NSLP for Metering Configuration Signaling
<draft-dressler-nsis-metering-nslp-01.txt>

Status of this Memo

This document is an Internet-Draft and is subject to all provisions of Section 3 of RFC 3667. By submitting this Internet-Draft, each author represents that any applicable patent or other IPR claims of which he or she is aware have been or will be disclosed, and any of which he or she become aware will be disclosed, in accordance with RFC 3668.

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.

This Internet-Draft will expire on June 4, 2005.

Copyright Notice

Copyright (C) The Internet Society (2004).

Abstract

Dressler, et al.  draft-dressler-nsis-metering-nslp-01.txt  [Page 1]
Monitoring, metering and accounting of packets are increasingly important functionality that needs to be provided in the Internet. This document proposes the definition of a new NSIS Signaling Layer Protocol (NSLP), named Metering NSLP, which allows the dynamic configuration of Metering Entities on the data path. An accompanying document [I-D.ietf-fessi-nsis-m-nslp-framework] makes a problem statement, describes scenarios for charging, Quality of Service monitoring, and monitoring for network security issues such as intrusion detection, elaborates requirements and discusses the applicability of NSIS to the problem. This document suggests a Metering NSIS protocol design, outlines protocol operation, discusses commonalities and differences to other NSLPs, and defines Metering NSLP messages.

Table of Contents

1. Introduction ........................................... 3
2. Terminology ........................................... 3
3. Basic Protocol Design ................................. 5
   3.1 Model of operation ................................ 5
   3.2 Protocol overview ................................ 7
      3.2.1 Message types ................................ 7
      3.2.2 Design Decisions ............................. 8
   3.3 Examples of operation ............................. 10
   3.4 Implications for GIMPS API ....................... 10
   3.5 Mapping onto M-NSLP Requirements ............... 11
4. M-SPEC ................................................. 12
5. Security considerations ............................. 12
6. Open issues ........................................... 13
7. Acknowledgements ................................... 13
8. References ............................................ 14
   8.1 Normative References .............................. 14
   8.2 Informative References ........................... 14

Authors’ Addresses ....................................... 15

Intellectual Property and Copyright Statements ....... 17
1. Introduction

Monitoring, metering and accounting of packets is an important functionality in many networks today. Several working groups have described mechanisms to collect and report usage data for resource consumption in a network by a particular entity. For example, the IPFIX WG defines a protocol to collect such data. RADIUS (see [RFC2865] and [RFC2866]) and DIAMETER (see [RFC3588] and [I-D.ietf-aaa-diameter-cc]) are also protocols which provide information about consumed resources. The Meter MIB [RFC2720] is a MIB for collecting flow information. However, it is also necessary to configure and coordinate the entities doing the metering. In more complex network topologies and architectures these entities are not only located at the edges of a network. Instead, these Metering Entities are distributed along the data path. While it is possible to configure these entities with protocols such as RADIUS or DIAMETER (or SNMP for the Meter MIB), it is also cumbersome.

Scenarios and requirements for Metering NSLP are described in [I-D.ietf-fessi-nsis-m-nslp-framework].

This draft introduces a new NSLP protocol, the Metering NSLP, for configuration and coordination of Metering Entities in a path-coupled fashion.

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 [RFC2119].

Furthermore, this document uses the following terms:

Metering Data

Metering Data describe utilized resources concerning a particular flow or service for a later charging process. Examples for such data are packet counter, time information, and information describing users or hosts.

Metering Record

A Metering Record represents aggregated and/or correlated Metering Data.
Monitoring Probe

A monitoring probe is an entity that examines the data flow in order to gather Metering Data. This Metering Data is exported to an Metering Entity.

Metering Entity

An Metering Entity produces Metering Data describing the resource utilization of a particular flow or service. Typically, this information is collected from associated monitoring probes.

Collector

A collector receives Metering Data from one or multiple Metering Entities. This Metering Data is aggregated, correlated, and stored in form of Metering Records.

Metering Configuration State

State used/kept by the Metering Manager to configure the Metering Entity and Monitoring Probes.

Metering Manager

A unit in the Metering Entity that communicates with M-NSLP processing. It holds Metering Configuration State which is used to Configure Monitoring Probes and the Metering Entity.

