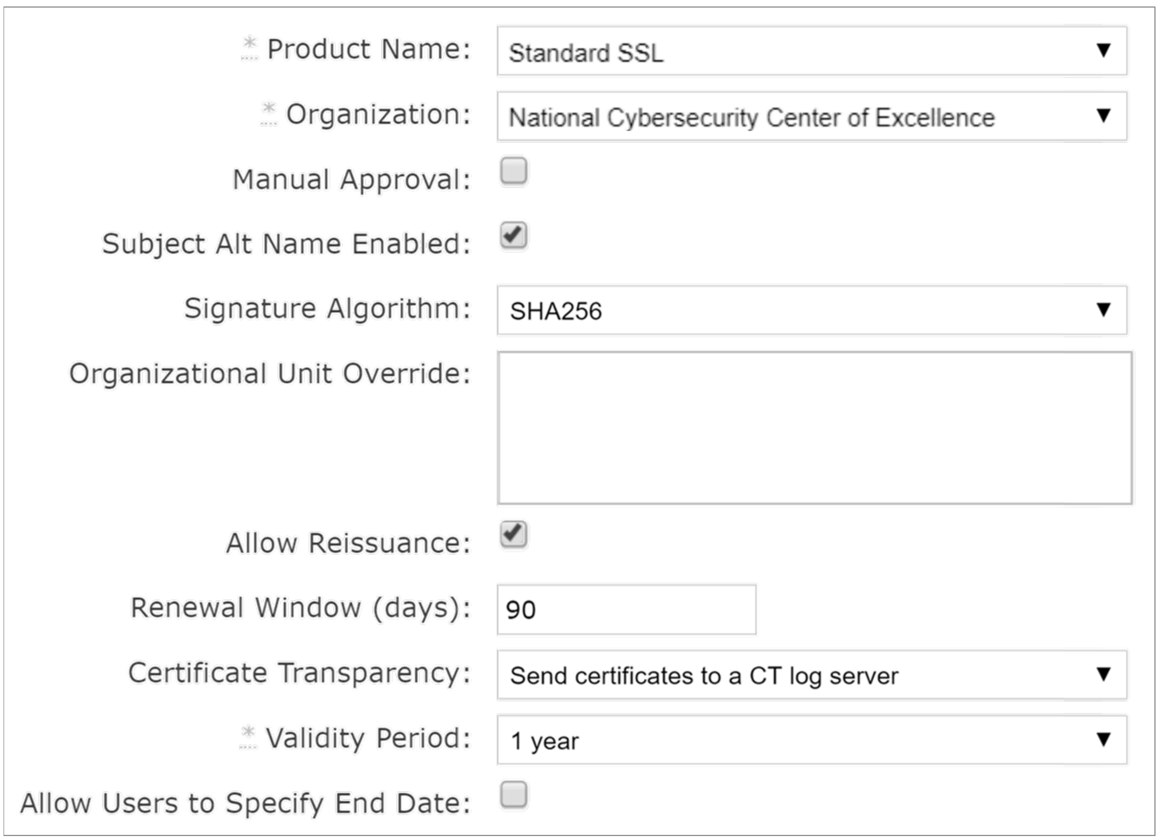
2.6.3 CA Integration

In our implementation, we integrated Venafi TPP with two CAs: DigiCert was used for publicly trusted certificates, and Active Directory Certificate Services for internally trusted certificates.

2.6.3.1 DigiCert

To configure integration with DigiCert so that Venafi TPP can automatically enroll for and retrieve certificates, follow the instructions in the "DigiCert CertCentral" section of the *Venafi Trust Protection Platform Certificate Authority and Hosting Platform Integration Guide*.

In our implementation, we used DigiCert Multi-SAN SSL certificates. The following configuration was used:

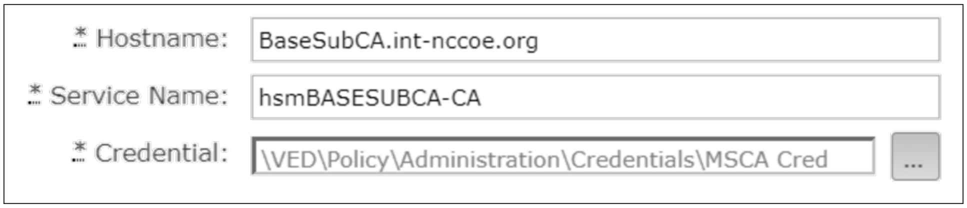


2.6.3.2 Active Directory Certificate Services

We used Microsoft ADCS to issue certificates to TLS servers inside the lab firewall. To configure integration with ADCS so Venafi can automatically enroll for and retrieve certificates, follow the instructions in the "Microsoft Active Directory Certificate Services (ADCS) - Enterprise and Standalone—

CA template configuration" section of the *Venafi Trust Protection Platform Certificate Authority and Hosting Platform Integration Guide*.

In our implementation, we configured the host name, service name, and credential information in Venafi TPP to access the ADCS Issuing CA:



In our implementation, a certificate template named "VenafiRSAWebServer" was configured in ADCS to issue TLS server certificates. The CA template object we used in Venafi TPP to request certificates pointed to this template in ADCS and had the following configuration:

We recommend enabling "Subject Alt Name Enabled" and "Automatically include CN as DNS SAN," as SANs in lieu of using CNs. Including a CN and SAN in certificates ensures backward compatibility with older clients that only support CNs and compatibility with newer clients that require SANs.

2.6.4 Folder Creation

To create a folder hierarchy for organizing certificate, application, and device objects, refer to the section titled "Managing your policies (folders)" in the *Venafi Trust Protection Platform Administration*

Guide. The following folder structure was created in our implementation of Venafi TPP to match the three ficticious departments of certifciate owners in the lab:

- Folder Root
  - Certificate Management
    - DMZ
      - DMZI
      - DMZE
    - Datacenter
      - DevOps
      - Linux Services
      - Windows Services
    - Datacenter Secure
  - System Management
    - DMZ
      - DMZI
      - DMZE
    - Datacenter
      - Linux Services
      - Windows Services
    - Datacenter Secure

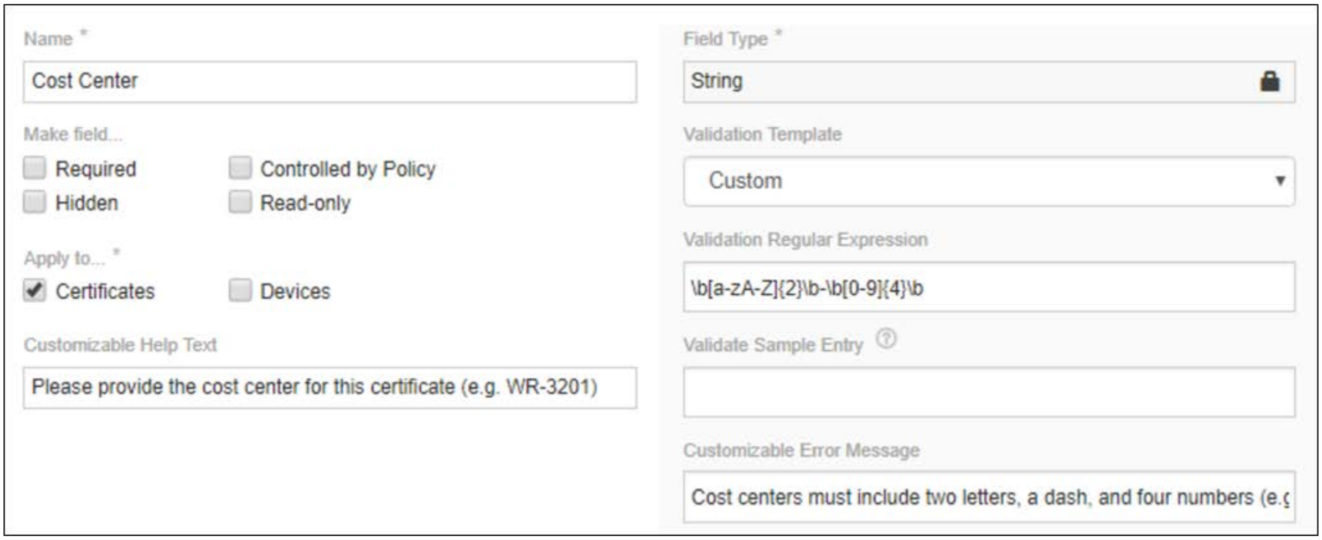
2.6.5 Custom Fields

Follow the instructions in the section titled “Working with Custom Fields” in the *Venafi Trust Protection Platform Administration Guide* to define additional metadata fields for certificates and other objects. Two custom fields were defined in our Venafi TPP implementation: Biz Owner and Cost Center.

We configured the Biz Owner custom field with a field type of “Identity” to allow the selection of user identities in AD.

The Cost Center custom field was configured with a “String” field type, including a regex to validate that the cost centers that were entered matched the pattern of two letters, one dash, and four numbers.

(e.g., AB-1234). A custom error message displays if a cost center doesn't match the regex pattern entered by a user.

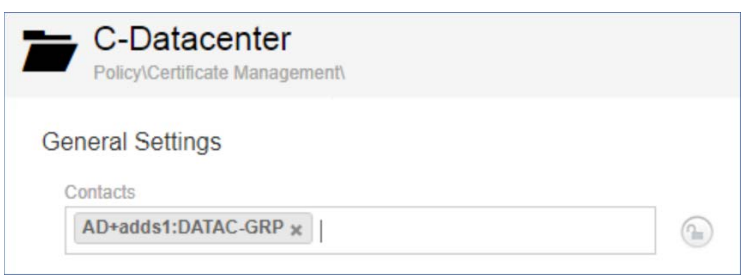


2.6.6 Assigning Certificate Owners

The assignment of certificate owners was done with AD groups Venafi TPP folders in our implementation, to ensure new certificates automatically had the correct owner assigned. The AD groups were created to represent the certificate owners in the four fictitious departments in our implementation. These groups were assigned as contacts and granted permissions at the folder level.

2.6.6.1 Contacts

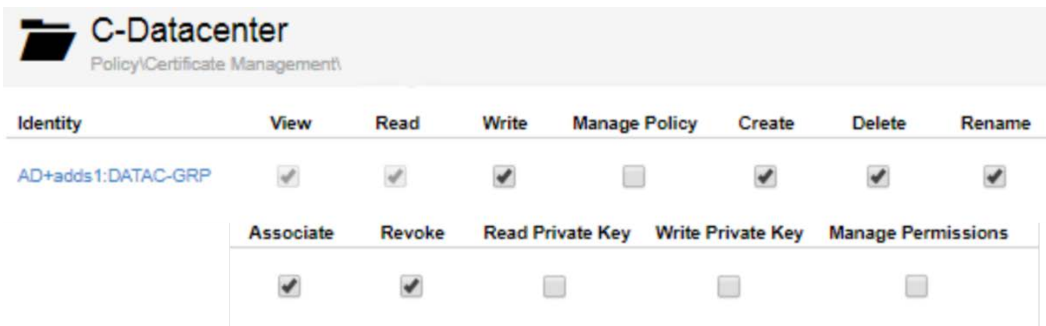
For information about assigning Contacts to folders in Venafi TPP, refer to the section titled "General configuration options" in the *Venafi Trust Protection Platform Administration Guide*. Each certificate owner AD group was assigned as a contact to their respective Venafi TPP folder, so they would receive notifications (e.g., impending expirations, errors, etc.).



2.6.6.2 Permissions

For instructions on assigning permissions in Venafi TPP, refer to the section titled "Assigning permissions to objects in Aperture" in the *Venafi Trust Protection Platform Administration Guide*. In our implementation, we assigned each group representing a certificate owner View, Read, Write, Create, Delete, Rename, Associate, and Revoke.

For example, the DATAC-GRP was assigned the following privileges to the C-Datacenter folder in our implementation of Venafi TPP.



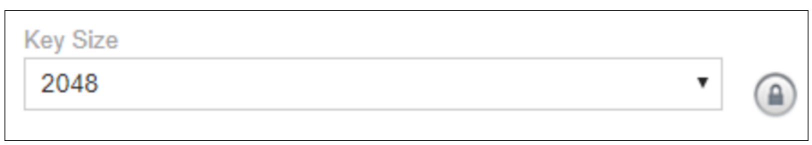
2.6.7 Setting Policies

For information about defining policies on folders in Venafi TPP, refer to the chapter titled "Using policies to manage encryption assets" in the *Venafi Trust Protection Platform Administration Guide*.

In our Venafi TPP implementation, the following policies were set:

- The Organization, City/Locality, State/Province, and Country fields within Subject DNs were locked on a top-level folder, so that those values were required in certificates across all groups.

- Specific domains were whitelisted. See the Domain Whitelisting section 2.6.8 of this document for more information.
- Approvers were assigned and locked at the folder level. See the "Workflow – RA Reviews" Section 2.6.9 of this document for more information.
- The key length was set to 2048 on the Certificate Management folder and locked.



- The following policies for certificate authorities were configured:
  - The internal Issuing CA was enforced on the following folders to ensure only internally issued certificates could be used:
    - DMZI
    - Datacenter
    - Datacenter Secure

- The publicly trusted DigiCert Mulit-SAN CA was enforced on the DMZE folder to ensure only publicly trusted EV certificates could be provisioned to the public facing interfaces of the F5 LTM.

2.6.8 Domain Whitelisting

To limit security exposure, control the domains for which certificates can be issued. For instructions on configuring the domains for which certificates can be requested in Venafi TPP (domain whitelisting), refer to the section titled "To configure certificate policy on a folder" in the *Venafi Trust Protection Platform Certificate Management Guide*.

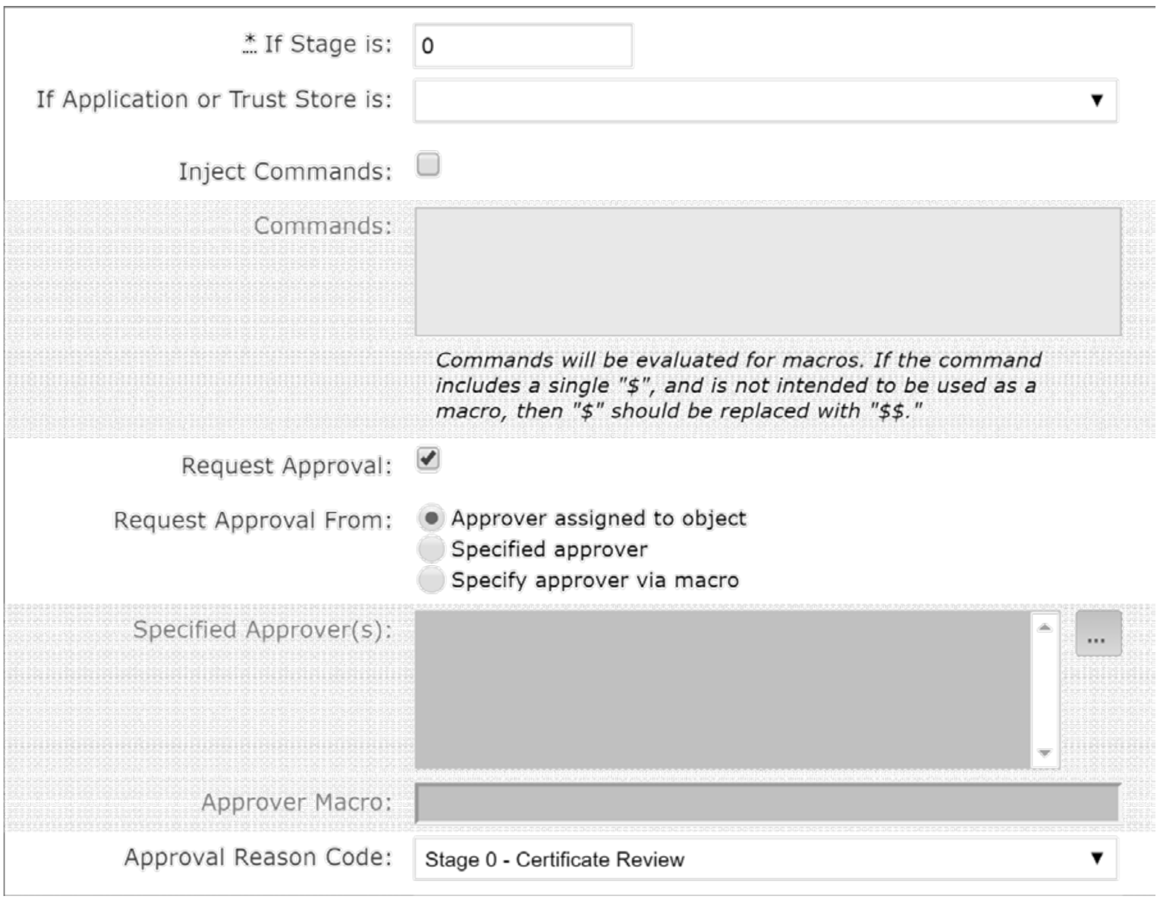
In our implementation, we allowed two internal domains (int-nccoe.org and ext-nccoe.org) for all folders that contained internal resources in Venafi TPP.

