NSIS Working Group
Internet-Draft
Expires: October 9, 2006

M. Stiemerling
NEC
H. Tschofenig
Siemens
C. Aoun
ENST
E. Davies
Folly Consulting

April 7, 2006

NAT/Firewall NSIS Signaling Layer Protocol (NSLP)
draft-ietf-nsis-nslp-natfw-11.txt

Status of this Memo

Copyright Notice

Abstract

This memo defines the NSIS Signaling Layer Protocol (NSLP) for Network Address Translators (NATs) and firewalls. This NSLP allows hosts to signal on the data path for NATs and firewalls to be

configured according to the needs of the application data flows. It enables hosts behind NATs to obtain a public reachable address and hosts behind firewalls to receive data traffic. The overall architecture is given by the framework and requirements defined by the Next Steps in Signaling (NSIS) working group. The network scenarios, the protocol itself, and examples for path-coupled signaling are given in this memo.

Table of Contents

1. Introduction

Firewalls and Network Address Translators (NAT) have both been used throughout the Internet for many years, and they will remain present for the foreseeable future. Firewalls are used to protect networks against certain types of attacks from internal networks and the Internet, whereas NATs provide a virtual extension of the IP address space. Both types of devices may be obstacles to some applications, since they only allow traffic created by a limited set of applications to traverse them, typically those that use protocols with relatively predetermined and static properties (e.g., most HTTP traffic, and other client/server applications). Other applications, such as IP telephony and most other peer-to-peer applications, which have more dynamic properties, create traffic that is unable to traverse NATs and firewalls unassisted. In practice, the traffic of many applications cannot traverse autonomous firewalls or NATs, even when they have additional functionality which attempts to restore the transparency of the network.

Several solutions to enable applications to traverse such entities have been proposed and are currently in use. Typically, application level gateways (ALG) have been integrated with the firewall or NAT to configure the firewall or NAT dynamically. Another approach is

middlebox communication (MIDCOM). In this approach, ALGs external to the firewall or NAT configure the corresponding entity via the MIDCOM protocol [7]. Several other work-around solutions are available, such as STUN [19]. However, all of these approaches introduce other problems that are generally hard to solve, such as dependencies on the type of NAT implementation (full-cone, symmetric, etc), or dependencies on certain network topologies.

NAT and firewall (NATFW) signaling shares a property with Quality of Service (QoS) signaling. The signaling of both must reach any device on the data path that is involved in, respectively, NATFW or QoS treatment of data packets. This means, that for both, NATFW and QoS, it is convenient if signaling travels path-coupled, meaning that the signaling messages follow exactly the same path that the data packets take. RSVP [13] is an example of a current QoS signaling protocol that is path-coupled. [27] proposes the use of RSVP as firewall signaling protocol but does not include NATs.

This memo defines a path-coupled signaling protocol for NAT and firewall configuration within the framework of NSIS, called the NATFW NSIS Signaling Layer Protocol (NSLP). The general requirements for NSIS are defined in [5] and the general framework of NSIS is outlined in [4]. It introduces the split between an NSIS transport layer and an NSIS signaling layer. The transport of NSLP messages is handled by an NSIS Network Transport Layer Protocol (NTLP, with General

Internet Signaling Transport (GIST) [1] being the implementation of the abstract NTLP). The signaling logic for QoS and NATFW signaling is implemented in the different NSLPs. The QoS NSLP is defined in [6].

The NATFW NSLP is designed to request the dynamic configuration of NATs and/or firewalls along the data path. Dynamic configuration includes enabling data flows to traverse these devices without being obstructed, as well as blocking of particular data flows at upstream firewalls. Enabling data flows requires the loading of firewall rules with an action that allows the data flow packets to be forwarded and creating NAT bindings. Blocking of data flows requires the loading of firewalls rules with an action that will deny forwarding of the data flow packets. A simplified example for enabling data flows: A source host sends a NATFW NSLP signaling message towards its data destination. This message follows the data

path. Every NATFW NSLP-enabled NAT/firewall along the data path intercepts these messages, processes them, and configures itself accordingly. Thereafter, the actual data flow can traverse all these configured firewalls/NATs.

It is necessary to distinguish between two different basic scenarios when operating the NATFW NSLP, independent of the type of middlebox to be configured.

1. Both, data sender and data receiver, are NSIS NATFW NSLP aware. This includes the cases where the data sender is logically decomposed from the NSIS initiator or the data receiver logically decomposed from the NSIS receiver, but both sides support NSIS. This scenario assumes deployment of NSIS all over the Internet, or at least at all NATs and firewalls. This scenario is referred as to end-to-end mode operation and is used as base assumption if not otherwise noted.

2. Only one end host or region of the network is NSIS NATFW NSLP aware, either data receiver or data sender. This scenario is referred to as proxy mode operation.

NATFW NSLP provides two basic signaling modes which are sufficient to cope with the various possible scenarios likely to be encountered before and after widespread deployment of NSIS:

CREATE mode: The basic mode for configuring a path downstream from a data sender to a data receiver.

RESERVE-EXTERNAL-ADDRESS (REA) mode: Used to locate upstream NATs/ firewalls and prime them to expect downstream signaling and at NATs to pre-allocate a public address. This is used for data

receivers behind these devices to enable their reachability.

Once there is full deployment of NSIS (i.e., end-to-end mode operations are possible), the requisite NAT and firewall state can be created using only CREATE mode. However, if the data receiver resides in a public addressing realm. If the data receiver resides in a private addressing realm, and needs to preconfigure the edge-NAT/edge-firewall to provide a (publicly) reachable address for use by the data sender, a combination of RESERVE-EXTERNAL-ADDRESS and

CREATE modes is used.

During the introduction of NSIS, it is likely that one or other of the data sender and receiver will not be NSIS aware. In these cases, the NATFW NSLP can utilize NSIS aware middleboxes on the path between the data sender and data receiver to provide proxy NATFW NSLP services (i.e., proxy mode operation). Typically, these boxes will be at the boundaries of the realms in which the end hosts are located.

All modes of operation create NATFW NSLP and NTLP state in NSIS entities. NTLP state allows signaling messages to travel in the forward (downstream) and the reverse (upstream) direction along the path between a NAT/firewall NSLP sender and a corresponding receiver. This state is managed using a soft-state mechanism, i.e., it expires unless it is refreshed from time to time. The NAT bindings and firewall rules being installed during the state setup are bound to the particular signaling session. However, the exact local implementation of the NAT bindings and firewall rules are NAT/ firewall specific.

This memo is structured as follows. Section 2 describes the network environment for NATFW NSLP signaling. Section 3 defines the NATFW signaling protocol and Section 4 defines the message components and the overall messages used in the protocol. The remaining parts of the main body of the document, covers security considerations Section 5, IAB considerations on UNilateral Self-Address Fixing (UNSAF) [15] in Section 6 and IANA considerations in Section 7. Please note that readers familiar with firewalls and NATs and their possible location within networks can safely skip Section 2.

1.1 Terminology and Abbreviations

The key words "MUST", "MUST NOT", "REQUIRED", "SHALL", "SHALL NOT", "SHOULD", "SHOULD NOT", "RECOMMENDED", "MAY", and "OPTIONAL" in this document are to be interpreted as described in [2].

This document uses a number of terms defined in [5] and [4]. The following additional terms are used:

- Policy rule: A policy rule is "a basic building block of a policy-based system. It is the binding of a set of actions to a set of

conditions - where the conditions are evaluated to determine whether the actions are performed" [20]. In the context of NSIS NATFW NSLP, the conditions are the specification of a set of packets to which the rule is applied. The set of actions always contains just a single element per rule, and is limited to either action "deny" or action "allow".

- Reserved policy rule: A policy rule stored at NATs or firewalls for activation by a later, different signaling exchange. This type of policy rule is kept in the NATFW NSLP and is not loaded into the firewall or NAT engine, i.e., it does not affect the data flow handling.

- Installed policy rule: A policy rule in operation at NATs or firewalls. This type of rule is kept in the NATFW NSLP and is loaded into the firewall or NAT engine, i.e., it is affecting the data flow.

- Remembered policy rule: A policy rule stored at NATs and firewalls for immediate use, as soon as the signaling exchange is successfully completed.

- Firewall: A packet filtering device that matches packets against a set of policy rules and applies the actions. In the context of NSIS NATFW NSLP we refer to this device as a firewall.

- Network Address Translator: Network Address Translation is a method by which IP addresses are mapped from one IP address realm to another, in an attempt to provide transparent routing between hosts (see [9]). Network Address Translators are devices that perform this work by modifying packets passing through them.

- Middlebox: "A middlebox is defined as any intermediate device performing functions other than the normal, standard functions of an IP router on the datagram path between a source host and a destination host" [11]. In the context of this document, the term middlebox refers to firewalls and NATs only. Other types of middlebox are outside of the scope of this document.

- Data Receiver (DR): The node in the network that is receiving the data packets of a flow.

- Data Sender (DS): The node in the network that is sending the data packets of a flow.

- NATFW NSLP session or signaling session: An application layer flow of information for which some network control state information is to be manipulated or monitored (as defined in [4]). The control state for NATFW NSLP consists of NSLP state and associated policy rules at a middlebox.

- NSIS peer or peer: An NSIS node with which an NSIS adjacency has been created as defined in [1].

- Edge-NAT: An edge-NAT is a NAT device with a globally routable IP address which is reachable from the public Internet.

- Edge-firewall: An edge-firewall is a firewall device that is located on the border line of an administrative domain.

- Public Network: "A Global or Public Network is an address realm with unique network addresses assigned by Internet Assigned Numbers Authority (IANA) or an equivalent address registry. This network is also referred as external network during NAT discussions" [9].

- Private/Local Network: "A private network is an address realm independent of external network addresses. Private network may also be referred alternately as Local Network. Transparent routing between hosts in private realm and external realm is facilitated by a NAT router" [9].

- Public/Global IP address: An IP address located in the public network according to Section 2.7 of [9].

- Private/Local IP address: An IP address located in the private network according to Section 2.8 of [9].

- Signaling Destination Address (SDA): An IP address generally taken from the public/global IP address range, although, the SDA may in certain circumstances be part of the private/local IP address range. This address is used in REA signaling message exchanges, if the data receiver's IP address is unknown.

1.2 Middleboxes

The term middlebox covers a range of devices which intercept the flow of packets between end hosts and perform actions other than standard forwarding expected in an IP router. As such, middleboxes fall into a number of categories with a wide range of functionality, not all of which is pertinent to the NATFW NSLP. Middlebox categories in the

scope of this memo are firewalls that filter data packets against a

set of filter rules, and NATs that translate packet addresses from one address realm to another address realm. Other categories of middleboxes, such as QoS traffic shapers, are out of scope of this memo.

The term NAT used in this document is a placeholder for a range of different NAT flavors. We consider the following types of NATs:

- Traditional NAT (basic NAT and NAPT)
- Bi-directional NAT
- Twice-NAT
- Multihomed NAT

For definitions and a detailed discussion about the characteristics of each NAT type please see [9].

All types of middleboxes under consideration here, use policy rules to make a decision on data packet treatment. Policy rules consist of a flow identifier which selects the packets to which the policy applies and an associated action; data packets matching the flow identifier are subjected to the policy rule action. A typical flow identifier is the 5-tuple selector which matches the following fields of a packet to configured values:

- Source and destination IP addresses
- Transport protocol number
- Transport source and destination port numbers

Actions for firewalls are usually one or more of:

- Allow: forward data packet
- Deny: block data packet and discard it
- Other actions such as logging, diverting, duplicating, etc

Actions for NATs include (amongst many others):

- Change source IP address and transport port number to a globally routeable IP address and associated port number.

- Change destination IP address and transport port number to a private IP address and associated port number.

It should be noted that a middlebox may contain two logical representations of the policy rule. The policy rule has a representation within the NATFW NSLP, comprising the message routing information (MRI) of the NTLP and NSLP information (such as the rule action). The other representation is the implementation of the NATFW NSLP policy rule within the NAT and firewall engine of the particular device. Refer to Appendix C for further details.

1.3 General Scenario for NATFW Traversal

The purpose of NSIS NATFW signaling is to enable communication between endpoints across networks, even in the presence of NAT and firewall middleboxes that have not been specially engineered to facilitate communication with the application protocols used. This removes the need to create and maintain application layer gateways for specific protocols that have been commonly used to provide transparency in previous generations of NAT and firewall middleboxes. It is assumed that these middleboxes will be statically configured in such a way that NSIS NATFW signaling messages themselves are allowed to reach the locally installed NATFW NSLP daemon. NSIS NATFW NSLP signaling is used to dynamically install additional policy rules in all NATFW middleboxes along the data path that will allow transmission of the application data flow(s). Firewalls are configured to forward data packets matching the policy rule provided by the NSLP signaling. NATs are configured to translate data packets matching the policy rule provided by the NSLP signaling. An additional capability, that is an exception to the primary goal of NSIS NATFW signaling, is that the NATFW nodes can request blocking of particular data flows instead of enabling these flows at upstream firewalls.

The basic high-level picture of NSIS usage is that end hosts are located behind middleboxes, meaning that there is a middlebox on the

data path from the end host in a private network and the external network (NATFW in Figure 1). Applications located at these end hosts try to establish communication with corresponding applications on other such end hosts. They trigger the NSIS entity at the local host to control provisioning for middlebox traversal along the prospective data path (e.g., via an API call). The NSIS entity in turn uses NSIS NATFW NSLP signaling to establish policy rules along the data path, allowing the data to travel from the sender to the receiver unobstructed.

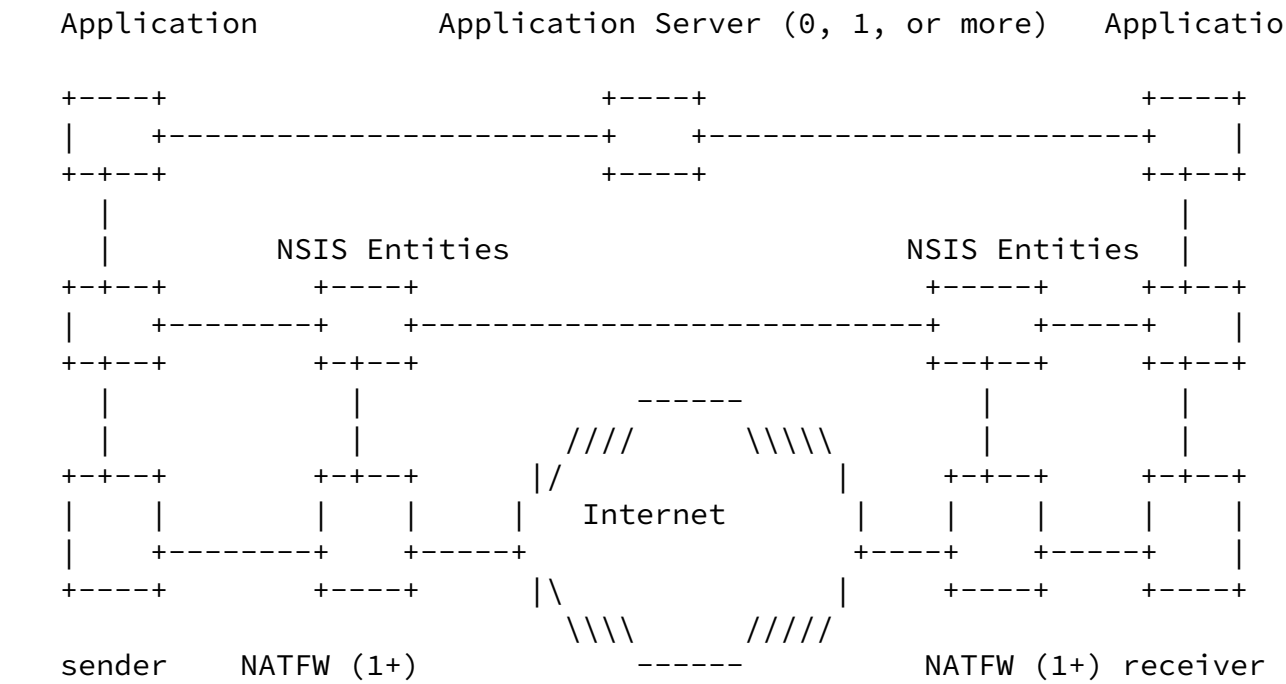


Figure 1: Generic View of NSIS with NATs and/or Firewalls

For end-to-end NATFW signaling, it is necessary that each firewall and each NAT along the path between the data sender and the data receiver implements the NSIS NATFW NSLP. There might be several NATs and FWs in various possible combinations on a path between two hosts. Section 2 presents a number of likely scenarios with different

combinations of NATs and firewalls.

2. Network Deployment Scenarios using the NATFW NSLP

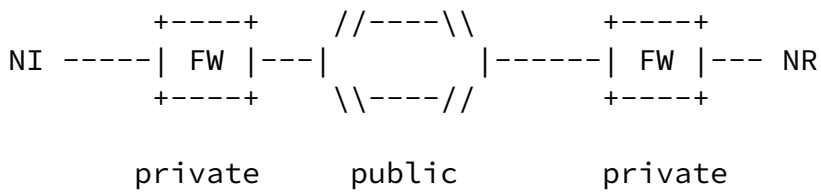
This section introduces several scenarios for middlebox placement within IP networks. Middleboxes are typically found at various different locations, including at Enterprise network borders, within enterprise networks, as mobile phone network gateways, etc. Usually, middleboxes are placed more towards the edge of networks than in network cores. Firewalls and NATs may be found at these locations either alone, or they may be combined; other categories of middleboxes may also be found at such locations, possibly combined with the NATs and/or firewalls. Using combined middleboxes typically reduces the number of network elements needed.

NSIS initiators (NI) send NSIS NATFW NSLP signaling messages via the regular data path to the NSIS responder (NR). On the data path, NATFW NSLP signaling messages reach different NSIS nodes that implement the NATFW NSLP. Each NATFW NSLP node processes the signaling messages according to Section 3 and, if necessary, installs policy rules for subsequent data packets.

Each of the following sub-sections introduces a different scenario for a different set of middleboxes and their ordering within the topology. It is assumed that each middlebox implements the NSIS NATFW NSLP signaling protocol.

2.1 Firewall Traversal

This section describes a scenario with firewalls only; NATs are not involved. Each end host is behind a firewall. The firewalls are connected via the public Internet. Figure 2 shows the topology. The part labeled "public" is the Internet connecting both firewalls.



FW: Firewall
NI: NSIS Initiator
NR: NSIS Responder

Figure 2: Firewall Traversal Scenario

Each firewall on the data path must provide traversal service for NATFW NSLP in order to permit the NSIS message to reach the other end host. All firewalls process NSIS signaling and establish appropriate

policy rules, so that the required data packet flow can traverse them.

There are several very different ways to place firewalls in a network topology. To distinguish firewalls located at network borders, such as administrative domains, from others located internally, the term edge-firewall is used. A similar distinction can be made for NATs, with an edge-NAT fulfilling the equivalent role.

2.2 NAT with two private Networks

Figure 3 shows a scenario with NATs at both ends of the network.

Therefore, each application instance, the NSIS initiator and the NSIS responder, are behind NATs. The outermost NAT, known as the edge-NAT, at each side is connected to the public Internet. The NATs are generically labeled as MB (for middlebox), since those devices certainly implement NAT functionality, but can implement firewall functionality as well.

Only two middleboxes MB are shown in Figure 3 at each side, but in general, any number of MBs on each side must be considered.

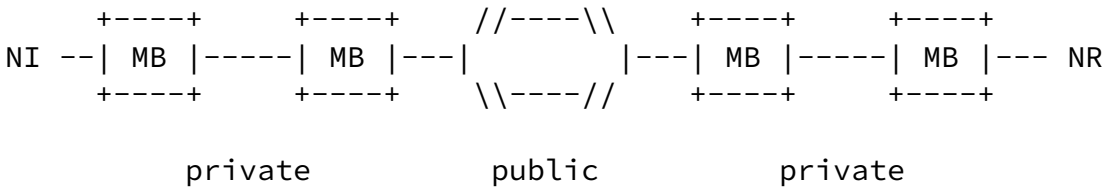


Figure 3: NAT with two Private Networks Scenario

Signaling traffic from NI to NR has to traverse all the middleboxes on the path, and all the middleboxes must be configured properly to allow NSIS signaling to traverse them. The NATFW signaling must configure all middleboxes and consider any address translation that will result from this configuration in further signaling. The sender (NI) has to know the IP address of the receiver (NR) in advance, otherwise it will not be possible to send any NSIS signaling messages towards the responder. Note that this IP address is not the private IP address of the responder. Instead a NAT binding (including a public IP address) has to be previously installed on the NAT that subsequently allows packets reaching the NAT to be forwarded to the receiver within the private address realm. This generally requires further support from an application layer protocol for the purpose of discovering and exchanging information. The receiver might have a

number of ways to learn its public IP address and port number (including the NATFW NSLP) and might need to signal this information to the sender using the application level signaling protocol.

2.3 NAT with Private Network on Sender Side

This scenario shows an application instance at the sending node that is behind one or more NATs (shown as generic MB, see discussion in Section 2.2). The receiver is located in the public Internet.

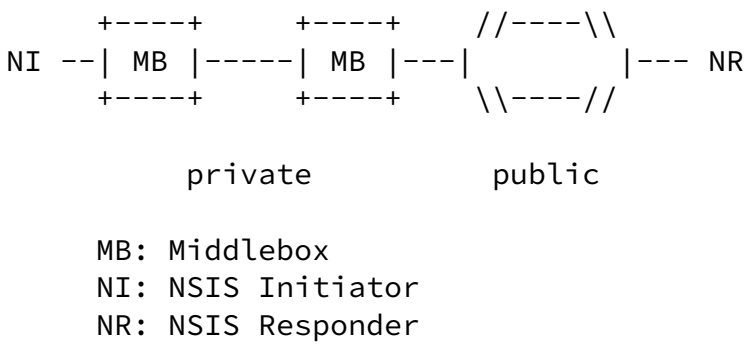


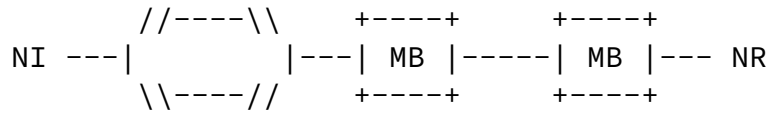
Figure 4: NAT with Private Network on Sender Side Scenario

The traffic from NI to NR has to traverse middleboxes only on the sender's side. The receiver has a public IP address. The NI sends its signaling message directly to the address of the NSIS responder. Middleboxes along the path intercept the signaling messages and configure the policy rules accordingly.

The data sender does not necessarily know whether the receiver is behind a NAT or not, hence, it is the receiving side that has to detect whether itself is behind a NAT or not. As described in Section 3.8.2 NSIS can also provide help for this procedure.

2.4 NAT with Private Network on Receiver Side Scenario

The application instance receiving data is behind one or more NATs shown as MB (see discussion in Section 2.2).



MB: Middlebox
NI: NSIS Initiator
NR: NSIS Responder

Figure 5: NAT with Private Network on Receiver Scenario

Initially, the NSIS responder must determine its publicly reachable IP address at the external middlebox and notify the NSIS initiator about this address. One possibility is that an application level protocol is used, meaning that the public IP address is signaled via this protocol to the NI. Afterwards the NI can start its signaling towards the NR and therefore establish the path via the middleboxes in the receiver side private network.

This scenario describes the use case for the RESERVE-EXTERNAL-ADDRESS mode of the NATFW NSLP.

2.5 Both End Hosts behind twice-NATs

This is a special case, where the main problem arises from the need to detect that both end hosts are logically within the same address space, but are also in two partitions of the address realm on either side of a twice-NAT (see [9] for a discussion of twice-NAT functionality).

