Network Working Group
Internet-Draft
Updates: RFC 2138
Category: Standards Track
<draft-ietf-radius-tunnel-auth-08.txt>

G. Zorn
Microsoft Corporation
D. Leifer
A. Rubens
Ascend Communications
J. Shriver
Intel Corporation
M. Holdrege
I. Goyret
Lucent Technologies
August 1999

RADIUS Attributes for Tunnel Protocol Support

1. Status of this Memo

This document is an Internet-Draft and is in full conformance with all provisions of Section 10 of RFC2026.

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."

The list of current Internet-Drafts can be accessed at http://www.ietf.org/ietf/1id-abstracts.txt

The list of Internet-Draft Shadow Directories can be accessed at http://www.ietf.org/shadow.html.

The distribution of this memo is unlimited. It is filed as <draft-ietf-radius-tunnel-auth-08.txt>, and expires February 9, 2000. Please send comments to the RADIUS Working Group mailing list (ietf-radius@livingston.com) or to the authors (leifer@del.com, acr@del.com, matt@lucent.com, igoyret@lucent.com, John.Shriver@intel.com and gwz@acm.org).

2. Abstract

This document defines a set of RADIUS attributes designed to support the provision of compulsory tunneling in dial-up networks.

3. Motivation

Many applications of tunneling protocols such as L2TP involve dial-up network access. Some, such as the provision of access to corporate intranets via the Internet, are characterized by voluntary tunneling: the tunnel is created at the request of the user for a specific purpose. Other applications involve compulsory tunneling: the tunnel is created without any action from the user and without allowing the user any choice in the matter. In order to provide this functionality, new RADIUS attributes are needed to carry the tunneling information from the RADIUS server to the tunnel end points; this document defines those attributes. Specific recommendations for, and examples of, the application of these attributes for L2TP can be found in draft-ietf-radius-tunnel-imp-XX.txt.

4. Specification of Requirements

In this document, the key words "MAY", "MUST, "MUST NOT", "optional", "recommended", "SHOULD", and "SHOULD NOT", are to be interpreted as described in [14].

5. Attributes

Multiple instances of each of the attributes defined below may be included in a single RADIUS packet. In this case, the attributes to be applied to any given tunnel SHOULD all contain the same value in their respective Tag fields; otherwise, the Tag field SHOULD NOT be used.

If the RADIUS server returns attributes describing multiple tunnels then the tunnels SHOULD be interpreted by the tunnel initiator as alternatives and the server SHOULD include an instance of the Tunnel-Preference Attribute in the set of Attributes pertaining to each alternative tunnel. Similarly, if the RADIUS client includes multiple sets of tunnel Attributes in an Access-Request packet, all the Attributes pertaining to a given tunnel SHOULD contain the same value in their respective Tag fiels and each set SHOULD include an appropriately valued instance of the Tunnel-Preference Attribute.

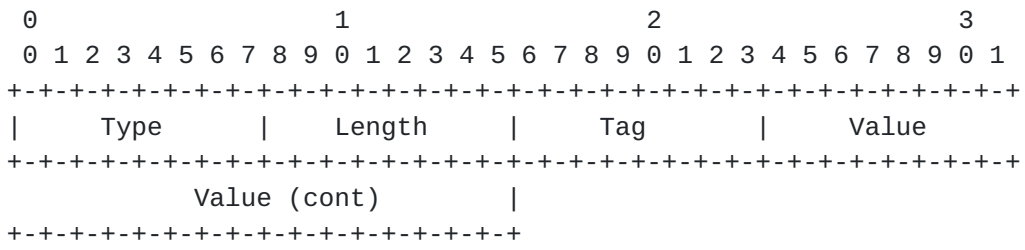
5.1. Tunnel-Type

Description

This Attribute indicates the tunneling protocol(s) to be used (in the case of a tunnel initiator) or the the tunneling protocol in use (in the case of a tunnel terminator). It MAY be included in Access-Request, Access-Accept and Accounting-Request packets. If the Tunnel-Type Attribute is present in an Access-Request packet sent from a tunnel initiator, it SHOULD be taken as a hint to the RADIUS server as to the tunnelling protocols supported by the tunnel end-point; the RADIUS server MAY ignore the hint, however. A tunnel initiator is not required to implement any of these tunnel types; if a tunnel initiator receives an Access-Accept packet which contains only unknown or unsupported Tunnel-Types, the tunnel initiator MUST behave as though an Access-Reject had been received instead.

