Workgroup: Internet Engineering Task Force
Internet-Draft: draft-adell-client-roaming-00
Published: 15 June 2022
Intended Status: Informational
Expires: 17 December 2022
Authors: E. Adell

Client Roaming Control

Abstract

This document describes the Client Roaming Control (CRC) technique to allow an organization to control the access to third-party applications over Internet. It specifies the _crc Global Underscored Node Name for organizations willing to implement this technique. A new Client Roaming Support (CRS) Resource Record is also introduced for the applications supporting an authorization mechanism honoring the CRC, in order to inform of this support.

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 https://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 17 December 2022.

Table of Contents

1. Introduction

Illegitimate access to professional restricted applications over Internet is a permanent threat for organizations and their staff. Different methods can be used to impersonate a user access, and in some cases an organization also wants to better prevent its own staff to access a third-party application from a network which is not under its control. On the contrary, an organization maybe wants to allow roaming then its users can access from different known places.

Associated to the Address Prefixes Lists (APL) [RFC3123] Resource Record (RR), the _crc global underscored node name acts a White-List and informs a compatible application from which networks its users are allowed to connect, be it a limited list of networks or broadly without any restriction.

At the application level, the identification of the user's organization domain can be based on an information carried during the authentication process, or a lookup on an information already known by the application. In both cases this information lets the application relate the user to its organization unequivocally. Finally, the corresponding user's domain DNS will be requested with the application's FQDN and port, and the application will know

whether an authorization is expected or not. The precise syntax of this request and some examples are given in this document.

The applications implementing this authorization control let the client organizations know this feature is available by using the Client Roaming Support (CRS) RR. The data associated with this record indicates if the client's organization expected support of the CRC is mandatory, optional, or ignored. This information stored in the CRS can be confirmed at the application level by a redundant data. The way the application handles the authorization mechanism, by consulting the associated CRS record in real-time or relying on a configuration file, is left to the implementor.

Although this mechanism is designed for improving the security between different organizations, there is no objection to use it for a same organization playing both roles of client and application , as an alternative or additional layer to a solution already in place, such as a firewall for example.

2. Conventions Used in This Document

This specification uses definitions from Domain Name System [RFC1035], and readers unfamiliar with it can also check DNS Terminology [RFC8499].

The key words "MUST", "MUST NOT", "REQUIRED", "SHALL", "SHALL NOT", "SHOULD", "SHOULD NOT", "RECOMMENDED", "NOT RECOMMENDED", "MAY", and "OPTIONAL" in this document are to be interpreted as described in BCP 14 [RFC2119] [RFC8174] when, and only when, they appear in all capitals, as shown here.

3. The CRC Global Underscored Node Name

The _crc Node Name (NN) purpose is to provide a list of IP ranges authorized to use a particular application. The APL RR being designed to store lists of address prefixes, it is used here in association with the _crc NN.

3.1. APL Applicability Statement

Using the APL RR requires to define the precise behavior for ordinary and particular syntaxes. Acting as a White-List, the following characteristics MUST be implemented for interpreting correctly the CRC value :

*multiple RRSets are allowed, and the expected meaning is an union of all their prefixes

*both address families 1 and 2 MUST be supported

*mixing different address families in the same record is NOT RECOMMENDED

*an empty RR is allowed but NOT RECOMMENDED

*the negated expression "!" has no meaning and is NOT RECOMMENDED, if used at all it MUST be ignored by applications

3.2. Leaf Node Name construction

The leaf node name is built by concatenating the application domain name, its listening port expressed as a subordinate underscored node name, and the CRC global underscored node name.

For example, the FTP application hosted on ftp.example.com and listening on port 21 will be associated with this leaf : ftp.example.com._21._crc

4. The CRS Resource Record

The CRS RR indicates which control is done on the client organizations, and thus which ones are authorized. A requirement field is used for this purpose, it has one of the following values and meanings when the checking is performed :

*"N" : Never, all organizations are authorized

*"A" : Always, only organizations with a CRC are authorized

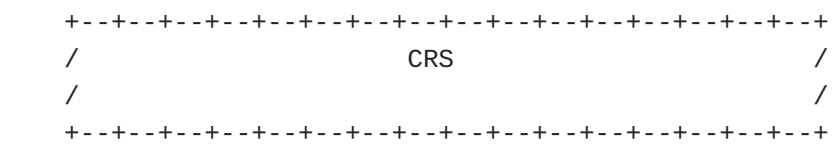
*"O" : Optional, any organization CRC is honored, other organizations are authorized

In addition to this value, an optional list of ports can be given. Indeed, multiple applications can be hosted on different ports under the same domain name, and an equivalent support was described for the CRC NN. In case of different requirement values, it is RECOMMENDED to have one dedicated RR for each although one single RR with all the information is supported. One particular port MUST NOT appear in more than one RR. When no port is mentioned, only one RR MAY be declared and its requirement value covers all applications for this domain name.

