              IP Flow Information Export (IPFIX) Information Elements
                        for Logging NAT Events

Abstract

   Network operators require NAT devices to log events like creation and
   deletion of translations and information about the resources that the
   NAT device is managing. In many cases, the logs are essential to
   identify an attacker or a host that was used to launch malicious
   attacks and for various other purposes of accounting. Since there is
   no standard way of logging this information, different NAT devices
   use proprietary formats; hence, it is difficult to expect consistent
   behavior. This lack of standardization makes it difficult to write
   the Collector applications that would receive this data and process
   it to present useful information. This document describes the
   formats for logging NAT events.

Status of This Memo

   This is an Internet Standards Track document.

   This document is a product of the Internet Engineering Task Force
   (IETF). It represents the consensus of the IETF community. It has
   received public review and has been approved for publication by the
   Internet Engineering Steering Group (IESG). Further information on
   Internet Standards is available in Section 2 of RFC 7841.

   Information about the current status of this document, any errata,
   and how to provide feedback on it may be obtained at
   https://www.rfc-editor.org/info/rfc8158.

Copyright Notice

Table of Contents

1. Introduction

   The IP Flow Information Export (IPFIX) Protocol [RFC7011] defines a
   generic push mechanism for exporting information and events. The
   IPFIX Information Model [IPFIX-IANA] defines a set of standard
   Information Elements (IEs) that can be carried by the IPFIX protocol.
   This document details the IPFIX IEs that MUST be logged by a NAT
   device that supports NAT logging using IPFIX and all the optional
   fields. The fields specified in this document are gleaned from
   [RFC4787] and [RFC5382].

   This document and [NAT-LOG] are written in order to standardize the
   events and parameters to be recorded using IPFIX [RFC7011] and SYSLOG
   [RFC5424], respectively. This document uses IPFIX as the encoding
   mechanism to describe the logging of NAT events. However, the
   information that is logged should be the same irrespective of what
   kind of encoding scheme is used. IPFIX is chosen because it is an
   IETF standard that meets all the needs for a reliable logging
   mechanism. IPFIX provides the flexibility to the logging device to
   define the datasets that it is logging. The IEs specified for
   logging must be the same irrespective of the encoding mechanism used.

1.1. Terminology

   The term "NAT device" in this document refers to any NAT44 or NAT64
   device. The term "Collector" refers to any device that receives
   binary data from a NAT device and converts it into meaningful
   information. This document uses the term "session" as defined in
   [RFC2663], and the term "Binding Information Base" (BIB) as defined
   in [RFC6146]. The term "Information Element" or "IE" is defined in
   [RFC7011]. The term "Carrier-Grade NAT" refers to a large-scale NAT
   device as described in [RFC6888]

   The IPFIX IEs that are NAT specific are created with NAT terminology.
   In order to avoid creating duplicates, IEs are reused if they convey
   the same meaning. This document uses the term "timestamp" for the
   IE, which defines the time when an event is logged; this is the same
   as the IPFIX term "observationTimeMilliseconds" as described in
   [IPFIX-IANA]. Since observationTimeMilliseconds is not self-
   explanatory for NAT implementors, the term "timeStamp" is used.
   Event templates, which refer to IPFIX Template Records, as well as
   log events, which refer to IPFIX Flow Records, are also used in this
   document.

1.2. Requirements Language

   The key words "MUST", "MUST NOT", "REQUIRED", "SHALL", "SHALL NOT",
   "SHOULD", "SHOULD NOT", "RECOMMENDED", "NOT RECOMMENDED", "MAY", and
   "OPTIONAL" in this document are to be interpreted as described in
   BCP 14 [RFC2119] [RFC8174] when, and only when, they appear in all
   capitals, as shown here.

2. Scope

   This document provides the information model to be used for logging
   the NAT events, including Carrier-Grade NAT (CGN) events. [RFC7011]
   provides guidance on the choices of the transport protocols used for
   IPFIX and their effects. This document does not provide guidance on
   transport protocols like TCP, UDP, or Stream Control Transmission
   Protocol (SCTP), which are to be used to log NAT events. The logs
   SHOULD be reliably sent to the Collector to ensure that the log
   events are not lost. The choice of the actual transport protocol is
   beyond the scope of this document.

   This document uses the allocated IPFIX IEs in the IANA "IPFIX
   Information Elements" registry [IPFIX-IANA] and registers some new
   ones.

   This document assumes that the NAT device will use the existing IPFIX
   framework to send the log events to the Collector. This would mean
   that the NAT device will specify the template that it is going to use
   for each of the events. The templates can be of varying length, and
   there could be multiple templates that a NAT device could use to log
   the events.

   The implementation details of the Collector application are beyond
   the scope of this document.

   The optimization of logging the NAT events is left to the
   implementation and is beyond the scope of this document.

3. Deployment

   NAT logging based on IPFIX uses binary encoding; hence, it is very
   efficient. IPFIX-based logging is recommended for environments where
   a high volume of logging is required, for example, where per-flow
   logging is needed or in case of Carrier-Grade NAT. However, IPFIX-
   based logging requires a Collector that processes the binary data and
   requires a network management application that converts this binary
   data to a human-readable format.

A Collector may receive NAT events from multiple CGN devices. The
Collector distinguishes between the devices using the source IP
address, source port, and Observation Domain ID in the IPFIX header.
The Collector can decide to store the information based on the
administrative policies that are in line with the operator and the
local jurisdiction. The retention policy is not dictated by the
Exporter and is left to the policies that are defined at the
Collector.

A Collector may have scale issues if it is overloaded by a large
number of simultaneous events. An appropriate throttling mechanism
may be used to handle the oversubscription.

The logs that are exported can be used for a variety of reasons. An
example use case is to do accounting based on when the users logged
on and off. The translation will be installed when the user logs on
and removed when the user logs off. These events create log records.
Another use case is to identify an attacker or a host in a provider
network. The network administrators can use these logs to identify
the usage patterns, the need for additional IP addresses, and etc.
The deployment of NAT logging is not limited to just these cases.

