CAPPORT Architecture
draft-larose-capport-architecture-00

Authors: Kyle Larose, Dave Dolson
Presenter: Kyle Larose

Introduction

- By nature, interfere with normal traffic flow
- Typically modify plain text HTTP
- Interruption is not standardized
- Lead to bad behaviour
- See [I-D.nottingham-capport-problem]

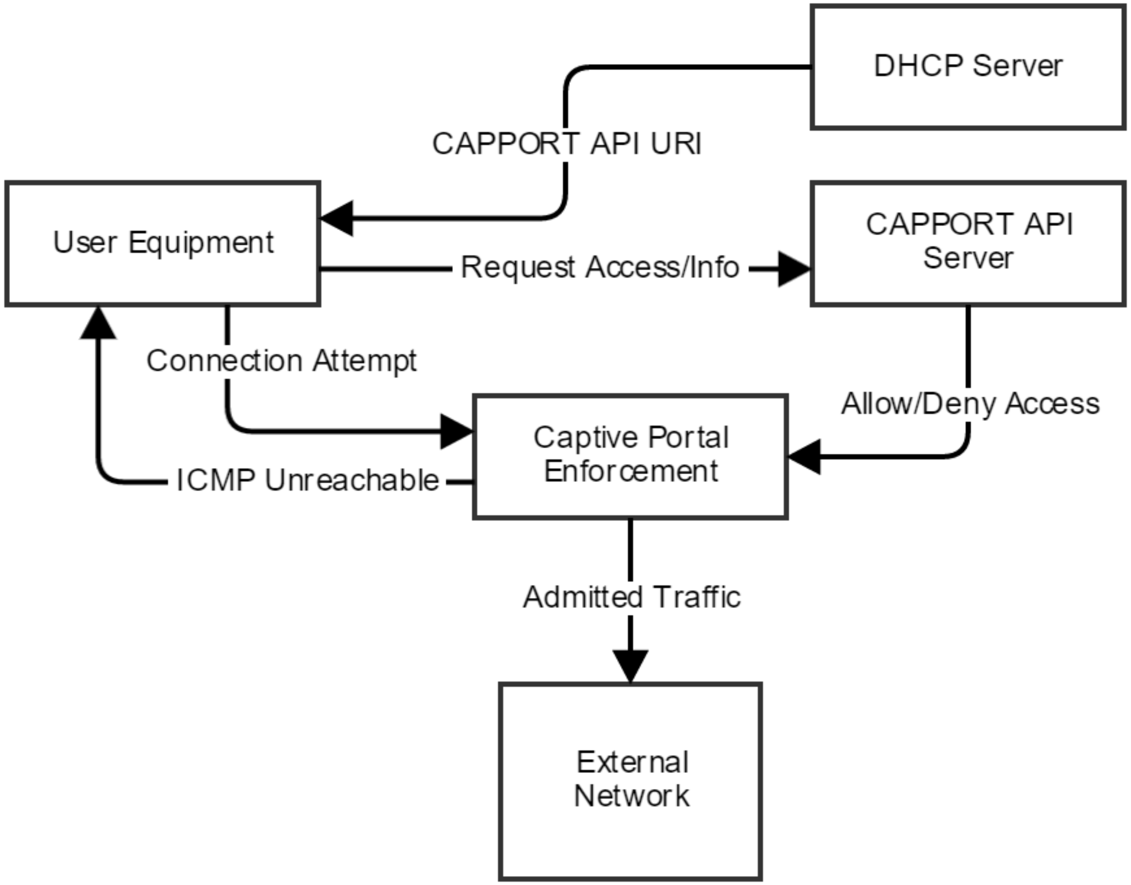
Introduction (continued)

- Working Group Charter:
  - Provide URI for interacting with captive portal
  - Allow user equipment to:
    - Detect captive portal
    - Learn about captive portal
    - Interact with captive portal
    - Do so possibly without human interaction
- Architecture condenses verbal/email communication to achieve charter

Architecture Goals

- Standard way to implement captive portals
- Standard way to interact with captive portals
- Minimize unexpected interactions with devices
- Allow non interactive devices access

Architecture



User Equipment

- DHCP Client
- CAPPORT API Client
- Maybe has a human
- Wants to communicate outside the captive network
- Understands ICMP Unreachable
- No interest in specifying user interface

DHCP Server

- Implements [RFC7710]
- Provides URI for CAPPORT AP via:
  - The Captive-Portal DHCPv(4|6) option, or
  - IPv6 RA option

