Making our PowerPoint simpler and more distinctive.
Big Problem

• We know how to manage large numbers of the same device (e.g., ca. 120 – 300 million iPhones)

• We don’t know how to manage larger numbers of *types* of devices
The Network Needs Two Pieces of Information

- What the device is
- How the network should protect it
We have some constraints

- Devices have very few resources to devote to security.
- The larger the footprint on the endpoint, the larger the threat surface (more code = more bugs)
- Strong security will not be possible in some instances.
What is in the option?

- A client-specified URI
- Fully specified in draft-lear-ietf-netmod-mud

Specifically:
https://authority/.well-known/mud/mudvers/...
...model/dev-rev?extras

- This draft normatively depends on the netmod work.
- More information about MUD can be found at
  • draft-lear-mud-framework-00.txt
Expressing Manufacturer Usage Descriptions

Device emits a URI using DHCP, LLDP, or through 802.1ar

Router or firewall queries connected.example.com for policy associated with that URI

Device emits a URI using DHCP, LLDP, or through 802.1ar

Router or firewall queries connected.example.com for policy associated with that URI

**Diagram:**
- Device
- Access Switch
- MUD Controller
- Internet
- MUD File Server

URL: https://example.com/.mud/…
Makes use of YANG-based XML

<?xml version = '1.0' encoding = 'UTF-8'? >
<edit-config
xmlns="urn:ietf:params:xml:ns:netconf:base:1.0"
xmlns:inet="urn:ietf:params:xml:ns:yang:ietf/inet-types"
xmlns:mud="urn:ietf:params:xml:ns:yang:cisco-manpolicy"
<mud:supportInformation>
<mud:lastUpdate>2015-05-12T20:00:50Z</mud:lastUpdate>
<mud:cacheValidity>1440</mud:cacheValidity>
</mud:supportInformation>
<config>
<top>
<acl:access-list>
<acl:access-list-entries>
<acl:access-list-entry>
<acl:rule-name>access-thermostat-controller</acl:rule-name>
<acl:matches>
/inet:hostname>controller.example.com</inet:hostname>
</acl:matches>
<acl:actions>
<acl:permit/>
</acl:actions>
</acl:access-list-entry>
<acl:access-list-entry>
<acl:rule-name>let-me-talk-to-other-thermostats</acl:rule-name>
<acl:matches>
<mud:sameManufacturer/>
</acl:matches>
<acl:actions>
<acl:permit/>
</acl:actions>
</acl:access-list-entry>
</acl:access-list-entries>
</acl:access-list>
</top>
</config>
</edit-config>
Expressing Manufacturer Usage Descriptions

https://example.com/.mud/…

More precise config is instantiated

Site returns abstracted XML (based on YANG) to device or firewall

Device
Access Switch
MUD Controller
MUD File Server

Allow access to just controller.connected.example.com

Internet

4 April 2016
Why a DHCP option?

• This is the 2nd choice to deliver the MUD URI
• IEEE 802.1AR has stronger security properties
• DHCP is still useful - assertion is from the device for its protection.
• Very easy to implement and deploy for any system already implementing DHCP
Who does what?

- Client sends URI
- Gateway passes URI
- Server processes URI or passes it to a controller
- Server acknowledges in its response
- Controller/server retrieves descriptions and applies what configuration it will
- Controller cleans up on release, carrier drop, or session termination
What is needed...

- Would like more eyes on the draft and the concept
- Can this be adopted as a WG draft?