NSIS
Internet-Draft
Expires: January 12, 2006

A. Pashalidis
H. Tschofenig
Siemens

July 11, 2005

NAT Traversal for GIMPS
draft-pashalidis-nsis-gimps-nattraversal-00.txt

Status of this Memo

Copyright Notice

Abstract

This document contains a number of mechanisms that may be used in order to enable General Internet Messaging Protocol for Signaling (GIMPS) messages to traverse different types of Network Address Translators that may be located along the path between two adjecent NSLP hosts.

Table of Contents

1. Introduction

Network Address Translators (NATs) modify certain fields in the IP header of the IP packets that traverse them. In the context of signalling as defined by the NSIS group, this behaviour, if not properly addressed, may lead to the installation of inconsistent and meaningless state at network nodes with respect to the actual traffic that traverses these nodes.

This document proposes a collection of algorithms that have to be implemented in order to enable GIMPS signalling, and the data flows to which this signalling refers, to traverse NATs in a way that preserves the consistency of state that is installed in the network, and in a manner transparent to signalling applications. The document is organised as follows. The next section introduces the terminology that is used throughout this document. Section 3 provides a detailed discussion of the problems that are addressed by this document. Section 4 list the assumptions on which the proposed mechanisms are based. Section 5 presents the proposed mechanisms for the case where no TLS or IPsec protection is required for the signalling traffic between two NSLP peers, and Section 6 presents the proposed mechanisms where such protection is required.

2. Terminology

The terminology, abbreviations and notational conventions that are used throughout the document are as follows.

- DR: Data Responder, as defined in [1]
- DS: Data Sender, as defined in [1]
- GaNAT: GIMPS-aware NAT - A GaNAT may implement a number of NSLPs, but does not implement the NATFW NSLP.
- GIMPS: General Internet Messaging Protocol for Signalling [1]
- NAT: Network Address Translator
- NI: NSIS Initiator, as defined in [1]
- NR: NSIS Responder, as defined in [1]
- NSIS: Next Steps in Signalling, as defined in [1]
- NSIS-aware: Implements GIMPS and zero or more NSLPs.
- NSIS-unaware: GIMPS-unaware, does not implement any NSLP.
- NSLP: NSIS Signalling Layer Protocol, as defined in [1]

- downstream: as defined in [1]
- upstream: as defined in [1]
- MRI: Message Routing Information, as defined in [1]
- NLI.IA: Interface Address field of the Network Layer Information header field, as defined in [1]
- <- : Assignment operator. The quantity to the right of the operator is assigned to the variable to its left.
- A.B: Element B of structure A. Example: [IP header].SourceIPAddress denotes the source IP address of an IP header.
- [data item]: This notation indicates that "data item" is a single identifier of a data structure. (Square brackets do not denote optional arguments in this document.)

3. Problem Statement

According to [1], all GIMPS messages carry IP addresses in order to define the data flow to which the signalling refers. Moreover, certain GIMPS messages also carry the IP address of the sending peer, in order to enable the receiving peer to address subsequent traffic to the sender. Packets that cross an addressing boundary, say from addressing space S1 to S2, have the IP addresses in the IP header translated from space S1 to S2 by the NAT; if GIMPS payloads are not translated in a consistent manner, the MRI in a GIMPS packet that crosses the boundary, e.g. from address space S1 to S2, refers to a flow that does not exist in S2. In fact, the flow is invalid in S2 because at least one of the involved IP addresses belongs to S1. Moreover, the IP address of the sending peer may also be invalid in the addressing space of the receiving peer. The purpose of this document is to describe a way for GIMPS messages to be translated in a way consistent with the translation that NATs apply to the IP headers of both signalling and data traffic.

A NAT may either be NSIS-unaware or NSIS-aware. The case of NSIS-unaware NATs is discussed in Section 7. If the NAT is NSIS-aware, it is typically also able to support at least one NSLP. Note that there

exists an NSLP, namely the NATFW NSLP [2], that specifically addresses NAT traversal for data flows. Inevitably, the NATFW NSLP also provides the necessary mechanisms for the related signalling to traverse the NATs involved. Therefore, we can further divide NSIS-aware NATs into two categories, namely GIMPS-and-NATFW-aware NATs and GIMPS-aware-and-NATFW-unaware NATs. In the sequel, we call the latter simply GIMPS-aware NATs (GaNATs). A GIMPS-and-NATFW-aware NAT performs NAT traversal according to the NATFW NSLP; the case of NAT traversal in the presence of such NATs is therefore beyond the scope of this document.

As is natural, a NATFW-aware NAT only translates the relevant fields for the NATFW signalling traffic in a way consistent with the relevant data flow. Consequently, GIMPS signalling in the presence of NATFW-unaware NATs and for NSLPs other than the NATFW NSLP remains an open problem. This document precisely addresses this problem, by proposing mechanisms that operate at the GIMPS layer.

In general, a given data flow between a data sender (DS) and a data receiver (DR) may have to traverse a number of NATs, some of which may be GIMPS-and-NATFW-aware, some may be GIMPS-aware, and some may be NSIS-unaware. Additionally, NSLP signalling for such a data flow may be required to traverse through a subset of those NATs. Whether or not the routing inftrastructure and state of the network causes the signalling for such a data flow to traverse the same NATs as the flow depends, among other things, on the signalling application.

While signalling of a QoS NSLP, for example, might not traverse any of the NATs that are traversed by the data flow, the signalling of the NATFW NSLP traverses at least those NATs that implement the NATFW NSLP (otherwise the signalling path would no longer be coupled to the data path, as this coupling is defined by the GIMPS QUERY/RESPONSE discovery mechanism). It is desirable for every possible combination of NATs, either on the data or the signalling path, to be functional and secure.

Due to the GIMPS QUERY/RESPONSE discovery mechanism (according to which QUERY messages are simply forwarded if the current node does not support the required NSLP), two GIMPS nodes identify themselves as NSLP peers only if they both implement the same NSLP, say NSLP X. This means that, if one or more X-unaware NATs are between them, then the two X peers are not able to discover each other at all. This is

because, even in the unlikely event that the bindings necessary for the GIMPS traffic to traverse the in-between NAT(s) exist, the NLI.IA Object included in the RESPONSE message sent by the downstream X-aware peer will be invalid (i.e. the IP address will be unreachable) in the address space of the upstream X peer. In order to overcome this limitation, either the two X peers need to cope with the in-between NAT(s), or, if the NAT(s) are GaNATs, they (the GaNATs) need to apply additional processing in order to transparently create and maintain the required consistency. Additionally, if X-aware NATs are on the data path (where X is any NSLP except NATFW), then these NATs should process X traffic in a way the preserves consistency after address translation. This processing deviates from the processing of X-aware non-NAT nodes. In the following sections we propose certain processing rules that aim to overcome the limitation of two adjacent X peers not being able to execute X in the presence of in-between NAT(s). We do not consider the case where X=NATFW and all NAT(s) on the path are NATFW-aware. This case is handled by the NATFW NSLP.

Note that we have to deal with a number of different situations, depending on whether X is supported by the GaNATs. Thus, we have the following three subcases.

- all GaNAT(s) are X-unaware
- all GaNAT(s) are X-aware (and X is not the NATFW NSLP)
- a combination of X-aware and X-unaware GaNATs are between to X peers.

In the following sections, we discuss the three cases separately.

4. Assumptions

The discussion in this document is based on the following assumptions. Note that X denotes a fixed NSLP, other than the NATFW NSLP.