Sender and receiver are both within a single private address realm but the two partitions potentially have overlapping IP address ranges. Figure 6 shows the arrangement of NATs. This is a common configuration in networks, particularly after the merging of companies that have used the same private address space, resulting in overlapping address ranges.

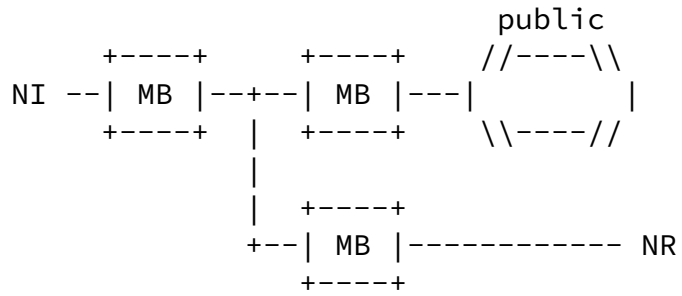


Figure 6: NAT to Public, Sender and Receiver on either side of a twice-NAT Scenario

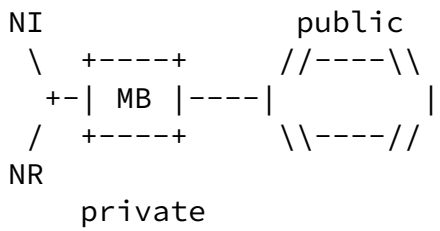
The middleboxes shown in Figure 6 are twice-NATs, i.e., they map IP addresses and port numbers on both sides, meaning the mapping of source and destination address at the private and public interfaces.

This scenario requires the assistance of application level entities, such as a DNS server. The application level entities must handle requests that are based on symbolic names, and configure the middleboxes so that data packets are correctly forwarded from NI to NR. The configuration of those middleboxes may require other middlebox communication protocols, such as MIDCOM [7]. NSIS signaling is not required in the twice-NAT only case, since middleboxes of the twice-NAT type are normally configured by other means. Nevertheless, NSIS signaling might be useful when there are also firewalls on the path. In this case NSIS will not configure any policy rule at twice-NATs, but will configure policy rules at the firewalls on the path. The NSIS signaling protocol must be at least robust enough to survive this scenario. This requires that twice-NATs must implement the NATFW NSLP also and participate in NATFW sessions but they do not change the configuration of the NAT, i.e., they only read the address mapping information out of the NAT and translate the Message Routing Information (MRI, [1]) within the NSLP and NTLP accordingly. For more information see Appendix C.4

2.6 Both End Hosts Behind Same NAT

When NSIS initiator and NSIS responder are behind the same NAT (thus being in the same address realm, see Figure 7), they are most likely not aware of this fact. As in Section 2.4 the NSIS responder must determine its public IP address in advance and transfer it to the NSIS initiator. Afterwards, the NSIS initiator can start sending the

signaling messages to the responder's public IP address. During this process, a public IP address will be allocated for the NSIS initiator at the same middlebox as for the responder. Now, the NSIS signaling and the subsequent data packets will traverse the NAT twice: from initiator to public IP address of responder (first time) and from public IP address of responder to responder (second time). This is the worst case in which both sender and receiver obtain a public IP address at the NAT, and the communication path is certainly not optimal in this case.



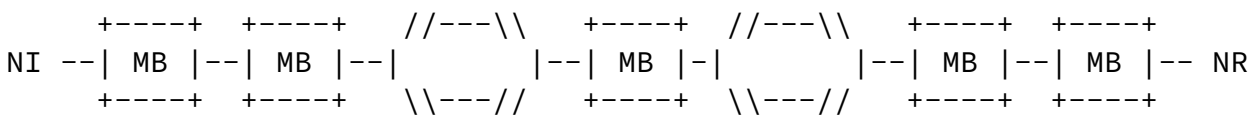
MB: Middlebox
NI: NSIS Initiator
NR: NSIS Responder

Figure 7: NAT to Public, Both Hosts Behind Same NAT

2.7 IPv4/v6 NAT with two Private Networks

This scenario combines the use case described in Section 2.2 with the IPv4 to IPv6 transition scenario involving address and protocol translation, i.e., using Network Address and Protocol Translators (NAT-PT, [10]).

The difference from the other scenarios is the use of IPv6 to IPv4 (and vice versa) address and protocol translation. Additionally, the base NTLP must support transport of messages in mixed IPv4 and IPv6 networks where some NSIS peers provide translation.



private public public private
IPv4 IPv6

MB: Middlebox
NI: NSIS Initiator
NR: NSIS Responder

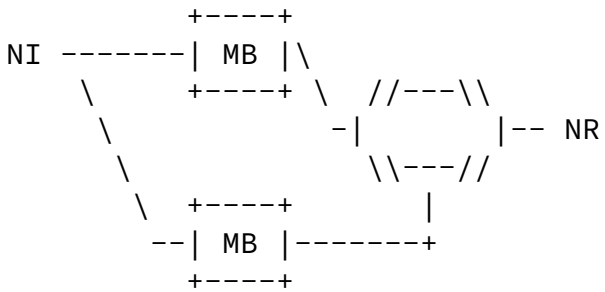
Figure 8: IPv4/v6 NAT with two Private Networks

This scenario needs the same type of application level support as described in Section 2.5, and so the issues relating to twice-NATs apply here as well.

Note that the current form of IPv4/v6 NAT known as the Network Address Translator - Protocol Translator (NAT-PT) [10] is being removed from the set of recommended mechanisms for general usage in IPv4/IPv6 transitions. This scenario is therefore not expected to be commonly seen.

2.8 Multihomed Network with NAT

The previous sub-sections sketched network topologies where several NATs and/or firewalls are ordered sequentially on the path. This section describes a multihomed scenario with two NATs placed on alternative paths to the public network.



private public

MB: Middlebox
NI: NSIS Initiator
NR: NSIS Responder

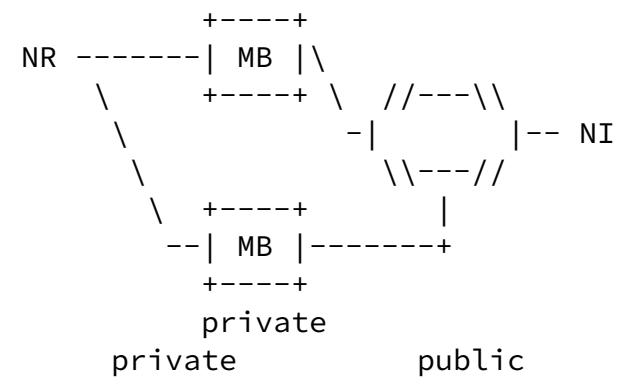
Figure 9: Multihomed Network with Two NATs

Depending on the destination, either one or the other middlebox is used for the data flow. Which middlebox is used, depends on local policy or routing decisions. NATFW NSLP must be able to handle this situation properly, see Section 3.8.2 for an extended discussion of this topic with respect to NATs.

2.9 Multihomed Network with Firewall

This section describes a multihomed scenario with two firewalls placed on alternative paths to the public network (Figure 10). The routing in the private and public network decides which firewall is being taken for data flows. Depending on the data flow's direction, either outbound or inbound, a different firewall could be traversed. This is a challenge for the REA mode of the NATFW NSLP where the NSIS

responder is located behind these firewalls within the private network. The REA mode is used to block a particular data flow on an upstream firewall. NSIS must route the REA mode message upstream from NR to NI probably without knowing which path the data traffic will take from NI to NR (see also Appendix B.



MB: Middlebox
NI: NSIS Initiator
NR: NSIS Responder

Figure 10: Multihomed Network with two Firewalls

3. Protocol Description

This section defines messages, objects, and protocol semantics for the NATFW NSLP.

3.1 Policy Rules

Policy rules, bound to a session, are the building blocks of middlebox devices considered in the NATFW NSLP. For firewalls the policy rule usually consists of a 5-tuple, source/destination addresses, transport protocol, and source/destination port numbers, plus an action, such as allow or deny. For NATs the policy rule consists of the action 'translate this address' and further mapping information, that might be, in the simplest case, internal IP address

and external IP address.

The NATFW NSLP carries, in conjunction with the NTLP's Message Routing Information (MRI), the policy rules to be installed at NATFW peers. This policy rule is an abstraction with respect to the real policy rule to be installed at the respective firewall or NAT. It conveys the initiator's request and must be mapped to the possible configuration on the particular used NAT and/or firewall in use. For pure firewalls one or more filter rules must be created and for pure NATs one or more NAT bindings must be created. In mixed firewall and NAT boxes, the policy rule must be mapped to filter rules and bindings observing the ordering of the firewall and NAT engine. Depending on the ordering, NAT before firewall or vice versa, the firewall rules must carry public or private IP addresses. However, the exact mapping depends on the implementation of the firewall or NAT which is different for each vendor.

The policy rule at the NATFW NSLP level comprises the message routing information (MRI) part, carried in the NTLP, and the information available in the NATFW NSLP. The information provided by the NSLP is stored in the 'extend flow information' (NATFW_EFI) and 'data terminal information' (NATFW_DTINFO) objects, and the message type, in particular the flow direction. Additional information, such as the external IP address and port number, stored in the NAT or firewall, will be used as well. The MRI carries the filter part of the NAT/firewall-level policy rule that is to be installed.

3.2 Basic Protocol Overview

The NSIS NATFW NSLP is carried over the General Internet Signaling Transport (GIST, the implementation of the NTLP) defined in [1]. NATFW NSLP messages are initiated by the NSIS initiator (NI), handled by NSIS forwarders (NF) and received by the NSIS responder (NR). It is required that at least NI and NR implement this NSLP, intermediate

NFs only implement this NSLP when they provide relevant middlebox functions. NSIS forwarders that do not have any NATFW NSLP functions just forward these packets as they have no interest in them.

A Data Sender (DS), intending to send data to a Data Receiver (DR) must first initiate NATFW NSLP signaling. This causes the NI associated with the data sender (DS) to launch NSLP signaling towards

the address of data receiver (DR) (see Figure 11). Although it is expected that the DS and the NATFW NSLP NI will usually reside on the same host, this specification does not rule out scenarios where the DS and NI reside on different hosts, the so-called proxy mode (see Section 1.)

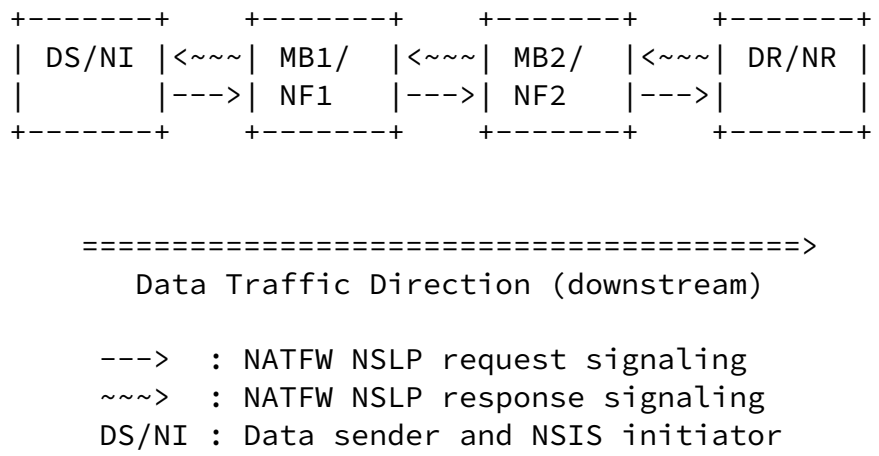


Figure 11: General NSIS signaling

The normal sequence of NSLP events is as follows:

- NSIS initiators generate NATFW NSLP request messages and send these towards the NSIS responder. Note, that the NSIS initiator may not necessarily be the data sender but may be the data receiver, for instance, when using the RESERVE-EXTERNAL-ADDRESS (REA) message.

- NSLP request messages are processed each time a NF with NATFW NSLP support is traversed. These nodes process the message, check local policies for authorization and authentication, possibly create policy rules, and forward the signaling message to the next NSIS node. The request message is forwarded until it reaches the

NSIS responder.

- NSIS responders will check received messages and process them if applicable. NSIS responders generate response messages and send them hop-by-hop back to the NI via the same chain of NFs (traversal of the same NF chain is guaranteed through the established reverse message routing state in the NTLP). Note, that the NSIS responder may not necessarily be the data receiver but may be any intermediate NSIS node that terminates the forwarding, for example, in a proxy mode case where an edge-NAT is replying to requests.

- The response message is processed at each NF that has been included in the prior signaling session setup.

- Once the NI has received a successful response, the data sender can start sending its data flow to the data receiver.

Because NATFW NSLP signaling follows the data path from DS to DR, this immediately enables communication between both hosts for scenarios with only firewalls on the data path or NATs on the sender side. For scenarios with NATs on the receiver side certain problems arise, as described in Section 2.

When the NR and the NI are located in different address realms and the NR is located behind a NAT, the NI cannot signal to the NR address directly. The DR and NR are not reachable from the NIs using the private address of the NR and thus NATFW signaling messages cannot be sent to the NR/DR's address. Therefore, the NR must first obtain a NAT binding that provides an address that is reachable for the NI. Once the NR has acquired a public IP address, it forwards this information to the DS via a separate protocol. This application layer signaling, which is out of scope of the NATFW NSLP, may involve third parties that assist in exchanging these messages.

The same holds partially true for NRs located behind firewalls that block all traffic by default. In this case, NR must tell its upstream firewalls of inbound NATFW NSLP signaling and corresponding data traffic. Once the NR has informed the upstream firewalls, it can start its application level signaling to initiate communication with the NI. This application layer signaling, which is out of scope of the NATFW NSLP, may involve third parties that assist in exchanging these messages. This mechanism can be used by machines hosting services behind firewalls as well. In this case, the NR informs the upstream firewalls as described, but does not need to communicate this to the NIs.

NATFW NSLP signaling supports this scenario by using the REA mode of

operation

1. The NR acquires a public address by signaling on the reverse path (NR towards NI) and thus making itself available to other hosts. This process of acquiring public addresses is called reservation. During this process the NR reserves publicly reachable addresses and ports suitable for further usage in application level signaling and the publicly reachable address for further NATFW NSLP signaling. However, the data traffic will not be allowed to use this address/port initially (see next point).

2. The NI signals directly to the NR, as the NI would do if there is no NAT in between, and creates policy rules at middleboxes. Note, that the reservation mode will only allow forwarding of signaling messages, but not data flow packets. Policy rules allowing forwarding of data flow packets set up by the prior REA mode signaling will be 'activated' by the signaling from NI towards NR. The RESERVE-EXTERNAL-ADDRESS (REA) mode of operation is detailed in Section 3.8.2

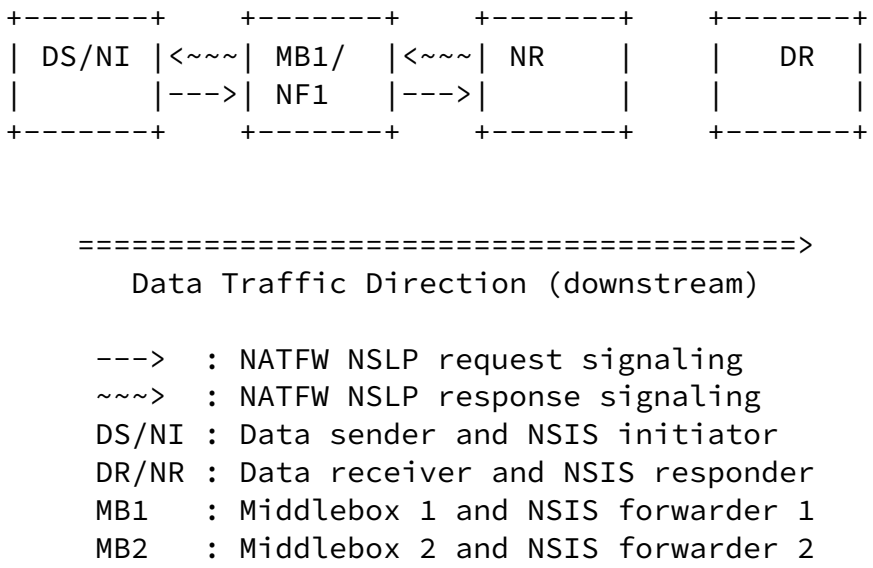


Figure 12: A NSIS proxy mode signaling

The above usage assumes that both ends of a communication support NSIS, but fails when NSIS is only deployed at one end of the path. In this case only one of the receiving or sending side is NSIS aware and not both at the same time. NATFW NSLP supports this scenario (i.e., the DR does not support NSIS) by using a proxy mode, as

described in Section 3.8.7; the proxy mode operation also supports scenarios with a data sender that does not support NSIS, i.e. the data receiver must act to enable data flows towards itself.

The basic functionality of the NATFW NSLP provides for opening firewall pin holes and creating NAT bindings to enable data flows to traverse these devices. Firewalls are normally expected to work on a 'deny-all' policy, meaning that traffic not explicitly matching any firewall filter rule will be blocked. Similarly, the normal behavior of NATs is to block all traffic that does not match any already configured/installed binding or session. However, some scenarios require support of firewalls having 'allow-all' policies, allowing data traffic to traverse the firewall unless it is blocked explicitly. Data receivers can utilize NATFW NSLP's REA message with action set to 'deny' to install policy rules at upstream firewalls to block unwanted traffic.

The protocol works on a soft-state basis, meaning that whatever state is installed or reserved on a middlebox will expire, and thus be de-installed or forgotten after a certain period of time. To prevent premature removal of state that is needed for ongoing communication, the NATFW NI involved will have to specifically request a session extension. An explicit NATFW NSLP state deletion capability is also provided by the protocol.

If the actions requested by a NATFW NSLP message cannot be carried out, NFs and the NR should return a failure, such that appropriate actions can be taken. They can do this either during a the request message handling (synchronously) by sending an error RESPONSE message, or at any time (asynchronously) by sending a notification message.

The next sections define the NATFW NSLP message types and formats, protocol operations, and policy rule operations.

3.2.1 Message Types

The protocol uses five messages types:

- CREATE: a request message used for creating, changing, refreshing, and deleting CREATE NATFW NSLP sessions, i.e., open the data path from DS to DR.

- RESERVE-EXTERNAL-ADDRESS (REA): a request message used for reserving, changing, refreshing, and deleting REA NATFW NSLP sessions. REA messages are forwarded to the edge-NAT or edge-firewall and allow inbound CREATE messages to be forwarded to the NR. Additionally, REA messages reserve an external address and,

if applicable, port number at an edge-NAT.

- TRACE: a request message to trace all involved NATFW NSLP nodes in a particular signaling session.

- NOTIFY: an asynchronous message used by NATFW peers to alert upstream NATFW peers about specific events (especially failures).

- RESPONSE: used as a response to CREATE, REA, and TRACE request messages.

3.2.2 Classification of RESPONSE Messages

RESPONSE messages will be generated synchronously by NSIS Forwarders and Responders to report success or failure of operations or some information relating to the session or a node.

All RESPONSE messages MUST carry a NATFW_INFO object which contains a severity class code and a response code (see Section 4.2.4). This section defines terms for groups of RESPONSE messages depending on the severity class.

- Successful RESPONSE: Messages carrying NATFW_INFO with severity class 'Success' (0x2).

- Informational RESPONSE: Messages carrying NATFW_INFO with severity class 'Informational' (0x1) (normally only used with NOTIFY messages).

- Error RESPONSE: Messages carrying NATFW_INFO with severity class other than 'Success' or 'Informational'.

3.2.3 NATFW NSLP Signaling Sessions

The general idea of signaling sessions is described in [4]. There is signaling session state stored at the NTLP layer and at the NATFW NSLP level. The signaling session state for the NATFW NSLP consists comprises NSLP state and the associated policy rules at a middlebox.

A NATFW NSLP signaling session can conceptually be in different states, implementations may use other or even more states. The signaling session can have these states at a node:

- Pending: The signaling session has been created and the node is waiting for a RESPONSE message to the request message. A signaling session in state 'Pending' MUST be marked as 'Dead' if

no corresponding RESPONSE message has been received within the time of the locally granted session lifetime of the forwarded request message (as described in Section 3.4).

- Established: The signaling session is established, i.e, the signaling has been successfully performed. A signaling session in state 'Established' MUST be marked as 'Dead' if no refresh message has been received within the time of the locally granted session lifetime of the RESPONSE message (as described in Section 3.4).

- Dead: The node has received an error RESPONSE message for the signaling session and the signaling session can be deleted.

- Transit: The node has received an asynchronous message, i.e., a NOTIFY, and can delete the signaling session if needed. When a node has received a NOTIFY message (for instance, indicating a route change) it marks it as 'Transit' and deletes this session if it is unused for some time specific to the local node. This idle time does not need to be fixed, since it can depend on the node local maintenance cycle, i.e., the session could be deleted if the node runs it garbage collection cycle.

3.3 Basic Message Processing

All NATFW messages are subject to some basic message processing when received at a node, independent of request or response messages. Initially, the syntax of the NSLP message is checked and a RESPONSE message with an appropriate error of class 'Protocol error' (0x1) code is generated if any problem is detected. If a message is delivered to the NATFW NSLP, this implies that the NTLP layer has been able to correlate it with the SID and MRI entries in its database. There is therefore enough information to identify the source of the message and routing information to route the message back to the NI through an established chain of MAs since the NATFW NSLP always requests reliable delivery resulting in the NTLP using C-mode. The message is not further forwarded if any error in the syntax is detected. The specific response codes stemming from the processing of objects are described in the respective object definition section (see Section 4). After passing this check, the NATFW NSLP node MUST first perform the checks defined on session ownership in Section 3.6 and authentication/authorization in Section 3.7. Further processing is executed only if these tests have been successfully passed, otherwise the processing stops and an error RESPONSE is returned, as described in these sections.

Further message processing stops whenever an error RESPONSE message is generated, and the request message is discarded.

3.4 Calculation of Session Lifetime

NATFW NSLP sessions, and the corresponding policy rules which may have been installed, are maintained via a soft-state mechanism. Each session is assigned a lifetime and the session is kept alive as long as the lifetime is valid. After the expiration of the lifetime, sessions and policy rules MUST be removed automatically and resources bound to them MUST be freed as well. Session lifetime is handled at every NATFW NSLP node. The NSLP forwarders and NSLP responder MUST NOT trigger lifetime extension refresh messages (see Section 3.8.3): this is the task of the NSIS initiator. This section describes how the session lifetime is set within a signaling session.

The NSIS initiator MUST choose a session lifetime value (expressed in seconds) before sending any message, including the initial message

which creates the session, to other NSLP nodes. The session lifetime value is calculated based on:

- The number of lost refresh messages that NFs should cope with;
- the end-to-end delay between the NI and NR;
- network vulnerability due to session hijacking ([8], session hijacking is made easier when the NI does not explicitly remove the session);
- the user application's data exchange duration, in terms of time and networking needs. This duration is modeled as M x R, with R the message refresh period (in seconds) and M as a multiplier for R;

The RSVP specification [13] provides an appropriate algorithm for calculating the session lifetime as well as means to avoid refresh message synchronization between sessions. [13] recommends:

1. The refresh message timer to be randomly set to a value in the range [0.5R, 1.5R].
2. To avoid premature loss of state, lt (with lt being the session lifetime) must satisfy lt >= (K + 0.5)*1.5*R, where K is a small integer. Then in the worst case, K-1 successive messages may be lost without state being deleted. Currently K = 3 is suggested as the default. However, it may be necessary to set a larger K value for hops with high loss rate. Other algorithms could be used to define the relation between the session lifetime and the refresh message period; the algorithm provided is only given as an example.

This requested lifetime value lt is stored in the NATFW_LT object of the NSLP message.

