             Captive-Portal Identification in DHCP / RA
                    draft-ietf-capport-rfc7710bis-00

Abstract

   In many environments offering short-term or temporary Internet access
   (such as coffee shops), it is common to start new connections in a
   captive portal mode. This highly restricts what the customer can do
   until the customer has authenticated.

   This document describes a DHCP option (and a Router Advertisement
   (RA) extension) to inform clients that they are behind some sort of
   captive-portal device, and that they will need to authenticate to get
   Internet access. It is not a full solution to address all of the
   issues that clients may have with captive portals; it is designed to
   be used in larger solutions. The method of authenticating to, and
   interacting with the captive portal is out of scope of this document.

   [This document is being collaborated on in Github at:
   https://github.com/wkumari/draft-ekwk-capport-rfc7710bis. The most
   recent version of the document, open issues, etc should all be
   available here. The authors (gratefully) accept pull requests. Text
   in square brackets will be removed before publication.]

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 January 3, 2020.

Table of Contents

[1](1). **Introduction**

   In many environments, users need to connect to a captive-portal
   device and agree to an Acceptable Use Policy (AUP) and / or provide
   billing information before they can access the Internet. It is
   anticipated that the IETF will work on a more fully featured protocol
   at some point, to ease interaction with Captive Portals. Regardless
   of how that protocol operates, it is expected that this document will
   provide needed functionality because the client will need to know
   when it is behind a captive portal and how to contact it.

In order to present users with the payment or AUP pages, the captive-portal device has to intercept the user's connections and redirect the user to the captive portal, using methods that are very similar to man-in-the-middle (MITM) attacks. As increasing focus is placed on security, and end nodes adopt a more secure stance, these interception techniques will become less effective and/or more intrusive.

This document describes a DHCP ([RFC2131]) option (Captive-Portal) and an IPv6 Router Advertisement (RA) ([RFC4861]) extension that informs clients that they are behind a captive-portal device and how to contact it.

1.1. Requirements Notation

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 [RFC2119].

2. The Captive-Portal Option

The Captive Portal DHCP / RA Option informs the client that it may be behind a captive portal and provides the URI to access an API as defined by [draft-ietf-capport-api]. This is primarily intended to improve the user experience by getting them to the captive portal faster and more reliably. Note that, for the foreseeable future, captive portals will still need to implement the interception techniques to serve legacy clients, and clients will need to perform probing to detect captive portals.

In order to support multiple "classes" of clients (e.g. IPv4 only, IPv6 only with DHCPv6 ([RFC3315]), IPv6 only with RA) the captive portal can provide the URI via multiple methods (IPv4 DHCP, IPv6 DHCP, IPv6 RA). The captive portal operator should ensure that the URIs handed out are equivalent to reduce the chance of operational problems. The maximum length of the URI that can be carried in IPv4 DHCP is 255 bytes, so URIs longer than 255 bytes should not be used in IPv6 DHCP or IPv6 RA.

In all variants of this option, the URI MUST be that of the captive portal API endpoint, conforming to the recommendations for such URIs [cite:API] (i.e. the URI SHOULD contain a DNS name and SHOULD reference a secure transport, e.g. https).

A captive portal MAY redirect requests that do not have an Accept header field ([RFC7231] Section 5.3) containing a field item whose content-type is "application/capport+json" to the URL conveyed in the "user-portal-url" API key. When performing such content negotiation

([RFC7231] Section 3.4), captive portals need to keep in mind that
such responses might be cached, and therefore SHOULD include an
appropriate Vary header field ([RFC7231] Section 7.1.4) or mark them
explicitly uncacheable (for example, using Cache-Control: no-store
[RFC7234] Section 5.2.2.3).

A captive portal MAY do content negotiation ([RFC7231] section 3.4)
and attempt to redirect clients querying without an explicit
indication of support for the captive portal API content type (i.e.
without application/capport+json listed explicitly anywhere within an
Accept header vis. [RFC7231] section 5.3). In so doing, the captive
portal SHOULD redirect the client to the value associated with the
"user-portal-url" API key.

The URI SHOULD NOT contain an IP address literal. The URI parameter
is not null terminated.

Networks with no captive portals MAY explicitly indicate this
condition by using this option with the IANA-assigned URI for this
purpose (see Section 5.1.1). Clients observing the URI value
"urn:ietf:params:capport-unrestricted" MAY forego time-consuming
forms of captive portal detection.

2.1. IPv4 DHCP Option

The format of the IPv4 Captive-Portal DHCP option is shown below.

```
     Code    Len              Data
    +------+------+------+------+------+--   --+-----+
    | code | len  |  URI                ...        |
    +------+------+------+------+------+--   --+-----+
```

o Code: The Captive-Portal DHCPv4 Option (160) (one octet)

o Len: The length, in octets of the URI.

o URI: The URI for the captive portal API endpoint to which the user
   should connect (encoded following the rules in [RFC3986]).

