IoT Open House
Tech Deep Dive
Build 3 – BRSKI

Michael Richardson
Sandelman Software Works Inc

RFC8995
BRSKI
RFC8366

Anxiety, keep on tryin' me
I feel it quietly
Tryin' to silence me, yeah
My anxiety, can't shake it off of me
Somebody's watchin' me
And my anxiety, yeah
Oh
Oh, oh, oh, oh, oh

https://www.sandelman.ca/SSW/talk/2025-ssw-nccoe-iot-build3/

B ootstraping
R emote
S ecure
K ey
I nfrastructure

B ootstraping
R emote
S ecure
K ey
I nfrastructure

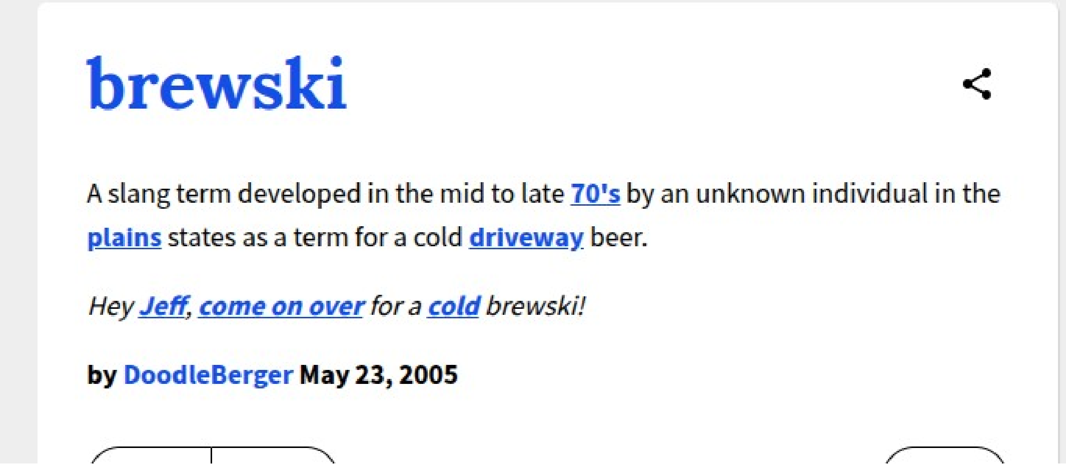
brewski

The way us canadians say beer. Though unlike what loserkid said i don't think it's all that hickish.

Hey dude can you grab me another brewski.

by Mos Def And Kweli are Blackstar **February 26, 2005**

B ootstraping
R emote
S ecure
K ey
I nfrastructure

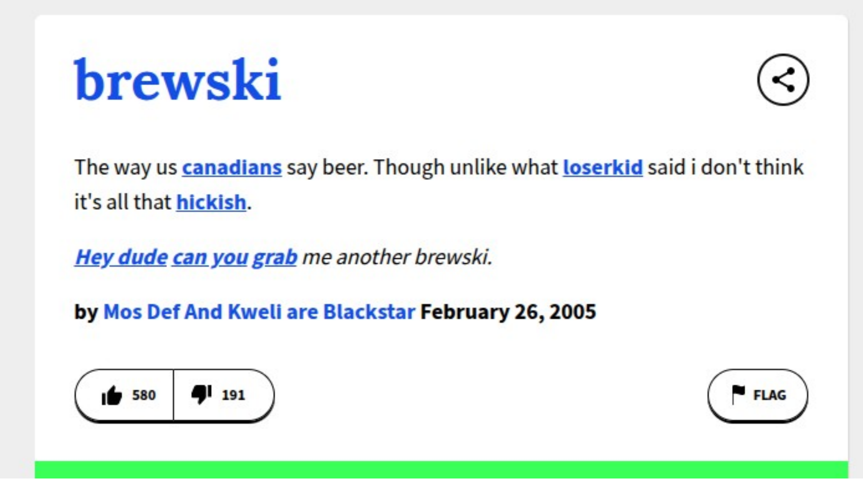


brewski

A slang term developed in the mid to late 70's by an unknown individual in the plains states as a term for a cold driveway beer.

Hey Jeff, come on over for a cold brewski!

by DoodleBerger **May 23, 2005**



brewski

The way us canadians say beer. Though unlike what loserkid said i don't think it's all that hickish.