If the Tunnel-Type Attribute is present in an Access-Request packet sent from a tunnel terminator, it SHOULD be taken to signify the tunnelling protocol in use. In this case, if the RADIUS server determines that the use of the communicated protocol is not authorized, it MAY return an Access-Reject packet. If a tunnel terminator receives an Access-Accept packet which contains one or more Tunnel-Type Attributes, none of which represent the tunneling protocol in use, the tunnel terminator SHOULD behave as though an Access-Reject had been received instead.

A summary of the Tunnel-Type Attribute format is shown below. The fields are transmitted from left to right.



Type
64 for Tunnel-Type

Length
Always 6.

Tag
The Tag field is one octet in length and is intended to provide a means of grouping attributes in the same packet which refer to the

same tunnel. Valid values for this field are 0x01 through 0x1F, inclusive. If the Tag field is unused, it MUST be zero (0x00).

Value
The Value field is three octets and contains one of the following values, indicating the type of tunnel to be started.

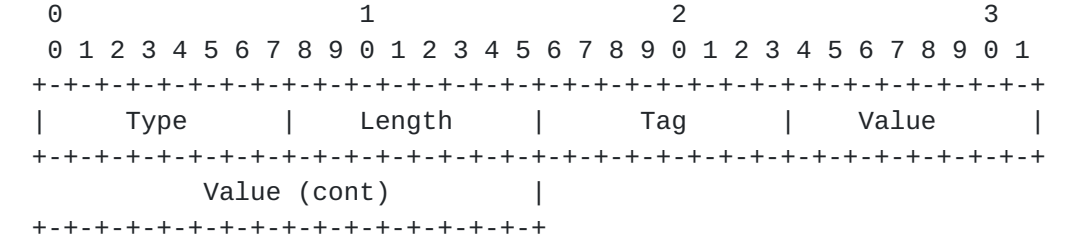
1	Point-to-Point Tunneling Protocol (PPTP) [1]
2	Layer Two Forwarding (L2F) [2]
3	Layer Two Tunneling Protocol (L2TP) [3]
4	Ascend Tunnel Management Protocol (ATMP) [4]
5	Virtual Tunneling Protocol (VTP)
6	IP Authentication Header in the Tunnel-mode (AH) [5]
7	IP-in-IP Encapsulation (IP-IP) [6]
8	Minimal IP-in-IP Encapsulation (MIN-IP-IP) [7]
9	IP Encapsulating Security Payload in the Tunnel-mode (ESP) [8]
10	Generic Route Encapsulation (GRE) [9]
11	Bay Dial Virtual Services (DVS)
12	IP-in-IP Tunneling [10]

5.2. Tunnel-Medium-Type

Description

The Tunnel-Medium-Type Attribute indicates which transport medium to use when creating a tunnel for those protocols (such as L2TP) that can operate over multiple transports. It MAY be included in both Access-Request and Access-Accept packets; if it is present in an Access-Request packet, it SHOULD be taken as a hint to the RADIUS server as to the tunnel media supported by the tunnel end-point. The RADIUS server MAY ignore the hint, however.

A summary of the Tunnel-Medium-Type Attribute format is given below. The fields are transmitted left to right.



Type
65 for Tunnel-Medium-Type

Length
6

Tag
The Tag field is one octet in length and is intended to provide a means of grouping attributes in the same packet which refer to the same tunnel. Valid values for this field are 0x01 through 0x1F, inclusive. If the Tag field is unused, it MUST be zero (0x00).

Value
The Value field is three octets and contains one of the values listed under "Address Family Numbers" in [14]. For the sake of convenience, a relevant excerpt of this list is reproduced below.

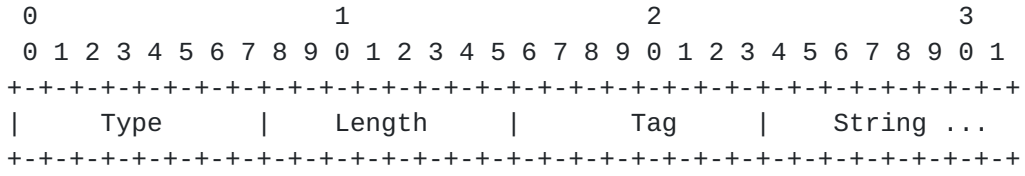
1	IPv4 (IP version 4)
2	IPv6 (IP version 6)
3	NSAP
4	HDLC (8-bit multidrop)
5	BBN 1822
6	802 (includes all 802 media plus Ethernet "canonical format")
7	E.163 (POTS)
8	E.164 (SMDS, Frame Relay, ATM)
9	F.69 (Telex)
10	X.121 (X.25, Frame Relay)
11	IPX
12	Appletalk
13	Decnet IV
14	Banyan Vines
15	E.164 with NSAP format subaddress

