

National Cybersecurity Center of Excellence

Mitigating IoT-Based DDoS

Build 1 Demonstration Presentation

April 10, 2019



> Challenge

- There will be 20.4 billion connected IoT devices by 2020 (per Gartner)
- As IoT devices become more common in homes and businesses, security concerns are also increasing
- IoT devices represent one of the largest attack surfaces – Some have minimal security, are unprotected or are difficult to secure
- DDoS attacks increased by 28% in 2017 (per Akamai)
- Recently IoT devices have been exploited to launch DDoS attacks (e.g. Mirai)





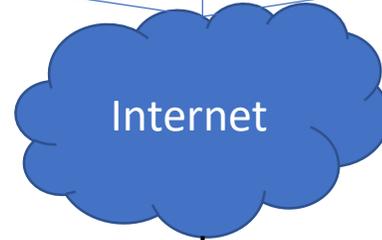
Typical Home/Small Business Network (Without MUD)



Attacker

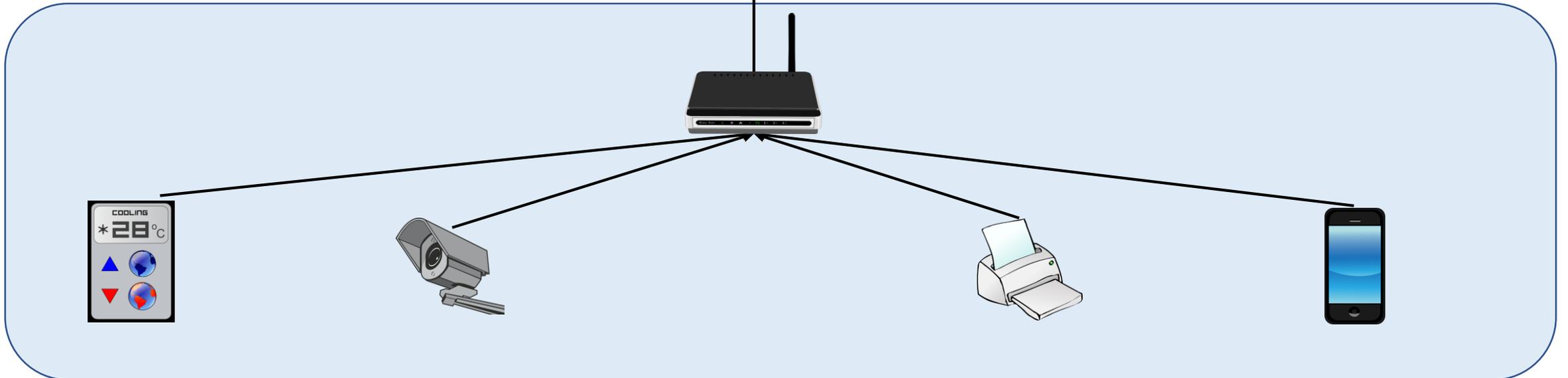


Manufacturer Server



Internet

Home/Small Business

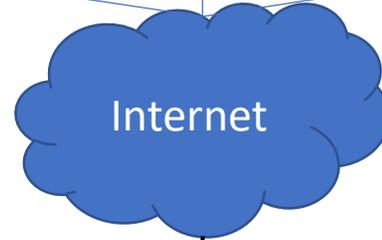




Attacker

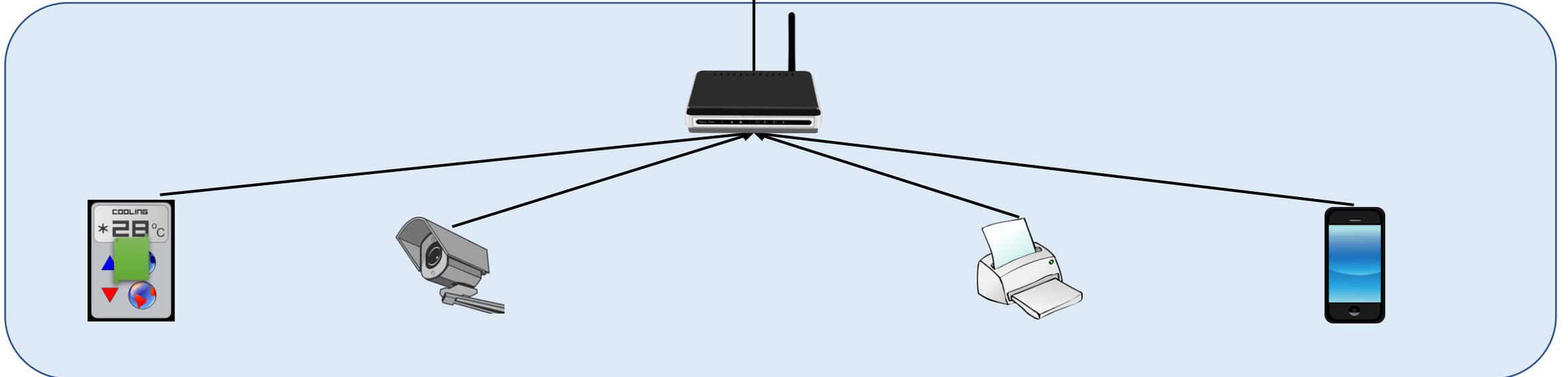


Manufacturer Server



Internet

Home/Small Business

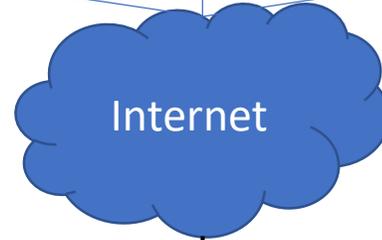




Attacker

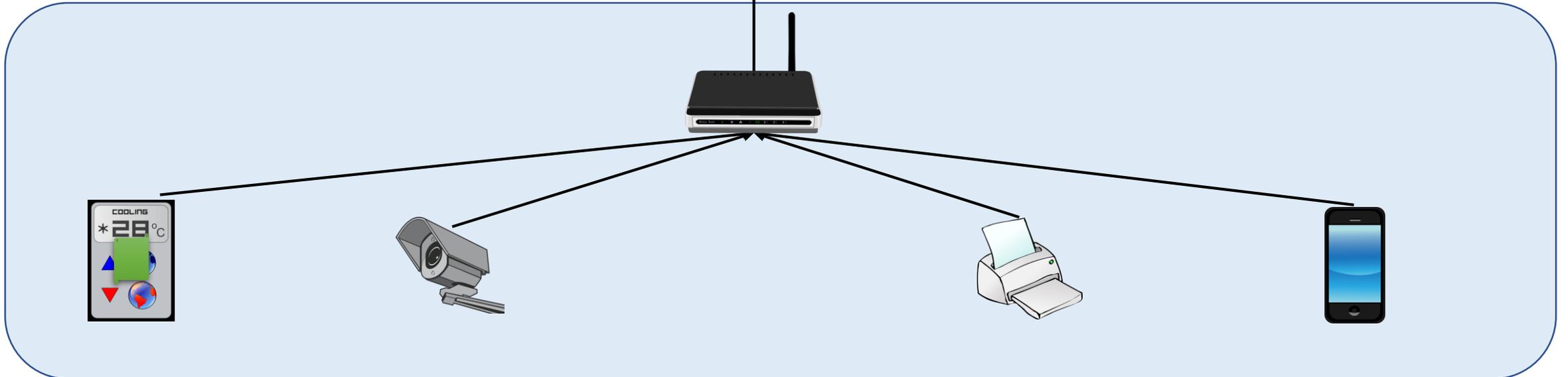


Manufacturer Server



