

# IETF95 Summary of summary slides of IETF94

Simplying assumption 1: 6tisch like has a PCE/JCE  
draft-pritikin-bootstrapping-keyinfrastructures-00  
→ draft-ietf-anima-bootstrapping-keyinfra-02

For 6tisch WG  
And netconf  
And ANIMA

Term mapping

JCE → ANIMA Registrar

Joint Assistant → ANIMA “Proxy”

Simplying assumption 2: leverage 802.1AR work  
Fundamental to anima-bootstrapping

Challenge 1: how does the network authenticate?

ANIMA bootstrap defines “ownership voucher”

## Contrast ANIMA and 6tisch

- Goal of ANIMA bootstrap is to create Enrollment over Secure Transport (RFC7030)
- ANIMA accomodates HTTPS or DTLS/CoAP + Blockwise. Hard sell to make DTLS Mandatory to Implement.
- Network is not constrained
  - After bootstrap, may be multi-gigabit
- While device is not constrained in aggregate, ANIMA ACP code may run on control plane/line-card CPU: some hardware offload available, but not universal.

VS

- Goal of 6tisch bootstrap is to create secured CoAP/6top transport from JCE/PCE to new node to transport YANG.
- DTLS/CoAP only + 6top, blockwise may be controversial?
- Network is constrained (not challenged)
- Devices are very code and ram constrained.
- Battery power is common (but not universal)
- 

ANIMA

6tisch

## Contrast ANIMA and NETCONF

- Goal of ANIMA bootstrap is to create Enrollment over Secure Transport (RFC7030)
- ANIMA accomodates HTTPS or DTLS/CoAP + Blockwise. Hard sell to make DTLS Mandatory to Implement.
- ANIMA replaces IDevID with LDevID ASAP.
- ANIMA assumes link-local connectivity, device owner is link network operator
- ANIMA tends to be for “infrastructure”

VS

- Goal of NETCONF is to provide signed bootstrap data (YANG) to device.
- Variety of sources: HTTP, HTTPS, DNS, mDNS, DHCP, removable storage...
- NETCONF uses IDevID directly
- NETCONF assumes device owner likely is not link operator, or operator is unsophisticated (home user)
- NETCONF more appliance, and high-volume access device focused, rather than core infrastructure.

ANIMA

Wild generalization!

NETCONF

## Contrast 6tisch and NETCONF!

- Goal of 6tisch bootstrap is to create secured CoAP/6top transport from JCE/PCE to new node to transport **YANG**.
- Devices and networks constrained.
- 6tisch will replace IDevID with LDevID for use with 802.15.9 or other per-link KMP
- No cheap broadcast/multicast, or service discovery
- Device owner is network owner.

VS

- Goal of NETCONF is to provide signed bootstrap data (**YANG**) to device.
- Variety of sources: HTTP, HTTPS, DNS, mDNS, DHCP, removable storage...
- NETCONF uses IDevID directly
- NETCONF assumes device owner likely is not link operator, or operator is unsophisticated (home user)
- NETCONF more appliance, and high-volume access device focused