4. Event-Based Logging

An event in a NAT device can be viewed as a state transition because
it relates to the management of NAT resources. The creation and
deletion of NAT sessions and bindings are examples of events, as they
result in resources (addresses and ports) being allocated or freed.
The events can happen through the processing of data packets flowing
through the NAT device, through an external entity installing
policies on the NAT router, or as a result of an asynchronous event
like a timer. The list of events is provided in Table 2. Each of
these events SHOULD be logged, unless this is administratively
prohibited. A NAT device MAY log these events to multiple Collectors
if redundancy is required. The network administrator will specify
the Collectors to which the log records are to be sent. It is
necessary to preserve the list of Collectors and its associated
information like the IPv4/IPv6 address, port, and protocol across
reboots so that the configuration information is not lost when the
device is restarted. The NAT device implementing the IPFIX logging
MUST follow the IPFIX specification in [RFC7011].

4.1. Logging Destination Information

Logging destination information in a NAT event is discussed in
[RFC6302] and [RFC6888]. Logging destination information increases
the size of each record and increases the need for storage
considerably. It increases the number of log events generated

because when the same user connects to a different destination, it
results in a log record per destination address. Logging the source
and destination addresses results in loss of privacy. Logging of
destination addresses and ports, pre- or post-NAT, SHOULD NOT be done
[RFC6888]. However, this document provides the necessary fields to
log the destination information in cases where they must be logged.

4.2. Information Elements

   The templates could contain a subset of the IEs shown in Table 1,
   depending upon the event being logged. For example, a NAT44 session
   creation template record will contain:

   {sourceIPv4Address, postNATSourceIPv4Address, destinationIPv4Address,
   postNATDestinationIPv4Address, sourceTransportPort,
   postNAPTSourceTransportPort, destinationTransportPort,
   postNAPTDestinationTransportPort, internalAddressRealm, natEvent,
   timeStamp}

   An example of the actual event data record is shown below in a human-
   readable form:

   {192.0.2.1, 203.0.113.100, 192.0.2.104, 192.0.2.104, 14800, 1024, 80,
   80, 0, 1, 09:20:10:789}

   A single NAT device could be exporting multiple templates, and the
   Collector MUST support receiving multiple templates from the same
   source.

The following table includes all the IEs that a NAT device would need
to export the events. The formats of the IEs and the IPFIX IDs are
listed. Detailed descriptions of the fields natInstanceID,
internalAddressRealm, externalAddressRealm, natQuotaExceededEvent,
and natThresholdEvent are included in the IANA Considerations
section.

Field Name	Size (bits)	IANA IPFIX ID	Description
timeStamp	64	323	System Time when the event occurred
natInstanceID	32	463	NAT Instance Identifier
vlanId	16	58	VLAN ID in case of overlapping networks
ingressVRFID	32	234	VRF ID in case of overlapping networks
sourceIPv4Address	32	8	Source IPv4 Address
postNATSourceIPv4Address	32	225	Translated Source IPv4 Address
protocolIdentifier	8	4	Transport protocol
sourceTransportPort	16	7	Source Port
postNAPTSourceTransportPort	16	227	Translated Source port
destinationIPv4Address	32	12	Destination IPv4 Address

postNATDestinationIPv4Address	32	226	Translated IPv4 destination address
destinationTransportPort	16	11	Destination port
postNAPTDestinationTransportPort	16	228	Translated Destination port
sourceIPv6Address	128	27	Source IPv6 address
destinationIPv6Address	128	28	Destination IPv6 address
postNATSourceIPv6Address	128	281	Translated source IPv6 address
postNATDestinationIPv6Address	128	282	Translated Destination IPv6 address
internalAddressRealm	(*)	464	Source Address Realm
externalAddressRealm	(*)	465	Destination Address Realm
natEvent	8	230	Type of Event
portRangeStart	16	361	Allocated port block start
portRangeEnd	16	362	Allocated Port block end
natPoolId	32	283	NAT pool Identifier

natQuotaExceededEvent	32	466	Limit event identifier
natThresholdEvent	32	467	Threshold event identifier
maxSessionEntries	32	471	Maximum session entries
maxBIBEntries	32	472	Maximum bind entries
maxEntriesPerUser	32	473	Maximum entries per-user
maxSubscribers	32	474	Maximum subscribers
maxFragmentsPendingReassembly	32	475	Maximum fragments for ressembly
addressPoolHighThreshold	32	476	High threshold for address pool
addressPoolLowThreshold	32	477	Low threshold for address pool
addressPortMappingHighThreshold	32	478	High threshold for address/port mapping
addressPortMappingLowThreshold	32	479	Low threshold for address/port mapping

addressPortMappingPerUserHighThre shold	32	480	High threshold for per-user address/port mapping
globalAddressMappingHighThreshold	32	481	High threshold for global address mapping

Note: (*) indicates octetArray

Table 1: NAT IE List

4.3. Definition of NAT Events

The following is the complete list of NAT events and the proposed
event type values. The natEvent IE is defined in the "IPFIX
Information Elements" registry [IPFIX-IANA];. The list can be
expanded in the future as necessary. The data record will have the
corresponding natEvent value to indicate the event that is being
logged.

