CAPPORT Protocol

Mark Donnelly (mark@painless-security.com)

Margaret Cullen

Painless Security, LLC

Protocol walkthrough

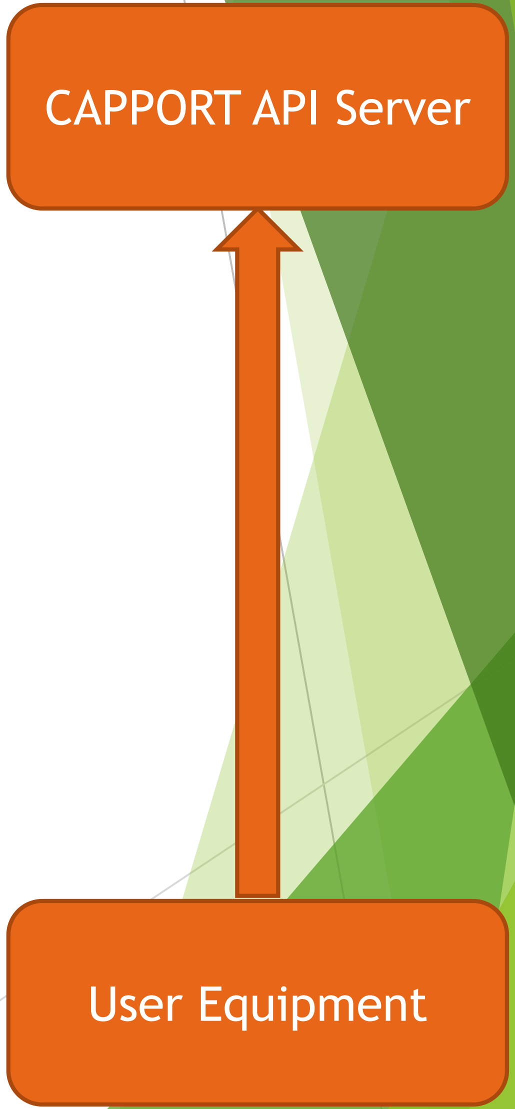
URL Acquisition

- CAPPORT API URL is obtained from DHCP or IPv6 RA (RFC ####)
- If the User Equipment does not support the CAPPORT protocol, it issues a GET request for the CAPPORT API URL in a browser, accepting text/html
- If the User Equipment does support the CAPPORT protocol, it issues a POST request to the CAPPORT API URL, accepting application/json.

CAPPORT Initial Request

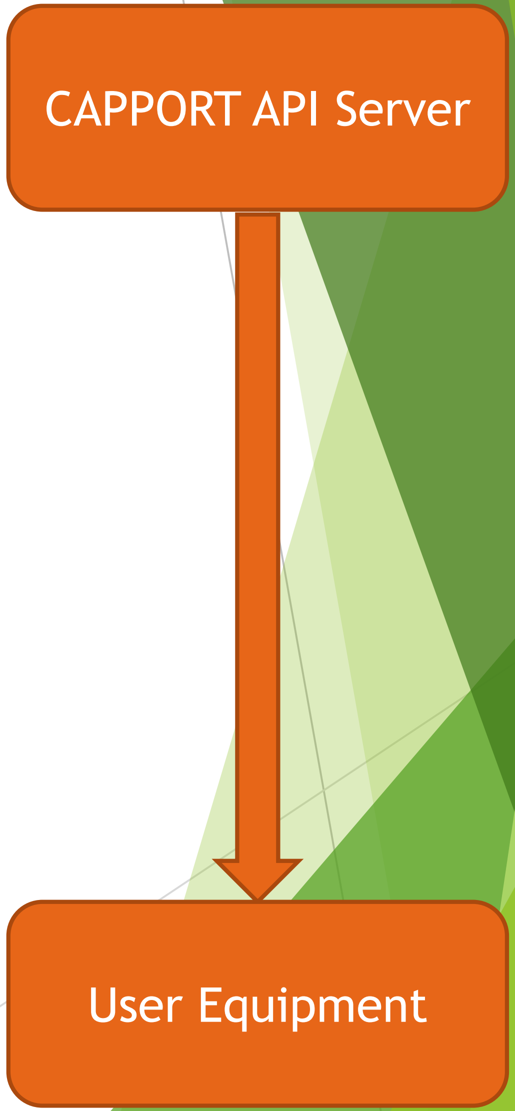
- User Equipment initiates a conversation with the CAPPORT API Server

-
  ```
  {
    "networks" : {},
    "session-token" : ""
  }
  ```