Hey dude can you grab me another brewski.

by Mos Def And Kweli are Blackstar **February 26, 2005**

Agenda

1. What is BRSKI.
2. What does Build-3 do.
   (a) Parts and Networks
   (b) Demo
3. How is BRSKI evolving?
4. Questions

Goals of BRSKI

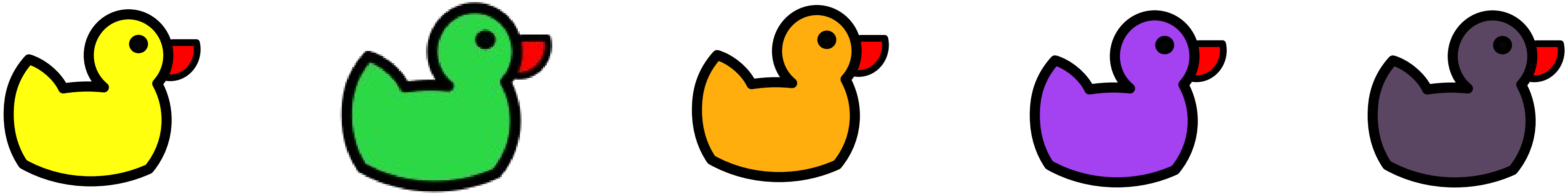
- Allow the network/operator to learn the identity of the new device. (But, EST/RFC7030 did this already)
- Allow the new device to learn the identity of the network/operator. (this part is new)
- Allow the network to provide an LDevID to the new device, allowing it to authenticate to other devices. (this is really the ultimate goal)
- BRSKI is an extension of EST (RFC7030)

Who/why/when

- Developed across IETF 6tisch (industrial IoT), IETF ANIMA (Enterprise/ISP), and NETCONF (device/CPE call home)
- Contributions from Juniper, Cisco, Huawei, and others into RFC8366, and RFC8995. Ongoing efforts from Siemens, Google/Thread, Huawei and others.
- Work started around 2015.
  - RFC8366 published 2018,
  - RFC8995 published May 2021, along with GRASP, ACP and Autonomic Networking.
- ZeroTouch configuration of devices via RFC8572 (SZTP)
- Many resources, presentations, and including more animations, at https://brski.org

Who is who

- New device: the Pledge.
- The icon is the duck, after the 1999 Ross Anderson paper: https://www.cl.cam.ac.uk/archive/rja14/Papers/duckling-ieee-final.pdf
- The duck imprints on whatever looks like it's mother. Hope it's not a wolf. See Konrad Lorenz. Note: BRSKI is not so vulnerable.



https://en.wikipedia.org/wiki/Konrad_Lorenz

Who is who (2)

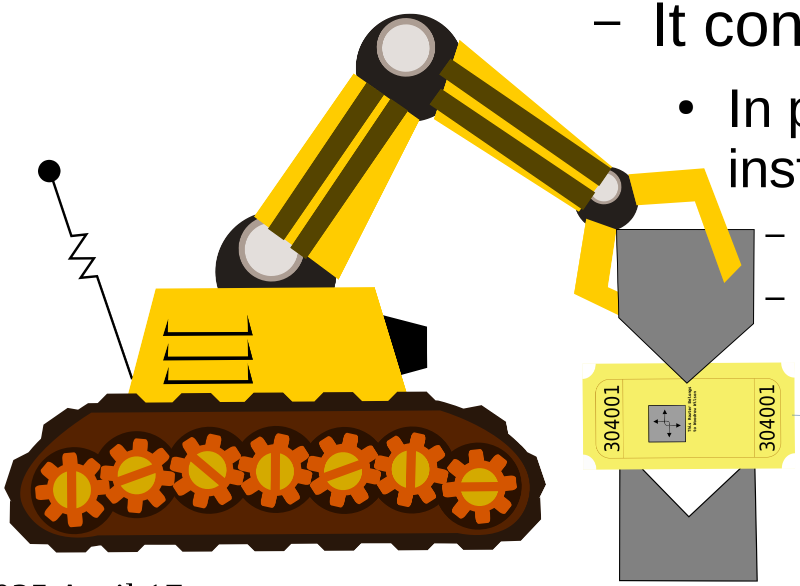
- Network Owner
  - This is the operator of the network.
  - Cryptographically, it’s the (private) Certification Authority (CA), which is owned by the operator.
  - The icon is this passport officer with the wifi hand:

A new variation is the cloud-registrar

(draft-ietf-anima-brski-cloud)

Who is who (3)

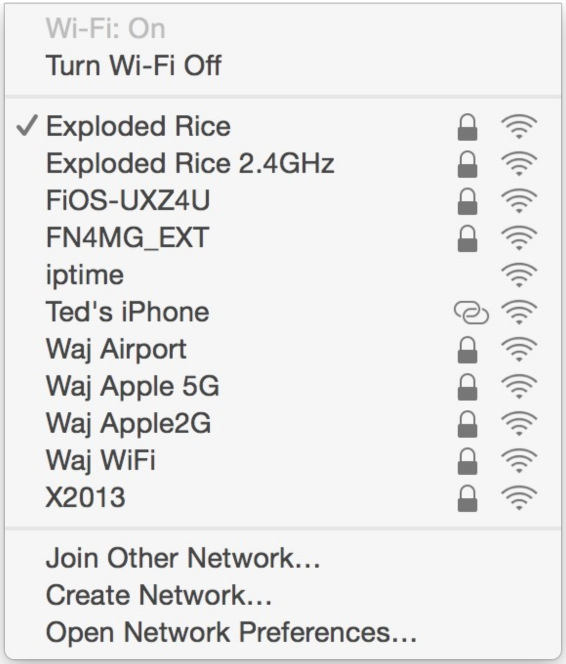
- Device manufacturer, and authorized signer
  - The entity that creates the Pledge, is the vendor.
  - It controls all software that goes into the Pledge
    - In particular, that means it controls all trust anchors installed.
      - MASA anchor, software update anchor
      - Also DNSSEC, any TLS anchors needed to download firmware



MASA Signs voucher

What is ZeroTouch?

- On a laptop, the human touches the device, and picks the right network. This is the authorization step
  - (then there is an authentication dance)
  - don’t join the Wolf Network
- On an IoT device, there is no human, no screen, and thus no way for the device to make an authorization decision.
  - How can device being trusted to make an imprinting decision?