In the DMZE folder containing all the external resources, we also allowed the externally accessible domain (tls.nccoe.org).

2.6.9 Workflow – RA Reviews

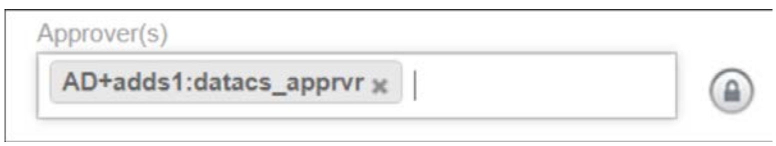
For instructions on configuring workflow gates in Venafi TPP, refer to the section titled “Creating a certificate workflow” in the *Venafi Trust Protection Platform Certificate Management Guide*. In our implementation, we established a workflow gate for the Datacenter Secure zone. To do so, perform the following steps:

1. Create a workflow object. Assign the stage to “0.” Select “Approver assigned to object” for Request Approval From.



2. Assign the workflow to the Datacenter Secure folder policy.

3. Assign the appropriate AD group (datacs_apprvr) to the **Approver(s)** for certificates on the Datacenter Secure folder.

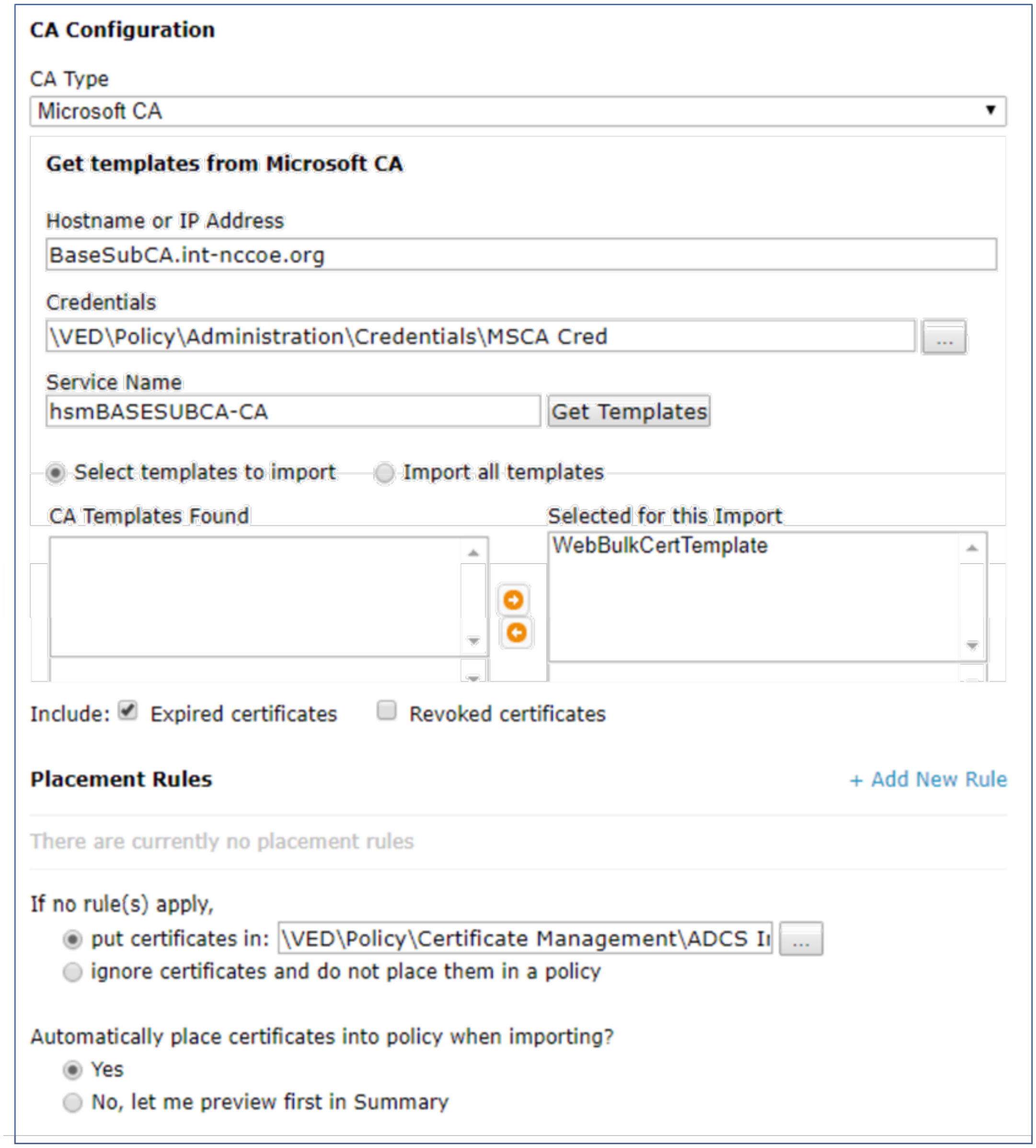


2.6.10 CA Import

Once folder structure, policies, certificate owners, and other configurations are completed, begin building the inventory of certificates—start by importing certificates from the ADCS-issuing CA.

For instructions on configuring imports from ADCS, refer to the chapter titled "Importing certificates from a certificate authority" in *Venafi Trust Protection Platform Administration Guide*.

In our implementation, we configured Venafi TPP to import certificates from a particular ADCS template named, “WebBulkCertTemplate.” We included expired—not revoked—certificates. We chose not to define any placement rules and placed all certificates into a single folder named **ADCS Import**.



A total of 523 certificates were imported from the ADCS issuing CA.

2.6.11 Network Discovery

It's possible to accomplish network discovery scanning for TLS server certificates in several ways, including using existing vulnerability assessment tools or the certificate management solution. In our implementation, we used Venafi TPP to perform network discovery scans using two different methods: scanning using Venafi TPP servers and the Scanafi utility.

Venafi TPP Server

In our implementation, we used Venafi TPP servers to perform network discovery scans in the Datacenter and Datacenter-Secure network zones. For instructions on performing network discoveries with Venafi TPP servers, see the chapter titled "Discovering certificates and keys" in the *Venafi Trust Protection Platform Certificate Management Guide*.

2.6.11.1 Scanafi

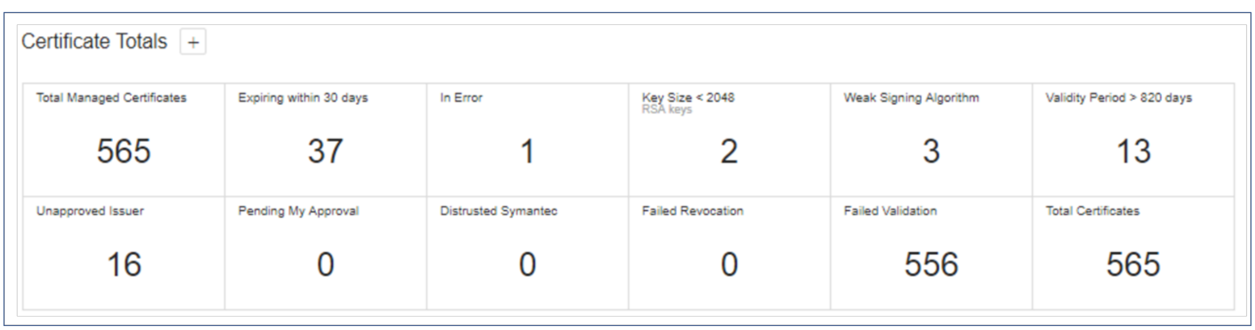
For information on using Scanafi to perform network discovery scans, refer to the section titled "Automatically calling Discovery/Import from Scanafi" in *Venafi Trust Protection Platform Web SDK Developer's Guide*.

In our implementation, we installed Scanafi on a Fedora Linux system in the DMZ network zone. The following command was used to execute a network discovery scan.

```
./scanafi_linux_x64 --tppurl=https://venafi1.int-nccoe.org \
--tppuser=vscanuser --tpppass=******** --range=192.168.4.0/23 \
--zone="\\VED\\Policy\\Certificate Management\\UNKNOWN ORIGIN" \
--certsonly
```

2.6.12 Identify Certificate Risks/Vulnerabilities

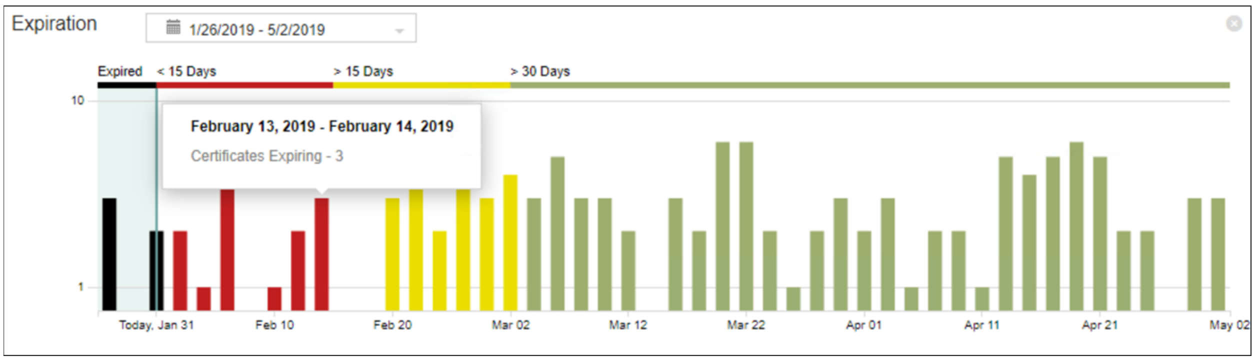
Following the import of certificates from the ADCS-issuing CA and the network discovery scans, we used the Venafi TPP dashboard to identify certificate risks and vulnerabilities. The following shows the dashboard micro-widgets for our implementation.



We used this information to identify certificates not compliant with policy (e.g., certificates issued by unapproved CAs or with weak lengths), so they could be replaced.

The dashboard was also used to identify outage risks related to certificate expirations. The following figure displays the Expiration widget of the dashboard that shows the expiration profile for certificates in our implementation.

Figure 2-2 Venafi Dashboard Expiration Widget showing the Certificate Expiration Profile



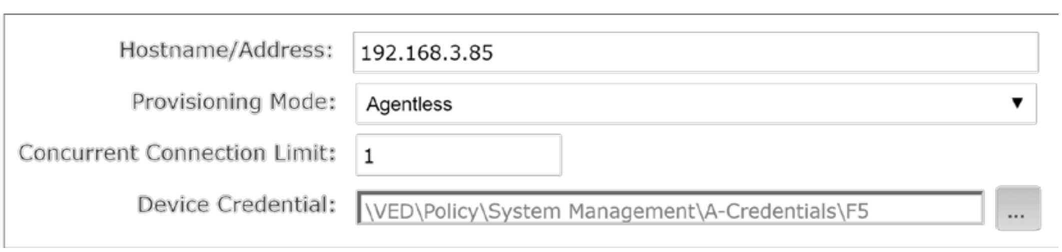
2.6.13 Automate Management

2.6.13.1 F5 BIG-IP LTM

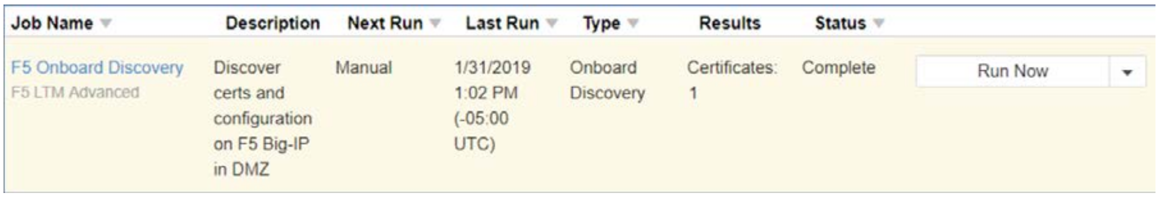
2.6.13.1.1 Discover Existing F5 Certificates and Manage

Venafi TPP can automatically discover existing certificates and configuration through its Onboard Discovery feature. Because most organizations have F5 systems with existing certificates installed, this is a common process for F5 systems we used in our implementation, which included the following steps:

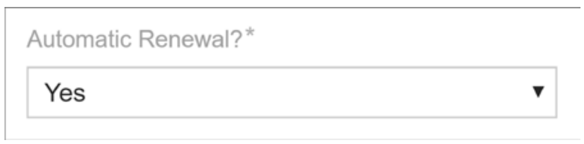
1. Create an Onboard discovery job to discover certificates on F5 systems. For instructions on how to create Onboard Discovery jobs, refer to the section titled “Using Onboard Discovery” in the *Venafi Trust Protection Platform Certificate Management Guide*.
2. Create a device object in Venafi TPP with the address and credentials for the F5 device on which you want to discover and manage certificates.



3. Run the F5 Onboard Discovery job by clicking **Run Now**.



4. Ensure the discovered certificate(s) are set to automatically renew when they are nearing expiration.

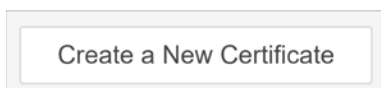


5. With this discovered configuration, including the certificate, Venafi TPP was set to automatically replace the existing certificate with a new certificate prior to expiration.

2.6.13.1.2 Install a New Certificate on F5

In our implementation, Venafi TPP was used to enroll for and install a new certificate on the F5 LTM in the DMZ. The following steps were used to perform these operations:

1. Create a new certificate object in the Venafi TPP Aperture console.



2. Select the appropriate folder.

3. Select a name for the certificate.

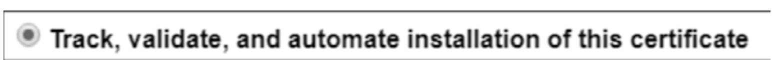
4. Select the "Provisioning" Management Type to configure the certificate for automated management.

