Internet Engineering Task Force
IP Secure Remote Access Working Group
Internet Draft
Expires May 2000

R. Pereira
Cisco Systems
S. Beaulieu
TimeStep Corporation
December 1999

Extended Authentication within ISAKMP/Oakley (XAUTH)

<draft-ietf-ipsec-isakmp-xauth-06.txt>

Status of this Memo

Abstract

This document describes a method for using existing unidirectional authentication mechanisms such as RADIUS, SecurID, and OTP within IPSec's ISAKMP protocol. The purpose of this draft is not to replace or enhance the existing authentication mechanisms described in [IKE], but rather to allow them to be extended using legacy authentication mechanisms.

Table of Contents

1 Introduction

The following technique allows IPSec's ISAKMP/Oakley [IKE] protocol to support extended authentication mechanisms like two-factor

authentication, challenge/response and other remote access unidirectional authentication methods.

These authentication mechanisms have a large deployment in remote access applications and many IT departments have requirements for these unidirectional authentication mechanisms.

1.1 Changes Since Last Revision

o The last revision of this document was published in the IPSec Working Group as
<draft-ietf-ipsec-isakmp-xauth-05.txt>

o Moved XAUTH Attribute ID numbers to private range of Isakmp-Config draft to avoid future collisions.

o Added a Feature / Vendor ID.

o Removed all of the authentication types which can use Generic.

o Made XAUTH_TYPE optional, with the default set to Generic if not present.

o Clarified the text which allows for a remote peer to abort in the middle of a transaction.

o Expanded on Security Considerations.

1.2 Extended Authentication

Two-factor authentication and challenge/response schemes like SDI's SecurID and RADIUS are forms of authentication that allow a gateway, firewall, or network access server to offload the user administration and authentication to a central management server. IPSec's ISAKMP/Oakley protocol supports certificates (RSA & DSS), shared-secret, and Kerberos as authentication methods, but since the authentication methods described within this document are only unidirectional authentication methods (client to a gateway/firewall), they cannot be used by themselves, but must be used in conjunction with the other standard ISAKMP authentication methods.

The technique described within this document utilizes ISAKMP to transfer the user's authentication information (name, password) to the gateway/firewall (edge device) in a secured ISAKMP message. The edge device would then use either the appropriate protocol (RADIUS,

SecurID, OTP) to authenticate the user. This allows the authentication server to be within the private network that the edge device is protecting.

1.3 Reader Prerequisites

It is assumed that the reader is familiar with the terms and concepts described in the "Security Architecture for the Internet Protocol" [ArchSec] and "IP Security Document Roadmap" [Thayer97] documents.

Readers are advised to be familiar with both [IKE] and [ISAKMP] as well as [IKECFG] since this document is an extension to that document.

1.4 Specification of Requirements

The keywords "MUST", "MUST NOT", "REQUIRED", "SHOULD", "SHOULD NOT", and "MAY" that appear in this document are to be interpreted

as described in [Bradner97].

2 Vendor ID

XAUTH currently uses attribute numbers from the private ranges of both [IKE] and [IKECFG]. In order to ensure interoperability with future and past implementations of XAUTH a Vendor ID has been added. The Vendor ID payload is sent during the phase 1 exchange as per [ISAKMP]. The Vendor ID for this revision of XAUTH is a truncated MD5 hash of the following ASCII text string: "draft-ietf-ipsra-isakmp-xauth-06.txt" without the quotation marks.

Vendor ID = 0x09002689DFD6B712

If an implementation receives the aforementioned Vendor ID, it can assume that the peer also has implemented this protocol and therefore is a "mutually consenting party".

If this document advances to the standard-track, then new numbers will be assigned by IANA from the appropriate number spaces of [IKE] and [IKECFG], thus eliminating the need for a Vendor ID payload.

3 Extended Authentication Method

This specification allows for extended authentication by allowing an edge device to request extended authentication from an IPSec

host (end-node), thus forcing the host to respond with its extended authentication credentials. The edge device will then respond with a failed or passed message.

When the edge device requests extended authentication, it will specify the type of extra authentication and any parameters required for it. These parameters MAY be the attributes that it requires for authentication and they MAY be information required for the IPSec host's reply (e.g. challenge string).

The Extended Authentication transaction is terminated either when

the edge device starts a SET/ACK exchange which includes an XAUTH_STATUS attribute or when the remote device sends a XAUTH_STATUS attribute in a REPLY message. Please note that a remote device can not set XAUTH_STATUS to anything but FAIL.