5.3. Tunnel-Client-Endpoint

Description

This Attribute contains the address of the initiator end of the tunnel. It MAY be included in both Access-Request and Access-Accept packets to indicate the address from which a new tunnel is to be initiated. If the Tunnel-Client-Endpoint Attribute is included in an Access-Request packet, the RADIUS server should take the value as a hint; the server is not obligated to honor the hint, however. This Attribute SHOULD be included in Accounting-Request packets which contain Acct-Status-Type attributes with values of either Start or Stop, in which case it indicates the address from which the tunnel was initiated. This Attribute, along with the Tunnel-Server-Endpoint and Acct-Tunnel-Connection-ID attributes, may be used to provide a globally unique means to identify a tunnel for accounting and auditing purposes.

A summary of the Tunnel-Client-Endpoint Attribute format is shown below. The fields are transmitted from left to right.



Type

66 for Tunnel-Client-Endpoint.

Length

>= 3

Tag

The Tag field is one octet in length and is intended to provide a means of grouping attributes in the same packet which refer to the same tunnel. If the value of the Tag field is greater than 0x00 and less than or equal to 0x1F, it SHOULD be interpreted as indicating which tunnel (of several alternatives) this attribute pertains. If the Tag field is greater than 0x1F, it SHOULD be interpreted as the first byte of the following String field.

String

The format of the address represented by the String field depends upon the value of the Tunnel-Medium-Type attribute.

If Tunnel-Medium-Type is IPv4 (1), then this string is either the fully qualified domain name (FQDN) of the tunnel client machine, or it is a "dotted-decimal" IP address. Conformant implementations MUST support the dotted-decimal format and SHOULD support the FQDN format for IP addresses.

If Tunnel-Medium-Type is IPv6 (2), then this string is either the FQDN of the tunnel client machine, or it is a text representation of the address in either the preferred or alternate form [17]. Conformant implementations MUST support the preferred form and SHOULD support both the alternate text form and the FQDN format for IPv6 addresses.

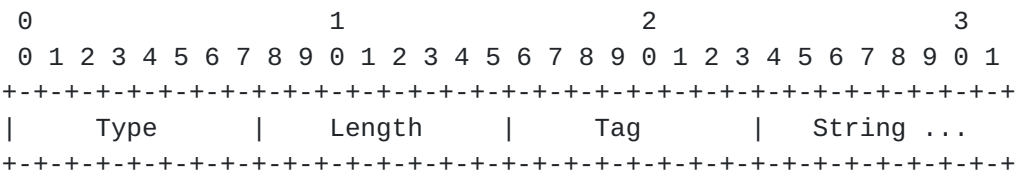
If Tunnel-Medium-Type is neither IPv4 nor IPv6, this string is a tag referring to configuration data local to the RADIUS client that describes the interface and medium-specific address to use.

5.4. Tunnel-Server-Endpoint

Description

This Attribute indicates the address of the server end of the tunnel. The Tunnel-Server-Endpoint Attribute MAY be included (as a hint to the RADIUS server) in the Access-Request packet and MUST be included in the Access-Accept packet if the initiation of a tunnel is desired. It SHOULD be included in Accounting-Request packets which contain Acct-Status-Type attributes with values of either Start or Stop and which pertain to a tunneled session. This Attribute, along with the Tunnel-Client-Endpoint and Acct-Tunnel-Connection-ID Attributes [11], may be used to provide a globally unique means to identify a tunnel for accounting and auditing purposes.

A summary of the Tunnel-Server-Endpoint Attribute format is shown below. The fields are transmitted from left to right.



Type

67 for Tunnel-Server-Endpoint.

Length

>= 3

Tag

The Tag field is one octet in length and is intended to provide a means of grouping attributes in the same packet which refer to the same tunnel. If the value of the Tag field is greater than 0x00 and less than or equal to 0x1F, it SHOULD be interpreted as indicating which tunnel (of several alternatives) this attribute pertains. If the Tag field is greater than 0x1F, it SHOULD be interpreted as the first byte of the following String field.

String

The format of the address represented by the String field depends upon the value of the Tunnel-Medium-Type attribute.

If Tunnel-Medium-Type is IPv4 (1), then this string is either the fully qualified domain name (FQDN) of the tunnel client machine, or it is a "dotted-decimal" IP address. Conformant implementations MUST support the dotted-decimal format and SHOULD

support the FQDN format for IP addresses.

If Tunnel-Medium-Type is IPv6 (2), then this string is either the FQDN of the tunnel client machine, or it is a text representation of the address in either the preferred or alternate form [17]. Conformant implementations MUST support the preferred form and SHOULD support both the alternate text form and the FQDN format for IPv6 addresses.