5. Enter the CN for the certificate.

6. Enter the SANs for the certificate.

7. Configure the certificate for automatic renewal and installation when it is nearing expiration.

8. Add a new installation for the certificate, and indicate that management will be automated for that installation.

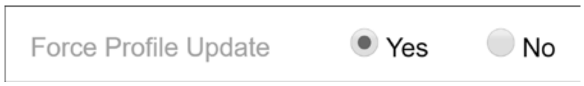


9. Select the F5 device where the certificate will be installed.

10. Indicate that the Installation Type is "F5 BIG-IP Local Traffic Manager."

11. The certificate we were installing was not for securing the administrative interface to the F5 LTM, therefore, we selected "No" for the Device Certificate.

12. We indicated that Venafi TPP should update the profile when the new certificate was installed. This ensures the configuration was properly set up to use the new certificate.

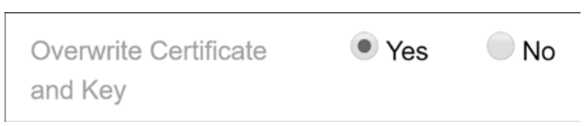


13. We instructed Venafi TPP to install the CA certificates with the new certificate—enabling clients connecting to the F5 to validate the certificate signature with the chain.

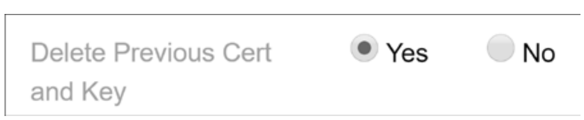
14. We chose to have Venafi TPP bundle the CA certificates with the new certificate (in the same file on the F5 device).

15. An HSM was not installed on the F5 device we were using, so we indicated this to Venafi TPP.

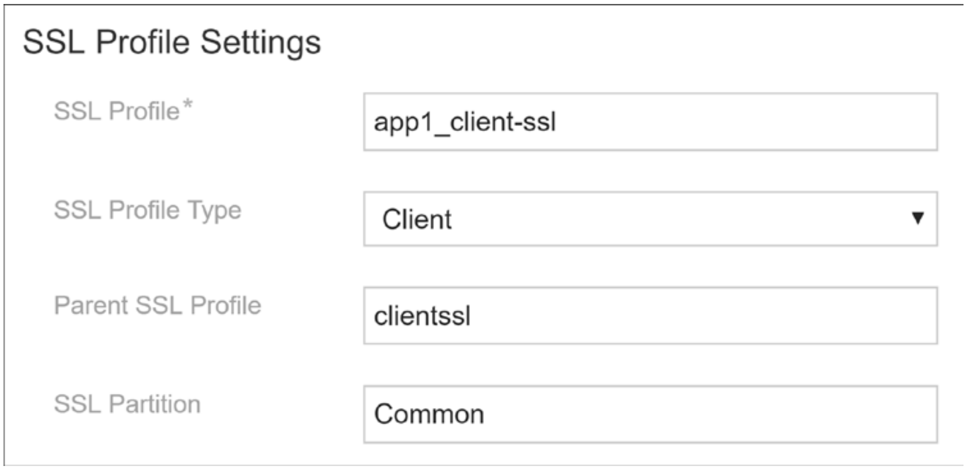
16. We instructed Venafi TPP to overwrite the existing certificate each time it installed a new certificate (prior to expiration).



17. We instructed Venafi TPP to delete the existing certificate when the new certificate was installed.

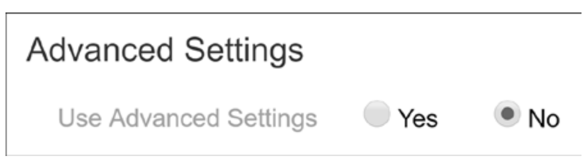


18. To ensure the certificate was associated with the correct SSL profile on the F5 LTM, we configured the following:



19. We provided Venafi TPP information about the virtual server where the certificate should be associated.

20. We indicated to Venafi TPP that we did not use mutual authentication or other advanced features on the F5 LTM.



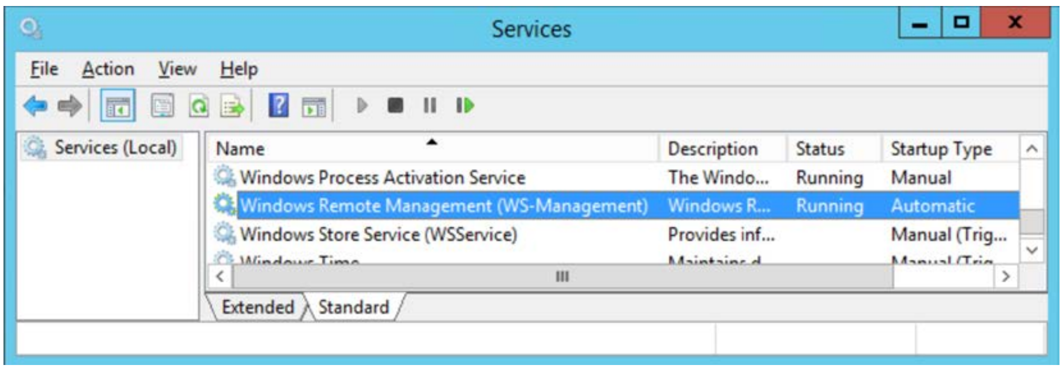
21. After configuring these settings, we clicked **Save**.

22. Click **Renew Now** on the certificate to start to enroll a new certificate and to install it on the F5 LTM with these configuration settings.

2.6.13.2 Microsoft IIS – Agentless

The Microsoft IIS system we used in our implementation to demonstrate automated management had an existing certificate. Venafi TPP can automatically discover existing certificates and configuration through its Onboard Discovery feature. Consequently, the following process was used:

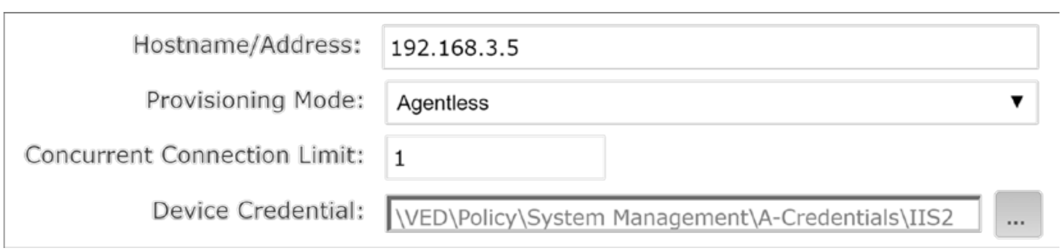
1. Create an Onboard discovery job to discover certificates on Microsoft IIS systems. For instructions on how to create Onboard Discovery jobs, refer to the section titled “Using Onboard Discovery” in the *Venafi Trust Protection Platform Certificate Management Guide*.
2. Confirm Windows Remote Management (WinRM) service was running on the Windows server hosting IIS.



3. Enable WinRM at the command line.

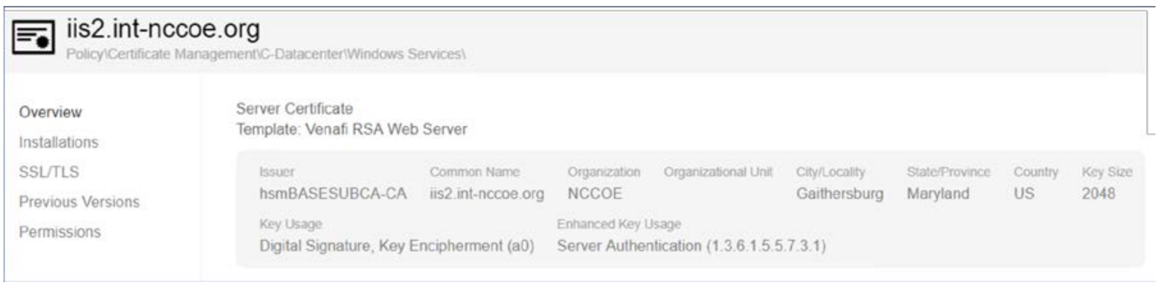
```
C:\>winrm quickconfig
```

4. Create a device object in Venafi TPP with the address of the Windows server hosting IIS and a credential for Venafi TPP to authenticate to the system.

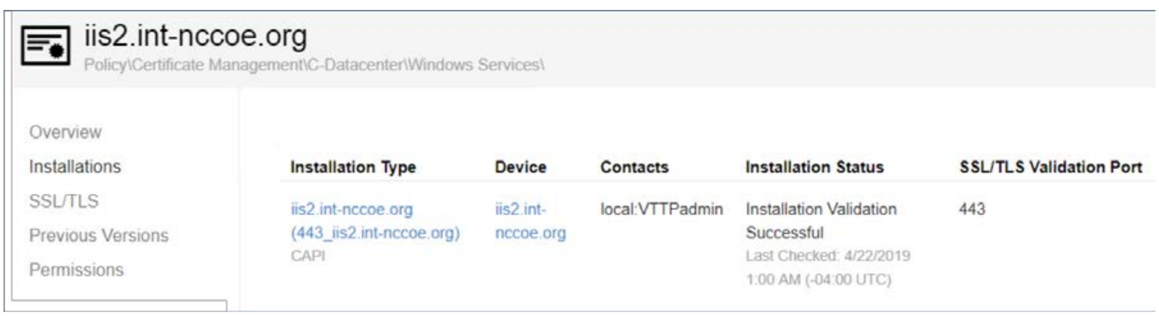


5. Execute the IIS Onboard Discovery job that applied to the folder where the device was located. The certificate and binding configuration on IIS were discovered.

6. The certificate is discovered.



7. In addition, IIS binding information is discovered, so that all the necessary configuration for automated management is populated in Venafi TPP.

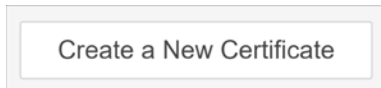


8. To ensure the certificate automatically renews and is replaced when nearing expiration, confirm the certificate was set to automatically renew prior to expiration.

2.6.13.3 Microsoft IIS with SafeNet AT HSM – Agentless

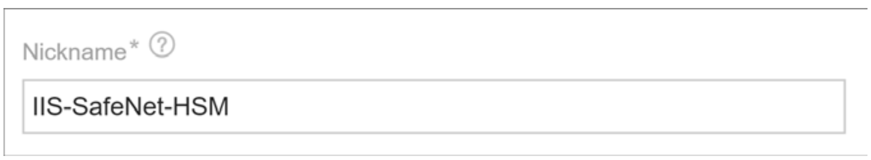
The Venafi TPP server was used to remotely trigger the generation of a key pair and CSR on the SafeNet AT HSM. The HSM is connected to the Microsoft IIS server in the Datacenter Secure zone and can enroll a certificate using the generated CSR. It can also install the certificate in the Windows server with the proper configuration for the Microsoft IIS server. The following steps are used to perform these operations:

1. Ensure the SafeNet AT HSM client is installed and configured on a Windows server hosting Microsoft IIS. See Section 2.2.2.4 for instructions.
2. Create a new certificate object in the Venafi TPP Aperture console.



3. Select the appropriate folder.

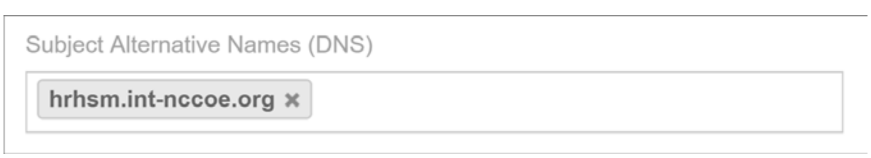
4. Select a name for the certificate.



5. Select the "Provisioning" Management Type to configure the certificate for automated management.

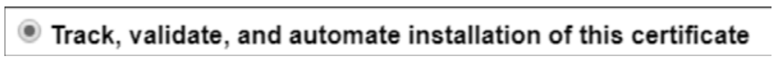
6. Enter the CN for the certificate.

7. Enter the SANs for the certificate.



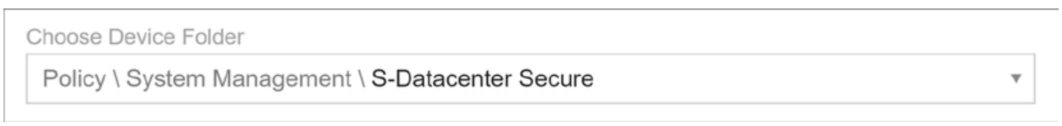
8. Configure the certificate for automatic renewal and installation when it is nearing expiration.

9. Add a new installation for the certificate and indicate that management is automated for that installation.



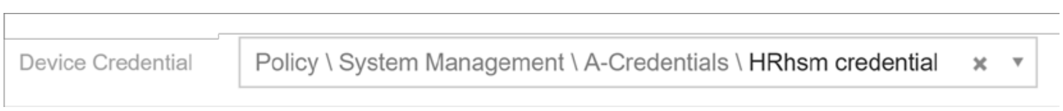
10. Enter the address for the device where the certificate will be installed.

11. Select the folder where the device object should be created.



12. Indicate that the application type for the installation is “Windows CAPI & IIS.”

13. Select the credential to authenticate to the system for management operations.

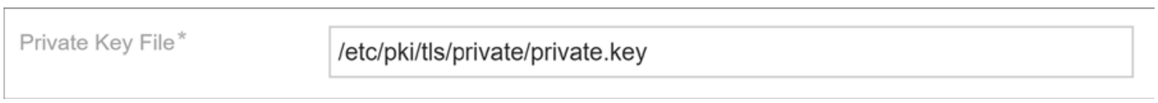


14. Enter a CAPI-friendly name for the certificate to be installed.

15. Click **Renew Now** on the certificate to start generating a new key pair on the HSM and to start getting a new corresponding certificate.

2.6.13.4 Apache – Agentless

1. Create a new certificate object in the Venafi TPP Aperture console. For instructions on creating a new certificate, refer to “Creating a new certificate in Aperture” in *Venafi Trust Protection Platform Working with Certificates*.
2. Add an installation location for the certificate for the Apache where the certificate will be installed. For instructions on adding an Apache installation in Aperture, refer to the section titled “Creating an Apache application object” in the *Venafi Trust Protection Platform Certificate Authority and Hosting Platform Configuration Guide*. Notable configuration information that we used in our implementation, includes:
    a. Set the private-key file location to correspond to the Virtual Host configuration on the Apache server.

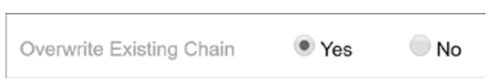
    b. Set the certificate file location to correspond to the Virtual Host configuration on the Apache server.

    c. Set the CA certificate chain file location to correspond to the Virtual Host configuration on the Apache server.

    d. Instruct Venafi TPP to update the CA chain.

