Diameter Maintenance and
Extensions (DIME)
Internet-Draft
Intended status: Standards Track
Expires: December 28, 2008

J. Korhonen
TeliaSonera
H. Tschofenig
Nokia Siemens Networks
M. Arumaithurai
University of Goettingen
M. Jones, Ed.
A. Lior
Bridgewater Systems

June 26, 2008

Quality of Service Attributes for Diameter
draft-ietf-dime-qos-attributes-07.txt

Status of this Memo

Abstract

This document extends the IPFilterRule AVP functionality of the Diameter Base protocol and the functionality of the QoS-Filter-Rule AVP defined in RFC 4005. The ability to convey Quality of Service information using the AVPs defined in this document is available to existing and future Diameter applications where permitted by the command ABNF.

Table of Contents

1. Introduction

This document defines a number of Diameter Quality of Service (QoS) related AVPs that can be used in existing and future Diameter applications where permitted by the command ABNF. The Extended-QoS-Filter-Rule AVP thereby replaces the IPFilterRule, defined in RFC 3588bis [I-D.ietf-dime-rfc3588bis], and the QoS-Filter-Rule, defined in RFC 4005 [RFC4005].

2. Terminology

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].

3. Diameter QoS Defined AVPs

3.1. QoS-Capability AVP

The QoS-Capability AVP (AVP Code TBD) is of type Grouped and contains a list of supported Quality of Service profile templates (and therefore the support of the respective parameter AVPs).

The QoS-Capability AVP may be used for a simple announcement of the QoS capabilities and QoS profiles supported by a peer. It may also be used to negotiate a mutually supported set of QoS capabilities and QoS profiles between two peers.

```
QoS-Capability ::= < AVP Header: XXX >
                   * [ QoS-Profile-Template ]
                   * [ Vendor-Specific-QoS-Profile-Template ]
                   * [ AVP ]
```

3.2. QoS-Profile-Template AVP

The QoS-Profile-Template AVP (AVP Code TBD) is of type Unsigned32 and contains a QoS profile template identifier. An initial QoS profile template is defined with value of 0 and is described in [I-D.ietf-dime-qos-parameters]. The registry for the QoS profile templates is created with the same document.

3.3. Vendor-Specific-QoS-Profile-Template AVP

The Vendor-Specific-QoS-Profile-Template AVP (AVP Code TBD) is of type Grouped and defines a vendor-specific QoS profile template.

The Vendor-Id AVP contains a 32 bit IANA SMI Network Management Private Enterprise Code and the QoS-Profile-Template AVP contains the template identifier assigned by the vendor.

```
Vendor-Specific-QoS-Profile-Template ::= < AVP Header: XXX >
                                           { Vendor-Id }
                                           { QoS-Profile-Template }
                                         * [ AVP ]
```

3.4. QoS-Resources AVP

The QoS-Resources AVP (AVP Code TBD) is of type Grouped and includes a description of the Quality of Service resources for policing traffic flows.

```
QoS-Resources ::= < AVP Header: XXX >
                  * [ Extended-QoS-Filter-Rule ]
                  * [ AVP ]
```

3.5. Extended-QoS-Filter-Rule AVP

The Extended-QoS-Filter-Rule AVP (AVP Code TBD) is of type Grouped and defines one or more traffic flows together with a set of QoS parameters that should be applied to the flow(s) by the Resource Management Function. This AVP uses the Classifier AVP (see Section 5) to describe traffic flows.

```
Extended-QoS-Filter-Rule ::= < AVP Header: XXX >
                               { QoS-Semantics }
                               [ QoS-Profile-Template ]
                               [ Vendor-Specific-QoS-Profile-Template ]
                               [ QoS-Parameters ]
                               [ QoS-Rule-Precedence ]
                               [ Classifier ]
                             * [ AVP ]
```

Either the QoS-Profile-Template or Vendor-Specific-QoS-Profile-Template AVP MUST appear in the Extended-QoS-Filter-Rule AVP.

3.6. QoS-Semantics

The QoS-Semantics AVP (AVP Code TBD) is of type Enumerated and provides the semantics for the QoS-Profile-Template and QoS-Parameters AVPs in the Extended-QoS-Filter-Rule AVP.

This document defines the following values:

(0): QoS-Desired
(1): QoS-Available
(2): QoS-Reserved
(3): Minimum-QoS
(4): QoS-Authorized

3.7. QoS-Parameters AVP

The QoS-Parameters AVP (AVP Code TBD) is of type OctetString and contains Quality of Service parameters. These parameters are defined in a separate document, see [I-D.ietf-dime-qos-parameters].

3.8. QoS-Rule-Precedence AVP

The QoS-Rule-Precedence AVP (AVP Code TBD) is of type Unsigned32 and specifies the execution order of the rules expressed in the QoS-Resources AVP. Rules with equal precedence MAY be executed in parallel if supported by the Resource Management Function. If the QoS-Rule-Precedence AVP is absent from the Extended-QoS-Filter-Rule AVP, the rules SHOULD be executed in the order in which they appear in the QoS-Resources AVP. The lower the numerical value of QoS-Rule-Precedence AVP, the higher the rule precedence.

4. Semantics of QoS Parameters

The QoS parameters carried in the QoS-Resources AVP may appear in different messages. The semantic of the QoS parameters depend on the information provided in the QoS-Semantics AVP which currently defines 5 values, namely QoS-Desired (0), QoS-Available (1), QoS-Reserved (2), Minimum-QoS (3), and QoS-Authorized (4).

The semantics of the different values are as follows:

Object Type	Direction	Semantic
QoS-Desired	C->S	Please authorize the indicated QoS
QoS-Desired	C<-S	NA
QoS-Available	C->S	Admission Control at interface indicates that this QoS is available. (note 1)
QoS-Available	C<-S	Indicated QoS is available. (note 2)
QoS-Reserved	C->S	Used for reporting during accounting.

QoS-Reserved	C<-S	NA
Minimum-QoS	C->S	Indicates that the client is not interested in authorizing QoS that is lower than Min. QoS.
Minimum-QoS	C<-S	The client must not provide QoS guarantees lower than Min. QoS.
QoS-Authorized	C->S	NA
QoS-Authorized	C<-S	Indicated QoS authorized

Legend:

C: Diameter client
S: Diameter server
NA: Not applicable to this document;
no semantic defined in this specification

Notes:

(1) QoS-Available is only useful in relationship with QoS-Desired (and optionally with Minimum-QoS).
(2) QoS-Available is only useful when the AAA server performs admission control and knows about the resources in the network.

5. Diameter Classifier AVPs

Classifiers are used in many applications to specify how to classify packets. For example in a QoS application, if a packet matches a classifier then that packet will be treated in accordance with a QoS specification associated with that classifier.

The Classifiers are sent to on on-path element (e.g. a router) which uses the classifier to match packets. Figure 1 shows a typical deployment.