NATFW NFs processing the request message along the path may change the requested lifetime to fit their needs and/or local policy. If an NF changes the lifetime value, it MUST store the new value in the 'lifetime' object. NFs MUST NOT increase the lifetime value; they MAY reject the requested lifetime immediately and MUST generate an

error RESPONSE message of class 'Signaling session failures' (0x6) with error response code 'Requested lifetime is too big' (0x02) upon rejection. The NSLP request message is forwarded until it reaches the NSLP responder. The NSLP responder may reject the requested lifetime value and MUST generate an error RESPONSE message of class 'Signaling session failures' (0x6) with response code 'Requested lifetime is too big' (0x02) upon rejection. The NSLP responder MAY also lower the requested lifetime to an acceptable value (based on its local policies). The NSLP responder generates a successful RESPONSE for the received request message, sets the lifetime value to the above granted lifetime and sends the message back hop-by-hop towards NSLP initiator.

Each NSLP forwarder processes the RESPONSE message, reads and stores the granted lifetime value. The forwarders MUST accept the granted lifetime, as long as this value is less than or equal to their proposed value. For received values greater than the proposed value, NSLP forwarders MUST generate an RESPONSE message of class 'Signaling session failures' (0x6) with response code 'Requested lifetime is too big' (0x02). Figure 13 shows the procedure with an example, where an initiator requests 60 seconds lifetime in the CREATE message and the lifetime is shortened along the path by the forwarder to 20 seconds and by the responder to 15 seconds.

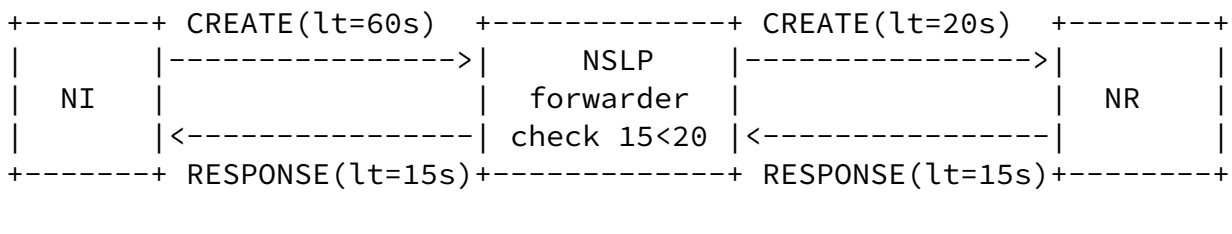


Figure 13: Lifetime Setting Example

3.5 Message Sequencing

NATFW NSLP messages need to carry an identifier so that all nodes along the path can distinguish messages sent at different points in time. Messages can be lost along the path or duplicated. So all NATFW NSLP nodes should be able to identify either old messages that have been received before (duplicated), or the case that messages have been lost before (loss). For message replay protection it is necessary to keep information about messages that have already been received and requires every NATFW NSLP message to carry a message sequence number (MSN), see also Section 4.2.6.

The MSN MUST be set by the NI and MUST NOT be set or modified by any other node. The initial value for the MSN MUST be generated randomly and MUST be unique only within the session for which it is used. The NI MUST increment the MSN by one for every message sent. Once the MSN has reached the maximum value, the next value it takes is zero. All NATFW NSLP nodes MUST use the algorithm defined in [3] to detect MSN wrap-arounds.

NSIS forwarders and the responder store the MSN from the initial CREATE or REA packet which creates the session as the start value for the session. NFs and NRs MUST include the received MSN value in the corresponding RESPONSE message that they generate.

When receiving a request message, a NATFW NSLP node uses the MSN given in the message to determine whether the state being requested is different to the state already installed. The message MUST be discarded if the received MSN value is equal to or lower than the stored MSN value. Such a received MSN value can indicate a duplicated and delayed message or replayed message. If the received MSN value is greater than the already stored MSN value, the NATFW NSLP MUST update its stored state accordingly, if permitted by all security checks (see Section 3.6 and Section 3.7), and stores the updated MSN value accordingly.

3.6 Session Ownership

Proof of session ownership is a fundamental part of the NATFW NSLP signaling protocol. It is used to validate the origin of a request, i.e., invariance of the message sender. Only request messages demonstrating a valid session ownership are processed further. Within the NATFW NSLP, the NSIS initiator is the ultimate session owner for all request messages. A proof of ownership MUST be provided for any request message sent downstream or upstream. All intermediate NATFW NSLP nodes MUST use this proof of ownership to validate the message's origin.

All NATFW nodes along the path must be able to verify that the sender of a request is the same entity that initially created the session. Generally, the path taken spans different administrative domains and cannot rely on using a common authentication scheme. This requirement demands a scheme independent of the local authentication scheme in use and administrative requirements being enforced. Relying on a public key infrastructure (PKI) for the purpose of prove of session ownership is not reasonable due to deployment problems of a global PKI.

The NATFW NSLP relies on the session ID (SID) carried in the NTLP for proof of session ownership. The session ID MUST be generated in a random way. Messages for a particular session are handled by the NTLP to the NATFW NSLP for further processing. Messages carrying a different session ID not associated with any NATFW NSLP are subject to the regular processing for new NATFW NSLP sessions.

3.7 Authentication, Authorization, and Policy Decisions

NATFW NSLP nodes receiving signaling messages MUST first check whether this message is authenticated and authorized to perform the requested action. The necessary information for these checks can be carried in the NATFW_CREDENTIAL object. NATFW NSLP nodes requiring more information than provided MUST generate an error RESPONSE of class 'Permanent failure' (0x5) with response code 'Authentication failed' (0x01) or with response code 'Authorization failed' (0x02).

The NATFW NSLP is expected to run in various environments, such as IP-based telephone systems, enterprise networks, home networks, etc. The requirements on authentication and authorization are quite different between these use cases. While a home gateway, or an Internet cafe, using NSIS may well be happy with a "NATFW signaling coming from inside the network" policy for authorization of signaling, enterprise networks are likely to require more strongly authenticated/authorized signaling. This enterprise scenario may require the use of an infrastructure and administratively assigned identities to operate the NATFW NSLP.

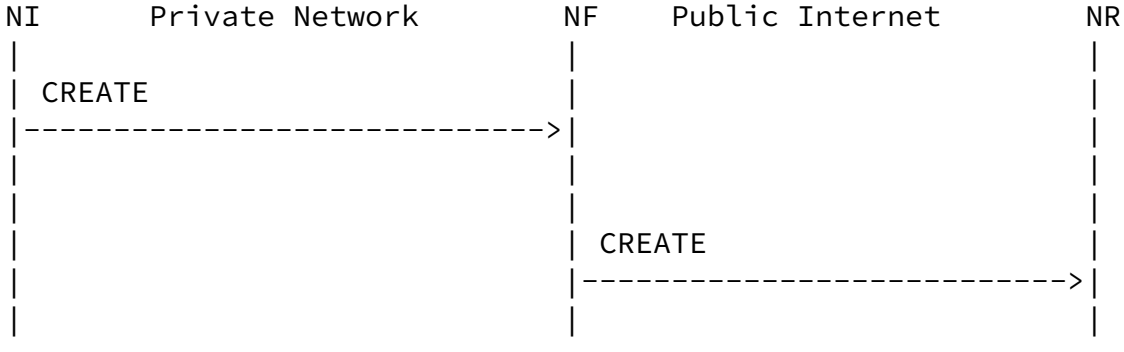
Once the NI is authenticated and authorized, another step is performed. The requested policy rule for the session is checked against a set of policy rules, i.e., whether the requesting NI is allowed to request the policy rule to be loaded in the device. If this fails the NF or NR must send an error RESPONSE of class 'Permanent failure' (0x5) and with response code 'Authorization failed' (0x02).

3.8 Protocol Operations

This section defines the protocol operations including, how to create sessions, maintain them, and how to reserve addresses. All the NATFW NSLP protocol messages MUST be transported via C-mode handling by the NTLP and MUST NOT be piggybacked into D-mode NTLP messages used during the NTLP path discovery/refresh phase. The usage of the NTLP by protocol messages is described in detail in Section 4.

3.8.1 Creating Sessions

Allowing two hosts to exchange data even in the presence of middleboxes is realized in the NATFW NSLP by use of the CREATE request message. The NI (either the data sender or a proxy) generates a CREATE message as defined in Section 4.3.1 and hands it to the NTLP. The NTLP forwards the whole message on the basis of the message routing information towards the NR. Each NSIS forwarder along the path that implements NATFW NSLP, processes the NSLP message. Forwarding is thus managed NSLP hop-by-hop but may pass transparently through NSIS forwarders which do not contain NATFW NSLP functionality and non-NSIS aware routers between NSLP hop way points. When the message reaches the NR, the NR can accept the request or reject it. The NR generates a response to the request and this response is transported hop-by-hop towards the NI. NATFW NSLP forwarders may reject requests at any time. Figure 14 sketches the message flow between NI (DS in this example), a NF (e.g., NAT), and NR (DR in this example).



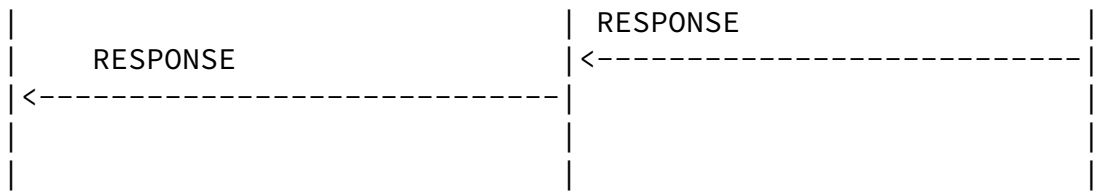


Figure 14: CREATE message flow with success RESPONSE

There are several processing rules for a NATFW peer when generating and receiving CREATE messages, since this message type is used for creating new signaling sessions, updating existing, extending the lifetime and deleting signaling session. The three latter functions operate in the same way for all kinds of request message, and are therefore described in separate sections:

- Extending the lifetime of signaling sessions is described in Section 3.8.3.
- Deleting signaling sessions is described in Section 3.8.4.
- Updating policy rules is described in Section 3.11.

For an initial CREATE message creating a new NATFW NSLP session, the processing of CREATE messages is different for every NATFW node type:

- NSLP initiator: An NI only generates CREATE messages and hands them over to the NTLP. The NI should never receive request messages and MUST discard it.
- NATFW NSLP forwarder: NFs that are unable to forward the request message to the next hop MUST generate an error RESPONSE of class 'Permanent failure' (0x6) with response code 'Did not reach the NR' (0x07). This case may occur if the NTLP layer cannot find an NATFW NSLP peer, either another NF or the NR, and returns an error via the GIST API. The NSLP message processing at the NFs depends on the middlebox type:
  - NAT: When the initial CREATE message is received at the public side of the NAT, it looks for a reservation made in advance, by using a REA message (see Section 3.8.2). The matching process

considers the received MRI information and the stored MRI information, as described in Section 3.9. If no matching reservation can be found, i.e. no reservation has been made in advance, the NSLP MUST return an error RESPONSE of class 'Signaling session failure' (0x6) with response code 'No reservation found matching the MRI of the CREATE request' (0x03) MUST be generated. If there is a matching reservation, the NSLP stores the data sender's address (and if applicable port number) as part of the source address of the policy rule ('the remembered policy rule') to be loaded and forwards the message with the destination address set to the internal (private in most cases) address of NR. When the initial CREATE message is received at the private side, the NAT binding is allocated, but not activated (see also Appendix C.3). The MRI information is updated to reflect the address, and if applicable port, translation. The NSLP message is forwarded

towards the NR with source address set to the NAT's external address from the newly remembered binding.

* Firewall: When the initial CREATE message is received, the NSLP just remembers the requested policy rule, but does not install any policy rule. Afterwards, the message is forwarded towards the NR.

* Combined NAT and firewall: Processing at combined firewall and NAT middleboxes is the same as in the NAT case. No policy rules are installed. Implementations MUST take into account the order of packet processing in the firewall and NAT functions within the device. This will be referred to as 'order of functions' and is generally different depending on whether the packet arrives at the external or internal side of the middlebox.

- NSLP receiver: NRs receiving initial CREATE messages MUST reply with a success RESPONSE of class 'Success' (0x2) with response code set to 'All successfully processed' (0x01), if they accept the CREATE request message. Otherwise they MUST generate a RESPONSE message with a suitable response code. RESPONSE messages are sent back NSLP hop-by-hop towards the NI, irrespective of the response codes, either success or error.

Remembered policy rules at middleboxes MUST be only installed upon receiving a corresponding successful RESPONSE message with the same SID and MSN as the CREATE message that caused them to be remembered. This is a countermeasure to several problems, for example, wastage of resources due to loading policy rules at intermediate NFs when the CREATE message does not reach the final NR for some reason.

Processing of a RESPONSE message is different for every NSIS node type:

- NSLP initiator: After receiving a successful RESPONSE, the data path is configured and the DS can start sending its data to the DR. After receiving an error RESPONSE message, the NI MAY try to generate the CREATE message again or give up and report the failure to the application, depending on the error condition.

- NSLP forwarder: NFs install the remembered policy rules, if a successful RESPONSE message with matching SID and MSN is received. If an ERROR RESPONSE message with matching SID and MSN is received, the session is marked as dead, no policy rule is installed and the remembered rule is discarded.

- NSIS responder: The NR should never receive RESPONSE messages and MUST silently drop any such messages received.

3.8.2 Reserving External Addresses

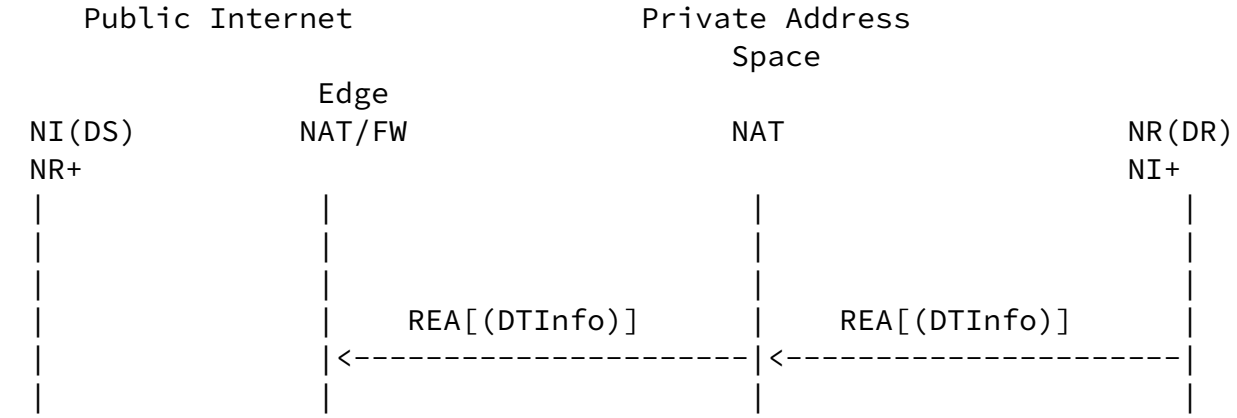
NSIS signaling is intended to travel end-to-end, even in the presence of NATs and firewalls on-path. This works well in cases where the data sender is itself behind a NAT or a firewall as described in Section 3.8.1. For scenarios where the data receiver is located behind a NAT or a firewall and it needs to receive data flows from outside its own network (usually referred to as inbound flows, see Figure 5) the problem is more troublesome.

NSIS signaling, as well as subsequent data flows, are directed to a particular destination IP address that must be known in advance and reachable. Data receivers must tell the local NSIS infrastructure (i.e., the upstream firewalls/NATs) about incoming NATFW NSLP

signaling and data flows before they can receive these flows. It is necessary to differentiate between data receivers behind NATs and behind firewalls for understanding the further NATFW procedures. Data receivers that are only behind firewalls already have a public IP address and they need only to be reachable for NATFW signaling. Unlike data receivers behind just firewalls, data receivers behind NATs do not have public IP addresses; consequently they are not reachable for NATFW signaling by entities outside their addressing realm.

The preceding discussion addresses the situation where a DR node that wants to be reachable is unreachable because the NAT lacks a suitable rule with the 'allow' action which would forward inbound data. However, in certain scenarios, a node situated behind upstream firewalls that do not block inbound data traffic (firewalls with "default to allow") unless requested might wish to prevent traffic being sent to it from specified addresses. In this case, NSIS NATFW signaling can be used to achieve this by installing a policy rule with its action set to 'deny' using the same mechanisms as for 'allow' rules.

The required result is obtained by sending a RESERVE-EXTERNAL-ADDRESS (REA) message in the upstream direction of the intended data flow. When using this functionality the NSIS initiator for the 'Reserve External Address' signaling is typically the node that will become the DR for the eventual data flow. To distinguish this initiator from the usual case where the NI is associated with the DS, the NI is denoted by NI+ and the NSIS responder is similarly denoted by NR+.



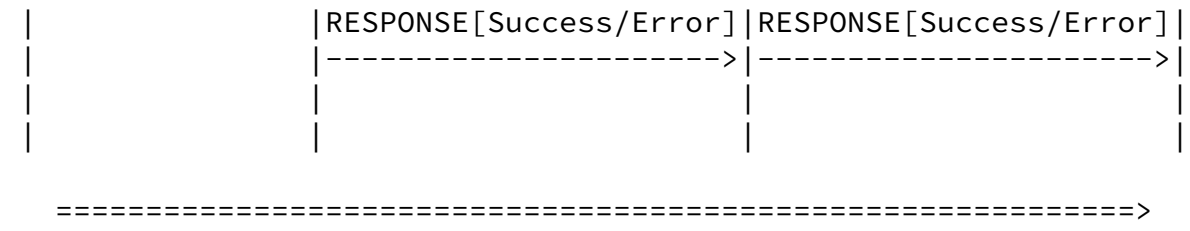


Figure 15: Reservation message flow for DR behind NAT or firewall

Figure 15 shows the REA message flow for enabling inbound NATFW NSLP signaling messages. In this case the roles of the different NSIS entities are:

- The data receiver (DR) for the anticipated data traffic is the NSIS initiator (NI+) for the RESERVE-EXTERNAL-ADDRESS (REA) message, but becomes the NSIS responder (NR) for following CREATE messages.

- The actual data sender (DS) will be the NSIS initiator (NI) for later CREATE messages and may be the NSIS target of the signaling (NR+).

- It may be necessary to use a signaling destination address (SDA) as the actual target of the REA message (NR+) if the DR is located behind a NAT and the address of the DS is unknown. The SDA is an arbitrary address in the outermost address realm on the other side of the NAT from the DR. Typically this will be a suitable public IP address when the 'outside' realm is the public Internet. This choice of address causes the REA message to be routed through the NATs towards the outermost realm and would force interception of the message by the outermost NAT in the network at the boundary between the private address and the public address realm (the edge-NAT). It may also be intercepted by other NATs and firewalls on the path to the edge-NAT.

Basically, there are two different signaling scenarios. Either

1. the DR behind the NAT/firewall knows the IP address of the DS in advance,

2. or the address of DS is not known in advance.

Case 1 requires the NATFW NSLP to request the path-coupled message routing method (PC-MRM) from the NTLP. The REA message MUST be sent with PC-MRM (see Section 5.8.1 in [1]) with the direction set to 'upstream'. The handling of case 2 depends on the situation of DR: If DR is solely located behind a firewall, the REA message MUST be sent with the PC-MRM, direction 'upstream', and data flow source IP address set to wildcard. If DR is located behind a NAT, the REA message MUST be sent with the loose-end message routing method (LE-MRM, see Section 5.8.2 in [1]), the destination-address set to the signaling destination address (SDA, see also Appendix A). For scenarios with DR being behind a firewall, special conditions apply (applicability statement, Appendix B). The data receiver is challenged to determine whether it is solely located behind firewalls or NATs, for choosing the right message routing method. This decision can depend on a local configuration parameter, possibly given through DHCP, or it could be discovered through other non-NSLP related testing of the network configuration.

For case 2 with NAT, the NI+ (which could be on the data receiver DR or on any other host within the private network) sends the REA message targeted to the signaling destination address. The message routing for the REA message is in the reverse direction to the normal message routing used for path-coupled signaling where the signaling is sent downstream (as opposed to upstream in this case). When establishing NAT bindings (and an NSIS session) the signaling direction does not matter since the data path is modified through route pinning due to the external IP address at the NAT. Subsequent NSIS messages (and also data traffic) will travel through the same NAT boxes. However, this is only valid for the NAT boxes, but not for any intermediate firewall. That is the reason for having a separate CREATE message enabling the reservations made with REA at the NATs and either enabling prior reservations or creating new pinholes at the firewalls which are encountered on the downstream path depending on whether the upstream and downstream routes coincide.

The REA signaling message creates an NSIS NATFW session at any intermediate NSIS NATFW peer(s) encountered, independent of the message routing method used. Furthermore, it has to be ensured that the edge-NAT or edge-firewall device is discovered as part of this process. The end host cannot be assumed to know this device -

instead the NAT or firewall box itself is assumed to know that it is located at the outer perimeter of the network. Forwarding of the REA message beyond this entity is not necessary, and MUST be prohibited as it may provide information on the capabilities of internal hosts. It should be noted, that it is the outermost NAT or firewall that is the edge-device that must be found during this discovery process. For instance, when there are a NAT and afterwards a firewall on the outbound path at the network border, the firewall is the edge-firewall. All messages must be forwarded to the topology-wise outermost edge-device, to ensure that this devices knows about the signaling sessions for incoming CREATE messages. However, the NAT is still the edge-NAT because it has a public globally routable IP address on its public side: this is not affected by any firewall between the edge-NAT and the public network.

Possible edge arrangements are:

Public Net ---------- Private net ----------

| Public Net|--|Edge-FW|--|FW|...|FW|--|DR|

| Public Net|--|Edge-FW|--|Edge-NAT|...|NAT or FW|--|DR|

| Public Net|--|Edge-NAT|--|NAT or FW|...|NAT or FW|--|DR|

The edge-NAT or edge-firewall device closest to the public realm responds to the REA message with a successful RESPONSE message. An edge-NAT includes an NATFW_EXT_IP object (see Section 4.2.2), carrying the public reachable IP address, and if applicable port number.

There are several processing rules for a NATFW peer when generating and receiving REA messages, since this message type is used for creating new reserve signaling sessions, updating existing, extending the lifetime and deleting signaling session. The three latter functions operate in the same way for all kinds of request message, and are therefore described in separate sections:

- Extending the lifetime of signaling sessions is described in Section 3.8.3.
- Deleting signaling sessions is described in Section 3.8.4.
- Updating policy rules is described in Section 3.11.

The NI+ MAY include a NATFW_DTINFO_IPv4 object in the REA message

when using the LE-MRM. The LE-MRM does not include enough

information for some types of NATs (basically, those NATs which also translate port numbers) to perform the address translation. This information is provided in the NATFW_DTINFO_IPv4 (see Section 4.2.7). This information SHOULD include at least the 'dst port number' and 'protocol' fields, in the DTInfo object as these may be required by en-route NATs, depending on the type of the NAT. All other fields MAY be set by the NI+ to restrict the set of possible NIs. An edge-NAT will use the information provided in the NATFW_DTINFO_IPv4 object to allow only NATFW CREATE message with the MRI matching ('src IPv4/v6 address', 'src port number', 'protocol') to be forwarded. A NAT requiring information carried in the NATFW_DTINFO_IPv4 can generate a number of error RESPONSE messages of class 'Signaling session failures' (0x6):

- 'Requested policy rule denied due to policy conflict' (0x04)
- 'DTINFO object is required' (0x07)
- 'Requested value in sub_ports field in NATFW_EFI not permitted' (0x08)
- 'Requested IP protocol not supported' (0x09)
- 'Plain IP policy rules not permitted -- need transport layer information' (0x0A)
- 'source IP address range is too large' (0x0C)
- 'destination IP address range is too large' (0x0D)
- 'source L4-port range is too large' (0x0E)
- 'destination L4-port range is too large' (0x0F)

Processing of REA messages is specific to the NSIS node type:

- NSLP initiator: NI+ only generate REA messages. When the data sender's address information is known in advance the NI+ MAY include a NATFW_DTINFO_IPv4 object in the REA message (as described above). When the data sender's IP address is not known, the NI+ MUST NOT include a NATFW_DTINFO_IPv4 object. The NI

should never receive request messages and MUST silently discard it.