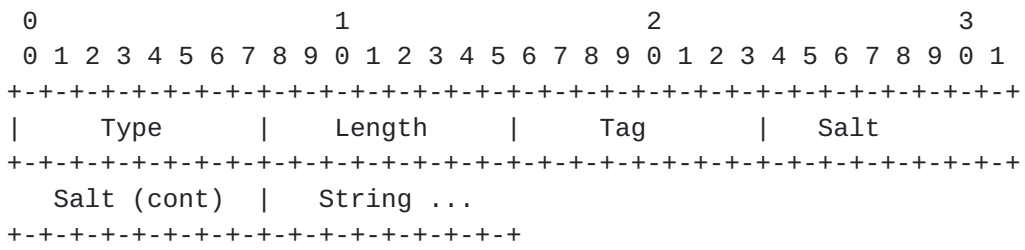
If Tunnel-Medium-Type is not IPv4 or IPv6, this string is a tag referring to configuration data local to the RADIUS client that describes the interface and medium-specific address to use.

5.5. Tunnel-Password

Description

This Attribute may contain a password to be used to authenticate to a remote server. It may only be included in an Access-Accept packet.

A summary of the Tunnel-Password Attribute format is shown below. The fields are transmitted from left to right.



Type
69 for Tunnel-Password

Length
>= 3

Tag
The Tag field is one octet in length and is intended to provide a means of grouping attributes in the same packet which refer to the same tunnel. Valid values for this field are 0x01 through 0x1F, inclusive. If the value of the Tag field is greater than 0x00 and less than or equal to 0x1F, it SHOULD be interpreted as indicating which tunnel (of several alternatives) this attribute pertains; otherwise, the Tag field SHOULD be ignored.

Salt

The Salt field is two octets in length and is used to ensure the uniqueness of the encryption key used to encrypt each instance of the Tunnel-Password attribute occurring in a given Access-Accept packet. The most significant bit (leftmost) of the Salt field MUST be set (1). The contents of each Salt field in a given Access-Accept packet MUST be unique.

String

The plaintext String field consists of three logical sub-fields: the Data-Length and Password sub-fields (both of which are required), and the optional Padding sub-field. The Data-Length sub-field is one octet in length and contains the length of the unencrypted Password sub-field. The Password sub-field contains the actual tunnel password. If the combined length (in octets) of the unencrypted Data-Length and Password sub-fields is not an even multiple of 16, then the Padding sub-field MUST be present. If it is present, the length of the Padding sub-field is variable, between 1 and 15 octets. The String field MUST be encrypted as follows, prior to transmission:

Construct a plaintext version of the String field by concatenating the Data-Length and Password sub-fields. If necessary, pad the resulting string until its length (in octets) is an even multiple of 16. It is recommended that zero octets (0x00) be used for padding. Call this plaintext P.

Call the shared secret S, the pseudo-random 128-bit Request Authenticator (from the corresponding Access-Request packet) R, and the contents of the Salt field A. Break P into 16 octet chunks p(1), p(2)...p(i), where i = len(P)/16. Call the ciphertext blocks c(1), c(2)...c(i) and the final ciphertext C. Intermediate values b(1), b(2)...c(i) are required. Encryption is performed in the following manner ('+' indicates concatenation):

```
b(1) = MD5(S + R + A)    c(1) = p(1) xor b(1)   C = c(1)
b(2) = MD5(S + c(1))     c(2) = p(2) xor b(2)   C = C + c(2)
            .                        .
            .                        .
            .                        .
b(i) = MD5(S + c(i-1))   c(i) = p(i) xor b(i)   C = C + c(i)
```

The resulting encrypted String field will contain c(1)+c(2)+...+c(i).

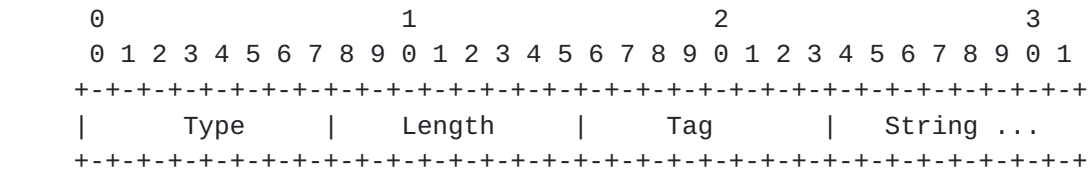
On receipt, the process is reversed to yield the plaintext String.