1. No IP addresses and port numbers are carried in the payloads of X.

2. The path taken by the signalling traffic between those X peers that have GaNATs in between is such that the responses to packets that a GaNAT sends on given interface arrive on the same interface (if such responses are sent at all).

3. The path taken by signalling traffic remains fixed between the two X peers, as far as the in-between GaNATs are concerned. That is, we assume that signalling traffic traverses the same GaNAT(s) until at least one of the following conditions is met.

   * The NSIS state that is installed at the two X peers expires.

   * The NSIS state that is installed at the two X peers is refreshed using a GIMPS QUERY.

   * A new GIMPS QUERY/RESPONSE exchange takes place due to other reasons, e.g. a detected route change.

   Note that this assumption is not necessarily met by "normal" data path coupled signalling. This is because, under "normal" data path coupled signalling, the signalling traffic is "coupled" to the data traffic at nodes that implement X. Thus, under "normal" path coupled signalling, it is not an error condition (e.g. a "route change"), for example, if the set of on-path (non-X) GIMPS nodes changes, as long as adjacent X peers remain the same.

4. The data flow traverses the same set of GaNATs as the signalling traffic. By assumption 3, this set of GaNATs is fixed until the next GIMPS QUERY/RESPONSE procedure is executed.

+-----+

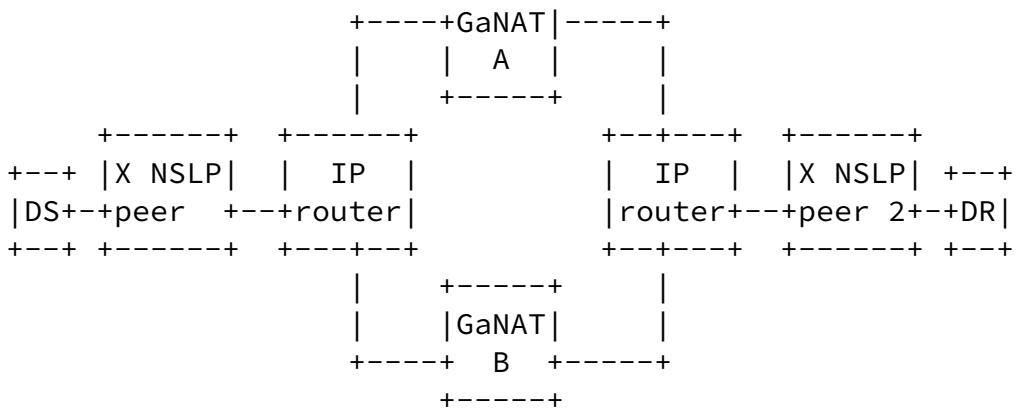

Figure 1: Network with more than one NAT at an addressing boundary

Figure 1 illustrates the importance of assumptions (3) and (4). With regard to that figure, suppose that a (D-mode) signalling session has been setup between the two adjacent X NSLP peers 1 and 2 and that both signalling and data traffic follows the path X NSLP peer 1 -> IP router -> GaNAT A -> IP router -> X NSLP peer 2. Suppose now that, after some time, X peer 1 decides to set up a C-mode connection with peer 2. Suppose moreover that the left IP router decides to forward the C-mode signalling traffic on the link towards GaNAT B. Thus, signalling traffic now follows the alternative path X NSLP peer 1 -> IP router -> GaNAT B -> IP router -> X NSLP peer 2. Note that this change in forwarding between the two adjacent X NSLP peers does not trigger a "route change" at the GIMPS layer because (a) it does not destroy the adjacency of peer 1 and 2 and (b) it does not destroy the coupling of the path taken by signalling traffic to that taken by data traffic (at X-aware nodes). Nevertheless, assumptions (3) and (4) mandate that this situation does not occur. However, even if such a situation occurs, the proposals in this document still work.

If assumption (1) does not hold, X has to provide additional mechanisms for the traversal of (Ga)NATs. These mechanisms must be compatible with the mechanisms described in this document. Assumptions (2), (3) and (4) hold if, at an addressing boundary, only one NAT exists. Due to security and management reasons, this is likely to be the case in many settings.

5. Traversal of GaNATs in the absence TLS or IPsec

This section describes the operation of GIMPS-aware NATs when no cryptographic protection of signalling data is requested by two NSLP peers. The situation when such protection is required is discussed in Section 6.

Recall that by GaNAT we mean a NAT that implements GIMPS but does not implement the NATFW NSLP. In this section we discuss the possibility of two NSIS peers that implement a given NSLP, denoted as X, to discover each other and subsequently exchange signalling messages in the presence of one or more GaNATs in between. Note that X may be any NSLP including the NATFW NSLP (however, if X=NATFW we do not consider X-aware GaNATs).

Note that we have to deal with three subcases, namely (a) the case where all GaNAT(s) are X-unaware, (b) the case where all GaNAT(s) are X-aware (and X is not the NATFW NSLP), and (c) the case where a combination of X-aware and X-unaware GaNATs are between to X peers. We discuss the three cases separately.

5.1 NSLP-unaware GaNATs

This section describes the algorithm that an X-unaware GaNAT must execute in order to enable the signalling traffic of two X peers to traverse the GaNAT in a transparent (for the two peers) manner. The notation A.B denotes the field B of data structure A.

Note that we have to deal with two types of GaNATs, namely those that are located at the NSIS initiator (NI-side), and those that are located at the NSIS responder (NR-side). This distinction arises due to the fact that NI-side and NR-side GaNATs obtain the destination IP address for forwarded packets in different ways.

5.1.1 NI-side NSLP-unaware GaNATs

For every arriving IP packet P, an NSLP-unaware, NI-side GaNAT executes the following algorithm.

1. If P has a RAO followed by the GIMPS header with an NSLP ID that is not supported, it is identified as a GIMPS QUERY. In this case the GaNAT performs the following.

    1. We denote P as GQ. It looks at the stack proposal ST in GQ. If it indicates that cryptographic protection is required, the algorithm that is executed is the one described in

Section 6.

2. The GaNAT remembers GQ along with the interface on which it arrived. We call this interface the "upstream link".

3. It searches its table of existing NAT bindings against entries that match the GQ.MRI. A matching entry means that the data flow, to which the signalling refers, already exists.

    1. If a matching entry is found, the GaNAT looks at which link the packets of the data flow are forwarded; we call this link the "downstream" link. Further, the GaNAT checks how the headers of the data flow (IP addresses, port numbers, etc) are translated according to this NAT binding. We denote [IP header].SourceIPAddress used on the downstream link as IPGaNATds, and the source port number used to forward the data traffic as SPNDTGaNATds. The NAT may also use a different source port number when forwarding signalling traffic. This port number is denoted as SPNSTGaNATds.

    2. If no matching entry is found, the GaNAT determines, based on its routing table, the link on which packets that match GQ.MRI (excluding GQ.MRI.SourceIPAddress) would be forwarded. We call this link the "downstream link". Then, the GaNAT acquires an IP address for itself on the downstream link. (This address could be dynamic or static.) This address will be used to forward both signalling and data traffic on the downstream link. If it also performs port translation, the GaNAT also acquires a source port number for the data traffic on the downstream link. This will be used with the NAT binding, if such a binding will be established for the data traffic at a later stage, and is denoted as SPNDTGaNATds . The signalling traffic packets may also be forwarded using the a different source port number as the incoming packets. We denote the acquired IP address as IPGaNATds and the source port number for the signalling traffic as SPNSTGaNATds.