The edge device MAY request multiple different authentication transactions within one Extended Authentication transaction. This is done by having multiple REQUEST/REPLY pairs, initiated by the edge device, before the transaction is terminated as described above. Each REQUEST/REPLY pair MAY have a different value for XAUTH_TYPE.

As with CHAP [CHAP], this protocol can also be used to periodically authenticate the user during the lifetime of a security association.

If the IPSec host does not have support for the authentication method requested by the edge device, then it would send back a REPLY with the XAUTH_STATUS attribute set to FAIL, thus failing the authentication but completing the transaction.

The Extended Authentication mechanism does not effect the nature of the phase 1 authentication mechanism in any way. Both peers MUST authenticate each other via the authentication methods described in [IKE]. There are Security Considerations involved in one of the authentication methods in [IKE] and this is described in "Security Considerations" below.

This method provides unidirectional authentication only, meaning that only one device is authenticated using both IKE authentication methods and Extended Authentication.

Here are some types of extended authentication that this specification supports:

3.1 Simple Authentication

Where a user name and password are required for authentication.

IPSec Host Edge Device

```
--------------                                    -----------------
                                    <-- REQUEST(NAME="" PASSWORD="")
REPLY(NAME="joe" PASSWORD="foobar") -->
                                                  <-- SET(STATUS=OK)
ACK(STATUS) -->
```

Some authentication mechanisms hide the user password by some type of encryption mechanism.

```
IPSec Host                                            Edge Device
--------------                                    -----------------
                    <-- REQUEST(TYPE=RADIUS-CHAP CHALLENGE="123456"
                                                    NAME="" PASSWORD="")
REPLY(TYPE=RADIUS-CHAP NAME="joe" PASSWORD="E4901AB7") -->
                                                  <-- SET(STATUS=OK)
ACK(STATUS) -->
```

NOTE: This is a conceptual example of RADIUS-CHAP, for a more detailed example, see Appendix A.

3.2 Challenge/Response

Where a challenge from the edge device must be incorporated with the reply. This makes each reply different.

```
IPSec Host                                            Edge Device
--------------                                    -----------------
                                    <-- REQUEST(NAME="" PASSWORD="")
REPLY(NAME="joe" PASSWORD="foobar") -->
                <-- REQUEST(MESSAGE="Enter your password followed by
                                  your pin number" NAME="" PASSWORD="")
REPLY(NAME="joe" PASSWORD="foobar0985124") -->
                                                  <-- SET(STATUS=OK)
ACK(STATUS) -->
```

If, however, the edge device knows that a challenge will be required it may skip the first exchange as follows:

```
   IPSec Host                                              Edge Device
   --------------                                         -----------------
                  <-- REQUEST(MESSAGE="Enter your password followed by
                                  your pin number" NAME="" PASSWORD="")
   REPLY(NAME="joe" PASSWORD="foobar0985124") -->
                                                          <-- SET(STATUS=OK)
   ACK(STATUS) -->
```

3.3 Two-Factor Authentication

This authentication method combines something the user knows (their password) and something that the user has (a token card).

```
   IPSec Host                                               Edge Device
   --------------                                          -----------------
                                      <-- REQUEST(NAME=""
                                              PASSWORD="" PASSCODE="")
   REPLY(NAME="joe"
    PASSWORD="foobar" PASSCODE="3412") -->
                                                          <-- SET(STATUS=OK)
   ACK(STATUS) -->
```

Some mechanisms allow for another optional request of the passcode.

```
   IPSec Host                                              Edge Device
   --------------                                         -----------------
                           <-- REQUEST(NAME="" PASSWORD="" PASSCODE="")
   REPLY(NAME="joe" PASSWORD="foobar" PASSCODE="323415") -->
                           <-- REQUEST(NAME="" PASSWORD="" PASSCODE="")
   REPLY(NAME="joe" PASSWORD="foobar" PASSCODE="513212") -->
                                                          <-- SET(STATUS=OK)
   ACK(STATUS) -->
```

3.4 One-Time-Password

Similar to the Challenge/Response method, this method allows authentication that is secure against passive attacks based on replaying captured passwords.

```
   IPSec Host                                              Edge Device
   --------------                                         -----------------
                   <-- REQUEST(TYPE=OTP CHALLENGE="otp-md5 499 ke1234"
                                                      NAME="" PASSWORD="")
   REPLY(TYPE=OTP NAME="joe" PASSWORD="5bf0 75d9 959d 036f") -->
```