ANIMA

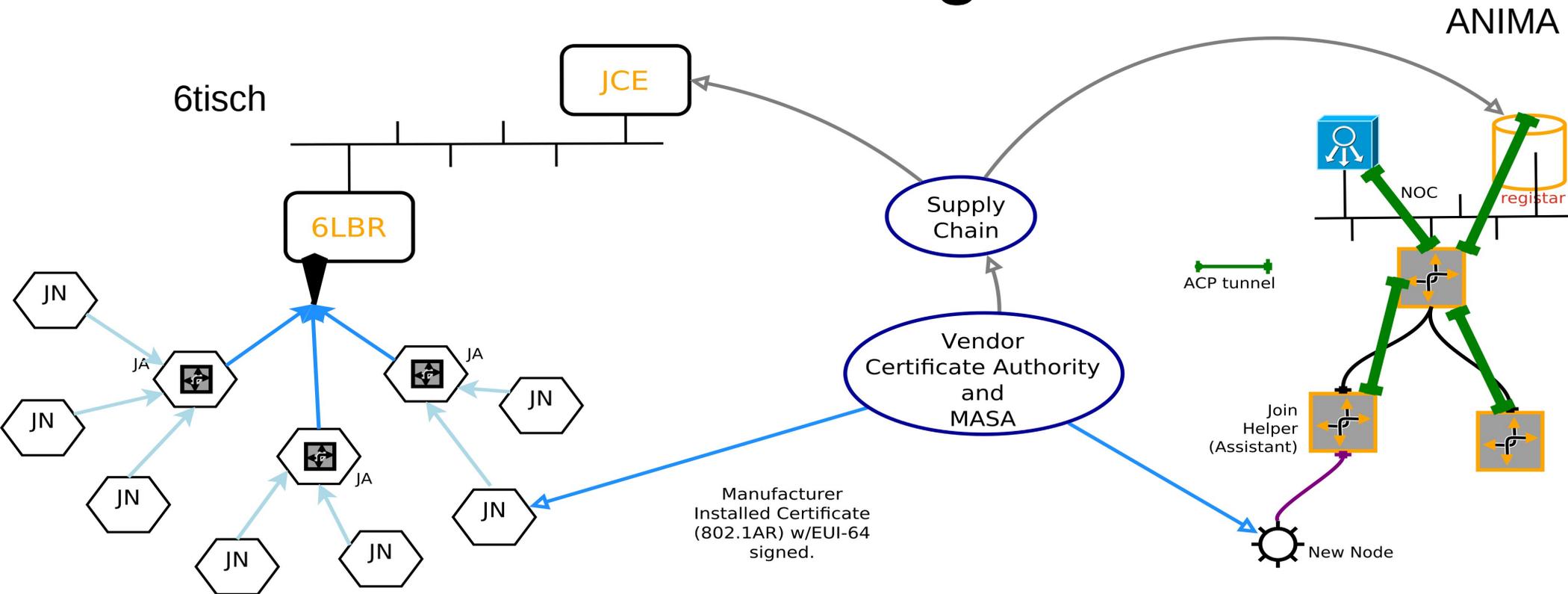
NETCONF

# Join Problem

How to let random uninitialized, “drop shipped”, potentially malicious nodes into your network without destroying the network.

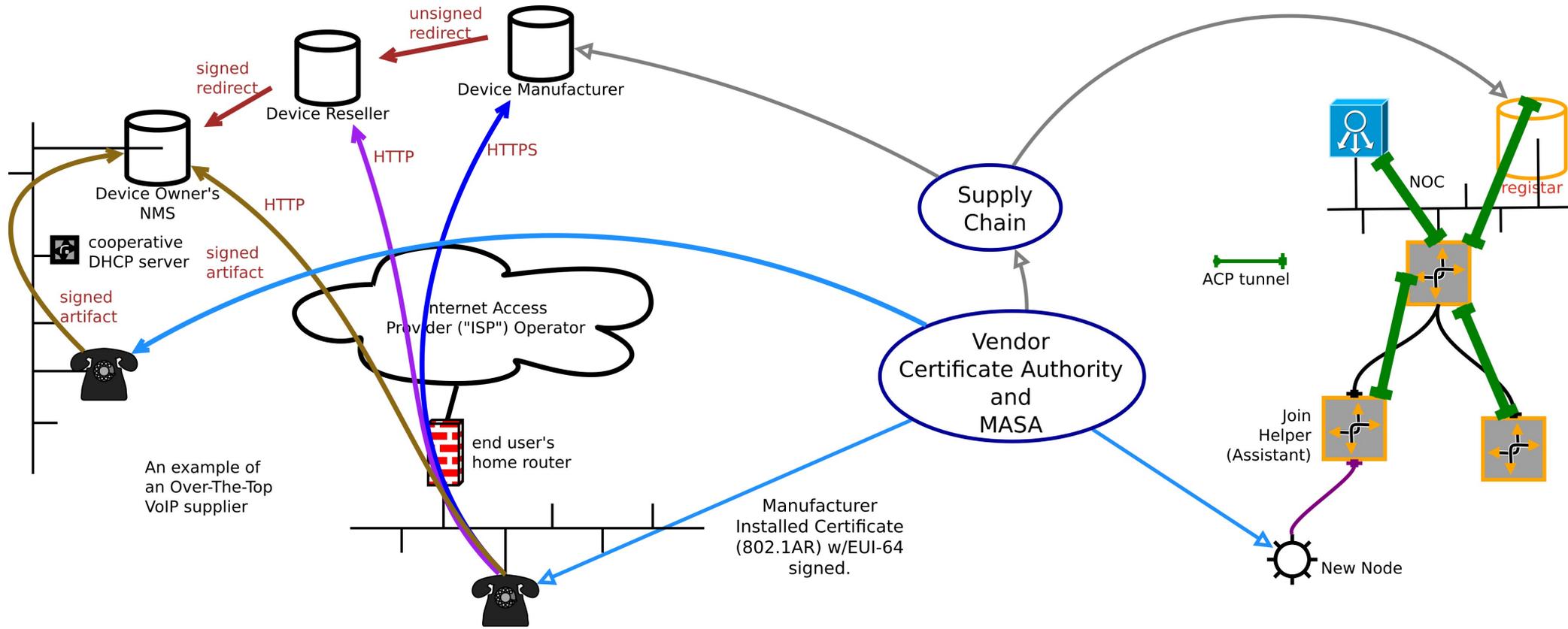
- 802.1x/EAP/PANA has this “solved” for initialized nodes which know which network they want to join; need to be pre-provisioned with certificates.
  - needs EAP-TLS to make this work, which then includes new layers of fragmentation. This code is used once.
  - PANA/1x authenticator function scales with number of nodes attempting to join, is subject to DoS attack, defending against may be too expensive for constrained nodes
  - 1x function for ANIMA **ACP** bootstrap may interfere with 1x function being provided by routers/switches for end-hosts!
- The goal is to provision new nodes with certificates, at which point “traditional” methods may be used to join network.