2.2. IPv6 DHCP Option

The format of the IPv6 Captive-Portal DHCP option is shown below.

```
 0                   1                   2                   3
 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|            option-code           |           option-len          |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
.                       URI (variable length)                    .
|                              ...                               |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```

   o option-code: The Captive-Portal DHCPv6Option (103) (two octets)

   o option-len: The length, in octets of the URI.

   o URI: The URI for the captive portal API endpoint to which the user
      should connect (encoded following the rules in [RFC3986]).

   See [RFC7227], Section 5.7 for more examples of DHCP Options with
   URIs.

2.3. The Captive-Portal IPv6 RA Option

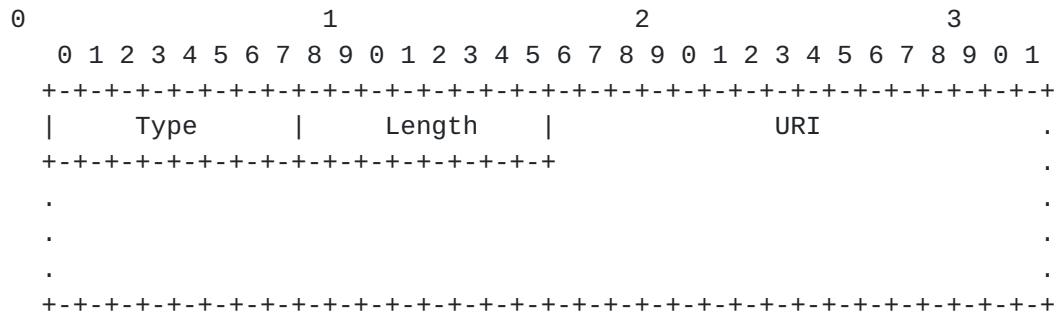   This section describes the Captive-Portal Router Advertisement
   option.