Issues: The reason why the GaNAT may also assign a different source port number to the signalling traffic, is to enable the GaNAT to demultiplex (i.e. forward to the correct internal address) the signalling responses that arrive from downstream. Of course, a GaNAT does not need to actually change the source port of signalling traffic; it can always use SPNSTGaNATds the same port as in the incoming packet. Such a GaNAT may use the GIMPS session id in order to demultiplex the traffic that

arrives from the downstream direction. It is unclear which of the two approaches is preferable.

4. It creates a new GIMPS QUERY packet GQ', as follows.

    1. GQ' <- GQ

    2. GQ'.MRI.SourceIPAddress <- IPGaNATds

    3. GQ'.MRI.SourcePortNumber <- SPNDTGaNATds

    4. GQ'.[IP header].SourceIPAddress <- IPGaNATds

    5. GQ'.[TRANSPORT_LAYER_HEADER].SourcePort <- SPNSTGaNATds

    6. GQ'.NLI.IA <- IPGaNATds

    7. GQ'.S <- true

5. It remembers GQ and GQ', the fact that they are associated, and the associated upstream and downstream links. (Note: The GaNAT does not have to remember the entire packets; for simplicity of exposition, however, we assume it does. An implementation SHOULD discard at this point all information that is not used later.)

6. It forwards GQ' on the downstream link.

2. Otherwise, if P carries a [IP header].DestinationIPAddress that belongs to the GaNAT, and if it is identified as a GIMPS response in D-mode with an NSLP ID that is not supported, the GaNAT does the following (P is denoted as GR).

1. It searches for a matching GQ' in its buffer. A GR is said to match a GQ' if they carry the same cookie value. If none is found, GR is discarded. Otherwise, the GaNAT may also perform further consistency checks on a matching GR/GQ' pair, such as checking that they contain the same session IDs, MRIs, NSLP IDs. If consistency checks succeed, the GaNAT constructs a new GIMPS response GR', as follows.

    1. GR' <- GR

    2. GR'.MRI <- GQ.MRI, where GQ is the packet associated with GQ'(as remembered previously), and GQ' is the packet that matches the received GR.

    3. GR'.[IP header].SourceIPAddress <- IPGaNATus, where IPGaNATus = GQ.[IP header].DestinationIPAddress.

    4. GR'.[IP header].DestinationIPAddress <- GQ.NLI.IA

    5. GP'.S <- true.

    6. It inspects the stack proposals in GR' and the corresponding GQ' to see if the upstream X peer has a choice of more than one possible stack. If such choice exists, the GaNAT removes as many stack proposals from GR' as necessary, until only one stack can be chosen by the upstream peer for the messaging association. We denote this stack as ST. The GaNAT remembers this ST and its association with GQ, GQ', GR, GR'. We say that, in this case, the GaNAT "installs" the ST.

2. It forwards GR' on the upstream link.

3. If no NAT binding for the data traffic was found in step 1.3.2, the GaNAT now installs a NAT binding (for the unidirectional data traffic) which says that "a packet K that arrives on the upstream link and for which it holds that

    + K.[IP

header].DestinationIPAddress=GQ.MRI.DestinationIPAddress,

+ K.[IP header].Protocol=GQ.MRI.Protocol, and

+ K.[TCP/UDP header].SourcePort=GQ.MRI.SourcePort

should be forwarded on the downstream link, with [IP header].SourceIPAddress = IPGaNATds.

Issues: there is a question of whether this NAT binding should also enable data traffic in the opposite direction to traverse the NAT; in order to be able to demultiplex upstream traffic that carries data that belongs to different flows, the GaNAT should keep the necessary per-flow state. From a signalling point of view, however, upstream data traffic that corresponds (on the application level) to the downstream flow to which this GIMPS session refers, is a separate flow for which, dependent on the application, there may or there may not exist a signalling session. If such a signalling session exists, then the GaNAT acts as an NR-side GaNAT for this session. Thus, during the processing of this signalling care has to be taken not to establish a NAT binding for a flow for which a NAT binding already exists. Finally, security issues

arise when traffic, for which no signalling exists, is allowed to traverse a GaNAT.

Another issue is about refreshing the NAT binding. A NAT binding that was established as a result of GIMPS signalling should remain in place as long as the associated GIMPS state in the GaNAT remains valid. If GIMPS signalling refers to a NAT binding that already exists, then the timeout of the NAT binding should occur according to the NAT policy, in a manner independing from GIMPS processing. (If signalling persists after the deletion of a NAT binding, then the NAT binding may be re-installed and then timeout together with GIMPS state).

3. Otherwise, if P.[IP header].DestinationIPAddress belongs to the GaNAT, and if P is a GIMPS packet (either in D-mode or C-mode), the GaNAT does the following. If P does not match an existing installed ST, P is silently discarded. (A packet P is said to "match" an installed ST, if it carries the transport protocol and

port numbers indicated by ST.) Otherwise, if P has not arrived on either the downstream or upstream link of some ST, it is silently discarded. Otherwise, P has arrived either on the upstream or the downstream of some ST. The GaNAT constructs an outgoing packet P' as follows (the variables used below refer to those stored together with the ST in question).

1. P' <- P

2. If P has arrived on the upstream link, then

    1. P'.[IP header].SourceIPAddress <- IPGaNATds

    2. P'.MRI <- GQ'.MRI

    3. P'.NLI.IA <- IPGaNATus

    4. The GaNAT forwards P' on the downstream link.

3. else (if P has arrived on the downstream link)

    1. P'.[IP header].SourceIPAddress <- IPGaNATus

    2. P'.MRI <- GQ.MRI

    3. P'.NLI.IA <- IPGaNATus

    4. The GaNAT forwards P' on the upstream link.

Note: the above step will fail if ST indicates security. That is, if traffic is encrypted, then the GaNAT cannot construct P', and if traffic is integrity-protected, performing this step will cause an error at the receiving X peer. However, recall that, in this section, we only discuss the scenario where such cryptographic protection is not required.

4. Otherwise, if P matches a (data) NAT binding, the GaNAT applies normal NAT processing and forwards the packet on the corresponding link.

5. Otherwise, P is silently discarded.

Brief discussion of the algorithm: The fact that the GaNAT replaces the X peer's NLI.IA with its own IP address (in both directions), causes the peers to send subsequent signalling messages to the GaNAT, in the belief that they talk to the their adjacent X peer. The GaNAT transparently forwards the signalling traffic and appropriately translates the fields in the GIMPS header, by making use of the state it creates bindings.

Due to the presence of the GaNATs, no data traffic can be sent from DS to DR until all necessary bindings are in place. The MRI that the NR sees includes as destination address the IP address of the DR (as expected), but as source address the IPGaNATds of the GaNAT that is closest to the NR.

5.1.2 NR-side NSLP-unaware GaNATs

The case of NR-side GaNATs is more subtle, since, in this setting, the DS does not learn the IP address of the DR (which is assumed to be on the same side of the GaNATs as the NR) and the NI does not learn the address of the NR. In this setting we assume that each NR-side GaNAT that is in between two X peers, a priori knows the IP address of the downstream GaNAT. The last GaNAT of this chain is assumed to know the IP address of the DR. In order to clarify this assumption, see, for example, Figure 2. In this figure, GaNAT A is assumed to know the IP address of GaNAT B, GaNAT B is assumed to know the IP address of GaNAT C, and GaNAT C is assumed to know the IP address of the DR. A given GaNAT that knows such an address, in effect anticipates to receive a signalling message from the upstream direction that refers to a data flow that terminates in a downstream node. In other words, such a GaNAT may typically have already a NAT binding in place for the data traffic. We call the IP address of the next downstream GaNAT (or, if the GaNAT is the last in the chain, the DR) the "pending" IP address. In the following description it is denoted by IPNext. How IPNext is made known to each GaNAT (e.g. how

the NAT binding for the data traffic is installed in the GaNAT) is outside the scope of this document.