5.6. Tunnel-Private-Group-ID

Description

This Attribute indicates the group ID for a particular tunneled session. The Tunnel-Private-Group-ID Attribute MAY be included in the Access-Request packet if the tunnel initiator can pre-determine the group resulting from a particular connection and SHOULD be included in the Access-Reply packet if this tunnel session is to be treated as belonging to a particular private group. Private groups may be used to associate a tunneled session with a particular group of users. For example, it may be used to facilitate routing of unregistered IP addresses through a particular interface. It SHOULD be included in Accounting-Request packets which contain Acct-Status-Type attributes with values of either Start or Stop and which pertain to a tunneled session.

A summary of the Tunnel-Private-Group-ID Attribute format is shown below. The fields are transmitted from left to right.



Type

81 for Tunnel-Private-Group-ID.

Length

>= 3

Tag

The Tag field is one octet in length and is intended to provide a means of grouping attributes in the same packet which refer to the same tunnel. If the value of the Tag field is greater than 0x00 and less than or equal to 0x1F, it SHOULD be interpreted as indicating which tunnel (of several alternatives) this attribute pertains. If the Tag field is greater than 0x1F, it SHOULD be interpreted as the first byte of the following String field.

String

This field must be present. The group is represented by the String field. There is no restriction on the format of group IDs.

5.7. Tunnel-Assignment-ID

Description

This Attribute is used to indicate to the tunnel initiator the particular tunnel to which a session is to be assigned. Some tunneling protocols, such as PPTP and L2TP, allow for sessions between the same two tunnel endpoints to be multiplexed over the same tunnel and also for a given session to utilize its own dedicated tunnel. This attribute provides a mechanism for RADIUS to be used to inform the tunnel initiator (e.g. PAC, LAC) whether to assign the session to a multiplexed tunnel or to a separate tunnel. Furthermore, it allows for sessions sharing multiplexed tunnels to be assigned to different multiplexed tunnels.

A particular tunneling implementation may assign differing characteristics to particular tunnels. For example, different tunnels may be assigned different QOS parameters. Such tunnels may be used to carry either individual or multiple sessions. The Tunnel-Assignment-ID attribute thus allows the RADIUS server to indicate that a particular session is to be assigned to a tunnel that provides an appropriate level of service. It is expected that any QOS-related RADIUS tunneling attributes defined in the future that accompany this attribute will be associated by the tunnel initiator with the ID given by this attribute. In the meantime, any semantic given to a particular ID string is a matter left to local configuration in the tunnel initiator.

The Tunnel-Assignment-ID attribute is of significance only to RADIUS and the tunnel initiator. The ID it specifies is intended to be of only local use to RADIUS and the tunnel initiator. The ID assigned by the tunnel initiator is not conveyed to the tunnel peer.

This attribute MAY be included in the Access-Accept. The tunnel initiator receiving this attribute MAY choose to ignore it and assign the session to an arbitrary multiplexed or non-multiplexed tunnel between the desired endpoints. This attribute SHOULD also be included in Accounting-Request packets which contain Acct-Status-Type attributes with values of either Start or Stop and which pertain to a tunneled session.

If a tunnel initiator supports the Tunnel-Assignment-ID Attribute, then it should assign a session to a tunnel in the following manner:

If this attribute is present and a tunnel exists between the specified endpoints with the specified ID, then the session

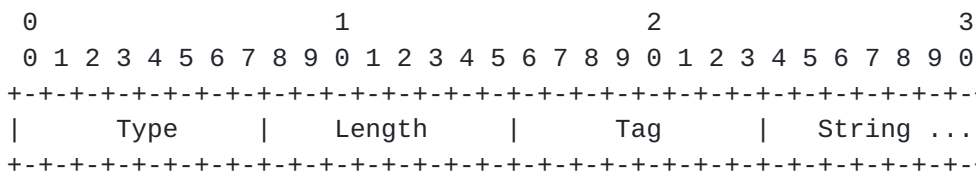
should be assigned to that tunnel.

If this attribute is present and no tunnel exists between the specified endpoints with the specified ID, then a new tunnel should be established for the session and the specified ID should be associated with the new tunnel.

If this attribute is not present, then the session is assigned to an unnamed tunnel. If an unnamed tunnel does not yet exist between the specified endpoints then it is established and used for this and subsequent sessions established without the Tunnel-Assignment-ID attribute. A tunnel initiator MUST NOT assign a session for which a Tunnel-Assignment-ID Attribute was not specified to a named tunnel (i.e. one that was initiated by a session specifying this attribute).

Note that the same ID may be used to name different tunnels if such tunnels are between different endpoints.

A summary of the Tunnel-Assignment-ID Attribute format is shown below. The fields are transmitted from left to right.