In the absence of such record, no roaming control is to be expected by the client, any of its CRC NNs will be ignored. It is equivalent to a CRS requirement value "N" indicating no control is performed.

4.1. CRS RDATA Wire Format

The CRS RDATA wire format is encoded as follows:



The CRS field contains a list of requirements followed by their respective optional ports.

4.2. CRS Presentation Format

The presentation format of the CRS record is:

```
CRS (single-rule / multiple-rules)

single-rule = "R=" ("N" / "A" / "O") *(,port)

multiple-rules = unit-rule 1*2(;unit-rule)

unit-rule = "R=" ("N" / "A" / "O") 1*(,port)

port = [1-9] *([DIGIT])
```

5. Examples

The following examples show some typical uses expected from this documentation. Particularly, the intended behaviors for different CRC and CRS values are explained, while the user identification is done directly through carried data or a deduction process.

5.1. Restricted Application

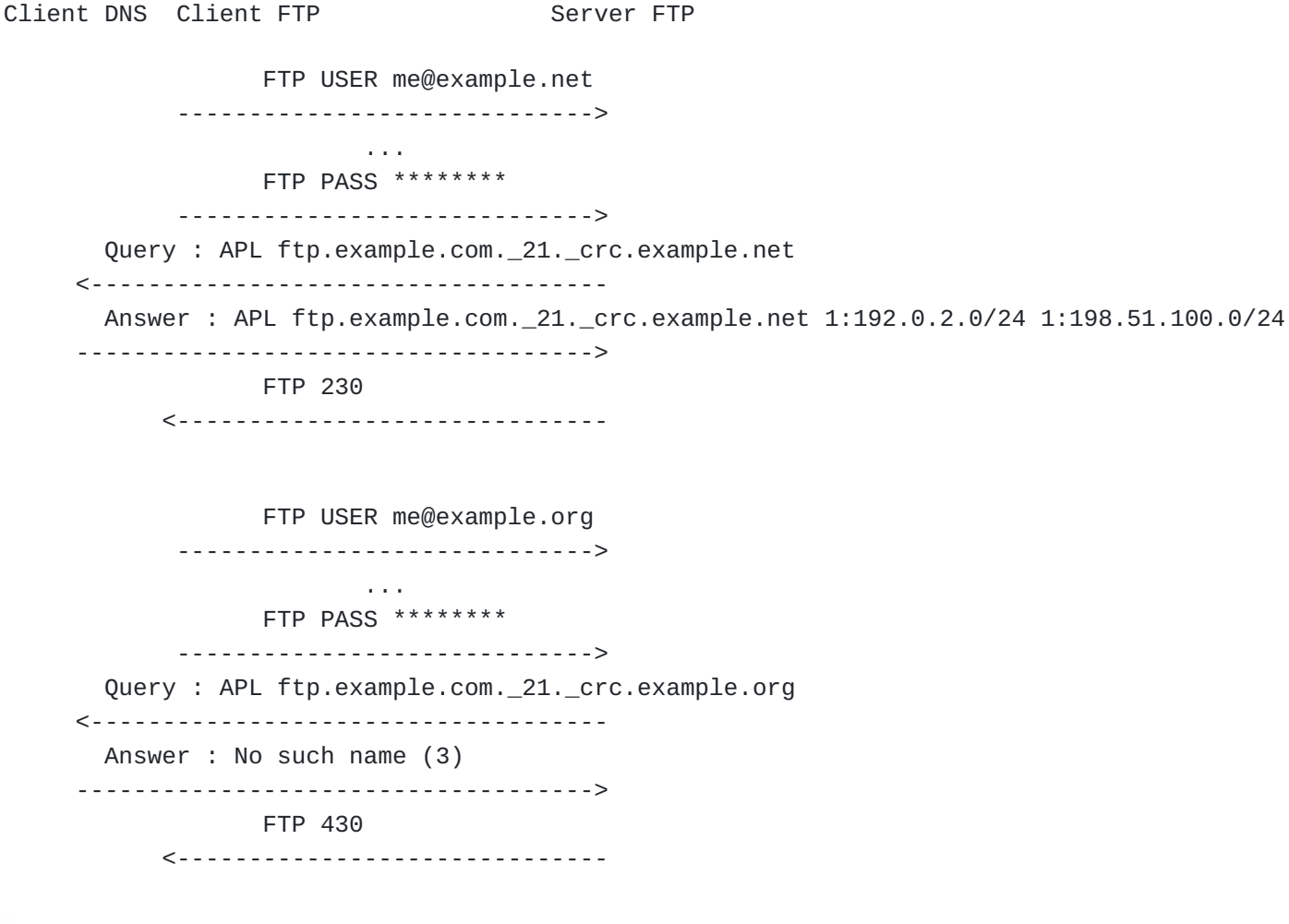
In this example, an application is only opened to organizations publishing their respective allowed networks. The requirement value of the CRS record equals "A", and any organization with an empty or missing CRC for this application will be denied access.