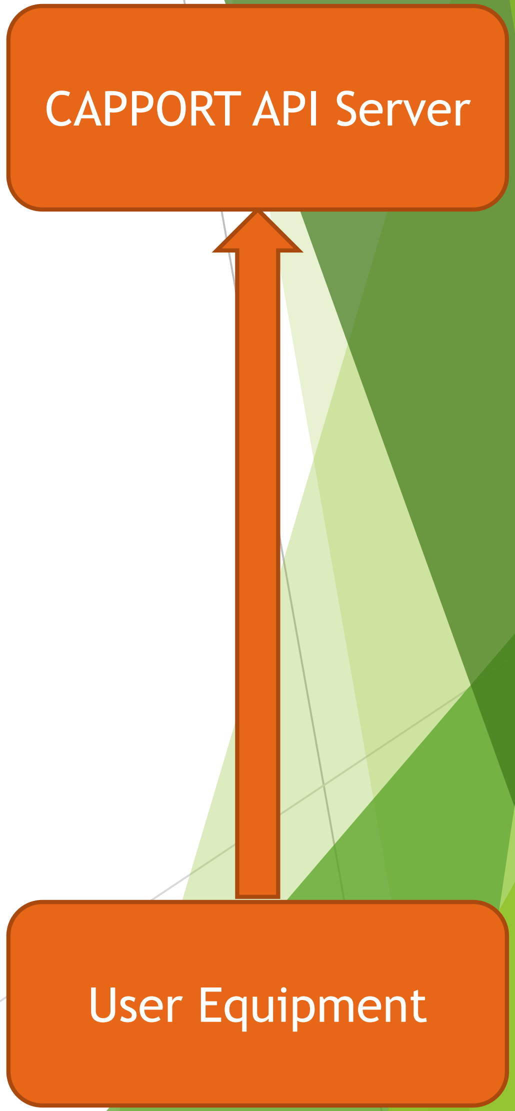
CAPPORT API Server Requirements

- CAPPORT API Server responds with available networks, network requirements, and a session token
- The CAPPORT API Server may control access to multiple networks. Each one gets its own key.
    - The “DEFAULT” network implies that this network should be chosen if the user hasn’t specified otherwise, and should ideally have access to the internet
-
```
{"networks" : {
   "DEFAULT": {
     "conditions" : [ {
       "id" : "...",
       "type" : "t&c",
       "requirement_details" : {
         "text" : “I will do whatever you say"
       }  } ],
     "state" : {"permitted" : false}} },
   "session-token" : "...“
 }
```



CAPPORT User Equipment Requirements Satisfaction

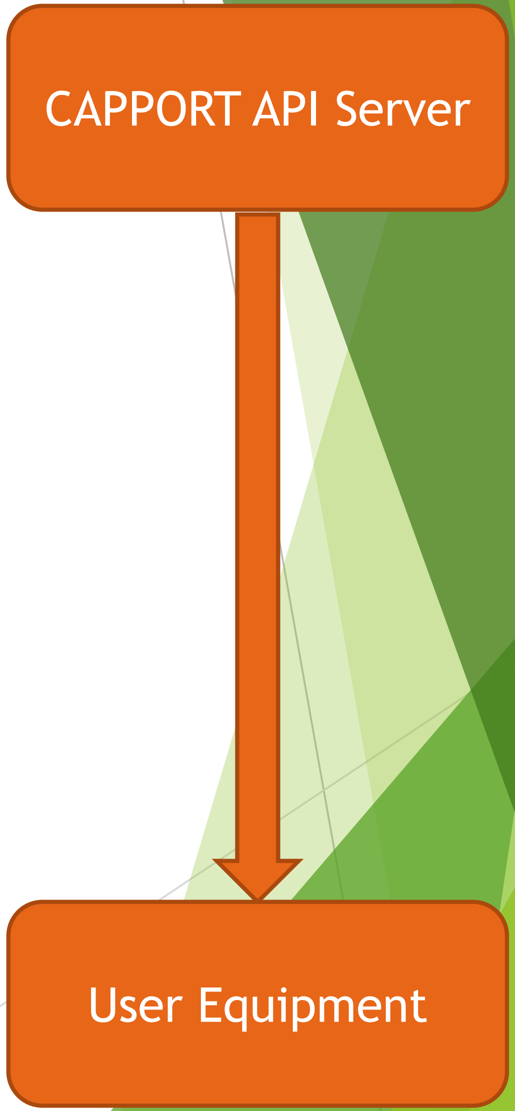
- User Equipment attempts to satisfy the requirements of the CAPPORT API Server
-
```
{
  "networks" : {
    "DEFAULT" : {
      "conditions" : [
        "id" : "...",
        "satisfaction_details" : {
          "text" : "1e173b7bb0d114bb38438c15b9eb9736"
        }
      ]
    }
  },
  "session-token" : "..."
}
```
- The CAPPORT API Server Requirements step and this step can be repeated
  - If the User Equipment wants to satisfy some but not all of the network access conditions at this time
  - If the CAPPORT API Server wants to generate new conditions based on how these requirements are satisfied



CAPPORT API Server grants access

- CAPPORT API Server has no more unmet conditions for network access
- The state is now permitted, optionally with expiration time, bytes remaining, or both

-
  ```
  {"networks" : {
    "DEFAULT": {
      "state" : {
        "permitted" : true,
        “expires” : “2036-01-01T00:00:00Z”,
        “bytes_remaining” : 987654321
      },
      "session-token" : "...”
  }
  ```



Conditions

Overview

- Each network will have an array of conditions for network access
- Each condition has an ID (UUID), a type, and details of the condition requirements
- The draft currently defines a type for Passcodes, and Terms & Conditions
  - Need more!
- Probably should start a registry for additional condition types
- If the User Equipment cannot satisfy the conditions on its own, it should open up a web browser to the CAPPORT API Server URL
- Questions
  - Do we need to return any errors from the conditions failing to be met?
  - Do we need a way to express complex logic, like "at least 2 of the following five conditions must be met"?

Terms & Conditions

- For agreeing to usage terms and conditons
- Type: “t&c”
- “requirement_details”
  - Specify plaintext formatted T&C, HTML formatted T&C, or both
- “satisfaction_details”
  - MD5 sum the plaintext, the HTML, or both
  - Question: Is MD5 the best way to go here?
  - Question: Is supplying both useful, or just a potential problem for what to do when one MD5 sum matches but the other does not?

Passcode

- For users proving possession of a passcode (such as getting a WiFi password from a hotel desk)
- Type: “passcode”
- “requirement_details”: an empty JSON hash
- “satisfaction_details”
  - “passcode”: “the passcode”

Sessions

Sessions

- How do we want to associate a particular User Equipment with a CAPPORT API Server session?
  - The draft document was written assuming that the CAPPORT API Server would use the source address for the session
  - Dave Dolson suggests an explicit request and response for creating a session, and another for destroying a session
  - The CAPPORT API URL could also be different for every User Equipment
- Draft would benefit from examples