Type
82 for Tunnel-Assignment-ID.

Length
> 3

Tag
The Tag field is one octet in length and is intended to provide a means of grouping attributes in the same packet which refer to the same tunnel. If the value of the Tag field is greater than 0x00 and less than or equal to 0x1F, it SHOULD be interpreted as indicating which tunnel (of several alternatives) this attribute pertains. If the Tag field is greater than 0x1F, it SHOULD be interpreted as the first byte of the following String field.

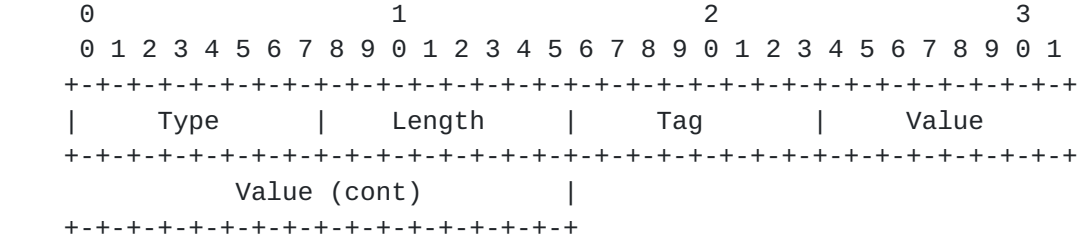
String
This field must be present. The tunnel ID is represented by the String field. There is no restriction on the format of the ID.

5.8. Tunnel-Preference

Description

If more than one set of tunneling attributes is returned by the RADIUS server to the tunnel initiator, this Attribute SHOULD be included in each set to indicate the relative preference assigned to each tunnel. For example, suppose that Attributes describing two tunnels are returned by the server, one with a Tunnel-Type of PPTP and the other with a Tunnel-Type of L2TP. If the tunnel initiator supports only one of the Tunnel-Types returned, it will initiate a tunnel of that type. If, however, it supports both tunnel protocols, it SHOULD use the value of the Tunnel-Preference Attribute to decide which tunnel should be started. The tunnel having the numerically lowest value in the Value field of this Attribute SHOULD be given the highest preference. The values assigned to two or more instances of the Tunnel-Preference Attribute within a given Access-Accept packet MAY be identical. In this case, the tunnel initiator SHOULD use locally configured metrics to decide which set of attributes to use. This Attribute MAY be included (as a hint to the server) in Access-Request packets, but the RADIUS server is not required to honor this hint.

A summary of the Tunnel-Preference Attribute format is shown below. The fields are transmitted from left to right.



Type
83 for Tunnel-Preference

Length
Always 6.

Tag
The Tag field is one octet in length and is intended to provide a means of grouping attributes in the same packet which refer to the same tunnel. Valid values for this field are 0x01 through 0x1F, inclusive. If the Tag field is unused, it MUST be zero (0x00).

Value
The Value field is three octets in length and indicates the

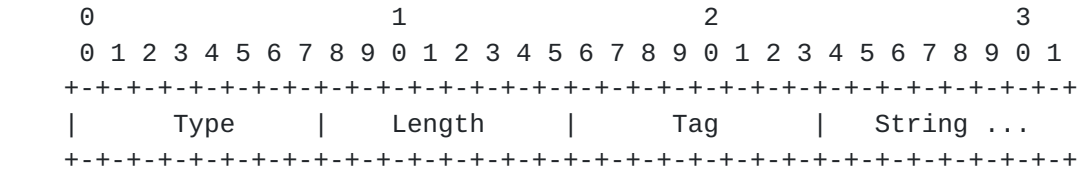
preference to be given to the tunnel to which it refers; higher preference is given to lower values, with 0x000000 being most preferred and 0xFFFFFF least preferred.

5.9. Tunnel-Client-Auth-ID

Description

This Attribute specifies the name used by the tunnel initiator during the authentication phase of tunnel establishment. The Tunnel-Client-Auth-ID Attribute MAY be included (as a hint to the RADIUS server) in the Access-Request packet, and MUST be included in the Access-Request packet if an authentication name other than the default is desired. This Attribute SHOULD be included in Accounting-Request packets which contain Acct-Status-Type attributes with values of either Start or Stop and which pertain to a tunneled session.

A summary of the Tunnel-Client-Auth-ID Attribute format is shown below. The fields are transmitted from left to right.



Type

90 for Tunnel-Client-Auth-ID.

