

Network Working Group
Internet-Draft
Intended status: Standards Track
Expires: April 11, 2013

A. DeKok
FreeRADIUS
G. Halwasia
S. Danda
M. Kumar
Cisco Systems
October 8, 2012

Capability Negotiation in RADIUS
draft-halwasia-radext-capability-negotiation-01

Abstract

This document describes procedure and mechanism to exchange and negotiate capabilities between RADIUS client and RADIUS server.

Requirements Language

The key words "MUST", "MUST NOT", "REQUIRED", "SHALL", "SHALL NOT", "SHOULD", "SHOULD NOT", "RECOMMENDED", "MAY", and "OPTIONAL" in this document are to be interpreted as described in RFC 2119 [RFC2119].

RADIUS-specific terminology is borrowed from [RFC2865] and [RFC2866].

Status of this Memo

This Internet-Draft is submitted in full conformance with the provisions of BCP 78 and BCP 79.

Internet-Drafts are working documents of the Internet Engineering Task Force (IETF). Note that other groups may also distribute working documents as Internet-Drafts. The list of current Internet-Drafts is at <http://datatracker.ietf.org/drafts/current/>.

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."

This Internet-Draft will expire on April 11, 2013.

Copyright Notice

Copyright (c) 2012 IETF Trust and the persons identified as the document authors. All rights reserved.

This document is subject to BCP 78 and the IETF Trust's Legal

Provisions Relating to IETF Documents
(<http://trustee.ietf.org/license-info>) in effect on the date of publication of this document. Please review these documents carefully, as they describe your rights and restrictions with respect to this document. Code Components extracted from this document must include Simplified BSD License text as described in Section 4.e of the Trust Legal Provisions and are provided without warranty as described in the Simplified BSD License.

Table of Contents

1. Introduction	3
1.1. Solution Description	3
2. RADIUS Packets	3
2.1. Capability-Request RADIUS Packet	3
2.2. Capability-Response RADIUS Packet	4
3. RADIUS Attributes	5
3.1. Capability-Add Attribute	6
3.2. Capability-Withdraw Attribute	6
3.3. Capability-Ack Attribute	7
4. RADIUS Capability-Id	8
5. RADIUS Client Behavior	8
6. RADIUS Server Behavior	9
7. Example	9
8. IANA Considerations	10
8.1. New Registry: Capability-Identifier	10
9. Security Considerations	11
10. Acknowledgements	11
11. References	11
11.1. Normative References	11
11.2. Informative References	12
Authors' Addresses	12

1. Introduction

Remote Authentication Dial In User Service (RADIUS) [RFC2865] and [RFC2866] is widely used protocol for Authentication, Authorization and Accounting. There are quite a lot of extensions which are being done on RADIUS protocol and considering that RADIUS is being deployed quite extensively, it would be nice if RADIUS Client and Server can negotiate the capability to support those extensions. This specification recommends and envision each proposed capability to be as precise and narrowly defined as possible and having said that we envision fairly large number of capabilities rather than few broadly defined ones. For example [I-D.ietf-radext-radius-extensions] proposes extended attributes space along with few other extensions and it would be nice if RADIUS Client and Server can signal and negotiate support for 'extended attributes'. This document describes procedure and mechanism to exchange and negotiate capabilities between RADIUS client and RADIUS server.

1.1. Solution Description

This specification proposes to define two new RADIUS packet types to negotiate capabilities between RADIUS client and RADIUS server as defined in section 2. It also proposes to define 3 new attributes to be carried inside new RADIUS packet types. New RADIUS packets to negotiate capability has been chosen as it has minimal impact on the RADIUS security model and existing implementations. Following sections describes the new RADIUS packet types and attributes and describes their usage in negotiating capabilities. As per this specification Capability-Request RADIUS packets MUST NOT be proxied.

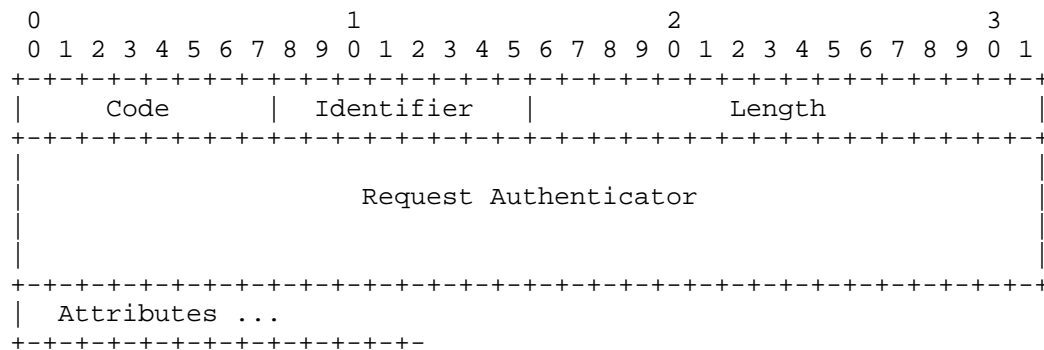
2. RADIUS Packets

This document defines following new RADIUS packet type to enable capability negotiation between RADIUS client and server.

