draft-lear-ietf-netmod-mud

Eliot Lear

18 July 2016

Brief reminder of what MUD is

- Manufacturer Usage Descriptions
- Goal:
  - Reduce threat surface of device by getting suggestions from the manufacturer to the operational network.
- Basic Mechanisms:
  - A URL from the device (via DHCP, 802.1X/AR, LLDP)
  - A yang model or two

Big Changes From Last Time

- Now using JSON
- Now signing
- Simplified URL
- Documents consolidated
  - Now contains DHCP option and non-critical X.509 extension

JSON Example

```
{
   "ietf-mud:support-information": {
     "last-update": "2016-05-18T20:00:50Z",
     "cache-validity": 1440
     },
   "ietf-access-control-list:access-lists":  {
     "acl": [ {
     "acl-name": "inbound-stuff",
     "acl-type" : "ipv4-acl",
     "ietf-mud:direction" : "to-device",
     "access-list-entries": {
       "ace": [
         {
           "rule-name": "access-cloud",
           "matches": {
             "ietf-acl-dnsname:source-hostname":
                "lighting-system.example.com",
             "protocol" : 8,
             "source-port-range" : {
               "lower-port" : 443,
               "upper-port" : 443
             },
           "actions" : {
             "permit" : [null]
             }
         }
        ]
       }
...
```

A recent example: Preventing PLC-Blaster

```
{
  "ietf-mud:supportInformation": {
    "lastUpdate":"2016-05-05T20:00:50Z",
    "cacheValidity":1440
  },
  "ietf-acl:access-list":{
    "ietf-mud:direction":"inbound",
    "access-list-entries":{
      "ace":[
        {
          "rule-name":"only-plc-controller",
          "matches":{
            "ietf-mud:controller":"https://example.com/.well-known/mud/v1/s7-1200",
            "protocol":"tcp",
            "destination-port-range":{
              "lower-port":102,
              "upper-port":102
            }
          },
          "actions":{
            "packet-handling":"permit"
                }
        }
      ]
    }
  }
}
```

Big Questions

- Did we choose the right approach to signing?
  - Could use JWS or PKCS#7 – chose PKCS#7
  - Using detached signatures
  - Canonical form is the actual file

Big Questions (2)

- How should we do extensibility?
  - Assumptions:
    - No capabilities exchange or negotiation (this isn't NETCONF - it's HTTP)
    - Three parties – MUD file server, MUD controller, and device
    - Implementing "some" functionality in a MUD file might be risky
  - Approaches:
    - Create a manifest file that groups capabilities to specific MUD files.
    - Use strict versioning
    - Use critical/non-critical constraints and when we need new critical constraints, bump the version

Other ideas?

- Reputation of the Manufacturer
  - Should we be looking at attestation mechanisms?

Next

- Will incorporate draft-lear-ietf-netmod-acl-dnsname

Request

- Can we adopt this work in opsawg?