- NSLP forwarder: The NSLP message processing at NFs depends on the middlebox type:

  * NAT: NATs check whether the message is received at the external (public in most cases) address or at the internal (private) address. If received at the external address a NF MUST generate an error RESPONSE of class 'Protocol error' (0x3) with response code 'Received REA request message on external side' (0x0B) MUST be generated. If received at the internal (private address) and the NATFW_EFI object contains the action 'deny', an error RESPONSE of class 'Protocol error' (0x3) with response code 'Requested rule action not applicable' (0x06) MUST be generated. If received at the internal address, an IP address, and if applicable a port, is reserved. If it is an edge-NAT and there is no edge-firewall beyond, the NSLP message is not forwarded any further and a successful RESPONSE message is generated containing an NATFW_EXT_IP object holding the translated address, and if applicable port, information in the binding reserved as a result of the REA message. The RESPONSE message is sent back towards the NI+. If it is not an edge-NAT, the NSLP message is forwarded further using the translated IP address as signaling source address and including the translated IP address/port in the MRI. The edge-NAT or any other NAT MAY reject REA messages not carrying a NATFW_DTINFO_IPv4 object or if the address information within this object is invalid or is not compliant with local policies (e.g., the information provided relates to a range of addresses ('wildcarded') but the edge-NAT requires exact information about DS' IP address and port) with the above mentioned response codes.

  * Firewall: Non edge-firewalls remember the requested policy rule, keep session state, and forward the message. Edge-firewalls stop forwarding the request message. The policy rule is immediately loaded if the action in the NATFW_EFI object is set to 'deny' and the node is an edge-firewall. The policy rule is remembered, but not activated, if the action in the

NATFW_EFI object is set to 'allow'. In both cases, a successful RESPONSE message is generated.

* Combined NAT and firewall: Processing at combined firewall and NAT middleboxes is the same as in the NAT case.

- NSLP receiver: This type of message should never be received by any NR+ and it MUST generate an error RESPONSE message of class 'Permanent failure' (0x5) with response code 'No edge-device here' (0x06).

Processing of a RESPONSE message is different for every NSIS node type:

- NSLP initiator: Upon receiving a successful RESPONSE message, the NI+ can rely on the requested configuration for future inbound sessions. If the response contains an NATFW_EXT_IP object, the NI can use IP address and port pairs carried for further application signaling. After receiving a error RESPONSE message, the NI+ MAY try to generate the REA message again or give up and report the failure to the application, depending on the error condition.

- NSLP forwarder: NFs simply forward this message as long as they keep state for the requested reservation, if the RESPONSE message contains NATFW_INFO object with class set to 'Success' (0x2). If the RESPONSE message contains NATFW_INFO object with class set not to 'Success' (0x2), the session is marked as dead.

- NSIS responder: This type of message should never be received by any NR+. The NF should never receive response messages and MUST silently discard it.

Reservations with action 'allow' made with REA MUST be enabled by a subsequent CREATE message. A reservation made with REA (independent of selected action) is kept alive as long as the NI+ refreshes the particular signaling session and it can be reused for multiple, different CREATE messages. An NI+ may decide to teardown a reservation immediately after receiving a CREATE message. Without using CREATE Section 3.8.1 or REA in proxy mode Section 3.8.7 no data traffic will be forwarded to DR beyond the edge-NAT or edge-firewall. The only function of REA is to ensure that subsequent CREATE messages

traveling towards the NR will be forwarded across the public-private boundary towards the DR. Correlation of incoming CREATE messages to REA reservation states is described in Section 3.9.

3.8.3 NATFW Session Refresh

NATFW NSLP sessions are maintained on a soft-state basis. After a specified timeout, sessions and corresponding policy rules are removed automatically by the middlebox, if they are not refreshed. Soft-state is created by CREATE and REA and the maintenance of this state must be done by these messages. State created by CREATE must be maintained by CREATE, state created by REA must be maintained by REA. Refresh messages, are messages carrying the same session ID as the initial message and a NATFW_LT lifetime object with a lifetime greater than zero. Messages with the same SID but carrying a different MRI are treated as updates of the policy rules and are processed as defined in Section 3.11. Every refresh request message MUST be acknowledged by an appropriate response message generated by the NR. Upon reception by each NSIS forwarder, the state for the given session ID is extended by the session refresh period, a period of time calculated based on a proposed refresh message period. The

lifetime extension of a session is calculated as current local time plus proposed lifetime value (session refresh period). Section 3.4 defines the process of calculating lifetimes in detail.

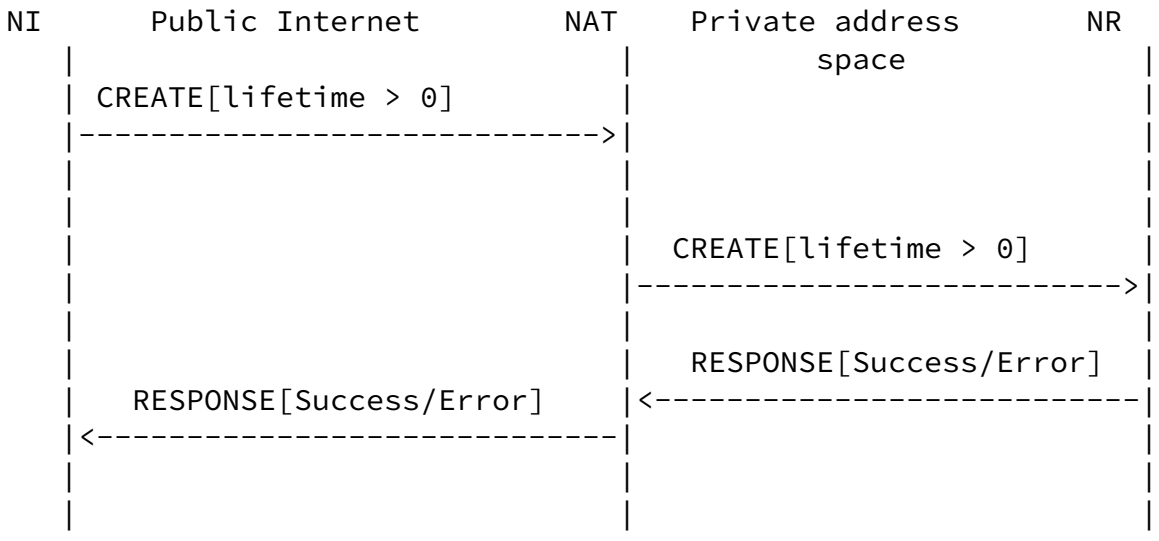


Figure 16: Successful Refresh Message Flow, CREATE as example

Processing of session refresh CREATE and REA messages is different for every NSIS node type:

- NSLP initiator: The NI/NI+ can generate session refresh CREATE/REA messages before the session times out. The rate at which the refresh CREATE/REA messages are sent and their relation to the session state lifetime is discussed further in Section 3.4.

- NSLP forwarder: Processing of this message is independent of the middlebox type and is as described in Section 3.4.

- NSLP responder: NRs accepting a session refresh CREATE/REA message generate a successful RESPONSE message, including the granted lifetime value of Section 3.4 in a NATFW_LT object.

3.8.4 Deleting Sessions

NATFW NSLP sessions can be deleted at any time. NSLP initiators can trigger this deletion by using a CREATE or REA messages with a lifetime value set to 0, as shown in Figure 17. Whether a CREATE or REA message type is used, depends on how the session was created.

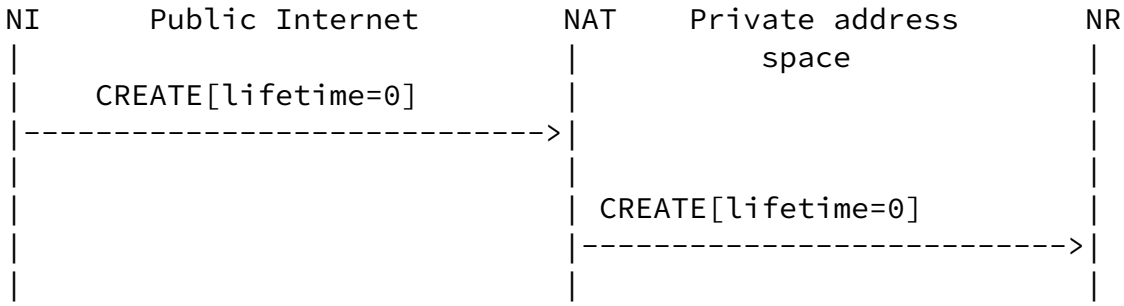


Figure 17: Delete message flow, CREATE as example

NSLP nodes receiving this message delete the session immediately.

Policy rules associated with this particular session MUST be also deleted immediately. This message is forwarded until it reaches the final NR. The CREATE/REA request message with a lifetime value of 0, does not generate any response, neither positive nor negative, since there is no NSIS state left at the nodes along the path.

NSIS initiators can use CREATE/REA message with lifetime set to zero in an aggregated way, such that a single request message is terminating multiple signaling sessions. NIs can follow this procedure if the like to aggregate session deletion requests: The NI uses the CREATE or REA request message with the session ID set to zero and the MRI's source-address set to its used IP address. All other fields of the respective sessions to be terminated are set as well, otherwise these fields are completely wildcarded. The NSLP message is transfered to the NTLP requesting 'explicit routing' as described in Sections 5.2.1 and 7.1.4. in [1].

The downstream NF receiving such an aggregated request message MUST reject the request with an error RESPONSE of class 'Permanent failure' (0x5) with response code 'Authentication failed' (0x01) if the authentication fails and with an error RESPONSE of class 'Permanent failure' (0x5) with response code 'Authorization failed' (0x02) if the authorization fails. Per session proof of ownership, as it is defined in this memo, is not possible anymore when using this aggregated mode. However, the downstream NF can use the relationship between the information of the received request message and the GIST messaging association where the request has been received. The downstream NF MUST only accept this aggregated request message through already established GIST messaging associations with the NI. The downstream NF MUST NOT propagate this aggregated request message but it MAY generate and forward per session request messages.

3.8.5 Reporting Asynchronous Events

NATFW NSLP forwarders and NATFW NSLP responders must have the ability

to report asynchronous events to other NATFW NSLP nodes, especially to allow reporting back to the NATFW NSLP initiator. Such asynchronous events may be premature session termination, changes in local policies, route change or any other reason that indicates change of the NATFW NSLP session state. Currently, asynchronous session termination, re-authorization required and route change

detected (see Section 3.10) are the only events that are defined, but other events may be defined in later revisions of this memo.

NFs and NRs may generate NOTIFY messages upon asynchronous events, with a NATFW_INFO object indicating the reason for event. These reasons can be carried in the NATFW_INFO object (class MUST be set to 'Informational' (0x1)) within the NOTIFY message. This list shows the response codes and the associated actions to take at NFs and the NI:

- 'Route change: possible route change on the downstream path' (0x01): Follow instructions in Section 3.10.

- 'Re-authentication required' (0x02): The NI should re-send the authentication.

- 'NATFW node is going down soon' (0x03): The NI and other NFs should be prepared for a service interruption at any time.

NOTIFY messages are sent hop-by-hop upstream towards NI until they reach NI.

The initial processing when receiving a NOTIFY message is the same for all NATFW nodes: NATFW nodes MUST only accept NOTIFY messages through already established NTLP messaging associations. The further processing is different for each NATFW NSLP node type and depends on the events notified:

- NSLP initiator: NIs analyze the notified event and behave appropriately based on the event type. NIs MUST NOT generate NOTIFY messages.

- NSLP forwarder: NFs analyze the notified event and behave based on the above description per response code. NFs SHOULD generate NOTIFY messages upon asynchronous events and forward them upstream towards the NI.

- NSLP responder: NRs SHOULD generate NOTIFY messages upon asynchronous events including a response code based on the reported event. The NF should never receive NOTIFY messages and MUST silently discard it.

NATFW NSLP forwarders, keeping multiple signaling sessions at the same time, can experience problems when shutting down service suddenly. This sudden shutdown can be result of node local failure, for instance, due to a hardware failure. This NF generates NOTIFY messages for each of the signaling sessions and tries to send them upstream. Due to the number of NOTIFY messages to be sent, the shutdown of the node may be unnecessarily prolonged, since not all messages can be sent at the same time. This case can be described as a NOTIFY storm, if a multitude of signaling sessions is involved.

To avoid the need of generating per signaling session NOTIFY messages in such a scenario described or similar cases, NFs SHOULD follow this procedure: The NF uses the NOTIFY message with the session ID in the NTLP set to zero, with the MRI completely wildcarded, using the 'explicit routing' as described in Sections 5.2.1 and 7.1.4. in [1]. The upstream NF receiving this type of NOTIFY immediately regards all signaling sessions from that peer matching the MRI as void. This message will typically result in multiple NOTIFY messages at the upstream NF, i.e., the NF can generate per terminated session a NOTIFY message. However, a NF MAY aggregate again the NOTIFY messages as described here.

3.8.6 Tracing Signaling Sessions

The NATFW NSLP provides a diagnosis capability to session owners (the NI or NI+). Session owners are able to trace the NSIS nodes being involved in a particular signaling session. The TRACE request message is used to trace the involved NSIS nodes along the signaling session and to return their identifiers.

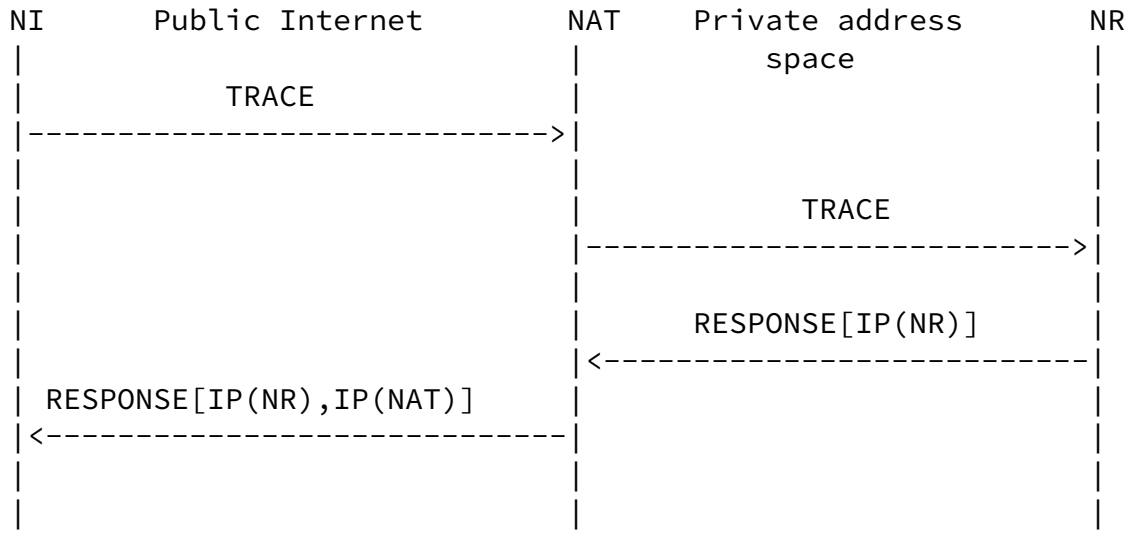


Figure 18: Example for tracing the signaling session path

The processing when receiving a TRACE message is the different for each type of NATFW node:

- NSLP initiator: NI generates TRACE request messages. The NI should never receive request messages and MUST silently discard it.

- NSLP forwarder: NFs solely forward the message if their local policies permits tracing. A NF MUST generate an error RESPONSE of class 'Permanent failure' (0x6) with response code 'Tracing is not allowed' (0x08) if the local policies do not allow tracing.

- NSLP responder: NRs receiving a TRACE request message terminate the forwarding and reply with a successful RESPONSE message. The NATFW_TRACE object MAY be filled by the NR with its IP address.

Processing of a RESPONSE message to a TRACE request message is different for every NSIS node type:

- NSLP initiator: The NI terminates the forwarding and checks the response message for further local processing.

- NSLP forwarder: NFs MAY include their identifier in the NATFW_TRACE object and increment the 'hop count' field by one. This memo defines IPv4 and IPv6 IP addresses as possible de identifier. NFs MUST forward this type of RESPONSE.

- NSLP responder: A NR should never see such a RESPONSE message and it MUST silently discard it.

3.8.7 Proxy Mode of Operation

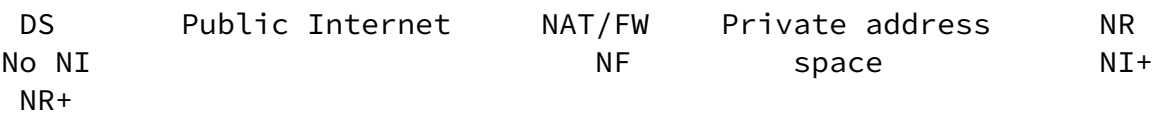
Some migration scenarios need specialized support to cope with cases where NSIS is only deployed in same areas of the Internet. End-to-end signaling is going to fail without NSIS support at or near both data sender and data receiver terminals. A proxy mode of operation is needed. This proxy mode of operation must terminate the NATFW NSLP signaling as topologically close to the terminal for which it is proxying and proxy all request and response messages. This NATFW NSLP node doing the proxying of the signaling messages becomes either

the NI or the NR for the particular signaling session, depending on whether it is the DS or DR that does not support NSIS. Typically, the edge-NAT or the edge-firewall would be used to proxy NATFW NSLP messages.

This proxy mode operation does not require any new request message type, but relies on extended CREATE and REA message types. They are called respectively CREATE-PROXY and REA-PROXY and are distinguished by setting the P flag in the NSLP header is set to P=1. This flag instructs edge-NATs and edge-firewalls receiving them to operate in proxy mode for the session in question. The semantics of the CREATE and REA message types are not changed and the behavior of the various node types is as defined in Section 3.8.1 and Section 3.8.2, except for the proxying node. The following paragraphs describe the proxy mode operation for data receivers behind middleboxes and data senders behind middleboxes.

3.8.7.1 Proxying for a Data Sender

The NATFW NSLP gives the NR the ability to install state on the upstream path towards the data sender for downstream data packets, even when only the receiving side is running NSIS (as shown in Figure 19). The goal of the method described is to trigger the edge-NAT/edge-firewall to generate a CREATE message on behalf of the data receiver. In this case, a NR can signal towards the network border as it is performed in the standard REA message handling scenario as in Section 3.8.2. The message is forwarded until the edge-NAT/ edge-firewall is reached. A public IP address and port number is reserved at an edge-NAT/edge-firewall. As shown in Figure 19, unlike the standard REA message handling case, the edge-NAT/edge-firewall is triggered to send a CREATE message on a new reverse path which traverse several firewalls or NATs. The new reverse path for CREATE is necessary to handle routing asymmetries between the edge-NAT/ edge-firewall and DR. It must be stressed that the semantics of the CREATE and REA request messages is not changed, i.e., each is processed as described earlier.



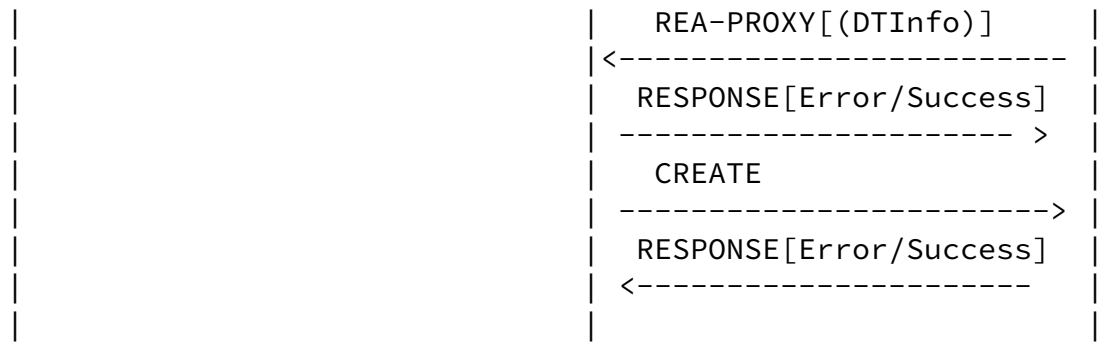


Figure 19: REA Triggering Sending of CREATE Message

A NATFW_NONCE object, carried in the REA and CREATE message, is used to build the relationship between received CREATEs at the message initiator. An NI+ uses the presence of the NATFW_NONCE object to correlate it to the particular REA-PROXY request. The absence of an NONCE object indicates a CREATE initiated by the DS and not by the edge-NAT. Therefore, these processing rules of REA-PROXY messages are added to the regular REA processing:

- NSLP initiator (NI+): The NI+ MUST choose a random value and place it in the NATFW_NONCE object.

- NSLP forwarder being either edge-NAT or edge-firewall: When the NF accepts a REA_PROXY message, it generates a successful RESPONSE message as if it were the NR and additionally, it generates a CREATE message as defined in Section 3.8.1 and includes a NATFW_NONCE object having the same value as of the received NATFW_NONCE object. The NF MUST not generate a CREATE-PROXY message. The NF MUST refresh the CREATE message session only if a REA-PROXY refresh message has been received first.

The scenario described in this section challenges the data receiver because it must make a correct assumption about the data sender's ability to use NSIS NATFW NSLP signaling. It is possible for the DR to make the wrong assumption in two different ways:

a) the DS is NSIS unaware but the DR assumes the DS to be NSIS aware and

b) the DS is NSIS aware but the DR assumes the DS to be NSIS

unaware.

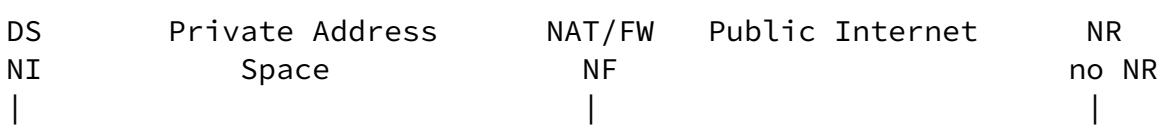
Case a) will result in middleboxes blocking the data traffic, since DS will never send the expected CREATE message. Case b) will result in the DR successfully requesting proxy mode support by the edge-NAT or edge-firewall. The edge-NAT/edge-firewall will send CREATE messages and DS will send CREATE messages as well. Both CREATE messages are handled as separated sessions and therefore the common rules per session apply; the NATFW_NONCE object is used to differentiate CREATE messages generated by the edge-NAT/edge-firewall from NI initiated CREATE messages. It is the NR's responsibility to decide whether to teardown the REA-PROXY sessions in the case where the data sender's side is NSIS aware, but was incorrectly assumed not to be so by the DR. It is RECOMMENDED that a DR behind NATs uses the proxy mode of operation by default, unless the DR knows that the DS is NSIS aware. The DR MAY cache information about data senders which it has found to be NSIS aware in past sessions.

There is a possible race condition between the RESPONSE message to

the REA-PROXY and the CREATE message generated by the edge-NAT. The CREATE message can arrive earlier than the RESPONSE message. An NI+ MUST accept CREATE messages generated by the edge-NAT even if the RESPONSE message to the REA-PROXY request was not received.