3.5 User Previously Authenticated

Some situations may occur where the edge device has already authenticated the host and no new authentication is required. This may happen when either the host or the edge device must rekey an existing phase 1 SA. It is important that this method not be used, unless the implementation can be sure that the current phase 1 SA was created with the same peer as the initial phase 1 SA, which was previously authenticated using XAUTH. There is currently no way defined to ensure that two separate phase 1 SAs actually belong to the same peer. One method suggested is to use the ID from the phase 1 negotiation (available in Main Mode and Aggressive Mode) but only if the ID is unique to the user and cannot not be forged. This concept is herein referred to as "ID-Checking".

Implementation hint:

- o In order to accomplish ID-Checking for Phase 1 Authenticated With a Pre-Shared Key (as defined in [IKE]), the pre-shared key lookup must be based on the phase 1 ID. Please note that this method only currently works for Aggressive Mode, and may work with modes defined in the future. A static IP address could also be used for shared secret lookup, however, the binding of the user to XAUTH session would have to use the IP address instead of the ID.

- o In order to accomplish ID-Checking for IKE Phase 1 Authenticated With Signatures (as defined in [IKE]), the implementation must ensure that the ID provided in the phase 1 exchange matches the ID in the peer's certificate which must be signed by a trusted third party.

In the situation where the peer does not require additional authentication, the following method is used.

```
IPSec Host                                          Edge Device
-------------                                      ----------------
```

3.6 Other Useful Examples

More useful examples are found in Appendix A.

4 Extensions to ISAKMP-Config

This protocol uses the mechanisms described in ISAKMP-Config [IKECFG] to accomplish its authentication transaction. This protocol uses Configuration Attributes from the private range of Isakmp-Config [IKECFG]. To ensure interoperability with past and future versions of Extended Authentication, a Vendor ID is provided in section 2.

All ISAKMP-Config messages in an extended authentication transaction MUST contain the same ISAKMP-Config transaction identifier. The Message ID in the ISAKMP header follows the rules defined by the ISAKMP-Config protocol.

This protocol can therefore be used in conjunction with any existing basic ISAKMP authentication method as defined in [IKE].

This authentication MUST be used after a phase 1 exchange has completed and before any other exchange with the exception of Info mode exchanges. If the extended authentication fails, then the phase 1 SA MUST be immediately deleted. The edge device MAY choose to retry an extended authentication request if the user failed to be authenticated, but must do so in the same ISAKMP-Config transaction, and MUST NOT send the SET message until the user is authenticated, or until the edge device wishes to stop retrying and fail the user.

Extended Authentication MAY be initiated by the edge device at any time after the initial authentication exchange. For example, RADIUS servers may specify that a user only be authenticated for a certain time period. Once that time period has elapsed (minus a possible jitter), the edge device may request a new Extended

Authentication exchange. If the Extended Authentication exchange fails, the edge device MUST tear down all phase 1 and phase 2 SAs associated with the user.

The following are extensions to the ISAKMP-Config [IKECFG] specification to support Extended Authentication.

4.1 Message Types

Type	Value
ISAKMP_CFG_REQUEST	(as defined in [IKECFG])
ISAKMP_CFG_REPLY	(as defined in [IKECFG])
ISAKMP_CFG_SET	(as defined in [IKECFG])
ISAKMP_CFG_ACK	(as defined in [IKECFG])

o ISAKMP_CFG_REQUEST - This message is sent from an edge device to

an IPSec host trying to request extended authentication. Attributes that it requires sent back in the reply MUST be included with a length of zero (0). Attributes required for the authentication reply, such as a challenge string MUST be included with the proper values filled in.

o ISAKMP_CFG_REPLY - This message MUST contain the filled in authentication attributes that were requested by the edge device or if the proper authentication attributes can not be retrieved, then this message MUST contain the XAUTH_STATUS attribute with a value of FAIL.

o ISAKMP_CFG_SET - This message is sent from an edge device and is only used, within the scope of this document, to state the success of the authentication. This message MUST only include the success of failure of the authentication and MAY contain some clarification text.

o ISAKMP_CFG_ACK - This message is sent from the IPSec host acknowledging receipt of the authentication result. Its attributes are not relevant and MAY be skipped entirely, thus no attributes SHOULD be included. This last message in the authentication transaction is used solely as an acknowledgement of the previous message and to eliminate problems with

unacknowledged messages over UDP.