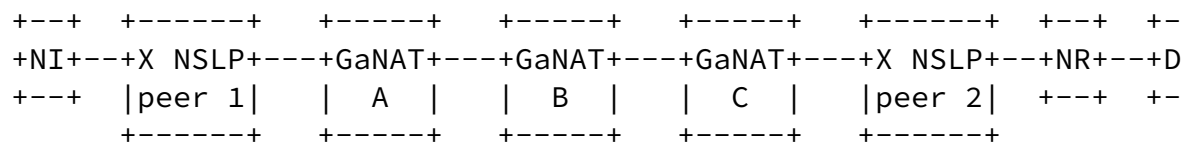

Figure 2: Network with NR-side GaNATs (the public Internet is assumed to be between NI and X NSLP peer 1)

For every arriving IP packet P, an X-unaware, NR-side GaNAT executes the following algorithm.

1. If P has a RAO followed by the GIMPS header with NSLP ID = X, it is identified as a GIMPS QUERY. In this case the GaNAT does the following.

    1. We denote P as GQ. The GaNAT looks at the stack proposal ST in GQ. If it indicates that cryptographic protection is required, the algorithm that is executed is the one described in section Section 6 below.

    2. The GaNAT remembers GQ along with the link on which it arrived. We call this link the "upstream" link.

    3. The GaNAT determines whether or not this GIMPS QUERY is anticipated, i.e. if a pending IPNext exists. If no IPNext is pending, P is discarded (it is a question whether or not an error message should be sent). Otherwise, additional checks may be performed (e.g. a DSInfo object may have to be checked against the GQ). If these checks fail, P is discarded. Otherwise, the GaNAT performs the following.

    4. It searches its table of existing NAT bindings against entries that match the GQ.MRI. A matching entry means that the data flow, to which the signalling refers, already exists.

        + If a matching entry is found, the GaNAT looks at which link the packets of the data flow are forwarded; we call this link the "downstream" link. Further, the GaNAT checks how the headers of the data flow (IP addresses, port numbers, etc) are translated according to this NAT binding. We denote [IP header].SourceIPAddress used on the downstream link as IPGaNATds, and the source port

number as SPNDTGaNATds. Note that the [IP header].DestinationIPAddress of this NAT binding should be equal to IPNext. If it is not, this should be handled as an auditive error condition. The GaNAT may also assign a new source port number to signalling traffic, which is denoted as SPNSTGaNATds.

+ If no matching entry is found, the GaNAT determines, based on its routing table, the link on which packets that match GQ.MRI (excluding GQ.MRI.SourceIPAddress and where GQ.MRI.DestinationIPAddress is replaced with IPNext) would be forwarded. We call this link the "downstream" link. Then, the GaNAT acquires an IP address for itself on the downstream link. (This address could be dynamic or static.) Depending on its type, the GaNAT may also acquire source port numbers for the translation of data traffic. We denote the acquired IP address as IPGaNATds and the source port numbers for data and signalling traffic as SPNDTGaNATds and SPNSTGaNATds respectively.

5. It creates a new GIMPS QUERY packet GQ', as follows.

    1. GQ' <- GQ

    2. GQ'.MRI.SourceIPAddress <- IPGaNATds

    3. GQ'.MRI.DestinationIPAddress <- IPNext.

    4. GQ'.MRI.SourcePort <- SPNDTGaNATds.

    5. GQ'.[IP header].SourceIPAddress <- IPGaNATds

    6. GQ'.[TRANSPORT_LAYER_HEADER].SourcePort <- SPNSTGaNATds

    7. GQ'.[IP header].Destination_IP_Address <- IPNext

    8. GQ'.NLI.IA <- IPGaNATds.

    9. GQ'.S <- true

6. It remembers GQ, GQ' the fact that they are associated, and the associated upstream and downstream links (interfaces).

7. It forwards GQ' on the downstream link.

Steps 2,3, 4 and 5 of the algorithm are analogous to the corresponding steps of the algorithm executed by X-unaware, NI-side

GaNATs, which was described in Section 5.1.1

5.2 NSLP-aware GaNATs

Recall that X may be any NSLP except NATFW. The difference of X-aware GaNATs and X-unaware GaNATs is that the former perform X processing in addition to the processing of the X-unaware GaNATs. Another way to see this is by observing that X-aware GaNATs should provide an "MRI translation service" (MRITS) in addition to normal GIMPS and X processing. The motivation behind the MRITS is for GIMPS to hide from the NSLP that signalling messages traverse an addressing boundary. In other words, the purpose of the MRITS it to make X believe that it is operating in a single IP addressing space. When and how the MRITS is invoked for a particular packet depends on (i) the direction of the packet (i.e. downstream or upstream) and (ii) the location of the GaNAT (i.e. NI-side or NR-side). It should also be noted that certain NSLP layer tasks must be carried out in consistency with the placement of the MRITS. This is to prevent events triggered by X to cause installation of inconsistent state. In order to clarify this, consider the scenario of the QoS NSLP running in a GaNAT that operates according to the mechanisms described in this section. Since the GaNAT only presents a single addressing space to the NSLP (say, the internal addressing space), the packet classifier of the GaNAT's QoS provisioning subsystem should classify packets based on internal addresses only (i.e. it should first translate packets that carry external addresses and then classify them). Whether the MRITS presents internal-only or external-only addresses to the NSLP is not significant, as long as NSLP layer operations are carried out consistently. In the remainder of this section we present the case where internal addresses are presented to the NSLP.

The MRITS is obviously invoked only on GIMPS packets that carry NSLP identifier = X. (For other GIMPS packets the GaNAT may adopt the role of an X-unaware GaNAT. Also, for non-GIMPS packets, normal NAT behaviour applies - whatever "normal" may mean.) Although the MRITS is part of GIMPS processing, in order to clarify our discussion, we view it as a somewhat separate processing step (i.e. like a subroutine). For NI-side, X-aware GaNATs, it holds that

- if a GIMPS/X packet is to be forwarded on the downstream link of an NI-side GaNAT, the MRITS is invoked after the packet has been

processed by X and before it is given to GIMPS, and

- o if a GIMPS/X packet is received on the downstream link, then the MRITS is invoked after GIMPS processing and before the packet is given to X.

The converse holds for NR-side X-aware GaNATs. In particular,

- o if a GIMPS/X packet is to be forwarded on the upstream link of an NI-side GaNAT, the PTS is invoked after the packet has been processed by X and before it is given to GIMPS, and

- o if a GIMPS/X packet is received on the upstream link, then the PTS is invoked after GIMPS processing and before X processing.

Figure 3 illustrates this idea.