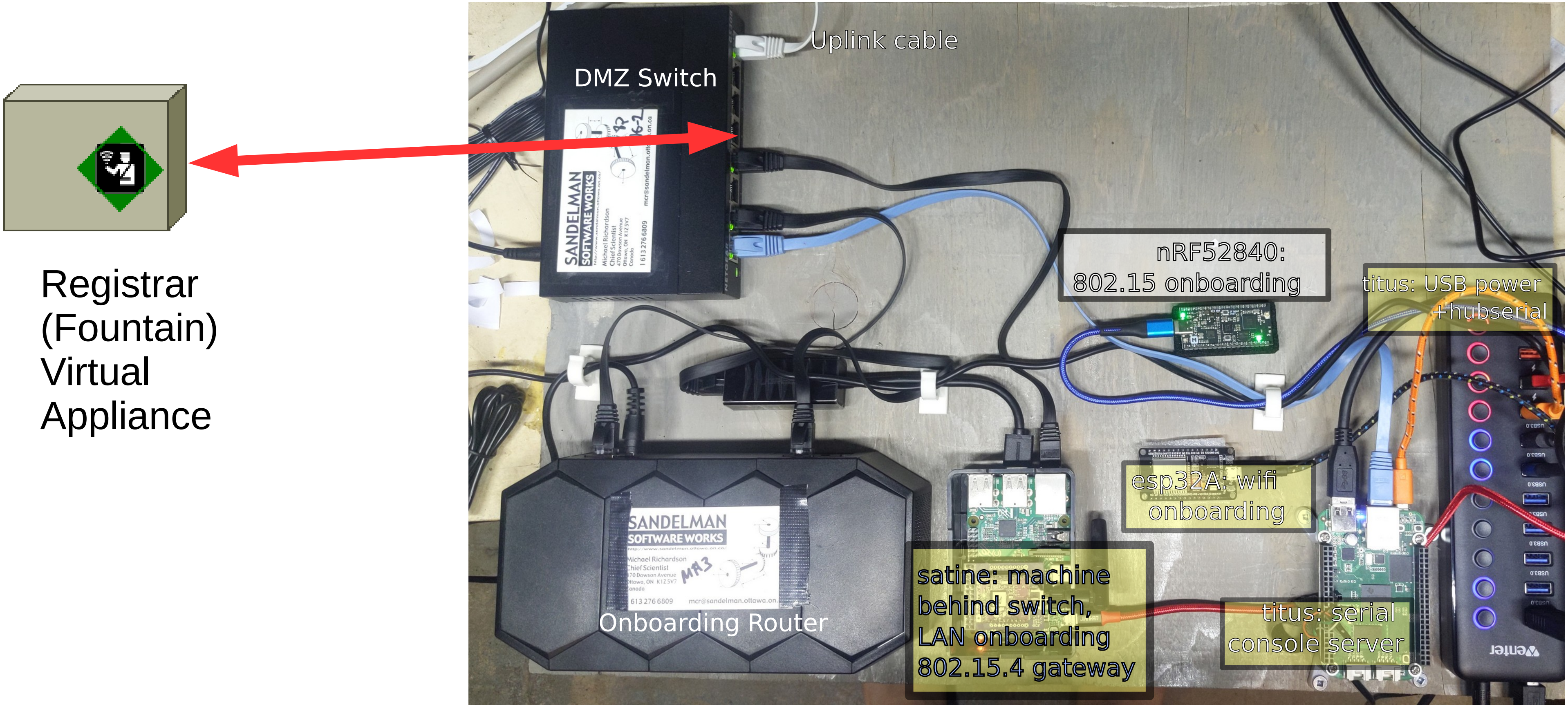


BRSKI’s voucher provides authorization

- RFC8366 voucher authorizes the device (the Pledge), to join the operator’s network
- Variations:
  - CMS signed JSON
  - JWS signed JSON (draft-ietf-anima-jws-voucher)
  - COSE signed CBOR (draft-ietf-anima-constrained-voucher)

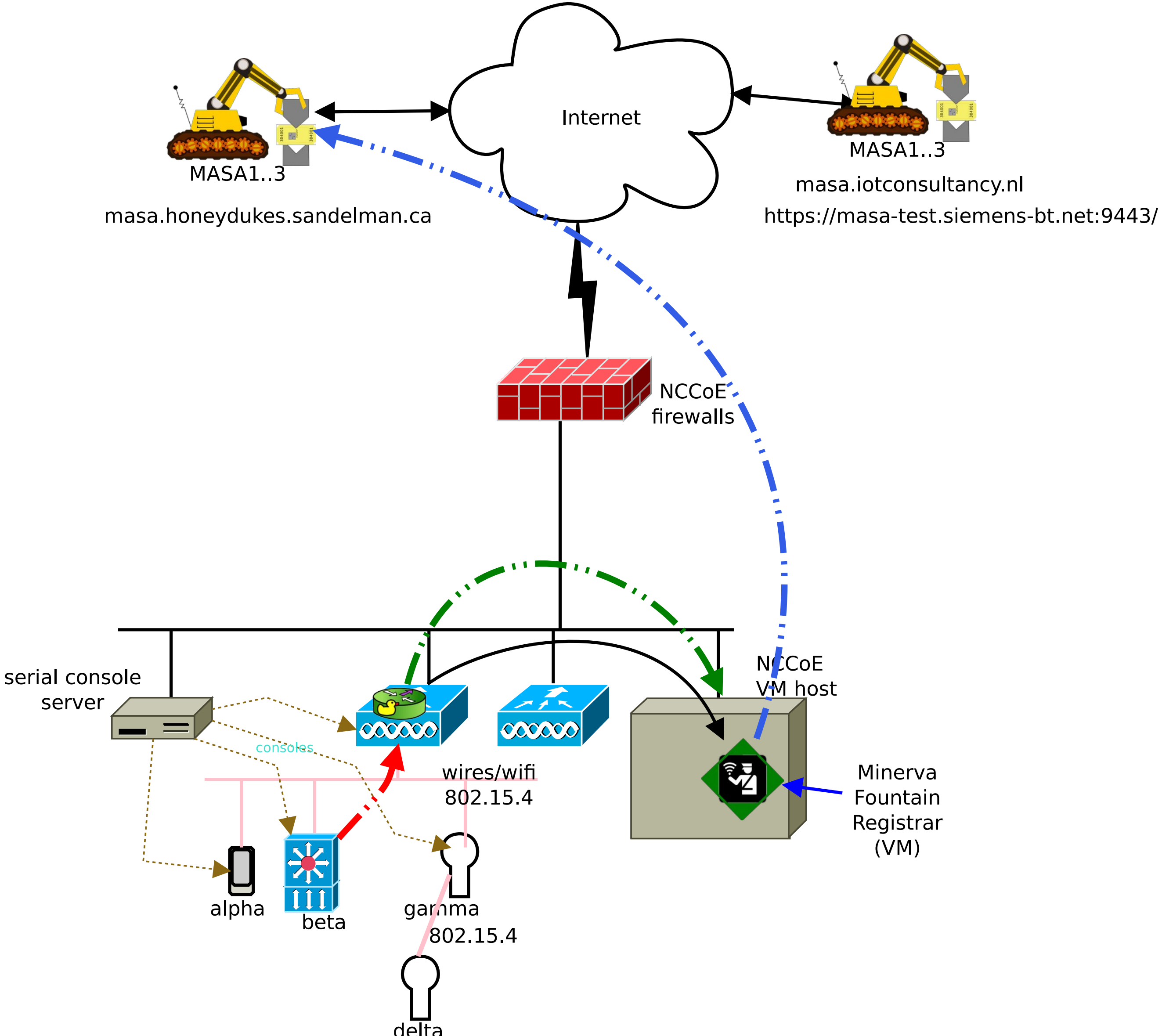
```
{
  "ietf-voucher:voucher": {
    "created-on": "2016-10-07T19:31:42Z",
    "assertion": "logged",
    "serial-number": "JADA123456789",
    "idevid-issuer": "base64encodedvalue==",
    "pinned-domain-cert": "base64encodedvalue==",
    "nonce": "base64encodedvalue=="
```