4.2 Attributes

Attribute	Value	Type
XAUTH_TYPE	16520	Basic
XAUTH_USER_NAME	16521	Variable ASCII string
XAUTH_USER_PASSWORD	16522	Variable ASCII string
XAUTH_PASSCODE	16523	Variable ASCII string
XAUTH_MESSAGE	16524	Variable ASCII string
XAUTH_CHALLENGE	16525	Variable ASCII string
XAUTH_DOMAIN	16526	Variable ASCII string
XAUTH_STATUS	16527	Basic

- XAUTH_TYPE - The type of extended authentication requested whose values are described in the next section. This is an optional attribute for the ISAKMP_CFG_REQUEST and ISAKMP_CFG_REPLY messages. If the XAUTH_TYPE is not present, then it is assumed to be Generic. The XAUTH_TYPE in a REPLY MUST be identical to the XAUTH_TYPE in the REQUEST. If the XAUTH_TYPE was not present in the REQUEST, then it MUST NOT be present in the REPLY. However, an XAUTH transaction MAY have multiple REQUEST/REPLY pairs with different XAUTH_TYPE values in each pair.

- XAUTH_USER_NAME - The user name MAY be any unique identifier of the user such as a login name, an email address, or a X.500 Distinguished Name.

- XAUTH_USER_PASSWORD - The user's password.

- XAUTH_PASSCODE - A token card's passcode.

- XAUTH_MESSAGE - A textual message from an edge device to an IPSec host. The message may contain a textual challenge or instruction. An example of this would be "Enter your password followed by your pin number". The message may also contain a reason why authentication failed or succeeded. This message SHOULD be displayed to the user.

- XAUTH_CHALLENGE - A challenge string sent from the edge device to the IPSec host for it to include in its calculation of a

password. This attribute SHOULD only be sent in an ISAKMP_CFG_REQUEST message. Typically, the XAUTH_TYPE attribute dictates how the receiving device should handle the challenge. For example, RADIUS-CHAP uses the challenge to hide the password.

- o XAUTH_DOMAIN - The domain to be authenticated in. This value will have different meaning depending on the authentication type.

- o XAUTH_STATUS - A variable that is used to denote authentication success (OK=1) or failure (FAIL=0). This attribute MUST be sent in the ISAKMP_CFG_SET message, in which case it may be set to either OK or FAIL, and MAY be sent in a REPLY message by a remote peer, in which case it MUST be set to FAIL.

4.3 Authentication Types

Value	Authentication Required
0	Generic
1	RADIUS-CHAP
2	OTP
3	S/KEY
4-32767	Reserved for future use
32768-65535	Reserved for private use

- o Generic - A catch-all type that allows for future extensibility and a generic mechanism to request authentication information. This method allows for any type of extended authentication which does not require specific processing, and should be used whenever possible. This is the default setting if no XAUTH_TYPE is present.

- o RADIUS-CHAP - RADIUS-CHAP is one method of authentication defined in [RADIUS] which uses a challenge to hide the password. In order to use the CHAP functionality defined in [RADIUS], the XAUTH_TYPE MUST be set to RADIUS-CHAP. For all other methods defined in [RADIUS] (i.e. PAP), the XAUTH_TYPE MUST be set to Generic.

- o OTP - One-Time-Passwords as defined in [OTP] uses a challenge string to request a certain generated password. The request SHOULD contain a user name, password and a challenge string while

the reply MUST contain the user name and the generated password. The challenge string is formatted as defined in [OTPEXT].

o S/KEY - This one-time-password scheme defined in [SKEY] was the precursor to OTP, thus the same rules applies.

5 Authentication Method Types

The following values relate to the ISAKMP authentication method attribute used in proposals. They optionally allow an XAUTH implementation to propose use of extended authentication after the initial phase 1 authentication. Values are taken from the private use range defined in [IKE] and should be used among mutually consenting parties. To ensure interoperability and avoid collisions, a Vendor ID is provided in section 2.

Method	Value
XAUTHInitPreShared	65001
XAUTHRespPreShared	65002
XAUTHInitDSS	65003
XAUTHRespDSS	65004
XAUTHInitRSA	65005
XAUTHRespRSA	65006
XAUTHInitRSAEncryption	65007
XAUTHRespRSAEncryption	65008
XAUTHInitRSARevisedEncryption	65009
XAUTHRespRSARevisedEncryption	65010

An Extended Authentication proposal has two characteristics.