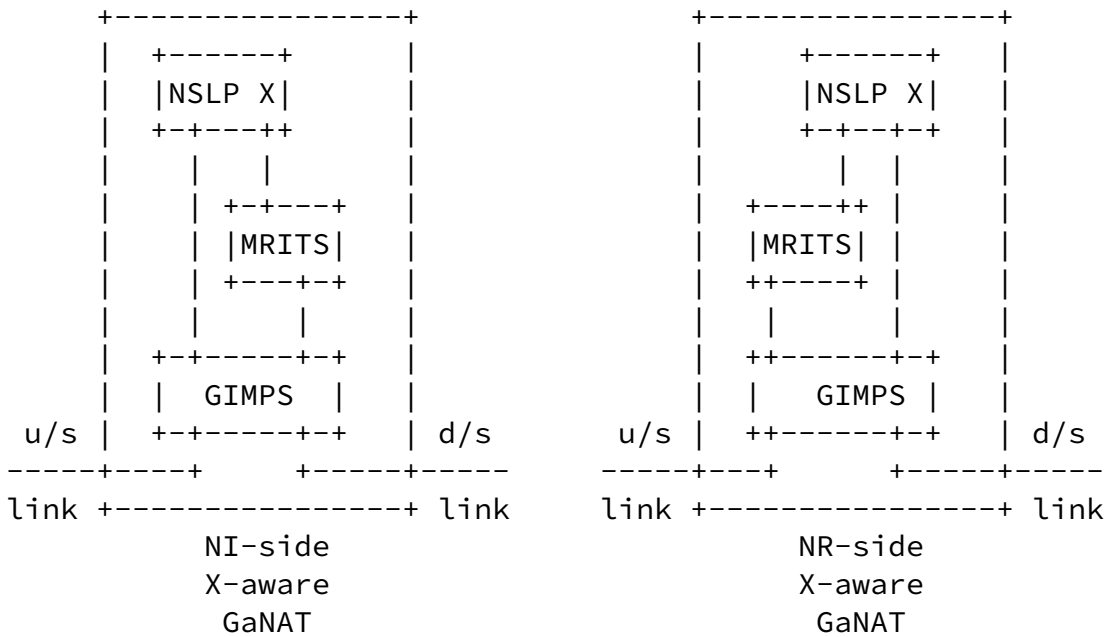

Figure 3: Operation of the MRI Translation Service

The reason for this construction is to give X the impression that it works only with flows that originate and terminate in the internal

address space. We now describe the operation of the MRITS and GIMPS in X-aware GaNATs. An NI-side X-aware GaNAT operates according to the following rules.

1. When X asks for a message to be sent towards the downstream X peer, the MRITS does the following (IPGaNATds and SPNDTGaNATds are obtained similarly to the case of an NSLP-unaware GaNAT).

    1. MRI.SourceIPAddress <- IPGaNATds

    2. MRI.SourcePort <- SPNDTGaNATds

2. Additionally, GIMPS performs the following on the resulting packet before it is forwarded on the downstream link (SPNSTGaNATds is obtained similarly to the case of an NSLP-

unaware GaNAT).

    1. [IP header].SourceIPAddress <- IPGaNATds

    2. [UDP/TCP header].SourcePort <- SPNSTGaNATds

    3. NLI.IA < IPGaNATds

    4. S <- true

3. If a message is received on the downstream link, the MRITS does the following before X is invoked.

    1. MRI.SourceIPAddress <- IPflow

    2. MRI.SourcePort <- SPNDTGaNATus, where IPflow is the IP address of the DS (as seen by the GaNAT) and SPNDTGaNATus is the destination port number used in the original MRI.

4. If, after X processing, a message is to be forwarded on the upstream link, GIMPS performs the following processing (note that no MRITS processing takes place in this case).

    1. [IP header].SourceIPAddress <- IPGaNATus

    2. [IP header].DestinationIPAddress <- IPpeer

3. NLI.IA <- IPGaNATus

4. S <- true, where IPGaNATus is the GaNATs IP address for the upstream link, IPpeer is the IPaddress of the NI (or the next GaNAT in the upstream direction), and IPflow is the IP address of the DS (as seen by the GaNAT). The GaNAT is assumed to determine the correct IPGaNATus and IPpeer from previous communications and in cooperation with GIMPS. [Issue: how exactly should IPGaNATus, IPpeer and IPflow be resolved; i.e. what exactly should the GaNAT remember?]

An NR-side X-aware GaNAT operates according to the following rules.

1. If the packet is received on the upstream link, the MRITS does the following, before X is notified.

    1. P.MRI.SourceIPAddress <- IPGaNATds

    2. P.MRI.DestinationIPAddress <- IPNext, where IPGaNATds is the GaNAT's IP address for the downstream link and IPNext is the address of the DR. IPNext is obtained in a way similar to

the case of an NSLP-unaware GaNAT.

2. If, after X processing, a message is to be forwarded on the downstream link, GIMPS performs the following processing (note that no MRITS processing takes place in this case).

    1. [IP header].SourceIPAddress <- IPGaNATds

    2. [IP header].DestinationIPAddress <- IPNext

    3. NLI.IA <- IPGaNATds

    4. S <- true, where IPGaNATds is the GaNATs IP address for the downstream link, IPNext is the IP address of the DR (or the next GaNAT in the downstream direction). The GaNAT is assumed to determine the correct IPNext in a way similar to the case of an NSLP-unaware GaNAT.

3. When X asks for a message to be sent towards the upstream X peer,

the MRITS does the following.

1. MRI.SourceIPAddress <- IPflow

2. MRI.Destination_IP_Address <- IPGaNATus

4. Additionally, GIMPS performs the following on the resulting packet before it is forwarded on the downstream link.

1. [IP header].SourceIPAddress <- IPGaNATus

2. [IP header].DestinationIPAddress <- IPpeer

3. NLI.IA <- IPGaNATus

4. S <- true, where IPGaNATus is the GaNATs IP address for the upstream link, IPpeer is the IP_address of the NI (or the next GaNAT in the upstream direction), and IPflow is the IP address of the DS. The GaNAT is assumed to determine the correct IPGaNATus and IPpeer fields from previous communications and in cooperation with GIMPS. [question: how exactly should IPGaNATus and IPpeer be resolved; i.e. what exactly should the GaNAT remember]?

5.3 Combination of NSLP-aware and NSLP-unaware GaNATs

In the absence of an adversary, a combination of NSLP-aware and NSLP-unaware GaNATs should work without further specification. However,

in the presence of an adversary, additional security issues may arise from the combination. These issues may introduce opportunities for attack that do not exist in setting where the on-path GaNATs are either all X-aware or all X-unaware.

6. GaNATs in the presence of TLS or IPSec

This section discusses GaNAT traversal for GIMPS in the case where two peers that run a particular NSLP, say NSLP X, require cryptographic protection of the signalling traffic they exchange. As

with the case where no cryptographic protection of signalling traffic is required, the case of the in-between GaNAT(s) being X-unaware is different from the case of them being X-aware.

6.1 NSLP-unaware GaNATs

If the two X peers require a C-mode protocol stack that indicates cryptographic protection, then, after the stack has been agreed by both peers and the underlying cryptographic protection is applied to messages, the GaNAT will be unable to translate the GIMPS header fields, in a way similar to the way described in Section 5.1.1. An approach to cope with this, is to inform the X peers about the presence of the NAT during discovery. This information will enable the X peers, rather than the GaNAT(s) to perform the translation of the fields involved, after the necessary cryptographic operations have been completed. In this scenario, the burden imposed on the GaNAT is considerably less, as the only type of GIMPS messages that it needs to process in a special way, are the GIMPS QUERY and GIMPS RESPONSE messages.