Note that the first two events are marked "Historic" and are listed
here for the sole purpose of completeness. Any compliant
implementation SHOULD NOT use the events that are marked "Historic".
These values were defined prior to the existence of this document and
outside the IETF. These events are not standalone and require more
information to be conveyed to qualify the event. For example, the
NAT translation create event does not specify if it is NAT44 or
NAT64. As a result, the Behave working group decided to have an
explicit definition for each one of the unique events.

```
+-------+----------------------------------+
| Value | Event Name                       |
+-------+----------------------------------+
| 0     | Reserved                         |
| 1     | NAT translation create (Historic)|
| 2     | NAT translation delete (Historic)|
| 3     | NAT Addresses exhausted          |
| 4     | NAT44 session create             |
| 5     | NAT44 session delete             |
| 6     | NAT64 session create             |
| 7     | NAT64 session delete             |
| 8     | NAT44 BIB create                 |
| 9     | NAT44 BIB delete                 |
| 10    | NAT64 BIB create                 |
| 11    | NAT64 BIB delete                 |
| 12    | NAT ports exhausted              |
| 13    | Quota Exceeded                   |
| 14    | Address binding create           |
| 15    | Address binding delete           |
| 16    | Port block allocation            |
| 17    | Port block de-allocation         |
| 18    | Threshold Reached                |
+-------+----------------------------------+
```

Table 2: NAT Event ID

4.4. Quota Exceeded Event Types

   The Quota Exceeded event is a natEvent IE described in Table 2. The
   Quota Exceeded events are generated when the hard limits set by the
   administrator have been reached or exceeded. The following table
   shows the sub-event types for the Quota Exceeded event. The events
   that can be reported are the maximum session entries limit reached,
   maximum BIB entries limit reached, maximum (session/BIB) entries per
   user limit reached, maximum active hosts or subscribers limit
   reached, and maximum Fragments pending reassembly limit reached.

```
+-------+--------------------------------------+
| Value | Quota Exceeded Event Name            |
+-------+--------------------------------------+
| 0     | Reserved                             |
| 1     | Maximum session entries              |
| 2     | Maximum BIB entries                  |
| 3     | Maximum entries per user             |
| 4     | Maximum active hosts or subscribers  |
| 5     | Maximum fragments pending reassembly |
+-------+--------------------------------------+
```

                    Table 3: Quota Exceeded Event

4.5. Threshold Reached Event Types

   The following table shows the sub-event types for the Threshold
   Reached event. The administrator can configure the thresholds, and
   whenever the threshold is reached or exceeded, the corresponding
   events are generated. The main difference between the Quota Exceeded
   and Threshold Reached events is that, once the Quota Exceeded events
   are hit, the packets are dropped or mappings will not be created,
   whereas the Threshold Reached events will provide the operator a
   chance to take action before the traffic disruptions can happen. A
   NAT device can choose to implement one or the other, or both.

   The address pool high threshold event will be reported when the
   address pool reaches a high-water mark as defined by the operator.
   This will serve as an indication that either the operator might have
   to add more addresses to the pool or the subsequent users may be
   denied NAT translation mappings.

   The address pool low threshold event will be reported when the
   address pool reaches a low-water mark as defined by the operator.
   This will serve as an indication that the operator can reclaim some
   of the global IPv4 addresses in the pool.

   The address and port mapping high threshold event is generated when
   the number of ports in the configured address pool has reached a
   configured threshold.

   The per-user address and port mapping high threshold is generated
   when a single user utilizes more address and port mapping than a
   configured threshold. We don't track the low threshold for per-user
   address and port mappings because, as the ports are freed, the
   address will become available. The address pool low threshold event
   will then be triggered so that the global IPv4 address can be
   reclaimed.

The global address mapping high threshold event is generated when the
maximum number of mappings per user is reached for a NAT device doing
paired-address pooling.

```
+-------+----------------------------------------------------------+
| Value | Threshold Exceeded Event Name                            |
+-------+----------------------------------------------------------+
| 0     | Reserved                                                 |
| 1     | Address pool high threshold event                        |
| 2     | Address pool low threshold event                         |
| 3     | Address and port mapping high threshold event            |
| 4     | Address and port mapping per user high threshold event   |
| 5     | Global address mapping high threshold event              |
+-------+----------------------------------------------------------+
```

                      Table 4: Threshold Event

4.6. Templates for NAT Events

   The following is the template of events that will be logged. The
   events below are identified at the time of this writing, but the set
   of events is extensible. A NAT device that implements a given NAT
   event MUST support the mandatory IEs in the templates. Depending on
   the implementation and configuration, various IEs that are not
   mandatory can be included or ignored.

4.6.1. NAT44 Session Create and Delete Events

   These events will be generated when a NAT44 session is created or
   deleted. The template will be the same; the natEvent will indicate
   whether it is a create or a delete event. The following is a
   template of the event.

   The destination address and port information is optional as required
   by [RFC6888]. However, when the destination information is
   suppressed, the session log event contains the same information as
   the BIB event. In such cases, the NAT device SHOULD NOT send both
   BIB and session events.

```
+---------------------------------+-------------+-----------+
| Field Name                      | Size (bits) | Mandatory |
+---------------------------------+-------------+-----------+
| timeStamp                       |     64      |    Yes    |
| natEvent                        |      8      |    Yes    |
| sourceIPv4Address               |     32      |    Yes    |
| postNATSourceIPv4Address        |     32      |    Yes    |
| protocolIdentifier              |      8      |    Yes    |
| sourceTransportPort             |     16      |    Yes    |
| postNAPTSourceTransportPort     |     16      |    Yes    |
| destinationIPv4Address          |     32      |    No     |
| postNATDestinationIPv4Address   |     32      |    No     |
| destinationTransportPort        |     16      |    No     |
| postNAPTDestinationTransportPort|     16      |    No     |
| natInstanceID                   |     32      |    No     |
| vlanID/ingressVRFID             |    16/32    |    No     |
| internalAddressRealm            | octetArray  |    No     |
| externalAddressRealm            | octetArray  |    No     |
+---------------------------------+-------------+-----------+
```

             Table 5: NAT44 Session Delete/Create Template

4.6.2. NAT64 Session Create and Delete Events

   These events will be generated when a NAT64 session is created or
   deleted. The following is a template of the event.

```
+---------------------------------+-------------+-----------+
| Field Name                      | Size (bits) | Mandatory |
+---------------------------------+-------------+-----------+
| timeStamp                       |     64      |    Yes    |
| natEvent                        |      8      |    Yes    |
| sourceIPv6Address               |    128      |    Yes    |
| postNATSourceIPv4Address        |     32      |    Yes    |
| protocolIdentifier              |      8      |    Yes    |
| sourceTransportPort             |     16      |    Yes    |
| postNAPTSourceTransportPort     |     16      |    Yes    |
| destinationIPv6Address          |    128      |    No     |
| postNATDestinationIPv4Address   |     32      |    No     |
| destinationTransportPort        |     16      |    No     |
| postNAPTDestinationTransportPort|     16      |    No     |
| natInstanceID                   |     32      |    No     |
| vlanID/ingressVRFID             |    16/32    |    No     |
| internalAddressRealm            | octetArray  |    No     |
| externalAddressRealm            | octetArray  |    No     |
+---------------------------------+-------------+-----------+
```

           Table 6: NAT64 Session Create/Delete Event Template

4.6.3. NAT44 BIB Create and Delete Events

   These events will be generated when a NAT44 Bind entry is created or
   deleted. The following is a template of the event.

