Internet Engineering Task Force K. Larose
Internet-Draft D. Dolson
Intended status: Informational Sandvine
Expires: September 10, 2017 March 9, 2017

CAPPORT Architecture
draft-larose-capport-architecture-00

Abstract

This document aims to document concensus on the CAPPORT architecture. DHCP, ICMP, and an HTTP API are used to provide the solution.

Status of This Memo

This Internet-Draft is submitted in full conformance with the provisions of BCP 78 and BCP 79.

Internet-Drafts are working documents of the Internet Engineering Task Force (IETF). Note that other groups may also distribute working documents as Internet-Drafts. The list of current Internet-Drafts is at http://datatracker.ietf.org/drafts/current/.

Internet-Drafts are draft documents valid for a maximum of six months and may be updated, replaced, or obsoleted by other documents at any time. It is inappropriate to use Internet-Drafts as reference material or to cite them other than as "work in progress."

This Internet-Draft will expire on September 10, 2017.

Copyright Notice

Table of Contents

1. Introduction

Problems with captive portals have been described in [I-D.nottingham-capport-problem].

This document standardizes an architecture for implementing captive portals that provides tools for addressing most of those problems.

The architecture also attempts to enable IoT devices, in particular devices without user interfaces, to navigate a captive portal.

The architecture uses the following mechanisms:

- DHCP/DHCP6 providing end-user devices with a URI in the Captive-Portal Router Advertisement option [RFC7710]. This URI is an API that the end-user devices access for information about what is required to escape captivity.

- Notifying end-user devices of captivity with ICMP/ICMP6 "unreachable" messages. This notification can work with any Internet protocol, not just clear-text HTTP. This notification does not carry the portal URI, rather triggers the DHCP-

provisioned portal to be accessed. This notification carries a "reason" that allows the devices to receive customized work-flows at the portal.

- Receipt of the ICMP/ICMP6 messages inform an end-user device that it is captive. This permits the device to take immediate action to satisfy the portal (according to its configuration/policy). The architecture recommends the device to query the DHCP-provisioned CAPPORT URI with the specified "reason". This API returns a status and a menu for navigating the captive portal. Typically one of the menu items is a web page suitable for browsing.

The architecture attempts to provide privacy, authentication, and safety mechanisms to the extent possible.

1.1. Requirements Language

The key words "MUST", "MUST NOT", "REQUIRED", "SHALL", "SHALL NOT", "SHOULD", "SHOULD NOT", "RECOMMENDED", "MAY", and "OPTIONAL" in this document are to be interpreted as described in RFC 2119 [RFC2119].

1.2. Terminology

Captive Network: A network for which communication outside of it is subject to a captive portal

Captive Portal Enforcement: The device which enforces the captive portal in the captive netowrk

Captive Portal User Equipment: Also known as User Equipment. A device which wants to communicate outside the captive network

2. Components

2.1. User Equipment

The User Equipment is the device that a user desires to communicate with a network. The User Equipment communication is typically restricted by the Captive Portal Enforcement, described in Section 2.4, until site-specific requirements have been met.

- May be interactive or non-iteractive

- May have different mechanisms for notifying the user of the captive portal

- Needs to recognize the ICMP unreachable message, and to invoke its captive portal handling in response to it.

- Needs to cache the URI for the captive portal API from the DHCP lease.

- May cache credentials to automatically respond to captive portal notifications

- Interactive User Equipment typically ask their users how to proceed through interacting with the captive portal. Interactions may be as simple as accepting a terms of agreement, or as complicated as filling out some forms.

- An example interactive User Equipment is a smart phone.

- Non interactive User Equipment may be provisioned with credentials out of band (e.g., via USB programming) in order to automatically gain access.

- An example non interactive User Equipment is an IoT device such as a smart thermostat.

- May need to distinguish between types of User Equipment here.

2.2. DHCP Server

A standard for providing a portal URI is described in [RFC7710]. The CAPPORT architecture expects this URI to access the API described in Section 2.3.

Although it is not clear from RFC7710 what protocol should be executed at the specified URI, it may have been assumed to be an HTML page, and hence there may be User Equipment assuming a browser should open this URI. For backwards compatibility, it might be necessary for the server to check Agent-Id when serving the URI.

2.3. Captive Portal API Server

The User Equipment performs GET at the DHCP-specified URI. The API is implemented at the CAPPORT API Server. The response is a JSON document. The following information should be available in the response document, allowing User Equipment devices to choose the next step:

- Quota information (remaining time/bytes/etc.)

- Whether the device is allowed through captive portal or blocked.

