INTERNET-DRAFT
<draft-yu-imap-client-id-00.txt>
Intended Status: Standards Track
Expires November 24, 2018

Y. Deion
LinuxMagic

May 24, 2018

IMAP Service Extension for Client Identity
<draft-yu-imap-client-id-00.txt>

Abstract

This document defines an Internet Message Access Protocol (IMAP) service extension called "CID" which provides a method for clients to indicate an identity to the server.

This identity is an additional token that may be used for security and/or informational purposes, and with it a server may optionally apply heuristics using this token.

Table of Contents

1. Introduction

The [IMAP] protocol and its extensions describe methods whereby an client may provide identity and/or authentication information to an IMAP server. However, these existing methods are subject to limitations and none offer a way to identify the IMAP client with absolute confidence. This document defines an IMAP service extension to provide an additional identity token which can represent the IMAP client with a higher degree of certainty when accessing the IMAP server.

Typically IMAP clients enter the authenticated state by using either the AUTHENTICATE or LOGIN command. IMAP servers are often subject to malicious clients attempting to use authorization credentials and/or identities not intended for their use (e.g. stolen credentials or brute force attacks). When such an attack is attempted, the IMAP server may be unable to identify the impersonation and restrict such an unintended use by someone other than the authorized user or said credentials. While there are ways to identify the source of the IMAP client such as its IP address, it would be useful if there was an

additional way to uniquely identify the client in a method solely available across an encrypted channel.

Using the CID extension, an IMAP client can provide an additional identity token to the server called its "client identity". The

client identity can provide unique characteristics about the client accessing the IMAP service and may be combined with existing identification mechanisms in order to identify the client. An IMAP server may then apply additional security policies using this identity such as restricting use of the service to clients presenting recognized client identities or only allowing use of authorized identities that match previously established client identities.

The CID extension is present in any IMAP implementation that returns "CID" as one of the supported capabilities to the CAPABILITY command.

2. Conventions Used in This Document

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 [KEYWORDS].

Formal syntax is specified using [ABNF].

Example lines prefaced by "C:" are sent by the client and ones prefaced by "S:" by the server.

"Connection" refers to the entire sequence of client/server interaction from the initial establishment of the network connection until its termination.

3. CID

3.1. CID Command

```
Arguments:  client identity type
            client identity token

Responses:  no specific responses for this command

Result:     OK - cid completed, client identity information stored
            BAD - command unknown or arguments invalid
```

Note that a valid CID command will never return the NO result because heuristics MUST NOT be applied to the CID arguments at this stage. Instead the client identity information SHOULD be stored and passed along to any and all [SASL] authentication mechanisms.

3.2. CID Arguments

The CID command takes the following two arguments:

1. client identity type: A string identifying the identity type the client is providing. It MUST be between 1 and 16 alphanumeric

characters.

2. client identity token: A string identifying the client. It MUST be between 1 and 128 printable characters.

The IMAP server MUST reject any CID command with badly formatted arguments. The IMAP server MUST accept the arguments from a valid CID command and SHOULD store it at the minimum for the remaining duration of the IMAP connection.

3.3. Advertising the CID capability

The CID capability is used to tell the IMAP client that the IMAP server supports the CID extension. However, certain conditions MUST be met before the IMAP server advertises the CID capability.

1. The IMAP server and IMAP client MUST negotiate encryption via STARTTLS/SSL or some other secure mechanism.

2. The IMAP server MUST be in the non-authenticated state.

3. The IMAP server MUST have the CID extension support enabled.

While all the conditions are met, the IMAP server MUST advertise the CID capability in all proceeding CAPABILITY commands.

3.4. Restrictions on the CID command

Under certain circumstances, the use of the CID command will be restricted:

1. Before the CID capability has been advertised, the IMAP server MUST reject any issued CID commmand and the IMAP client MUST NOT issue the CID command.

2. Outside of the non-authenticated state, the IMAP server MUST reject any CID command issued by the IMAP client and the IMAP client MUST NOT issue the CID command.

3. Once a valid CID command has been issued, the IMAP server MUST reject any further CID command issued by the IMAP client and the IMAP client MUST NOT issue any subsequent CID commands.

4. Formal Syntax

The following syntax specification uses the Augmented Backus-Naur Form notation as specified in [ABNF]. [IMAP] defines the non-terminals "capability" and "command-nonauth".

Except as noted otherwise, all alphabetic characters are case-insensitive. The use of upper or lower case characters to define token strings is for editorial clarity only. Implementations MUST accept these strings in a case-insensitive fashion.

```
capability      =/ "CID"
```

```
   command-nonauth =/ cid

   cid             = "CID" SP client-id-type SP client-id-token
```

```
client-id-type  = 1*16 ALPHA / DIGIT
                ;; alphanumeric

client-id-token = 1*128 VCHAR
                ;; any printable US-ASCII character
```

5. Discussion

5.1. Applying heuristics to CID

This section discusses the possible heuristics that can be applied to the information that is presented via the CID command. This information includes whether a valid CID command was issued, the client identity type and the client identity token.