3. Click **Install** in the Actions menu to deploy the certificate to the Apache system.

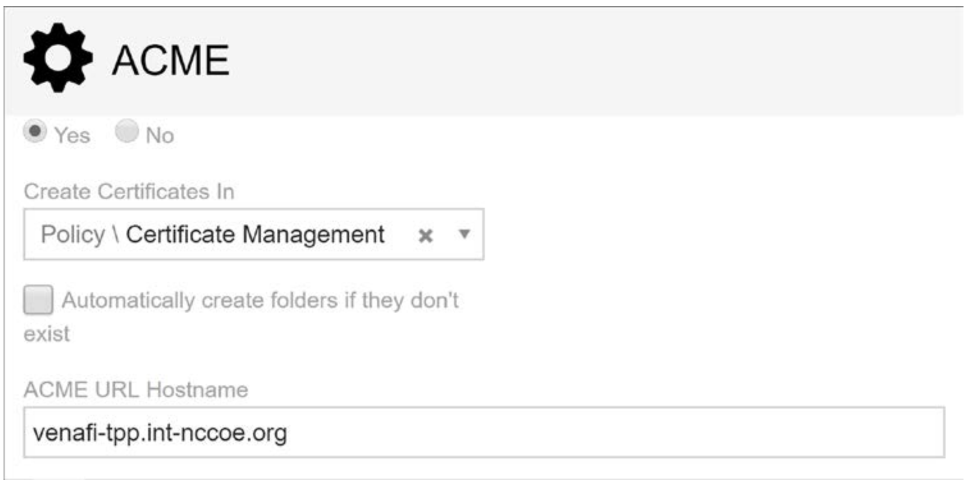
2.6.13.5 Apache – ACME

Venafi TPP was configured as an ACME server in our implementation to support ACME-based requests from internal systems. For instructions on using ACME with Venafi TPP, refer to the section titled “ACME integration with Trust Protection Platform” in the *Venafi Trust Protection Platform Certificate Management Guide*.

2.6.13.6 Configuring Venafi TPP for ACME

The following steps are needed for configuring Venafi TPP to request certificates using an ACME client.

1. Configure Venafi TPP to enable the ACME server.
    a. The ACME server is not enabled by default in Venafi TPP.
    b. When ACME is enabled, select the folder where ACME-enrolled certificates are placed.
    c. Enter the address of the Venafi TPP server that will service ACME clients.



2. Assign an email address to the requesting account. The ACME protocol requires an email address be provided during the registration process. Venafi TPP must be able to find the entered email address in the local Venafi TPP identity directory or AD (depending on which directory is used).

2.6.13.7 Configuring Certbot for Apache

Certbot is the standard client use for ACME on many systems. Find instructions on installing certbot at the following address: https://certbot.eff.org/. We installed certbot on a Fedora Linux system to automate certificate requests and installation for Apache.

We performed the following steps in our implementation.

1. Ensure the virtual host is configured in Apache.
2. Install certbot for Apache.

```
sudo dnf install certbot certbot-apache
```

3. The root certificate for the CA that issued the Venafi TPP server's certificate must be trusted on the system where certbot is run. This is done by adding it to one of the following files depending on the OS:

```
/etc/ssl/certs/ca-certificates.crt",                // Debian/Ubuntu/Gentoo etc.
/etc/pki/tls/certs/ca-bundle.crt",                  // Fedora/RHEL 6
/etc/ssl/ca-bundle.pem",                            // OpenSUSE
/etc/pki/tls/cacert.pem",                           // OpenELEC
/etc/pki/ca-trust/extracted/pem/tls-ca-bundle.pem", // CentOS/RHEL 7
```

4. Run certbot to request a certificate. A certificate was installed on the Apache system.

```
certbot certonly \
--server “https://venafi1.int-nccoe.org/vacme/v1/directory” \
--cert-name apache1 --domains apache1.int-nccoe.org \
--apache --email acmeuser@int-nccoe.org --no-eff-email
```

2.6.13.8 Kubernetes

Instructions for installing, configuring, and using Kubernetes are available on https://kubernetes.io/.

We installed a three-node Kubernetes cluster on three CentOS Linux systems in the Datacenter network zone in our implementation. We installed the following for the Kubernetes deployment:

- Docker version 18.09.3, build 774a1f4
- kubelet, kubeadm, and kubectl v1.13.4
- Weave (as our overlay network)

Once these components were installed, we installed and configured cert-manager in Kubernetes to automatically request certificates for ingresses in Kubernetes. We performed the following steps:

1. Verified a user account with Venafi TPP WebSDK access and permissions to the folder(s) where certificates are being requested from cert-manager (see the definition of the issuer below). We created a user named “vapirequester” in AD for this purpose. The account was granted Create, Write, Read, and View permissions to a folder named DevOps. We also granted that account WebSDK access.

2. Verified Jetstack Cert-Manager was installed with the necessary components to request certificates from Venafi TPP. This automatically creates a namespace named “cert-manager,” which we used for the rest of our configuration.

```
[ec2-user@kubemaster ~]$ kubectl describe deployment cert-manager -n cert-manager
Name:                   cert-manager
Namespace:              cert-manager
CreationTimestamp:      Wed, 06 Mar 2019 03:15:23 +0000
Labels:                 app=cert-manager
                        chart=cert-manager-v0.6.0-venafi.0
                        heritage=Tiller
                        release=cert-manager
Annotations:            deployment.kubernetes.io/revision: 2
                        kubectl.kubernetes.io/last-applied-configuration:
                          {"apiVersion":"apps/v1beta1","kind":"Deployment","metadata":
{"annotations":{},"labels":{"app":"cert-manager","chart":"cert-manager-v0.6.0-...
Selector:               app=cert-manager,release=cert-manager
Replicas:               1 desired | 1 updated | 1 total | 1 available | 0 unavailable
StrategyType:           RollingUpdate
MinReadySeconds:        0
RollingUpdateStrategy:  25% max unavailable, 25% max surge
Pod Template:
  Labels:           app=cert-manager
                    release=cert-manager
  Service Account:  cert-manager
  Containers:
   cert-manager:
    Image:      quay.io/jetstack/cert-manager-controller:venafi-0
    Port:       <none>
    Host Port:  <none>
    Args:
      --cluster-resource-namespace=$(POD_NAMESPACE)
      --leader-election-namespace=$(POD_NAMESPACE)
    Requests:
      cpu:      10m
      memory:   32Mi
    Environment:
      POD_NAMESPACE:    (v1:metadata.namespace)
    Mounts:           <none>
  Volumes:            <none>
Conditions:
  Type           Status  Reason
  ----           ------  ------
  Progressing    True    NewReplicaSetAvailable
  Available      True    MinimumReplicasAvailable
OldReplicaSets:  <none>
NewReplicaSet:   cert-manager-7d9f97d789 (1/1 replicas created)
Events:          <none>
[ec2-user@kubemaster ~]$
```

```
kubectl apply -f https://raw.githubusercontent.com/jetstack \
/cert-manager/venafi/contrib/manifests/cert-manager/with-rbac.yaml
```

3. Created Kubernetes secret for authenticating to Venafi TPP.

```
kubectl create secret generic tppsecret \
--from-literal=username='vapirequester' \
--from-literal=password='********' \
--namespace cert-manager
```

4. Copied the Root CA certificate that the certificate on the Venafi TPP chains up to (this is used by cert-manager to validate the Venafi TPP certificate). This was copied to a file named *rootca.pem*.
5. Generated a base64 representation of the Root CA certificate.

```
cat rootca.pem | base64 | tr -d '\n'
```

6. Created a yaml file (*tppvenafiissuer.yaml*) for the configuration for a cert-manager issuer that points to Venafi TPP. Note that the base64 representation of the Root CA certificate is placed after “caBundle:” with a single space separating (there is no carriage return). The “zone” sets the folder where the requested certificate will be placed.


```
apiVersion: certmanager.k8s.io/v1alpha1
kind: Issuer
metadata:
  name: tppvenafiissuer
  namespace: cert-manager
spec:
  venafi:
    zone: 'Certificate Management\C-Datacenter\DevOps'
    tpp:
      url: https://venafi1.int-nccoe.org/vedsdk
      credentialsRef:
        name: tppsecret
      caBundle:
LS0tLS1CRUdJTiBDRVJUSUZJQ0FURS0tLS0tCk1JSUMvVENDQWVXZ0F3SUJBZ0lRSnBydys5NUMyNnh
Kd2FEeXFsWUhXekFOQmdrcWhraUc5dzBCQVFzRkFEQVIKTVE4d0RRWURWUVFERXdaU1QwOVVRMEV3SG
hjTk1UZ3dOekE1TWpNME1EUTVXaGNOTWpBd056QTVNak0xTURRNApXakFSTVE4d0RRWURWUVFERXdaU
1QwOVVRMEV3Z2dFaU1BMEdDU3FHU0liM0RRRUJBUVVBQTRJQkR3QXdnZ0VLCkFvSUJBUURaaHZxUXk3
ckZrTnlWenZxSW5GeE4ydVBLTEJRdzl1Mk5kb1NmTXhMTVU5TlB4UUcwOVNyT1V1SSsKYmhkckJNeEt
FbStzMm5PTUNtY3g2SDN1dGp0UmtWU2pxQVZkYnQrVkN0TmtQWlZYTlRKaWlkOFVlTmRYY1dDMQpjMk
M5RUVBNDVUOG94eG10TEkvd0l0N2RaMHpwVldxSitvT1VLVGFIZWpRTFcveUxYWkIvU3AvZzFuUmFOM
XhqCjFZVllRQ2dCMWxVZ0lGQ3lXUzJJSmwvQXMrRjN6ckFOazg1K0krYlBCQ050ZUFYVTNkS0xTU0Nx
WmxqdVZ1YncKa2QwVzhzMDRPRmdCR2lCM2o2MXBydEZZc1N5WlZKYjNKVDRFWnpTMlNBbXlHZlFteVF
heEpJWC9RbmIzSGp5NwpHa0ViaVFqT1FLNE9mYlZiU2tKcTh5bHdmNkhEQWdNQkFBR2pVVEJQTUFzR0
ExVWREd1FFQXdJQmhqQVBCZ05WCkhSTUJBZjhFQlRBREFRSC9NQjBHQTFVZERnUVdCQlRZKzBtL3dwR
EptaEdmUCtxbHJQcUI2M0t5akRBUUJna3IKQmdFRUFZSTNGUUVFQXdJQkFEQU5CZ2txaGtpRzl3MEJB
UXNGQUFPQ0FRRUFGZk5EeWVlK1ZSSGhrUEx1Y1pGeQpmTlNEb0d0alZQckl5Q2J3aXMyQUFOL0xYV2J
MVzlYUG1YOWVwSFJOQ3Zla1RFa0RQam1OVWxFd0cwTGUwbnByCmM3bTVrbDhjYTBNaHhkMUhURm1Xbm
tydjdmRy80dmt6eUhXR0FwekNTcFlyUEhsS0lEaisxUlpmY1VrQ2lWWVQKb2RJL3V3K1A1RTNHalNJZ
HdaK0RoODRFVURhQ0JHc1I1MzZOMnlaMURjekRTUWg5SHBPaTh6b3dYcnFWbzdkcApCYVpsUUNRUGlj
N0hRaE0rS0VLMlVha1J4U1Z2ciszOEJRVyszOS9zbUFET1QxN2o0MmxEcHFpdjRBTWd4cUxWCmdXMFR
sc1pwK1FHRnU1TExjSnVqS3llT09nM2NYanI3S1lwU0FoOVpWNzFpcFRzL2Q4NzdidWdPYURkL2Yrdl
kKSFE9PQotLS0tLUVORCBDRVJUSUZJQ0FURS0tLS0tCgo=
```


7. Created the issuer in Kubernetes using the newly created file.

```
kubectl apply -f tppvenafiissuer.yaml
```

8. Created a yaml file for the ingress to the nginx service. Note the annotation ‘certmanager.k8s.io/issuer: "tppvenafiissuer"’ in the yaml file. This tells Jetstack Cert-Manager that it should automatically request and install a certificate from this ingress using the issuer we

defined earlier. Cert-manager uses the host name under **tls** and **hosts** (kube-ingress.int-nccoe.org) for the CN and SAN it submits in the certificate request to Venafi TPP.

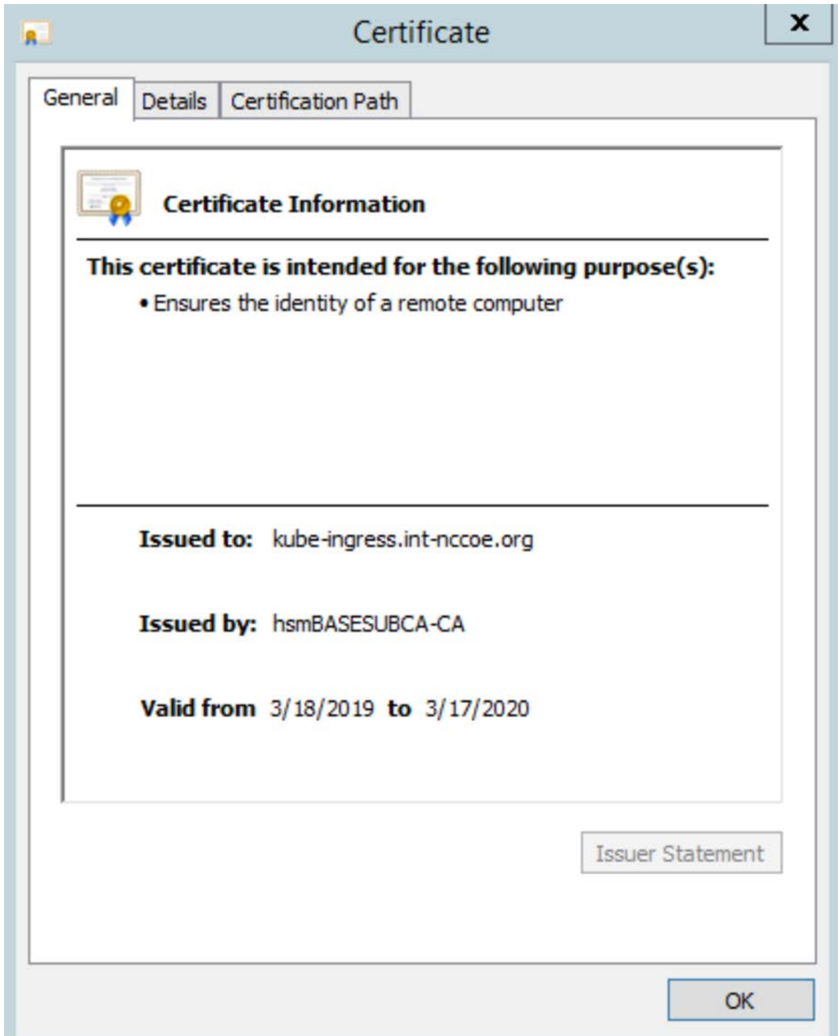
```
apiVersion: extensions/v1beta1
kind: Ingress
metadata:
  name: nginx-ingress
  namespace: cert-manager
  annotations:
    kubernetes.io/ingress.class: "nginx"
    certmanager.k8s.io/issuer: "tppvenafiissuer"

spec:
  tls:
  - hosts:
    - kube-ingress.int-nccoe.org
    secretName: nginx-cert
  rules:
  - host: kube-ingress.int-nccoe.org
    http:
      paths:
      - path: /
        backend:
          serviceName: nginx
          servicePort: 80
```

9. Created the ingress.

```
kubectl create -f nginx-ingress.yaml
```

10. Once the ingress was created, connected with a browser kube-ingress.int-nccoe.org to confirm that a certificate was properly issued through Venafi TPP and installed for the ingress.