Field Name	Size (bits)	Mandatory
timeStamp	64	Yes
natEvent	8	Yes
sourceIPv4Address	32	Yes
postNATSourceIPv4Address	32	Yes
protocolIdentifier	8	No
sourceTransportPort	16	No
postNAPTSourceTransportPort	16	No
natInstanceID	32	No
vlanID/ingressVRFID	16/32	No
internalAddressRealm	octetArray	No
externalAddressRealm	octetArray	No

             Table 7: NAT44 BIB Create/Delete Event Template

4.6.4. NAT64 BIB Create and Delete Events

   These events will be generated when a NAT64 Bind entry is created or
   deleted. The following is a template of the event.

Field Name	Size (bits)	Mandatory
timeStamp	64	Yes
natEvent	8	Yes
sourceIPv6Address	128	Yes
postNATSourceIPv4Address	32	Yes
protocolIdentifier	8	No
sourceTransportPort	16	No
postNAPTSourceTransportPort	16	No
natInstanceID	32	No
vlanID/ingressVRFID	16/32	No
internalAddressRealm	octetArray	No
externalAddressRealm	octetArray	No

             Table 8: NAT64 BIB Create/Delete Event Template

4.6.5. Addresses Exhausted Event

   This event will be generated when a NAT device runs out of global
   IPv4 addresses in a given pool of addresses. Typically, this event
   would mean that the NAT device won't be able to create any new
   translations until some addresses/ports are freed. This event SHOULD
   be rate-limited, as many packets hitting the device at the same time
   will trigger a burst of addresses exhausted events.

   The following is a template of the event.

```
             +---------------+-------------+-----------+
             | Field Name    | Size (bits) | Mandatory |
             +---------------+-------------+-----------+
             | timeStamp     |     64      |    Yes    |
             | natEvent      |      8      |    Yes    |
             | natPoolID     |     32      |    Yes    |
             | natInstanceID |     32      |    No     |
             +---------------+-------------+-----------+
```

              Table 9: Addresses Exhausted Event Template

4.6.6. Ports Exhausted Event

   This event will be generated when a NAT device runs out of ports for
   a global IPv4 address. Port exhaustion shall be reported per
   protocol (UDP, TCP, etc.). This event SHOULD be rate-limited, as
   many packets hitting the device at the same time will trigger a burst
   of port exhausted events.

   The following is a template of the event.

```
        +-------------------------+-------------+-----------+
        | Field Name              | Size (bits) | Mandatory |
        +-------------------------+-------------+-----------+
        | timeStamp               |     64      |    Yes    |
        | natEvent                |      8      |    Yes    |
        | postNATSourceIPv4Address |    32      |    Yes    |
        | protocolIdentifier      |      8      |    Yes    |
        | natInstanceID           |     32      |    No     |
        +-------------------------+-------------+-----------+
```

            Table 10: Ports Exhausted Event Template

4.6.7. Quota Exceeded Events

   This event will be generated when a NAT device cannot allocate
   resources as a result of an administratively defined policy. The
   Quota Exceeded event templates are described below.

4.6.7.1. Maximum Session Entries Exceeded

   The maximum session entries exceeded event is generated when the
   administratively configured NAT session limit is reached. The
   following is the template of the event.

```
+----------------------+-------------+-----------+
| Field Name           | Size (bits) | Mandatory |
+----------------------+-------------+-----------+
| timeStamp            |     64      |    Yes    |
| natEvent             |      8      |    Yes    |
| natQuotaExceededEvent |    32      |    Yes    |
| maxSessionEntries    |     32      |    Yes    |
| natInstanceID        |     32      |    No     |
+----------------------+-------------+-----------+
```

         Table 11: Session Entries Exceeded Event Template

4.6.7.2. Maximum BIB Entries Exceeded

   The maximum BIB entries exceeded event is generated when the
   administratively configured BIB entry limit is reached. The
   following is the template of the event.

```
+----------------------+-------------+-----------+
| Field Name           | Size (bits) | Mandatory |
+----------------------+-------------+-----------+
| timeStamp            |     64      |    Yes    |
| natEvent             |      8      |    Yes    |
| natQuotaExceededEvent |    32      |    Yes    |
| maxBIBEntries        |     32      |    Yes    |
| natInstanceID        |     32      |    No     |
+----------------------+-------------+-----------+
```

           Table 12: BIB Entries Exceeded Event Template

4.6.7.3. Maximum Entries per User Exceeded

   This event is generated when a single user reaches the
   administratively configured NAT translation limit. The following is
   the template of the event.

```
       +----------------------+------------+---------------+
       | Field Name           | Size (bits) |  Mandatory    |
       +----------------------+------------+---------------+
       | timeStamp            |     64     |      Yes       |
       | natEvent             |      8     |      Yes       |
       | natQuotaExceededEvent |    32     |      Yes       |
       | maxEntriesPerUser    |     32     |      Yes       |
       | sourceIPv4Address    |     32     | Yes for NAT44  |
       | sourceIPv6Address    |    128     | Yes for NAT64  |
       | natInstanceID        |     32     |      No        |
       | vlanID/ingressVRFID  |   16/32    |      No        |
       +----------------------+------------+---------------+
```