+-----------+

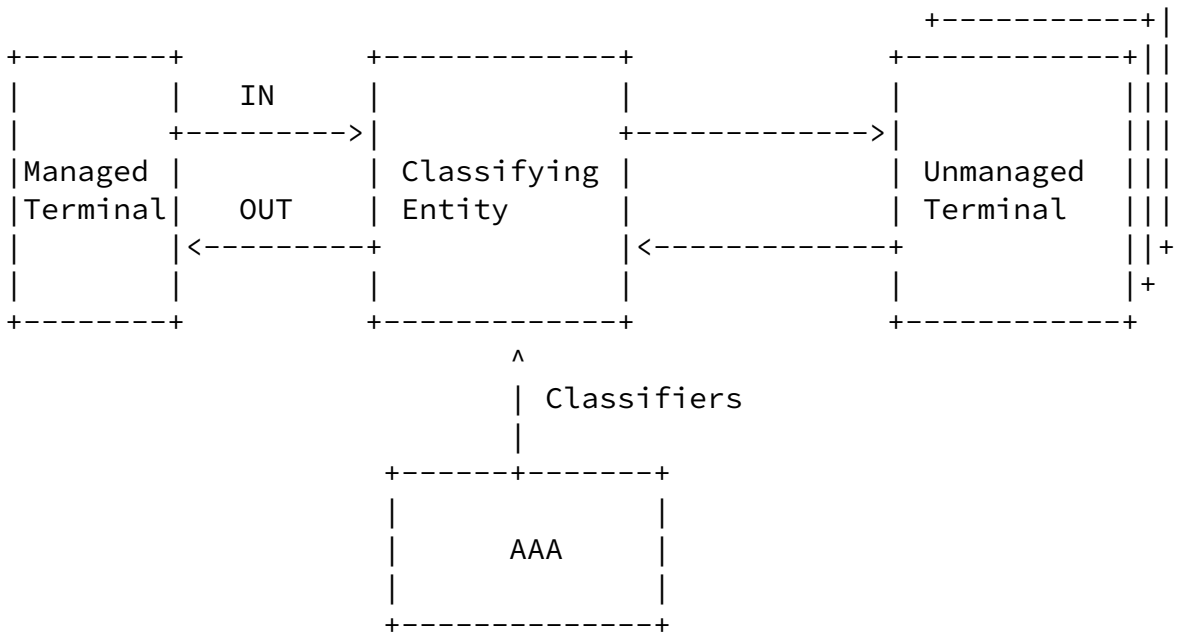


Figure 1: Example of a Classifier Architecture

The managed terminal, the terminal for which the classifiers are being specified is located on the left of the Classifying Entity. The unmanaged terminal, the terminal that receives packets from the Managed terminal or sends packets to the managed terminal is located to the right side of the Classifying Entity.

The Classifying Entity is responsible for classifying packets that are incoming (IN) from the Managed Terminal or packets outgoing (OUT) to the Managed Terminal.

A Classifier consists of a group of attributes that specify how to match a packet. Each set of attributes expresses values about aspects of the packet - typically the packet header. Different protocols therefore would use different attributes.

In general a Classifier consists of the following:

Identifier:

The identifier uniquely identifies this classifier and may be used to reference the classifier from another structure.

From:

Specifies the rule for matching the source part of the packet.

To:

Specifies the rule for matching the destination part of the packet.

Protocol:

Specifies the matching protocol of the packet.

Direction:

Specifies whether the classifier is to apply to packets flowing from the Managed Terminal (IN) or to packets flowing to the Managed Terminal (OUT), or packets flowing in both direction.

Options:

Associated with each protocol or layer, or various values specific to the header of the protocol or layer. Options allow matching on those values.

Each protocol type will have a specific set of attributes that can be used to specify a classifier for that protocol. These attributes will be grouped under a grouped AVP called a Classifier AVP.

5.1. Classifier AVP

The Classifier AVP (AVP Code TBD) is a grouped AVP that consists of a set of attributes that specify how to match a packet.

```
Classifier ::= < AVP Header: XXX >
               { Classifier-ID }
               { Protocol }
               { Direction }
             * [ From-Spec ]
             * [ To-Spec ]
             * [ Diffserv-Code-Point ]
               [ Fragmentation-Flag ]
             * [ IP-Option ]
             * [ TCP-Option ]
               [ TCP-Flags ]
             * [ ICMP-Type ]
             * [ ETH-Option ]
             * [ AVP ]
```

5.2. Classifier-ID AVP

The Classifier-ID AVP (AVP Code TBD) is of type OctetString and uniquely identifies the classifier. Each application will define the uniqueness scope of this identifier, e.g. unique per terminal or globally unique. Exactly one Classifier-ID AVP MUST be contained within a Classifier AVP.

5.3. Protocol AVP

The Protocol AVP (AVP Code TBD) is of type Enumerated and specifies the protocol being matched. The attributes included in the Classifier AVP must be consistent with the value of the Protocol AVP. Exactly one Protocol AVP MUST be contained within a Classifier AVP. The values for this AVP are managed by IANA under the Protocol Numbers registry [PROTOCOL].

5.4. Direction AVP

The Direction AVP (AVP Code TBD) is of type Enumerated that specifies in which direction to apply the Classifier. The values of the enumeration are: "IN","OUT","BOTH". In the "IN" and "BOTH" directions, the From-Spec refers to the address of the Managed Terminal and the To-Spec refers to the unmanaged terminal. In the "OUT" direction, the From-Spec refers to the Unmanaged Terminal whereas the To-Spec refers to the Managed Terminal.

Value	Name and Semantic
0	RESERVED
1	IN - The classifier applies to flows from the Managed Terminal.
2	OUT - The classifier applies to flows to the Managed Terminal.
3	BOTH - The classifier applies to flows both to and from the Managed Terminal.

5.5. From-Spec AVP

The From-Spec AVP (AVP Code TBD) is a grouped AVP that specifies the Source Specification used to match the packet. Zero or more of these AVPs may appear in the Classifier. If this AVP is absent from the Classifier then all packets are matched regardless of the source address. If more than one instance of this AVP appears in the Classifier then the source of the packet can match any From-Spec AVP. The contents of this AVP are protocol specific.

If more than one instance of the IP address AVPs (IP-Address, IP-Address-Range, IP-Address-Mask, Use-Assigned-Address) appear in the From-Spec AVP then the source IP address of the packet must match one of the addresses represented by these AVPs.

If more that one instance of the layer 2 address AVPs (MAC-Address, MAC-Address-Mask, EUI64-Address, EUI64-Address-Mask) appears in the From-Spec then the the source layer 2 address of the packet must match one of the addresses represented in these AVPs.

If more that one instance of the VLAN-ID AVP appears in the From-Spec then the VLAN-ID of the packet must match one of the VLAN-IDs represented in these AVPs.

If more that one instance of the port AVPs (Port, Port-Range) appears in the From-Spec AVP then the source port number must match one of the port numbers represented in these AVPs.

If the IP address, MAC address and port AVPs appear in the same From-Spec AVP then the source packet must match all the specifications, i.e. match the IP address AND MAC address AND port number.

```
From-Spec ::= < AVP Header: XXX >
              * [ IP-Address ]
              * [ IP-Address-Range ]
              * [ IP-Address-Mask ]
              * [ MAC-Address ]
              * [ MAC-Address-Mask]
              * [ EUI64-Address ]
              * [ EUI64-Address-Mask]
              * [ VLAN-ID ]
```

```
                * [ Port ]
                * [ Port-Range ]
                  [ Negated ]
                  [ Use-Assigned-Address ]
                * [ AVP ]
```

5.6. To-Spec AVP

The To-Spec AVP (AVP Code TBD) is a grouped AVP that specifies the Destination Specification used to match the packet. Zero or more of these AVPs may appear in the Classifier. If this AVP is absent from the Classifier then all packets are matched regardless of the destination address. If more than one instance of this AVP appears in the Classifier then the destination of the packet can match any To-Spec AVP. The contents of this AVP are protocol specific.