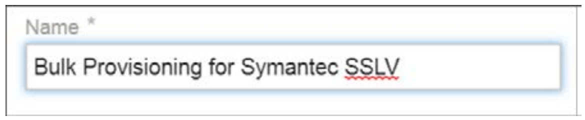
2.6.13.9 Symantec SSL Visibility

In our implementation, we configured Venafi TPP to automatically install TLS certificates and private keys used on several of the TLS servers—including IIS and Apache—onto the Symantec SSL Visibility to inspect traffic going to those servers.

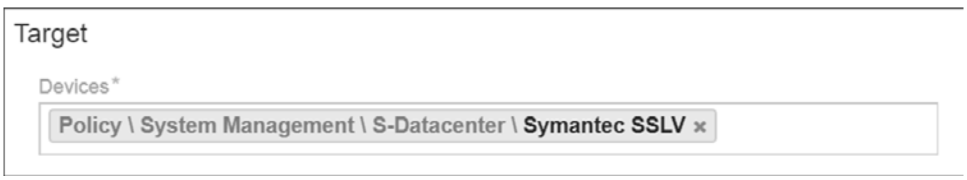
1. Device object was created in Venafi TPP with the address and credentials for the Symantec SSL Visibility. For instructions on adding a device object, refer to the section titled “Adding Objects” in the *Venafi Trust Protection Platform Administration Guide*.

2. To ensure all required certificates and private keys are copied to the TLS inspection device, Venafi includes a feature called Bulk Provisioning. We created a bulk provisioning job.

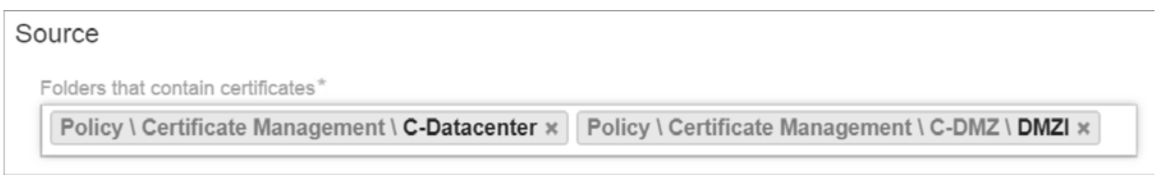
3. We named the job to distinguish it from other bulk provisioning jobs.



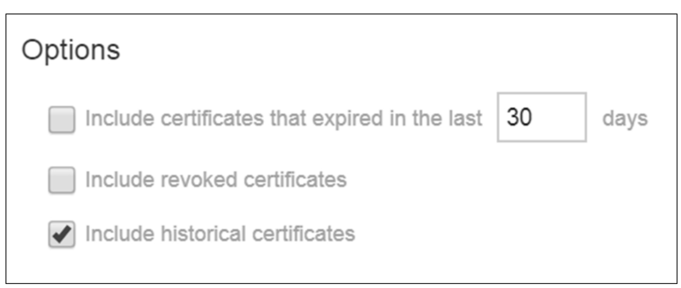
4. We selected the device object created above for the Symantec SSL Visibility Appliance as the target to which private keys would be provisioned.



5. Venafi TPP was instructed to provision private keys associated with certificates in two folders:

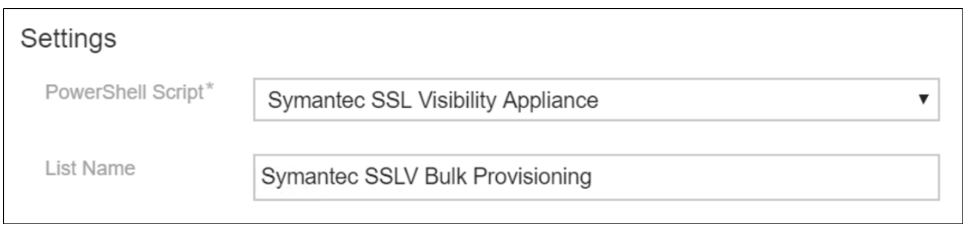


6. The default options excluded expired and revoked certificates and included historical certificates. Historical certificates are certificates that Venafi replaced by Venafi TPP. These certificates are still valid (not expired) and active on certain systems, though a new certificate was issued. Consequently, it is important to provision them to the TLS inspection appliance to ensure all traffic can be decrypted.



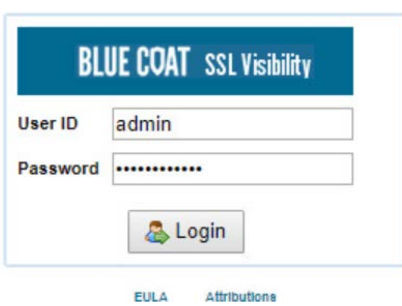
7. The bulk provisioning job was configured to run every Sunday at midnight to ensure new certificates and private keys are deployed to the TLS inspection device.

8. Venafi TPP uses an adaptable framework for bulk provisioning, so these jobs can be customized based on the environment's requirements. To support bulk provisioning to the Symantec SSL Visibility, the bulk provisioning script has the Venafi TPP copied into the *C:\Program Files\Venafi\Scripts\AdaptableBulk* directory. The bulk provisioning job was configured to use this script.

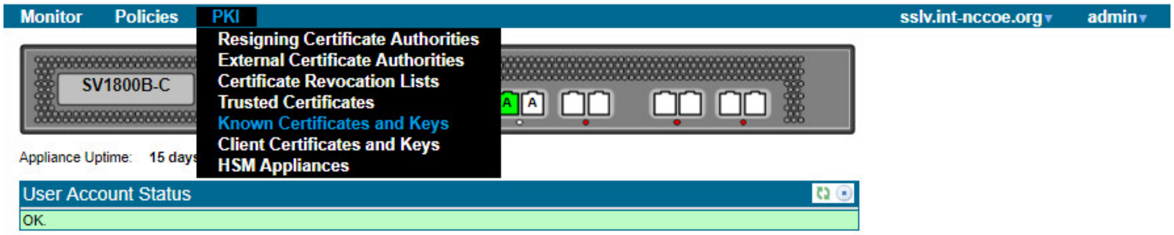


9. The bulk provisioning job will run once it is saved. The private keys were confirmed to be on the device.

10. To check if keys are saved in the SSL VISIBILITY, login to the SSL VISIBILITY WebUI by going to *https://192.168.1.95*



11. Go to **PKI > Known Certificates and Keys.**



12. In the **Known Certificates with Keys** Lists field, click on the **all-known-certificates-with-keys** field.

Known Certificates with Keys Lists
Name
all-known-certificates-with-keys

13. The imported certificates and keys are then shown under the Known Certificate with Keys field.

Known Certificates with Keys Lists
Name
all-known-certificates-with-keys

Known Certificate with Keys

Summary	Key Type
apache3.ext-nccoe.org, NCCOE, TLSLAB	RSA
iis2.int-nccoe.org, NCCOE	RSA
iis2.int-nccoe.org, NCCOE [2]	RSA
iis2.int-nccoe.org, NCCOE [3]	RSA
iis2.int-nccoe.org, NCCOE [4]	RSA
ws1.int-nccoe.org, NCCOE, TLSLAB	RSA
ws2.int-nccoe.org, NCCOE, TLSLAB	RSA
ws3.int-nccoe.org, NCCOE, TLSLAB	RSA

2.6.14 Continuous Monitoring

Venafi TPP provides several tools that can continuously monitor TLS certificates within an enterprise, including scheduled network discovery scanning, monitoring certificates for expiration, and monitoring the operational status of known certificates.

2.6.14.1 Regular Network Scanning

In the lab, Venafi TPP was configured to perform weekly network discovery scans of the Datacenter and Datacenter Secure networks zones from the Venafi TPP server. The scans were scheduled to run at 2:00 a.m. each Sunday. The lab network was small enough for network scans to complete within a few minutes. Nonetheless, blackout periods were configured from 6:00 a.m. to 7:00 p.m. weekdays to ensure network scans were not performed during "normal business hours."

A notification rule was defined to send an alert to the certificate services team upon discovery of either new certificates or previously unknown certificates (indicating they may have been issued and installed outside of standard processes) installations.

2.6.14.2 Certificate Expiration Monitoring

Significant application outages can occur when a certificate expires while in use. Consequently, it is critical that certificate owners track certificate expiration dates and replace them. The certificate services team can help certificate owners by implementing automated processes that monitor certificate expiration dates and notify the owners.

We used Venafi TPP in the lab to monitor certificate expiration dates and notify certificate owners. The methodology used in the lab followed the recommendations in *SP 1800-16 Volume B*. A weekly expiration report was scheduled giving certificate owners a list of certificates set to expire within the next 120 days. The following shows an example expiration report from the lab environment. The top of the report summarizes the status of certificates associated with a particular certificate owner.

EXPIRATION REPORT

This report contains details about the upcoming expiration dates of your certificates. Expiration dates are displayed from most urgent to least urgent, as defined when the report was generated.

Please see Appendix for source details and other information regarding this report.

Status	Range	Certificates (135)	Percentage of Total
Expired	0-0 Days	5	3.7 %
Immediate	0-5 Days	9	6.7 %
Near Term	5-30 Days	35	25.9 %
Long Term	30-90 Days	86	63.7 %

The expiration report lists all of the applicable certificates.

Common Name	Valid To	Contact	Issuer	Type	Days
9cka1wpk.tls.nccoe.org	2/28/2019	Administrators	hsmBASESUBCA-CA	Prov	0
ck0jb30u.tls.nccoe.org	2/28/2019	Administrators	hsmBASESUBCA-CA	Prov	0
nltc1wv8.tls.nccoe.org	2/28/2019	Administrators	hsmBASESUBCA-CA	Prov	0
4tpbc539.int-nccoe.org	3/1/2019	Administrators	hsmBASESUBCA-CA	Prov	0
-m7pgw09.int-nccoe.org	3/1/2019	Administrators	hsmBASESUBCA-CA	Prov	0
i-8r4ol9.ext-nccoe.org	3/2/2019	Administrators	hsmBASESUBCA-CA	Prov	1
wdw7yww7.ext-nccoe.org	3/2/2019	Administrators	hsmBASESUBCA-CA	Prov	1
owg82h5z.tls.nccoe.org	3/3/2019	Administrators	hsmBASESUBCA-CA	Prov	2
axz8jof2.int-nccoe.org	3/4/2019	Administrators	hsmBASESUBCA-CA	Prov	3

In addition to the reports, notification rules were configured to send emails to the owners of certificates expiring within 30 days. These notifications were configured to send daily, until the certificate was replaced. For any certificate expiring in less than 20 days, a notification rule was configured to send an additional email to escalation contacts, including the person identified as the Biz Owner and an incident response team. The objective was to minimize the amount of email that certificate owners received if all of their certificates were replaced in a timely fashion—ensuring sufficient alerts were sent for those certificates that still needed replacement.

2.6.14.3 Certificate Operation Monitoring

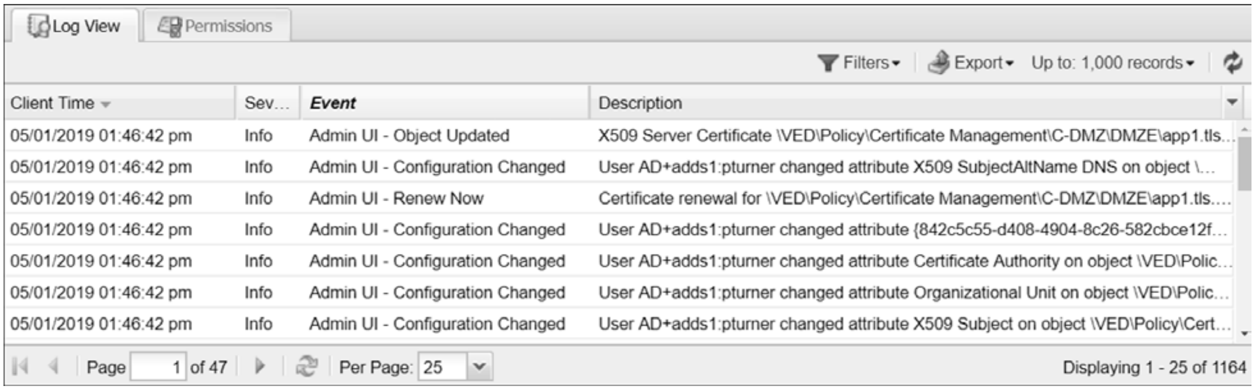
Network discovery scans provide insight into newly installed certificates, however, it's equally important to monitor the operational state of known certificates. For example, a certificate owner may get a replacement certificate for an installed certificate set to expire. If the certificate isn't installed prior to its expiration date, an outage can result. They may install the new certificate on several but not all of the systems where the existing certificate is installed, causing the systems that were not updated to fail when the existing certificate expires. Finally, they may install the new certificate in all necessary locations, but not reset the application so the new certificate is read and use by the application, resulting in an outage, because the application is continuing to use the existing certificate that expires.

Venafi TPP provides a service call network certificate validation that automatically checks deployed certificates to ensure the correct certificate is installed and operational, thereby addressing the issues described above. If a certificate issue is detected, the certificate owner is notified. Network certificate validation was enabled on Venafi TPP in the lab.

2.6.14.4 Logging of Certificate-related Security Events

Venafi TPP logs all management operations performed on certificates, including changes that administrators make within the user interfaces, changes via API, and all automated operations that are performed. Errors are also logged. All logged events are automatically stored in the Venafi TPP database. These events can be reviewed in the Venafi TPP console. It also is possible to sort, filter, and export the log events.

The following provides an example of several administrative events logged in our implementation, created by filtering on specific types of administrative events focused on configuration changes:

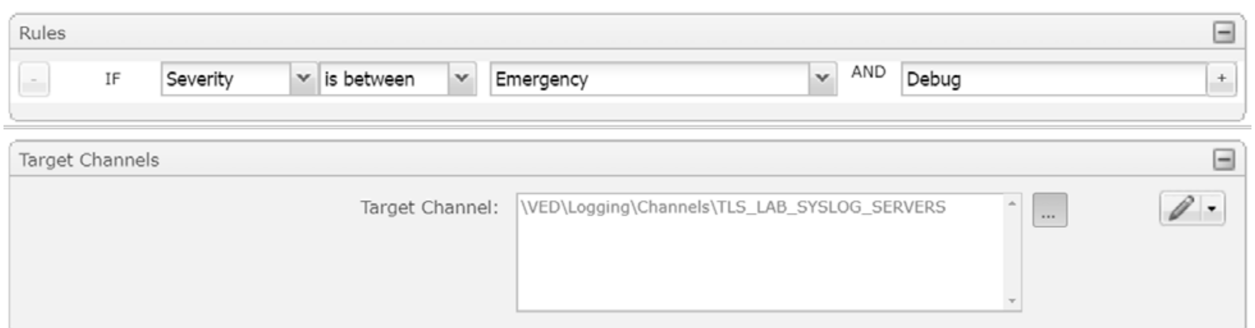


In addition to manually reviewing events within the console, it is possible to configure rules that will automatically send events. These events can be sent via a variety of different channels, including via email, to Splunk, to a syslog server, to an SNMP server, to a file, or to a database. Rules can be defined to send events based on specific criteria. For example, it is possible to send alerts prior to certificate expiration based on a configured set of days prior to expiration.

In our implementation, we configured Venafi TPP to send all events to the syslog server described in Section 1.5.5.6.

A syslog channel was created that pointed to the syslog server.