Internet

Home/Small Business

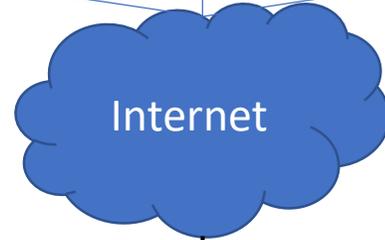




Attacker



Manufacturer Server



Internet

Home/Small Business



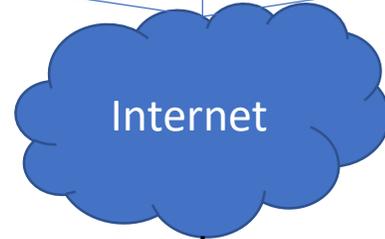


Attacker

 DynDNS



Manufacturer Server



Internet

Home/Small Business

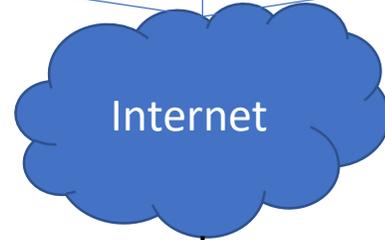




Attacker

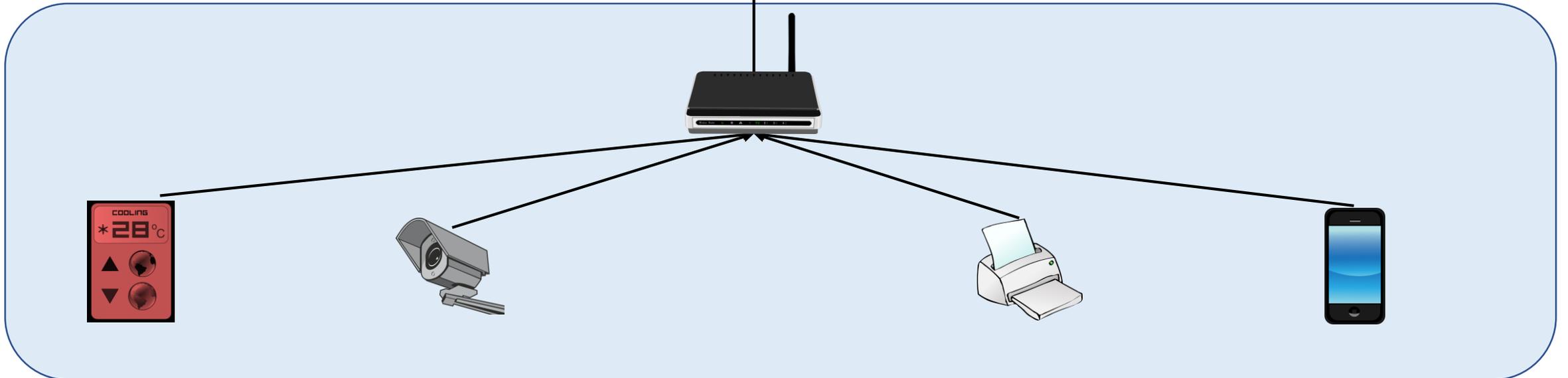


Manufacturer Server



Internet

Home/Small Business

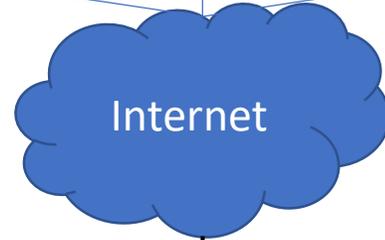




Attacker

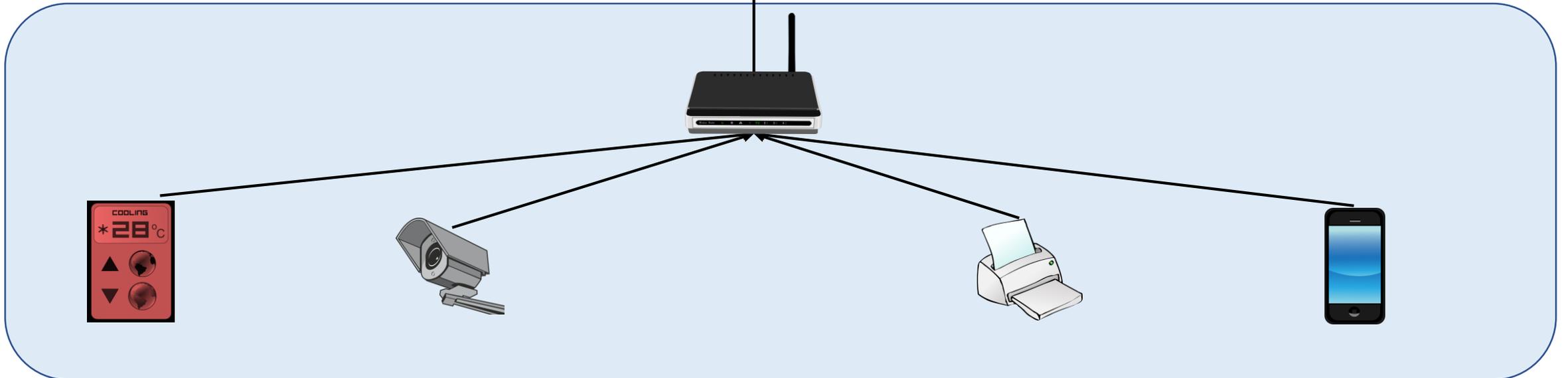


Manufacturer Server



Internet

Home/Small Business

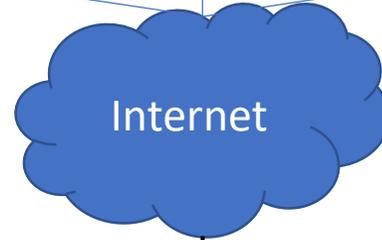




Attacker

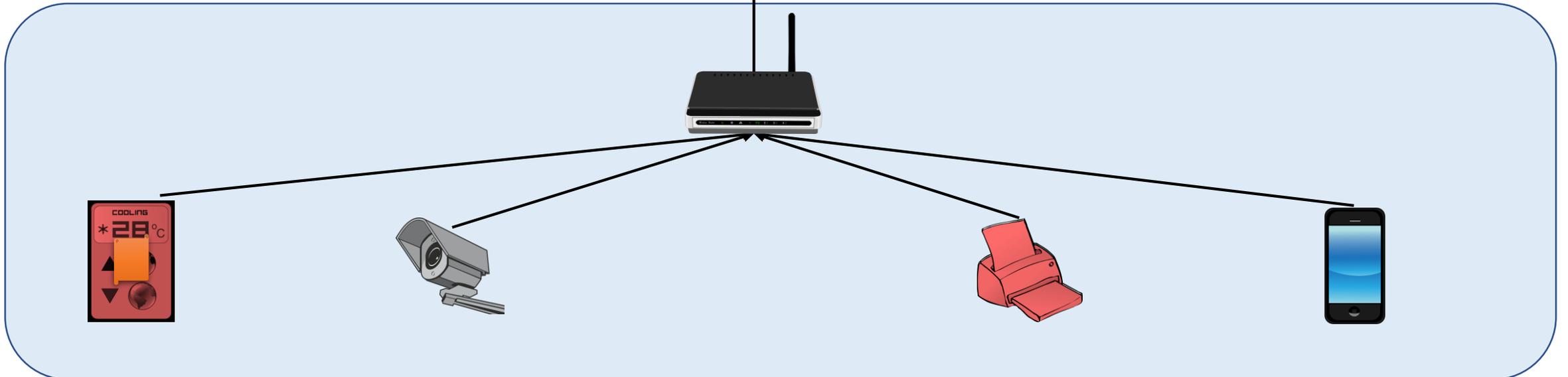


Manufacturer Server



Internet

Home/Small Business

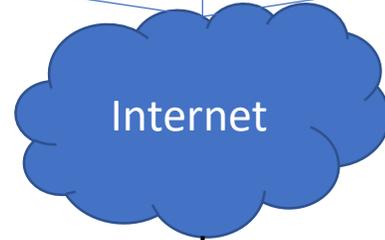




Attacker

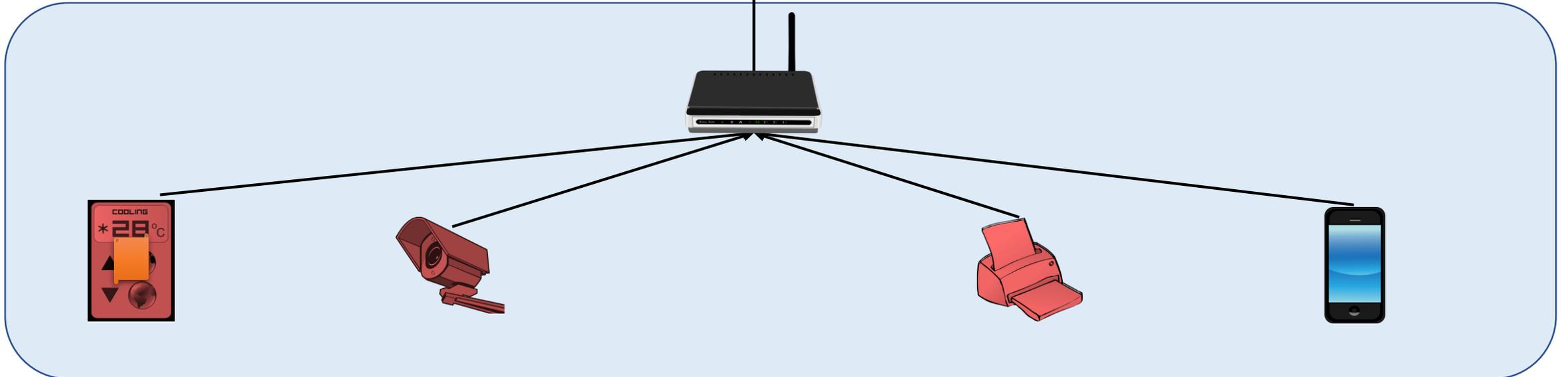


Manufacturer Server



Internet

Home/Small Business

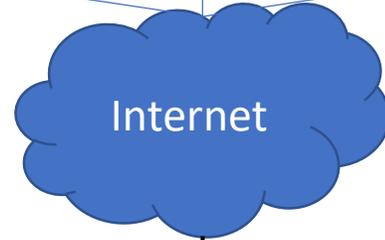




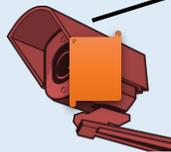
Attacker



Manufacturer Server



Home/Small Business





Typical Home/Small Business Network (With MUD)