If more than one instance of the IP address AVPs (IP-Address, IP-Address-Range, IP-Address-Mask, Use-Assigned-Address) appear in the To-Spec AVP then the destination IP address of the packet must match one of the addresses represented by these AVPs.

If more that one instance of the layer 2 address AVPs (MAC-Address, MAC-Address-Mask, EUI64-Address, EUI64-Address-Mask) appears in the To-Spec then the the destination layer 2 address of the packet must match one of the addresses represented in these AVPs.

If more that one instance of the VLAN-ID AVP appears in the From-Spec then the VLAN-ID of the packet must match one of the VLAN-IDs represented in these AVPs.

If more that one instance of the port AVPs (Port, Port-Range) appears in the To-Spec AVP then the destination port number must match one of the port numbers represented in these AVPs.

If the IP address, MAC address and port AVPs appear in the same To-Spec AVP then the destination packet must match all the specifications, i.e. match the IP address AND MAC address AND port number.

```
To-Spec ::= < AVP Header: XXX >
```

```
            * [ IP-Address ]
            * [ IP-Address-Range ]
            * [ IP-Address-Mask ]
            * [ MAC-Address ]
            * [ MAC-Address-Mask]
            * [ EUI64-Address ]
            * [ EUI64-Address-Mask]
            * [ VLAN-ID ]
            * [ Port ]
            * [ Port-Range ]
              [ Negated ]
              [ Use-Assigned-Address ]
            * [ AVP ]
```

5.7. Source and Destination AVPs

For packet classification the contents of the From-Spec and To-Spec can contain the following AVPs.

By combining several of these AVPs within a From-Spec or To-Spec AVP and using more than one From-Spec or To-Spec AVP in the Classifier AVP, one can express many different types of address pools.

5.7.1. Negated AVP

The Negated AVP (AVP Code TBD) of type Enumerated containing the values of True or False. Exactly zero or one of these AVPs may appear in the From-Spec or To-Spec AVP. When set to True the meaning of the match in the To-Spec and From-Spec are negated, causing all other addresses to be matched instead.

When set to False, or when the AVP is not included in the From-Spec or To-Spec AVP then the meaning of the match is not inverted, causing only the addresses specified to be matched.

Note that the negation does not impact the port comparisons.

Value	Name
0	False

```
      1   | True
```

5.7.2. IP-Address AVP

The IP-Address AVP (AVP Code TBD) is of type Address and specifies a single IP address (IPv4 or IPv6) address to match.

5.7.3. IP-Address-Range AVP

The IP-Address-Range AVP (AVP Code TBD) is of type Grouped and specifies an inclusive IP address range.

```
IP-Address-Range ::= < AVP Header: XXX >
                       [ IP-Address-Start ]
                       [ IP-Address-End ]
                     * [ AVP ]
```

If the IP-Address-Start AVP is not included then the address range starts from the first valid IP address up to and including the specified IP-Address-End address.

If the IP-Address-End AVP is not included then the address range starts at the address specified by the IP-Address-Start AVP and includes all the remaining valid IP addresses.

For the IP-Address-Range AVP to be valid, the IP-Address-Start AVP MUST contain a value that is less than that of the IP-Address-End AVP.

5.7.4. IP-Address-Start AVP

The IP-Address-Start AVP (AVP Code TBD) is of type Address and specifies the first IP address (IPv4 or IPv6) address of an IP address range.

5.7.5. IP-Address-End AVP

The IP-Address-End AVP (AVP Code TBD) is of type Address and specifies the last IP address (IPv4 or IPv6) address of an address range.

5.7.6. IP-Address-Mask AVP

The IP-Address-Mask AVP (AVP Code TBD) is of type Grouped and specifies an IP address range using a base IP address and the bit-width of the mask. For example, a range expressed as 1.2.3.0/24 will match all IP addresses from 1.2.3.0 up to and including 1.2.3.255. The bit-width MUST be valid for the type of IP address.

```
IP-Address-Mask ::= < AVP Header: XXX >
                    { IP-Address }
                    { IP-Bit-Mask-Width }
                  * [ AVP ]
```

5.7.7. IP-Mask-Bit-Mask-Width AVP

The IP-Bit-Mask-Width AVP (AVP Code TBD) is of type OctetString. The value is a single octet and specifies the width of an IP address bit-mask.

5.7.8. MAC-Address AVP

The MAC-Address AVP (AVP Code TBD) is of type OctetString and specifies a single layer 2 address in MAC-48 format. The value is a 6 octets encoding of the address as it would appear in the frame header.

5.7.9. MAC-Address-Mask AVP

The MAC-Address-Mask AVP (AVP Code TBD) is of type Grouped and specifies a set of MAC addresses using a bit mask to indicate the bits of the MAC addresses which must fit to the specified MAC address attribute. For example, a MAC-Address-Mask with the MAC-Address as 00-10-A4-23-00-00 and with a MAC-Address-Mask-Pattern of FF-FF-FF-FF-00-00 will match all MAC addresses from 00-10-A4-23-00-00 up to and including 00-10-A4-23-FF-FF.


```
MAC-Address-Mask ::= < AVP Header: XXX >
                     { MAC-Address }
                     { MAC-Address-Mask-Pattern }
                   * [ AVP ]
```

5.7.10. MAC-Address-Mask-Pattern AVP

The MAC-Address-Mask-Pattern AVP (AVP Code TBD) is of type OctetString. The value is a 6 octets specifying the bit positions of a MAC address, that are taken for matching.

5.7.11. EUI64-Address AVP

The EUI64-Address AVP (AVP Code TBD) is of type OctetString and specifies a single layer 2 address in EUI-64 format. The value is a 8 octets encoding of the address as it would appear in the frame header.

5.7.12. EUI64-Address-Mask AVP

The EUI64-Address-Mask AVP (AVP Code TBD) is of type Grouped and specifies a set of EUI64 addresses using a bit mask to indicate the bits of the EUI64 addresses which must fit to the specified EUI64 address attribute. For example, a EUI64-Address-Mask with the EUI64-Address as 00-10-A4-FF-FE-23-00-00 and with a EUI64-Address-Mask-Pattern of FF-FF-FF-FF-FF-FF-00-00 will match all EUI64 addresses from 00-10-A4-FF-FE-23-00-00 up to and including 00-10-A4-FF-FE-23-FF-FF.

```
EUI64-Address-Mask ::= < AVP Header: XXX >
                       { EUI64-Address }
                       { EUI64-Address-Mask-Pattern }
                     * [ AVP ]
```

5.7.13. EUI64-Address-Mask-Pattern AVP

The EUI64-Address-Mask-Pattern AVP (AVP Code TBD) is of type OctetString. The value is a 8 octets specifying the bit positions of a EUI64 address, that are taken for matching.

5.7.14. VLAN-ID AVP

VLAN-ID AVP (AVP Code TBD) is of type OctetString. The value is a double octet encoded in Network Byte Order. The value of this field specifies the matching value for the IEEE 802.1Q VLAN-ID bits. Only the lower (i.e., rightmost) 12 bits of the specified 2 octet VLAN-ID field are significant; the upper four bits shall be ignored for comparison. If this field is omitted, then comparison of the IEEE

802.1Q VLAN-ID bits for this entry is irrelevant. If this parameter is specified for an entry, then Ethernet packets without IEEE 802.1Q encapsulation shall not match this entry.