A rule was created to send a range of events from a severity of emergency to debug to the syslog channel.



This approach to sending certificate-related events to an external security information and event management (SIEM) system enables all security-related events to be centralized and analyzed cohesively.

Appendix A Passive Inspection

The example implementation demonstrates the ability to perform passive inspection of encrypted TLS connections. The question of whether or not to perform such an inspection is complex. There are important tradeoffs between traffic security and traffic visibility that each organization should consider. Some organizations prefer to decrypt internal TLS traffic, so it can be inspected to detect attacks that may be hiding within encrypted connections. Such inspection can detect intrusion, malware, and fraud, and can conduct troubleshooting, forensics, and performance monitoring. For these organizations, TLS inspection may serve as both a standard practice and a critical component of their threat detection and service assurance strategies.

The example implementation uses Symantec's SSL Visibility to perform passive inspection and is one example of how to accomplish passive inspection. The implementation demonstrates how to securely copy private keys from several different TLS servers to the SSL Visibility Appliance. The SSL Visibility Appliance can also securely replace expiring keys on servers—and immediately copy those keys to the SSL Visibility Appliance before expiration—manually and via standardized automated certificate installation.

This appendix discusses how the SSL Visibility Appliance was configured to support passive inspection. The goal was to demonstrate how to provision and revoke TLS certificates in an enterprise environment. To verify this is being done, analysis of the traffic between the TLS clients and the TLS servers was executed. The SSL Visibility Appliance can inspect traffic while located in line between the TLS clients and TLS servers on the network, or it can perform passive observation of all the network traffic between all the clients and servers mirrored to a port accessible to the server. The TLS lab configured its switching fabric to support passive monitoring of traffic utilizing traffic mirroring.

Mirroring the traffic from the virtual TLS lab environment to its physical appliances presented a few challenges. The TLS lab environment is housed within a larger VMWare and physical networking architecture. VMware's Virtual Distributed Switch Virtual Distributed Switch (VDS) provides a centralized interface for the virtual machines' access switching in the larger NCCoE environment where the TLS lab lives as a resident. The TLS lab also has its own physical switching connections several routing hops away from the NCCoE datacenter where VMWare resides. The VDS can route traffic internally between multiple labs and virtual machines within each lab. However, VDS does not mirror VMWare's local east-west traffic between virtual machines to other physical systems outside of the VDS environment. This design limits the traffic that can be mirrored from TLS' virtual machines that live on VMWare to physical switches in the TLS lab.

To remediate this issue, the NCCoE IT team worked with VMWare senior engineers on a solution. VMware advised the NCCoE IT team to configure remote SPAN (RSPAN) on the VDS. The IT team mapped the traffic to a RSPAN port that resided in a VLAN on an external switch. This external switch connects all the VMWare TLS hosts to the physical TLS lab. An additional RSPAN instance was configured on the TLS lab external switch, which is a physical NCCoE-managed and controlled device connected to all the TLS team-managed and controlled physical internal switches. The external switch was configured to carry the RSPAN traffic to the internal physical access switch in the TLS lab. A SPAN was created on the internal access switch in the TLS lab and configured as source from the RSPAN VLAN. The destination was set to the physical interface connected to the SSL Visibility Appliance.

Network packets captured from VMWare vSphere workloads must be forwarded to the physical remote monitoring appliance; the packet must traverse the switch fabric between the VMWare ESXi cluster and the physical remote monitoring appliance. Two factors must be considered from a solution feasibility perspective:

- **Low end switches**–Have limitations on how many Remote SPAN sessions can be configured to run concurrently. The switch fabric must establish a Remote SPAN Session between the VMWare ESXi cluster and physical remote monitoring appliance. An alternative solution is to deploy a robust network physical tap in lieu of leveraging the switch fabric between the VMWare ESXi cluster and physical remote monitoring appliance.
- **VMWare vSphere workloads**–VMWare High Availability Features move from one ESXi host to another, as computer resources are monitored and workloads are rescheduled. This requires the ESXi cluster to automatically re-route the path that captured packets will take from a given VM workload, as it moves from one ESXi host to another when migrated or when rescheduled by Distributed Resource Scheduler to run on another host. The captured packets must egress the ESXi cluster from the specific ESXi host on which the VM workload is running.

Successful deployment of this use case requires selection of the appropriate VMWare vSphere 6.x Port Mirroring configuration option. VMWare vSphere 6.x offers 5 options:

- Distributed Port Mirroring
- Remote Mirroring Source
- Remote Mirroring Destination
- Encapsulated Remote Mirroring (L3) Source
- Distributed Port Mirroring (Legacy)

This use case that depends on the switch fabric having a Remote SPAN configured to pass traffic between the VMWare ESXi cluster and the physical remote monitoring appliance, option 2, Remote Mirroring Source, is the appropriate choice. When configured, this option will establish a Remote SPAN VLAN that will span the VMWare distributed switch. It also utilizes the physical switch fabric and leverages a distributed port group mapped to a pre-selected/pre-configured NIC on each ESXi host in the ESXi cluster. Packets are automatically re-routed from captured VM workloads that are transient between the ESXi hosts in a VMWare vSphere ESXi cluster. When a VM workload moves, vSphere will note the change of the networking state of the VM and automatically re-establish an egress path for captured packets on the NIC of the ESXi host on which the VM is running.

Appendix B Hardening Guidance

Hardening secures systems to reduce their vulnerabilities and minimizes the attack surface, which improves security. To harden the systems, the TLS team implemented the Defense Information Agency's Security Technical Implementation Guides (STIGs). STIGs are technical configurations applied to systems to maintain their security posture. This hardening guidance provides the baseline standard for a variety of Operating Systems—see the link below to download the STIG guidance:

https://public.cyber.mil/stigs/

NIST's Security Content Automation Protocol (SCAP) is used to generate compliance reports of the security health of systems. To further strengthen security of systems, use SCAP in conjunction with STIGs. Nessus is another option that can scan for vulnerabilities and misconfigurations.

STIGs are implemented through GPOs that define policy settings for computer and user settings across the network. Configure GPOs in AD to comply with STIGs. Refer to the link below to download the current DISA STIG GPO Package and select those applicable to your environment.

https://public.cyber.mil/stigs/gpo/

Follow the steps below to implement STIGs using GPOs in AD:

1. Open Group Policy Management Console (GPMC):
    - Go to **Start** > **Administrative Tools** > **Group Policy Management**.
2. Create an OU in the domain:
    - Go to **GPMC** > right-click on the **<YOUR DOMAIN>** > click **New Organizational Unit.**
    - In the Name box on the New OU dialog box, type a descriptive name for the OU > click **OK.**
3. Create a GPO in the domain:
    - Go to **GPMC** > **<YOUR DOMAIN>** > right-click **Group Policy Objects** > click **New.**
    - In **New GPO** dialog box enter a descriptive name > click **OK.**
4. Import DISA GPOs:
    - Go to **GPMC** > **<YOUR DOMAIN>** > **Group Policy Objects** > right-click on the GPO to edit > click **Import Settings.**
    - The **Import Settings Wizard** appears > click **Next >** select the folder location of the DISA GPO being used. The TLS lab used GPOs for MS Computer, MS User, DC Computer and DC User.

    Note: To apply desired security configurations edit settings in the specific GPO.

5. Edit a GPO in the domain, an OU, or the Group Policy objects folder:
    - Go to **GPMC** > **<YOUR DOMAIN>** > select **Group Policy Objects** to display all GPOs in the domain.
    - Right-click the desired GPO > click **Edit** > the GPO will open in the Group Policy Management Editor (GPME).
    - In the GPME, edit the Group Policy settings as preferred.
6. Link a GPO to a domain or OU:
    - Go to **GPMC**> right-click **<YOUR DOMAIN>** or OU to link to the GPO > click **Link an Existing GPO.**
    - The **Select GPO** dialog box appears - > select the GPO you want linked to the domain or OU > click **OK.**

    *Shortcut: Drag the GPO from the Group Policy Objects folder and drop it onto the OU you want it linked to.
7. Optional:
    - Unlink a GPO from a domain or OU:
        - Go to **GPMC** > click **<YOUR DOMAIN>** or OU containing the GPO you want to unlink.
        - Right-click the **GPO** > click **Delete.**
        - In the Group Policy Management dialog box, confirm deletion and click **OK.**

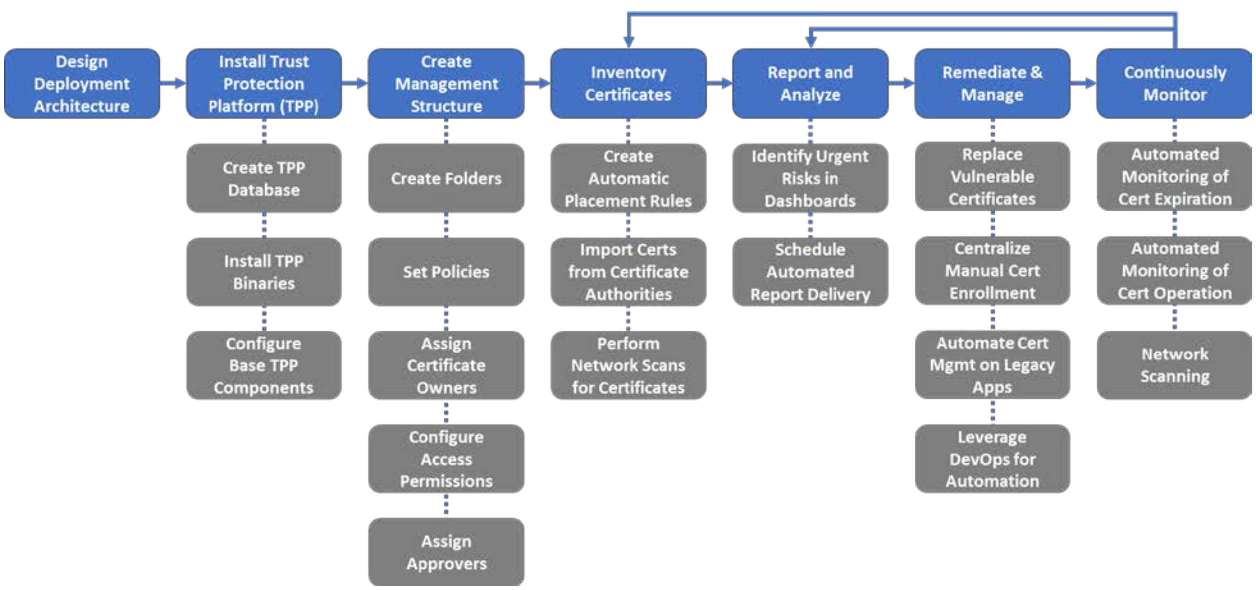
        Note: Unlink a GPO when it no longer applies. Unlinking a GPO from a domain or OU does not delete the GPO—it deletes the link. After unlinking the GPO, you can still find it in the Group Policy Objects folder.
    - Add computer to OU:
        - Go to **Start** > **Administrative Tools** > **Active Directory Users and Computers.**
        - Click on **<YOUR DOMAIN>** > refresh. The newly added OU will appear.
        - Go to **Computers** > right-click the desired computer > click **Move**.
        - Select the desired OU to move the computer to > **click OK**.
        - To apply new settings > log out and log back in.

Appendix C Venafi Underlying Concepts

The following background information may help users better understand some of the configurations we made in the configuration management databases (CMDBs) implementation of Venafi TPP.

Venafi TPP is one machine identity protection platform that enables enterprises to address TLS server certificate security and operational risks. Venafi TPP served as the certificate management platform for the TLS lab.

The following diagram illustrates the process of architecting, deploying, configuring, and using Venafi TPP to manage certificates and keys in enterprises.



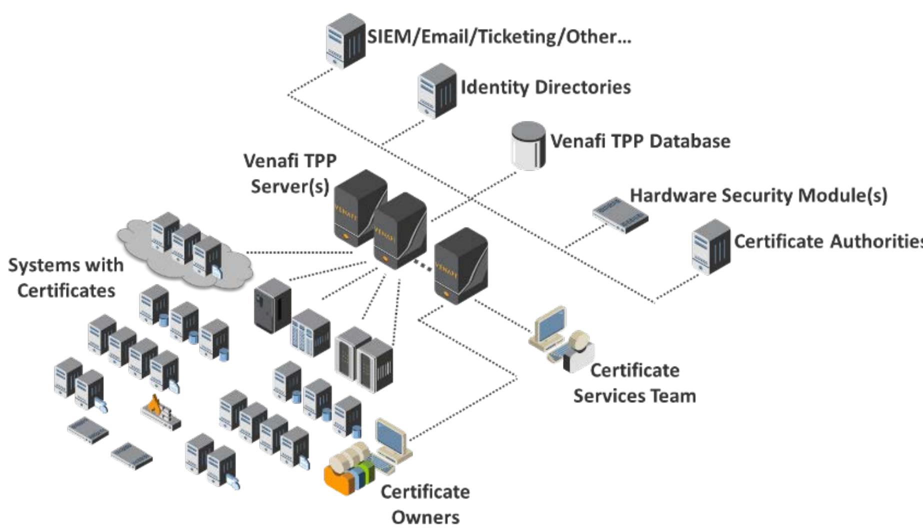
Venafi TPP interfaces with a variety of different types of systems and people/groups, including:

1. **Venafi TPP Database:** Venafi TPP requires a database to store certificates, private keys, and configuration information (all private keys and credentials are encrypted prior to storage in the database). Venafi TPP supports the use of Microsoft SQL Server to host its database.
2. **HSM:** Stores and protects the symmetric key used to encrypt private keys and credentials in the Venafi TPP database.
3. **Identity Directory:** Venafi TPP integrates with identify management systems such as AD, LDAP directories, or proprietary directories, and enables the use of existing user accounts and groups.
4. **CAs:** Venafi TPP integrates supports direct integration with over two dozen public and private CAs for the automated enrollment, renewal, and revocation of certificates.
5. **SIEM/Email/Ticketing:** Venafi TPP integrates with SIEM systems to pass certificate and cryptographic key event information. It integrates with ticketing systems for the automated

creation of change tickets and approvals and with email systems for the notifications to certificate owners for impending expirations or errors.

6. **Other Enterprise Systems:** Venafi TPP can be integrated with a variety of other enterprise systems, such as CMDBs, enterprise dashboards, and custom applications.
7. **Systems with Certificates:** Venafi TPP communicates directly with systems with certificates to automatically discover and manage those certificates.
8. **Certificate Services Team:** This team manages the Venafi TPP servers and supports Certificate Owners.
9. **Certificate Owners:** These are groups and individuals responsible for systems where certificates are deployed using Venafi TPP for automating a variety of functions, including scanning, inventory, enrollments, and installation of certificates.

The following diagram is a high-level view of these components.



