Network Working Group
INTERNET-DRAFT
<draft-aboba-ppp-01.txt>
19 November 1997

Bernard Aboba
Microsoft

Lightweight Directory Access Protocol (v3): Extension for PPP Authentication

1. Status of this Memo

This document is an Internet-Draft. Internet-Drafts are working documents of the Internet Engineering Task Force (IETF), its areas, and its working groups. Note that other groups may also distribute working documents as Internet-Drafts.

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.''

To learn the current status of any Internet-Draft, please check the ``1id-abstracts.txt'' listing contained in the Internet-Drafts Shadow Directories on ds.internic.net (US East Coast), nic.nordu.net (Europe), ftp.isi.edu (US West Coast), or munnari.oz.au (Pacific Rim).

The distribution of this memo is unlimited. It is filed as <draft-aboba-ppp-01.txt>, and expires May 1, 1998. Please send comments to the authors.

2. Abstract

This document defines the ''PPP Authentication Operation'' for LDAP. This operation provides for PPP authentication in an LDAP association and is defined in terms of an LDAP extended operation.

It is expected that this extended operation will be useful in integrating authentication protocols such as RADIUS and TACACS+ with LDAP-based directory services. Consolidation of information stores is desirable since it results in lessened administrative workload and a consistent view of user information throughout the enterprise.

3. Introduction

Currently RADIUS (described in [6]-[8]) and TACACS+ (described in [12]) authentication servers typically include their own stores of user data. In order to simplify user administration, it is desirable to be able to integrate these services with an LDAP-based directory service.

This document is one of three related specifications which describe how a RADIUS server may be integrated with an LDAP-based directory service. Reference [16] specifies how user data utilized by a RADIUS server may be stored in an LDAP-based directory service. Reference [17] describes a schema designed for tracking sessions in progress. Such information can be useful for a variety of purposes including security incident response; simultaneous usage control; or monitoring of connection quality, login time, packet size or bandwidth usage. Due to the frequency of changes to this data, dynamic attributes must be employed, as described in [5].

PPP authentication protocols are described in [11],[14] and [15]. This document describes an LDAP extension supporting validation of user credentials submitted during PPP authentication. This makes it possible for the RADIUS server to validate user credentials received in the Access-Request packet.

3.1. Alternatives for integration of PPP authentication methods

In order for a RADIUS server to be able to respond to an Access-Request, a means must be available for validating user credentials. However, current LDAP security mechanisms do not support PPP authentication methods so that extensions or protocol modifications are required.

Several alternatives present themselves. One alternative is to add support for PPP authentication methods to SASL, and utilize the secure BIND mechanisms described in [18]. In this alternative, the RADIUS server will impersonate the user and bind using the credentials submitted in the Access-Request. In this scenario, only the user would need to have the access rights to retrieve RADIUS attributes from the directory. There would not be a need to make these attributes accessible to a privileged account used by the RADIUS server, or to any network devices. This is desirable from a security point of view.

Using this alternative, support for PPP authentication methods would be added to SASL. The RADIUS server would set up an SSL/TLS connection on startup, and would execute a BIND operation for each authentication; the server would only UNBIND on shutdown. This avoids the overhead of an UNBIND and SSL/TLS connection setup for each authentication.

However, it should be noted that merely adding CHAP support to SASL will not solve the problem posed here. This is because an LDAP-server utilizing a CHAP extension to SASL would generate its own challenge, rather than accepting a CHAP challenge and response submitted to it by the RADIUS server. While such an approach would be compatible with

EAP-MD5, where the RADIUS server generates its own challenge, it would not be compatible with CHAP, where the NAS generates the challenge and passes both the CHAP challenge and the response to the RADIUS server for evaluation.

Thus to solve the problem, it must be possible to submit both the CHAP challenge and response to SASL. However, making it possible to authenticate to an LDAP server using such a mechanism is not desirable since it would make LDAP authentication susceptible to a replay attack.

Another alternative is to provide support for PPP authentication within an LDAP extended operation. In this alternative, the RADIUS server binds to the directory on startup using a special account, and unbinds on shutdown. In between the bind and unbind, the RADIUS server may submit as many PPP authentication requests as necessary. In this scenario, the account used by the RADIUS server needs to have the access rights to retrieve RADIUS attributes for any user.

Since the LDAP extension only returns a yes or no, but does not gain the requester any privileges, it does not have the security problem inherent in the SASL-based scheme described above. It is also believed that this approach will utilize fewer resources on most implementations, since the continual execution of BIND operations, without corresponding UNBINDs, is likely to result in steady memory consumption on the RADIUS server.

3.2. Overview

PPP Authentication is an extended operation initiated by an LDAP client (RADIUS server) in order to request authentication of a user by the LDAP-based directory. The LDAP client sends a PPP Authentication request to the LDAP server, indicating the PPP authentication method, and including the user's credentials, and the server then responds with a message indicating the success or failure of the authentication.

