

IPsec Working Group
INTERNET-DRAFT
Category: Informational
<[draft-ietf-ipsec-nat-reqts-04.txt](#)>
3 March 2003

Bernard Aboba
William Dixon
Microsoft

IPsec-NAT Compatibility Requirements

Status of this Memo

This document is an Internet-Draft and is in full conformance with all provisions of [Section 10 of RFC 2026](#).

Internet-Drafts are working documents of the Internet Engineering Task Force (IETF), its areas, and its working groups. Note that other groups may also distribute working documents as Internet-Drafts.

Internet-Drafts are draft documents valid for a maximum of six months and may be updated, replaced, or obsoleted by other documents at any time. It is inappropriate to use Internet-Drafts as reference material or to cite them other than as "work in progress."

The list of current Internet-Drafts can be accessed at
<http://www.ietf.org/ietf/1id-abstracts.txt>

The list of Internet-Draft Shadow Directories can be accessed at
<http://www.ietf.org/shadow.html>.

Copyright Notice

Copyright (C) The Internet Society (2003). All Rights Reserved.

Abstract

Perhaps the most common use of IPsec is in providing virtual private networking capabilities. One very popular use of VPNs is to provide telecommuter access to the corporate Intranet. Today NATs are widely deployed in home gateways, as well as in other locations likely to be used by telecommuters, such as hotels. The result is that IPsec-NAT incompatibilities have become a major barrier to deployment of IPsec in one of its principal uses. This draft describes known incompatibilities between NAT and IPsec, and describes the requirements for addressing them.

Table of Contents

1.	Introduction	3
1.1	Requirements language	3
2.	Known incompatibilities between NA(P)T and IPsec	3
2.1	Intrinsic NA(P)T issues	4
2.2	NA(P)T implementation weaknesses	7
2.3	Helper incompatibilities	8
3.	Requirements for IPsec-NAT compatibility	8
4.	Existing solutions	11
4.1	IPsec tunnel mode	11
4.2	RSIP	12
4.3	6to4	13
5.	Security considerations	13
6.	Normative references	14
7.	Informative references	15
	ACKNOWLEDGMENTS	16
	AUTHORS' ADDRESSES	16
	Full Copyright Statement	16
	Intellectual property statement	17

1. Introduction

Perhaps the most common use of IPsec [[RFC2401](#)] is in providing virtual private networking capabilities. One very popular use of VPNs is to provide telecommuter access to the corporate Intranet. Today NATs, as described in [[RFC3022](#)] and [[RFC2663](#)], are widely deployed in home gateways, as well as in other locations likely to be used by telecommuters, such as hotels. The result is that IPsec-NAT incompatibilities have become a major barrier to deployment of IPsec in one of its principal uses. This draft describes known incompatibilities between NAT and IPsec, and describes the requirements for addressing them.

1.1. Requirements language

In this document, the key words "MAY", "MUST", "MUST NOT", "optional", "recommended", "SHOULD", and "SHOULD NOT", are to be interpreted as described in [[RFC2119](#)].

Please note that the requirements specified in this document are to be used in evaluating protocol submissions. As such, the requirements language refers to capabilities of these protocols; the protocol documents will specify whether these features are required, recommended, or optional. For example, requiring that a protocol support confidentiality is NOT the same thing as requiring that all protocol traffic be encrypted.

A protocol submission is not compliant if it fails to satisfy one or more of the MUST or MUST NOT requirements for the capabilities that it implements. A protocol submission that satisfies all the MUST, MUST NOT, SHOULD and SHOULD NOT requirements for its capabilities is said to be "unconditionally compliant"; one that satisfies all the MUST and MUST NOT requirements but not all the SHOULD or SHOULD NOT requirements for its protocols is said to be "conditionally compliant."

2. Known incompatibilities between NA(P)T and IPsec

The incompatibilities between NA(P)T and IPsec can be divided into three categories:

- [1] Intrinsic NA(P)T issues. These incompatibilities derive directly from the NA(P)T functionality described in [[RFC3022](#)]. These incompatibilities will therefore be present in any NA(P)T device.
- [2] NA(P)T implementation weaknesses. These incompatibilities are not intrinsic to NA(P)T, but are present in many NA(P)T implementations. Included in this category are problems in handling inbound or outbound fragments. Since these issues are not

intrinsic to NA(P)T, they can in principle be addressed in future NA(P)T implementations. However, since the implementation problems appear to be wide spread, they need to be taken into account in a NA(P)T traversal solution.

- [3] Helper issues. These incompatibilities are present in NA(P)T devices which attempt to provide for IPsec NA(P)T traversal. Ironically, this "helper" functionality creates further incompatibilities, making an already difficult problem harder to solve. While IPsec traversal "helper" functionality is not present in all NA(P)Ts, these features are becoming sufficiently popular that they also need to be taken into account in a NA(P)T traversal solution.