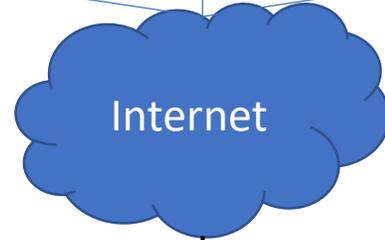


Attacker

 DynDNS

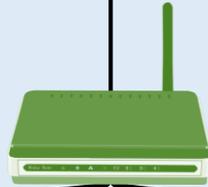


Manufacturer Server



Internet

Home/Small Business

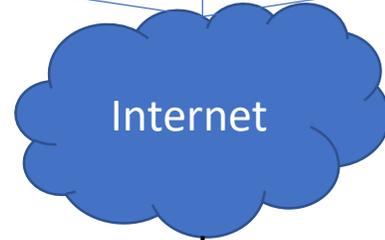




Attacker

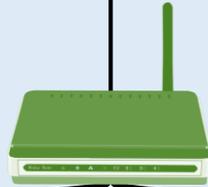


Manufacturer Server



Internet

Home/Small Business

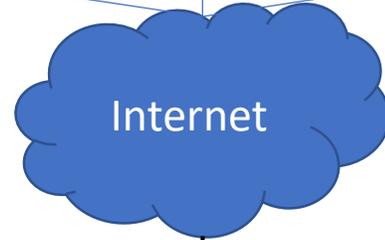




Attacker

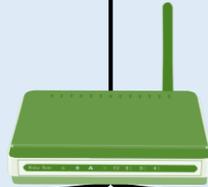


Manufacturer Server



Internet

Home/Small Business

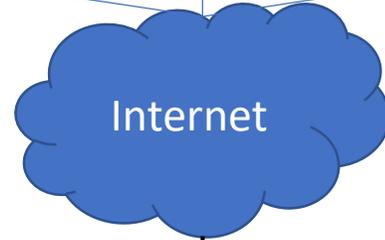




Attacker



Manufacturer Server



Internet

Home/Small Business

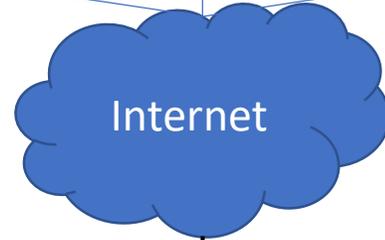




Attacker



Manufacturer Server



Internet

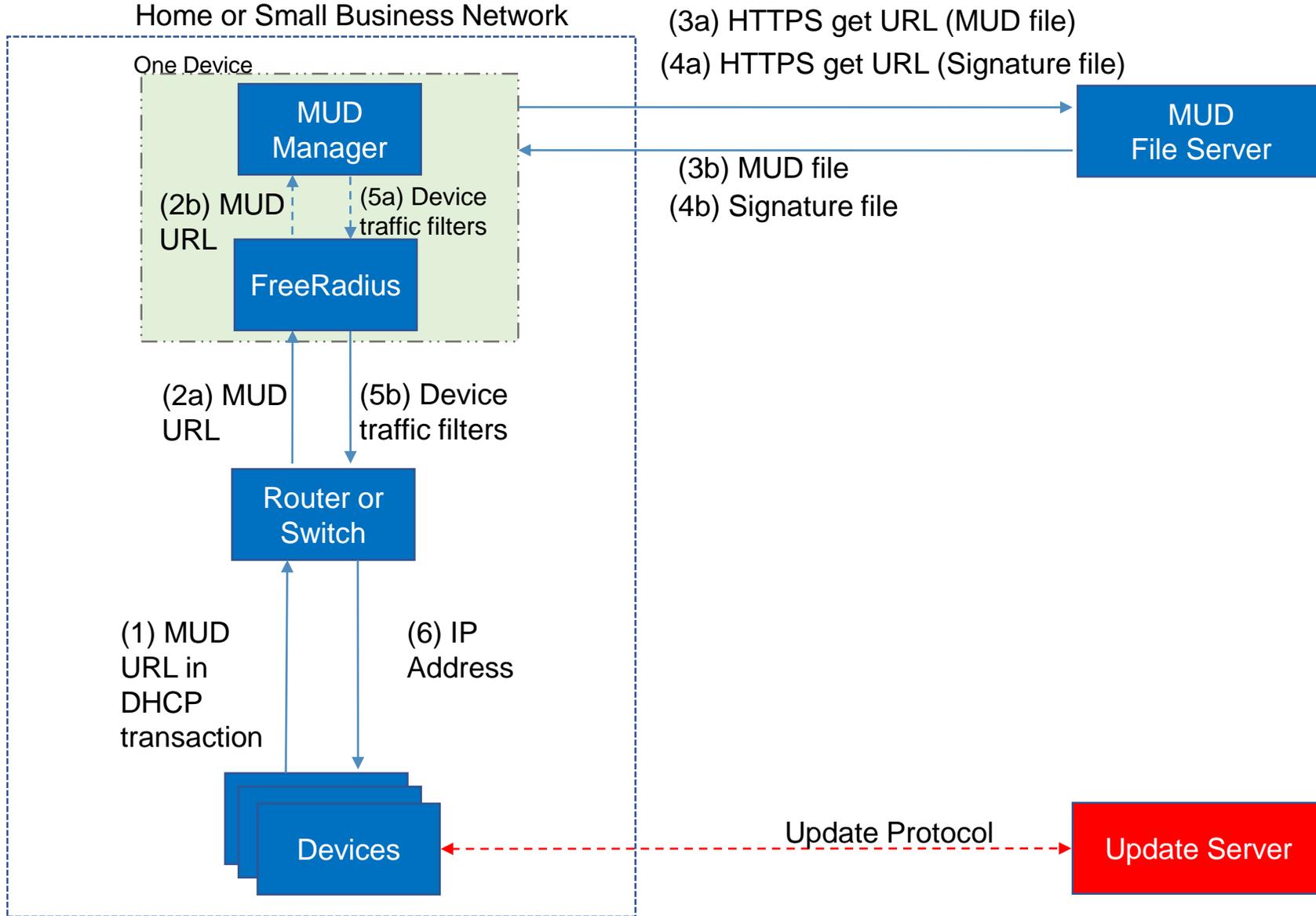
Home/Small Business





Architecture Overview

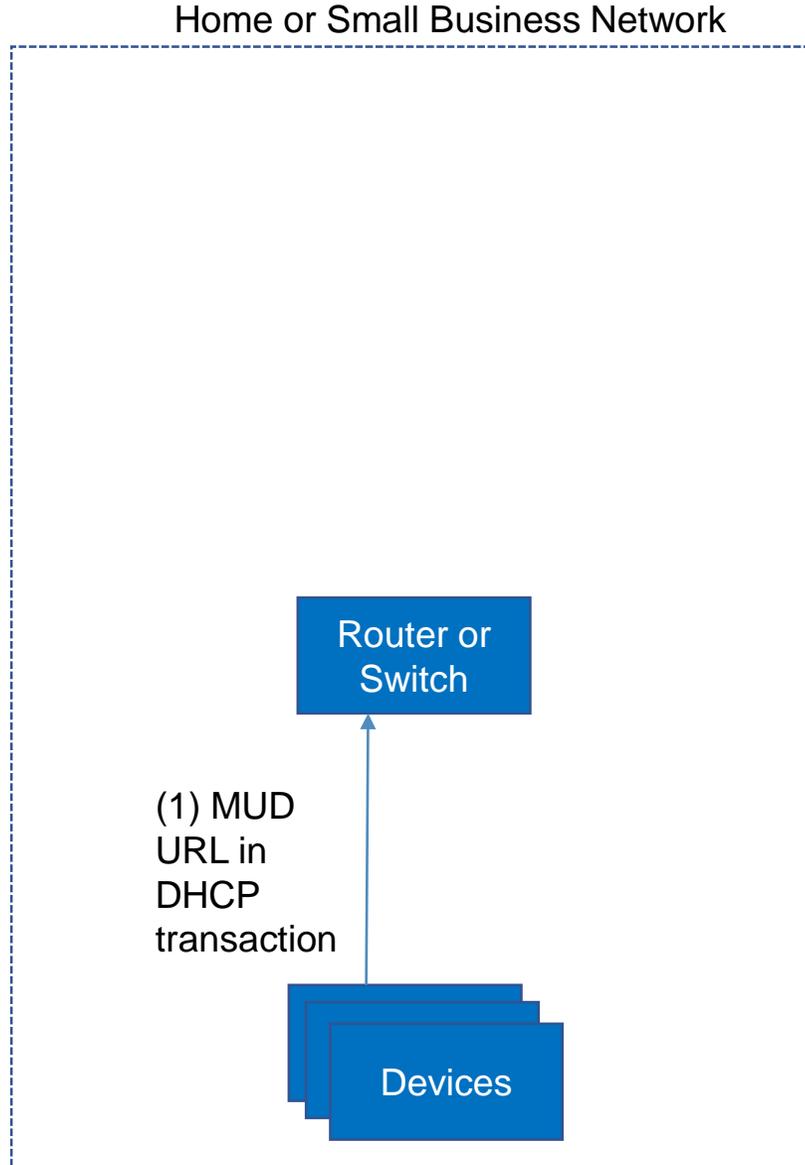
> Logical Architecture





Demonstration

> Step 1: Connect Device



> Step 1: Connect Device

1. No session on interface

Router or
Switch

```
Build1#sho access-session int g1/0/19 det  
No sessions match supplied criteria.
```