           Table 13: Per-User Entries Exceeded Event Template

4.6.7.4. Maximum Active Hosts or Subscribers Exceeded

   This event is generated when the number of allowed hosts or
   subscribers reaches the administratively configured limit. The
   following is the template of the event.

```
       +----------------------+------------+-----------+
       | Field Name           | Size (bits) | Mandatory |
       +----------------------+------------+-----------+
       | timeStamp            |     64     |    Yes    |
       | natEvent             |      8     |    Yes    |
       | natQuotaExceededEvent |    32     |    Yes    |
       | maxSubscribers       |     32     |    Yes    |
       | natInstanceID        |     32     |    No     |
       +----------------------+------------+-----------+
```

        Table 14: Maximum Hosts/Subscribers Exceeded Event Template

4.6.7.5. Maximum Fragments Pending Reassembly Exceeded

   This event is generated when the number of fragments pending
   reassembly reaches the administratively configured limit. Note that
   in the case of NAT64, when this condition is detected in the IPv6-to-
   IPv4 direction, the IPv6 source address is mandatory in the template.
   Similarly, when this condition is detected in IPv4-to-IPv6 direction,
   the source IPv4 address is mandatory in the template below. The
   following is the template of the event.

```
+-----------------------------+------------+----------------+
| Field Name                  | Size (bits)|   Mandatory    |
+-----------------------------+------------+----------------+
| timeStamp                   |     64     |      Yes       |
| natEvent                    |      8     |      Yes       |
| natQuotaExceededEvent       |     32     |      Yes       |
| maxFragmentsPendingReassembly|    32     |      Yes       |
| sourceIPv4Address           |     32     |  Yes for NAT44 |
| sourceIPv6Address           |    128     |  Yes for NAT64 |
| natInstanceID               |     32     |      No        |
| vlanID/ingressVRFID         |   16/32    |      No        |
| internalAddressRealm        | octetArray |      No        |
+-----------------------------+------------+----------------+
```

     Table 15: Maximum Fragments Pending Reassembly Exceeded Event
                             Template

4.6.8. Threshold Reached Events

   This event will be generated when a NAT device reaches an operator-
   configured threshold when allocating resources. The Threshold
   Reached events are described in the section above. The following is
   a template of the individual events.

4.6.8.1. Address Pool High or Low Threshold Reached

   This event is generated when the high or low threshold is reached for
   the address pool. The template is the same for both high and low
   threshold events

```
+-------------------------------------------+--------+-----------+
| Field Name                                | Size   | Mandatory |
|                                           | (bits) |           |
+-------------------------------------------+--------+-----------+
| timeStamp                                 |   64   |    Yes    |
| natEvent                                  |    8   |    Yes    |
| natThresholdEvent                         |   32   |    Yes    |
| natPoolID                                 |   32   |    Yes    |
| addressPoolHighThreshold/                 |   32   |    Yes    |
| addressPoolLowThreshold                   |        |           |
| natInstanceID                             |   32   |    No     |
+-------------------------------------------+--------+-----------+
```

   Table 16: Address Pool High/Low Threshold Reached Event Template

4.6.8.2. Address and Port Mapping High Threshold Reached

   This event is generated when the high threshold is reached for the
   address pool and ports.

   +--+--------+-----------+
   | Field Name | Size | Mandatory |
   | | (bits) | |
   +--+--------+-----------+
timeStamp	64	Yes
natEvent	8	Yes
natThresholdEvent	32	Yes
addressPortMappingHighThreshold/	32	Yes
addressPortMappingLowThreshold		
natInstanceID	32	No
   +--+--------+-----------+

        Table 17: Address Port High Threshold Reached Event Template

4.6.8.3. Address and Port Mapping per User High Threshold Reached

   This event is generated when the high threshold is reached for the
   per-user address pool and ports.

   +--+--------+-----------+
   | Field Name | Size | Mandatory |
   | | (bits) | |
   +--+--------+-----------+
timeStamp	64	Yes
natEvent	8	Yes
natThresholdEvent	32	Yes
addressPortMappingHighThreshold/	32	Yes
addressPortMappingLowThreshold		
sourceIPv4Address	32	Yes for
		NAT44
sourceIPv6Address	128	Yes for
		NAT64
natInstanceID	32	No
vlanID/ingressVRFID	16/32	No
   +--+--------+-----------+

      Table 18: Address and Port Mapping per User High Threshold Reached
                              Event Template

4.6.8.4. Global Address Mapping High Threshold Reached

   This event is generated when the high threshold is reached for the
   per-user address pool and ports. This is generated only by NAT
   devices that use a paired-address-pooling behavior.

```
+-----------------------------------+-------------+-----------+
| Field Name                        | Size (bits) | Mandatory |
+-----------------------------------+-------------+-----------+
| timeStamp                         |     64      |    Yes    |
| natEvent                          |      8      |    Yes    |
| natThresholdEvent                 |     32      |    Yes    |
| globalAddressMappingHighThreshold |     32      |    Yes    |
| natInstanceID                     |     32      |    No     |
| vlanID/ingressVRFID               |    16/32    |    No     |
+-----------------------------------+-------------+-----------+
```

        Table 19: Global Address Mapping High Threshold Reached Event
                               Template

4.6.9. Address Binding Create and Delete Events

   These events will be generated when a NAT device binds a local
   address with a global address and when the global address is freed.
   A NAT device will generate the binding events when it receives the
   first packet of the first flow from a host in the private realm.

```
+-------------------------+-------------+---------------+
| Field Name              | Size (bits) |   Mandatory   |
+-------------------------+-------------+---------------+
| timeStamp               |     64      |      Yes      |
| natEvent                |      8      |      Yes      |
| sourceIPv4Address       |     32      | Yes for NAT44 |
| sourceIPv6Address       |    128      | Yes for NAT64 |
| postNATSourceIPv4Address |    32      |      Yes      |
| natInstanceID           |     32      |      No       |
+-------------------------+-------------+---------------+
```

                 Table 20: NAT Address Binding Template

4.6.10. Port Block Allocation and De-allocation

   This event will be generated when a NAT device allocates/de-allocates
   ports in a bulk fashion, as opposed to allocating a port on a per-
   flow basis.

portRangeStart represents the starting value of the range.

portRangeEnd represents the ending value of the range.

