Network Working Group
Request for Comments: 2245
Category: Standards Track

C. Newman
Innosoft

November 1997

Anonymous SASL Mechanism

Status of this Memo

Copyright Notice

Abstract

It is common practice on the Internet to permit anonymous access to various services. Traditionally, this has been done with a plain text password mechanism using "anonymous" as the user name and optional trace information, such as an email address, as the password. As plaintext login commands are not permitted in new IETF protocols, a new way to provide anonymous login is needed within the context of the SASL [SASL] framework.

1. 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].

2. Anonymous SASL mechanism

The mechanism name associated with anonymous access is "ANONYMOUS". The mechanism consists of a single message from the client to the server. The client sends optional trace information in the form of a human readable string. The trace information should take one of three forms: an Internet email address, an opaque string which does not contain the '@' character and can be interpreted by the system administrator of the client's domain, or nothing. For privacy reasons, an Internet email address should only be used with permission from the user.

A server which permits anonymous access will announce support for the ANONYMOUS mechanism, and allow anyone to log in using that mechanism, usually with restricted access.

The formal grammar for the client message using Augmented BNF [ABNF] follows.

```
message         = [email / token]

TCHAR           = %x20-3F / %x41-7E
                  ;; any printable US-ASCII character except '@'

email           = addr-spec
                  ;; as defined in [IMAIL], except with no free
                  ;; insertion of linear-white-space, and the
                  ;; local-part MUST either be entirely enclosed in
                  ;; quotes or entirely unquoted

token           = 1*255TCHAR
```

3. Example

Here is a sample anonymous login between an IMAP client and server. In this example, "C:" and "S:" indicate lines sent by the client and server respectively. If such lines are wrapped without a new "C:" or "S:" label, then the wrapping is for editorial clarity and is not part of the command.

Note that this example uses the IMAP profile [IMAP4] of SASL. The base64 encoding of challenges and responses, as well as the "+ " preceding the responses are part of the IMAP4 profile, not part of SASL itself. Newer profiles of SASL will include the client message with the AUTHENTICATE command itself so the extra round trip below (the server response with an empty "+ ") can be eliminated.

In this example, the user's opaque identification token is "sirhc".

```
S: * OK IMAP4 server ready
C: A001 CAPABILITY
S: * CAPABILITY IMAP4 IMAP4rev1 AUTH=CRAM-MD5 AUTH=ANONYMOUS
S: A001 OK done
C: A002 AUTHENTICATE ANONYMOUS
S: +
C: c2lyaGM=
S: A003 OK Welcome, trace information has been logged.
```

4. Security Considerations

The anonymous mechanism grants access to information by anyone. For this reason it should be disabled by default so the administrator can make an explicit decision to enable it.

If the anonymous user has any write privileges, a denial of service attack is possible by filling up all available space. This can be prevented by disabling all write access by anonymous users.

If anonymous users have read and write access to the same area, the server can be used as a communication mechanism to anonymously exchange information. Servers which accept anonymous submissions should implement the common "drop box" model which forbids anonymous read access to the area where anonymous submissions are accepted.

If the anonymous user can run many expensive operations (e.g., an IMAP SEARCH BODY command), this could enable a denial of service attack. Servers are encouraged to limit the number of anonymous users and reduce their priority or limit their resource usage.

If there is no idle timeout for the anonymous user and there is a limit on the number of anonymous users, a denial of service attack is enabled. Servers should implement an idle timeout for anonymous users.

The trace information is not authenticated so it can be falsified. This can be used as an attempt to get someone else in trouble for access to questionable information. Administrators trying to trace abuse need to realize this information may be falsified.

A client which uses the user's correct email address as trace information without explicit permission may violate that user's privacy. Information about who accesses an anonymous archive on a sensitive subject (e.g., sexual abuse) has strong privacy needs. Clients should not send the email address without explicit permission of the user and should offer the option of supplying no trace token -- thus only exposing the source IP address and time. Anonymous proxy servers could enhance this privacy, but would have to consider the resulting potential denial of service attacks.

Anonymous connections are susceptible to man in the middle attacks which view or alter the data transferred. Clients and servers are encouraged to support external integrity and encryption mechanisms.

Protocols which fail to require an explicit anonymous login are more susceptible to break-ins given certain common implementation techniques. Specifically, Unix servers which offer user login may

initially start up as root and switch to the appropriate user id after an explicit login command. Normally such servers refuse all data access commands prior to explicit login and may enter a restricted security environment (e.g., the Unix chroot function) for anonymous users. If anonymous access is not explicitly requested, the entire data access machinery is exposed to external security attacks without the chance for explicit protective measures. Protocols which offer restricted data access should not allow anonymous data access without an explicit login step.

5. References

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

[IMAIL] Crocker, D., "Standard for the Format of Arpa Internet Text Messages", STD 11, RFC 822, August 1982.

[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", RFC 2119, March 1997.

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

6. Author’s Address

Chris Newman
Innosoft International, Inc.
1050 Lakes Drive
West Covina, CA 91790 USA

Email: chris.newman@innosoft.com

7. Full Copyright Statement