Length

> 3

Tag

The Tag field is one octet in length and is intended to provide a means of grouping attributes in the same packet which refer to the same tunnel. If the value of the Tag field is greater than 0x00 and less than or equal to 0x1F, it SHOULD be interpreted as indicating which tunnel (of several alternatives) this attribute pertains. If the Tag field is greater than 0x1F, it SHOULD be interpreted as the first byte of the following String field.

String

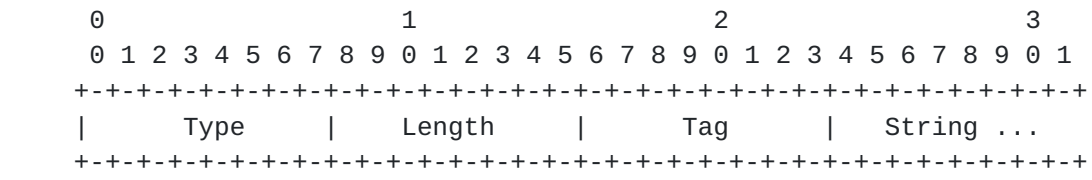
This field must be present. The String field contains the authentication name of the tunnel initiator. The authentication name SHOULD be represented in the UTF-8 charset.

5.10. Tunnel-Server-Auth-ID

Description

This Attribute specifies the name used by the tunnel terminator during the authentication phase of tunnel establishment. The Tunnel-Client-Auth-ID Attribute MAY be included (as a hint to the RADIUS server) in the Access-Request packet, and MUST be included in the Access-Request packet if an authentication name other than the default is desired. This Attribute SHOULD be included in Accounting-Request packets which contain Acct-Status-Type attributes with values of either Start or Stop and which pertain to a tunneled session.

A summary of the Tunnel-Server-Auth-ID Attribute format is shown below. The fields are transmitted from left to right.



Type

91 for Tunnel-Server-Auth-ID.

Length

> 3

Tag

The Tag field is one octet in length and is intended to provide a means of grouping attributes in the same packet which refer to the same tunnel. If the value of the Tag field is greater than 0x00 and less than or equal to 0x1F, it SHOULD be interpreted as indicating which tunnel (of several alternatives) this attribute pertains. If the Tag field is greater than 0x1F, it SHOULD be interpreted as the first byte of the following String field.

String

This field must be present. The String field contains the authentication name of the tunnel terminator. The authentication name SHOULD be represented in the UTF-8 charset.

6. Table of Attributes

The following table provides a guide to which of the above attributes may be found in which kinds of packets, and in what quantity.

Request	Accept	Reject	Challenge	Acct-Request	#	Attribute
0+	0+	0	0	0-1	64	Tunnel-Type
0+	0+	0	0	0-1	65	Tunnel-Medium-Type
0+	0+	0	0	0-1	66	Tunnel-Client-Endpoint
0+	0+	0	0	0-1	67	Tunnel-Server-Endpoint
0	0+	0	0	0	69	Tunnel-Password
0+	0+	0	0	0-1	81	Tunnel-Private-Group-ID
0	0+	0	0	0-1	82	Tunnel-Assignment-ID
0+	0+	0	0	0	83	Tunnel-Preference
0+	0+	0	0	0-1	90	Tunnel-Client-Auth-ID
0+	0+	0	0	0-1	91	Tunnel-Server-Auth-ID

The following table defines the meaning of the above table entries.

0	This attribute MUST NOT be present in packet.
0+	Zero or more instances of this attribute MAY be present in packet.
0-1	Zero or one instance of this attribute MAY be present in packet.

7. Security Considerations

The Tunnel-Password Attribute may contain information which should only be known to a tunnel endpoint. However, the method used to hide the value of the attribute is such that intervening RADIUS proxies will have knowledge of the contents. For this reason, the Tunnel-Password Attribute SHOULD NOT be included in Access-Accept packets which may pass through (relatively) untrusted RADIUS proxies. In addition, the Tunnel-Password Attribute SHOULD NOT be returned to an unauthenticated client; if the corresponding Access-Request packet did not contain a verified instance of the Signature Attribute [15], the Access-Accept packet SHOULD NOT contain an instance of the Tunnel-Password Attribute.

Tunnel protocols offer various levels of security, from none (e.g., PPTP) to strong (e.g., IPSec). Note, however, that in the compulsory tunneling case any security measures in place only apply to traffic between the tunnel endpoints. In particular, end-users SHOULD NOT rely upon the security of the tunnel to protect their data; encryption and/or integrity protection of tunneled traffic MUST NOT be considered as a replacement for end-to-end security.

8. IANA Considerations

This document defines a number of "magic" numbers to be maintained by the IANA. This section explains the criteria to be used by the IANA to assign additional numbers in each of these lists. The following subsections describe the assignment policy for the namespaces defined elsewhere in this document.