The ftp.example.com domain is dedicated to hosting an FTP application, which extracts the client's domain from the username used during the authentication process. This information is then used for requesting the client APL record and finally comparing its content with the client's IP. The client organization example.net allows its users from its own network 192.0.2.0/24 and from a cloud service located at 198.51.100.0/24. A second organization example.org has no APL record and its users are rejected.

Application FQDN : ftp.example.com
Application CRS record : ftp.example.com. IN CRS R=A,21

```
   Client FQDN : example.net
   Client organization APL record :
   ftp.example.com._21._crc.example.net. IN APL 1:192.0.2.0/24
   1:198.51.100.0/24

   Client FQDN : example.org
   No client organization APL record
```



5.2. Controlled Application

The www.example.com domain hosts a Web application on port 443 using client certificates for authenticating its users. The application extracts the client domains from the certificates, which are used to retrieve their APL records. Users from the example.net organization are allowed only if they connect from an authorized network listed in the APL record, while users from example.org are always granted access since this one has no APL declared.

```
   Application FQDN : www.example.com
   Application CRS record : www.example.com. IN CRS R=O,443
```

```
   Client FQDN : example.net
   Client organization APL record :
   www.example.com._443._crc.example.net. IN APL 1:192.0.2.0/24
   1:198.51.100.0/24

   Client FQDN : example.org
   No client organization APL record

Client DNS  Client browser               Web application


                        .....
             Client certificate me@example.net
           ----------------------------------->
     Query : APL www.example.com._443._crc.example.net
    <-----------------------------------------
     Answer : APL www.example.com._443._crc.example.net 1:192.0.2.0/24 1:198.51.100.0/24
    ----------------------------------------->
                        .....
                200 OK
          <-----------------------------------


                        .....
             Client certificate me@example.org
           ----------------------------------->
     Query : APL www.example.com._443._crc.example.org
    <-----------------------------------------
     Answer : No such name (3)
    ----------------------------------------->
                        .....
                200 OK
          <-----------------------------------
```

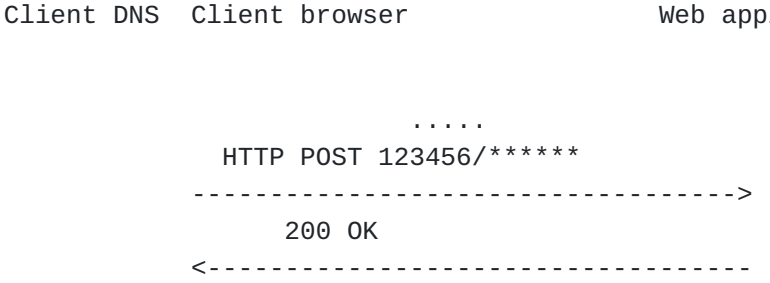
5.3. Opened Application

A company is testing the CRC and CRS behaviors before opening a new service to its customers. Its first test described below consists in configuring both sides to be completely opened, likely before hardening the CRS, then the APL, and testing again.

The application.example.com domain hosts a Web application on port 443 where users are logged in by sending a numerical identifier and a password. The application uses a dictionary data type to identify the user's domain. The client.example.net domain is temporarily using 2 APL records (one for each IP version) indicating a free access from anywhere.

Application FQDN : application.example.com
Application CRS record : application.example.com. IN CRS R=N,443

Client FQDN : client.example.net
Client organization APL records :
application.example.com._443._crc.example.net. IN APL 1:0.0.0.0/24
application.example.com._443._crc.example.net. IN APL 2:fe80::/10



6. IANA Considerations

According to Guidelines for Writing an IANA Considerations Section in RFCs [RFC8126] it is asked to IANA to add into the Resource Record (RR) TYPEs registry located at https://www.iana.org/assignments/dns-parameters/dns-parameters.xhtml#dns-parameters-4 this CRS entry.