# Network Diagram



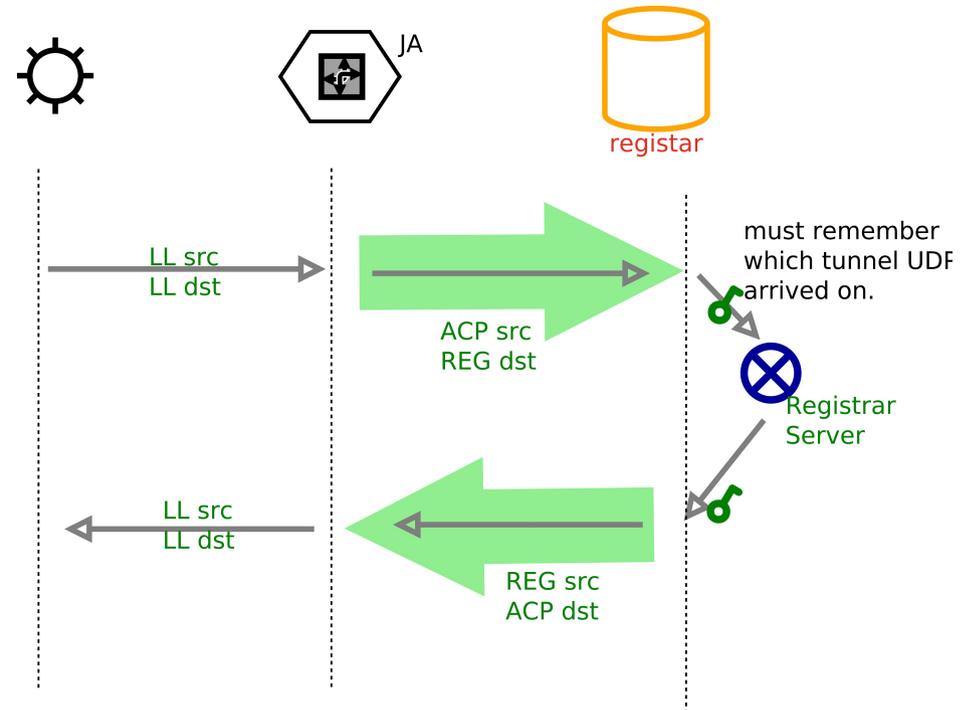
Both 6tisch/LLN, ANIMA and NETCONF share Manufacturer Installed Certificates (“MIC”) [IDevID], and have a supply chain relationship with network operator via which Ownership Vouchers can be communicated.

# Network Diagram: NETCONF



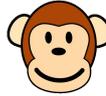
# New Node /Registrar communications

- New Node ↔ Proxy use Link Local addresses.
  - (or HTTPS/TCP)
- Communication is CoAP/DTLS over UDP
  - Proxy is neither trusted, nor needs to be trustworthy
- Proxy ↔ Registrar communication is forwarded (D)TLS traffic; proxy is uninvolved in security.
  - Proxy is neither trusted, nor needs to be trustworthy
- Green Encapsulation arrow can be implemented in different ways



# Proxy/Join Assistant proxy methods

## HTTPS

1. Via circuit proxy (process per connection), or HTTP proxy. 
2. Via NAT66 of link-layer enrollment addresses to ACP ULA address 
3. Stateless IPIP encapsulation of link-local traffic to registrar 

Brian Carpenter  
was visibly ill

## CoAP/DTLS

1. UDP circuit proxy
2. NAT66 of link-layer to ACP ULA address
3. Stateless IPIP encapsulation of link-local traffic to registrar
  - a) Essentially this is routing-dispatch IPIP encapsulation

Least amount of new  
Code for constrained  
Devices, highest  
Resistance to DoS  
Costs some bandwidth

See draft-richardson-anima-state-for-joinrouter-00: Considerations for stateful vs stateless join router in ANIMA bootstrap, for longer discussion

# Funny Icons for other slides