5.7.15. Port AVP

The Port AVP (AVP Code TBD) is of type Integer32 in the range of 0 to 65535 and specifies the TCP or UDP port number to match.

5.7.16. Port-Range AVP

The Port-Range AVP (AVP Code TBD) is of type Grouped and specifies an inclusive range of ports.

```
Port-Range ::= < AVP Header: XXX >
               [ Port-Start ]
               [ Port-End ]
             * [ AVP ]
```

If the Port-Start AVP is omitted then port 0 is assumed. If the Port-End AVP is omitted then port 65535 is assumed.

5.7.17. Port-Start AVP

The Port-Start AVP (AVP Code TBD) is of type Integer32 and specifies the first port number of an IP port range.

5.7.18. Port-End AVP

The Port-End AVP (AVP Code TBD) is of type Integer32 and specifies the last port number of an IP port range.

5.7.19. Use-Assigned-Address AVP

In some scenarios, the AAA does not know the IP address assigned to the Managed Terminal at the time that the Classifier is sent to the Classifying Entity. The Use-Assigned-Address AVP (AVP Code TBD) is of type Enumerated containing the values of True or False. When present and set to True, it represents the IP address assigned to the Managed Terminal.

Value	Name
0	False
1	True

5.8. Header Option AVPs

The Classifier AVP may contain one or more of the following AVPs to match on the various possible IP, TCP or ICMP header options.

5.8.1. Diffserv-Code-Point AVP

The Diffserv-Code-Point AVP (AVP Code TBD) is of type Enumerated and specifies the Differentiated Services Field Codepoints to match in the IP header. The values are managed by IANA under the Differentiated Services Field Codepoints registry [DSCP].

5.8.2. Fragmentation-Flag AVP

The Fragmentation-Flag AVP (AVP Code TBD) is of type Enumerated and specifies the packet fragmentation flags to match in the IP header.

Value	Name and Semantic
0	RESERVED
1	Don't Fragment (DF)
2	More Fragments (MF)

5.8.3. IP-Option AVP

The IP-Option AVP (AVP Code TBD) is of type Grouped and specifies an IP header option that must be matched.

```
IP-Option ::= < AVP Header: XXX >
              { IP-Option-Type }
            * [ IP-Option-Value ]
              [ Negated ]
            * [ AVP ]
```

If one or more IP-Option-Value AVPs are present, one of the values MUST match the value in the IP header option. If the IP-Option-Value AVP is absent, the option type MUST be present in the IP header but the value is wild carded.

The Negated AVP is used in conjunction with the IP-Option-Value AVPs to specify IP header options which do not match specific values. The Negated AVP is used without the IP-Option-Value AVP to specify IP headers which do not contain the option type.

5.8.4. IP-Option-Type AVP

The IP-Option-Type AVP (AVP Code TBD) is of type Enumerated and the values are managed by IANA under the IP Option Numbers registry [IPOPTIONS].

5.8.5. IP-Option-Value AVP

The IP-Option-Value AVP (AVP Code TBD) is of type OctetString and contains the option value that must be matched.

5.8.6. TCP-Option AVP

The TCP-Option AVP (AVP Code TBD) is of type Grouped and specifies a TCP header option that must be matched.

```
TCP-Option ::= < AVP Header: XXX >
                 { TCP-Option-Type }
               * [ TCP-Option-Value ]
                 [ Negated ]
               * [ AVP ]
```

If one or more TCP-Option-Value AVPs are present, one of the values MUST match the value in the TCP header option. If the TCP-Option-Value AVP is absent, the option type MUST be present in the TCP header but the value is wild carded.

The Negated AVP is used in conjunction which the TCP-Option-Value AVPs to specify TCP header options which do not match specific values. The Negated AVP is used without the TCP-Option-Value AVP to specify TCP headers which do not contain the option type.

5.8.7. TCP-Option-Type AVP

The TCP-Option-Type AVP (AVP Code TBD) is of type Enumerated and the values are managed by IANA under the TCP Option Numbers registry [TCPOPTIONS].

5.8.8. TCP-Option-Value AVP

The TCP-Option-Value AVP (AVP Code TBD) is of type OctetString and contains the option value that must be matched.

5.8.9. TCP-Flags AVP

The TCP-Flags AVP (AVP Code TBD) is of type Grouped and specifies a set of TCP control flags that must be matched.


```
TCP-Flags ::= < AVP Header: XXX >
              1* { TCP-Flag-Type }
                 [ Negated ]
                 * [ AVP ]
```

If the Negated AVP is not present, the TCP-Flag-Type AVPs specifies which flags MUST be set. If the Negated AVP is present, the TCP-Flag-Type AVPs specifies which flags MUST be cleared.

5.8.10. TCP-Flag-Type AVP

The TCP-Flag-Type AVP (AVP Code TBD) is of type Enumerated and specifies a TCP control flag type that must be matched.

Value	Name and Semantic
0	RESERVED
1	CWR - Congestion Window Reduced.
2	ECE - ECN-Echo. TCP peer is ECN capable.
3	URG - URGent pointer field is significant.
4	ACK - ACKnowledgment field is significant.
5	PSH - Push function.
6	RST - Reset the connection.
7	SYN - Synchronize sequence numbers.
8	FIN - No more data from sender.

5.8.11. ICMP-Type

The ICMP-Type AVP (AVP Code TBD) is of type Grouped and specifies a ICMP message type that must be matched.

```
ICMP-Type ::= < AVP Header: XXX >
              { ICMP-Type-Number }
            * [ ICMP-Code ]
              [ Negated ]
            * [ AVP ]
```

If the ICMP-Code AVP is present, the value MUST match that in the ICMP header. If the ICMP-Code AVP is absent, the ICMP type MUST be present in the ICMP header but the code is wild carded.

The Negated AVP is used in conjunction which the ICMP-Code AVPs to specify ICMP codes which do not match specific values. The Negated AVP is used without the ICMP-Code AVP to specify ICMP headers which do not contain the ICMP type.

5.8.12. ICMP-Type-Number AVP

The ICMP-Type-Number AVP (AVP Code TBD) is of type Enumerated and the values are managed by IANA under the ICMP Type Numbers registry [ICMPTYPE].

5.8.13. ICMP-Code AVP

The ICMP-Code AVP (AVP Code TBD) is of type Enumerated and the values are managed by IANA under the ICMP Type Numbers registry [ICMPTYPE].

5.8.14. ETH-Option AVP

The ETH-Option AVP (AVP Code TBD) is of type Grouped and specifies Ethernet specific classifiers.

```
ETH-Option ::= < AVP Header: XXX >
               { ETH-Proto-Type }
```

```
                * [ ETH-VLAN-ID ]
                * [ ETH-Priority-Range ]
                * [ AVP ]
```

5.8.15. ETH-Proto-Type AVP

The Eth-Proto-Type AVP (AVP Code TBD) is of type Grouped and specifies the encapsulated protocol type. ETH-Ether-Type and ETH-SAP are mutually exclusive.

```
ETH-Proto-Type ::= < AVP Header: XXX >
                   * [ ETH-Ether-Type ]
                   * [ ETH-SAP ]
                   * [ AVP ]
```

5.8.16. ETH-Ether-Type AVP

The ETH-Ether-Type AVP (AVP Code TBD) is of type OctetString. The value is a double octet the contains the value of the Ethertype that the packet shall match in order to match the rule. It might be present in case of DIX or if SNAP is present at 802.2 (SAP shall not be present in this case).