3.8.7.2 Proxying for a Data Receiver

As with data receivers behind middleboxes, data senders behind middleboxes can require proxy mode support. The issue here is that there is no NSIS support at the data receiver's side and, by default, there will be no response to CREATE request messages. This scenario requires the last NSIS NATFW NSLP aware node to terminate the forwarding and to proxy the response to the CREATE message, meaning that this node is generating RESPONSE messages. This last node may be an edge-NAT/edge-firewall, or any other NATFW NSLP peer, that detects that there is no NR available (probably as a result of GIST timeouts but there may be other triggers).



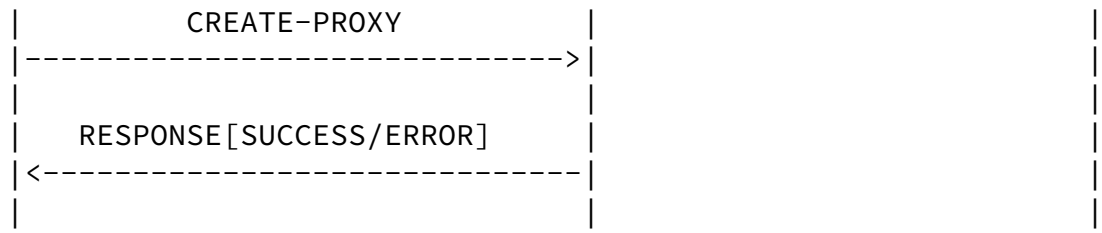


Figure 20: Proxy Mode CREATE Message Flow

The processing of CREATE-PROXY messages and RESPONSE messages is similar to Section 3.8.1, except that forwarding is stopped at the edge-NAT/edge-firewall. The edge-NAT/edge-firewall responds back to NI according the situation (error/success) and will be the NR for future NATFW NSLP communication.

The NI can choose the proxy mode of operation although the DR is NSIS aware. The CREATE-PROXY mode would not configure all NATs and firewalls along the data path, since it is terminated at the edge-device. Any device beyond this point will never received any NATFW NSLP signaling for this flow.

3.9 De-Multiplexing at NATs

Section 3.8.2 describes how NSIS nodes behind NATs can obtain a public reachable IP address and port number at a NAT and and how the

resulting mapping rule can be activated by using CREATE messages (see Section 3.8.1). The information about the public IP address/port number can be transmitted via an application level signaling protocol and/or third party to the communication partner that would like to send data toward the host behind the NAT. However, NSIS signaling flows are sent towards the address of the NAT at which this particular IP address and port number is allocated and not directly to the allocated IP address and port number. The NATFW NSLP forwarder at this NAT needs to know how the incoming NSLP requests are related to reserved addresses, meaning how to de-multiplex incoming NSIS requests.

The de-multiplexing method uses information stored at the local NATFW NSLP node and the of the policy rule. The policy rule uses the LE-MRM MRI source-address (see [1]) as the flow destination IP address

and the network-layer-version as IP version. The external IP address at the NAT is stored as the external flow destination IP address. All other parameters of the policy rule other than the flow destination IP address are wildcarded if no NATFW_DTINFO_IPv4 object is included in the REA request message. The LE-MRM MRI destination-address MUST NOT be used in the policy rule, since it is solely a signaling destination address.

If the NATFW_DTINFO_IPv4 object is included in the REA request message, the policy rule is filled with further information. The 'dst port number' field of the NATFW_DTINFO_IPv4 is stored as the flow destination port number. The 'protocol' field is stored as the flow protocol. The 'src port number' field is stored as the flow source port number. The 'data sender's IPv4 address' is stored as the flow source IP address. Note that some of these field can contain wildcards.

When receiving a CREATE message at the NATFW NSLP it uses the flow information stored in the MRI to do the matching process. This table shows the parameters to be compared against each others. Note that not all parameters can be present in a MRI at the same time.

Flow parameter (Policy Rule)	MRI parameter (CREATE message)
IP version	network-layer-version
Protocol	IP-protocol
source IP address (w)	source-address (w)

external IP address	destination-address
destination IP address (n/u)	N/A
source port number (w)	L4-source-port (w)
external port number (w)	L4-destination-port (w)
destination port number (n/u)	N/A
IPsec SPI	ipsec-SPI

Table entries marked with (w) can be wildcarded and entries marked with (n/u) are not used for the matching.

Table 1

3.10 Reacting to Route Changes

The NATFW NSLP needs to react to route changes in the data path. This assumes the capability to detect route changes, to perform NAT and firewall configuration on the new path and possibly to tear down session state on the old path. The detection of route changes is described in Section 7 of [1] and the NATFW NSLP relies on notifications about route changes by the NTLP. This notification will be conveyed by the API between NTLP and NSLP, which is out of scope of this memo.

A NATFW NSLP node other than the NI or NI+ detecting a route change, by means described in the NTLP specification or others, generates a NOTIFY message indicating this change and sends this upstream towards NI. Intermediate NFs on the way to the NI can use this information to decide later if their session can be deleted locally, if they do not receive an update within a certain time period, as described in Section 3.2.3. It is important to consider the transport limitations of NOTIFY messages as mandated in Section 3.8.5.

The NI receiving this NOTIFY message MAY generate a new request

CREATE or REA message and sends it respectively downstream or upstream as for the initial exchange using the same session ID. All the remaining processing and message forwarding, such as NSLP next hop discovery, is subject to regular NSLP processing as described in the particular sections. Normal routing will guide the new request to the correct NFs along the changed route. NFs that were on the original path receiving these new request messages (see also Section 3.11), can use the session ID (session ownership information, see also Section 3.6) to update the existing session, whereas NFs that were not on the original path will create new state for this session. The next section describes how policy rules are updated.

3.11 Updating Policy Rules

NSIS initiators can request an update of the installed/reserved policy rules at any time within a signaling session. Updates to policy rules can be required due to node mobility (NI is moving from one IP address to another), route changes (this can result in a different NAT mapping at a different NAT device), or the wish of the NI to simply change the rule. NIs can update policy rules in existing signaling sessions by sending an appropriate request message (similar to Section 3.4) with modified message routing information (MRI) as compared with that installed previously, but using the existing session ID to identify the intended target of the update. With respect to authorization and authentication, this update request message is treated in exactly the same way as any initial request. Therefore, any node along in the signaling session can reject the update with an error RESPONSE message, as defined in the previous sections.

The request/response message processing and forwarding is executed as defined in the those sections. A NF or the NR receiving an update, simply replaces the installed policy rules installed in the firewall/ NAT. The local procedures on how to update the MRI in the firewall/ NAT is out of scope of this memo.

4. NATFW NSLP Message Components

A NATFW NSLP message consists of a NSLP header and one or more objects following the header. The NSLP header is common for all NSLPs and objects are Type-Length-Value (TLV) encoded using big endian (network ordered) binary data representations. Header and objects are aligned to 32 bit boundaries and object lengths that are not multiples of 32 bits must be padded to the next higher 32 bit multiple.

The whole NSLP message is carried as payload of a NTLP message.

Note that the notation 0x is used to indicate hexadecimal numbers.

4.1 NSLP Header

The NSLP header is common to all NSLPs and is the first part of all NSLP messages. It contains two fields, the NSLP message type and a reserved field. The total length is 32 bits. The layout of the NSLP header is defined by Figure 21.

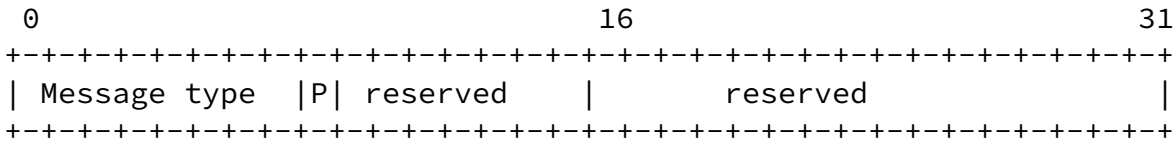


Figure 21: Common NSLP header

The reserved field MUST be set to zero in the NATFW NSLP header before sending and MUST be ignored during processing of the header.

The message types identify requests and responses. Defined messages types are:

- IANA-TBD(1) : CREATE
- IANA-TBD(2) : RESERVE-EXTERNAL-ADDRESS(REA)
- IANA-TBD(3) : TRACE
- IANA-TBD(4) : RESPONSE
- IANA-TBD(5) : NOTIFY

If a message with another type is received, an error RESPONSE of class 'Protocol error' (0x3) with response code 'Illegal message type' (0x01) MUST be generated.

The P flag indicates the usage of proxy mode. If proxy mode is used it MUST be set to 1. Proxy mode usage is only allowed in combination with the message types CREATE and REA, P=1 MUST NOT be set with message types other than CREATE and REA. The P flag MUST be ignored when processing messages with type RESPONSE. An error RESPONSE message of class 'Protocol error' (0x3) and type 'Bad flags value' (0x03) MUST be generated, if the P flag is set in TRACE or NOTIFY messages.

4.2 NSLP Objects

NATFW NSLP objects use a common header format defined by Figure 22. The object header contains two fields, the NSLP object type and the object length. Its total length is 32 bits.

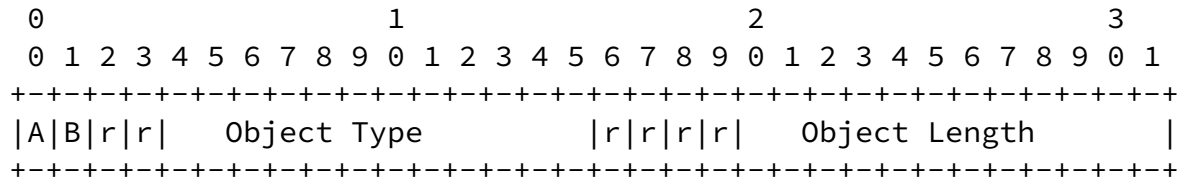


Figure 22: Common NSLP object header

The object length field contains the total length of the object without the object header. The unit is a word, consisting of 4 octets. The particular values of type and length for each NSLP object are listed in the subsequent sections that define the NSLP objects. An error RESPONSE of class 'Protocol error' (0x3) with response code 'Wrong object length' (0x07) MUST be generated if the length given for the object in the object header did not match the length of the object data present. The two leading bits of the NSLP object header are used to signal the desired treatment for objects

whose treatment has not been defined in this memo (see [1], Section A.2.1), i.e., the Object Type has not been defined. NATFW NSLP uses a subset of the categories defined in GIST:

- AB=00 ("Mandatory"): If the object is not understood, the entire message containing it MUST be rejected with an error RESPONSE of class 'Protocol error' (0x3) with response code 'Unknown object present' (0x06).

- AB=01 ("Optional"): If the object is not understood, it should be deleted and then the rest of the message processed as usual.

- AB=10 ("Forward"): If the object is not understood, it should be retained unchanged in any message forwarded as a result of message processing, but not stored locally.

The combination AB=11 MUST NOT be used and an error RESPONSE of class 'Protocol error' (0x3) with response code 'Invalid Flag-Field combination' (0x09) MUST be generated.

The following sections do not repeat the common NSLP object header, they just name the type and the length.

4.2.1 Session Lifetime Object

The session lifetime object carries the requested or granted lifetime of a NATFW NSLP session measured in seconds.

Type: NATFW_LT (IANA-TBD)

Length: 1

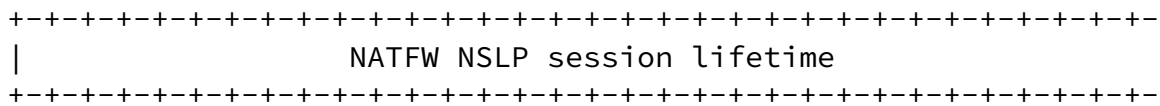


Figure 23: Lifetime object

4.2.2 External Address Object

The external address object can be included in RESPONSE messages (Section 4.3.3) only. It carries the publicly reachable IP address, and if applicable port number, at an edge-NAT.

Type: NATFW_EXT_IP (IANA-TBD)

Length: 2

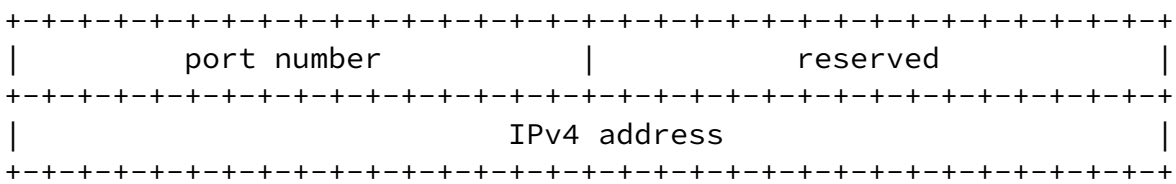


Figure 24: External Address Object for IPv4 addresses

Please note that the field 'port number' MUST be set to 0 if only an IP address has been reserved, for instance, by a traditional NAT. A port number of 0 MUST be ignored in processing this object.

4.2.3 Extended Flow Information Object

In general, flow information is kept in the message routing information (MRI) of the NTLP. Nevertheless, some additional information may be required for NSLP operations. The 'extended flow information' object carries this additional information about the action of the policy rule for firewalls/NATs and contiguous port .

Type: NATFW_EFI (IANA-TBD)

Length: 1

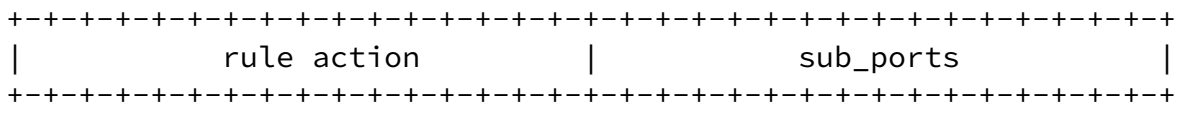


Figure 25: Extended Flow Information

This object has two fields, 'rule action' and 'sub_ports'. The 'rule action' field has these meanings:

- 0x0001: Allow: A policy rule with this action allows data traffic to traverse the middlebox and the NATFW NSLP MUST allow NSLP signaling to be forwarded.

- 0x0002: Deny: A policy rule with this action blocks data traffic from traversing the middlebox and the NATFW NSLP MUST NOT allow NSLP signaling to be forwarded.

If the 'rule action' field contains neither 0x0001 nor 0x0002, an

error RESPONSE of class 'Signaling session error' (0x6) with response code 'Unknown policy rule action' (0x05) MUST be generated.

The 'sub_ports' field contains the number of contiguous transport layer ports to which this rule applies. The default value of this field is 0, i.e., only the port specified in the NTLP's MRM is used for the policy rule. A value of 1 indicates that additionally to the port specified in the NTLP's MRM (port1), a second port (port2) is used. This value of port 2 is calculated as: port2 = port1 + 1. Other values than 0 or 1 MUST NOT be used in this field and an error RESPONSE of class 'Signaling session error' (0x6) with response code 'Requested value in sub_ports field in NATFW_EFI not permitted' (0x08) MUST be generated. Further version of this memo may allow other values for the 'sub_ports' field. This two contiguous port numbered ports, can be used by legacy voice over IP equipment. This legacy equipment assumes that two adjacent port numbers for its RTP/RTCP flows respectively.

4.2.4 Information Code Object

This object carries the response code, which may be indications for either a successful request or failed request depending on the value of the 'response code' field.

Type: NATFW_INFO (IANA-TBD)

Length: 1

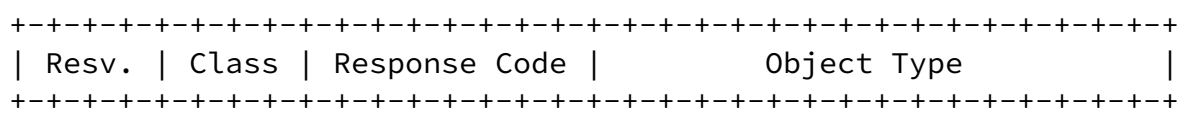


Figure 26: Information Code Object

The field 'resv.' is reserved for future extensions and MUST be set to zero when generating such an object and MUST be ignored when receiving. The 'Object Type' field contains the type of the object causing the error. The value of 'Object Type' is set to 0, if no object is concerned. The 4 bit class field contains the severity class. The following classes are defined:

- 0x1: Informational (NOTIFY only)

- 0x2: Success
- 0x3: Protocol error
- 0x4: Transient failure
- 0x5: Permanent failure
- 0x6: Signaling session failures

Within each severity class a number of responses values are defined

- Informational:
  - 0x01: Route change: possible route change on the downstream path.
  - 0x02: Re-authentication required.
  - 0x03: NATFW node is going down soon.
- Success:
  - 0x01: All successfully processed.
- Protocol error:
  - 0x01: Illegal message type: the type given in the Message Type field of the NSLP header is unknown.
  - 0x02: Wrong message length: the length given for the message in the NSLP header does not match the length of the message data.
  - 0x03: Bad flags value: an undefined flag or combination of flags was set in the NSLP header.
  - 0x04: Mandatory object missing: an object required in a message of this type was missing.
  - 0x05: Illegal object present: an object was present which must not be used in a message of this type.
  - 0x06: Unknown object present: an object of an unknown type was present in the message.
  - 0x07: Wrong object length: the length given for the object in the object header did not match the length of the object data present.

  - 0x08: Unknown object field value: a field in an object had an unknown value.
  - 0x09: Invalid Flag-Field combination: An object contains an invalid combination of flags and/or fields.

    * 0x0A: Duplicate object present.

    * 0x0B: Received REA request message on external side.

  o Transient failure:

    * 0x01: Requested resources temporarily not available.

  o Permanent failure:

    * 0x01: Authentication failed.

    * 0x02: Authorization failed.

    * 0x03: Unable to agree transport security with peer.

    * 0x04: Internal or system error.

    * 0x05: No NAT here.

    * 0x06: No edge-device here.

    * 0x07: Did not reach the NR.

    * 0x08: Tracing is not allowed.

  o Signaling session failures:

    * 0x01: Session terminated asynchronously.

    * 0x02: Requested lifetime is too big.

    * 0x03: No reservation found matching the MRI of the CREATE request.

    * 0x04: Requested policy rule denied due to policy conflict.

    * 0x05: Unknown policy rule action.

    * 0x06: Requested rule action not applicable.

* 0x07: DTINFO object is required.

* 0x08: Requested value in sub_ports field in NATFW_EFI not permitted.

* 0x09: Requested IP protocol not supported.

* 0x0A: Plain IP policy rules not permitted -- need transport layer information.

* 0x0B: ICMP type value not permitted.

* 0x0C: source IP address range is too large.

* 0x0D: destination IP address range is too large.

* 0x0E: source L4-port range is too large.

* 0x0F: destination L4-port range is too large.

4.2.5 Nonce Object

This object carries the nonce value as described in Section 3.8.7.

Type: NATFW_NONCE (IANA-TBD)

Length: 1

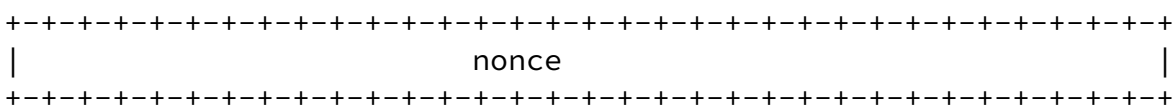


Figure 27: Nonce Object

4.2.6 Message Sequence Number Object

This object carries the MSN value as described in Section 3.5.

Type: NATFW_MSN (IANA-TBD)

Length: 1

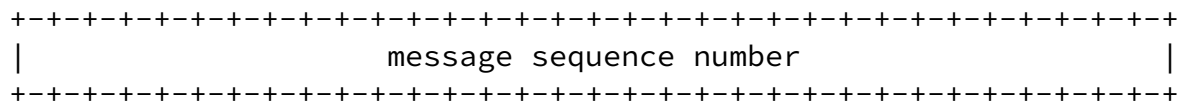


Figure 28: Message Sequence Number Object

4.2.7 Data Terminal Information Object

The 'data terminal information' object carries additional information possibly needed during REA operations. REA messages are transported by the NTLP using the Loose-End message routing method (LE-MRM). The LE-MRM contains only DR's IP address and a signaling destination address (destination address). This destination address is used for message routing only and is not necessarily reflecting the address of the data sender. This object contains information about (if applicable) DR's port number (the destination port number), DS' port number (the source port number), the used transport protocol, the prefix length of the IP address, and DS' IP address.

Type: NATFW_DTINFO_IPv4 (IANA-TBD)

Length: 3

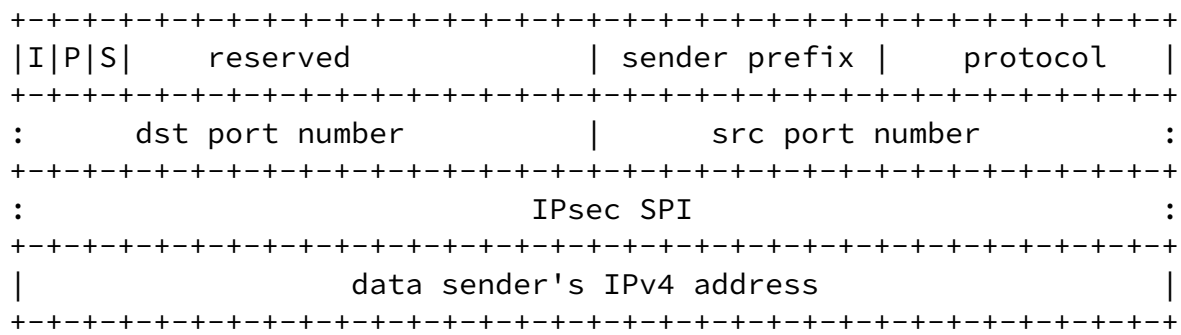


Figure 29: Data Terminal IPv4 Address Object

The flags are:

- I: I=1 means that 'protocol' should be interpreted.

- P: P=1 means that 'dst port number' and 'src port number' are present and should be interpreted.

- S: S=1 means that SPI is present and should be interpreted.

The SPI field is only present if S is set. The port numbers are only present if P is set. The flags P and S MUST NOT be set at the same time. An error RESPONSE of class 'Protocol error' (0x3) with response code 'Invalid Flag-Field combination' (0x09) MUST be generated if they are both set. If either P or S is set, I MUST be set as well and the protocol field MUST carry the particular protocol. An error RESPONSE of class 'Protocol error' (0x3) with response code 'Invalid Flag-Field combination' (0x09) MUST be generated if S or P is set but I is not set.

The fields MUST be interpreted according these rules:

- (data) sender prefix: This parameter indicates the prefix length of the 'data sender's IP address' in bits. For instance, a full IPv4 address requires 'dest prefix' to be set to 32. A value of 0 indicates an IP address wildcard.

- protocol: The IPv4 protocol field. This field MUST be interpreted if I=1, otherwise it MUST be set to 0 and MUST be ignored.

- dst port number: A value of 0 indicates a port wildcard, i.e., the destination port number is not known. Any other value indicates the destination port number.

- src port number: A value of 0 indicates a port wildcard, i.e., the source port number is not known. Any other value indicates the source port number.

- data sender's IPv4 address: The source IP address of the data sender. This field MUST be set to zero if no IP address is provided, i.e., a complete wildcard is desired (see dest prefix field above).

4.2.8 Trace Object

The NATFW_TRACE object may contain zero or more identifiers of visited NATFW NSLP peers. However, it is only possible to store a single type of identifier, either IPv4 or IPv6 addresses.