In order to support the scenario of X-unaware GaNATs, a new GIMPS payload type has to be defined that encodes the aforementioned information. We call this payload type the "NAT Traversal Object" (NTO). The NTO is an optional payload in the GIMPS header of a GIMPS QUERY, and is added, and processed, by the GaNAT(s) through which the QUERY traverses. The information in the NTO must enable the two X peers to locally translate the MRI in the same way as it would have been translated by the in-between GaNAT(s) if no cryptographic protection was applied to the signalling traffic. Note that there may be more than one GaNAT between the two X peers. We now describe the algorithm that an X-unaware GaNAT must execute in order to enable the two X peers to reach this goal.

The two types of GaNATs, namely those at the NSIS initiator (NI) side, and those at the NSIS responder (NR) side, follow different algorithms.

6.1.1 NI-side NSLP-unaware GaNATs

For every arriving IP packet P, an X-unaware, NI-side GaNAT executes the following algorithm.

1. If P has a RAO followed by the GIMPS header with an NSLP ID that is not supported, it is identified as a GIMPS QUERY. In this case the GaNAT does the following.

    1. We denote P as GQ. The GaNAT looks at the stack proposal ST in GQ. If it does not indicate that cryptographic protection is required, the algorithm that is executed is the one described in Section 5.1.1 above.

    2. The GaNAT remembers GQ along with the link on which it arrived. We call this link the "upstream" link.

    3. The GaNAT searches its table of existing NAT bindings against entries that match the GQ.MRI. A matching entry means that the data flow, to which the signalling refers, already exists.

        + If a matching entry is found, the GaNAT looks at which link the packets of the data flow are forwarded; we call this link the "downstream" link. Further, the GaNAT checks how the headers of the data flow (IP addresses, source port number, etc) are translated according to this NAT binding.

        + If no matching entry is found, the GaNAT determines, based on its routing table, the link on which packets that match GQ.MRI (excluding GQ.MRI.SourceIPAddress) would be forwarded. We call this link the "downstream" link. Then, the GaNAT acquires an IP address for itself on the downstream link. (This address could be dynamic or static.)

    4. We denote [IP header].SourceIPAddress used on the downstream link as IPGaNATds, and the source port number for the data and signalling traffic as SPNDTGaNATds and SPNSTGaNATds respectively.

    5. It creates a new GIMPS QUERY packet GQ', as follows (note that the new packet contains the same MRI as GQ).

        1. GQ' <- GQ

        2. GQ'.[IP header].SourceIPAddress <- IPGaNATds.

        3. GQ'.[UDP].SourcePort <- SPNSTGaNATds.

        4. GQ'.S <- true

5. It checks whether or not a NTO is included in the GQ.

   - If none is included, it adds a new one to GQ' such that GQ'.NTO=[IPGaNATds || SPNDTGaNATds]

   - If one is included, it replaces it as GQ'.NTO=[IPGaNATds || SPNDTGaNATds]

6. It remembers GQ, GQ' the fact that they are associated, and the associated upstream and downstream links.

7. It forwards GQ' on the downstream link.

2. Otherwise, if P carries a [IP header].DestinationIPAddress that belongs to the GaNAT, and if it is identified as a GIMPS response in D-mode with an NSLP ID that is not supported, the GaNAT does the following (P is denoted as GR).

   1. It searches for a matching GQ' in its buffer. A GQ' is said to be matching if it carries the same cookie value. If none is found, GR is discarded. Otherwise, the GaNAT should also make sure that the session ID in GR is the same as in GQ' and that the NSLP IDs match. If these consistency checks fail, GR should be discarded. Otherwise, the GaNAT constructs a new GIMPS response GR', as follows (note that no changes are made to the MRI).

      1. GR' <- GR

      2. GR'.[IP header].SourceIPAddress <- IPGaNATus, where IPGaNATus = GQ.[IP header].DestinationIPAddress.

      3. GR'.[IP header].DestinationIPAddress <- GQ.NLI.IA

      4. GP'.S <- true.

      5. It checks whether or not a NTO is included in the GQ.

         - If none is included, it adds a new one to GQ' such that GQ'.NTO=[IPGaNATus || PNGaNATus]

         - If one is included, it replaces it such that GQ'.NTO=[

IPGaNATus || PNGaNATus]

6. It remembers GQ, GQ' the fact that they are associated, and the associated upstream and downstream links.

7. It forwards GQ' on the downstream link.

2. It forwards GR' on the upstream link.

3. If no NAT binding for the data traffic was found in step 1.3.2, the GaNAT now installs a NAT binding (for the unidirectional data traffic) which says that "a packet K that arrives on the upstream link and for which it holds that

   + K.[IP header].DestinationIPAddress=GQ.MRI.DestinationIPAddress,

   + K.[IP header].Protocol=GQ.MRI.Protocol, and

   + K.[TCP/UDP header].PortNumbers=GQ.MRI.PortNumbers

   should be forwarded on the upstream link, with [IP header].SourceIPAddress = IPGaNATus.

   Issues: there is a question of whether this NAT binding should also enable data traffic in the opposite direction to traverse the NAT; in order to be able to demultiplex upstream traffic that carries data that belongs to different flows, the GaNAT should keep the necessary per-flow state. From a signalling point of view, however, upstream data traffic that corresponds (on the application level) to the downstream flow to which this GIMPS session refers, is a separate flow for which, dependent on the application, there may or there may not exist a signalling session. If such a signalling session exists, then the GaNAT acts as an NR-side GaNAT for this session. Thus, during the processing of this signalling care has to be taken not to establish a NAT binding for a flow for which a NAT binding already exists. Finally, security issues arise when traffic, for which no signalling exists, is allowed to traverse a GaNAT.

3. Otherwise, if P carries a [IP header].DestinationIPAddress that belongs to the GaNAT, and if it is identified as a GIMPS CONFIRM in D-mode with an NSLP ID that is not supported, the GaNAT does the following (P is denoted as GC).

    1. It creates a new GIMPS CONFIRM packet GC', as follows (note that the variables below refer to the variables that were used in the translation of the GIMPS QUERY that corresponds to GC.

        1. GC' <- GC

        2. GC'.[IP header].SourceIPAddress <- IPGaNATds.

        3. GC'.NLI.IA <- IPGaNATds

        4. GC'.S <- true

        5. It checks whether or not a NTO is included in the GC.

            - If none is included, it adds a new one to GC' such that GC'.NTO=[IPGaNATds || SPNDTGaNATds]

            - If one is included, it replaces it as GC'.NTO=[IPGaNATds || SPNDTGaNATds]

        6. It forwards GC' on the downstream link.

4. Otherwise, if P matches an existing NAT binding, normal NAT processing is applied.

5. Otherwise, P is silently discarded.

6.1.2 NR-side NSLP-unaware GaNATs

As is the case with NR-side NSLP-unaware GaNATs without security, an NR-side NSLP-unaware GaNAT must know a "pending" IP address, as described in Section 5.1.2. This IP address is denoted as IPNext.

For every arriving IP packet P, an NSLP-unaware, NR-side GaNAT executes the following algorithm.

1. If P has a RAO followed by the GIMPS header with an unsupported NSLPID, it is identified as a GIMPS QUERY. In this case the GaNAT does the following.

    1. We denote P as GQ. The GaNAT looks at the stack proposal ST in GQ. If it indicates that no cryptographic protection is required, the algorithm that is executed is the one described in Section 5.1.2 above.