```
 0                   1                   2                   3
   0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1
  +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
  |     Type      |    Length     |              URI              .
  +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+                               .
  .                                                              .
  .                                                              .
  .                                                              .
  +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
              Figure 2: Captive-Portal RA Option Format
```

   Type   37

   Length 8-bit unsigned integer. The length of the option (including
      the Type and Length fields) in units of 8 bytes.

   URI The URI for the captive portal API endpoint to which the user
      should connect. This MUST be padded with NULL (0x00) to make the
      total option length (including the Type and Length fields) a
      multiple of 8 bytes.

3. The Captive-Portal Link Relation Type

Some captive portal network deployments may be unable to change, or
unwilling to risk changing, the network infrastructure necessary to
use any of the above options. In such deployments, when clear text
HTTP intercept and redirection are used, a Link relation header
([RFC8288], Section 3.3) MAY be inserted to convey to a HTTP client
(user agent) the associated Captive Portal API URI.

HTTP user agents MUST ignore this link relation in any context other
than when explicitly probing to detect the presence of a captive
portal. Failure to do so could allow an attacker to inject a Captive
Portal API URI other than the correct URI for a given network or for
networks where there is no captive portal present at all.

4. Precedence of API URIs

A device may learn about Captive Portal API URIs through more than
one of (or indeed all of) the above options. It is a network
configuration error if the learned URIs are not all identical.

However, if the URIs learned are not in fact all identical the
captive device MUST prioritize URIs learned from network provisioning
or configuration mechanisms before all other URIs. Specifically,
URIs learned via any of the options in Section 2 should take
precedence over any URI learned via a mechanism like the one
described in Section 3.

If the URIs learned via more than one option described in Section 2
are not all identical, this condition should be logged for the device
owner or administrator. URI precedence in this situation is not
specified by this document.

5. IANA Considerations

This document requests two new IETF URN protocol parameter
([RFC3553]) entries.

Thanks IANA!

5.1. IETF params Registration

5.1.1. Registry name: Captive Portal Unrestricted Identifier

Registry name: Captive Portal Unrestricted Identifier

URN: urn:ietf:params:capport-unrestricted

Specification: RFC TBD (this document)

Repository: RFC TBD (this document)

Index value: Only one value is defined (see URN above). No hierarchy
is defined and therefore no sub-namespace registrations are possible.

5.1.2. **Registry name: Captive Portal API Link Relation Type**

Registry name: Captive Portal API Link Relation Type

URN: urn:ietf:params:capport-api

Specification: RFC TBD (this document)

Repository: RFC TBD (this document)

Index value: Only one value is defined (see URN above). No hierarchy
is defined and therefore no sub-namespace registrations are possible.

6. **Security Considerations**

An attacker with the ability to inject DHCP messages, RAs, or HTTP
headers into cleartext HTTP communications could include an option or
link relation from this document and so force users to contact an
address of his choosing. As an attacker with this capability could
simply list himself as the default gateway (and so intercept all the
victim's traffic); this does not provide them with significantly more
capabilities, but because this document removes the need for
interception, the attacker may have an easier time performing the
attack. As the operating systems and application that make use of
this information know that they are connecting to a captive-portal
device (as opposed to intercepted connections) they can render the
page in a sandboxed environment and take other precautions, such as
clearly labeling the page as untrusted. The means of sandboxing and
user interface presenting this information is not covered in this
document - by its nature it is implementation specific and best left
to the application and user interface designers.

Devices and systems that automatically connect to an open network
could potentially be tracked using the techniques described in this
document (forcing the user to continually authenticate, or exposing
their browser fingerprint). However, similar tracking can already be
performed with the standard captive portal mechanisms, so this
technique does not give the attackers more capabilities.

Captive portals are increasingly hijacking TLS connections to force
browsers to talk to the portal. Providing the portal's URI via a

DHCP or RA option is a cleaner technique, and reduces user
expectations of being hijacked - this may improve security by making
users more reluctant to accept TLS hijacking, which can be performed
from beyond the network associated with the captive portal.

By simplifying the interaction with the captive portal systems, and
doing away with the need for interception, we think that users will
be less likely to disable useful security safeguards like DNSSEC
validation, VPNs, etc. In addition, because the system knows that it
is behind a captive portal, it can know not to send cookies,
credentials, etc. By handing out a URI using which is protected with
TLS, the captive portal operator can attempt to reassure the user
that the captive portal is not malicious.

Operating systems should conduct all interactions with the API in a
sand-boxed environment and with a configuration that minimizes
tracking risks.

7. Acknowledgements

This document is a -bis of RFC7710. Thanks to all of the original
authors (Warren Kumari, Olafur Gudmundsson, Paul Ebersman, Steve
Sheng), and original contributors.

Also thanks to the CAPPORT WG for all of the discussion and
improvements.

8. 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, <https://www.rfc-
           editor.org/info/rfc2119>.

[RFC2131] Droms, R., "Dynamic Host Configuration Protocol",
           RFC 2131, DOI 10.17487/RFC2131, March 1997,
           <https://www.rfc-editor.org/info/rfc2131>.

[RFC3315] Droms, R., Ed., Bound, J., Volz, B., Lemon, T., Perkins,
           C., and M. Carney, "Dynamic Host Configuration Protocol
           for IPv6 (DHCPv6)", RFC 3315, DOI 10.17487/RFC3315, July
           2003, <https://www.rfc-editor.org/info/rfc3315>.

[RFC3553] Mealling, M., Masinter, L., Hardie, T., and G. Klyne, "An
           IETF URN Sub-namespace for Registered Protocol
           Parameters", BCP 73, RFC 3553, DOI 10.17487/RFC3553, June
           2003, <https://www.rfc-editor.org/info/rfc3553>.

   [RFC3986] Berners-Lee, T., Fielding, R., and L. Masinter, "Uniform
              Resource Identifier (URI): Generic Syntax", STD 66,
              RFC 3986, DOI 10.17487/RFC3986, January 2005,
              <https://www.rfc-editor.org/info/rfc3986>.

   [RFC4861] Narten, T., Nordmark, E., Simpson, W., and H. Soliman,
              "Neighbor Discovery for IP version 6 (IPv6)", RFC 4861,
              DOI 10.17487/RFC4861, September 2007, <https://www.rfc-
              editor.org/info/rfc4861>.

   [RFC7227] Hankins, D., Mrugalski, T., Siodelski, M., Jiang, S., and
              S. Krishnan, "Guidelines for Creating New DHCPv6 Options",
              BCP 187, RFC 7227, DOI 10.17487/RFC7227, May 2014,
              <https://www.rfc-editor.org/info/rfc7227>.

   [RFC7231] Fielding, R., Ed. and J. Reschke, Ed., "Hypertext Transfer
              Protocol (HTTP/1.1): Semantics and Content", RFC 7231,
              DOI 10.17487/RFC7231, June 2014, <https://www.rfc-
              editor.org/info/rfc7231>.

   [RFC7234] Fielding, R., Ed., Nottingham, M., Ed., and J. Reschke,
              Ed., "Hypertext Transfer Protocol (HTTP/1.1): Caching",
              RFC 7234, DOI 10.17487/RFC7234, June 2014,
              <https://www.rfc-editor.org/info/rfc7234>.

   [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, <https://www.rfc-editor.org/info/rfc7710>.

   [RFC8288] Nottingham, M., "Web Linking", RFC 8288,
              DOI 10.17487/RFC8288, October 2017, <https://www.rfc-
              editor.org/info/rfc8288>.

Appendix A. Changes / Author Notes.

   [RFC Editor: Please remove this section before publication]

   From initial to -00.

   o Import of RFC7710.

Appendix B. Changes from RFC 7710

   This document incorporates the following changes from [RFC7710].

   1. Clarify that IP string literals are NOT RECOMMENDED.

   2. Clarify that the option URI SHOULD be that of the captive portal
       API endpoint.

   3. Clarify that captive portals MAY do content negotiation.

   4. Added text about Captive Portal API URI precedence in the event
       of a network configuration error.

   5. Added urn:ietf:params:capport-unrestricted URN.

   6. Added urn:ietf:params:capport-api URN.

Authors' Addresses

   Warren Kumari
   Google
   1600 Amphitheatre Parkway
   Mountain View, CA 94043
   US

   Email: warren@kumari.net

   Erik Kline
   Loon
   1600 Amphitheatre Parkway
   Mountain View, CA 94043
   US

   Email: ek@google.com