The first is the direction of the authentication. Each type identifies whether the Initiator or the Responder is the device which should be authenticated using XAUTH. For example XAUTHInitPreShared is a type which demands that the Initiator be authenticated.

Note that an edge device would typically initiate with one of the following:
o XAUTHRespPreShared

o XAUTHRespDSS
o XAUTHRespRSA
o XAUTHRespRSAEncryption
o XAUTHRespRSARevisedEncryption

and would typically only accept proposals with the following authentication methods:
o XAUTHInitPreShared
o XAUTHInitDSS
o XAUTHInitRSA
o XAUTHInitRSAEncryption
o XAUTHInitRSARevisedEncryption

The second characteristic is the IKE Authentication method to be used. The following table illustrates which keywords in the methods described above relate to which Authentication Methods described in [IKE] Appendix A.

```
"PreShared"            -> pre-shared key
"DSS"                  -> DSS signatures
"RSA"                  -> RSA signatures
"RSAEncryption"        -> Encryption with RSA
"RSARevisedEncryption" -> Revised encryption with RSA
```

6 Other Scenarios for Extended Authentication

Although this document described a scenario where an IPSec host (eg. mobile user) was being authenticated by an edge device (eg. firewall/gateway), the methods described can also be used for edge device to edge device authentication as well as IPSec host to IPSec host authentication.

7 Extensibility

Although this protocol was initially developed for the corporate "Road Warrior" with a dynamic IP address to connect to a corporate Net, there may be certain applications where static IP addresses are used by the "Road Warrior" or where this protocol is used in a non remote-user environment where the IP address is static. There are Security Considerations for certain applications of this protocol in certain deployment scenarios. Please consult the "Security Considerations" section below for more detail.

[IKE] defines many different ways to authenticate a user and generate keying material. There are two basic phase 1 modes defined: Main Mode and Aggressive Mode. There are also at least 5 different authentication schemes which can be used with each mode.

New authentication schemes are being developed and surely more will be standardized in the future. Similarly new phase 1 modes are being proposed to address weaknesses or missing functionality in Main Mode and/or Aggressive mode.

It is for this reason that XAUTH was designed to be fully extensible. Since XAUTH extends the phase 1 authentication provided by [IKE], it is an important design goal that a legacy user authentication scheme in IPsec be able to use the strengths of current and future authentication and key generation schemes.

XAUTH accomplishes this by working with all modes which allow the negotiation of a phase 1 authentication method in ISAKMP. Any new authentication methods defined in the future which are not addressed by this document need simply to take values from the "consenting parties" ranges of [IKE]. Such an example would be the introduction of Encryption with El-Gamal and Revised Encryption with El-Gamal which were introduced in [IKEv2] which is a proposed standard.

Furthermore, any new modes defined, such as [HYBRID] and Base Mode [BASE], will automatically be able to use the functionality of XAUTH as no new numbers are needed.

Finally, any new or forgotten Legacy User Authentication Schemes which are not part of XAUTH can be easily incorporated by taking numbers from the "consenting parties" ranges of XAUTH, or by requesting reserved numbers from IANA.

8 Security Considerations

Care should be taken when sending sensitive information over public networks such as the Internet. A user's password should never be sent in the clear and when sent encrypted, the destination MUST have been previously authenticated. The use of ISAKMP-Config [IKECFG] addresses these issues.

The protocol described in this memo strictly extends the authentication methods described in [IKE]. It does not in any way affect the authenticated nature of the phase 1 security association. In fact, this protocol heavily relies on the

authenticated nature of the phase 1 SA. Without complete phase 1 authentication, this protocol does not provide *any* authentication

at all, since it becomes easily vulnerable to Man-in-the-Middle (MitM) attacks.

This protocol was designed to be extensible, and can be used in many possible combinations of phase 1 Modes and authentication methods. However, certain combinations of scenarios could lead to weaker than desired security, and are therefore discouraged.

When using XAUTH with Pre-Shared keys, where the peer's IP address is dynamic, Main Mode SHOULD NOT be used, and is STRONGLY DISCOURAGED. In this particular scenario, the phase 1 authentication becomes suspect as the administrator has little choice but to use one single Shared-Key for all users, and group-shared keys are susceptible to "social engineering attacks".

However, the choice of implementation of this functionality is left up to the implementers of this protocol. There may be some applications where this functionality is desired. Some examples are: proof of concept deployments and small deployments where the proper management of a group shared-key is less difficult.