TYPE	Value	Description	Reference
CRS	TBD1	Client Roaming Support	this RFC

Table 1

It is also asked to IANA to add into the Underscored and Globally Scoped DNS Node Names registry located at https://www.iana.org/assignments/dns-parameters/dns-parameters.xml#underscored-globally-scoped-dns-node-names this CRC entry

RR TYPE	_NODE NAME	Reference
APL	_crc	this RFC

Table 2

7. Security Considerations

This section is meant to inform developers and users of the security implications of the CRC/CRS mechanism described by this document. While the CRS RR mostly plays an informative role, the CRC Node Name delivers important data which requires attention from the developers and administrators. Some particular points are discussed here.

7.1. DNS Security

Client and application administrators are encouraged to take care of their DNS infrastructure and operation management. In particular, the client DNS becoming unavailable or unresponsive could in turn make the application unavailable. The restricted and controlled scenarios are expected to just bring down the application in such case, and not disable the authorization turning things into an open scenario.

As the CRC node names are supposed to be requested during an application authentication process, reflection attacks could be built to target a client organization, even one not hosting any CRC entry at all.
In a general manner, administrators may consider an adequate TTL setting to be resilient to short time DNS unavailability and to not overload client organizations, enable TCP as the preferred transport, and rely on DNSSEC to warrant data authenticity and integrity.

7.2. DNS misconfiguration

Any DNS CRS misconfiguration such as multiple records with different requirement values but with the same port value can get a client confused. In this case the client does not know without testing the actual configuration, if its organization is protected against roaming, and contacting the application administrator to fix the situation is a possibility.

While CRC misconfigurations are leading to more or less serious security problems, administrators need to pay attention when dealing with multiple networks or records. Particularly, multiple records for the same network range or overlapping networks should be avoided.

7.3. Application Security

The following points are of concern to developers:

Encryption:
Whenever possible, the application protocol should be encrypted to prevent eavesdropping and man-in-the-middle attacks. It is a critical point for applications maintaining a user session with anything like a token or cookie, as it can lead to session hijacking as discussed below.

Timing attack:
All authentication systems need to be careful to not deliver any information derived from the computing time to a denied user, even the ones involving multiple factors or steps like the one described

in this document. In particular, the order in which these steps are executed and their respective implementations, need to defeat statistical hypotheses.

Intermediate systems:
Some applications are not directly Internet facing and cannot access to the real client's IP address without involving a mechanism to forward this IP at the application layer. For example with HTTP, the common practice based on the non-standard X-Forwarded-For header, or its alternative standard Forwarded [RFC7239], are playing this role. Such practice requires a correct sanitizing of user data to avoid false injected IPs.

Session hijacking:
A well-known attack called Session Hijacking is not meant to be defeated by this document alone. Application developers must ensure that any receveid session token, such as an HTTP Cookie, belongs to the same IP address than the one which started this session.

8. References

8.1. Normative References

[RFC1035] Mockapetris, P., "Domain names - implementation and specification", STD 13, RFC 1035, DOI 10.17487/RFC1035, November 1987, <https://www.rfc-editor.org/info/rfc1035>.

[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>.

[RFC8174] Leiba, B., "Ambiguity of Uppercase vs Lowercase in RFC 2119 Key Words", BCP 14, RFC 8174, DOI 10.17487/RFC8174, May 2017, <https://www.rfc-editor.org/info/rfc8174>.

[RFC8552] Crocker, D., "Scoped Interpretation of DNS Resource Records through "Underscored" Naming of Attribute Leaves", BCP 222, RFC 8552, DOI 10.17487/RFC8552, March 2019, <https://www.rfc-editor.org/info/rfc8552>.

8.2. Informative References

[RFC7239] Petersson, A. and M. Nilsson, "Forwarded HTTP Extension", RFC 7239, DOI 10.17487/RFC7239, June 2014, <https://www.rfc-editor.org/info/rfc7239>.

[RFC8499] Hoffman, P., Sullivan, A., and K. Fujiwara, "DNS Terminology", BCP 219, RFC 8499, DOI 10.17487/RFC8499, January 2019, <https://www.rfc-editor.org/info/rfc8499>.

Author's Address

Eugene Adell

Email: eugene.adell@gmail.com