                            POP URL Scheme

Status of this Memo

   This document specifies an Internet standards track protocol for the
   Internet community, and requests discussion and suggestions for
   improvements. Please refer to the current edition of the "Internet
   Official Protocol Standards" (STD 1) for the standardization state
   and status of this protocol. Distribution of this memo is unlimited.

1. Introduction

   [POP3] is a widely-deployed mail access protocol. Many programs
   access POP3 message stores, and thus need POP3 configuration
   information. Since there are multiple configuration elements which
   are required in order to access a mailbox, a single string
   representation is convenient.

   A POP3 mailbox (like an [IMAP4] mailbox) is a network resource, and
   URLs are a widely-supported generalized representation of network
   resources.

   A means of specifying a POP3 mailbox as a URL will likely be useful
   in many programs and protocols. [ACAP] is one case where a string
   encapsulation of elements required to access network services is
   needed. For example, an [IMAP4] message store is usually specified
   in ACAP datasets as an [IMAP-URL].

   This memo defines a URL scheme for referencing a POP mailbox.

2. Conventions Used in this Document

   The key words "MUST", "MUST NOT", "SHOULD", "SHOULD NOT", and "MAY"
   in this document are to be interpreted as defined in "Key words for
   use in RFCs to Indicate Requirement Levels" [KEYWORDS].

3. POP Scheme

   The POP URL scheme designates a POP server, and optionally a port
   number, authentication mechanism, authentication ID, and/or
   authorization ID.

   The POP URL follows the common Internet scheme syntax as defined in
   RFC 1738 [BASIC-URL] except that clear text passwords are not
   permitted. If :<port> is omitted, the port defaults to 110.

   The POP URL is described using [ABNF] in Section 8.

   A POP URL is of the general form:

        pop://<user>;auth=<auth>@<host>:<port>

   Where <user>, <host>, and <port> are as defined in RFC 1738, and some
   or all of the elements, except "pop://" and <host>, may be omitted.

4. POP User Name and Authentication Mechanism

   An authorization (which mailbox to access) and authentication (whose
   password to check against) identity (referred to as "user name" for
   simplicity) and/or authentication mechanism name may be supplied.
   These are used in a "USER", "APOP", "AUTH" [POP-AUTH], or extension
   command after making the connection to the POP server. If the URL
   doesn't supply an authentication identifier, the program interpreting
   the POP URL SHOULD request one from the user.

   An authentication mechanism can be expressed by adding ";AUTH=<enc-
   auth-type>" to the end of the user name. If the authentication
   mechanism name is not preceded by a "+", it is a SASL POP [SASL]
   mechanism. If it is preceded by a "+", it is either "APOP" or an
   extension mechanism.

   When an <enc-auth-type> is specified, the client SHOULD request
   appropriate credentials from that mechanism and use the "AUTH",
   "APOP", or extension command instead of the "USER" command. If no
   user name is specified, one SHOULD be obtained from the mechanism or
   requested from the user as appropriate.

   The string ";AUTH=*" indicates that the client SHOULD select an
   appropriate authentication mechanism. It MAY use any mechanism
   supported by the POP server.

   If an <enc-auth-type> other than ";AUTH=*" is specified, the client
   SHOULD NOT use a different mechanism without explicit user
   permission.

If a user name is included with no authentication mechanism, then
";AUTH=*" is assumed.

Since URLs can easily come from untrusted sources, care must be taken
when resolving a URL which requires or requests any sort of
authentication. If authentication credentials are supplied to the
wrong server, it may compromise the security of the user's account.
The program resolving the URL should make sure it meets at least one
of the following criteria in this case:

(1) The URL comes from a trusted source, such as a referral server
which the client has validated and trusts according to site policy.
Note that user entry of the URL may or may not count as a trusted
source, depending on the experience level of the user and site
policy.

(2) Explicit local site policy permits the client to connect to the
server in the URL. For example, if the client knows the site domain
name, site policy may dictate that any hostname ending in that domain
is trusted.

(3) The user confirms that connecting to that domain name with the
specified credentials and/or mechanism is permitted.

(4) A mechanism is used which validates the server before passing
potentially compromising client credentials.

(5) An authentication mechanism is used which will not reveal
information to the server which could be used to compromise future
connections.

A URL containing ";AUTH=*" should be treated with extra care since it
might fall back on a weaker security mechanism. Finally, clients are
discouraged from using a plain text password as a fallback with
";AUTH=*" unless the connection has strong encryption (e.g., a key
length of greater than 56 bits).

Note that if unsafe or reserved characters such as " " or ";" are
present in the user name or authentication mechanism, they MUST be
encoded as described in RFC 1738 [BASIC-URL].

5. Relative POP URLs

Relative POP URLs are not permitted.

6. Multinational Considerations

   Since 8-bit characters are not permitted in URLs, [UTF8] characters
   are encoded as required by the URL specification [BASIC-URL].