Depending on an organization's needs, it's possible to deploy one or more Venafi TPP servers centrally or distributed in different network zones as well as different geographies. The number and placement of Venafi TPP servers is an important step to create an effective certificate management solution that supports the environmental and operational needs of an enterprise. The criteria driving the number and placement of Venafi TPP servers includes:

1. **Venafi TPP Services:** Each Venafi TPP can host one or more services, including network discovery scanning, certificate enrollment, certificate installation, administrative UI, etc. Depending on the size and structure of an organization, these services can be deployed on a single Venafi TPP server or, more likely, across multiple servers. The services that a Venafi TPP server can be configured to perform include:
    a. Hosting administrative and user interfaces

    b. Network discovery scanning
    c. Onboard discovery
    d. CA import
    e. Certificate expiration monitoring
    f. Certificate operation monitoring (validation)
    g. Automated certificate enrollment
    h. Agentless certificate installation
    i. Agent management
    j. CRL expiration monitoring
    k. Revocation status monitoring
    l. Report generation
    m. Venafi TPP REST API access
    n. Log event management and notifications
    o. Trust store management
2. **Load and Performance Requirements:** The number of certificates and systems that must be managed by Venafi TPP plays an important part in the choice of how many Venafi TPP servers to deploy. Venafi TPP is a based on a load-balanced architecture that enables multiple servers to share in the processing of work.
3. **Fault Tolerance:** Due to the critical role of certificate management, deployment architectures may include multiple Venafi TPP servers deployed across primary and disaster recovery sites to ensure continuous availability of certificate management services.
4. **Network Zones and Boundaries:** Network architectures often place limits on the type of traffic that can traverse between network zones (across firewalls). For example, a firewall may limit the allowed ports between two network zones, necessitating the placement of a Venafi TPP server directly inside a network zone to enable network discovery scans to run.
5. **Geographic Distribution:** Organizations are often distributed across multiple cities, states, countries, and continents. Ensuring that network latencies do not negatively impact the performance of certificate management services at each geographic location often involves distributing Venafi TPP servers near the systems and certificates being managed.

C.1 Venafi TPP Object Model

To understand how Venafi TPP maintains inventory information, first review the Venafi TPP data model. Venafi TPP uses an object-based storage model where configuration information for certificates, associated devices, and applications are stored as objects and attributes in the Venafi TPP database. Several different object types exist in Venafi TPP—each of which includes associated attributes that store data relevant to the object. For example, a certificate object includes attributes for issuer, key length, common name, organization, etc.

The object types in Venafi TPP include:

1. **Folder:** Folders are containers that facilitate the hierarchical organization certificates, devices, applications, and other objects within Venafi TPP.
2. **Certificate:** These objects hold configuration data for certificates managed by Venafi TPP, including certificate authority (CA), key length, certificate owner, approver, and other information. A certificate object can have one or more applications objects—each indicating a location where the certificate is installed.
3. **Device:** These objects hold configuration information about the systems where certificates are deployed, including the network address and port, authentication credentials, and other information for the system.
4. **Application:** These objects hold information about the specific application (e.g., Apache, F5, Java, etc.) that uses a certificate on a device. Each device may have one or more applications that use certificates. The attributes and information stored in an application object depends on the type of application. For example, an F5 application object stores information such as the SSL profile, virtual server, and partition for the associated certificate on the F5 device.
5. **Workflow:** Workflow objects store the rules that are enforced for workflow gates within Venafi TPP. They include the stage of the certificate lifecycle where approval is needed, the required approvers, and even actions that may be automatically perform when the workflow gate is triggered.
6. **CA Template:** These objects store information about CAs from which Venafi TPP requests certificates and the specific certificate templates that the CAs will use.
7. **Credential:** These objects hold credential information that Venafi TPP uses to authenticate to other systems, including CAs, systems where certificates are managed via agentless management, etc. Passwords and private keys used in credentials are stored in encrypted form in the Venafi TPP database.

C.2 Certificate Metadata in Venafi TPP

Certificates are stored in Venafi TPP in binary form (i.e., the DER encoded version of the certificate). In addition, the individual X.509 fields and extensions of each certificate are parsed and stored in unique database fields, to enable rapid searching and filtering. The certificate fields parsed and stored for rapid searching in Venafi TPP include:

- **X.509 Version:** V1, V2, or V3
- **Serial Number:** A unique identifier assigned by the issuing certificate authority
- **Issuer Distinguished Name**: The full X.500 distinguished name of the issuing-CA.
- **Valid From:** The date and time from which the certificate was issued. This is commonly referred to as an issue date.
- **Valid To:** The date and time after which the certificate should no longer be considered valid. This is commonly referred to as the expiration date.

- **Subject Distinguished Name (SAN):** The full X.500 distinguished name for the subject of the certificate (the entity to which the certificate was issued)—for example: “CN = iis2.int-nccoe.org, O = NCCOE, L = Gaithersburg, S = Maryland, C = US”.
- **Subject Alternative Names:** One or more identifiers for the subject of the certificate (the entity to which the certificate was issued). There could be additional DNS host names (e.g., server1.int-nccoe.org), IP address, or other types of identifiers.
- **Signature Algorithm:** The asymmetric and hashing algorithms that sign the certificate (e.g., sha256RSA).
- **Subject Key Identifier:** A unique identifier for the public key within the certificate. Because the public and private key are inextricably associated, this identifier applies to both of them.
- **Authority Key Identifier:** A unique identifier for the public/private key that the certificate authority uses to sign the certificate.
- **CRL Distribution Points:** One or more addresses where the CRL for the CA that issued the certificate can be retrieved.
- **AIA:** The location(s) where information and services, such as where to retrieve the CA certificate chain or access online certificate status protocol for the CA that issued the certificate.
- **Key Usage:** Defines the purposes for which the key within the certificate can be used, including digital signature, key encipherment, and key agreement.
- **Enhanced Key Usage:** Defines the purposes for which the certified public key within the certificate may be used, including server authentication, client authentication, and code signing.
- **Basic Constraints:** Defines whether the subject of the certificate is a CA and the maximum depth of certification path (number of CAs below this CA allowed).
- **Policy:** Policies defined within the certificate.
- **Key Size:** The length of the public key in the certificate.

In addition to certificate field and extension information, Venafi TPP stores other metadata relevant to each certificate, including:

- **Certificate Owner(s):** Groups and/or individual assigned to manage and receive notifications (e.g., expiration notices, processing errors, etc.) for the certificate
- **Approver(s):** Groups and/or individuals assigned to approve operations for the certificate
- **Processing Status:** Indicates whether the certificate processing is proceeding normally, is in error, or has completed
- **Processing Stage:** The current stage of processing (e.g., creating CSR, retrieving certificate from CA, installing certificate) for the certificate

- **Last Network Validation Time & Date:** The last date and time a network validation was performed to determine the operational status of the certificate
- **Network Validation Status:** The result of last network validation
- **Installation Location(s):** The devices and applications where the certificate is installed
- **CA Chain:** The chain of CA certificates from the root to the TLS server certificate
- **Management Method:** Determines if the certificate should be automatically enrolled and installed, or manually enrolled and installed
- **Log Information:** Logs of all administrative changes and automated operations performed on the certificate via Venafi TPP

C.3 Custom Fields

With thousands of certificates, it is critical that organizationally-relevant information—such as cost center, application identifiers, business unit, and applicable regulations—can be associated with certificates. As a result, searches and reporting can return the certificates most relevant to a particular group or business function. Venafi TPP supports the definition of "custom fields" that can be assigned to certificates. The value of the custom fields (e.g., Cost Center = "B123") can be assigned to individual certificates or folders, thereby flowing down and applying to all subordinate certificates. It should be noted that custom fields can be assigned to other assets such as devices associated with certificates.

C.3.1 Organizing Certificate Inventory

Many large enterprises have thousands or tens of thousands of certificates, often with hundreds of certificate owners across many different groups. To help effectively manage certificates across these broad environments, Venafi TPP enables the creation of a hierarchical folder structure where certificates and associated system configuration information can be placed.

The design of a Venafi TPP folder hierarchy for the organization of certificates is dependent on the needs and requirements of an enterprise—similar to having multiple approaches to create folder hierarchies when organizing files. However, through experience in working with many large enterprises, Venafi professional services has developed a set of guidelines, including:

- **Certificate Ownership:** The primary factor for designing a Venafi TPP hierarchy is based on the organization of certificate owners. Once a folder is assigned to a certificate owner, certificates and other assets placed within the folder automatically inherit the permissions, contacts, and approvers, so that ownership does not need to be managed on individual certificates (though ownership information can be managed on individual certificates in Venafi TPP, if necessary).
- **Policies**: Policies such as allowed key lengths, signing algorithms, and CAs are an important consideration in the organization of Venafi TPP folders.

- **Workflow and Approvals:** Workflow rules are assigned at the folder level in Venafi TPP. If an enterprise applies different workflow rules across their organizational groups, the design of the folder hierarchy may be adjusted to easily assign those rules as needed.

C.3.2 Policy Enforcement

Venafi TPP supports the enforcement of written policies through the assignment of policies to any folder within the hierarchy. It is possible to define Venafi TPP policies for a broad set of areas, including allowed CAs, allowable domains, certificate contents (e.g., key length), approvers, and application configurations.

Policies set on a folder flow down to subordinate folders and objects within the folders. This makes it possible to configure group-specific policies on folders assigned to those groups and policies with broader applicability to higher level folders, so that they apply to all certificates, devices, applications across subordinate folders. Policies can be set as suggested, to provide a default value that users are able to change if desired, or enforced, where users are required to use the set value.

C.4 Domain Whitelisting

Because certificates serve as trusted credentials, they should only be issued for authorized domains. To aid in this, Venafi TPP supports the whitelisting of domains that can be used in certificates. For example, it is possible to only allow common names (CNs) and subject alternative names (SANs) that have the suffix ".int-nccoe.org", which only allow CNs and SANs such as server1.int-nccoe.org and server2.ops.int-nccoe.org.

C.4.1 Certificate Owner Assignment

The assignment and maintenance of certificate ownership is critical to prevent outages and respond to security incidents. Depending on the size of groups and the number certificates they manage, certificate management responsibilities may be assigned to one person or distributed among several different individuals. For larger groups managing greater numbers of certificates across a broad set of systems, the roles may vary for each team member. For example, a core group of technical people may be responsible for managing the configuration of certificates. That same group plus a manager may need to receive alerts and reports. To accommodate these differences in roles, Venafi TPP enables the assignment of permissions and contact information (for sending alerts) at the certificate or folder level.

C.4.2 Permissions

In Venafi TPP, groups and individual users can be granted permissions to folders and individual objects (e.g., certificates). Venafi TPP can assign the following permissions:

- **View:** See an object in a folder and select it (but not see its configuration parameters). For example, an administrator with view rights to an application can associate that application to a certificate for which they are responsible.
- **Read:** Read an object's configuration parameters and status.
- **Write:** Edit an object's configuration parameters.
- **Create:** Create new objects under the object to which the Create permission is assigned. Applies only to objects that contain other objects.
- **Delete:** Delete the specified object or objects contained within it (unless blocked below).
- **Rename:** Rename the object.
- **Revoke:** Revoke a certificate. This only applies to certificates only but can be set on policies, devices, or applications for any certificates contained under them.
- **Associate:** Associate a certificate to one or more applications from within that certificate object.
- **Admin:** Grant users or groups permissions to the object.
- **Private-Key Read:** Retrieve the private-key for a certificate only applies to certificates but can be set on policies, devices, or applications for any certificates contained under them.
- **Private-Key Write:** Upload or overwrite the private-key for a certificate. This only applies to certificates but can be set on policies, devices, or applications for any certificates contained within them. The private-key write privilege is required for an administrator to extract a private-key and certificate from an application to be stored in the Venafi TPP database.
- **Permissions:** Permissions assigned to a folder are inherited subordinate objects and folders. Wherever possible, it's a best practice to assign permissions to groups to quickly grant a new team member the needed permissions simply by being added to the group. It is also best to assign permissions at the folder level, applying to all subordinate certificates. When a new system and certificate are needed, they can be added within the folder and the permissions automatically apply.

C.4.3 Contacts

Effectively managing certificates in an enterprise requires the ability to automatically notify the certificate owners of impending expirations, errors, or other events that affect their certificates. It's possible to assign one or more groups or individuals as "contacts" to folders or individual objects in Venafi TPP. Contact assignment to folders are inherited by the objects below them.

Appendix D List of Acronyms

ACME	Automated Certificate Management Environment
AD	Active Directory
ADCS	Active Directory Certificate Services
ADS	Active Directory Services
AIA	Authority Information Access
API	Application Programming Interface
CA	Certificate Authority
CAPI	Cryptographic Application Programming Interface (also known variously as CryptoAPI, Microsoft Cryptography API, MS-CAPI or simply CAPI)
CDP	CRL Distribution Point
CEP	Certificate Enrollment Policy
CES	Certificate Enrollment Service
CMDB	Configuration Management Database
CN	Common Name
CNG	Cryptography API: Next Generation
CPU	Central Processing Units
CRL	Certificate Revocation List
CSR	Certificate Signing Request
DB	Database
DC	Domain Controller
DevOps	Development Operations
DMZ	Demilitarized Zone
DNS	Domain Name System
EULA	End User License Agreement

EV	Extended Validation
FIPS	Federal Information Processing Standards
FQDN	Fully Qualified Domain Name
GPMC	Group Policy Management Console
GPO	Group Policies Objects
HSM	Hardware Security Module
HTML	Hypertext Markup Language
http	Hypertext Transfer Protocol
https	Hypertext Transfer Protocol Secure
IdP	Identity Provider
IETF	Internet Engineering Task Force
IIS	Internet Information Server (Microsoft Windows)
IMAP	Internet Message Access Protocol
IP	Internet Protocol
IT	Information Technology
ITL	Information Technology Laboratory
KSP	Key Storage Provider
LDAP	Lightweight Directory Access Protocol
LTM	Local Traffic Manager (F5)
MSQL	Microsoft SQL
MTA	Mail Transfer Agent
MUA	Mail User Agent
NAT	Network Address Translation
NCCoE	National Cybersecurity Center of Excellence
NIST	National Institute of Standards and Technology

NTL	Network Trust Link
NTLS	Network Trust Link Service
OS	Operating System
OVA	Open Virtualization Appliance
OVF	Open Virtualization Format
PCI-DSS	Payment Card Industry Data Security Standard
PED	PIN Entry Device
PIN	Personal Identification Number
PKI	Public Key Infrastructure
PSCP	PuTTY Secure Copy Protocol
RA	Registration Authority
RAM	Random Access Memory
REST	Representational State Transfer (API)
RHEL	Red Hat Enterprise Linux
RMF	Risk Management Framework
RSA	Rivest, Shamir, & Adleman (public key encryption algorithm)
RSPAN	Remote Switched Port Analyzer
SafeNet AT	SafeNet Assured Technologies
SAN	Subject Alternative Name
SCAP	Security Content Automation Protocol
SCEP	Simple Certificate Enrollment Protocol
SCP	Secure Copy Protocol
SIEM	Security Information and Event Management
SMTP	Simple Mail Transfer Protocol
SOAP	Simple Object Access Protocol

SP	Special Publication
SPAN	Switched Port Analyzer
SQL	Structured Query Language
SSL	Secure Socket Layer (protocol)
SSL VISIBILITY	SSL Visibility (Symantec Appliance)
STIGs	Security Technical Implementation Guides
TCP	Transmission Control Protocol
TLS	Transport Layer Security (protocol)
TMSH	Traffic Management Shell
TPP	Trust Protection Platform (Venafi)
UCS	User Configuration Set
UDP	User Datagram Protocol
UPN	User Principal Name
URL	Uniform Resource Locator
VDS	Virtual Distributed Switch
VE	Virtual Edition
VLAN	Virtual Local Area Network
WinRM	Windows Remote Management

4291

4292 # Appendix E Glossary

Active Directory	A Microsoft directory service for the management of identities in Windows domain networks.
Application	1. The system, functional area, or problem to which information technology (IT) is applied. The application includes related manual procedures as well as automated procedures. Payroll, accounting, and management information systems are examples of applications. (NIST SP 800-16) 2. A software program hosted by an information system. (NIST SP 800-137)
Authentication	Verifying the identity of a user, process, or device, often as a prerequisite to allowing access to a system's resources. (NIST SP 800-63-3)
Automated Certificate Management Environment	A protocol defined in IETF RFC 8555 that provides for the automated enrollment of certificates.
Certificate	A set of data that uniquely identifies an entity, contains the entity's public key and possibly other information, and is digitally signed by a trusted party, thereby binding the public key to the entity. Additional information in the certificate could specify how the key is used and its validity period. (NIST SP 800-57 Part 1 Rev. 4 under Public-key certificate) (Certificates in this practice guide are based on IETF RFC 5280.)
Certificate Authority	A trusted entity that issues and revokes public key certificates. (NISTIR 8149)
Certificate Chain	An ordered list of certificates that starts with an end-entity certificate, includes one or more certificate authority (CA) certificates, and ends with the end-entity certificate's Root CA certificate, where each certificate in the chain is the certificate of the CA that issued the previous certificate. By checking to see if each certificate in the chain was issued by a trusted CA, the receiver of an end-user certificate can determine whether it should trust the end-entity certificate by verifying the signatures in the chain of certificates.