2.1. Capability-Request RADIUS Packet

Capability-Request RADIUS Packets are sent to a RADIUS server to convey the capabilities RADIUS client intends to add and withdraw.

A summary of the Capability-Request packet format is shown below. The fields are transmitted from left to right.



Code

TBA1 - Capability-Request.

Identifier

The Identifier field MUST be changed whenever the content of the Attributes field changes, and whenever a valid reply has been received for a previous request. For retransmissions, the Identifier MUST remain unchanged.

Request Authenticator

The Request Authenticator value MUST be changed each time a new Identifier is used. It is calculated the same way as calculated for Access-Request RADIUS Packet as described in section 3 of RFC2865.

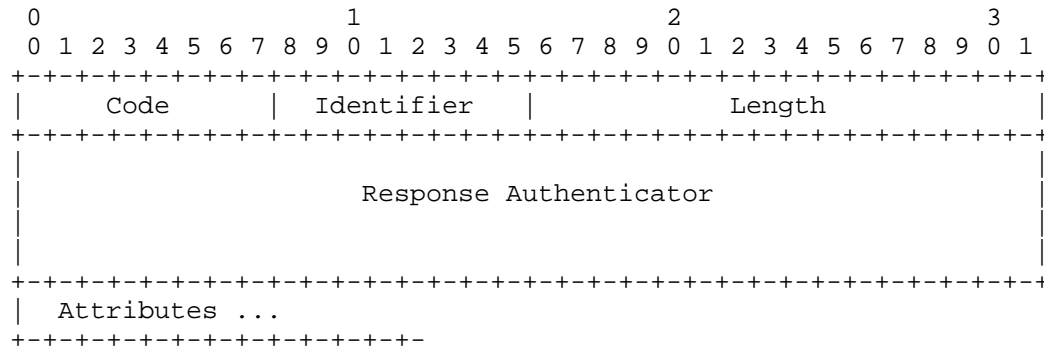
Attributes

The Attribute field is variable in length, and contains the list of Attributes that are required.

2.2. Capability-Response RADIUS Packet

Capability-Withdraw RADIUS Packets are sent to a RADIUS client in response to Capability-Request packet from RADIUS client.

A summary of the Capability-Withdraw packet format is shown below. The fields are transmitted from left to right.



Code

TBA2 - Capability-Response.

Identifier

The Identifier field is a copy of the Identifier field of the Capability-Request which caused this Capability-Response.

Response Authenticator

The Response Authenticator value is calculated from the Capability-Request value. It is calculated the same way as calculated for Access-Accept RADIUS Packet similar to as described in section 3 of RFC2865.

Attributes

The Attribute field is variable in length, and contains the list of Attributes that are required.

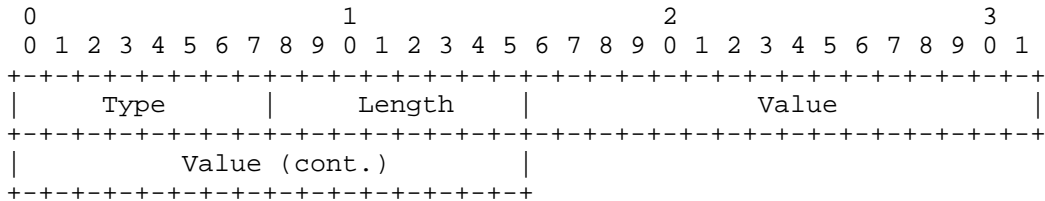
3. RADIUS Attributes

This document defines following new RADIUS attributes to enable capability negotiation between RADIUS client and server.

3.1. Capability-Add Attribute

This attribute indicates the capability which the client wants to add.

The format of the Capability-Add Attribute is:



Type

TBA3 - Capability-Add

Length

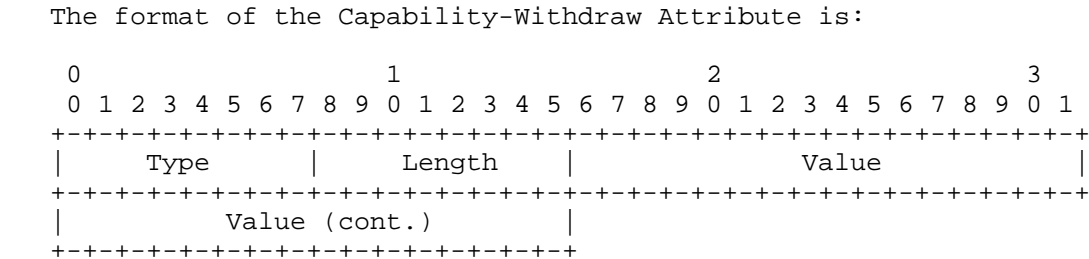
6

Value

Enumerated Data Type in 4-Octet unsigned integer defined in [RFC6158]. This field contains the capability-id as specified in section 4 of this document.