7. Examples

   The following examples demonstrate how a POP client program might
   translate various POP URLs into a series of POP commands. Commands
   sent from the client to the server are prefixed with "C:", and
   responses sent from the server to the client are prefixed with "S:".

   The URL:

        <pop://rg@mailsrv.qualcomm.com>

   Results in the following client commands:

        <request password from user>
        <connect to mailsrv.qualcomm.com, port 110>
        S: +OK POP3 server ready <1896.697170952@mailsrv.qualcomm.com>
        C: USER rg
        S: +OK
        C: PASS secret
        S: +OK rg's mailbox has 2 messages (320 octets)

   The URL:

        <pop://rg;AUTH=+APOP@mail.eudora.com:8110>

   Results in the following client commands:

        <client requests password from user>
        <connect to mail.eudora.com, port 8110>
        S: +OK POP3 server ready <1896.697170952@mail.eudora.com>
        C: APOP rg c4c9334bac560ecc979e58001b3e22fb
        S: +OK mailbox has 1 message (369 octets)

   The URL:

        <pop://baz;AUTH=SCRAM-MD5@foo.bar>

   Results in the following client commands:

        <connect to foo.bar, port 110>

        S: +OK POP3 server ready <1896.697170952@foo.bar>
        C: AUTH SCRAM-MD5 AGNocmlzADx0NG40UGFiOUhCMEFtL1FMWEI3MmVnQGVsZW

```
          Fub3IuaW5ub3NvZnQuY29tPg==
     S: + dGVzdHNhbHQBAAAAaW1hcEBlbGVhbm9yLmlubm9zb2Z0LmNvbQBq
        aGNOWmxSdVBiemlGcCt2TFYrTkN3
     C: AQAAAMg9jU8CeB4KOfk7sUhSQPs=
     S: + U0odqYw3B7XIIW0oSz65OQ==
     C:
     S: +OK mailbox has 1 message (369 octets)
```

8. ABNF for POP URL scheme

   The POP URL scheme is described using [ABNF]:

```
     achar               = uchar / "&" / "=" / "~"
                              ; see [BASIC-URL] for "uchar" definition

     auth                = ";AUTH=" ( "*" / enc-auth-type )

     enc-auth-type       = enc-sasl / enc-ext

     enc-ext             = "+" ("APOP" / 1*achar)
                             ;APOP or encoded extension mechanism name

     enc-sasl            = 1*achar
                              ;encoded version of [SASL] "auth_type"

     enc-user            = 1*achar
                              ;encoded version of [POP3] mailbox

     pop-url             = "pop://" server

     server              = [user-auth "@"] hostport
                              ;See [BASIC-URL] for "hostport" definition

     user-auth           = enc-user [auth]
```

9. Security Considerations

   Security considerations discussed in the [POP3] specification and the
   [BASIC-URL] specification are relevant. Security considerations
   related to authenticated URLs are discussed in section 4 of this
   document.

   Many email clients store the plain text password for later use after
   logging into a POP server. Such clients MUST NOT use a stored
   password in response to a POP URL without explicit permission from
   the user to supply that password to the specified host name.

10. Acknowledgements

   This document borrows heavily from Chris Newman's [IMAP-URL]
   specification, and has attempted to follow the advice in [URL-
   GUIDELINES].

11. References

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

   [ACAP]   Newman, C., and J. Myers, "ACAP -- Application
                     Configuration Access Protocol", RFC 2244, November
                     1997.

   [BASIC-URL]   Berners-Lee, T., Masinter, L., and M. McCahill,
                     "Uniform Resource Locators (URL)", RFC 1738,
                     December 1994.

   [IMAP-URL]   Newman, C., "IMAP URL Scheme", RFC 2192, September
                     1997.

   [IMAP4]   Crispin, M., "Internet Message Access Protocol -
                     Version 4rev1", RFC 2060, December 1996.

   [KEYWORDS]   Bradner, S., "Key words for use in RFCs to Indicate
                     Requirement Levels", BCP 14, RFC 2119, March 1997.

   [POP-AUTH]   Myers, J., "POP3 AUTHentication command", RFC 1734,
                     December 1994.

   [POP3]   Myers, J., and M. Rose, "Post Office Protocol --
                     Version 3", STD 53, RFC 1939, May 1996.

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

   [URL-GUIDELINES] Masinter, Alvestrand, Zigmond, "Guidelines for new
                     URL Schemes", Work in Progress.

   [UTF8]   Yergeau, F., "UTF-8, a transformation format of ISO
                     10646", RFC 2279, January 1998.

12. Author's Address

   Randall Gellens
   QUALCOMM, Incorporated
   6455 Lusk Blvd.
   San Diego, CA 92121-2779
   U.S.A.

   Phone: +1 619 651 5115
   Fax: +1 619 651 5334
   EMail: Randy@Qualcomm.Com

13. Full Copyright Statement