Type: NATFW_TRACE (IANA-TBD)

Length: Variable

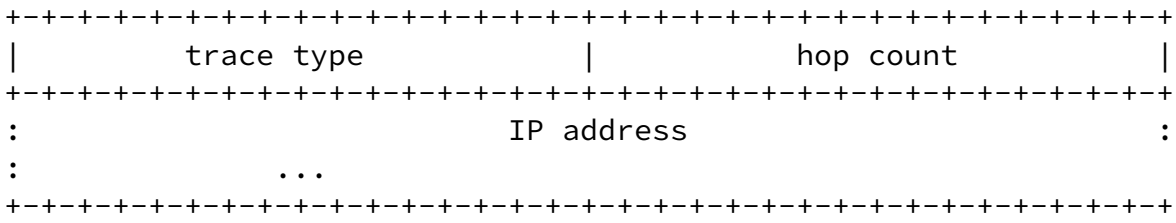


Figure 30: Trace object

The NATFW_TRACE object may contain zero or more identifiers. The type of identifier is given by the value of 'trace type' field. This memo is defining the values for the 'trace type' field: 0x01 for IPv4 addresses and 0x02 for IPv6 addresses. Other trace types MUST generate an error RESPONSE of class 'Protocol error' (0x3) with response code 'Unknown object field value' (0x08). The 'hop count' field counts the total number of visited NATFW NSLP nodes that are willing to include their identifier in this object. Each such node appends its identifier at the end of the object.

4.2.9 NI Credential Object

This object is a container intended to carry credentials provided by the NI.

Type: NATFW_CREDENTIAL (IANA-TBD)

Length: Variable

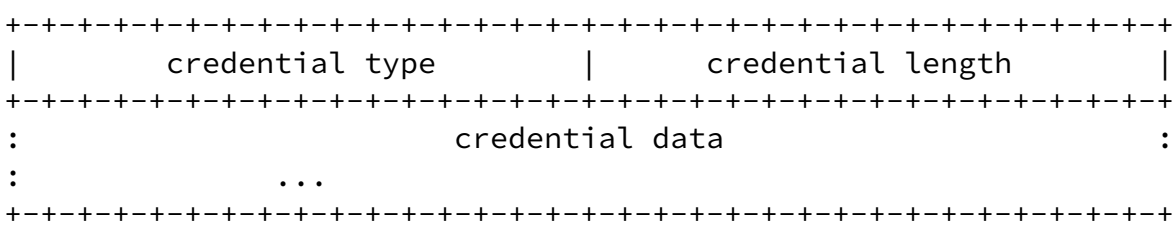


Figure 31: Credential Object

The field 'credential type' field contains one of these values:

o 0x0002: Token

Other trace types MUST generate an error RESPONSE of class 'Protocol

error' (0x3) with response code 'Unknown object field value' (0x08).

The field 'credential length' counts the total number of bytes of the included 'credential data'. Note that the total number of bytes contained in the NATFW_CREDENTIAL object may not end on a 32 bit word boundary. In this case a padding must be included at the end of the object right after the 'credential data' field. The padding must fill the NATFW_CREDENTIAL object to next 32 bit word boundary.

4.2.10 ICMP Types Object

The 'ICMP types' object contains additional information needed to configure a NAT of firewall with rules to control ICMP traffic. The object contains a number of values of the ICMP Type field for which a filter action should be set up:

Type: NATFW_ICMP_TYPES (IANA-TBD)

Length: Variable = ((Number of Types carried + 1) + 3) DIV 4

Where DIV is an integer division.

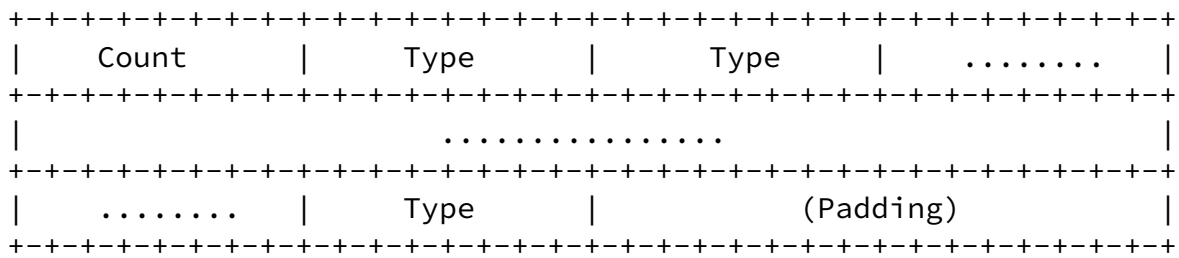


Figure 32: ICMP Types Object

The fields MUST be interpreted according these rules:

count: 8 bit integer specifying the number of 'Type' entries in the object.

type: 8 bit field specifying an ICMP Type value to which this rule applies.

padding: Sufficient 0 bits to pad out the last word so that the total size of object is an even multiple of words. Ignored on reception.

4.3 Message Formats

This section defines the content of each NATFW NSLP message type. The message types are defined in Section 4.1. First, the request messages are defined with their respective objects to be included in the message. Second, the response messages are defined with their respective objects to be included.

Basically, each message is constructed of NSLP header and one or more NSLP objects. The order of objects is not defined, meaning that objects may occur in any sequence. Objects are marked either with mandatory [M] or optional [O]. Where [M] implies that this particular object MUST be included within the message and where [O] implies that this particular object is OPTIONAL within the message.

Objects defined in this memo carry always the flag combination AB=00 in the NSLP object header. An error RESPONSE message of class 'Protocol error' (0x3) with response code 'Mandatory object missing' (0x02) MUST be generated if a mandatory declared object is missing. An error RESPONSE message of class 'Protocol error' (0x3) with response code 'Illegal object present' (0x05) MUST be generated if an object was present which must not be used in a message of this type. An error RESPONSE message of class 'Protocol error' (0x3) with response code 'Duplicate object present' (0x0A) MUST be generated if an object appears more than once in a message.

Each section elaborates the required settings and parameters to be set by the NSLP for the NTLP, for instance, how the message routing information is set.

4.3.1 CREATE

The CREATE request message is used to create NSLP sessions and to create policy rules. Furthermore, CREATE messages are used to refresh sessions and to delete them.

The CREATE request message carries these objects:

- Lifetime object [M]
- Extended flow information object [M]
- Message sequence number object [M]
- Credential object [O]
- Nonce object [M] if P flag set to 1 in the NSLP header, otherwise [O]

- ICMP Types Object [O]

The message routing information in the NTLP MUST be set to DS as source address and DR as destination address. All other parameters MUST be set according the required policy rule. CREATE messages MUST be transported by using the path-coupled MRM with direction set to downstream.

4.3.2 RESERVE-EXTERNAL-ADDRESS (REA)

The RESERVE-EXTERNAL-ADDRESS (REA) request message is used to a) reserve an external IP address/port at NATs, b) to notify firewalls about NSIS capable DRs, or c) to block incoming data traffic at upstream firewalls.

The REA request message carries these objects:

- Lifetime object [M]
- Message sequence number object [M]
- Extended flow information object [M]
- Credential object [O]
- Data terminal information object [O]
- Nonce object [M if P flag set to 1 in the NSLP header, otherwise [O]
- ICMP Types Object [O]

The selected message routing method of the REA request message depends on a number of considerations. Section 3.8.2 describes it exhaustively how to select the correct method. REA request messages can be transported via the path-coupled message routing method (PC-MRM) or via the loose-end message routing method (LE-MRM). In the case of PC-MRM, the source-address is set to DS' address and the destination address is set to DR's address, the direction is set to upstream. In the case of LE-MRM, the destination-address is set to DR's address or to the signaling destination address. The source-address is set to DS's address.

4.3.3 RESPONSE

RESPONSE messages are responses to CREATE and REA messages.

The RESPONSE message for the class 'Success' (0x2) carries these

objects:

- Lifetime object [M]
- Message sequence number object [M]
- Information code object [M]
- External address object [O]
- Trace object [O]

The RESPONSE message for other classes than 'Success' (0x2) carries these objects:

- Message sequence number object [M]
- Information code object [M]

This message is routed upstream hop-by-hop, using existing NTLP messaging associations.

4.3.4 NOTIFY

The NOTIFY messages is used to report asynchronous events happening along the signaled path to other NATFW NSLP nodes.

The NOTIFY message carries this object:

- Information code object [M].

The NOTIFY message is forwarded upstream hop-by-hop using the existing upstream node messaging association entry within the node's Message Routing State table.

4.3.5 TRACE

The TRACE request message is used to trace the involved NATFW NSLP nodes along a signal session.

The TRACE request message carries these objects:

- Message sequence number object [M]
- Trace object [M]

TRACE request messages are sent path-coupled (PC-MRM).

5. Security Considerations

Security is of major concern particularly in case of firewall traversal. This section provides security considerations for the NAT/firewall traversal and is organized as follows.

In Section 5.1 we describe the participating entities relate to each other from a security point of view. This subsection also motivates a particular authorization model.

Security threats that focus on NSIS in general are described in [8] and they are applicable to this document as well. Within Section 5.5 we extend this threat investigation by considering NATFW NSLP specific threats in detail. Based on the investigated security threats we derive security requirements.

Finally, we illustrate how the security requirements that were created based on the security threats can be fulfilled by specific security mechanisms. These aspects will be elaborated in Section 5.14.

5.1 Authorization Framework

The NATFW NSLP is a protocol which may involve a number of NSIS nodes and is, as such, not a two-party protocol. Figure 1 and Figure 2 of [8] already depict the possible set of communication patterns. In this section we will re-evaluate these communication patters with respect to the NATFW NSLP protocol interaction.

The security solutions for providing authorization have a direct impact on the treatment of different NSLPs. As it can be seen from the QoS NSLP [6] and the corresponding Diameter QoS work [23] accounting and charging seems to play an important role for QoS reservations, whereas monetary aspects might only indirectly effect authorization decisions for NAT and firewall signaling. Hence, there are differences in the semantic of authorization handling between QoS and NATFW signaling. A NATFW aware node will most likely want to authorize the entity (e.g., user or machine) requesting the establishment of pinholes or NAT bindings. The outcome of the authorization decision is either allowed or disallowed whereas a QoS authorization decision might indicate that a different set of QoS parameters is authorization (see [23] as an example).

5.2 Peer-to-Peer Relationship

Starting with the simplest scenario, it is assumed that neighboring nodes are able to authenticate each other and to establish keying material to protect the signaling message communication. An addition

to authentication the nodes will have to authorize each other. We use the term 'Security Context' as a placeholder for referring to the entire security procedure, the necessary infrastructure that needs to be in place in order for this to work (e.g., key management) and the established security related state. The required long-term key (symmetric or asymmetric keys) used for authentication are either made available using an out-of-band mechanism between the two NSIS NATFW nodes or they are dynamically established using mechanisms not further specified in this document. Note that the deployment environment will most likely have an impact on the choice of credentials being used. The choice of these credentials used is also outside the scope of this document.

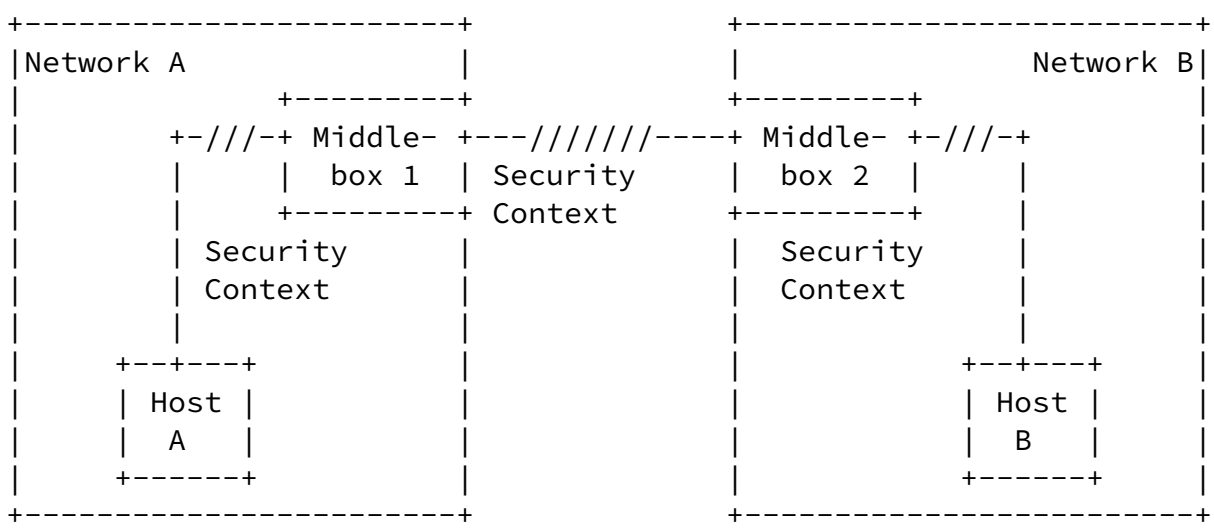


Figure 33: Peer-to-Peer Relationship

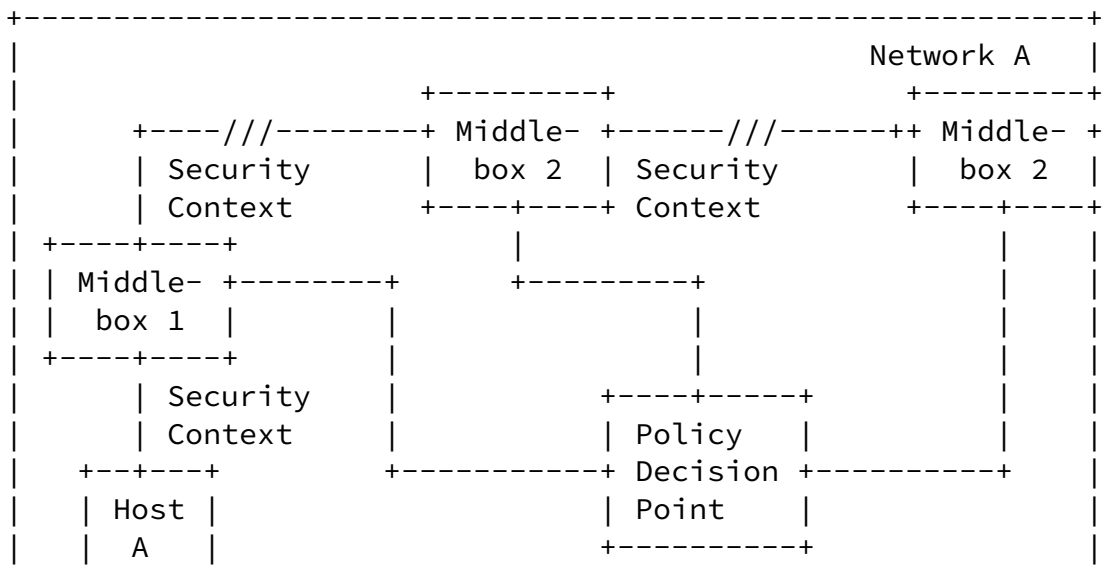
Figure 33 shows a possible relationship between participating NSIS aware nodes. Host A might be, for example, a host in an enterprise network that has keying material established (e.g., a shared secret) with the company's firewall (Middlebox 1). The network administrator of Network A (company network) has created access control lists for

Host A (or whatever identifiers a particular company wants to use). Exactly the same procedure might also be used between Host B and Middlebox 2 in Network B. For the communication between Middlebox 1 and Middlebox 2 a security context is also assumed in order to allow authentication, authorization and signaling message protection to be successful.

5.3 Intra-Domain Relationship

In larger corporations, for example, a middlebox is used to protect individual departments. In many cases, the entire enterprise is controlled by a single (or a small number of) security department,

which gives instructions to the department administrators. In such a scenario, a the previously discussed peer-to-peer relationship might be prevalent. Sometimes it might be necessary to preserve authentication and authorization information within the network. As a possible solution, a centralized approach could be used, whereby an interaction between the individual middleboxes and a central entity (for example a policy decision point - PDP) takes place. As an alternative, individual middleboxes exchange the authorization decision with another middlebox within the same trust domain. Individual middleboxes within an administrative domain may exploit their relationship instead of requesting authentication and authorization of the signaling initiator again and again. Figure 34 illustrates a network structure which uses a centralized entity.



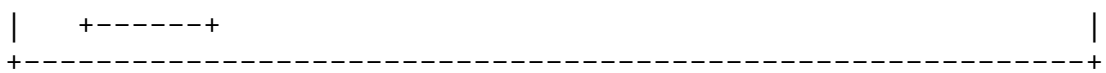


Figure 34: Intra-domain Relationship

The interaction between individual middleboxes and a policy decision point (or AAA server) is outside the scope of this document.

5.4 End-to-Middle Relationship

The peer-to-peer relationship between neighboring NSIS NATFW NSLP nodes might not always be sufficient. Network B might require additional authorization of the signaling message initiator (in addition to the authorization of the neighboring node). If authentication and authorization information is not attached to the initial signaling message then the signaling message arriving at Middlebox 2 would result in an error message being created, which indicates the additional authorization requirement. In many cases the signaling message initiator might already aware of the

additionally required authorization before the signaling message exchange is executed.

Figure 35 shows this scenario.

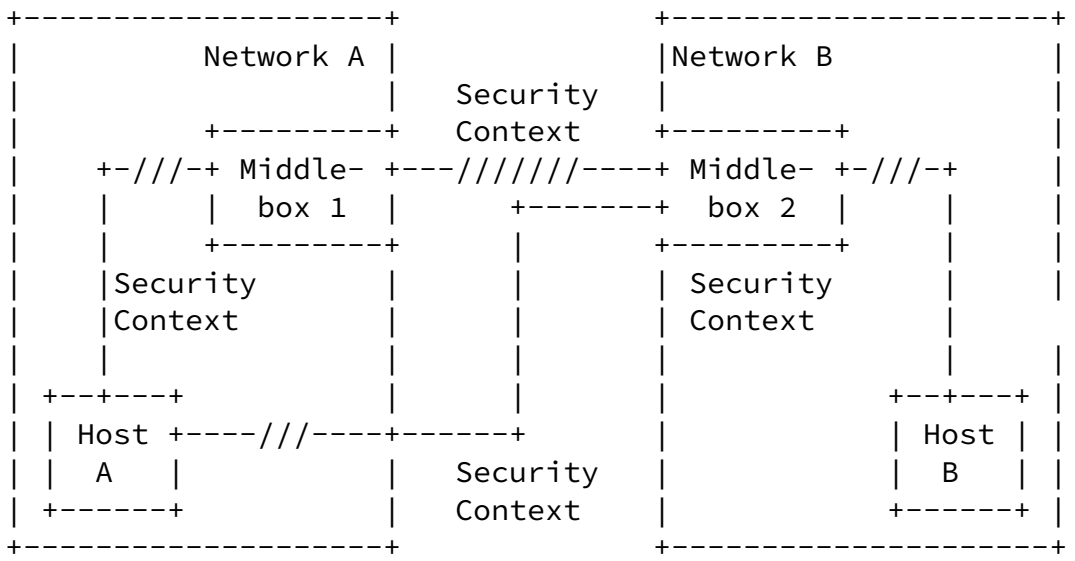
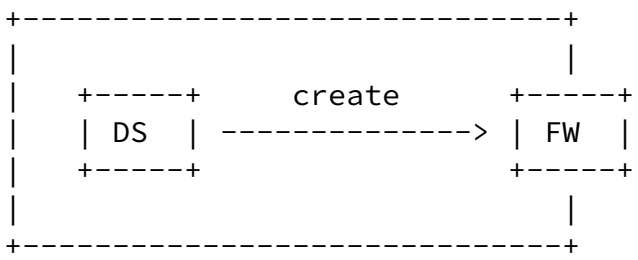


Figure 35: End-to-Middle Relationship

5.5 Security Threats and Requirements

This section describes NATFW specific security threats and requirements.

5.5.1 Data Sender (DS) behind a firewall



DS sends a CREATE message to request the traversal of a data flow.

The following attacks are possible:

- DS could open a firewall pinhole with a source address different from its own host.
- DS could open firewall pinholes for incoming data flows that are not supposed to enter the network.

- DS could request installation of any policy rules and allow all traffic go through.

SECURITY REQUIREMENT: The middlebox MUST authenticate and authorize the neighboring NAT/FW NSLP node requesting an action. Authentication and authorization of the initiator SHOULD be provided to NATs and firewalls along the path.

5.5.2 Data Sender (DS) behind a NAT

The case 'DS behind a NAT' is analogous to the case 'DS behind a firewall'.

Figure 37 illustrates such a scenario:

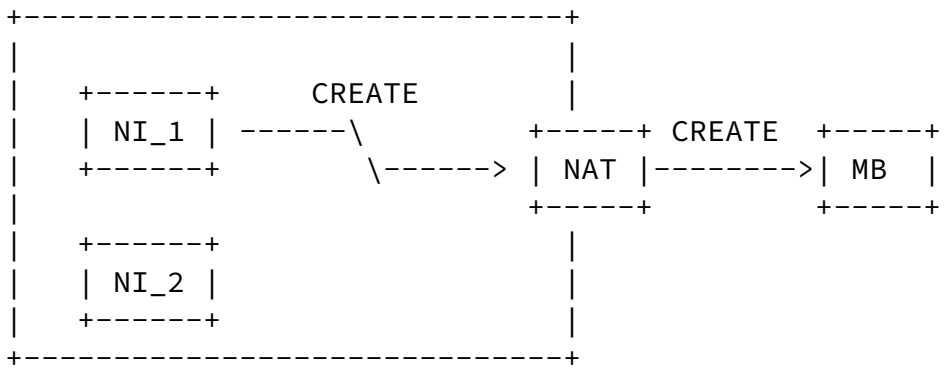


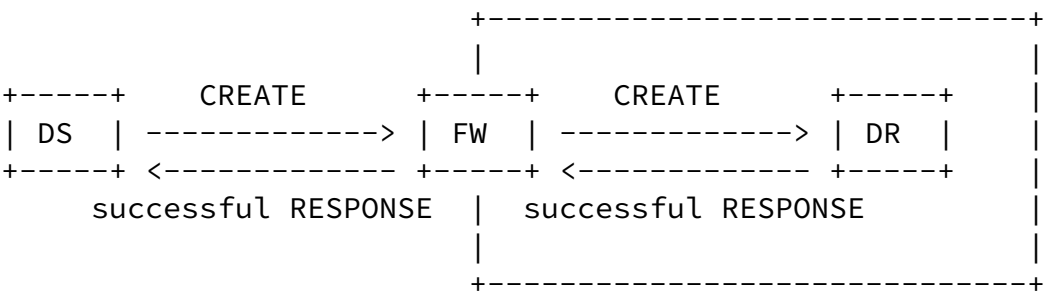
Figure 37: Several NIs behind a NAT

In this case the middlebox MB does not know who is the NSIS Initiator since both NI_1 and NI_2 are behind a NAT (which is also NSIS aware). Authentication needs to be provided by other means such as the NSLP or the application layer.

SECURITY REQUIREMENT: The middlebox MUST authenticate and ensure that the neighboring NAT/FW NSLP node is authorized to request an action. Authentication and authorization of the initiator (which is the DR in this scenario) to the non-neighboring middleboxes SHOULD be provided.

5.5.3 Data Receiver (DR) behind a firewall

In this case a CREATE message comes from an entity DS outside the network towards the DR inside the network.



Since policy rules at middleboxes must only be installed after receiving a successful response it is necessary that the middlebox waits until the Data Receiver DR confirms the request of the Data Sender DS with a successful RESPONSE message. This is, however, only necessary

- if the action requested with the CREATE message cannot be authorized and

- if the middlebox is still forwarding the signaling message towards the end host (without state creation/deletion/modification).

This confirmation implies that the data receiver is expecting the data flow.

At this point we differentiate two cases:

1. DR knows the IP address of the DS (for instance because of some previous application layer signaling) and is expecting the data flow.

2. DR might be expecting the data flow (for instance because of some previous application layer signaling) but does not know the IP address of the Data Sender DS.

For the second case, Figure 39 illustrates a possible attack: an adversary Mallory M could be sniffing the application layer signaling and thus knows the address and port number where DR is expecting the data flow. Thus it could pretend to be DS and send a CREATE message towards DR with the data flow description (M -> DR). Since DR does not know the IP address of DS, it is not able to recognize that the request is coming from the "wrong guy". It will send a success RESPONSE message back and the middlebox will install policy rules that will allow Mallory M to inject its data into the network.