3.2. Capability-Withdraw Attribute

This attribute indicates the capability which the client wants to withdraw.



Type

TBA4 - Capability-Withdraw

Length

6

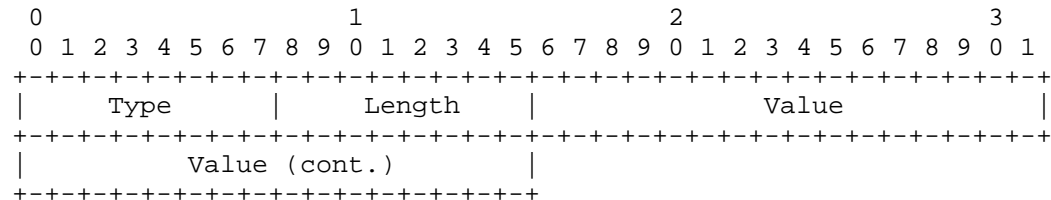
Value

Enumerated Data Type in 4-Octet unsigned integer defined in [RFC6158]. This field contains the capability-id as specified in section 4 of this document.

3.3. Capability-Ack Attribute

This attribute indicates the capability which the server wants to Acknowledge.

The format of the Capability-Ack Attribute is:



Type

TBA5 - Capability-Ack

Length

6

Value

Enumerated Data Type in 4-Octet unsigned integer defined in [RFC6158]. This field contains the capability-id as specified in section 4 of this document.

4. RADIUS Capability-Id

Value field of Capability-Add and Capability-Withdraw attributes defined above contains the Capability-Id of the capability RADIUS client wants to negotiate with RADIUS server. This document does not define any new capability and it's associated Capability-Id. Any specification which wants to use this mechanism of capability negotiation MUST define a new capability. Each new capability MUST be registered with IANA to get a capability-id from capability-id registry.

5. RADIUS Client Behavior

RADIUS Client willing to negotiate capabilities SHOULD send Capability-Request RADIUS Packet (defined in section 2) towards the RADIUS Server. RADIUS Client MUST include Capability-Add Attribute in Capability-Request RADIUS Packet for the capability client wants to add/negotiate. Client can also include Capability-Withdraw Attribute in the RADIUS packet in case it wants to withdraw the

capability it has negotiated earlier. Client MUST NOT add the Capability-Withdraw Attribute in the Capability-Request RADIUS Packet in case it has not negotiated the corresponding attribute earlier. Client can include multiple Capability-Add and/or Capability-Withdraw Attributes in the same Capability-Request RADIUS Packet. RADIUS client MUST add at least one Capability-Add and/or Capability-Withdraw Attribute in Capability-Request RADIUS Packet. Client MUST NOT include the Capability-Add and Capability-Withdraw Attribute for the same capability in the same Capability-Request RADIUS Packet.

Apart from including Capability-Add and/or Capability-Withdraw Attributes in the Capability-Request RADIUS Packet, Client can include NAS-Identifier [RFC2865] or one of the NAS-IP-Address[RFC2865]/NAS-IPv6-Address [RFC3162] for the purpose of RADIUS server to identify client.

6. RADIUS Server Behavior

RADIUS Server implementing this specification MUST respond back with Capability-Response RADIUS Packet after receiving a valid Capability-Request RADIUS Packet from the RADIUS Client. As specified in section 5, Client will advertise its capabilities by including Capability-Add and/or Capability-Withdraw Attributes in the same Capability-Request RADIUS Packet. RADIUS Server will find out the common set of agreed upon capabilities based upon the intersection in between capabilities received from client and its own capabilities. RADIUS Server MUST include a Capability-Ack Attribute for each of the agreed upon capabilities in the Capability-Response RADIUS Packet. RADIUS Server MUST NOT include Capability-Ack attributes for all those capabilities which it does not want to support/share with RADIUS Client. If the RADIUS Server does not support any of the capabilities specified in Capability-Request RADIUS Packet, it SHOULD send back an empty Capability-Response RADIUS Packet without including any Capability-Ack attribute.

RADIUS Server implementation which does not support capability negotiation specified in this specification MUST silently discard Capability-Request RADIUS Packet received from RADIUS Client.

7. Example

Following example figure shows the sequence of message exchanges which happens between RADIUS Client and RADIUS Server to negotiate capabilities.