    2. The GaNAT remembers GQ along with the link on which it arrived. We call this link the "upstream" link.

    3. The GaNAT determines whether or not this GIMPS QUERY is anticipated, i.e. if a pending IPNext exists. If no IPNext is pending, GQ is discarded (it is a question whether or not an error message should be sent). Otherwise, additional checks may be performed (e.g. a DSInfo object may have to be

checked against the GQ). If these checks fail, GQ is discarded. Otherwise, the GaNAT performs the following.

    4. It searches its table of existing NAT bindings against entries that match the GQ.MRI. A matching entry means that the data flow, to which the signalling refers, already exists.

        + If a matching entry is found, the GaNAT looks at which link the packets of the data flow are forwarded; we call this link the "downstream" link. Further, the GaNAT checks how the headers of the data flow (IP addresses, port numbers, etc) are translated according to this NAT binding. We denote [IP header].SourceIPAddress used on the downstream link as IPGaNATds, and the port numbers as PNGaNATds. Note that the [IP header].DestinationIPAddress of this NAT binding should be equal to IPNext. If it is not, this should be handled as an auditive error condition. (This check is done as a consistency check.)

        + If no matching entry is found, the GaNAT determines, based

on its routing table, the link on which packets that match GQ.MRI (excluding GQ.MRI.SourceIPAddress and where GQ.MRI.DestinationIPAddress is replaced with IPNext) would be forwarded. We call this link the "downstream" link. Then, the GaNAT acquires an IP address for itself on the downstream link. (This address could be dynamic or static.) Depending on its type, the GaNAT may also acquire (UDP) port numbers for the translation of GQ. We denote the acquired IP address as IPGaNATds and the associated port numbers as PNGaNATds.

5. It creates a new GIMPS QUERY packet GQ', as follows (note that the new packet contains the same MRI as GQ).

    1. GQ' <- GQ

    2. GQ'.[IP header].SourceIPAddress <- IPGaNATds.

    3. GQ'.S <- true

    4. It checks whether or not a NTO is included in the GQ.

        - If none is included, it adds a new one to GQ' such that GQ'.NTO=[IPGaNATds || SPNDTGaNATds]

        - If one is included, it replaces it as GQ'.NTO=[IPGaNATds || SPNDTGaNATds]

    5. It remembers GQ, GQ' the fact that they are associated, and the associated upstream and downstream links.

    6. It forwards GQ' on the downstream link.

The remaining steps of the algorithm are analogous to the algorithm of NSLP-unaware, NI-side GaNATs, which was described in the previous section.

6.1.3 Additional GIMPS peer processing

In the presence of GaNATs on the signalling path between two NSLP peers, and if cryptographic protection of the signalling traffic between these two peers is required, the translation of the GIMPS

header fields that need to be translated for consistency, must be carried out by the X peers. The GIMPS processing that performs this task, is described next. Note that this processing is in addition to the processing described in [1] and that we assume that the in-between GaNATs adopt the behaviour described in the two preceding sections.

A GIMPS peer that receives a GIMPS packet that carries (a) an NSLPID for a supported NSLP, and (b) an NTO in its header, executes the following algorithm, before the processing described in [1] takes place.

1. If the packet is a GIMPS QUERY or CONFIRM in D-mode, denoted G, the peer constructs a new packet, denoted G', as follows.

    1. G' <- G

    2. G'.MRI.Source_IP_Address <- G.NTO.IPGaNATds

    3. G'.MRI.SourcePort <- G.NTO.SPNDTGaNATds

    4. G'.NLI.IA <- G.NTO.IPGaNATds

    5. G'.NTO is removed.

    and forwards G' to GIMPS for further processing. If G is a GIMPS QUERY and local policy demands the installation of state without the reception of a GIMPS CONFIRM message, then the peer must store the NTO carried by G together with the routing state information about the sending GIMPS peer. If G is a GIMPS CONFIRM and local policy demands the installation of state only after reception of a valid CONFIRM, then the peer stores, after validating the cookie in the CONFIRM, the NTO carried by G together with the routing state information about the sending

    GIMPS peer.

2. Otherwise, if the packet is a GIMPS RESPONSE in D-mode, denoted GR, the peer constructs a new packet, denoted GR', as follows.

    1. GR' <- G

2. GR'.MRI.Source_IP_Address <- GR.NTO.IPGaNATus

3. GR'.MRI.SourcePort <- GR.NTO.PNGaNATus

4. GR'.NLI.IA <- GR.NTO.IPGaNATus

5. GR'.NTO is removed.

and forwards GR' to GIMPS for further processing. If the cookie in GR' is verified sucessfully, the peer stores the NTO carried by GR together with the routing state information about the sending GIMPS peer.

A peer that receives a GIMPS packet P (in this case, the packet will be a cryptographically protected GIMPS packet) the peer does the following substitutions after the cryptographic processing is (successfully) completed and before the processing described in [1] takes place.

1. P.MRI.SourceIPAddress <- P.NTO.IPGaNATds

2. P.MRI.SourcePort <- P.NTO.SPNDTGaNATds

3. P.NLI.IA <- P.NTO.IPGaNATds

4. P.NTO is removed.

A peer that intends to send a GIMPS packet (in this case, cryptographic protection will be required for the packet), the peer does the following after the processing described in [1] and before the packet is passed to the process that applies the cryptographic protection. Note that the NTO refers to the NTO that is stored together with the routing state information of the peer that is to receive the packet.

1. P.MRI.Source_IP_Address <- NTO.IPGaNATds

2. P.MRI.SourcePort <- NTO.PNGaNATds

3. P.NLI.IA <- NTO.IPGaNATds

6.2 NSLP-aware GaNATs

The cryptographic protection applies to of signalling messages terminates at NSLP-aware GaNATs. The processing performed by such GaNATs is therefore identical to the processing described in Section 5.2, with the exception that the GaNATs additionally perform cryptographic operations. In this case, there is no requirement for the NSLP to perform any translation for the purposes of NAT traversal.

7. NSIS-unaware NATs

The following may serve as indications for the existence of an NSIS-unaware NAT between two GIMPS peers. These indications can only be detected by the receiver of a GIMPS message. The first occasion these indications may be detected is with the reception of a GIMPS QUERY, typically by the downstream peer. (Note that != denotes inequality).

- The MRI.SourceIPAddress does not belong to the addressing space of the receiving peer.
- The MRI.DestinationIPAddress does not belong to the addressing space of the receiving peer.
- The IP address in the NLI.IA object does not belong to the addressing space of the receiving peer.
- The D flag of a received GIMPS packet denotes downstream direction and the S flag is not set and [IP header].SourceIPAddress != MRI.SourceIPAddress.
- The D flag of a received GIMPS packet denotes upstream direction and the S flag is not set and [IP header].SourceIPAddress != MRI.DestinationIPAddress.
- This is a GIMPS QUERY and [IP header].DestinationIPAddress != MRI.DestinationIPAddress.

Note that these are only indications. In the presence of an adversary, a GIMPS peer may be tricked into believing that an NSIS-unaware NAT exists between itself and one of its neighbouring peers, while in reality this may not be the case.

When a downstream GIMPS peer detects such an indication, it may notify the upstream peer about the error. It may include additional information that enables the upstream peer to construct a GIMPS packet in such a way that, after it traverses the NSIS-unaware NAT, the IP addresses in the MRI field and the NLI.IA object are consistent with those in the IP header (which match the addressing space of the receiving peer). However, this requires the specification of new data structures and formats, processing rules, and requires the peers to maintain additional state.