8.1. Tunnel-Type Attribute Values

Values 1-12 of the Tunnel-Type Attribute are defined in Section 5.1; the remaining values are available for assignment by the IANA with IETF Consensus [16].

8.2. Tunnel-Medium-Type Attribute Values

Values 1-15 of the Tunnel-Medium-Type Attribute are defined in Section 5.2; the remaining values are available for assignment by the IANA with IETF Consensus [16].

9. References

[1] Hamzeh, K., Pall, G., Verthein, W., Taarud, J., Little, W., Zorn, G., "Point-to-Point Tunneling Protocol (PPTP)", RFC 2637, July 1999

[2] Valencia, A., Littlewood, M., Kolar, T., "Cisco Layer Two Forwarding (Protocol) 'L2F'", RFC 2341, May 1998

[3] Townsley, W. M., Valencia, A., Rubens, A., Pall, G. S., Zorn, G., Palter, B., "Layer Two Tunnelling Protocol (L2TP)", RFC 2661, August 1999

[4] Hamzeh, K., "Ascend Tunnel Management Protocol - ATMP", RFC 2107, February 1997

[5] Kent, S., Atkinson, R., "Security Architecture for the Internet Protocol", RFC 2401, November 1998

[6] Perkins, C., "IP Encapsulation within IP", RFC 2003, October 1996

[7] Perkins, C., "Minimal Encapsulation within IP", RFC 2004, October 1996

[8] Atkinson, R., "IP Encapsulating Security Payload (ESP)", RFC 1827, August 1995

[9] Hanks, S., Li, T., Farinacci, D., Traina, P., "Generic Routing Encapsulation (GRE)", RFC 1701, October 1994

[10] Simpson, W., "IP in IP Tunneling", RFC 1853, October 1995

[11] Zorn, G., Mitton, D., "RADIUS Accounting Modifications for Tunnel Protocol Support", draft-ietf-radius-tunnel-acct-03.txt (work in progress), April 1999

[12] Rigney, C., Rubens, A., Simpson, W., Willens, S., "Remote Authentication Dialin User Service (RADIUS)", RFC 2138, April 1997

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

[14] Reynolds, J., Postel, J., "Assigned Numbers", STD 2, RFC 1700, October 1994

[15] Rigney, C., Willats, W., Calhoun, P., "RADIUS Extensions", draft-ietf-radius-ext-04.txt (work in progress), May 1999

[16] Narten, T., Alvestrand, H., "Guidelines for writing an IANA Considerations Section in RFCs", BCP 26, RFC 2434, October 1998

[17] Hinden, R., Deering, S., "IP Version 6 Addressing Architecture", RFC 2373, July 1998

10. Acknowledgements

Thanks to Dave Mitton for pointing out a nasty circular dependency in the original Tunnel-Password attribute definition and (in no particular order) to Kory Hamzeh, Bertrand Buclin, Andy Valencia, Bill Westfield, Kris Michielsen, Gurdeep Singh Pall, Ran Atkinson, Aydin Edguer, and Bernard Aboba for useful input and review.

11. Chair's Address

The RADIUS Working Group can be contacted via the current chair:

Carl Rigney
Livingston Enterprises
4464 Willow Road
Pleasanton, California 94588

Phone: +1 510 426 0770
E-Mail: cdr@livingston.com

12. Authors' Addresses

Questions about this memo can also be directed to:

Glen Zorn
Microsoft Corporation
One Microsoft Way

Redmond, Washington 98052

Phone: +1 425 703 1559
E-Mail: gwz@acm.org

Dory Leifer
Ascend Communications
1678 Broadway
Ann Arbor, MI 48105

Phone: +1 734 747 6152
E-Mail: leifer@del.com

John Shriver
Intel Corporation
28 Crosby Drive
Bedford, MA 01730

Phone: +1 781 687 1329
E-Mail: John.Shriver@intel.com

Allan Rubens
Ascend Communications
1678 Broadway
Ann Arbor, MI 48105

Phone: +1 313 761 6025
E-Mail: acr@del.com

Matt Holdrege
Lucent Technologies
One Ascend Plaza
1701 Harbor Bay Parkway
Alameda, CA 94502

Phone: +1 510 769 6001
E-Mail: matt@lucent.com

Ignacio Goyret
Lucent Technologies
One Ascend Plaza
1701 Harbor Bay Parkway
Alameda, CA 94502

Phone: +1 510 769 6001
E-Mail: igoyret@lucent.com

13. Expiration Date

This memo is filed as <draft-ietf-radius-tunnel-auth-08.txt>, and expires February 9, 2000.