2. Connect MUD enabled IoT Device

Devices

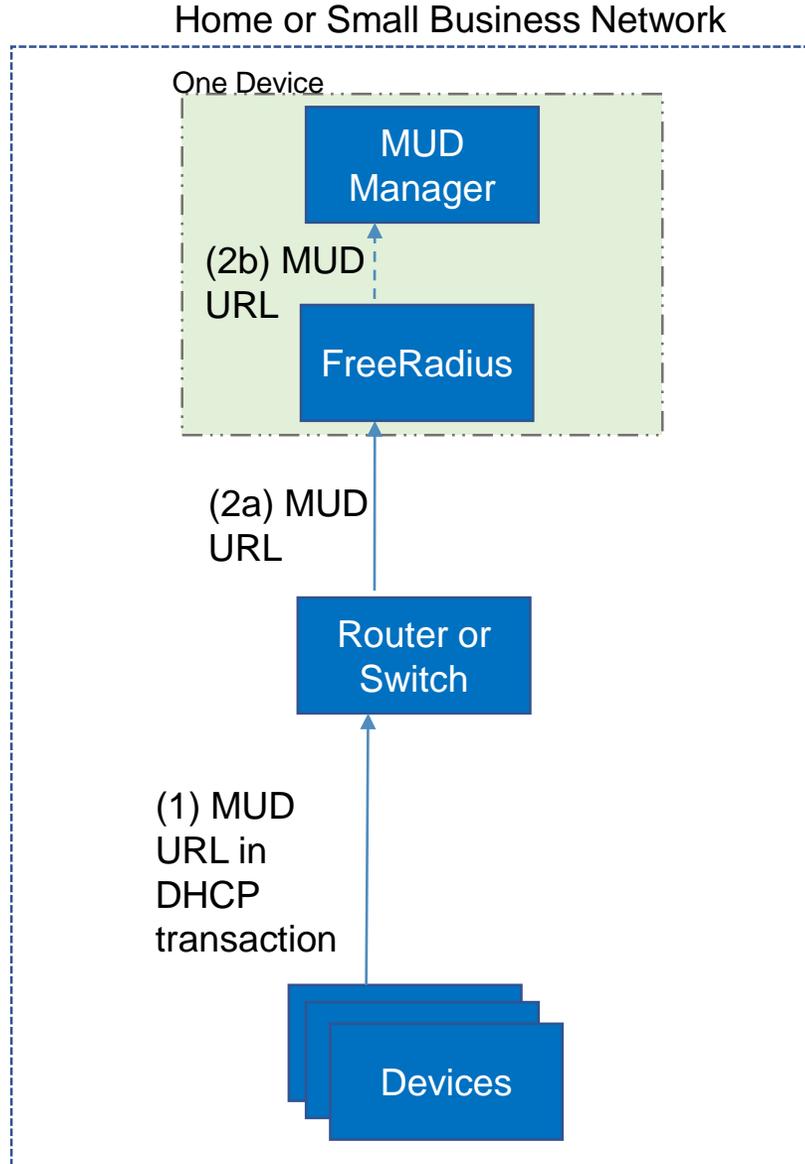
```
pi@raspberrypi:~$ sudo dhclient -v
```

3. Interface state changed to up

Router or
Switch

```
Build1#sho access-session int g1/0/19 det  
No sessions match supplied criteria.  
  
Build1#  
*Mar 26 14:19:29.140: %LINK-3-UPDOWN: Interface GigabitEthernet1/0/19, changed state to up  
*Mar 26 14:19:30.141: %LINEPROTO-5-UPDOWN: Line protocol on Interface GigabitEthernet1/0/19,
```

> Step 2a/2b: Send MUD URL to MUD Manager



> Step 2a/2b: Send MUD URL to MUD Manager

1. FreeRadius service receives and passes MUD URL

FreeRadius Server started:

Ready to process requests

Accounting Request from Switch:

```
(0) Received Accounting-Request Id 198 from 192.168.11.1:43714 to 192.168.11.45:1813 length 944
(0) Cisco-AVPair = "dhcp-option=\\000\\014\\000\\013raspberrypi"
(0) Cisco-AVPair = "dhcp-option=\\000\\377\\000aspberrypi\\241\\036https://mudfileserver/ciscopi27\\r\\
\\001\\034\\002\\003\\017\\006w\\014,\\/\\032y*\\377\\367\\007D\\212\\221$\\316\\004c\\021\\303A\\026\\370\\
\\035W\\230\\233\\224\\346o\\276L\\203E\\022\\317g\\270\\320\\332\\027e\\223\\365UT\\262\\305E"
(0) User-Name = "b827eb6c8b"
```

MUD URL and Hardware Address extracted:

```
rlm_perl: Returning MUD URL from DHCP Option: https://mudfileserver/ciscopi2
rlm_perl: Returning User-Name from 'User-Name': b827eb6c8b
```

Post sent to MUD Manager:

```
(0) rest: Sending HTTP POST to "http://127.0.0.1:8000//getaclname"
(0) rest: EXPAND \\{"%\\{Url-DataType\\}": "%\\{Url-Data\\}", "%\\{Url-AddDataType\\}": "%\\{Url-AddData\\}", "%\\{Url-
NasType\\}": "%\\{Url-Nas\\}", "%\\{Url-SessidType\\}": "%\\{Url-Sessid\\}"\\}
(0) rest: --> \\{"MAC_ADDR": "b827eb6c8b", "MUD_URI": "https://mudfileserver/
ciscopi2", "NAS": "192.168.11.1", "SESS_ID": "00000006"\\}
```

> Step 2b: Send MUD URL to MUD Manager

2. MUD Manager receives MUD enabled IoT Device information from FreeRadius Service

MUD
Manager

MUD Manager started:

```
***MUDC [INFO][main:2992]--> Starting RESTful server on port 8000
```

Post received:

```
***MUDC [INFO][mudc_print_request_info:2185]--> print parsed HTTP request header info
```

```
***MUDC [INFO][mudc_print_request_info:2186]--> request method: POST
```

```
***MUDC [INFO][mudc_print_request_info:2187]--> request uri: /getaclname
```

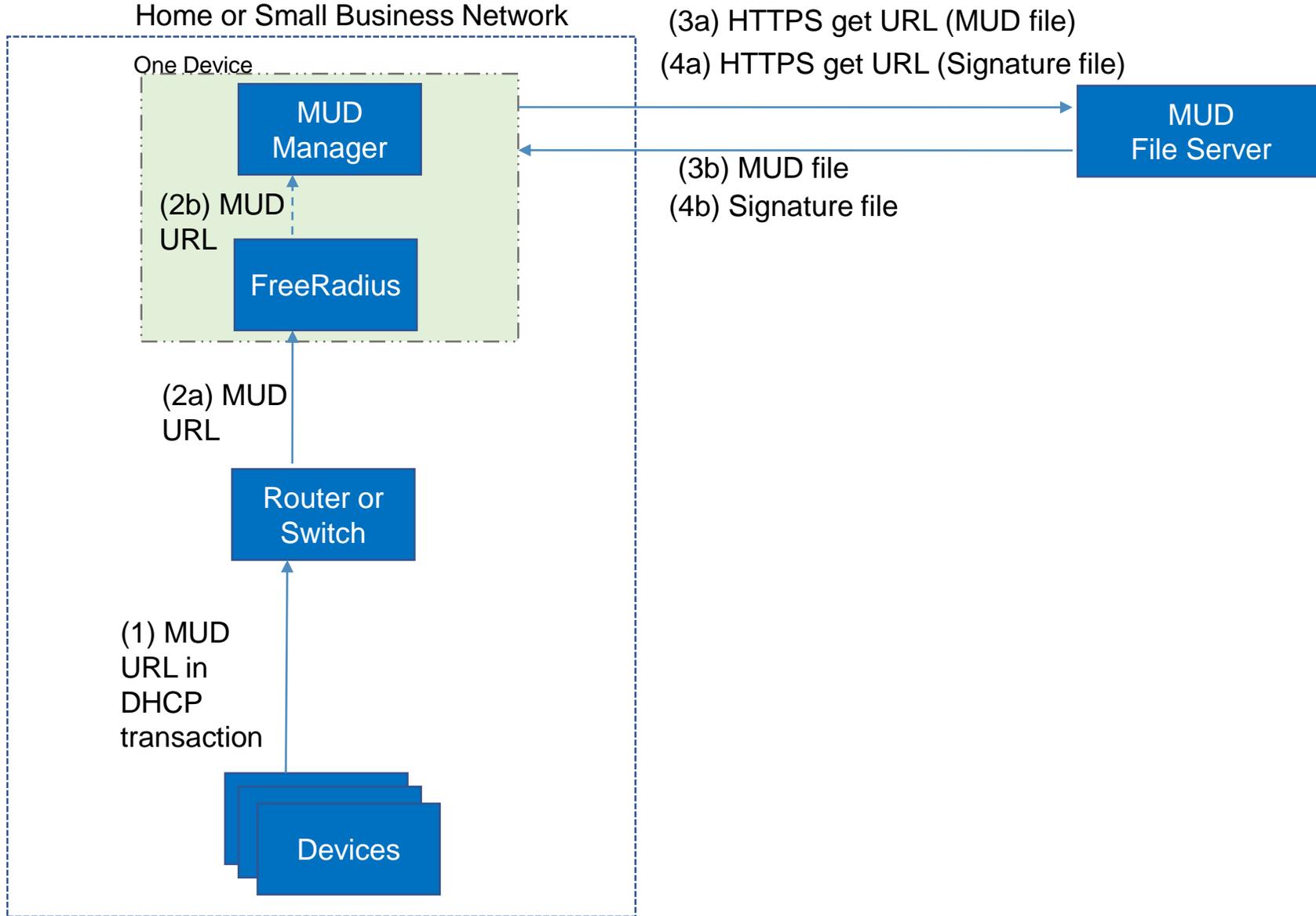
```
***MUDC [INFO][mudc_print_request_info:2199]--> header(1): name: <User-Agent>, value: <FreeRADIUS 3.0.17>
```