2.1. Intrinsic NA(P)T issues

Incompatibilities that are intrinsic to NA(P)T include:

- a) Incompatibility between IPsec AH [[RFC2402](#)] and NAT. Since the AH header incorporates the IP source and destination addresses in the keyed message integrity check, NAT or reverse NAT devices making changes to address fields will invalidate the message integrity check. Since IPsec ESP [[4](#)] does not incorporate the IP source and destination addresses in its keyed message integrity check, this issue does not arise for ESP.
- b) Incompatibility between checksums and NAT. TCP and UDP checksums have a dependency on the IP source and destination addresses through inclusion of the "pseudo-header" in the calculation. As a result, where checksums are calculated and checked on receipt, they will be invalidated by passage through a NAT or reverse NAT device.

As a result, IPsec ESP will only pass unimpeded through a NAT if TCP/UDP protocols are not involved (as in IPsec tunnel mode or IPsec/GRE), or checksums are not calculated (as is possible with IPv4 UDP). As described in [[RFC793](#)], TCP checksum calculation and verification is required in IPv4. UDP/TCP checksum calculation and verification is required in IPv6.

SCTP as defined in [[RFC2960](#)] and [[RFC3309](#)] uses a CRC32C algorithm calculated only on the SCTP packet (common header + chunks), so that the IP header is not covered. As a result, NATs do not invalidate the SCTP CRC, and so the problem does not arise.

Note that since transport mode IPsec traffic is integrity protected and authenticated using strong cryptography, modifications to the packet can be detected prior to checking UDP/TCP

checksums. Thus, checksum verification only provides assurance against errors made in internal processing.

c) Incompatibility between IKE address identifiers and NAT.

Where IP addresses are used as identifiers in IKE MM [[RFC2409](#)] or QM, modification of the IP source or destination addresses by NATs or reverse NATs will result in a mismatch between the identifiers and the addresses in the IP header. As described in [[RFC2409](#)], IKE implementations are required to discard such packets.

In order to avoid use of IP addresses as IKE MM and QM identifiers, userIDs and FQDNs can be used instead. Where user authentication is desired, an ID type of ID_USER_FQDN can be used, as described in [[RFC2407](#)]. Where machine authentication is desired, an ID type of ID_FQDN can be used. In either case it is necessary to verify that the proposed identity matches that enclosed in the certificate. However, while use of USER_FQDN or FQDN identity types is possible within IKE, there are usage scenarios (e.g. SPD entries describing subnets) that cannot be accommodated this way.

d) Incompatibility between fixed IKE destination ports and NAT.

Where multiple hosts behind the NAT initiate IKE SAs to the same responder, a mechanism is needed to allow the NAT to demultiplex the incoming IKE packets from the responder. This is typically accomplished by translating the IKE UDP source port on outbound packets from the initiator. Thus responders must be able to accept IKE traffic from a UDP source port other than 500, and must reply to that port. Care must be taken to avoid unpredictable behavior during re-keys. If the floated source port is not used as the destination port for the re key, the NAT may not be able to send the re-key packets to the correct destination.

e) Incompatibilities between overlapping SPD entries and NAT.

Where hosts behind a NAT negotiate overlapping SPD entries with the same destination in IKE QM, packets may be sent down the wrong IPsec SA. This occurs because to the sender, the IPsec SAs appear to be equivalent, since they exist between the same endpoints and can be used to pass the same traffic.

f) Incompatibilities between IPsec SPI selection and NAT.

Since IPsec ESP traffic is encrypted and thus opaque to the NAT, the NAT must use elements of the IP and IPsec header to demultiplex incoming IPsec traffic. The combination of the destination IP address, security protocol (AH/ESP) and IPsec SPI

is typically used for this purpose.

However, since the outgoing and incoming SPIs are chosen independently, there is no way for the NAT to determine what incoming SPI corresponds to what destination host merely by inspecting outgoing traffic. Thus, were two hosts behind the NAT to attempt to bring up IPsec SAs to the same destination simultaneously, it is possible that the NAT will send the incoming IPsec packets to the wrong destination.

Note that this is not an incompatibility with IPsec per se, but rather with the way it is typically implemented. With both AH and ESP, the receiving host specifies the SPI to use for a given SA. At present, the combination of Destination IP, SPI, and Security Protocol (AH, ESP) uniquely identifies a Security Association. This means that the receiving host can select SPIs such that it has one Security Association (SA) with (SPI=470, Dest IP=1.2.3.4) and a different Security Association with (SPI=470, Dest IP=2.3.4.5).