5.8.17. ETH-SAP AVP

The ETH-SAP AVP (AVP Code TBD) is of type OctetString. The value is a double octet representing the 802.2 SAP as specified in "IEEE Standards for Local Area Networks: Logical Link Control". The first

octet contains the DSAP and the second the SSAP.

5.8.18. ETH-Priority-Range AVP

The ETH-Priority-Range AVP (AVP Code TBD) is of type Grouped and specifies a valid priority range in between the Low-priority AVP to the High-priority AVP specified. An Ethernet packet with IEEE 802.1D user_priority value "priority" matches these parameters if priority is greater than or equal to pri-low and priority is less than or equal to pri-high. If this field is omitted, then comparison of the IEEE 802.1D user_priority bits for this entry is irrelevant.

```
ETH-Priority-Range ::= < AVP Header: XXX >
                       * [ ETH-Low-Priority ]
                       * [ ETH-High-Priority ]
                       * [ AVP ]
```

5.8.19. ETH-Low-Priority AVP

The ETH-Low-Priority AVP (AVP Code TBD) is of type OctetString. The value is a single octet with a valid range from 0 to 7.

5.8.20. ETH-High-Priority AVP

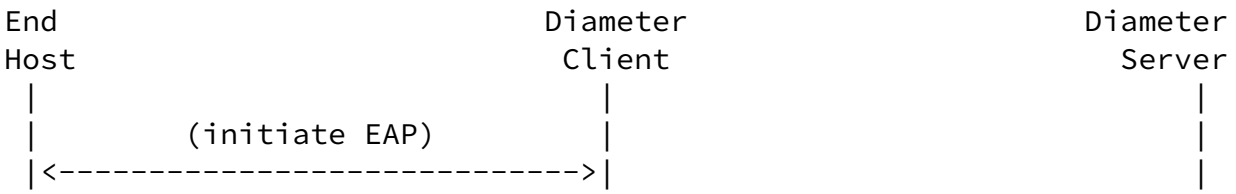
The ETH-High-Priority AVP (AVP Code TBD) is of type OctetString. The value is a single octet with a valid range from 0 to 7.

6. Examples

This section shows a number of signaling flows where QoS negotiation and authorization is part of the conventional NASREQ, EAP or Credit Control applications message exchanges. The signalling flows for the Diameter QoS Application are described in [I-D.ietf-dime-diameter-qos].

6.1. Diameter EAP with QoS Information

Figure 2 shows a simple signaling flow where a NAS (Diameter Client) announces its QoS awareness and capabilities included into the DER message and as part of the access authentication procedure. Upon completion of the EAP exchange, the Diameter Server provides a pre-provisioned QoS profile with the QoS-Semantics in the Extended-QoS-Filter-Rule AVP set to "QoS-Authorized", to the NAS in the final DEA message.



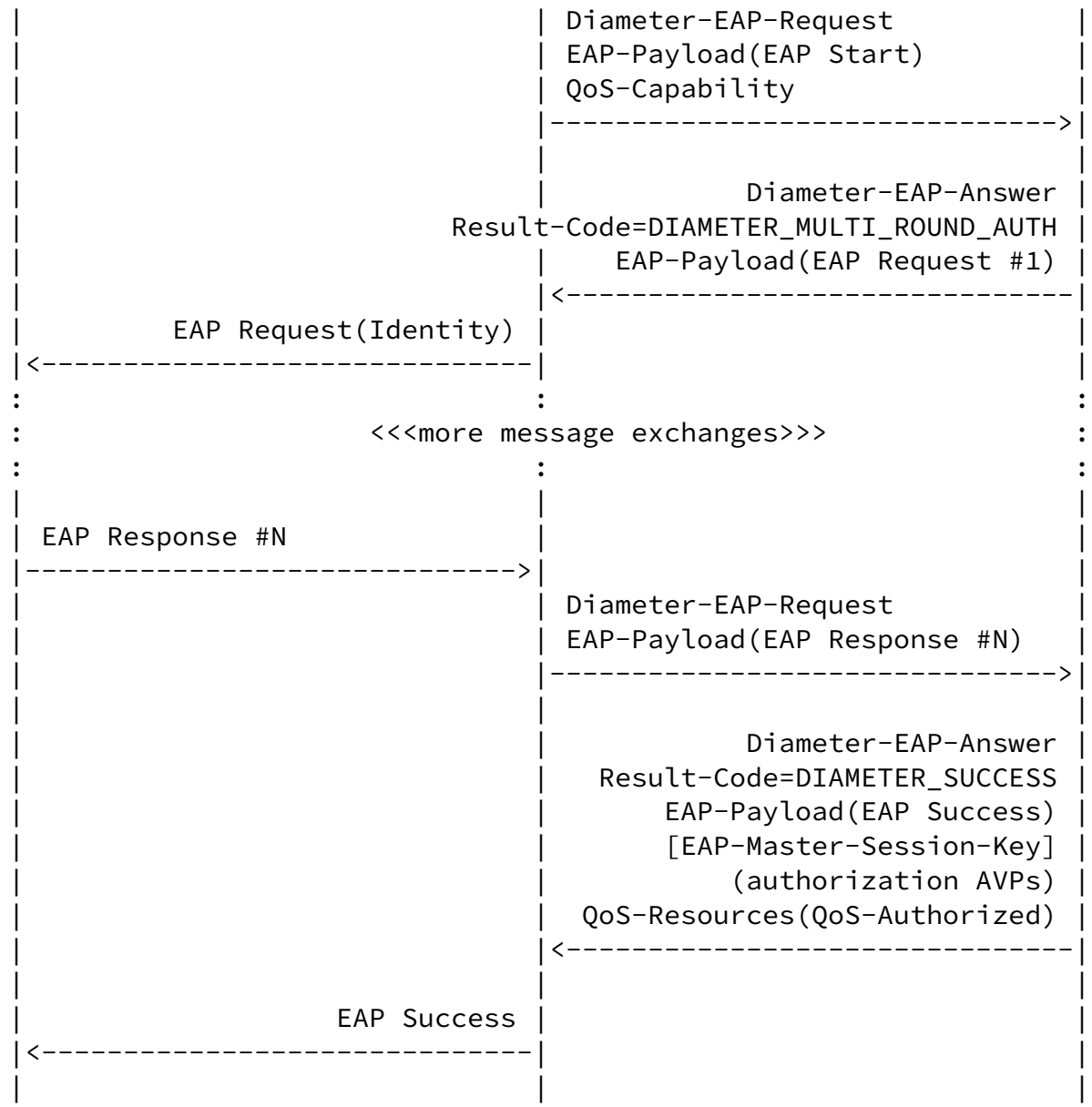


Figure 2: Example of a Diameter EAP enhanced with QoS Information

6.2. Diameter NASREQ with QoS Information

Figure 3 shows a similar pre-provisioned QoS signaling as in Figure 2 but using the NASREQ application instead of EAP application.