MNE

An NSIS Entity (NE) which supports the Metering NSLP.

MNI

The first node in the sequence of MNEs that issues a configuration message for a flow or aggregate.

MNR

The last node in the sequence of MNEs that receives a configuration message for a flow or aggregate.

MNF

A MNE that is neither MNI nor MNR.
3. Basic Protocol Design

The basic design of a Metering NSLP and the processing of Metering NSLP messages is quite similar to QoS NSLP. In fact much of the subsequent text is modeled after the corresponding text in [I-D.ietf-nsis-qos-nslp]. The main difference compared to QoS NSLP is that Metering NSLP allows individual Metering NEs (MNEs) to pull out of a signaling session.

3.1 Model of operation

Figure 1 shows an example logical model of the operation of Metering NSLP and the associated metering mechanism in a MNE.
In this example, the MNE is collocated on a single node with a Metering Entity and one Monitoring Probe (MP, cf. [I-D.ietf-fessi-nsis-m-nslp-framework]). The MP collects Measurement Data which is handed to the Metering Entity where it is processed to become Metering Data. From the Metering Entity, Metering Data is sent to a Collector.

A M-NSLP message transports metering configuration information. This information is extracted by M-NSLP Processing and passed to the Metering Manager, where it is interpreted and used to install Metering Configuration State. The Metering Manager uses this state to configure the Metering Entity and the Monitoring Probe. The Policy Control determines whether the sender of the M-NSLP message has administrative rights to configure the Metering Entity.
From the perspective of a single node, the request for configuration of Metering Entities may result from processing a local management request, or from processing an incoming M-NSLP message. The local management may in turn be triggered by a local application, e.g. a video being requested from a video server, or by a physically separate knowledgeable network node.

3.2 Protocol overview

3.2.1 Message types

The Metering NSLP uses four message types:

**CONFIGURE**

The CONFIGURE message is the only message that manipulates M-NSLP configuration state. It is used to create, refresh, modify and remove such state. The CONFIGURE message is idempotent; the resultant effect is the same whether a message is received once or many times. In fact this message is equivalent to the QoS NSLP RESERVE message.

**QUERY**

A QUERY message is used to request information about the current configuration without changing it. The QUERY message is impotent, because it does not cause any state to be installed or modified. It is equivalent to the QoS NSLP QUERY message.

**RESPONSE**

The RESPONSE message is used to provide information about the result of a previous M-NSLP message. This includes explicit confirmation of the state manipulation signaled in the CONFIGURE message, the response to a QUERY message or an error code if a MNE is unable to provide the requested information or if the response is negative. The RESPONSE message is impotent, and equivalent to the QoS NSLP RESPONSE message.

**NOTIFY**

NOTIFY messages are used to convey information to a MNE. They differ from RESPONSE messages in that they are sent asynchronously and need not refer to any particular state or previously received message. The information conveyed by a NOTIFY message is typically related to error conditions. An example would be notification to an upstream peer about state being torn. Obviously it is equivalent to the QoS NSLP NOTIFY message.
Each protocol message has a common header which indicates the message type and contains flags. Metering NSLP messages contain three types of objects:

Control Information

Control information objects carry general information for the M-NSLP Processing, such as sequence numbers or whether a response is required.

Metering specification (MSpecs)

MSpec objects describe the actual Configuration information. They are interpreted in the Metering Manager and opaque to M-NSLP Processing.

Policy objects

Policy objects contain data used to authorize the configuration of the MNEs. They are interpreted by Policy Control.

3.2.2 Design Decisions

3.2.2.1 Soft State

NSIS State is always soft state and needs to be refreshed. The Control Information carries an object that determines the life time of M-NSLP state. It is for further study whether life time of M-NSLP state for a particular flow must be the same for all MNEs along the signaling path as in NATFW-NSLP [I-D.ietf-nsis-nslp-natfw], or whether it is a decision local to pairs of MNEs as in QoS-NSLP. In fact the refreshment mechanism depends on the authorization model. Currently, it is expected that the the authorization model follows that in QoS NSLP (‘New Jersey Turnpike’, i.e. a chain-of-trust is established by peering relationships between neighboring networks / entities). Therefore, the refreshing model probably will be similar to that of QoS NSLP.