Check Database for Hardware Address of Device:

```
***MUDC [INFO][handle_get_aclname:2506]--> Mac address <b827ebeb6c8b>
```

```
***MUDC [INFO][handle_get_aclname:2522]--> No URL found in macaddr db for MAC address b827ebeb6c8b
```

> Step 3/4: Get MUD and Signature File



> Step 3/4: Send MUD URL to MUD Manager

1. MUD Manager receives message

MUD
Manager

```
MUD Manager started:  
***MUDC [INFO][main:2992]--> Starting RESTful server on port 8000  
Post received:  
***MUDC [INFO][mudc_print_request_info:2185]--> print parsed HTTP request header info  
***MUDC [INFO][mudc_print_request_info:2186]--> request method: POST  
***MUDC [INFO][mudc_print_request_info:2187]--> request uri: /getaclname  
***MUDC [INFO][mudc_print_request_info:2199]--> header(1): name: <User-Agent>, value: <FreeRADIUS 3.0.17>
```

2. Get MUD and Signature file

MUD
Manager

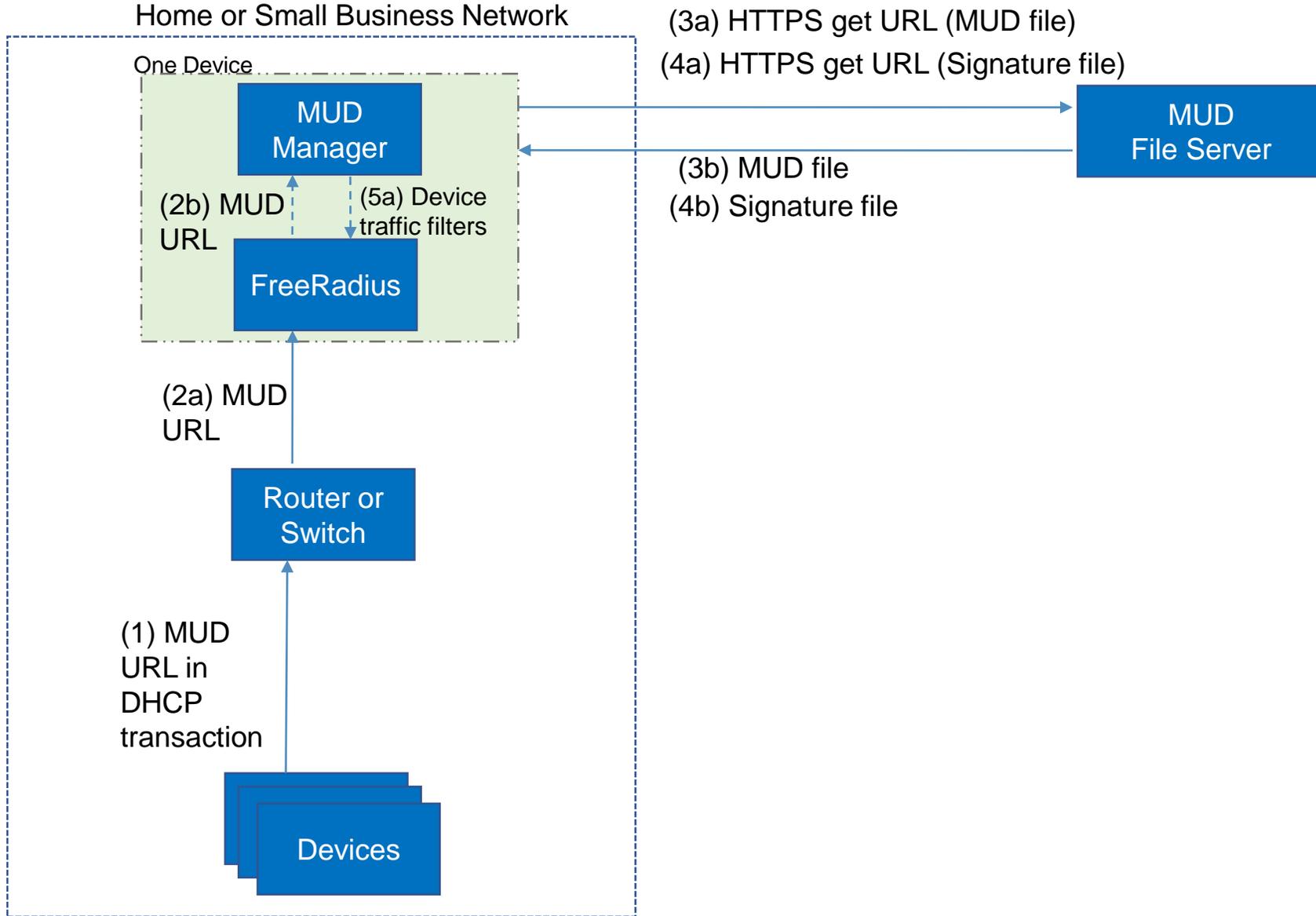
```
Get MUD File:  
***MUDC [INFO][handle_get_aclname:2558]--> Got URL from message <https://mudfileserver/ciscopi2>  
***MUDC [STATUS][send_mudfs_request:2005]--> Request URI <https://mudfileserver/ciscopi2> </home/  
mudtester/mud-intermediate.pem>  
> GET /ciscopi2.json HTTP/1.1  
***MUDC [INFO][send_mudfs_request:2033]--> MUD file successfully retrieved  
Get MUD Signature:  
***MUDC [STATUS][send_mudfs_request:2060]--> Request signature URI <https://mudfileserver/ciscopi2.p7s>  
</home/mudtester/mud-intermediate.pem>  
> GET /ciscopi2.p7s HTTP/1.1  
***MUDC [INFO][send_mudfs_request:2088]--> MUD signature file successfully retrieved
```

3. Verify MUD file

MUD
Manager

```
Verify MUD File:  
***MUDC [INFO][verify_mud_content:1609]--> Verification Successful
```

> Step 5a: Send Device Traffic Filters



> Step 5a: Send Device Traffic Filters

MUD
Manager

1. MUD File parsed and translated to ACL (rules)

MUD File Parsed and Rules Create:

```
***MUDC [INFO][create_cisco_dacl_policy:63]--> ACLName <mud-81726-v4fr> 0
***MUDC [INFO][create_cisco_dacl_policy:95]--> Ace Count <7>
***MUDC [INFO][create_cisco_dacl_policy:243]--> Returning parsed_json [{
  "DACL_Name": "ACS:CiscoSecure-Defined-ACL=mud-81726-v4fr.in",
  "DACL": ["ip:inacl#10=permit tcp any host 192.168.4.7 range 80 80 syn ack", "ip:inacl#20=permit tcp
any host 192.168.10.104 range 80 80", "ip:inacl#30=permit tcp any host 192.168.10.105 range 80 80",
"ip:inacl#40=permit tcp any host 192.168.10.125 range 80 80", "ip:inacl#50=permit tcp any 192.168.10.0
0.0.0.255 range 80 80", "ip:inacl#60=permit tcp any 192.168.13.0 0.0.0.255 range 80 80",
"ip:inacl#70=permit tcp any 192.168.14.0 0.0.0.255 range 80 80", "ip:inacl#80=permit tcp any eq 22 any",
"ip:inacl#81=permit udp any eq 68 any eq 67", "ip:inacl#82=permit udp any any eq 53", "ip:inacl#83=deny
ip any any"],
  "VLAN": 3
}]
```

2. MUD Manager sends ACL

MUD
Manager

Send Rules to Switch through FreeRadius Server:

```
***MUDC [INFO][attempt_coa:1915]--> Initiating CoA for Acct-Session-Id: 00000006
Sent CoA-Request Id 89 from 0.0.0.0:36772 to 192.168.11.1:1700 length 89
```