When the RADIUS server receives an Access-Request packet from a NAS or VPN server, it examines the User-Name attribute to determine the user that is being authenticated. Based on the User-Name, the server may also retrieve the authenticationType attribute for the user, and will then check the authentication method specified in the Access-Request against the permitted types. If there is a mis-match, then the server will formulate and send an Access-Reject packet.

If the authentication indicated in the Access-Request is one of the permitted types, and PAP or CHAP authentication is being used, the RADIUS server utilizes the LDAP extension for PPP authentication specified in this document in order to verify the user's identity. Alternatively, the PPP authentication operation may be carried out synchronously with retrieval of the RADIUS attributes described in [16], and an Access-Reject can be sent if an authentication type mismatch is detected after the retrieval (and possibly the PPP authentication

operation) is complete.

If the user authentication is unsuccessful, then the RADIUS server will formulate and send an Access-Reject packet. If the user is successfully authenticated, then the RADIUS server will formulate an Access-Accept based on the attributes retreived from the LDAP-based

directory service, specified in [16].

If the Access-Request contains an EAP-Message attribute with a specified identity, then the RADIUS server will retrieve the user's RADIUS-related information from the LDAP-based directory service in order to determine the type of EAP authentication for this user. Depending on the eapType, the RADIUS server will then either handle the authentication internally (such as for MD5), or may forward the request to a security server. As a result, the PPP authentication operation described in this document does not need to support EAP.

4. Protocol Additions

4.1. The Start PPP Authentication Operation

A client may perform a PPP authentication operation by transmitting an LDAP PDU containing an ExtendedRequest. An LDAP ExtendedREquest is defined as follows:

```
ExtendedRequest ::= [APPLICATION 23] SEQUENCE {
                requestName             [0] LDAPOID,
                requestValue            [1] OCTET STRING }
```

The requestName field must be set to the string "<OID-to-be-assigned>".

This request is permitted to be invoked when LDAP is carried by a connectionless transport.

When using a connection-oriented transport, there is no requirement that this operation be on the same particular connection as any other. A client may open multiple connections, or close and then reopen a connection.

4.1.1. CHAP Authentication

When Challenge-Handshake Authentication Protocol (CHAP) authentication is desired, the requestValue field will contain as a value the DER-encoding of the following ASN.1 data type:

```
        SEQUENCE {
                authenticationProtocol  [0] INTEGER,
                algorithm               [1] INTEGER,
                name                    [2] OCTET STRING,
                challenge               [3] OCTET STRING,
                chapIdent               [4] OCTET STRING,
```

```
            response                [5] OCTET STRING
        }
```

The authenticationProtocol field is an integer containing the Authentication-Protocol number for CHAP, c223 (hex). The algorithm is an

integer indicating the one-way hash method to be used. Values include MD5 (5). The name is an octet string identifying the user to be authenticated. The challenge is a 16 octet string containing the CHAP challenge sent by the NAS to a PPP CHAP user. The chapIdent is a single octet containing the CHAP Identifer from the user's CHAP Response. The response is a 16 octet field containing the CHAP Response from the user.

4.1.2. PAP Authentication

When Password Authentication Protocol (PAP) authentication is desired, the requestValue field will contain as a value the DER-encoding of the following ASN.1 data type:

```
SEQUENCE {
        authenticationProtocol  [0] INTEGER,
        name                    [1] OCTET STRING,
        password                [2] OCTET STRING
}
```

The authenticationProtocol field is an integer containing the Authentication-Protocol number for PAP, c023 (hex). The name is an octet string identifying the user to be authenticated. The password is an octet string providing the user's password.

4.2. PPP Authentication Response

If a server implements this extension, then when the request is made it will return an LDAP PDU containing an ExtendedResponse. An LDAP ExtendedResponse is defined as follows:

```
ExtendedResponse ::= [APPLICATION 24] SEQUENCE {
        responseName            [0] LDAPOID OPTIONAL,
        response                [1] OCTET STRING OPTIONAL
        standardResponse        [2] LDAPResult }
```

The responseName field contains the same string as that present in the PPP Authentication request. The response field is absent. The server MUST set the resultCode of the standardResponse to either success or one of the other values outlined below.

4.3. Error Messages

If the operation was successful, the errorCode field in the standardResponse part of an ExtendedResponse will be set to success.

In case of an error, the errorCode field will contain an appropriate value. If the authentication is not successful (due either to invalid credentials or an invalid userName), the errorCode field will contain the invalidCredentials result code. If the authentication protocol

given by authenticationProtocol could not be located, the errorCode field will contain the protocolError result code. If the authentication protocol is not permitted, the errorCode field will contain strongAuthRequired. If the requester does not have permission to perform the PPP authentication, the errorCode field will contain insufficientAccessRights. If the server does not do PPP authentication, but knows another server that does, the errorCode field will contain referral. If There is a major problem with PPP authentication, or the server is shutting down the errorCode field will contain unavailable. If the server is overloaded, the errorCode field will contain busy.

If a server does not implement this extension, it will return an LDAP PDU containing an ExtendedResponse, which contains only the standardResponse element (the responseName and response elements will be absent). The LDAPResult element will indicate the protocolError result code.