1. The IMAP server MAY choose to require that a successful CID command be issued or that a particular client identity type be presented.

2. The IMAP server MAY reject any CID command with a client identity type that is not recognized by the IMAP server.

3. The IMAP server MAY reject any CID command with a client identity type that is not supported by the IMAP server.

4. An IMAP server MAY reject any CID command that contains a client identity type or client identity token that the server chooses not to accept for any reason such as by policy.

5. An IMAP server MAY reject any CID command that contains a client identity type or client identity that the server has chosen to disable or revoke use of either temporarily or permanently.

The IMAP server SHOULD only ever reject an IMAP client based on CID information during or after the authentication process/handler. In the interest of limiting the amount of information being revealed, the rejection message SHOULD be as generic as possible and SHOULD NOT reveal any information on the heuristics.

Even if the client identity type and/or client identity token are not recognized, supported or permitted by the server and/or the owner the authentication credentials, the presented information may still be useful for analysis.

5.2. Utility of CID

Regardless of how much it is frowned upon, common authorization information like the username and password pair are reused across multiple web services. When this authorization is compromised on a single web service, malicious actors usually also gain access to

other web services. Based on this information alone, the utility of CID as an additional layer of authentication that is only available across an encrypted channel becomes more apparent.

The utility of CID may be seen by considering the following:

1. An IMAP client may utilize the same IMAP server with multiple different authorized identities, so an identity that persists across authorized identities is lacking.

2. An authorized identity may make use of multiple discrete devices over different IMAP sessions, so an identity persisting on one device is lacking.

3. Existing identity information available from the connection such as network address or IP changes frequently as devices are becoming more mobile in nature.

4. Individual IMAP services have no method to determine if devices types should be permitted e.g. private IMAP services that do not persist across different connections.

5. There is no method for legacy authentication methods to associate a given set of authentication tokens to an individual and or that individuals registered devices.

5.3. Use Cases of CID

With CID the IMAP server has additional information it may use in its interactions with the client. It may:

1. Restrict use of an authorization tokens to a set of client identities, thereby offering an added level of security. For example the use of an authorization token may only be accompanied by a specified set of CID tokens and/or types.

2. Identify that the same CID token is used to access multiple authorized identities, and restrict access to the IMAP service. For example a malicious client that has attempted to gain access using multiple authorization tokens may be identified through its unusual behavior.

3. Retain knowledge of CID tokens previously presented with specific authorization credentials, and if the token has not been previously seen, restrict access to the IMAP service.

4. Require that the IMAP client present a token such as a license key established outside of the IMAP session in order to make use of any authorized identity.

5. Apply different security policies to clients that provide a CID token versus those which do not. For example, provide clients providing such an identity with additional trust.

5.4. Other IMAP Client Identifiers

The [IMAP] protocol and its extensions describe methods whereby an IMAP client may provide identity information to an IMAP server. Some

of these identitiers are listed for contrast:

1. The client connection provides a source IP address associated with the IMAP session. This may be accompanied by a PTR record and/or GeoIP information.

2. The AUTHENTICATE and LOGIN command allows the client to present a user and/or password/authentication mechanism for an IMAP session.

5.5. Future Considerations

In the future there may be a demand for being able to provide multiple CID commands with different cid types.

6. Client Identity Types

This document does not specify any CID identity type that MUST be supported. Some examples of identity type are UUID, LICENSE, DEVICE_ID, MAC and COOKIE. To start with certain types such as UUID and LICENSE SHOULD be supported. It is intended that any CID type be accepted but in the future standards on types may be set but a IMAP server SHOULD NOT reject an unidentified CID type, except for specific policy use cases.

It is envisioned that in the future it will be useful to propose identity types to support.

1. UUID

   UUID is a common practice to represent either a individual user, hardware device or software installation associated with a specific individual. The support of UUID enables existing UUID implementations to be used to semi-uniquely identify a device associated with an individual.

2. LICENSE

   An IMAP client may find it useful to identify the license key of software it is using. Such licenses are typically crafted such that they are unique and useful to identify a software installation.

3. DEVICE_ID

   Many hardware devices are designed to be used by a single individual and already have an associated hardware device id.

4. MAC

The MAC address is not always available or consistent. However, for certain use cases the MAC may be the only information available to specify a specific device.

5. COOKIE

   While not guranteed to be consistent many web applications are designed to access IMAP directly and may need to have a semi-unique identifier available as part of the web based transaction.

This document recommends that an IMAP server handle any given client identity type from a CID command in one or more of the following manners.