> Step 5a: Send Device Traffic Filters

3. FreeRadius receives ACL from MUD Manager

FreeRadius

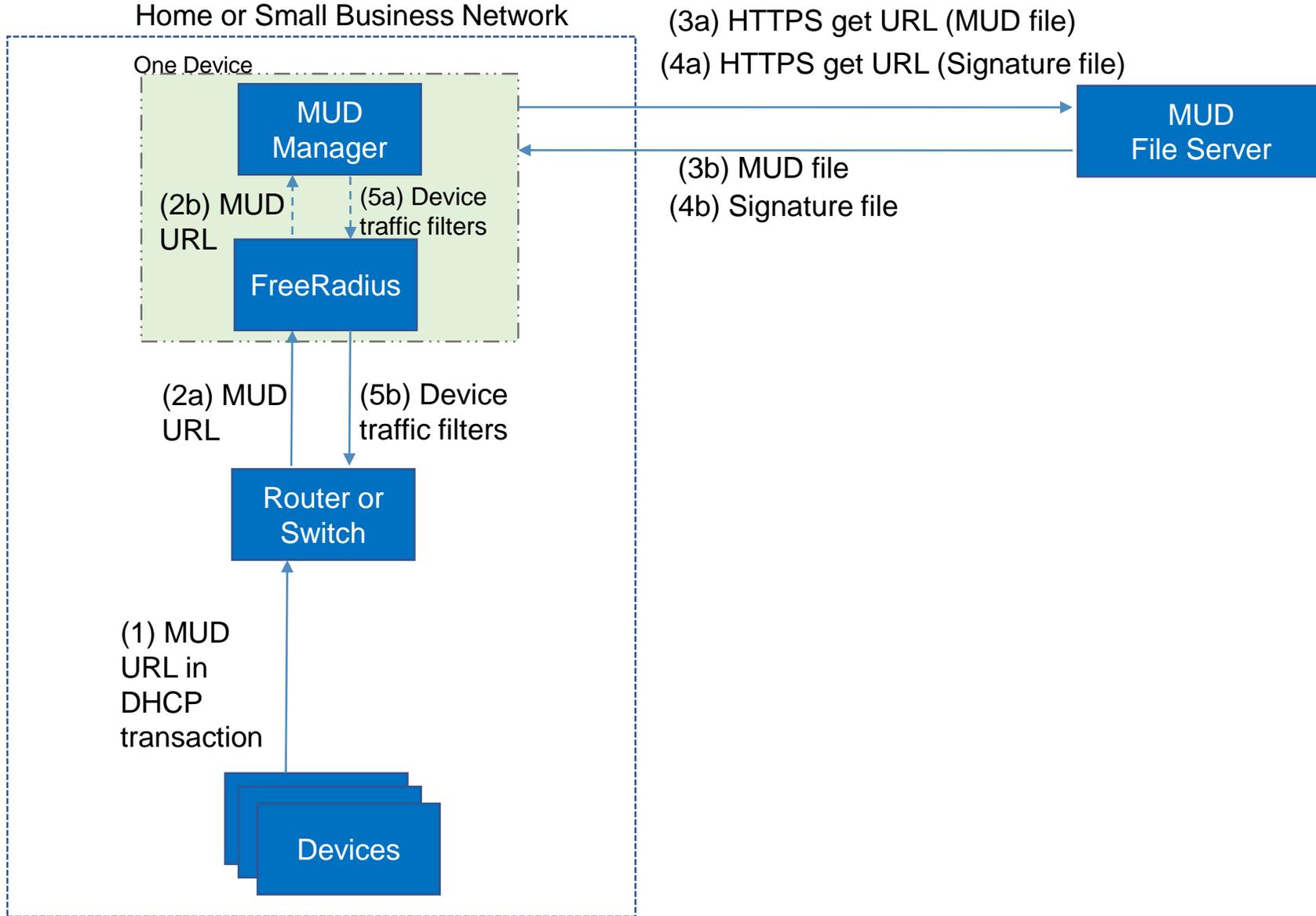
Post sent to MUD Manager:

```
(0) rest: Sending HTTP POST to "http://127.0.0.1:8000//getaclname"  
(0) rest: EXPAND \{"%\{Url-DataType\}": "%\{Url-Data\}", "%\{Url-AddDataType\}": "%\{Url-AddData\}", "%\{Url-NasType\}": "%\{Url-Nas\}", "%\{Url-SessidType\}": "%\{Url-Sessid\}"\}  
(0) rest: --> \{"MAC_ADDR": "b827eb6c8b", "MUD_URI": "https://mudfileserver/ciscopi2", "NAS": "192.168.11.1", "SESS_ID": "00000006"\}
```

ACL received:

```
(0) rest: Parsing attribute "Cisco-AVPair"  
(0) rest: EXPAND ACS:CiscoSecure-Defined-ACL=mud-81726-v4fr.in  
(0) rest: --> ACS:CiscoSecure-Defined-ACL=mud-81726-v4fr.in  
(0) rest: Cisco-AVPair := "ACS:CiscoSecure-Defined-ACL=mud-81726-v4fr.in"
```

> Step 5b: Send Device Traffic Filters



> Step 5b: Send Device Traffic Filters

FreeRadius

1. FreeRadius sends ACL to switch

Sending ACLs to Switch:

```
Sent Accounting-Response Id 198 from 192.168.11.45:1813 to 192.168.11.1:43714 length 0  
(0) Cisco-AVPair = "ACS:CiscoSecure-Defined-ACL=mud-81726-v4fr.in"
```

Request completed:

```
(0) Finished request
```

2. ACL received and configurations applied

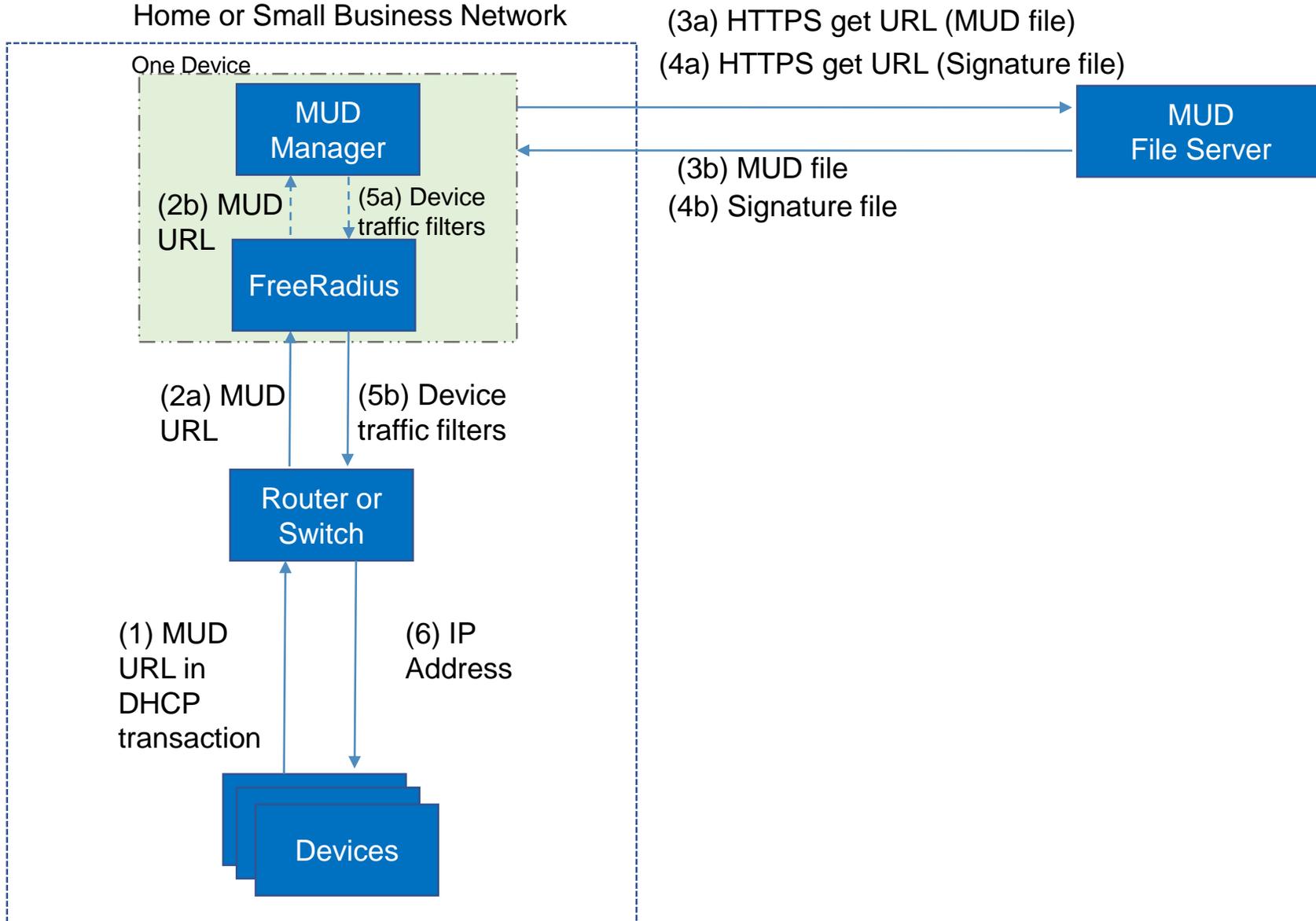
Router or
Switch

```
Build1#sho access-session int g1/0/19 det  
No sessions match supplied criteria.
```