Build 3 – Wired BRSKI



Registrar
(Fountain)
Virtual
Appliance

Build 3 – Wired BRSKI logical view



2025 April 17

Minerva (.sandelman.ca)

- Minerva is a production ready reference implementation of RFC8995, RFC8366.
- Server components written in Ruby-on-Rails, as both Registrar and MASA are essentially an HTTP (API) service. It scales like other such services.
- Minerva is named for
  - is the Roman goddess of wisdom, justice, law, victory, and the sponsor of arts, **trade**, and strategy. Often also associated with tools.
  - Professor Minerva McGonagall is a fictional character in the Harry Potter series of novels by J. K. Rowling. McGonagall is a professor at Hogwarts School of Witchcraft and Wizardry. She is a well-known ***ANIMA****gus*: a person who can transform into an

both definitions from wikipedia

Minerva (.sandelman.ca)

- Minerva components include
  - highway: the MASA and vendor’s device management system. This integrates with a vendor PKI, or includes one to create the IDevID, track device, and sign vouchers. This is a CLOUD component.
  - fountain: the network operator’s controller. It acts as an RFC7030 Registrar, processes voucher requests, and integrates with the network operator’s PKI (or includes its own).
    - it is an HTTPS or CoAPS server in the southbound direction
    - it is an HTTPS client in the northbound (MASA) direction
  - reach: a demo/validating pledge client library written in ruby.
  - bootstrap: a pledge client written in Rust, aimed at embedded devices
  - connect: a join proxy written in Rust, aimed at router platforms

DEMO

Future evolution of BRSKI

Three Directions

- Different transports
- Different voucher formats
- Different voucher signatures

- Different interaction models
- *MASA-less and/or authorized resale*

- Different Certification Authority interactions
- Cloud Registration

IoT / Constrained

- Different transports

Use CoAP(S) instead of HTTPS
(draft-ietf-anima-constrained-voucher)
RFC9148 (EST-CoAPS)
Use EDHOC instead of HTTPS
(draft-ietf-lake-authz)

- Different voucher formats

Use CBOR for voucher instead of JSON
(draft-ietf-anima-constrained-voucher)
RFC9254, RFC9595