- Method of providing credentials to gain access.
- Describe the required credentials to gain access.
- URL of a web page for devices with browsers and humans.
- A token used to verify later ICMP messages are valid.

The CAPPORT API is intended to provide information and a menu of choices to support options for interactive or non-interactive User Equipment.

The CAPPORT API should support TLS for privacy. [Does this API need to be secure, or do we place security at the interfaces it points to?]

2.4. Captive Portal Enforcement

The Captive Portal Enforcement component restricts network access to User Equipment according to site-specific policy. Typically User Equipment is denied network access until it has performed some action.

The Captive Portal Enforcement component:

- Allows traffic through for allowed User Equipment.
- Blocks traffic and sends ICMP notifications for disallowed User Equipment.
- Permits disallowed User Equipment to access necessary APIs and web pages to fulfill requirements of exiting captivity.
- May modify responses to canary URLs, or perform other methods of notification.
- Updates policy per User Equipment in response to operations from the Captive Portal API.

2.5. ICMP/ICMP6

A mechanism to trigger captive portal work-flows in the User Equipment is proposed earlier in [I-D.wkumari-capport-icmp-unreach]. Additionally, the Unreachable message carries a token to prove it is a valid notification.

The Captive Portal Enforcement function is required to send such ICMP messages when disallowed User Equipment attempts to send to the network.

The ICMP messages MUST NOT be sent to the Internet devices. The indications are only sent to the User Equipment.

The User Equipment MUST verify that the token matches the token received earlier via the CAPPORT API. If tokens do not match, the ICMP message MUST be discarded with no further impact. (It MAY be counted.)

The User Equipment does not necessarily deliver the impact of the ICMP message to the application that triggered it. The User Equipment may be able to satisfy the Captive Portal requirements quickly enough that existing transport connections are not impacted.

2.6. Component Diagram

The following diagram shows the communication between each component.

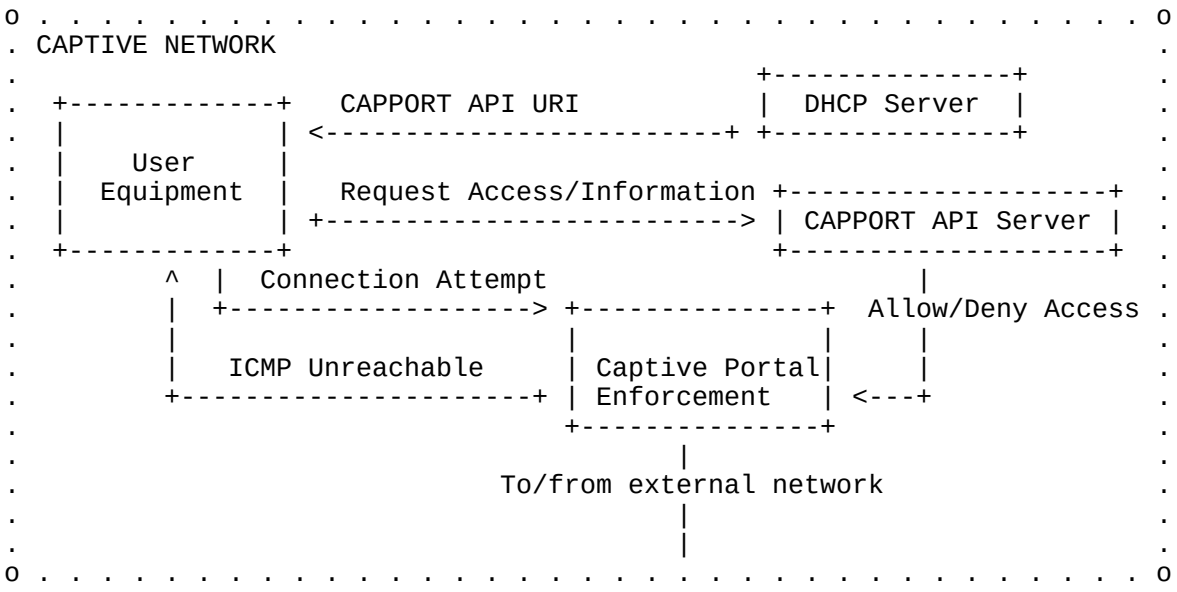

Figure 1: Captive Portal Architecture Component Diagram

In the diagram:

- The User Equipment communicates with the DHCP Server to get access to the captive network, and learn about the CAPPORT API URI.
- The User Equipment attempts to communicate through the captive portal enforcement device.
- The Captive Portal Enforcement device either lets the User Equipment’s traffic through, or responds with an ICMP Unreachable
- The User Equipment requests access to outside the captive network, or requests more information, from the CAPPORT API server
- The CAPPORT API server directs the Captive Portal Enforcement device to either allow or deny access in response to requests from the User Equipment or quota/timing restrictions.