NAT devices would do this in order to reduce logs and to potentially limit the number of connections a subscriber is allowed to use. In the following Port Block allocation template, the portRangeStart and portRangeEnd MUST be specified.

It is up to the implementation to choose to consolidate log records in case two consecutive port ranges for the same user are allocated or freed.

```
+-------------------------+------------+---------------+
| Field Name              | Size (bits)|   Mandatory   |
+-------------------------+------------+---------------+
| timeStamp               |     64     |      Yes       |
| natEvent                |      8     |      Yes       |
| sourceIPv4Address       |     32     | Yes for NAT44 |
| sourceIPv6Address       |    128     | Yes for NAT64 |
| postNATSourceIPv4Address |    32     |      Yes       |
| portRangeStart          |     16     |      Yes       |
| portRangeEnd            |     16     |      No        |
| natInstanceID           |     32     |      No        |
+-------------------------+------------+---------------+
```

                Table 21: NAT Port Block Allocation Event Template

5. Management Considerations

   This section considers requirements for management of the log system
   to support logging of the events described above. It first covers
   requirements applicable to log management in general. Any additional
   standardization required to fulfill these requirements is out of
   scope of the present document. Some management considerations are
   covered in [NAT-LOG]. This document covers the additional
   considerations.

5.1. Ability to Collect Events from Multiple NAT Devices

   An IPFIX Collector MUST be able to collect events from multiple NAT
   devices and decipher events based on the Observation Domain ID in the
   IPFIX header.

5.2. Ability to Suppress Events

   The exhaustion events can be overwhelming during traffic bursts;
   hence, they SHOULD be handled by the NAT devices to rate-limit them
   before sending them to the Collectors. For example, when the port
   exhaustion happens during bursty conditions, instead of sending a
   port exhaustion event for every packet, the exhaustion events SHOULD
   be rate-limited by the NAT device.

6. IANA Considerations

6.1. Information Elements

   IANA has registered the following IEs in the "IPFIX Information
   Elements" registry at [IPFIX-IANA].

6.1.1. natInstanceID

   ElementID: 463

   Name: natInstanceID

   Description: This Information Element uniquely identifies an Instance
   of the NAT that runs on a NAT middlebox function after the packet
   passes the Observation Point. natInstanceID is defined in [RFC7659].

   Abstract Data Type: unsigned32

   Data Type Semantics: identifier

   Reference: See [RFC791] for the definition of the IPv4 source address
   field. See [RFC3022] for the definition of NAT. See [RFC3234] for
   the definition of middleboxes.

6.1.2. internalAddressRealm

   ElementID: 464

   Name: internalAddressRealm

   Description: This Information Element represents the internal address
   realm where the packet is originated from or destined to. By
   definition, a NAT mapping can be created from two address realms, one
   from internal and one from external. Realms are implementation
   dependent and can represent a Virtual Routing and Forwarding (VRF)
   ID, a VLAN ID, or some unique identifier. Realms are optional and,
   when left unspecified, would mean that the external and internal
   realms are the same.

Abstract Data Type: octetArray

Data Type Semantics: identifier

Reference: See [RFC791] for the definition of the IPv4 source address
field. See [RFC3022] for the definition of NAT. See [RFC3234] for
the definition of middleboxes.

6.1.3. externalAddressRealm

ElementID: 465

Name: externalAddressRealm

Description: This Information Element represents the external address
realm where the packet is originated from or destined to. The
detailed definition is in the internal address realm as specified
above.

Abstract Data Type: octetArray

Data Type Semantics: identifier

Reference: See [RFC791] for the definition of the IPv4 source address
field. See [RFC3022] for the definition of NAT. See [RFC3234] for
the definition of middleboxes.

6.1.4. natQuotaExceededEvent

ElementID: 466

Name: natQuotaExceededEvent

Description: This Information Element identifies the type of a NAT
Quota Exceeded event. Values for this Information Element are listed
in the "NAT Quota Exceeded Event Type" registry, see [IPFIX-IANA].
Initial values in the registry are defined by the table below. New
assignments of values will be administered by IANA and are subject to
Expert Review [RFC8126]. Experts need to check definitions of new
values for completeness, accuracy, and redundancy.

```
      +--------+-------------------------------------+
      | Value  | Quota Exceeded Event Name           |
      +--------+-------------------------------------+
      | 0      | Reserved                            |
      | 1      | Maximum session entries             |
      | 2      | Maximum BIB entries                 |
      | 3      | Maximum entries per user            |
      | 4      | Maximum active hosts or subscribers |
      | 5      | Maximum fragments pending reassembly|
      +--------+-------------------------------------+
```

              Note: This is the same as Table 3.

   Abstract Data Type: unsigned32

   Data Type Semantics: identifier

   Reference: See [RFC791] for the definition of the IPv4 source address
   field. See [RFC3022] for the definition of NAT. See [RFC3234] for
   the definition of middleboxes.

6.1.5. natThresholdEvent

   ElementID: 467

   Name: natThresholdEvent

   Description: This Information Element identifies a type of a NAT
   Threshold event. Values for this Information Element are listed in
   the "NAT Threshold Event Type" registry, see [IPFIX-IANA]. Initial
   values in the registry are defined by the table below. New
   assignments of values will be administered by IANA and are subject to
   Expert Review [RFC8126]. Experts need to check definitions of new
   values for completeness, accuracy, and redundancy.

```
   +--------+----------------------------------------------------------+
   | Value  | Threshold Exceeded Event Name                            |
   +--------+----------------------------------------------------------+
   | 0      | Reserved                                                 |
   | 1      | Address pool high threshold event                        |
   | 2      | Address pool low threshold event                         |
   | 3      | Address and port mapping high threshold event            |
   | 4      | Address and port mapping per user high threshold event   |
   | 5      | Global address mapping high threshold event              |
   +--------+----------------------------------------------------------+
```

              Note: This is the same as Table 4.