It is also possible for the receiving host to allocate a unique SPI to each unicast Security Association. In this case, the Destination IP Address need only be checked to see if it is "any valid unicast IP for this host", not checked to see if it is the specific Destination IP address used by the sending host. This approach is completely backwards compatible and only requires the particular receiving host to make a change to its SPI allocation and IPsec_esp_input() code.

- g) Incompatibilities between embedded IP addresses and NAT. Since the payload is integrity protected, any IP addresses enclosed within IPsec packets will not be translatable by a NAT. This renders ineffective Application Layer Gateways (ALGs) implemented within NATs. Protocols that utilize embedded IP addresses include FTP, IRC, SNMP, LDAP, H.323, SIP, SCTP (optionally) and many games. To address this issue, it is necessary to install ALGs on the host or security gateway which can operate on application traffic prior to IPsec encapsulation and after IPsec decapsulation.
- h) Implicit directionality of NA(P)T. NA(P)Ts often require an initial outbound packet to flow through them in order to create inbound mapping state. Directionality prohibits unsolicited establishment of IPsec SAs to hosts behind the NA(P)T.

2.2. NA(P)T implementation weaknesses

Implementation problems present in many NA(P)Ts include:

- i) Inability to handle non-UDP/TCP traffic. Some NAPT's discard non-UDP/TCP traffic. Such NAPT's are unable to pass SCTP, ESP (protocol 50) or AH (protocol 51) traffic.
- j) NAT mapping timeouts. NA(P)Ts vary in the time for which a UDP mapping will be maintained in the absence of traffic. Thus, even where IKE packets can be correctly translated, the translation state may be removed prematurely.
- k) Inability to handle outgoing fragments. Most NA(P)Ts can properly fragment outgoing IP packets in the case where the IP packet size exceeds the MTU on the outgoing interface. However, proper translation of outgoing packets that are already fragmented is difficult and most NAPT's do not handle this correctly. As noted in [Section 6.3 of \[RFC3022\]](#), where two hosts originate fragmented packets to the same destination, the fragment identifiers can overlap. Since the destination host relies on the fragmentation identifier and fragment offset for reassembly, the result will be data corruption. Few NA(P)Ts protect against identifier collisions by supporting identifier translation. Identifier collisions are not an issue when NATs perform the fragmentation, since the fragment identifier need only be unique within a source/destination IP address pair.

Since a fragment can be as small as 68 octets [[RFC790](#)], there is no guarantee that the first fragment will contain a complete TCP header. Thus, a NA(P)T looking to recalculate the TCP checksum may need to modify a subsequent fragment. Since fragments can be reordered, and IP addresses can be embedded and possibly even split between fragments, the NA(P)T will need to perform reassembly prior to completing the translation. Few NA(P)Ts support this.

- l) Inability to handle incoming fragments. Since only the first fragment will typically contain a complete IP/UDP/SCTP/TCP header, NAPT's need to be able to perform the translation based on the source/dest IP address and fragment identifier alone. Since fragments can be reordered, the headers to a given fragment identifier may not be known if a subsequent fragment arrives prior to the initial one, and the headers may be split between fragments. As a result, the NAPT may need to perform reassembly prior to completing the translation. Few NAPT's support this. Note that with NAT, the source/dest IP address is enough to determine the translation,

so that this does not arise. However, it is possible for the IPsec or IKE headers to be split between fragments, so that reassembly may still be required.

2.3. Helper incompatibilities

Incompatibilities between IPsec and NAT "helper" functionality include:

- m) ISAKMP header inspection. Today some NAT implementations attempt to use IKE cookies to de-multiplex incoming IKE traffic. As with source-port de-multiplexing, IKE cookie de-multiplexing results in problems with re-keying, since Phase 1 re-keys typically will not use the same cookies as the earlier traffic.
- n) Special treatment of port 500. Since some IKE implementations are unable to handle non-500 UDP source ports, some NATs do not translate packets with a UDP source port of 500. This means that these NATs are limited to one IPsec client per destination gateway, unless they inspect details of the ISAKMP header to examine cookies which creates the problem noted above.
- o) ISAKMP payload inspection. NA(P)T implementations that attempt to parse ISAKMP payloads may not handle all payload ordering combinations, or support vendor_id payloads for IKE option negotiation.

3. Requirements for IPsec-NAT compatibility

The goal of an IPsec-NAT compatibility solution is to expand the range of usable IPsec functionality beyond that available in an NAT-compatible IPsec tunnel mode solution described above.

In evaluating a solution to IPsec-NAT incompatibility, the following criteria should be kept in mind:

Deployment

Since IPv6 will address the address scarcity issues that frequently lead to use of NA(P)Ts with IPv4, the IPsec-NAT compatibility issue is a transitional problem that needs to be solved in the time frame prior to widespread deployment of IPv6. Therefore, to be useful an IPsec-NAT compatibility solution MUST be deployable on a shorter time scale than IPv6.