3.2.2.2 Message Sequencing

The order in which CONFIGURE messages arrive influences the eventual reservation state that will be stored at a MNE. Therefore M-NSLP supports the detection of CONFIGURE message duplication by means of a Configuration Sequence Number (CSN), which corresponds to the Reservation Sequence Number (RSN) of QoS-NSLP.
3.2.2.3 Message Scoping

In order to realize the requirement of scoping of M-NSLP messages up to certain types of Metering Entities, it is necessary to have an abstract language that can name Metering Entity types. This information travels in the MSpec and is hence interpreted in the Metering Management. When the Metering Management recognizes it is responsible for a Metering Entity of the type specified, it informs M-NSLP processing to terminate the signaling.

3.2.2.4 Rerouting

M-NSLP automatically adapts to rerouting events because state along the old path times out, and a refreshing CONFIGURE message will install state along the new path. In QoS NSLP it is necessary to detect a rerouting event in order establish state on the new path, and to tear down reservations on the old path. To this end, QoS NSLP introduces an additional object, the SII. When the SII received by a QNE changes, a rerouting event has occurred.

For M-NSLP it is generally not important to quickly tear down configuration state along the old path, however for some applications, e.g. charging, it is vital to quickly configure MNEs on the new path. Therefore an object equivalent to the SII is needed.

3.2.2.5 Selection of MNEs

An interesting feature of M-NSLP is that only a subset of MNEs on the data path might take part in the actual metering. Metering Entities taking part in the metering process are determined based, for example, on their type or number. This feature is the most striking difference to QoS NSLP with a number of implications.

When the first CONFIGURE message is sent, each MNE on the data path needs to determine whether it should take part in the metering process or not by inspecting the MSpec object. Here we can reuse the ability from above to abstractly name types of Metering Entities. When the MNE finds out it is not involved in the metering it should remove itself from the signaling session for this flow. This however implies it is difficult to later update this signaling session to again include the MNE, simply because it is not going to read the updating CONFIGURE message. See the discussion on implications for the GIMPS API in Section 3.4.

3.2.2.6 Aggregation

The metering configuration should allow aggregation of Metering Data
belonging to the same user or application. Such aggregation can be done in two ways.

- A Metering Entity separately collects and reports data for each micro flow (e.g., given for all different combinations of port numbers) contained in the macro flow that is signalled by the NTLP.
- A Metering Entity separately collects microflows but reports all flows in a single record.

### 3.3 Examples of operation

The basic signaling sequences are very similar to QoS NSLP: To start a configuration, the MNI constructs a CONFIGURE message with a MSpec object, and sends it to the MNR. The message is interpreted by MNEs on the data path. The MNR replies with a RESPONSE message.

```
+----+ CONFIGURE +----+ CONFIGURE +----+ CONFIGURE +----+
|   |------------|   |------------|   |------------|   |
|MNI|             |MNF|             |MNF|             |MNR|
|   |<------------|   |<------------|   |<------------|   |
+----+   RESPONSE +----+   RESPONSE +----+   RESPONSE +----+
```

Similarly, a QUERY message can be initiated by MNI or MNR, travels to the MNR resp. MNI where a RESPONSE is issued and sent back. It needs to be investigated whether there is a use case for enabling any MNE to issue a QUERY which in this case would need to be sent both upstream and downstream.

```
+----+ QUERY +----+ QUERY +----+ QUERY +----+
|   |------------|   |------------|   |------------|   |
|MNI|             |MNF|             |MNF|             |MNR|
|   |<------------|   |<------------|   |<------------|   |
+----+   RESPONSE +----+   RESPONSE +----+   RESPONSE +----+
```

### 3.4 Implications for GIMPS API

In [I-D.ietf-nsis-ntlp], an API is defined between GIMPS and NSLPs. The requirement to flexibly select what Metering Entities become part of a given signaling session implies requirements on the API that may currently not be covered.

1. When a given MNE x discovers it is not part of a signaling session it needs to be able to tell GIMPS to not install message routing state. This needs to imply that the next MNE downstream (which wants to participate in the signaling session) does not
invoke a Messaging Association with MNE_x but rather with the
next MNE upstream that participates in the session. In fact
something similar is also necessary for supporting stateless /
reduced state NSIS Entities. Therefore the API already defines a
message that asks GIMPS to not retain state. Whether this is
sufficient to cover M-NSLP requirements still needs to be worked
out in more detail.