Abstract Data Type: unsigned32

Data Type Semantics: identifier

Reference: See [RFC791] for the definition of the IPv4 source address
field. See [RFC3022] for the definition of NAT. See [RFC3234] for
the definition of middleboxes.

6.1.6. natEvent

The original definition of this Information Element specified only
three values: 1, 2, and 3. This definition has been replaced by a
registry, to which new values can be added. The semantics of the
three originally defined values remain unchanged. IANA maintains the
"NAT Event Type (Value 230)" registry for values of this Information
Element at [IPFIX-IANA].

ElementID: 230

Name: natEvent

Description: This Information Element identifies a NAT event. This
IE identifies the type of a NAT event. Examples of NAT events
include, but are not limited to, NAT translation create, NAT
translation delete, Threshold Reached, or Threshold Exceeded, etc.
Values for this Information Element are listed in the "NAT Event
Type" registry, see [IPFIX-IANA]. The NAT event values in the
registry are defined by Table 2 in Section 4.3. New assignments of
values will be administered by IANA and are subject to Expert Review
[RFC8126]. Experts need to check definitions of new values for
completeness, accuracy, and redundancy.

Abstract Data Type: unsigned8

Data Type Semantics: identifier

Reference: See [RFC3022] for the definition of NAT. See [RFC3234]
for the definition of middleboxes. See RFC 8158 for the definitions
of values 4-16.

6.1.7. maxSessionEntries

ElementID: 471

Name: maxSessionEntries

Description: This element represents the maximum session entries that
can be created by the NAT device.

   Abstract Data Type: unsigned32

   Data Type Semantics: identifier

   Reference: See [RFC3022] for the definition of NAT. See [RFC3234]
   for the definition of middleboxes.

6.1.8. maxBIBEntries

   ElementID: 472

   Name: maxBIBEntries

   Description: This element represents the maximum BIB entries that can
   be created by the NAT device.

   Abstract Data Type: unsigned32

   Data Type Semantics: identifier

   Reference: See [RFC3022] for the definition of NAT. See [RFC3234]
   for the definition of middleboxes.

6.1.9. maxEntriesPerUser

   ElementID: 473

   Name: maxEntriesPerUser

   Description: This element represents the maximum NAT entries that can
   be created per user by the NAT device.

   Abstract Data Type: unsigned32

   Data Type Semantics: identifier

   Reference: See [RFC3022] for the definition of NAT. See [RFC3234]
   for the definition of middleboxes.

6.1.10. maxSubscribers

   ElementID: 474

   Name: maxSubscribers

   Description: This element represents the maximum subscribers or
   maximum hosts that are allowed by the NAT device.

   Abstract Data Type: unsigned32

   Data Type Semantics: identifier

   Reference: See [RFC3022] for the definition of NAT. See [RFC3234]
   for the definition of middleboxes.

6.1.11. maxFragmentsPendingReassembly

   ElementID: 475

   Name: maxFragmentsPendingReassembly

   Description: This element represents the maximum fragments that the
   NAT device can store for reassembling the packet.

   Abstract Data Type: unsigned32

   Data Type Semantics: identifier

   Reference: See [RFC3022] for the definition of NAT. See [RFC3234]
   for the definition of middleboxes.

6.1.12. addressPoolHighThreshold

   ElementID: 476

   Name: addressPoolHighThreshold

   Description: This element represents the high threshold value of the
   number of public IP addresses in the address pool.

   Abstract Data Type: unsigned32

   Data Type Semantics: identifier

   Reference: See [RFC3022] for the definition of NAT. See [RFC3234]
   for the definition of middleboxes.

6.1.13. addressPoolLowThreshold

   ElementID: 477

   Name: addressPoolLowThreshold

   Description: This element represents the low threshold value of the
   number of public IP addresses in the address pool.

   Abstract Data Type: unsigned32

   Data Type Semantics: identifier

   Reference: See [RFC3022] for the definition of NAT. See [RFC3234]
   for the definition of middleboxes.

6.1.14. addressPortMappingHighThreshold

   ElementID: 478

   Name: addressPortMappingHighThreshold

   Description: This element represents the high threshold value of the
   number of address and port mappings.

   Abstract Data Type: unsigned32

   Data Type Semantics: identifier

   Reference: See [RFC3022] for the definition of NAT. See [RFC3234]
   for the definition of middleboxes.

6.1.15. addressPortMappingLowThreshold

   ElementID: 479

   Name: addressPortMappingLowThreshold

   Description: This element represents the low threshold value of the
   number of address and port mappings.

   Abstract Data Type: unsigned32

   Data Type Semantics: identifier

   Reference: See [RFC3022] for the definition of NAT. See [RFC3234]
   for the definition of middleboxes.

6.1.16. addressPortMappingPerUserHighThreshold

   ElementID: 480

   Name: addressPortMappingPerUserHighThreshold

   Description: This element represents the high threshold value of the
   number of address and port mappings that a single user is allowed to
   create on a NAT device.

   Abstract Data Type: unsigned32

   Data Type Semantics: identifier

   Reference: See [RFC3022] for the definition of NAT. See [RFC3234]
   for the definition of middleboxes.

6.1.17. globalAddressMappingHighThreshold

   ElementID: 481

   Name: globalAddressMappingHighThreshold

   Description: This element represents the high threshold value of the
   number of address and port mappings that a single user is allowed to
   create on a NAT device in a paired address pooling behavior.

   Abstract Data Type: unsigned32

   Data Type Semantics: identifier

   Reference: See [RFC3022] for the definition of NAT. See [RFC3234]
   for the definition of middleboxes. See [RFC4787] for the definition
   of paired address pooling behavior.

7. Security Considerations

   The security considerations listed in detail for IPFIX in [RFC7011]
   apply to this document as well. As described in [RFC7011], the
   messages exchanged between the NAT device and the Collector MUST be
   protected to provide confidentiality, integrity, and authenticity.
   Without those characteristics, the messages are subject to various
   kinds of attacks. These attacks are described in great detail in
   [RFC7011].