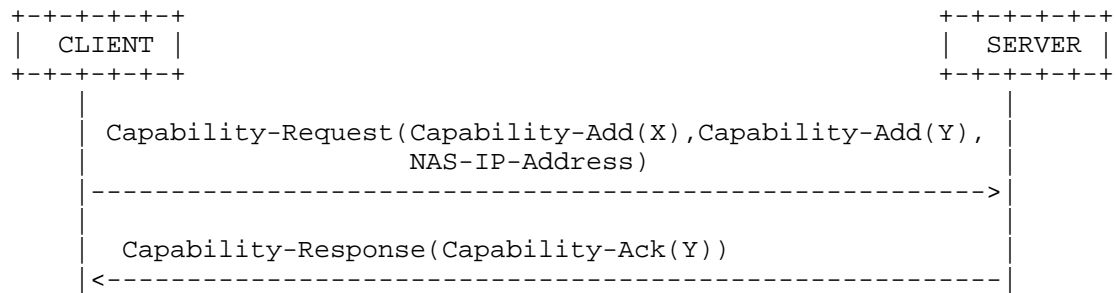


Figure 1: Capability Negotiation between Client and Server

Capability X = 'Understands 64-Bit Integers'

Capability Y = 'Supports Larger than 4K RADIUS packets'

8. IANA Considerations

The authors request that Packet Type, Attribute Types and Attribute Values defined in this document be registered by the Internet Assigned Numbers Authority (IANA) from the RADIUS namespaces as described in the "IANA Considerations" section of RFC 3575 [RFC3575], in accordance with BCP 26 [RFC5226]. For RADIUS packets, attributes and registries created by this document IANA is requested to place them at <http://www.iana.org/assignments/radius-types>.

This document defines the following RADIUS messages:

- Capability-Request
- Capability-Response

This document defines the following attributes:

- Capability-Add
- Capability-Withdraw
- Capability-Ack

Additionally, IANA is requested to create the following new registries listed in the subsections below.

8.1. New Registry: Capability-Identifier

This document also defines an Capability-Identifier registry (used in the value field of Capability-Add, Capability-Withdraw and Capability-Ack Attributes). IANA is requested to just allocate space

for this registry and this document does not request IANA to allocate any value from this registry.

Requests to IANA for a new value for a Capability Identifier will be approved by Expert Review. A designated expert will be appointed by the IESG.

9. Security Considerations

This document defines new RADIUS message types and new Attribute types, but otherwise makes no changes to the security of the RADIUS protocol.

10. Acknowledgements

11. References

11.1. Normative References

- [I-D.ietf-radext-radius-extensions]
DeKok, A. and A. Lior, "Remote Authentication Dial In User Service (RADIUS) Protocol Extensions",
draft-ietf-radext-radius-extensions-06 (work in progress),
June 2012.
- [RFC2119] Bradner, S., "Key words for use in RFCs to Indicate Requirement Levels", BCP 14, RFC 2119, March 1997.
- [RFC2865] Rigney, C., Willens, S., Rubens, A., and W. Simpson, "Remote Authentication Dial In User Service (RADIUS)", RFC 2865, June 2000.
- [RFC3162] Aboba, B., Zorn, G., and D. Mitton, "RADIUS and IPv6", RFC 3162, August 2001.
- [RFC3575] Aboba, B., "IANA Considerations for RADIUS (Remote Authentication Dial In User Service)", RFC 3575, July 2003.
- [RFC5226] Narten, T. and H. Alvestrand, "Guidelines for Writing an IANA Considerations Section in RFCs", BCP 26, RFC 5226, May 2008.
- [RFC6158] DeKok, A. and G. Weber, "RADIUS Design Guidelines", BCP 158, RFC 6158, March 2011.

11.2. Informative References

[RFC2866] Rigney, C., "RADIUS Accounting", RFC 2866, June 2000.

Authors' Addresses

Alan DeKok
FreeRADIUS

Phone:
Email: aland@deployingradius.com

Gaurav Halwasia
Cisco Systems
Cessna Business Park, Sarjapura Marathalli Outer Ring Road
Bangalore, KARNATAKA 560 087
India

Phone: +91 80 4429 2703
Email: ghalwasi@cisco.com

Satyanarayana Danda
Cisco Systems
Cessna Business Park, Sarjapura Marathalli Outer Ring Road
Bangalore, KARNATAKA 560 087
India

Phone: +91 80 4429 2684
Email: sdanda@cisco.com

Manoj Kumar
Cisco Systems
Cessna Business Park, Sarjapura Marathalli Outer Ring Road
Bangalore, KARNATAKA 560 087
India

Phone: +91 80 4429 2635
Email: magoyal@cisco.com