2. The list of MNEs participating in a particular metering session
may be changed, particularly more, or other, types of Metering
Entities may have to be included. These entities however do not
participate currently in the ongoing signaling session.
Therefore means must be provided to include them at a later
point. One possibility is for M-NSLP to tell GIMPS a rerouting
event occurred, and message routing state needs to be updated.
This would prompt GIMPS to rediscover peers. However care needs
to be taken that GIMPS doesn’t use this information to warn other
NSLPs about the assumed rerouting. Also here the problem and
possible solutions still need to be analyzed in more detail.

3.5 Mapping onto M-NSLP Requirements

With the design described above, the requirements from
[I-D.ietf-fessi-nsis-m-nslp-framework] are at this point satisfied as
follows:

- Extensibility. The actual configuration information is
  encapsulated in the MSpec. Depending on how flexible the MSpec is
designed this requirement can be satisfied. Furthermore, M-NSLP
needs to be flexible regarding the message sequencing that is
possible. The current design is still open in that respect.
- Interoperability. Again, whether different accounting solutions
can interwork depends on how the MSpec is designed. In QoS NSLP,
the QSpec template design [I-D.ietf-nsis-qspec] aims at similar
extensibility and interoperability. It needs to be studied
whether or not the solutions chosen by the QSpec can also be
applied to the MSpec.
- Flexible metering models. As above, this is an issue of MSpec
design, and flexibility of message sequencing.
- Distinguishing flows. The aggregation feature detailed in this
  requirement can be realized as described in Section 3.2.2.6.
- Flexible data collection. Another issue that needs to be fixed in
  the MSpec.
- Location of Metering Entities. Location of Metering Entities.
  MNEs, including MNI and MNR can be located anywhere on the data
  path.
- Access parameters of the Collector Information. Access parameters
  of the Collector Information on how to deliver flow records to the
  Collector is coded in the MSpec.
o Configuration of Metering Entities. The protocol can configure Metering Entities that are MNEs, or that are controlled by MNEs.

- Selection of Metering Entities. As described in Section 3.2.2.5, a MNE should be able to decide to pull out of the metering process. How this is realized regarding the interaction between M-NSLP and GIMPS is not yet clarified in detail. Furthermore, an MSpec template must provide a language to abstractly describe types of Metering Entities that are (not) to become part of the metering process.

- Metering Configuration State installation and removal. By means of the CONFIGURE message, the protocol can install, refresh and remove Metering Configuration State.

- Initiation and maintenance of metering tasks. Triggers and correlation identifier are transported in the MSpec. The protocol implicitly reacts to rerouting because a refreshing CONFIGURE message installs state along the new route, as described in Section 3.2.2.4.

- Collection of information on available Metering Entities. This can be achieved by means of the QUERY message.

- Scoping of configuration. The MSpec needs to provide sufficient means for flexible scoping signaling messages.

Requirements not mentioned in this list are not yet addressed.

4. M-SPEC

   compilation of M-SPEC parameters analyzed in the Metering NSLP Scenarios Draft

5. Security considerations

   The process of configuring entities to start and stop metering and to transmit collected resource records to a third party introduces security challenges.

   First, the application domain needs to be considered. If a malicious user is capable of stop metering of requested services then fraud is possible. It must not be possible to configure Metering Entities in such a way that other users are charged for the usage of a service which they have not used.

   Second, interworking between multiple domains causes authorization problems. For example, network domain A might want to collect resource records in network domain B to offer the user with a more consistent bill covering both the price of the network resource consumption and the application usage. A high degree of trust is required to allow other domains to configure Metering Entities and to collect the resource usage of particular users. In any case it needs
to be prevented that arbitrary resource records associated with users
are collected by other domains. It has to be noted that the process
of charging involves other states than only the collection of usage
records.

Third, it must be avoided that a denial of service attack is mounted
on either Collectors or Metering Entities. Metering Entities can be
subject to DoS attacks if a large number of resource have to be
collected or ‘unlimited’ per-flow states are created. Collectors can
be subject to DoS attacks if they are flooded with Metering Records.