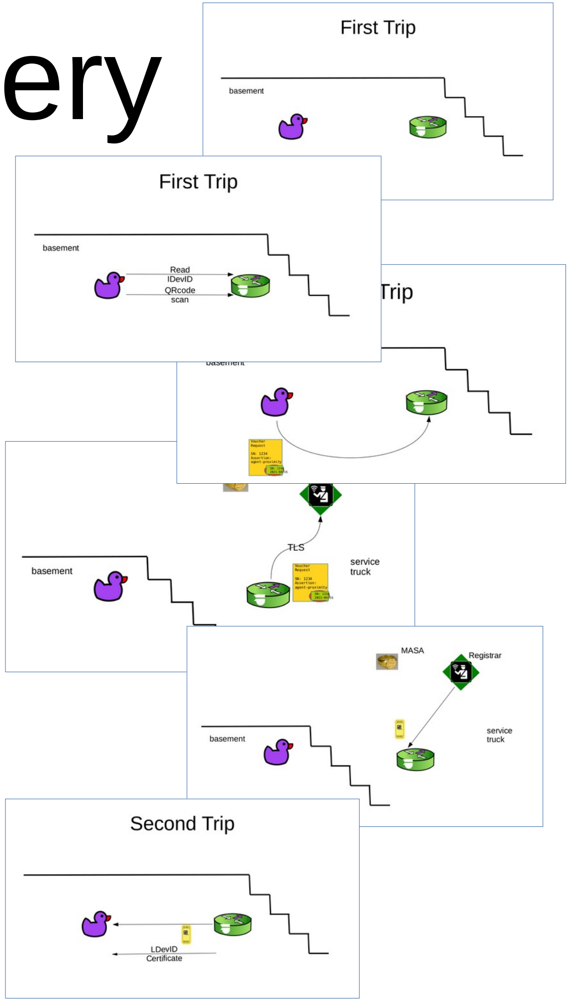
- Different voucher signatures

Use JOSE for signatures instead of CMS
(draft-ietf-anima-jws-voucher)
Use COSE for signatures instead of CMS
(draft-ietf-anima-constrained-voucher)

Offline Voucher Delivery

BRSKI-PRM
Pledge in
Responder **M**ode

- Different interaction models
- (a sort of delay tolerant transfer)



https://datatracker.ietf.org/meeting/111/materials/slides-111-anima-sessa-update-new-on-brski-ae-support-for-asynchronous-enrollment-00

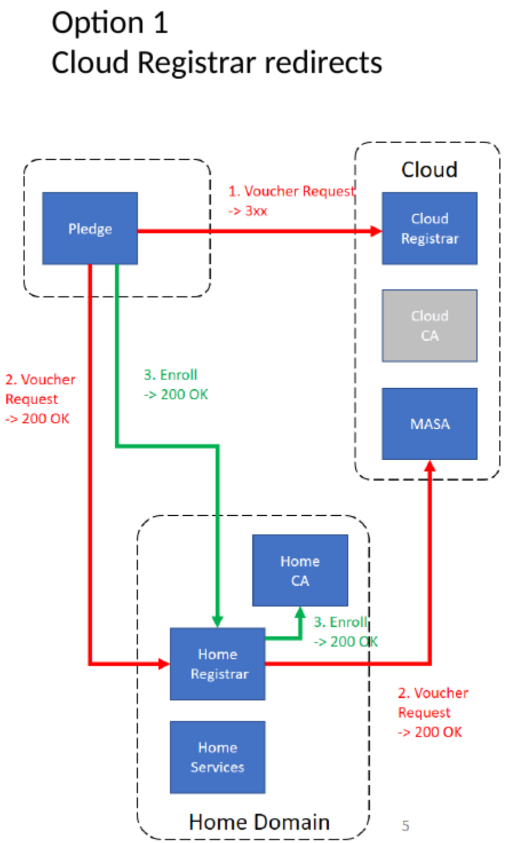
Three Directions

RFC 9733
BRSKI with Alternative Enrollment (BRSKI-AE)

USES CMC rather than EST

- Different Certification Authority interactions

Three Directions



Option 1
Cloud Registrar redirects

- Cloud registrar
- A standard way to call home.
- But still with ownership transfer
- VoIP phones at employee homes, is a significant use case driving this

Questions