If at some point restrictions are introduced in one of the IPsec Standard RFC documents which prohibit the use of group pre-shared keys, then this protocol will, by default, conform, and these Security Considerations will no longer be of concern.

9 References

[Bradner97] S. Bradner, "Key words for use in RFCs to Indicate Requirement Levels", RFC2119

[BASE] Y. Dayan, S. Bitan, "IKE Base Mode", draft-ietf-ipsec-ike-base-mode-01.txt

[CHAP] W. Simpson, "PPP Challenge Handshake Authentication

Protocol (CHAP)", RFC1994

[DIAMETER] P. Calhoun, A. Rubens, "DIAMETER - Base Protocol", draft-calhoun-diameter-02.txt

[HYBRID] M. Litvin, R. Shamir, T. Zegman, "A Hybrid Authentication Mode for IKE", draft-ietf-ipsec-isakmp-hybrid-auth-02

[IKE] D. Harkins, D. Carrel, "The Internet Key Exchange (IKE)", RFC2409

[IKEv2] D. Harkins, D. Carrel, "The Internet Key Exchange (IKE)", draft-ietf-ipsec-ike-00.txt

[IKECFG] R. Pereira, "The ISAKMP Configuration Method", draft-ietf-ipsec-isakmp-cfg-05

[RADIUS] C. Rigney, A. Rubens, W. Simpson, S. Willens, "Remote Authentication Dial In User Service (RADIUS)", RFC2138

[OTP] N. Haller, C. Metz, "A One-Time Password System", RFC1938

[SKEY] N. Haller, "The S/KEY One-Time Password System", RFC1760

[TACACS] C. Finseth, "An Access Control Protocol, Sometimes Called TACACS", RFC1492

[TACACS+] D. Carrel, L. Grant, "The TACACS+ Protocol Version 1.77", draft-grant-tacacs-01.txt

[OTPEXT] C. Metz, "OTP Extended Responses", RFC 2243

10 Acknowledgements

The authors would like to thank Tamir Zegmen, Moshe Litven, Dan Harkins and all those from the IPsec community who have helped

improve the XAUTH protocol either. We would also like to thank Tim Jenkins, Ajai Puri, Laurie Shields, Andrew Krywaniuk, Gabriela Dinescu, Paul Kierstead and Scott Fanning for their continued support, and many sanity checks along the way.

11 Authors' Addresses

Roy Pereira
<royp@cisco.com>
Cisco Systems
+1 (613) 788-7207

Stephane Beaulieu
<sbeaulieu@timestep.com>
TimeStep Corporation
+1 (613) 599-3610 x 4709

The IPSRA working group can be contacted via the IPSec working group's mailing list (ietf-ipsra@vpnc.org) or through its chairs:

Roy Pereira
royp@cisco.com
Cisco Systems

Sara Bitan
sarab@radguard.com
Radguard

11 Expiration

This draft expires May, 2000

12 Full Copyright Statement

Appendix A

This appendix gives more useful examples of Extended Authentication.

Secure ID Next PIN mode
=======================

```
Ipsec Client                                              Ipsec Gateway
------------                                              -------------
                            <-- REQUEST(Username = '', Password = '')
REPLY(Username = 'joe', Password = '1637364856') -->
         <-- REQUEST(Password = '', XAUTH_MESSAGE = 'The system has
               assigned you a new PIN, do you wish to see it now?')
REPLY(Password = 'y') -->
                             <-- REQUEST(Password = '', XAUTH_MESSAGE
                                          = 'Your new pin is 1234'
REPLY(Password = '1234764456') -->
                                                <-- SET(XAUTH_STATUS = OK)
ACK(XAUTH_STATUS) -->
```

```
RADIUS Chap Challenge
=====================

Ipsec Client                                              Ipsec Gateway
------------                                              -------------
      <-- REQUEST(TYPE = RADIUS-CHAP, Username = '', Password = '',
                    Challenge = 0x0102030405060708091011121314151 6)
REPLY(TYPE = RADIUS-CHAP, Username = 'joe', Password =
'0xaa111213141516171819202122232425 26') -->
                                                <-- SET(XAUTH_STATUS = OK)
ACK(XAUTH_STATUS) -->
```

where the Challenge in the REQUEST is the random number generated by the edge device, and the Password in the reply contains the ID used to calculate the hash 'aa' concatenated with the hash of the (ID+challenge+secret)