The introduced mechanisms allow a number of entities to configure
metering entities. This might introduce some weaknesses compared to
a centralized approach where a single entity (or a few selected
entities) are authorized to perform this action. The authorization
configuration of a decentralized approach is more difficult to secure
since a single malicious entity is able to start/stop/modify the
process of Metering Record collection within a single domain or even
beyond this domain.

6. Open issues

Details need to be worked out how the configuration information in
the MSpec is expressed, and how it is interpreted.

For example, the MSpec could specify exactly one Metering Entity
of a particular type X, e.g. one that is able to measure bandwidth
received, should participate in the metering. This implies
1. the first MNE (MNE_X) on the signaling path being of type X
should volunteer to take on this task
2. MNE_X needs to modify the MSpec to signal to subsequent MNEs that
a Metering Entity of type X has already been found.

However, appropriate action needs to be taken if the signaling
arrives at the MNR and no Metering Entity of type X was found.
Furthermore, when a rerouting event occurs, and MNE_X is no longer on
the signaling path, this needs to be detected, and a replacement must
be found. Support of such functionality is not necessary in QoS
NSLP. It is possible that on this basis more design differences
between QoS NSLP and M-NSLP will be detected in the future.

Additional open issues appear in the main body of the text.

7. Acknowledgements

The authors would like to thank Robert Hancock for valuable input.
8. References

8.1 Normative References


8.2 Informative References


[I-D.ietf-nsis-qspec]

[I-D.ietf-nsis-ntlp]

[I-D.ietf-fessi-nsis-m-nslp-framework]

[I-D.ietf-aaa-diameter-cc]

[TS32.240]

Authors' Addresses

Falko Dressler
University of Erlangen
Department of Computer Science 7
Martensstr. 3
Erlangen 91058
Germany

Phone: +49 9131 85-27914
Email: dressler@informatik.uni-erlangen.de
URI: http://www7.informatik.uni-erlangen.de/
Georg Carle
University of Tuebingen
Wilhelm-Schickard-Institute for Computer Science
Auf der Morgenstelle 10C
Tuebingen 71076
Germany
Phone: +49 7071 29-70505
Email: carle@informatik.uni-tuebingen.de
URI: http://net.informatik.uni-tuebingen.de/

Juergen Quittek
NEC
Kurfuersten-Anlage 36
Heidelberg 69115
Germany
Phone: +49 6221 90511-15
Email: quittek@netlab.nec.de
URI: http://www.netlab.nec.de/

Cornelia Kappler
Siemens AG
Siemensdamm 62
Berlin 13627
Germany
Phone: +49 30 386-32894
Email: cornelia.kappler@siemens.com

Hannes Tschofenig
Siemens AG
Otto-Hahn-Ring 6
Munich, Bayern 81739
Germany
Email: Hannes.Tschofenig@siemens.com
Intellectual Property Statement

The IETF takes no position regarding the validity or scope of any Intellectual Property Rights or other rights that might be claimed to pertain to the implementation or use of the technology described in this document or the extent to which any license under such rights might or might not be available; nor does it represent that it has made any independent effort to identify any such rights. Information on the procedures with respect to rights in RFC documents can be found in BCP 78 and BCP 79.

Copies of IPR disclosures made to the IETF Secretariat and any assurances of licenses to be made available, or the result of an attempt made to obtain a general license or permission for the use of such proprietary rights by implementers or users of this specification can be obtained from the IETF on-line IPR repository at http://www.ietf.org/ipr.

The IETF invites any interested party to bring to its attention any 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.

Disclaimer of Validity

This document and the information contained herein are provided on an "AS IS" basis and THE CONTRIBUTOR, THE ORGANIZATION HE/SHE REPRESENTS OR IS SPONSORED BY (IF ANY), THE INTERNET SOCIETY AND THE INTERNET ENGINEERING TASK FORCE DISCLAIM ALL WARRANTIES, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO ANY WARRANTY THAT THE USE OF THE INFORMATION HEREIN WILL NOT INFRINGE ANY RIGHTS OR ANY IMPLIED WARRANTIES OF MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE.

Copyright Statement

Copyright (C) The Internet Society (2004). This document is subject to the rights, licenses and restrictions contained in BCP 78, and except as set forth therein, the authors retain all their rights.

Acknowledgment

Funding for the RFC Editor function is currently provided by the Internet Society.