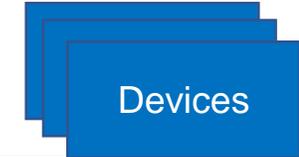
```
Build1#
```

```
*Mar 26 14:19:29.140: %LINK-3-UPDOWN: Interface GigabitEthernet1/0/19, changed state to down  
*Mar 26 14:19:30.141: %LINEPROTO-5-UPDOWN: Line protocol on Interface GigabitEthernet1/0/19, changed state to down  
*Mar 26 14:20:14.301: %LINK-3-UPDOWN: Interface Vlan3, changed state to up  
*Mar 26 14:20:15.301: %LINEPROTO-5-UPDOWN: Line protocol on Interface Vlan3, changed state to up
```

> Step 6: IP Address Assigned



> Step 6: IP address assigned



1. IoT Device receives IP address

```
DHCPDISCOVER on eth0 to 255.255.255.255 port 67 interval 16
DHCPREQUEST of 192.168.13.22 on eth0 to 255.255.255.255 port 67
DHCPOFFER of 192.168.13.22 from 192.168.13.1
DHCPACK of 192.168.13.22 from 192.168.13.1
Too few arguments.
Too few arguments.
bound to 192.168.13.22 -- renewal in 19835 seconds.
pi@raspberrypi:~$
```

> Step 6: IP address assigned

1. Show access-session

Router or
Switch

```
Build1#sho access-session int g1/0/19 det
  Interface: GigabitEthernet1/0/19
    IIF-ID: 0x125ECD95
  MAC Address: b827.ebcf.7b81
  IPv6 Address: Unknown
  IPv4 Address: 192.168.13.22
  User-Name: b827ebcf7b81
  Status: Authorized
  Domain: DATA
  Oper host mode: multi-auth
  Oper control dir: both
  Session timeout: N/A
  Common Session ID: C0A80A0200000068BA5F00E3
  Acct Session ID: 0x00000012
  Handle: 0x9b00005e
  Current Policy: mud-mab-test

Server Policies:
  ACS ACL: mud-81726-v4fr.in
  Vlan Group: Vlan: 3

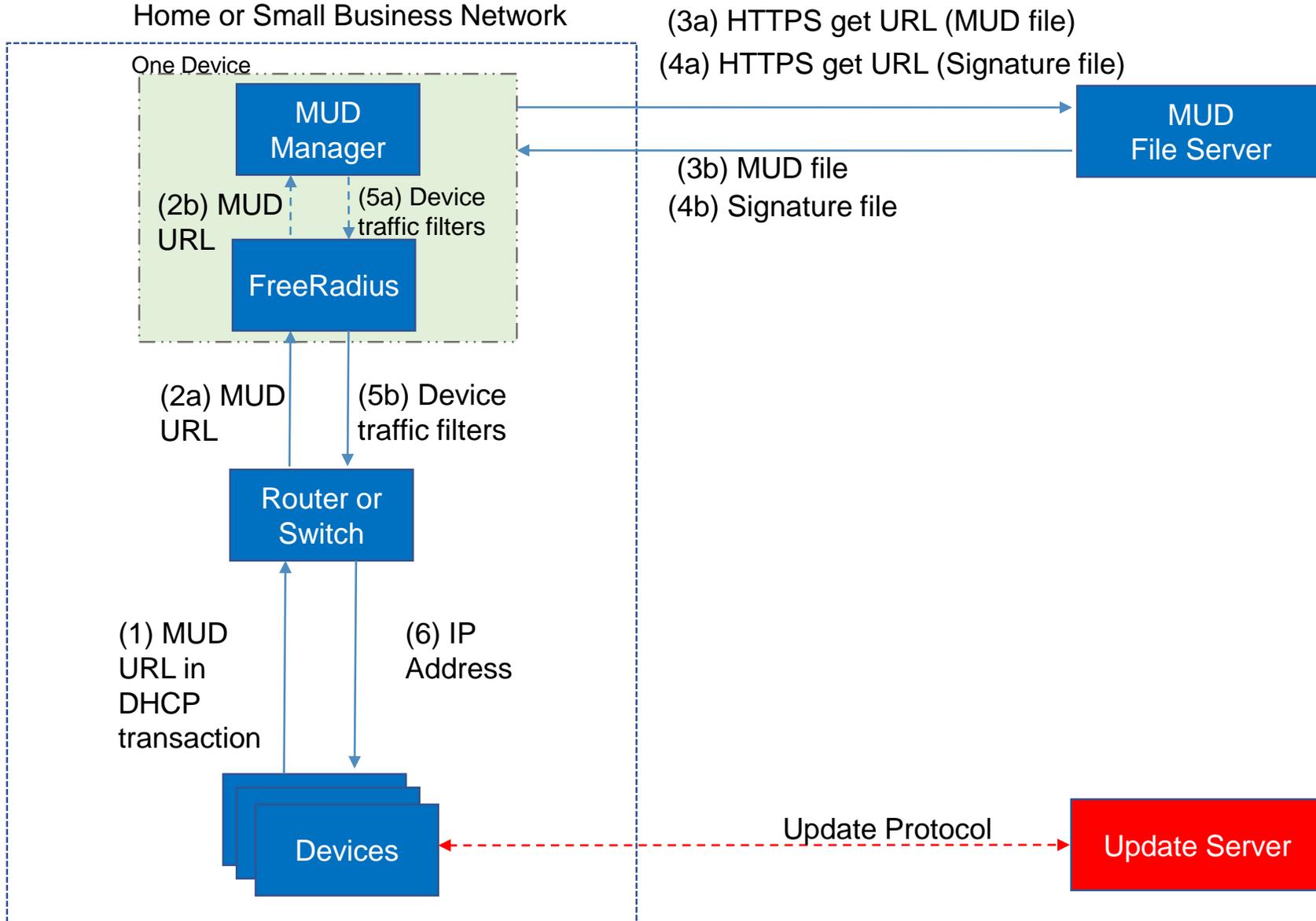
Method status list:
  Method      State
  mab         Authc Success
```

2. Show access-lists

Router or
Switch

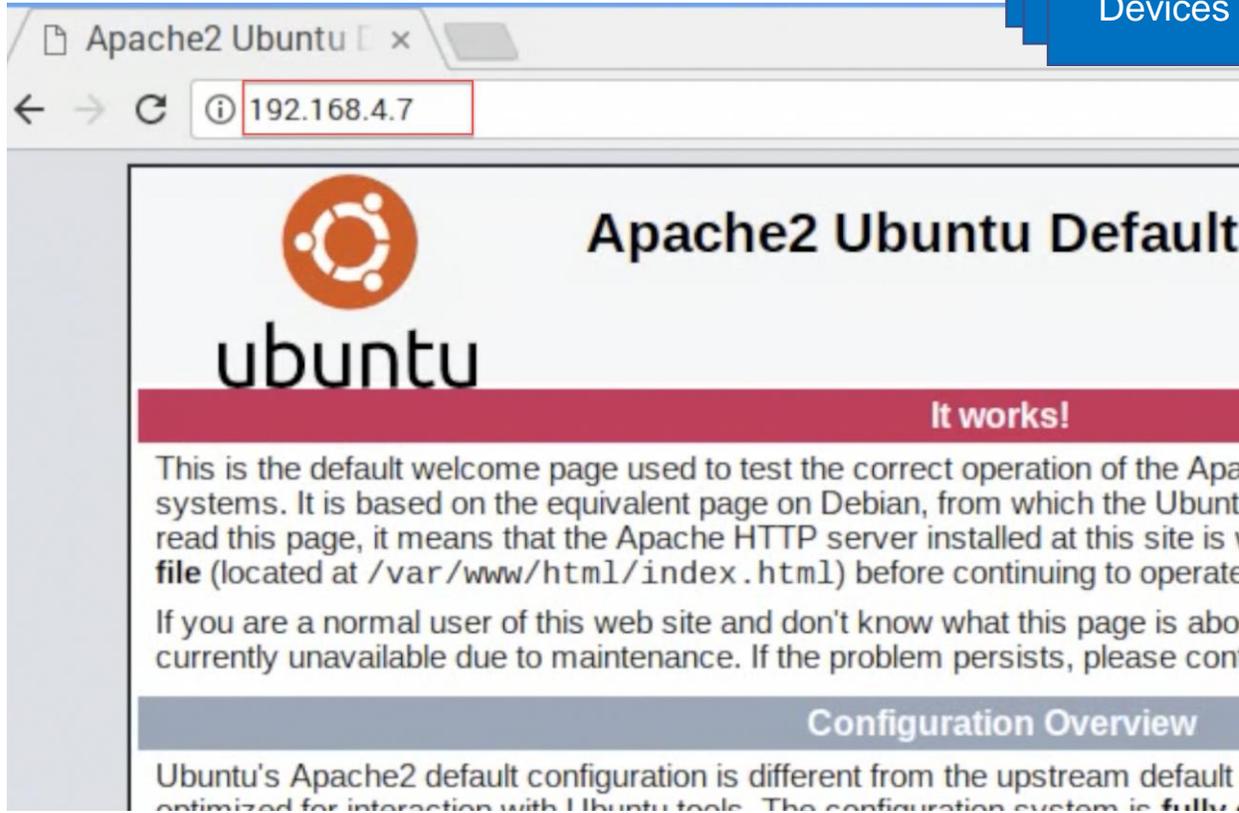
```
Build1#sho access-lists
Extended IP access list mud-81726-v4fr.in
 10 permit tcp any host 192.168.4.7 eq www ack syn
 20 permit tcp any host 192.168.10.104 eq www
 30 permit tcp any host 192.168.10.105 eq www
 40 permit tcp any host 192.168.10.125 eq www
 50 permit tcp any 192.168.10.0 0.0.0.255 eq www
 60 permit tcp any 192.168.13.0 0.0.0.255 eq www
 70 permit tcp any 192.168.14.0 0.0.0.255 eq www
 80 permit tcp any eq 22 any
 81 permit udp any eq bootpc any eq bootps
 82 permit udp any any eq domain
 83 deny ip any any
```