Certificate Management	Process whereby certificates (as defined above) are generated, stored, protected, transferred, loaded, used, and destroyed. (CNSSI 4009-2015) (In the context of this practice guide, it also includes inventory, monitoring, enrolling, installing, and revoking.)
Certificate Revocation List	A list of digital certificates that have been revoked by an issuing CA before their scheduled expiration date and should no longer be trusted.
Certificate Signing Request	A request sent from a certificate requester to a CA to apply for a digital identity certificate. The certificate signing request contains the public key as well as other information to be included in the certificate and is signed by the private key corresponding to the public key.
Client	1. A machine or software application that accesses a cloud over a network connection, perhaps on behalf of a consumer. (NIST SP 800-146) 2. A function that uses the PKI to obtain certificates and validate certificates and signatures. Client functions are present in CAs and end entities. Client functions may also be present in entities that are not certificate holders. That is, a system or user that verifies signatures and validation paths is a client, even if it does not hold a certificate itself. (NIST SP 800-15)
Cloud Computing	A model for enabling ubiquitous, convenient, on-demand network access to a shared pool of configurable computing resources (e.g., networks, servers, storage, applications, and services) that can be rapidly provisioned and released with minimal management effort or service provider interaction. (NIST SP 800-145)
Common Name	An attribute type commonly found within a Subject Distinguished Name in an X.500 directory information tree. When identifying machines, it is composed of a fully qualified domain name or IP address.
Configuration Management	A collection of activities focused on establishing and maintaining the integrity of IT products and information systems, through control of processes for initializing, changing, and monitoring the configurations of those products and systems throughout the system development life cycle. (NIST SP 800-53 Rev. 4)

Container	A method for packaging and securely running an application within an application virtualization environment. Also known as an application container or a server application container. (NIST SP 800-190)
Cryptographic Application Programming Interface	An application programming interface (API) included with Microsoft Windows operating systems that provides services to enable developers to secure Windows-based applications using cryptography. While providing a consistent API for applications, the Cryptographic Application Programming Interface (CAPI) allows for specialized cryptographic modules (cryptographic service providers) to be provided by third parties, such as Hardware Security Module (HSM) manufacturers. This enables applications to leverage the additional security of HSMs while using the same APIs they use to access built-in Windows cryptographic service providers. (Also known variously as CryptoAPI, Microsoft Cryptography API, MS-CAPI or simply CAPI)
Cryptography API: Next Generation	The long-term replacement for the CAPI.
Demilitarized Zone	A perimeter network or screened subnet separating a more-trusted internal network from a less-trusted external network.
Development Operations (DevOps)	A set of practices for automating the processes between software development and IT operations teams, so they can build, test, and release software faster and more reliably. The goal is to shorten the systems development life cycle and improve reliability while delivering features, fixes, and updates frequently in close alignment with business objectives.
Digital Certificate	Certificate (as defined above).
Digital Signature	The result of a cryptographic transformation of data that, when properly implemented, provides origin authentication, assurance of data integrity and signatory non-repudiation. (NIST SP 800-133)
Digital Signature Algorithm	A Federal Information Processing Standard for digital signatures, based on the mathematical concept of modular exponentiations and the discrete logarithm problem. (FIPS 186-4)
Directory Service	A distributed database service capable of storing information, such as certificates and CRLs, in various nodes or servers distributed across a network. (NIST SP 800-15) (In the context of this practice

	guide, a directory services stores identity information and enables the authentication and identification of people and machines.)
Distinguished Name	An identifier that uniquely represents an object in the X.500 directory information tree. (RFC 4949 Ver 2)
Domain	A distinct group of computers under a central administration or authority.
Domain Name	A label that identifies a network domain using the Domain Naming System.
Domain Name System	The system by which Internet domain names and addresses are tracked and regulated as defined by IETF RFC 1034 and other related RFCs.
Extended Validation (EV) Certificate	A certificate used for https websites and software that includes identity information, subjected to an identity verification process standardized by the CA Browser Forum in its Baseline Requirements, which verifies the identified owner of the website for which the certificate has been issued has exclusive rights to use the domain; exists legally, operationally, and physically; and has authorized the issuance of the certificate.
Federal Information Processing Standards (FIPS)	A standard for adoption and used by federal departments and agencies that has been developed within the Information Technology Laboratory (ITL) and published by the National Institute of Standards and Technology, a part of the U.S. Department of Commerce. A FIPS covers some topic in IT to achieve a common level of quality or some level of interoperability. (NIST SP 800-161)
Hardware Security Module (HSM)	A physical computing device that provides tamper-evident and intrusion-resistant safeguarding and management of digital keys and other secrets, as well as crypto-processing. FIPS 140-2 specifies requirements for HSMs.
Host Name	Host names are most commonly defined and used in the context of DNS. The host name of a system typically refers to the fully qualified DNS domain name of that system.
Hypertext Transfer Protocol (HTTP)	A standard method for communication between clients and Web servers. (NISTIR 7387)

Internet Engineering Task Force (IETF)	The internet standards organization made up of network designers, operators, vendors, and researchers that defines protocol standards (e.g., IP, TCP, DNS) through process of collaboration and consensus.
Internet Message Access Protocol	A method of communication used to read electronic mail stored in a remote server. (NISTIR 7387)
Internet Protocol (IP)	The IP, as defined in IETF RFC 6864, is the principal communications protocol in the IETF Internet protocol suite for specifying system address information when relaying datagrams across network boundaries.
Lightweight Directory Access Protocol (LDAP)	The LDAP is a directory access protocol. In this document, LDAP refers to the protocol defined by RFC 1777, which is also known as LDAP V2. LDAP V2 describes unauthenticated retrieval mechanisms. (NIST SP 800-15)
Microservice	A set of containers that work together to compose an application. (NIST SP 800-190)
Organization	An entity of any size, complexity, or positioning within an organizational structure (e.g., a federal agency or, as appropriate, any of its operational elements). (NIST SP 800-39) This publication is intended to provide recommendations for organizations that manage their own networks (e.g., that have a chief information officer).
Outage	A period when a service or an application is not available or when equipment is not operational.
Payment Card Industry Data Security Standard	An information security standard administered by the Payment Card Industry Security Standards Council that is for organizations that handle branded credit cards from the major card schemes.
PIN Entry Device	An electronic device used in a debit, credit or smart card-based transaction to accept and encrypt the cardholder's personal identification number.
Post Office Protocol	A mailbox access protocol defined by IETF RFC 1939. POP is one of the most commonly used mailbox access protocols. (NIST SP 800-45 Version 2)

Private Key	The secret part of an asymmetric key pair that is used to digitally sign or decrypt data. (NIST SP 800-63-3)
Public CA	A trusted third party that issues certificates as defined in IETF RFC 5280. A CA is considered public if its root certificate is included in browsers and other applications by the developers of those browsers and applications. The CA/Browser Forum defines the requirements public CAs must follow in their operations.
Public Key	The public part of an asymmetric key pair that is used to verify signatures or encrypt data. (NIST SP 800-63-3)
Public Key Cryptography	Cryptography that uses separate keys for encryption and decryption; also known as asymmetric cryptography. (NIST SP 800-77)
Public Key Infrastructure (PKI)	The framework and services that provide for the generation, production, distribution, control, accounting, and destruction of public key certificates. Components include the personnel, policies, processes, server platforms, software, and workstations used for the purpose of administering certificates and public-private key pairs, including the ability to issue, maintain, recover, and revoke public key certificates. (NIST SP 800-53 Rev. 4)
Registration Authority	An entity authorized by the certification authority system (CAS) to collect, verify, and submit information provided by potential Subscribers which is to be entered into public key certificates. The term RA refers to hardware, software, and individuals that collectively perform this function. (CNSSI 4009-2015)
Representational State Transfer (REST)	A software architectural style that defines a common method for defining APIs for web services.
Risk Management Framework	The Risk Management Framework (RMF), presented in NIST SP 800-37, provides a disciplined and structured process that integrates information security and risk management activities into the system development life cycle.
Rivest, Shamir, & Adleman (RSA)	An algorithm approved in [FIPS 186] for digital signatures and in [SP 800-56B] for key establishment. (NIST SP 800-57 Part 1 Rev. 4)
Root certificate	A self-signed certificate, as defined by IETF RFC 5280, issued by a root certificate authority. A root certificate is typically securely

	installed on systems, so they can verify end-entity certificates the receive.
Root certificate authority	In a hierarchical public key infrastructure (PKI), the CA whose public key serves as the most trusted datum (i.e., the beginning of trust paths) for a security domain. (NIST SP 800-32)
Subject Alternative Name	A field in an X.509 certificate that identifies one or more fully qualified domain names, IP addresses, email addresses, URIs, or UPNs to be associated with the public key contained in a certificate.
Simple Certificate Enrollment Protocol (SCEP)	A protocol defined in an IETF internet draft specification that is used by numerous manufacturers of network equipment and software who are developing simplified means of handling certificates for large-scale implementation to everyday users, as well as referenced in other industry standards.
Secure Hash Algorithm 256	A hash algorithm that can be used to generate digests of messages. The digests are used to detect whether messages have been changed since the digests were generated. (FIPS 180-4 [March 2012])
Secure Transport	Transfer of information using a transport layer protocol that provides security between applications communicating over an IP network.
Server	A computer or device on a network that manages network resources. Examples include file servers (to store files), print servers (to manage one or more printers), network servers (to manage network traffic), and database servers (to process database queries). (NIST SP 800-47)
Service Provider	A provider of basic services or value-added services for operation of a network; generally refers to public carriers and other commercial enterprises. (NISTIR 4734)
Simple Mail Transfer Protocol (SMTP)	The primary protocol used to transfer electronic mail messages on the internet. (NISTIR 7387)
Special Publication	A type of publication issued by NIST. Specifically, the Special Publication 800-series reports on the ITL's research, guidelines, and outreach efforts in computer security, and its collaborative activities

	with industry, government, and academic organizations. The 1800 series reports the results of NCCoE demonstration projects.
System Administrator	Individual responsible for the installation and maintenance of an information system, providing effective information system utilization, adequate security parameters, and sound implementation of established Information Assurance policy and procedures. (CNSSI 4009-2015)
Team	A number of persons associated together in work or activity. (Merriam Webster) As used in this publication, a team is a group of individuals assigned by an organization's management the responsibility to carry out a defined function or set of defined functions. Designations for teams as used in this publication are simply descriptive. Different organizations may have different designations for teams that carry out the functions described herein.
Transport Layer Security (TLS)	An authentication and security protocol widely implemented in browsers and web servers. TLS is defined by RFC 5246 and RFC 8446.
Trust Protection Platform (TPP)	The Venafi Machine Identity Protection platform used in the example implementation described in this practice guide.
User Principal Name	In Windows Active Directory, this is the name of a system user in email address format, i.e., a concatenation of username, the "@" symbol, and domain name.
Validation	The process of determining that an object or process is acceptable according to a pre-defined set of tests and the results of those tests. (NIST SP 800-152)
Web Browser	A software program that allows a user to locate, access, and display web pages.

Appendix F References

U.S. Department of Commerce, *Security Requirements for Cryptographic Modules*, Federal Information Processing Standards (FIPS) Publication 140-2, (including change notices as of 12-03-2002)

Joint Task Force Transformation Initiative, *Risk Management Framework for Information Systems and Organizations: A System Life Cycle Approach for Security and Privacy*, NIST SP 800-37 Revision 2, December 2018. https://nvlpubs.nist.gov/nistpubs/SpecialPublications/NIST.SP.800-37r2.pdf

NIST Computer Security Resource Center Risk Management Framework guidance [Website], https://csrc.nist.gov/projects/risk-management/risk-management-framework-quick-start-guides

Joint Task Force Transformation Initiative, *Security and Privacy Controls for Information Systems and Organizations*, Draft NIST Special Publication (SP) 800-53 Revision 5, August 2017. https://csrc.nist.gov/CSRC/media//Publications/sp/800-53/rev-5/draft/documents/sp800-53r5-draft.pdf

E. Barker, *Recommendation for Key Management: Part 1: General*, NIST Special Publication (SP) 800-57 Part 1, Revision 4, January 2016. http://doi.org/10.6028/NIST.SP.800-57pt1r4.

P. Grassi, M. Garcia, J Fenton; *Digital Identity Guidelines*, NIST Special Publication (SP) 800-63-3, June 2017. https://csrc.nist.gov/publications/detail/sp/800-63/3/final

S. Frankel et al., *Guide to IPsec VPNs*, NIST Special Publication (SP) 800-77, Dec. 2005. http://nvlpubs.nist.gov/nistpubs/Legacy/SP/nistspecialpublication800-77.pdf

Framework for Improving Critical Infrastructure Cybersecurity, Version 1.1, National Institute of Standards and Technology, April 16, 2018. See https://nvlpubs.nist.gov/nistpubs/CSWP/NIST.CSWP.04162018.pdf

T. Dierks, E. Rescorla, *The Transport Layer Security (TLS) Protocol Version 1.2*, RFC 5246, Internet Engineering Task Force, August 2008. https://www.ietf.org/rfc/rfc5246.txt

E. Rescorla, *The Transport Layer Security (TLS) Protocol Version 1.*3, draft-ietf-tls-tls13-21, Internet Engineering Task Force, April 2006. https://www.ietf.org/rfc/rfc4346.txt

Appendix G Supplemental Architecture Configurations

G.1 Mail Server Configuration Files

The Postfix mail server and Dovecot mail client were both used to create an alert and administrative email server for all alerts received from the various TLS security components used in the TLS lab. The main.cf is the primary configuration file for Postfix and the dovecot.conf is used to configure the Dovecot mail user agent. Links to both files used in the TLS lab are provided below as a quick start to setting up the same mail server and client used in the TLS lab. The main.cf and dovecot.conf files are stored in the same repository as this Volume D document on the NCCoE web page.

- https://www.nccoe.nist.gov/sites/default/files/library/supplemental-files/sp1800-16/main.cf

- https://www.nccoe.nist.gov/sites/default/files/library/supplemental-files/sp1800-16/dovecote.conf