   This document re-emphasizes the use of Transport Layer Security (TLS)
   or Datagram Transport Layer Security (DTLS) for exchanging the log
   messages between the NAT device and the Collector. The log events
   sent in cleartext can result in confidential data being exposed to
   attackers, who could then spoof log events based on the information
   in cleartext messages. Hence, the log events SHOULD NOT be sent in
   cleartext.

   The logging of NAT events can result in privacy concerns as a result
   of exporting information such as the source address and port
   information. The logging of destination information can also cause
   privacy concerns, but it has been well documented in [RFC6888]. A
   NAT device can choose to operate in various logging modes if it wants

   to avoid logging of private information. The Collector that receives
   the information can also choose to mask the private information but
   generate reports based on abstract data. It is outside the scope of
   this document to address the implementation of logging modes for
   privacy considerations.

8. References

8.1. Normative References

   [RFC2119]   Bradner, S., "Key words for use in RFCs to Indicate
               Requirement Levels", BCP 14, RFC 2119,
               DOI 10.17487/RFC2119, March 1997,
               <https://www.rfc-editor.org/info/rfc2119>.

   [RFC4787]   Audet, F., Ed. and C. Jennings, "Network Address
               Translation (NAT) Behavioral Requirements for Unicast
               UDP", BCP 127, RFC 4787, DOI 10.17487/RFC4787, January
               2007, <https://www.rfc-editor.org/info/rfc4787>.

   [RFC5382]   Guha, S., Ed., Biswas, K., Ford, B., Sivakumar, S., and P.
               Srisuresh, "NAT Behavioral Requirements for TCP", BCP 142,
               RFC 5382, DOI 10.17487/RFC5382, October 2008,
               <https://www.rfc-editor.org/info/rfc5382>.

   [RFC6146]   Bagnulo, M., Matthews, P., and I. van Beijnum, "Stateful
               NAT64: Network Address and Protocol Translation from IPv6
               Clients to IPv4 Servers", RFC 6146, DOI 10.17487/RFC6146,
               April 2011, <https://www.rfc-editor.org/info/rfc6146>.

   [RFC6302]   Durand, A., Gashinsky, I., Lee, D., and S. Sheppard,
               "Logging Recommendations for Internet-Facing Servers",
               BCP 162, RFC 6302, DOI 10.17487/RFC6302, June 2011,
               <https://www.rfc-editor.org/info/rfc6302>.

   [RFC6888]   Perreault, S., Ed., Yamagata, I., Miyakawa, S., Nakagawa,
               A., and H. Ashida, "Common Requirements for Carrier-Grade
               NATs (CGNs)", BCP 127, RFC 6888, DOI 10.17487/RFC6888,
               April 2013, <https://www.rfc-editor.org/info/rfc6888>.

   [RFC7011]   Claise, B., Ed., Trammell, B., Ed., and P. Aitken,
               "Specification of the IP Flow Information Export (IPFIX)
               Protocol for the Exchange of Flow Information", STD 77,
               RFC 7011, DOI 10.17487/RFC7011, September 2013,
               <https://www.rfc-editor.org/info/rfc7011>.

   [RFC7659] Perreault, S., Tsou, T., Sivakumar, S., and T. Taylor,
              "Definitions of Managed Objects for Network Address
              Translators (NATs)", RFC 7659, DOI 10.17487/RFC7659,
              October 2015, <https://www.rfc-editor.org/info/rfc7659>.

   [RFC8174] Leiba, B., "Ambiguity of Uppercase vs Lowercase in RFC
              2119 Key Words", BCP 14, RFC 8174, DOI 10.17487/RFC8174,
              May 2017, <https://www.rfc-editor.org/info/rfc8174>.

8.2. Informative References

   [IPFIX-IANA]
              IANA, "IPFIX Information Elements",
              <http://www.iana.org/assignments/ipfix>.

   [NAT-LOG] Chen, Z., Zhou, C., Tsou, T., and T. Taylor, Ed., "Syslog
              Format for NAT Logging", Work in Progress, draft-ietf-
              behave-syslog-nat-logging-06, January 2014.

   [RFC791]   Postel, J., "Internet Protocol", STD 5, RFC 791,
              DOI 10.17487/RFC0791, September 1981,
              <https://www.rfc-editor.org/info/rfc791>.

   [RFC2663] Srisuresh, P. and M. Holdrege, "IP Network Address
              Translator (NAT) Terminology and Considerations",
              RFC 2663, DOI 10.17487/RFC2663, August 1999,
              <https://www.rfc-editor.org/info/rfc2663>.

   [RFC3022] Srisuresh, P. and K. Egevang, "Traditional IP Network
              Address Translator (Traditional NAT)", RFC 3022,
              DOI 10.17487/RFC3022, January 2001,
              <https://www.rfc-editor.org/info/rfc3022>.

   [RFC3234] Carpenter, B. and S. Brim, "Middleboxes: Taxonomy and
              Issues", RFC 3234, DOI 10.17487/RFC3234, February 2002,
              <https://www.rfc-editor.org/info/rfc3234>.

   [RFC5424] Gerhards, R., "The Syslog Protocol", RFC 5424,
              DOI 10.17487/RFC5424, March 2009,
              <https://www.rfc-editor.org/info/rfc5424>.

   [RFC8126] Cotton, M., Leiba, B., and T. Narten, "Guidelines for
              Writing an IANA Considerations Section in RFCs", BCP 26,
              RFC 8126, DOI 10.17487/RFC8126, June 2017,
              <https://www.rfc-editor.org/info/rfc8126>.

Authors' Addresses

   Senthil Sivakumar
   Cisco Systems
   7100-8 Kit Creek Road
   Research Triangle Park, NC 27709
   United States of America

   Phone: +1 919 392 5158
   Email: ssenthil@cisco.com

   Reinaldo Penno
   Cisco Systems
   170 W Tasman Drive
   San Jose, CA 95035
   United States of America

   Email: repenno@cisco.com