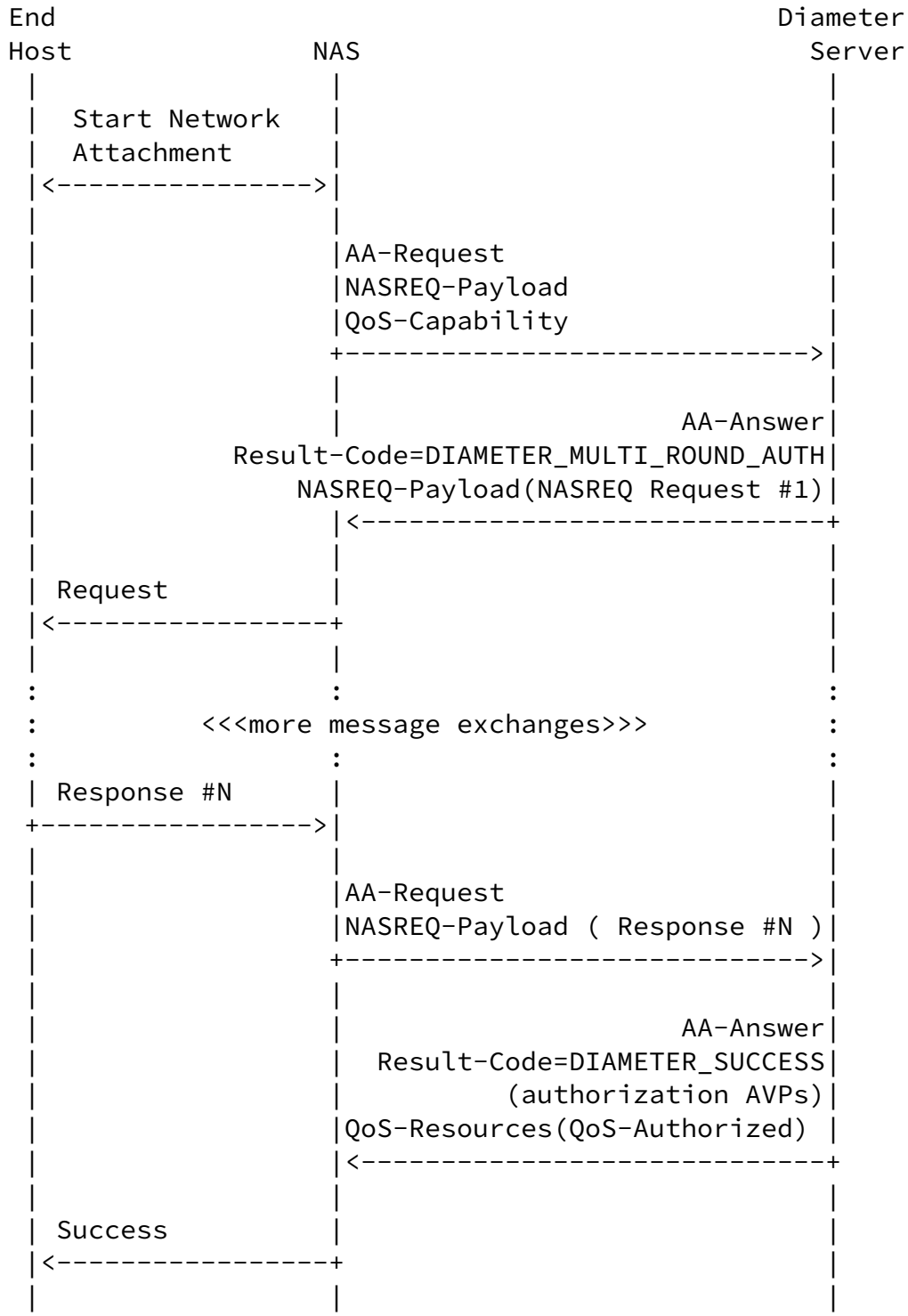


Figure 3: Example of a Diameter NASREQ enhanced with QoS Information

6.3. QoS Authorization

Figure 4 shows an example of authorization only QoS signaling as part of the NASREQ message exchange. The NAS provides the Diameter server with the "QoS-Desired" QoS-Semantics AVP included in the QoS-Resources AVP. The Diameter server then either authorizes the

indicated QoS or rejects the request and informs the NAS about the result. In this scenario the NAS does not need to include the QoS-Capability AVP in the AAR message as the QoS-Resources AVP implicitly does the same and also the NAS is authorizing a specific QoS profile, not a pre-provisioned one.

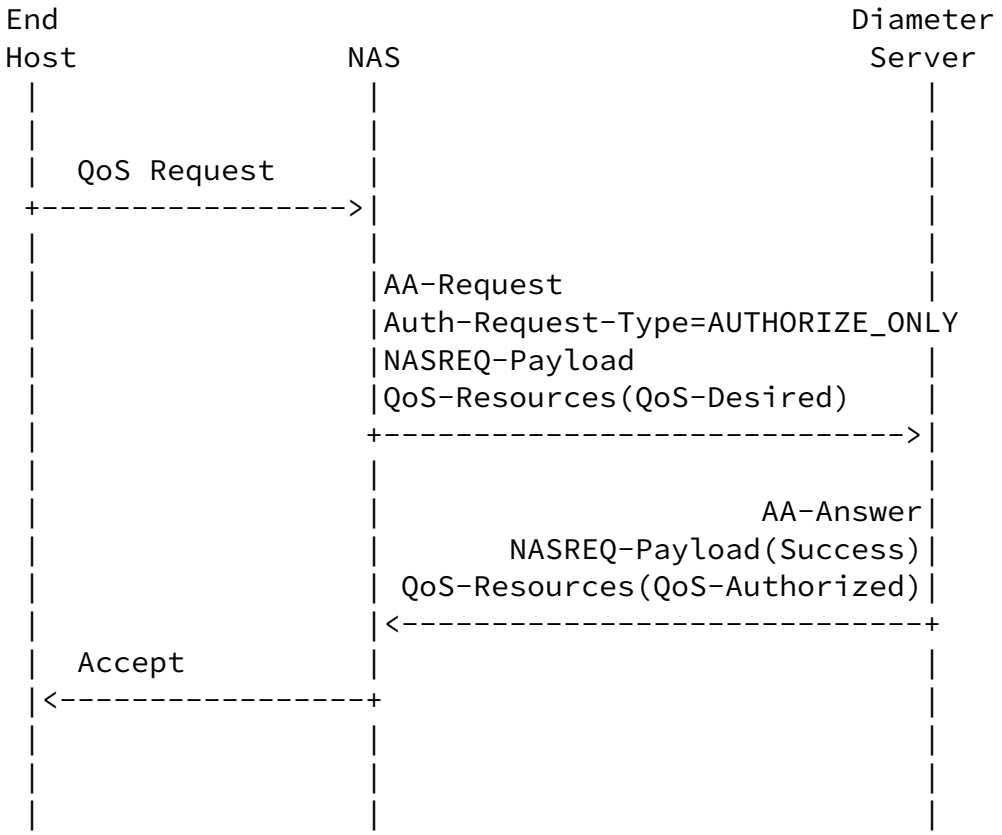


Figure 4: Example of an Authorization-Only Message Flow

6.4. Diameter Server Initiated Re-authorization of QoS

Figure 5 shows a message exchange for a Diameter server initiated QoS re-authorization procedure. The Diameter server sends the NAS a RAR message requesting re-authorization for an existing session and the NAS acknowledges it with a RAA message. The NAS is aware of its existing QoS profile and information for the ongoing session that the Diameter server requested for re-authorization. Thus, the NAS must initiate re-authorization of the existing QoS profile. The re-authorization procedure is the same as in Figure 4.