CAPPORT API Server

- REST API. E.g. [draft-donnelly-capport-detection]
- Coupled with the Captive Portal Enforcement device to inform it of User Equipment
- Various authentication methods (e.g. a menu of authentication options)
- Should provide a non-interactive authentication method

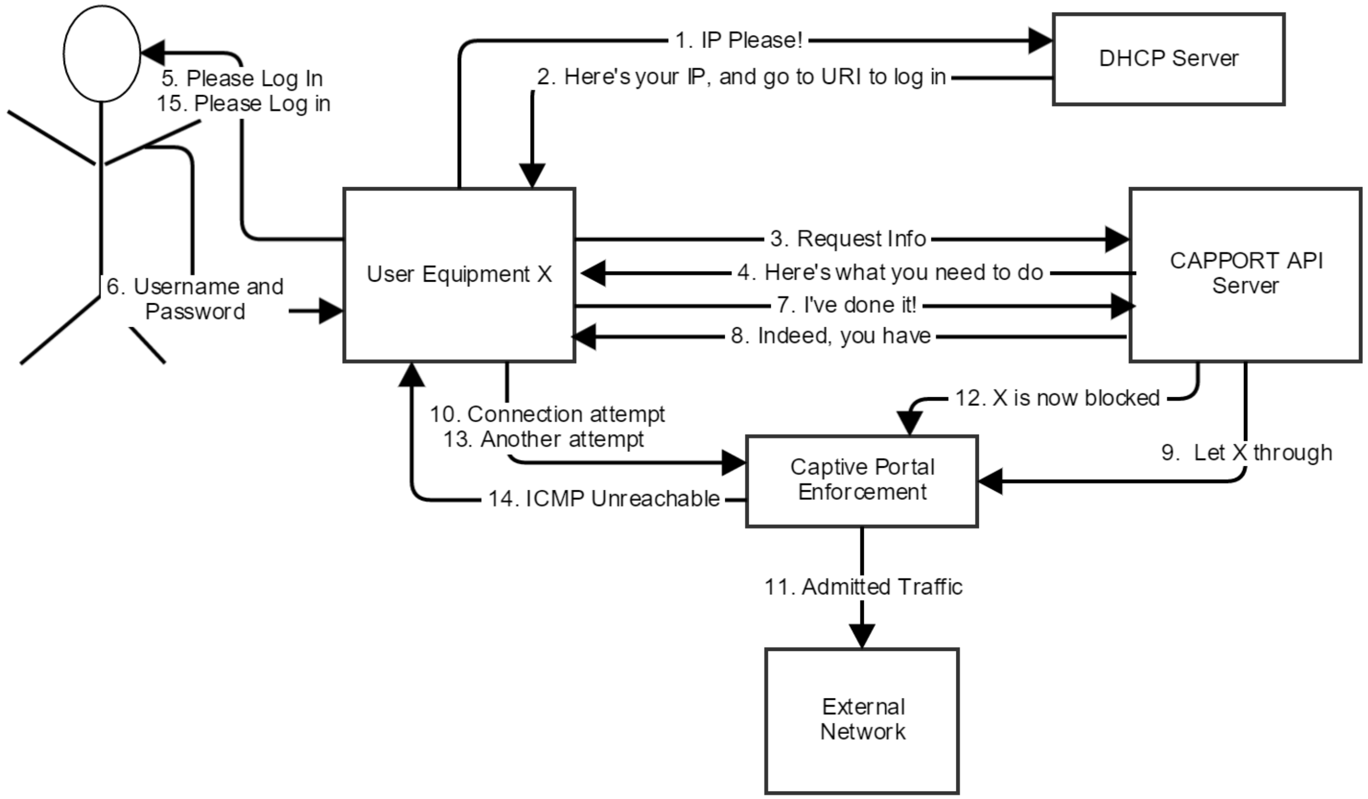
Captive Portal Enforcement

- Decides whether a packet is allowed through to an external network
- Example: a wifi hotspot or home router
- If blocking traffic, sends an ICMP unreachable message to the blocked user equipment
- May allow access to a walled garden