Application Layer signaling

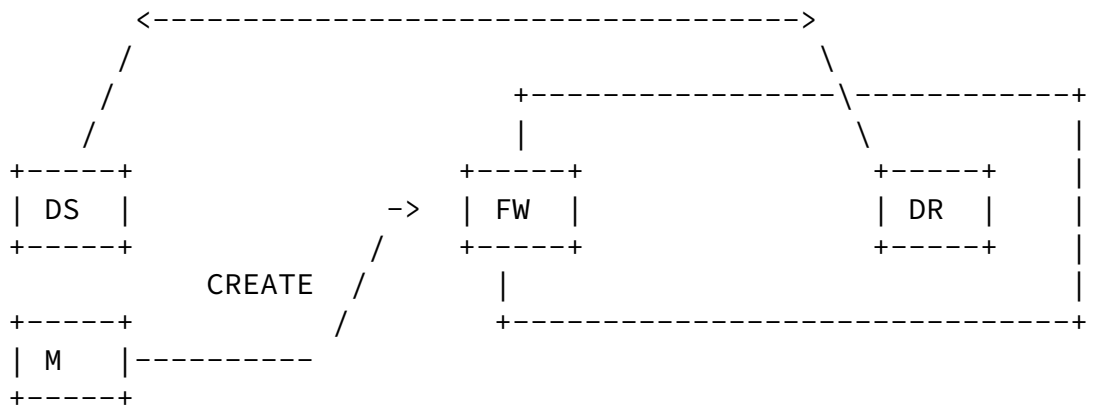


Figure 39: DR behind a firewall with an adversary

Network administrators will probably not rely on a DR to check the IP address of the DS. Thus we have to assume the worst case with an attack such as in Figure 39. Many operators might not allow NSIS signaling message to traverse the firewall in Figure 39 without having the DR to interact with the FW first.

SECURITY REQUIREMENT: No requirements are created by this scenario.

5.5.4 Data Receiver (DR) behind a NAT

When a data receiver DR behind a NAT sends a RESERVE-EXTERNAL-ADDRESS (REA) message to get a public reachable address that can be used as a contact address by an arbitrary data sender if the DR was unable to restrict the future data sender. The NAT reserves an external address and port number and sends them back to DR. The NAT adds an address mapping entry in its reservation list which links the public and private addresses as follows:

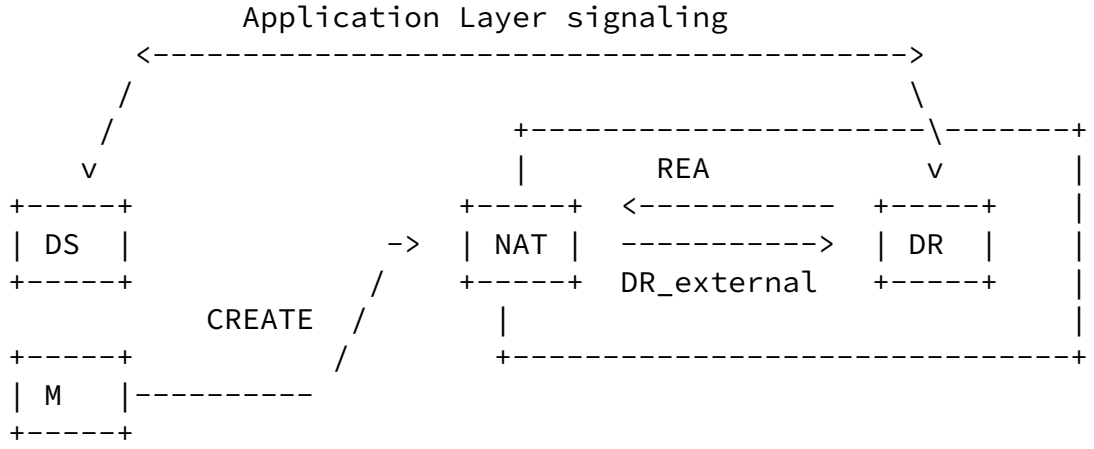
(DR_ext <=> DR_int) (*).

The NAT sends a RESPONSE message with the external address' object back to the DR with the address DR_ext. DR informs DS about the public address that it has recently received, for instance, by means of application layer signaling.

When a data sender sends a CREATE message towards DR_ext then the message will be forwarded to the DR. The data sender might want to update the NAT binding stored at the edge-NAT to make it more restrictive.

We assume that the adversary Mallory M obtains the contact address (i.e., external address and port) allocated at the NAT possibly by

eavesdropping on the application layer signaling and sends a CREATE message. As a consequence Mallory would be able to communicate with DR (if M is authorized by the edge-NAT and if the DR accepts CREATE and returns a RESPONSE.



SECURITY REQUIREMENT: The DR MUST be able to specify which data sender are allowed to traverse the NAT in order to be forwarded to DRs address.

5.5.5 NSLP Message Injection

Malicious hosts, located either off-path or on-path, could inject arbitrary NATFW NSLP messages into the signaling path. These problems apply when no proper authorization and authentication scheme is available.

By injecting a bogus CREATE message with lifetime set to zero, a malicious host could try to teardown NATFW NSLP session state partially or completely on a data path, causing a service interruption.

By injecting a bogus responses or NOTIFY message, for instance, timeout, a malicious host could try to teardown NATFW NSLP session state as well. This could affect the data path partially or totally, causing a service interruption.

SECURITY REQUIREMENT: Messages, such as NOTIFY, can be misused by malicious hosts, and therefore MUST be authorized by the respective NATFW NLSP entities.

5.6 Denial-of-Service Attacks

In this section we describe several ways how an adversary could launch a Denial of service (DoS) attack on networks running NSIS for middlebox configuration to exhaust their resources.

5.6.1 Flooding with CREATE messages from outside

5.6.1.1 Attacks due to NSLP state

A CREATE message requests the NSLP to store state information such as a NAT binding or a policy rule.

The policy rules requested in the CREATE message will be installed at the arrival of a confirmation from the Data Receiver with a success RESPONSE message. A successful RESPONSE message includes the session ID. So the NSLP looks up the NSIS session and installs the requested policy rules.

An adversary could launch a DoS attack with an arbitrary number of CREATE messages. For each of these messages the middlebox needs to store state information such as the policy rules to be loaded, i.e., the middlebox could run out of memory. This kind of attack is also mentioned in [8] Section 4.8.

SECURITY REQUIREMENT: A NAT/FW NSLP node MUST authorize the establishment of state information.

5.6.1.2 Attacks due to authentication complexity

This kind of attack is possible if authentication is based on mechanisms that require computing power, for example, digital signatures.

For a more detailed treatment of this kind of attack, the reader is encouraged to see [8].

SECURITY REQUIREMENT: A NAT/FW NSLP node MUST NOT introduce new

denial of service attacks based on authentication or key exchange mechanisms.

5.6.1.3 Attacking Endpoints

The NATFW NSLP requires firewalls to forward NSLP messages, a malicious node may keep sending NSLP messages to a target. This may consume the access network resources of the victim, drain the battery

of the victim's terminal and may force the victim to pay for the received although undesired data.

This threat may be more particularly be relevant in networks where access link is a limited resource, for instance in cellular networks, and where the terminal capacities are limited.

SECURITY REQUIREMENT: A NATFW NSLP node MUST be configurable to block unauthorized signaling message.

5.6.2 Flooding with REA messages from inside

Although we are more concerned with possible attacks from outside the network, we need also to consider possible attacks from inside the network.

An adversary inside the network could send arbitrary RESERVE-EXTERNAL-ADDRESS messages. At a certain point the NAT will run out of port numbers and the access for other users to the outside will be disabled.

SECURITY REQUIREMENT: The NAT/FW NSLP node MUST authorize state creation for the RESERVE-EXTERNAL-ADDRESS message. Furthermore, the NAT/FW NSLP implementation MUST prevent denial of service attacks involving the allocation of an arbitrary number of NAT bindings or the installation of a large number of packet filters.

5.7 Man-in-the-Middle Attacks

Figure 41 illustrates a possible man-in-the-middle attack using the

RESERVE-EXTERNAL-ADDRESS (REA) message. This message travels from DR towards the public Internet. The message might not be intercepted because there are no NSIS aware middleboxes.

Imagine such an NSIS signaling message is then intercepted by an adversary Mallory (M). M returns a faked RESPONSE message whereby the adversary pretends that a NAT binding was created. This NAT binding is returned with the RESPONSE message. Mallory might insert it own IP address in the response, the IP address of a third party or the address of a black hole. In the first case, the DR thinks that the address of Mallory M is its public address and will inform the DS about it. As a consequence, the DS will send the data traffic to Mallory M.

The data traffic from the DS to the DR will re-directed to Mallory M. M will be able to read, modify or block the data traffic (if the end-

to-end communication itself does not experience protection). Eavesdropping and modification is only possible if the data traffic is itself unprotected.

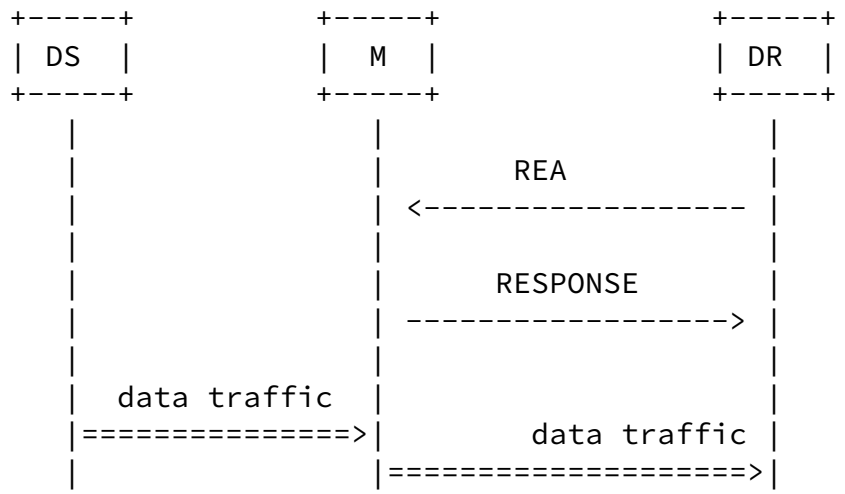


Figure 41: MITM attack using the RESERVE-EXTERNAL-ADDRESS message

SECURITY REQUIREMENT: Mutual authentication between neighboring NATFW NSLP MUST be provided. To ensure that only legitimate nodes along the path act as NSIS entities the initiator MUST authorize the responder. In the example in Figure 41 the firewall FW must

perform an authorization with the neighboring entities.

5.8 Message Modification by non-NSIS on-path node

An unauthorized on-path node along the path towards the destination could easily modify, inject or just drop an NSIS message. It could also hijack or disrupt the communication.

SECURITY REQUIREMENT: Message integrity, replay protection and data origin authentication between neighboring NAT/FW NSLPs MUST be provided.

5.9 Message Modification by malicious NSIS node

Message modification by an NSIS node that became malicious is more serious. An adversary could easily create arbitrary pinholes or NAT bindings. For example:

- NATs need to modify the source/destination of the data flow in the 'create session' message.

- Each middlebox along the path may change the requested lifetime in the CREATE message to fit their needs and/or local policy.

SECURITY REQUIREMENT: Malicious NSIS NATs and firewalls will not be addressed by this specification.

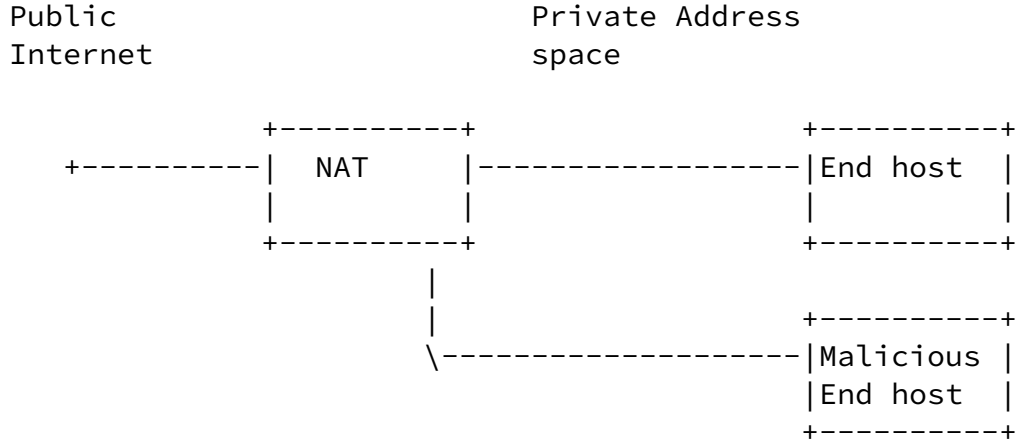
5.10 Session Modification/Deletion

Section 4.10 in [8] describes a threat where an adversary is able to modify previously installed state information at NATFW NSLP nodes along the path. An adversary therefore needs to know session specific information, such as the session identifier and MRI information.

SECURITY REQUIREMENT: No countermeasure will be provided as part of this document. The fact that the adversary needs to learn the randomly generated Session-ID already provides some degree of protection (although not perfect protection).

5.10.1 Misuse of mobility in NAT handling

Another kind of session modification is related to mobility scenarios. NSIS allows end hosts to be mobile, it is possible that an NSIS node behind a NAT needs to update its NAT binding in case of address change. Whenever a host behind a NAT initiates a data transfer, it is assigned an external IP and port number. In typical mobility scenarios, the DR might also obtain a new address according to the topology and it should convey its new IP address to the NAT. The NAT is assumed to modify these NAT bindings based on the new IP address conveyed by the end host.



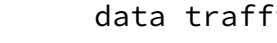
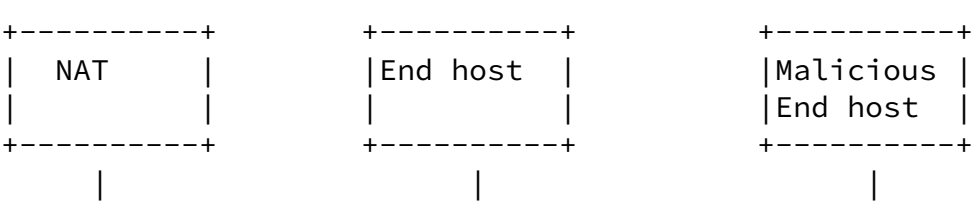


Figure 42: Misuse of mobility in NAT binding

A NAT binding can be changed with the help of NSIS signaling. When a DR moves to a new location and obtains a new IP address, it sends an NSIS signaling message to modify the NAT binding. It would use the Session-ID and the new flow-id to update the state. The NAT updates the binding and the DR continues to receive the data traffic. Consider the scenario in Figure 42 where an the end host(DR) and the adversary are behind a NAT. The adversary pretending that it is the end host could generate a spurious signaling message to update the state at the NAT. This could be done for these purposes:

- Redirecting packets to the attacker as in Figure 43.
- Third party flooding by redirecting packets to arbitrary hosts
- Service disruption by redirecting to non-existing hosts



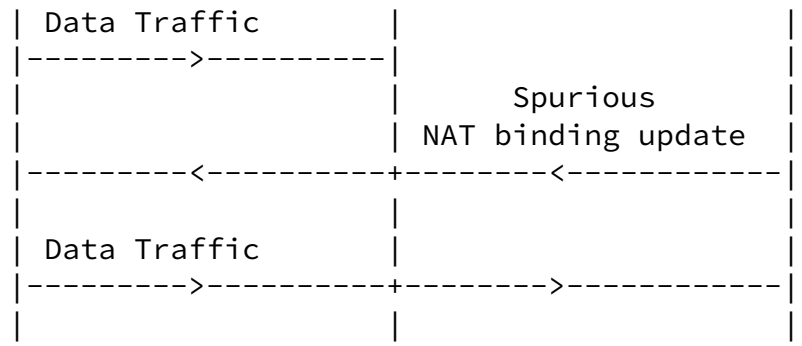


Figure 43: Connection Hijacking

SECURITY REQUIREMENT: A NAT/FW signaling message MUST be authenticated, integrity and replay protected between neighboring NAT/FW NSLP nodes. The NSIS NATFW NSLP aware NAT MUST authorize the end host to insure that the messages are indeed belonging to the previously established session.

5.11 Misuse of unreleased sessions

Assume that DS (N1) initiates NSIS session with DR (N2) through a series of middleboxes as in Figure 44. When the DS is sending data to DR, it might happen that the DR disconnects from the network (crashes or moves out of the network in mobility scenarios). In such cases, it is possible that another node N3 (which recently entered the network protected by the same firewall) is assigned the same IP address that was previously allocated to N2. The DS could take advantage of the firewall policies installed already, if the refresh interval time is very high. The DS can flood the node (N3), which will consume the access network resources of the victim forcing it to pay for unwanted traffic as shown in Figure 45. Note that here we make the assumption that the data receiver has to pay for receiving data packets.

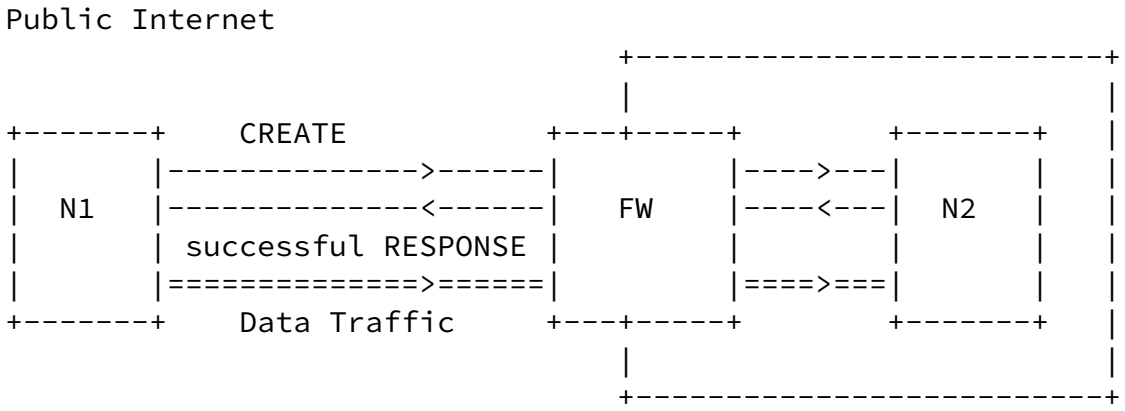


Figure 44: Before mobility

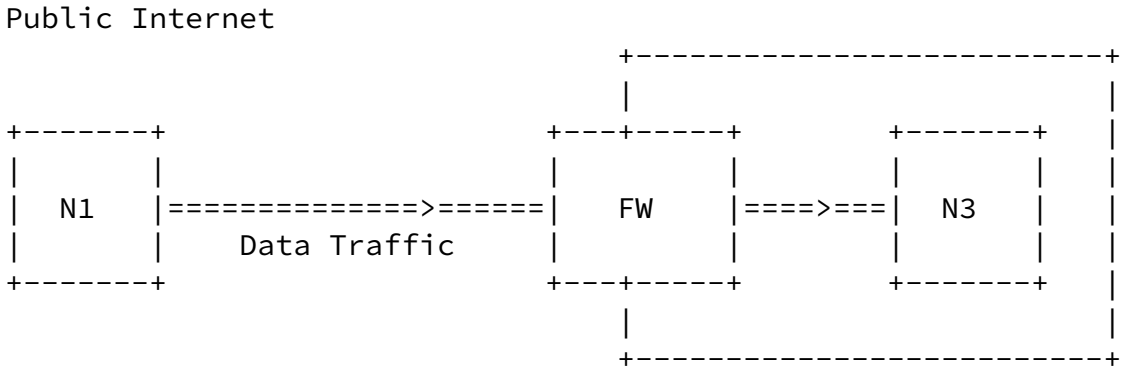


Figure 45: After mobility

Also, this threat is valid for the other direction as well. The DS which is communicating with the DR may disconnect from the network and this IP address may be assigned to a new node that had recently entered the network. This new node could pretend to be the DS and send data traffic to the DR in conformance with the firewall policies and cause service disruption.

SECURITY REQUIREMENT: In order to allow firewalls to verify that a legitimate end host indeed transmitted data traffic it is necessary to provide data origin authentication. This is, however, outside the scope of this document. Hence, there are no security requirements imposed by this threat, which will be addressed by the NATFW NSLP.

5.12 Data Traffic Injection

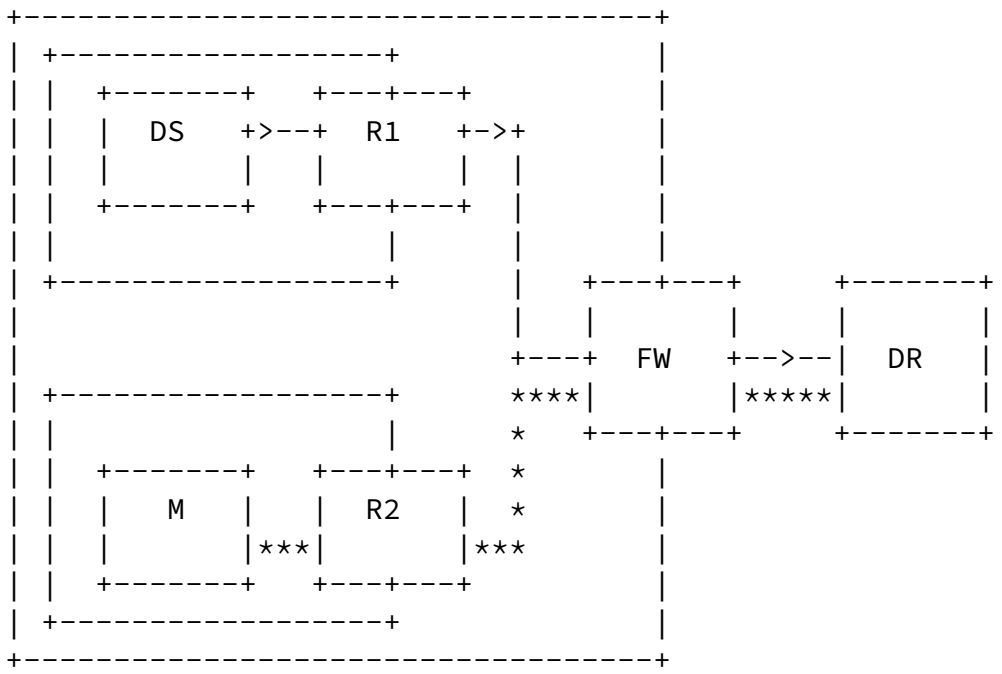
In some environments, such as enterprise networks, it is still common to perform authorization for access to a service based on the source IP address of the service requester. There is no doubt that this

practice by itself represents a security weakness. Using IPy spoofing a connection, an attacker an adversary is able to reach the target machines if they match , using the existing firewall rules.

The adversary is able to inject its own data traffic in conformance with the firewall policies simultaneously along with the genuine DS.

SECURITY REQUIREMENT: Since IP spoofing is a general limitation of non-cryptographic packet filters no countermeasures need to be taken by the NAT/FW NSLP. Techniques such as ingress filtering (described below) and data origin authentication (such as provided with IPsec based VPNs) can help mitigate this threat. This issue is, however, outside the scope of this document.

Ingress Filtering: Consider the scenario shown in Figure 46. In this scenario the DS is behind a router (R1) and a malicious node (M) is behind another router (R2). The DS communicates with the DR through a firewall (FW). The DS initiates NSIS signaling and installs firewall policies at FW. But the malicious node is also able to send data traffic using DS's source address. If R2 implements ingress filtering, these spoofed packets will be blocked. But this ingress filtering may not work in all scenarios. If both the DS and the malicious node are behind the same router, then the ingress filter will not be able to detect the spoofed packets as both the DS and the malicious node are in the same address range.