Unfortunately, this approach is likely to fail in many circumstances. In order to see this, consider the behaviour of an NSIS-unaware NAT when it receives an IP packet. The packet either

1. matches an existing NAT binding in which case its IP header is translated and the packet it is forwarded on another link, or

2. matches an existing policy rule which causes a new binding to be established and then (1) happens, or

3. is discarded because neither (1) nor (2) applies.

With NSIS-unaware NATs it is a matter of local policy (i.e. the rules that exist in case (2) above) whether or not traffic will be allowed to traverse the NAT. This obviously applies to both signalling and data traffic, as an NSIS-unaware NAT is unable to distinguish the two types of traffic. It may be the case that GIMPS node A is unable to contact GIMPS node B which is "behind" a NAT, even if communication in from B to A may be possible because such communication would match a policy rule; typically, in a scenarios where A is towards the NI and B is towards the NR, the NAT would have this behaviour.

Another approach to deal with NSIS-unaware NATs is similar to the NAT traversal approach taken by IKEv2, i.e. by encapsulating GIMPS messages into UDP datagrams, rather than directly into IP datagrams. This technique requires the inclusion of additional fields into a GIMPS QUERY, as follows. The sender adds (a hash of) its own IP address and the IP address of what it believes to be the DR into the GIMPS payload. The receiver of this GIMPS messages compares these addresses to the [IP header].SourceIPAddress and the [IP header].DestinationIPAddress respectively. If at least one of them is unequal, the receiver deduces that a NAT is between sender and receiver. After the detection of a NAT, the remainder of the communication is encapsulated into UDP datagrams that are addressed to a specified port.

Unfortunately, the IKEv2 NAT traversal mechanism cannot be used "as is" for NAT traversal in GIMPS. This is because of a number of reasons, including the following.

- The NAT may use an IP address for the forwarding of data traffic that is different from the IP address it uses to forward GIMPS traffic. Since the NAT is NSIS-unaware it cannot update the MRI in the GIMPS messages such that it matches the translation applies to the data traffic. Moreover, neither the GIMPS sending, nor the GIMPS receiving peer can perform this update; the sending peer cannot predict the translation that the NAT will apply, and the receiving peer does not have enough information to associate data flows to signalling messages.

- It is unclear whether or not the IKEv2 NAT traversal mechanism supports cascades of NATs.

- It seems to be inappropriate to use UDP encapsulation for certain C-mode scenarios. For example, using UDP encapsulation for TCP C-mode would result in GIMPS to appear in TCP over UDP over IP.

8. Security Considerations

The mechanisms proposed in this document give rise to a number of threats that must be considered. In the following, a subset of these threats is mentioned.

8.1 Service Denial Attacks

As described in Section 5.1 and Section 6.1, NSLP-unaware GaNATs create some state whenever they receive a GIMPS QUERY message. This state is necessary in order for the GaNAT to be able to map a GIMPS RESPONSE that arrives from the downstream direction to the corresponding GIMPS QUERY and thereby to perform the required translation.

The threat here is an attacker flooding the GaNAT with maliciously constructed GIMPS QUERIES with the aim of exhausting the GaNAT's memory. The attacker might use a variety of methods to construct such GIMPS QUERIES, including the following.

1. Use as [IP header].SourceIPAddress the address of some other node or an unallocated IP address. This method is also known as IP

spoofing.

2. Use an invalid NSLPID, in order to make sure that all on-path GaNAT(s) will behave like NSLP-unaware GaNATs.

3. For each packet, use a different value for the cookie field.

4. For each packet, use a different value for the session ID field.

5. Combinations of the above.

How vulnerable a GaNAT is to the above service denial attack depends on a variaty of factors, including the following.

- The amount of state allocated at the receipt of a GIMPS QUERY. This amount may vary depending on whether or not the data flow to which the signalling refers, already exists (i.e. whether or not the GaNAT already maintains a NAT binding for it).

- The mechanism that the GaNAT uses to map RESPONSEs to QUERIEs.

- Whether or not the GaNAT acriques dynamic IP addresses and ports for the downstream link.

In order to decrease the exposure of a GaNAT to service denial attacks, the following recommendations are made.

- The GaNAT should perform ingress filtering. This limits the amount of locations from which an attacker can perform IP spoofing without being detected.

- The GaNAT should allocate the minimum amount of state required at the reception of a GIMPS QUERY.

- All state allocated by the GaNAT should timeout according to a local policy. If the GaNAT detects heavy loads (which may indicate a service denial attack in progress), the GaNAT should timeout the state allocated as a result of a received GIMPS QUERY quicker, proportionally to the experienced load.

- The installation of a NAT binding for the data traffic (if such a binding does not exist prior to signalling) should be postponed

until the correct GIMPS REPONSE traverses the NAT.

The service denial threats mentioned in this section do not apply to an NSLP-aware GaNAT, as such a GaNAT is required, in accordance with its local policy, to verify the validity of the cookie(s) before allocating any state, including the state required by the mechanisms in this document.

8.2 Network Intrusions

Although the primary goal of a NAT is to perform address translation between two addressing spaces, NATs are sometimes also used to provide a security service similar to the security service provided by firewalls. That is, a NAT can be configured so that it does not forward packets from the external into the internal network, unless it determines that the packets belong to a communication session that was originally initiated from an internal node and are, as such, solicited.

If an NSLP-unaware GaNAT performs the above security-relevant function in addition to address translation, then the presence of GIMPS signalling and, in particular the mechanisms described in this document, might allow an adversary cause the installation of NAT bindings in the GaNAT using these mechansisms. These NAT bindings would then enable the adversary to inject unsolicited traffic into the internal network, a capability that it may not have in the absence of the mechanisms described in this document.

The administrator of an NSLP-unaware GaNAT should therefore make security-concious decisions regarding the operation of the GaNAT. An NSLP-aware GaNAT, on the other hand, follows an NSLP policy which indicates the required security mechanisms. This policy should account for the fact that this NSLP-aware node performs also NAT and

the associated packet filtering.

9. Acknowledgments

The authors would like to thank Robert Hancock, Cedric Aoun and Martin Stiemerling for their feedback. Furthermore, we would like to mention that this document builds on top of a previous document regarding migration scenarios.

10. IAB Considerations

[Editor's Note: A future version of this document will provide information regarding IAB considerations.

11. IANA Considerations

This document does not require actions by the IANA.

12. Normative References

[1] Schulzrinne, H. and R. Handcock, "GIMPS: General Internet Messaging Protocol for Signalling", draft-ietf-nsis-ntlp-06 (work in progress), May 2005.

[2] Stiemerling, M., Tschofenig, H., and C. Aoun, "NAT/Firewall NSIS Signaling Layer Protocol (NSLP)", draft-ietf-nsis-nslp-natfw-06 (work in progress), May 2005.

Authors' Addresses

Andreas Pashalidis
Siemens
Otto-Hahn-Ring 6
Munich, Bavaria 81739
Germany

Email: Andreas.Pashalidis@siemens.com

Hannes Tschofenig
Siemens
Otto-Hahn-Ring 6
Munich, Bavaria 81739
Germany

Email: Hannes.Tschofenig@siemens.com

Intellectual Property Statement

Disclaimer of Validity

INFORMATION HEREIN WILL NOT INFRINGE ANY RIGHTS OR ANY IMPLIED WARRANTIES OF MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE.

Copyright Statement

Acknowledgment

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