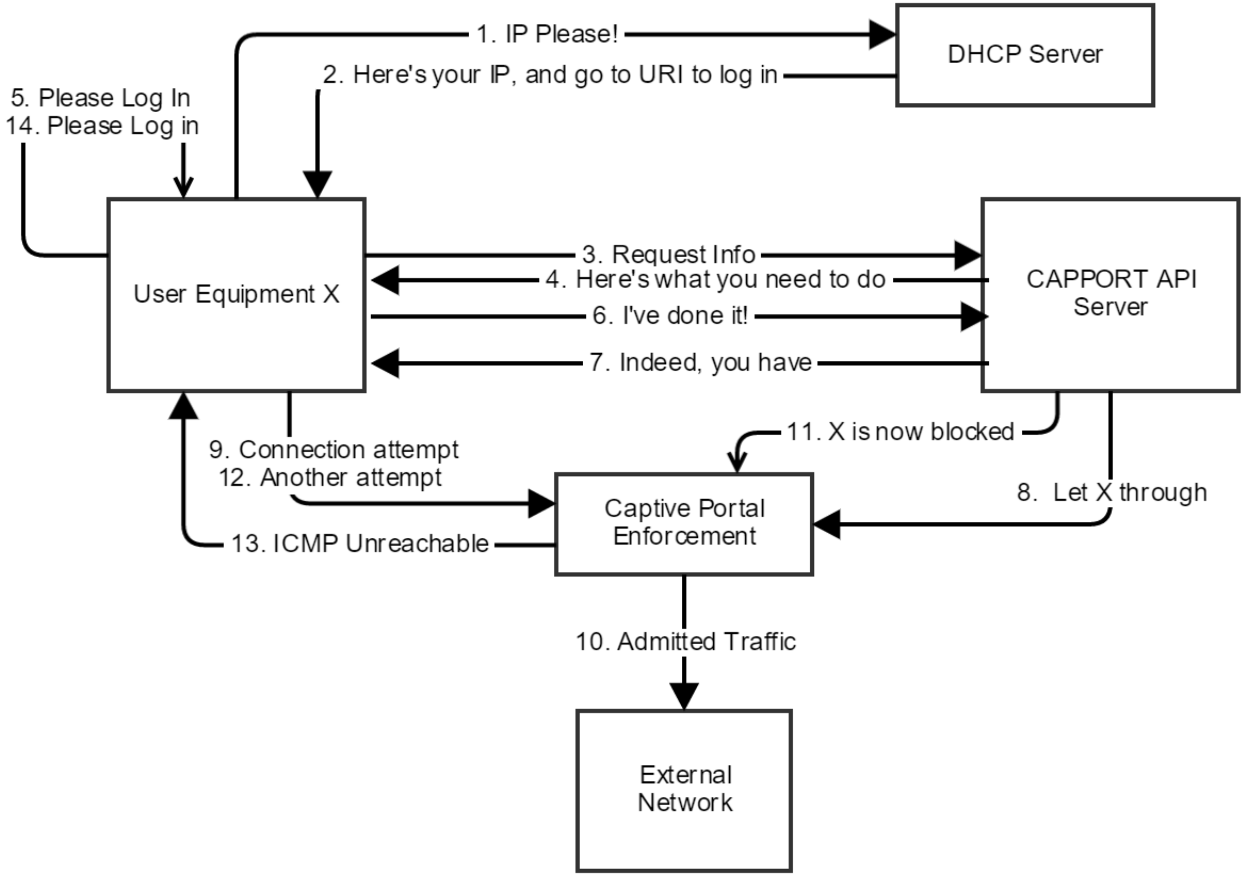
ICMP Unreachable Message

- ICMP message : a captive portal has blocked the connection attempt.
  - E.g. [I-D.wkumari-capport-icmp-unreach]
- Intended to:
  - Allow user equipment to gracefully react to connection issues
  - Allow automatic reauthentication, or a GUI “pop-up” indicating that the user must take action
- Uses a token for authentication
  - Note that I-D.wkumari-capport-icmp-unreach does not currently include said token

Workflow – Web Browser



Workflow – IoT Device



Security Concerns

- Is the token approach sufficient for ICMP validation?
- Is server authentication of API required?

Security Benefits

- No longer man in the middle
- Portal restricted to what DHCP/RA said

Unanswered Questions

- Do we recommend a transition strategy into using this architecture?
- Where do the various components live?
  - Does ICMP exist on the same device as enforcement?
  - How many L3 hops away can things be?
- Does the document need to explain how to configure the system (e.g. allow access to CAPPORT API in walled garden)?
- Is describing how a non-interactive device actually gets authentication credentials in scope?

Next Steps for the Draft

- Does the WG want to keep working on this?

Questions?

References

- [I-D.nottingham-capport-problem] https://tools.ietf.org/html/draft-nottingham-capport-problem-01
- [RFC 7710] http://www.rfc-editor.org/info/rfc7710
- [draft-donnelly-capport-detection] https://tools.ietf.org/html/draft-donnelly-capport-detection-01
- [draft-wkumari-capport-icmp-unreach] https://tools.ietf.org/html/draft-wkumari-capport-icmp-unreach-01