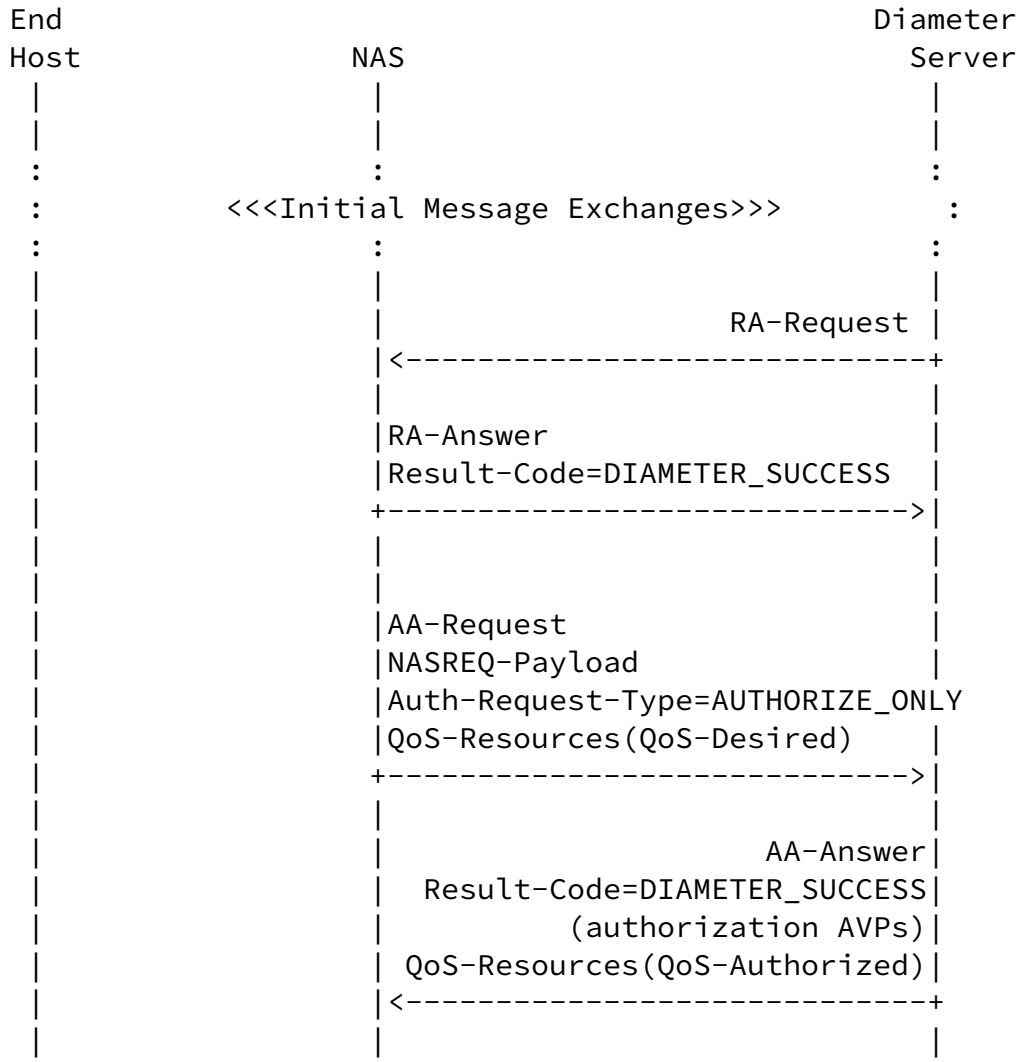


Figure 5: Example of a Server-initiated Re-Authorization Procedure

6.5. Diameter Credit Control with QoS Information

In this case the User is charged as soon as the Service Element (CC client) receives the service request. In this case the client uses

the "QoS-Desired" QoS-Semantics parameter in the QoS-Resources AVP that it sends to the Accounitng server. The server responds with a "QoS-Available" QoS-Semantics parameter in the QoS-Resources AVP

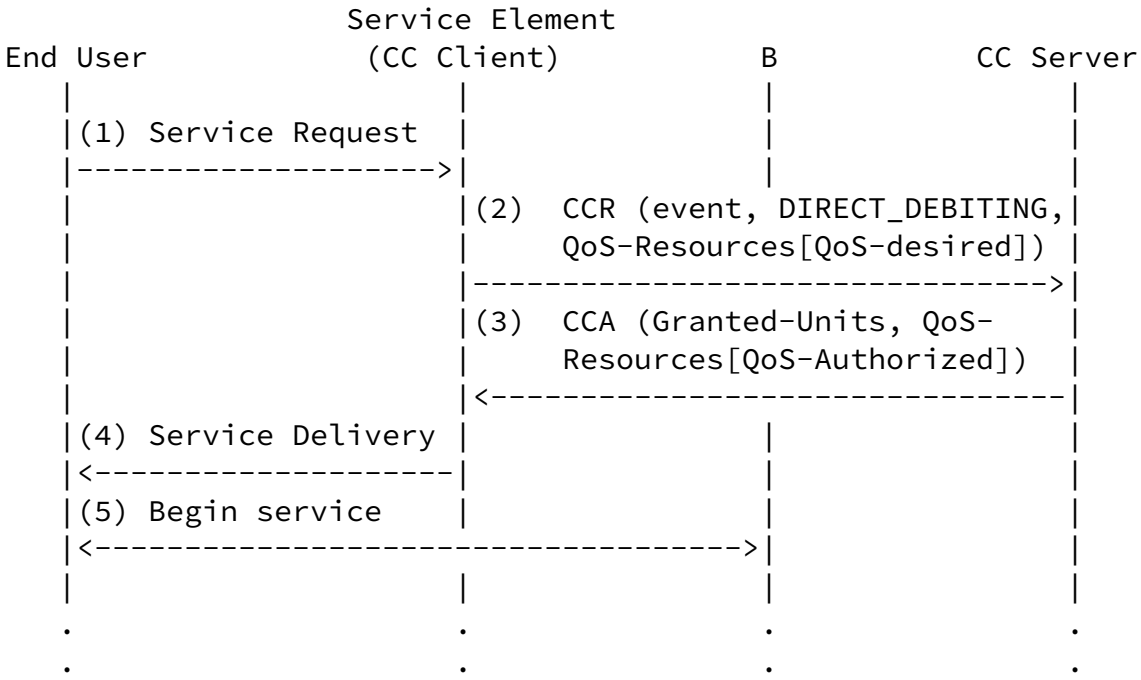


Figure 6: Example for a One-Time Diameter Credit Control Charging Event

6.6. Classifier Examples

Example: Classify all packets from hosts on subnet 12.34.56.00/24 to ports 80, 8090 or 443 on web servers 23.45.67.123, 23.45.68.124, 23.45.69.125.

```
Classifer = {
    Classifier-Id = "web_svr_example";
    Protocol = TCP;
    Direction = OUT;
    From-Spec = {
        IP-Address-Mask = {
            IP-Address = 12.34.56.00;
            IP-Bit-Mask-Width = 24;
        }
    }
    To-Spec = {
        IP-Address = 23.45.67.123;
        IP-Address = 23.45.68.124;
        IP-Address = 23.45.69.125;
        Port = 80;
        Port = 8080;
        Port = 443;
    }
}
```


Example: Any SIP signalling traffic from a device with a MAC address of 01:23:45:67:89:ab to servers with IP addresses in the range 34.56.78.90 to 34.56.78.190.

```
Classifer = {
    Classifier-Id = "web_svr_example";
    Protocol = UDP;
    Direction = OUT;
    From-Spec = {
        MAC-Address = 01:23:45:67:89:ab;
    }
    To-Spec = {
        IP-Address-Range = {
            IP-Address-Start = 34.56.78.90;
            IP-Address-End = 34.56.78.190;
        }
        Port = 5060;
        Port = 3478;
        Port-Range = {
            Port-Start = 16348;
```

```
                Port-End = 32768;
            }
        }
    }
```

7. Acknowledgments

We would like to thank Victor Fajardo, Tseno Tsenov, Robert Hancock, Jukka Manner, Cornelia Kappler, Xiaoming Fu, Frank Alfano,Tolga Asveren, Mike Montemurro,Glen Zorn, Avri Doria, Dong Sun, Tina Tsou, Pete McCann, Georgios Karagiannis and Elwyn Davies for their comments.