3. Solution Workflow

This section describes the general workflow of solutions adhering to the architecture.

3.1. Initial Connection

1. The User Equipment joins the captive network by acquiring a DHCP lease
2. The User Equipment learns the URI for the Captive Portal API from the DHCP response ([RFC7710]).
3. The User Equipment accesses the CAPPORT API to receive parameters of the Captive Network, including the token.
4. The User Equipment communicates with the CAPPORT API to gain access to the outside network.
5. The Captive Portal API server indicates to the Captive Portal Enforcement device that the User Equipment is allowed through
6. The User Equipment attempts a connection outside the captive network
7. If the requirements have been satisfied, the access is permitted; otherwise the "Expired" behavior occurs
8. The User Equipment accesses the network until conditions Expire

3.2. Connection About to Expire

1. The User Equipment sends a packet to the outside network.

2. The Captive Portal Enforcement detects that the User Equipment's access is about to expire (low quota/time/etc)

3. The Captive Portal Enforcement sends an ICMP unreachable to the User Equipment indicating that it needs to refresh its access. [I-D.wkumari-capport-icmp-unreach]. The message contains the token given to the User Equipment earlier.

4. The User Equipment verifies the message, including the token

5. The User Equipment handles this message by invoking its captive portal handling infrsatructure.

6. The captive portal handling infrastructure communicates with the Captive Portal API to gain access to outside the captive network

7. The Captive Portal API Server gives more quota (time, bytes, etc.) to the User Equipment by indicating to the Captive Portal Enforcement the new, extended quota.

8. The User Equipment continues unaffected.

3.3. Connection expired

1. The User Equipment sends a packet to the outside network.

2. The Captive Portal Enforcement device detects that the User Equipment's access has expired.

3. The remaining workflow is that same as for the initial connection.

User Equipment may attempt to maintain transport connections, leaving it to the application to determine timeouts.

User Equipment may preemptively invoke its captive portal handling infrastructure when receiving the DHCP response indicating that it is behind a captive portal, rather than waiting for the ICMP unreachable message.

4. IANA Considerations

This memo includes no request to IANA.

5. Security Considerations

5.1. Authenticated APIs

The solution described here assumes that when the User Equipment needs to trust the API server, server authentication will be utilized.

TODO: this document has not specified the authentication mechanism.

5.2. Risk of Nuisance Captive Portal

It is possible for any user on the Internet to send ICMP packets in an attempt to cause the receiving equipment to go to the captive portal. This has been considered and addressed in the following ways:

The ICMP packet does not carry the URL, making this method safer than 307-redirect methods currently in use.

The ICMP packet carries a token that would not be available, even to an on-path attacker. Although possible to guess by brute force, the impact is nuisance due to other precautions. We suggest a 32-bit token would be sufficient to deter nuisance attacks.

Even when redirected, the User Equipment securely authenticates with API servers.

5.3. User Options

The ICMP messaging informs the end-user device it is being held captive. There is no requirement that the device do something about this. Devices may permit users to disable automatic reaction to captive-portal indications. Hence, end-user devices may allow users to manually control captive portal interactions.

6. References

6.1. Normative References

[RFC2119] Bradner, S., "Key words for use in RFCs to Indicate Requirement Levels", BCP 14, RFC 2119, DOI 10.17487/RFC2119, March 1997, <http://www.rfc-editor.org/info/rfc2119>.

[RFC7710] Kumari, W., Gudmundsson, O., Ebersman, P., and S. Sheng, "Captive-Portal Identification Using DHCP or Router Advertisements (RAs)", RFC 7710, DOI 10.17487/RFC7710, December 2015, <http://www.rfc-editor.org/info/rfc7710>.

6.2. Informative References

[I-D.nottingham-capport-problem] Nottingham, M., "Captive Portals Problem Statement", draft-nottingham-capport-problem-01 (work in progress), April 2016.

[I-D.wkumari-capport-icmp-unreach] Bird, D. and W. Kumari, "Captive Portal ICMP Destination Unreachable", draft-wkumari-capport-icmp-unreach-01 (work in progress), April 2015.

Authors' Addresses

Kyle Larose
Sandvine
408 Albert Street
Waterloo, ON N2L 3V3
Canada

Phone: +1 519 880 2400
Email: klarose@sandvine.com

David Dolson
Sandvine
408 Albert Street
Waterloo, ON N2L 3V3
Canada

Phone: +1 519 880 2400
Email: ddolson@sandvine.com