5. Security considerations

Enabling an LDAP-based directory service to perform PPP authentication operations in an efficient manner may result in a number of security threats, including password guessing attacks and sniffing attacks.

In order to prevent a rogue RADIUS server from initiating password guessing attacks, it is desirable for an implementation to close a connection after a number of consecutive authentication failures.

In order to prevent sniffing of passwords, where PAP authentication is being used for transmission of cleartext passwords, the RADIUS server MUST seek to ensure confidentiality by using SSL/TLS or IPSEC. An LDAP server receiving a PAP authentication request representing a cleartext password without confidentiality services in place MUST return an error message.

6. Acknowledgments

Thanks to Gurdeep Singh Pall and Narendra Gidwani of Microsoft for useful discussions of this problem space.

7. References

[1] W. Yeong, T. Howes, S. Kille, "Lightweight Directory Access Protocol." RFC 1777, March 1995.

[2] "Information Processing Systems - Open Systems Interconnection - The Directory: Overview of Concepts, Models and Service." ISO/IEC JTC

1/SC21, International Standard 9594-1, 1988.

[3] "Information Processing Systems - Open Systems Interconnection - The Directory: Selected Object Classes." Recommendation X.521 ISO/IEC JTC 1/SC21, International Standard 9594-7, 1993.

[4] M. Wahl, A. Coulbeck, T. Howes, S. Kille, "Lightweight Directory Access Protocol (v3): Attribute Syntax Definitions. " Internet Draft (work in progress), draft-ietf-asid-ldapv3-attributes-08.txt, Critical Angle, Isode, Netscape, October 1997.

[5] Y. Yaacovi, M. Wahl, T. Genovese, "Lightweight Directory Access Protocol (v3): Extensions for Dynamic Directory Services. " Internet Draft (work in progress), draft-ietf-asid-ldapv3-dynamic-06.txt, Microsoft, Critical Angle, September 1997.

[6] C. Rigney, A. Rubens, W. Simpson, S. Willens. "Remote Authentication Dial In User Service (RADIUS)." RFC 2138, Livingston, Merit, Daydreamer, April 1997.

[7] C. Rigney. "RADIUS Accounting." RFC 2139, Livingston, April 1997.

[8] C. Rigney, W. Willats. "RADIUS Extensions." Work in progress, draft-ietf-radius-ext-01.txt, Livingston, June 1997.

[9] R. Rivest, S. Dusse. "The MD5 Message-Digest Algorithm", RFC 1321, MIT Laboratory for Computer Science, RSA Data Security Inc., April 1992.

[10] P. Calhoun, A. C. Rubens, B. Aboba. "Extensible Authentication Protocol Support in RADIUS." Internet Draft (work in progress), April, 1997, draft-ietf-radius-eap-02.txt, 3Com, Merit Network, Microsoft.

[11] L. J. Blunk, J. R. Vollbrecht. "PPP Extensible Authentication Protocol (EAP)." Work in progress, draft-ietf-pppext-eap-auth-02.txt, Merit Network, Inc., June, 1996.

[12] D. Carrel, L. Grant. "The TACACS+ Protocol Version 1.77." Work in progress, draft-grant-tacacs-01.txt, Cisco Systems, October, 1996.

[13] Simpson, W., Editor. "The Point-to-Point Protocol (PPP)", STD 51, RFC 1661, DayDreamer, July 1994.

[14] Simpson, W. "PPP Challenge Handshake Authentication Protocol (CHAP)", RFC 1994, DayDreamer, August 1996.

[15] B. Lloyd, Simpson, W. "PPP Authentication Protocols", RFC 1334, L&A, DayDreamer, October 1992.

[16] B. Aboba, "Lightweight Directory Access Protocol (v3): Schema for the Remote Access Dialin User Service (RADIUS) " Internet Draft (work in progress), draft-aboba-radius-01.txt, Microsoft, November 1997.

[17] B. Aboba, "Lightweight Directory Access Protocol (v3): Dynamic Attributes for the Remote Access Dialin User Service (RADIUS)" Internet Draft (work in progress), draft-aboba-dynradius-01.txt, Microsoft, November 1997.

[18] M. Wahl, T. Hoews, S. Kille, "Lightweight Directory Access Protocol (v3)." Internet Draft (work in progress), draft-ietf-asid-protocol-08.txt, Critical Angle, Netscape, Isode, October 1997.

[19] J. Hodges, R.L. Morgan, M. Wahl, "Lightweight Directory Access Protocol (v3): Extension for Transport Layer Security." Internet Draft (work in progress), draft-ietf-asid-ldapv3-tls-01.txt, Stanford, Critical Angle, June 1997.

[20] Y. Yaacovi, M. Wahl, T. Genovese, "Lightweight Directory Access Protocol: Dynamic Attributes." Internet Draft (work in progress), draft-ietf-asid-dynatt-00.txt, Microsoft, Critical Angle, July 1997.

8. Authors' Addresses

Bernard Aboba
Microsoft Corporation
One Microsoft Way
Redmond, WA 98052

Phone: 425-936-6605
EMail: bernarda@microsoft.com