8. IANA Considerations

IANA is requested to allocate AVP codes for the following AVPs that are defined in this document.

Attribute Name	AVP Code	Section Defined	Data Type
QoS-Capability	TBD	3.1	Grouped
QoS-Profile-Template	TBD	3.2	Unsigned32

Vendor-Specific-QoS-Profile-Template	TBD	3.3	Grouped
Extended-QoS-Filter-Rule	TBD	3.5	Grouped
QoS-Semantics	TBD	3.6	Enumerated
QoS-Parameters	TBD	3.7	OctetString
QoS-Rule-Precedence	TBD	3.8	Unsigned32
Classifier	TBD	5.1	Grouped
Classifier-ID	TBD	5.2	OctetString
Protocol	TBD	5.3	Enumerated
Direction	TBD	5.4	Enumerated
From-Spec	TBD	5.5	Grouped
To-Spec	TBD	5.6	Grouped
Negated	TBD	5.7.1	Enumerated
IP-Address	TBD	5.7.2	Address
IP-Address-Range	TBD	5.7.3	Grouped

IP-Address-Start	TBD	5.7.4	Address
IP-Address-End	TBD	5.7.5	Address
IP-Address-Mask	TBD	5.7.6	Grouped
IP-Mask-Bit-Mask-Width	TBD	5.7.7	OctetString
MAC-Address	TBD	5.7.8	OctetString
MAC-Address-Mask	TBD	5.7.9	Grouped
MAC-Address-Mask-Pattern	TBD	5.7.10	OctetString
EUI64-Address	TBD	5.7.11	OctetString
EUI64-Address-Mask	TBD	5.7.12	Grouped
EUI64-Address-Mask-Pattern	TBD	5.7.13	OctetString
VLAN-ID	TBD	5.7.14	OctetString
Port	TBD	5.7.15	Integer32
Port-Range	TBD	5.7.16	Grouped
Port-Start	TBD	5.7.17	Integer32
Port-End	TBD	5.7.18	Integer32
Use-Assigned-Address	TBD	5.7.19	Enumerated
Diffserv-Code-Point	TBD	5.8.1	Enumerated
Fragmentation-Flag	TBD	5.8.2	Enumerated
IP-Option	TBD	5.8.3	Grouped
IP-Option-Type	TBD	5.8.4	Enumerated
IP-Option-Value	TBD	5.8.5	OctetString
TCP-Option	TBD	5.8.6	Grouped
TCP-Option-Type	TBD	5.8.7	Enumerated
TCP-Option-Value	TBD	5.8.8	OctetString
TCP-Flags	TBD	5.8.9	Grouped
TCP-Flag-Type	TBD	5.8.10	Enumerated
ICMP-Type	TBD	5.8.11	Grouped
ICMP-Type-Number	TBD	5.8.12	Enumerated
ICMP-Code	TBD	5.8.13	Enumerated
ETH-Option	TBD	5.8.14	Grouped
ETH-Proto-Type	TBD	5.8.15	Grouped
ETH-Ether-Type	TBD	5.8.16	OctetString
ETH-SAP	TBD	5.8.17	OctetString
ETH-Priority-Range	TBD	5.8.18	Grouped

ETH-Low-Priority	TBD	5.8.19	OctetString
ETH-High-Priority	TBD	5.8.20	OctetString

IANA is also requested to allocate a registry for the QoS-Semantics. The following values are allocated by this specification.

(0): QoS-Desired
(1): QoS-Available
(2): QoS-Reserved
(3): Minimum-QoS
(4): QoS-Authorized

A specification is required to add a new value to the registry. A standards track document is required to depreciate, delete, or modify existing values.

9. Security Considerations

This document describes the extension of Diameter for conveying Quality of Service information. The security considerations of the Diameter protocol itself have been discussed in RFC 3588bis [I-D.ietf-dime-rfc3588bis]. Use of the AVPs defined in this document MUST take into consideration the security issues and requirements of the Diameter Base protocol.

10. References

10.1. Normative References

[DSCP] IANA,, "Differentiated Services Field Codepoints", http://www.iana.org/assignments/dscp-registry.

[I-D.ietf-dime-qos-parameters]
Korhonen, J. and H. Tschofenig, "Quality of Service Parameters for Usage with the AAA Framework", draft-ietf-dime-qos-parameters-06 (work in progress), May 2008.

[I-D.ietf-dime-rfc3588bis]
Fajardo, V., Arkko, J., Loughney, J., and G. Zorn, "Diameter Base Protocol", draft-ietf-dime-rfc3588bis-10 (work in progress), January 2008.

[ICMPTYPE]
IANA,, "ICMP Type Numbers",

http://www.iana.org/assignments/icmp-parameters.

[IPOPTIONS]
IANA,, "IP Option Numbers",
http://www.iana.org/assignments/ip-parameters.

[PROTOCOL]
IANA,, "Protocol Types",
http://www.iana.org/assignments/protocol-numbers.

[RFC2119] Bradner, S., "Key words for use in RFCs to Indicate Requirement Levels", BCP 14, RFC 2119, March 1997.

[RFC4005] Calhoun, P., Zorn, G., Spence, D., and D. Mitton, "Diameter Network Access Server Application", RFC 4005, August 2005.

[TCPOPTIONS]
IANA,, "TCP Option Numbers",
http://www.iana.org/assignments/tcp-parameters.

10.2. Informative References

[I-D.ietf-dime-diameter-qos]
Sun, D., McCann, P., Tschofenig, H., Tsou, T., Doria, A., and G. Zorn, "Diameter Quality of Service Application", draft-ietf-dime-diameter-qos-05 (work in progress), February 2008.

Authors' Addresses

Jouni Korhonen
TeliaSonera
Teollisuuskatu 13
Sonera FIN-00051
Finland

Email: jouni.korhonen@teliasonera.com

Hannes Tschofenig
Nokia Siemens Networks
Linnoitustie 6
Espoo 02600
Finland

Phone: +358 (50) 4871445
Email: Hannes.Tschofenig@gmx.net
URI: http://www.tschofenig.priv.at

Mayutan Arumaithurai
University of Goettingen

Email: mayutan.arumaithurai@gmail.com

Mark Jones (editor)
Bridgewater Systems
303 Terry Fox Drive
Ottawa, Ontario K2K 3J1
Canada

Email: mark.jones@bridgewatersystems.com

Avi Lior
Bridgewater Systems
303 Terry Fox Drive, Suite 500
Ottawa, Ontario
Canada K2K 3J1

Phone: +1 613-591-6655
Email: avi@bridgewatersystems.com

Full Copyright Statement

Intellectual Property

copyrights, patents or patent applications, or other proprietary rights that may cover technology that may be required to implement this standard. Please address the information to the IETF at ietf-ipr@ietf.org.