1. Handled but treat as not presented
2. Store in session but treat as not presented (useful for debugging)
3. System log
4. User log
5. Use for authentication
6. Use for alert when authentication fails
7. Use for alert when authentication succeeds
8. Unused

7. Examples

7.1. UUID as Client Identity

```
C: [connection established over a plaintext connection]
C: a001 CAPABILITY
S: * CAPABILITY IMAP4rev1 STARTTLS AUTH=GSSAPI LOGINDISABLED
S: a001 OK CAPABILITY completed
C: a002 STARTTLS
S: a002 OK STARTLS completed
<TLS negotiation, further commands are under [TLS] layer>
C: a003 CAPABILITY
S: * CAPABILITY IMAP4rev1 AUTH=GSSAPI AUTH=PLAIN CID
S: a003 OK CAPABILITY completed
C: a004 CID UUID 23bf83be-aad7-46aa-9e0f-39191ccf402f
S: a004 OK CID completed
C: a005 LOGIN joe password
S: a005 OK LOGIN completed
```

7.2. Malformed CID Command

```
C: [connection established over a plaintext connection]
C: a001 CAPABILITY
S: * CAPABILITY IMAP4rev1 STARTTLS AUTH=GSSAPI LOGINDISABLED
S: a001 OK CAPABILITY completed
C: a002 STARTTLS
S: a002 OK STARTLS completed
<TLS negotiation, further commands are under [TLS] layer>
C: a003 CAPABILITY
```

```
S: * CAPABILITY IMAP4rev1 AUTH=GSSAPI AUTH=PLAIN CID
S: a003 OK CAPABILITY completed
C: a004 CID UUID
S: a004 BAD Error in IMAP command received by server
```

The IMAP server rejects the CID command as it is not well formed due to there being only a single parameter provided.

7.3. Client Identity without TLS/SSL Session

```
C: [connection established over a plaintext connection]
C: a001 CAPABILITY
S: * CAPABILITY IMAP4rev1 STARTTLS AUTH=GSSAPI LOGINDISABLED
S: a001 OK CAPABILITY completed
C: a002 CID UUID 23bf83be-aad7-46aa-9e0f-39191ccf402f
S: a002 BAD Unknown IMAP command received by server
```

The IMAP server rejects use of the CID command as the CID capability had not been advertised because no encryption was negotiated between the IMAP server and IMAP client.

7.4. Client Identity Leading to Rejection

```
C: [connection established over a plaintext connection]
C: a001 CAPABILITY
S: * CAPABILITY IMAP4rev1 STARTTLS AUTH=GSSAPI LOGINDISABLED
S: a001 OK CAPABILITY completed
C: a002 STARTTLS
S: a002 OK STARTLS completed
<TLS negotiation, further commands are under [TLS] layer>
C: a003 CAPABILITY
S: * CAPABILITY IMAP4rev1 AUTH=GSSAPI AUTH=PLAIN CID
S: a003 OK CAPABILITY completed
C: a004 CID UUID 23bf83be-aad7-46aa-9e0f-39191ccf402f
S: a004 OK CID completed
C: a005 LOGIN joe password
S: a005 BAD Failed to authenticate
```

The IMAP server rejects use of the system during the LOGIN command after deciding that the provided client identity does not establish sufficient privileges. Note that the error message that's returned to the client is very generic and does not reveal any information about CID and/or the existence of 'joe' and/or the validity of the password.

8. Security Considerations

As this extension provides an additional means of communicating information from a client to a server it is clear there is additional information divulged to the server. This may have privacy considerations depending on the client identity type or its contents. For example, it may reveal a MAC address of the device used to communicate with a server that would not previously have been revealed. While it has been useful to use identifier such as email

address for authentication it is easy for these authetication tokens to be shared and/or reused and/or be publically available for other purposes. An IMAP server and or its operators SHOULD not share any CID information presented with a third party as it may represent or be linked to an individual and SHOULD never be shared in

association with authentication tokens.

As well, while this service extension requires that the identity information only be transmitted over an encrypted channel to reduce the risk of eavesdropping, it does not specify any policies or practices required in the establishment of such a channel, and so it is the responsibility of the client and the server to determine that the communication medium meets their requirements.

9. IANA Considerations

The IANA is requested to add CID to the "IMAP 4 Capabilities" registry, http://www.iana.org/assignments/imap4-capabilities.

10. References

10.1. Normative References

[ABNF] Crocker, D. and P. Overell, "Augmented BNF for Syntax Specifications: ABNF", RFC 2234, November 1997.

[IMAP] Crispin, M., "INTERNET MESSAGE ACCESS PROTOCOL - VERSION 4rev1", RFC 3501, March 2003.

[KEYWORDS] 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>.

[SASL] Myers, J., "Simple Authentication and Security Layer (SASL)", RFC 2222, October 1997.

[TLS] Dierks, T. and C. Allen, "The TLS Protocol Version 1.0", RFC 2246, January 1999.

Contributors

Michael Peddemors
LinuxMagic

Authors' Addresses

Deion Yu
LinuxMagic
#405 - 860 Homer St.
Vancouver, British Columbia
CA V6B 2W5

EMail: deiony@linuxmagic.com