> Step 7: Test communication

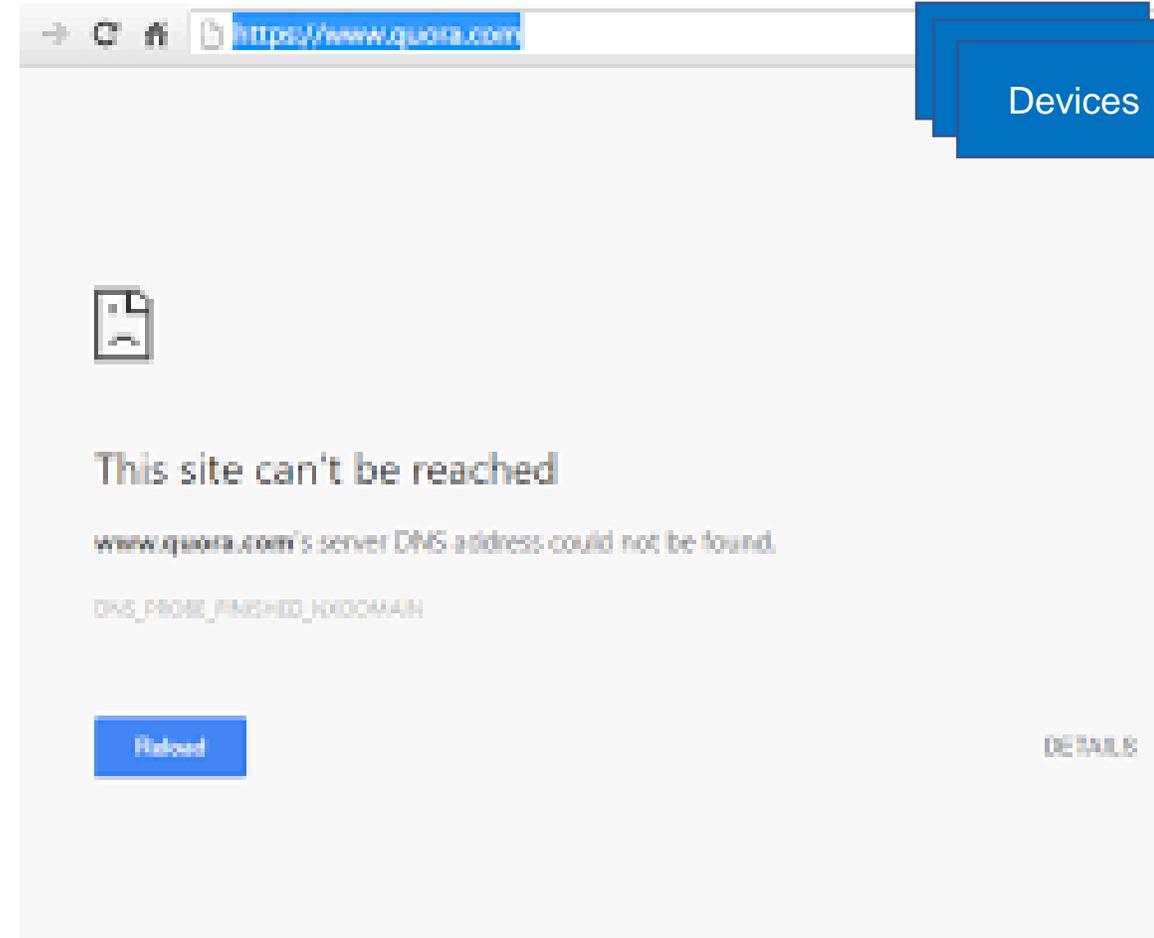


> Step 7: Test communication

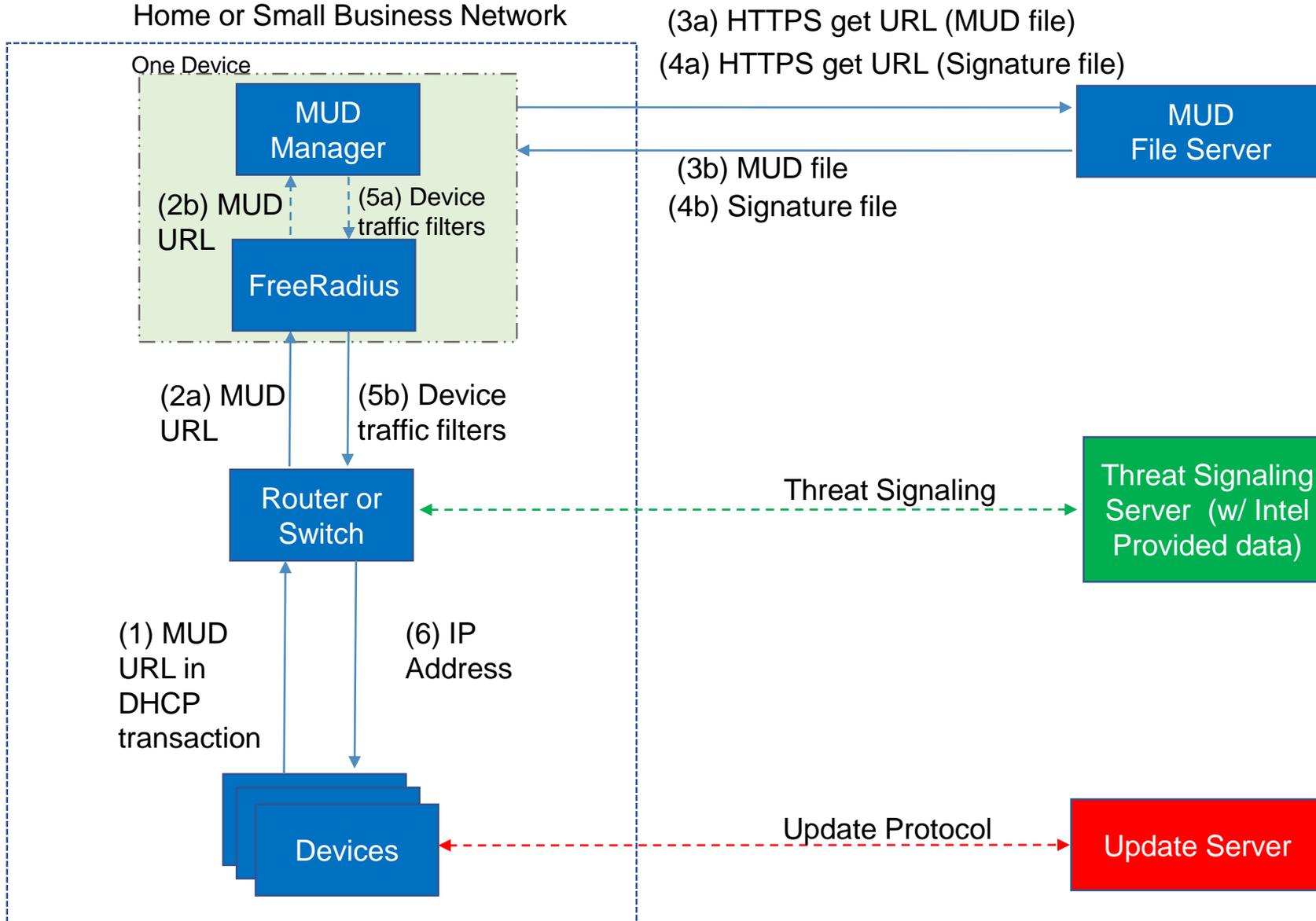
1. Test browsing to “Update Server”



2. Test browsing to unapproved server



> Next Steps





Questions