----->---- = genuine data traffic
********* = spoofed data traffic

Figure 46: Ingress filtering

5.13 Eavesdropping and Traffic Analysis

By collecting NSLP messages, an adversary is able to learn policy rules for packet filters and knows which ports are open. It can use this information to inject its own data traffic due to the IP spoofing capability already mentioned in Section 5.12. An on-path adversary could also observe the data traffic and he could conclude that it is possible to traverse a firewall.

An adversary could learn authorization tokens included in CREATE

messages and use them to launch replay-attacks or to create a session with its own address as source address. This threat is discussed in the respective document suggesting the usage of authorization token in the NSIS protocol suite.

SECURITY REQUIREMENT: The threat of eavesdropping itself does not mandate the usage of confidentiality protection since an adversary can also eavesdrop on data traffic. In the context of a particular security solutions (e.g., authorization tokens) it MAY be necessary to offer confidentiality protection. The latter aspect is outside the scope of this document.

5.14 Security Framework for the NAT/Firewall NSLP

Based on the identified threats a list of security requirements has been created.

5.14.1 Security Protection between neighboring NATFW NSLP Nodes

Based on the analyzed threats it is necessary to provide, between neighboring NATFW NSLP nodes, the following mechanism: provide

- data origin authentication
- replay protection
- integrity protection and
- optionally confidentiality protection

To consider the aspect of authentication and key exchange the security mechanisms provided in [1] between neighboring nodes MUST be enabled when sending NATFW signaling messages. The proposed security mechanisms at GIST provide support for authentication and key exchange in addition to denial of service protection. Depending on the chosen security protocol, support for multiple authentication protocols might be provided. The mandatory support for security, demands the usage of C-MODE for the delivery of data packets and the usage of D-MODE only to discover the next NATFW NSLP aware node along the path. Almost all security threats at the NATFW NSLP layer can be

prevented by using a mutually authenticated Transport Layer secured connection and by relying on authorization by the neighboring NATFW NSLP entities.

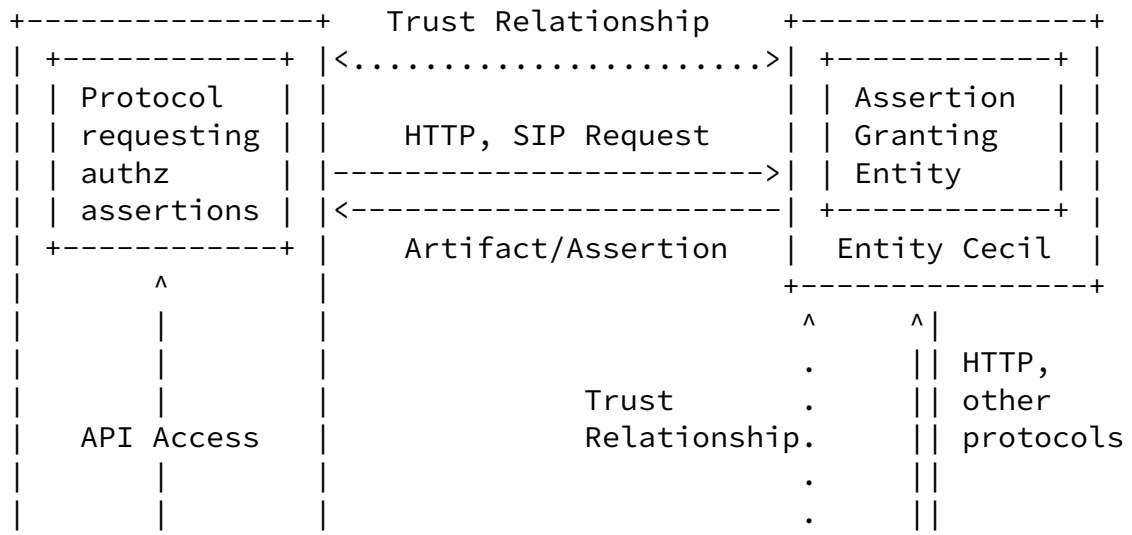
5.14.2 Security Protection between non-neighboring NATFW NSLP Nodes

Based on the security threats and the listed requirements it was noted that some scenarios threats also demand authentication and authorization of a NATFW signaling entity (including the initiator) towards a non-neighboring node. This mechanism mainly demands entity authentication. Additionally, security protection of certain payloads MAY is be required between non-neighboring signaling entities and the Cryptographic Message Syntax (CMS) [17] SHOULD be used. The most important information exchanged at the NATFW NSLP is information related to the establishment for firewall pinholes and NAT bindings. This information can, however, not be protected over multiple NSIS NATFW NSLP hops since this information might change depending on the capability of each individual NATFW NSLP node. Protection using CMS is not described in this document.

Some scenarios might also benefit from the usage of authorization tokens. Their purpose is to associate two different signaling protocols (e.g., SIP and NSIS) and their authorization decision. These tokens are obtained by non-NSIS protocols, such as SIP or as part of network access authentication. When a NAT or firewall along the path receives the token it might be verified locally or passed to the AAA infrastructure.

Examples of authorization tokens or assertions can be found in RFC 3520 [21] and RFC 3521 [22]. Security Assertion Markup Language (SAML) is an example for a more recent mechanisms carrying authorization specific assertions. For details about SAML see [24], [25] and [26]. Figure 47 shows an example of this protocol interaction. An authorization token is provided by the SIP proxy, which acts as the assertion generating entity and gets delivered to the end host with proper authentication and authorization. When the NATFW signaling message is transmitted towards the network, the authorization token is attached to the signaling messages to refer to the previous authorization decision. The assertion verifying entity needs to process the token or it might be necessary to interact with the assertion granting entity using HTTP (or other protocols). As a

result of a successfully authorization by a NATFW NSLP node, the requested action is executed and later a RESPONSE message is generated.



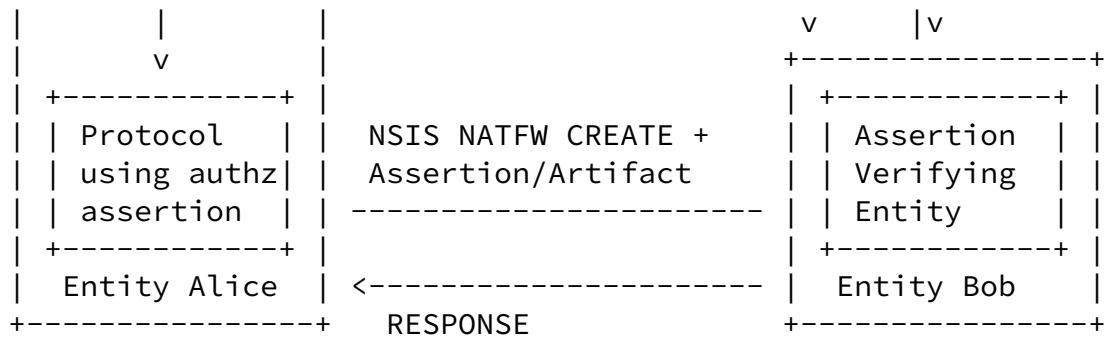


Figure 47: Authorization Token Usage

Threats against the usage of authorization tokens have been mentioned in [8] and also in Section 5.5. Hence, it is required to provide confidentiality protection to avoid allowing an eavesdropper to learn the token and to use it in another session (replay attack). The token itself also needs to be protected against tempering.

This document does provide an initial specification of an NATFW NSLP object for usage of authorization tokens. The NATFW_CREDENTIAL object can carry authorization token or any other type.

6. IAB Considerations on UNSAF

UNilateral Self-Address Fixing (UNSAF) is described in [15] as a process at originating endpoints that attempt to determine or fix the address (and port) by which they are known to another endpoint. UNSAF proposals, such as STUN [RFC3489] are considered as a general class of workarounds for NAT traversal and as solutions for scenarios

with no middlebox communication.

This memo specifies a path-coupled middlebox communication protocol, i.e., the NSIS NATFW NSLP. NSIS in general and the NATFW NSLP are not intended as a short-term workaround, but more as a long-term solution for middlebox communication. In NSIS, endpoints are involved in allocating, maintaining, and deleting addresses and ports at the middlebox. However, the full control of addresses and ports at the middlebox is at the NATFW NSLP daemon located to the respective NAT.

Therefore, this document addresses the UNSAF considerations in [RFC3424] by proposing a long-term alternative solution.

7. IANA Considerations

This section provides guidance to the Internet Assigned Numbers Authority (IANA) regarding registration of values related to the NATFW NSLP, in accordance with BCP 26 RFC 2434 [16].

The NATFW NSLP requires IANA to create a number of new registries. These registries may require further coordination with the registries of the NTLP [1] and the QoS NSLP [6].

NATFW NSLP Message Type Registry
The NATFW NSLP Message Type is a 8 bit value. The allocation of values for new message types requires standards action. Updates and deletion of values from the registry is not possible. This specification defines five NATFW NSLP message types, which form the initial contents of this registry. IANA is requested to add these five NATFW NSLP Message Types: CREATE, REA, TRACE, RESPONSE, and NOTIFY.

NATFW NSLP Header Flag Registry
NATFW NSLP messages have a messages-specific 8 bit flags/reserved field in their header. The registration of flags is subject to IANA registration. The allocation of values for flag types requires standards action. Updates and deletion of values from the registry is not possible. This specification defines only one flag, the P flag in Figure 21.

NSLP Object Type Registry
This document defines 10 objects for the NATFW NSLP: NATFW_LT, NATFW_EXT_IP, NATFW_EFI, NATFW_INFO, NATFW_NONCE, NATFW_MSN, NATFW_DTINFO_IPv4, NATFW_TRACE, NATFW_CREDENTIAL, NATFW_ICMP_TYPES. The allocation of values for new message types requires standards action. IANA is request to assigned values for them from NSLP Object Type registry and to replace the corresponding IANA-TBD tags with the numeric values.

NSLP Response Code Registry
In addition it defines a number of Response Codes for the NATFW NSLP. These can be found in Section 4.2.4 and are to be assigned values from NSLP Response Code registry. The allocation of values for Response Codes Codes requires standards action. IANA is request to assigned values for them from NSLP Response Code registry.

Furthermore, IANA is requested to add a new value to the NSLP Identifiers (NSLPID) registry defined in [1] for the the NATFW NSLP.

8. Open Issues

A more detailed list of open issue can be found at: https://kobe.netlab.nec.de/roundup/nsis-natfw-nslp/index

9. Acknowledgments

We would like to thank the following individuals for their contributions to this document at different stages:

- Marcus Brunner and Henning Schulzrinne for work on work on IETF drafts which lead us to start with this document,
- Miquel Martin for his help on the initial version of this document,
- Srinath Thiruvengadam and Ali Fessi work for their work on the NAT/firewall threats draft,
- Henning Peters for his comments and suggestions,
- and the NSIS working group.

10. References

10.1 Normative References

[1] Schulzrinne, H. and R. Hancock, "GIST: General Internet Signaling Transport", draft-ietf-nsis-ntlp-08 (work in progress), September 2005.

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

[3] Elz, R. and R. Bush, "Serial Number Arithmetic", RFC 1982, August 1996.

10.2 Informative References

[4] Hancock, R., Karagiannis, G., Loughney, J., and S. Van den Bosch, "Next Steps in Signaling (NSIS): Framework", RFC 4080, June 2005.

[5] Brunner, M., "Requirements for Signaling Protocols", RFC 3726, April 2004.

[6] Bosch, S., "NSLP for Quality-of-Service signalling", draft-ietf-nsis-qos-nslp-08 (work in progress), October 2005.

[7] Srisuresh, P., Kuthan, J., Rosenberg, J., Molitor, A., and A. Rayhan, "Middlebox communication architecture and framework", RFC 3303, August 2002.

[8] Tschofenig, H. and D. Kroeselberg, "Security Threats for Next Steps in Signaling (NSIS)", RFC 4081, June 2005.

[9] Srisuresh, P. and M. Holdrege, "IP Network Address Translator (NAT) Terminology and Considerations", RFC 2663, August 1999.

[10] Tsirtsis, G. and P. Srisuresh, "Network Address Translation - Protocol Translation (NAT-PT)", RFC 2766, February 2000.

[11] Carpenter, B. and S. Brim, "Middleboxes: Taxonomy and Issues", RFC 3234, February 2002.

[12] Srisuresh, P., Tsirtsis, G., Akkiraju, P., and A. Heffernan, "DNS extensions to Network Address Translators (DNS_ALG)", RFC 2694, September 1999.

[13] Braden, B., Zhang, L., Berson, S., Herzog, S., and S. Jamin, "Resource ReSerVation Protocol (RSVP) -- Version 1 Functional

Specification", RFC 2205, September 1997.

[14] Yadav, S., Yavatkar, R., Pabbati, R., Ford, P., Moore, T., Herzog, S., and R. Hess, "Identity Representation for RSVP", RFC 3182, October 2001.

[15] Daigle, L. and IAB, "IAB Considerations for UNilateral Self-Address Fixing (UNSAF) Across Network Address Translation", RFC 3424, November 2002.

[16] Narten, T. and H. Alvestrand, "Guidelines for Writing an IANA Considerations Section in RFCs", BCP 26, RFC 2434, October 1998.

[17] Housley, R., "Cryptographic Message Syntax (CMS)", RFC 3369, August 2002.

[18] Rosenberg, J., Schulzrinne, H., Camarillo, G., Johnston, A., Peterson, J., Sparks, R., Handley, M., and E. Schooler, "SIP: Session Initiation Protocol", RFC 3261, June 2002.

[19] Rosenberg, J., Weinberger, J., Huitema, C., and R. Mahy, "STUN

- Simple Traversal of User Datagram Protocol (UDP) Through Network Address Translators (NATs)", RFC 3489, March 2003.

[20] Westerinen, A., Schnizlein, J., Strassner, J., Scherling, M., Quinn, B., Herzog, S., Huynh, A., Carlson, M., Perry, J., and S. Waldbusser, "Terminology for Policy-Based Management", RFC 3198, November 2001.

[21] Hamer, L-N., Gage, B., Kosinski, B., and H. Shieh, "Session Authorization Policy Element", RFC 3520, April 2003.

[22] Hamer, L-N., Gage, B., and H. Shieh, "Framework for Session Set-up with Media Authorization", RFC 3521, April 2003.

[23] Alfano, F., "Diameter Quality of Service Application", draft-alfano-aaa-qosprot-05 (work in progress), October 2005.

[24] Maler, E., Philpott, R., and P. Mishra, "Bindings and Profiles for the OASIS Security Assertion Markup Language (SAML) V1.1", September 2003.

[25] Maler, E., Philpott, R., and P. Mishra, "Assertions and Protocol for the OASIS Security Assertion Markup Language (SAML) V1.1", September 2003.

[26] Maler, E. and J. Hughes, "Technical Overview of the OASIS

Security Assertion Markup Language (SAML) V1.1", March 2004.

[27] Roedig, U., Goertz, M., Karten, M., and R. Steinmetz, "RSVP as firewall Signalling Protocol", Proceedings of the 6th IEEE Symposium on Computers and Communications, Hammamet, Tunisia p

Authors' Addresses

Martin Stiemerling
Network Laboratories, NEC Europe Ltd.
Kurfuersten-Anlage 36
Heidelberg 69115
Germany

Phone: +49 (0) 6221 905 11 13
Email: stiemerling@netlab.nec.de
URI: http://www.stiemerling.org

Hannes Tschofenig
Siemens AG
Otto-Hahn-Ring 6
Munich 81739
Germany

Phone:
Email: Hannes.Tschofenig@siemens.com
URI: http://www.tschofenig.com

Cedric Aoun
Ecole Nationale Superieure des Telecommunications
Paris
France

Email: cedric@caoun.net

Elwyn Davies
Folly Consulting
Soham
UK

Phone: +44 7889 488 335
Email: elwynd@dial.pipex.com

Appendix A. Selecting Signaling Destination Addresses for REA

As with all other message types, REA messages need a reachable final destination IP address. But as many applications do not provide a destination IP address in the first place, there is a need to choose a destination address for REA messages. This destination address can be the final target, but for applications which do not provide an upfront address, the destination address has to be chosen independently. Choosing the 'correct' destination IP address may be

difficult and it is possible there is no 'right answer'.

1. Public IP address of the data sender:

    * Assumption:

        + The data receiver already learned the IP address of the data sender (e.g., via a third party).

    * Problems:

        + The data sender might also be behind a NAT. In this case the public IP address of the data receiver is the IP address allocated at this NAT.

        + Due to routing asymmetry it might be possible that the routes taken by a) the data sender and the application server b) the data sender and NAT B might be different. As a consequence it might be necessary to advertise a new (and different) external IP address within the application (which may or may not allow that) after using NSIS to establish a NAT binding.

2. Public IP address of the data receiver:

    * Assumption:

        + The data receiver already learned his externally visible IP address (e.g., based on the third party communication).

    * Problems:

        + Communication with a third party is required.

3. IP address of the Application Server:

    * Assumption:

        + An application server (or a different third party) is available.

* Problems:

  + If the NSIS signaling message is not terminated at the NAT of the local network then an NSIS unaware application server might discard the message.

  + Routing might not be optimal since the route between a) the data receiver and the application server b) the data receiver and the data sender might be different.

Appendix B. Applicability Statement on Data Receivers behind Firewalls

Section 3.8.2 describes how data receivers behind middleboxes can instruct upstream firewalls/NATs to forward NATFW NSLP signaling towards them. Finding an upstream edge-NAT in address environment with NAT'ed addresses is quite easy. It is only required to find some edge-NAT, as the data traffic will be route-pinned to the NAT, which is done with the LE-MRM. Locating the appropriate edge-firewall with the PC-MRM, sent upstream is difficult. For cases with a single, symmetric route from the Internet to the data receiver, it is quite easy; simply follow the default route in the upstream direction.

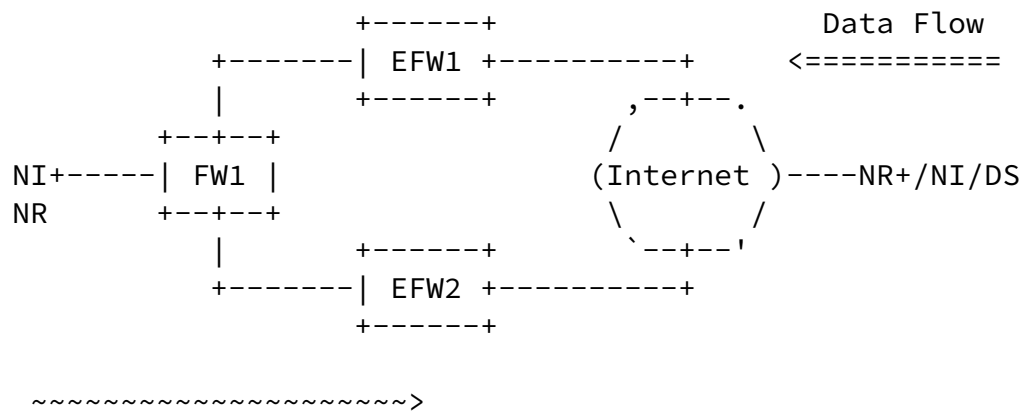


Figure 48: Data receiver behind multiple, parallel located firewalls

When a data receiver, and thus NR, is located in a network site that is multihomed with several independently firewalled connections to the public Internet (as shown in Figure 48), the specific firewall through which the data traffic will be routed has to be ascertained. NATFW NSLP signaling messages sent from the NI+/NR during the REA request message exchange towards the NR+ must be routed by the NTLP to the edge-firewall that will be passed by the data traffic as well. The NTLP would need to be aware about the routing within the Internet to determine the path between DS and DR. Out of this, the NTLP could determine which of the edge-firewalls, either EFW1 or EFW2, must be selected to forward the NATFW NSLP signaling. Signaling to the wrong edge-firewall, as shown in Figure 48, would install the NATFW NSLP policy rules at the wrong device. This causes either a blocked data flow (when the policy rule is 'allow') or an ongoing attack (when the policy rule is 'deny'). Requiring the NTLP to know all about the routing within the Internet is definitely a tough challenge and usually not possible. In such described case, the NTLP must basically give up and return an error to the NSLP level, indicating

that the next hop discovery is not possible.

Appendix C. Firewall and NAT Resources

This section gives some examples on how NATFW NSLP policy rules could be mapped to real firewall or NAT resources. The firewall rules and NAT bindings are described in a natural way, i.e., in a way one will find it in common implementation.

C.1 Wildcarding of Policy Rules

The policy rule/MRI to be installed can be wildcarded to some degree. Wildcarding applies to IP address, transport layer port numbers, and the IP payload (or next header in IPv6). Processing of wildcarding splits into the NTLP and the NATFW NSLP layer. The processing at the NTLP layer is independent of the NSLP layer processing and per layer constraints apply. For wildcarding in the NTLP see Section 5.8 of [1].

Wildcarding at the NATFW NSLP level is always a node local policy decision. A signaling message carrying a wildcarded MRI (and thus policy rule) arriving at an NSLP node can be rejected if the local policy does not allow the request. For instance, a MRI with IP addresses set (not wildcarded), transport protocol TCP, and TCP port numbers completely wildcarded. Now the local policy allows only requests for TCP with all ports set and not wildcarded. The request is going to be rejected.

C.2 Mapping to Firewall Rules

This section describes how a NSLP policy rule signaled with a CREATE request message is mapped to a firewall rule. The MRI is set as follows:

- network-layer-version=IPv4
- source-address=192.0.2.100, prefix-length=32
- destination-address=192.0.50.5, prefix-length=32
- IP-protocol=UDP

- L4-source-port=34543, L4-destination-port=23198

The NATFW_EFI object is set to action=allow and sub_ports=0.

The resulting policy rule (firewall rule) to be installed might look like: allow udp from 192.0.2.100 port=34543 to 192.0.50.5 port=23198

C.3 Mapping to NAT Bindings

This section describes how a NSLP policy rule signaled with a REA request message is mapped to a NAT binding. It is assumed that the REA message is sent by a NI+ being located behind a NAT and does contain a NATFW_DTINFO object. The MRI is set following using the signaling destination address, since the IP address of the real data sender is not known:

- network-layer-version=IPv4
- source-address= 192.168.5.100
- destination-address=SDA
- IP-protocol=UDP

The NATFW_EFI object is set to action=allow and sub_ports=0. The NATFW_DTINFO object contains these parameters:

- P=1
- dest prefix=0
- protocol=UDP
- dst port number = 20230, src port number=0
- src IP=0.0.0.0

The edge-NAT allocates the external IP 192.0.2.79 and port 45000.

The resulting policy rule (NAT binding) to be installed could look

like: translate from any to 192.0.2.79 port=45000 to 192.168.5.100 port=20230

C.4 NSLP Handling of Twice-NAT

The dynamic configuration of twice-NATs requires application level support, as stated in Section 2.5. The NATFW NSLP cannot be used for configuring twice-NATs if application level support is needed. Assuming application level support performing the configuration of the twice-NAT and the NATFW NSLP being installed at this devices, the NATFW NSLP must be able to traverse it. The NSLP is probably able to traverse the twice-NAT, as any other data traffic, but the flow information stored in the NTLP's MRI will be invalidated through the translation of source and destination address. The NATFW NSLP implementation on the twice-NAT MUST intercept NATFW NSLP and NTLP

signaling messages as any other NATFW NSLP node does. For the given signaling flow, the NATFW NSLP node MUST look up the corresponding IP address translation and modify the NTLP/NSLP signaling accordingly. The modification results in an updated MRI with respect to the source and destination IP addresses.

Intellectual Property Statement

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

Disclaimer of Validity

Copyright Statement

Acknowledgment

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