Since IPv6 deployment requires changes to routers as well as hosts, a IPsec-NAT compatibility solution which requires changes to both routers and hosts will be deployable on approximately the same time scale as IPv6. Thus, an IPsec-NAT

compatibility solution SHOULD require changes only to hosts, and not to routers.

Among other things, this implies that communication between the host and the NA(P)T SHOULD NOT be required by an IPsec-NAT compatibility solution, since that would require changes to the NA(P)Ts, and interoperability testing between the host and NA(P)T implementations. In order to enable deployment in the short term, it is necessary for the solution to work with existing router and NA(P)T products within the deployed infrastructure.

Protocol Compatibility

An IPsec NAT traversal solution is not expected to resolve issues with protocols that cannot traverse NA(P)T when unsecured with IPsec. Therefore, ALGs may still be needed for some protocols, even when an IPsec NAT traversal solution is available.

Security Since NA(P)T directionality serves a security function, IPsec NA(P)T traversal solutions should not allow arbitrary incoming IPsec or IKE traffic from any IP address to be received by a host behind the NA(P)T, although mapping state should be maintained once bidirectional IKE and IPsec communication is established.

Telecommuter scenario

Since one of the primary uses of IPsec is remote access to corporate Intranets, a NA(P)T traversal solution MUST support NA(P)T traversal via either IPsec tunnel mode or L2TP over IPsec transport mode [[RFC 3193](#)]. This includes support for traversal of more than one NA(P)T between the remote client and the VPN gateway.

The client may have a routable address and the VPN gateway may be behind at least one NA(P)T, or alternatively, both the client and the VPN gateway may be behind one or more NA(P)Ts. Telecommuters may use the same private IP address, each behind their own NA(P)T, or many telecommuters may reside on a private network behind the same NA(P)T, each with their own unique private address, connecting to the same VPN gateway. Since IKE uses UDP port 500 as the destination, it is not necessary to enable multiple VPN gateways operating behind the same external IP address.

In a gateway-gateway scenario, a privately addressed network (DMZ) may be inserted between the corporate network and the Internet. In this design, IPsec security gateways connecting

portions of the corporate network will have private addresses on their external interfaces. A NA(P)T connects the DMZ network to the Internet.

A NAT-IPsec solution MUST enable secure host-host communication via IPsec as well as host-gateway communications. A host on a private network MUST be able to bring up IPsec-protected TCP connections or UDP sessions to another host with one or more NA(P)Ts between them. For example, NA(P)Ts may be deployed within branch offices connecting to the corporate network, with an additional NA(P)T connecting the corporate network to the Internet. This may require special processing of TCP and UDP traffic on the host.

Bringing up SCTP connections to another host with one or more NA(P)Ts between them may present special challenges. SCTP supports multi-homing. If more than one IP address is used, these addresses are transported as part of the SCTP packet during the association setup (in the INIT and INIT-ACK chunks). If only single homed SCTP end-points are used [\[RFC2960\] section 3.3.2.1](#) states:

Note that not using any IP address parameters in the INIT and INIT-ACK is an alternative to make an association more likely to work across a NAT box.

This implies that IP addresses should not be put into the SCTP packet unless necessary. If NATs are present and IP addresses are included then association setup will fail. Recently [ADDIP] has been proposed which allows the modification of the IP address once an association is established. The modification messages have also IP addresses in the SCTP packet, and so will be adversely affected by NATs.

Firewall Compatibility

Since firewalls are widely deployed, a NAT-IPsec compatibility solution MUST enable a firewall administrator to create simple, static access rule(s) to permit or deny IKE and IPsec NA(P)T traversal traffic. This implies, for example, that dynamic allocation of IKE or IPsec destination ports is to be avoided.

Scaling

An IPsec-NAT compatibility solution should be capable of being deployed within an installation consisting of thousands of telecommuters. In this situation, it is not possible to assume that only a single host is communicating with a given destination at a time. Thus, an IPsec-NAT compatibility solution MUST address the issue of overlapping SPD entries and

de-multiplexing of incoming packets.

Mode support

At a minimum, an IPsec-NAT compatibility solution MUST support traversal of the IPsec modes required for support within [\[RFC2401\]](#). For example, an IPsec gateway MUST support ESP tunnel mode NA(P)T traversal, and an IPsec host MUST support IPsec transport mode NA(P)T traversal. The purpose of AH is to protect immutable fields within the IP header (including addresses), and NA(P)T translates addresses, invalidating the AH integrity check. As a result, NA(P)T and AH are fundamentally incompatible and there is no requirement that an IPsec-NAT compatibility solution support AH transport or tunnel mode.

Backward Compatibility and Interoperability

An IPsec-NAT compatibility solution MUST be interoperable with existing IKE/IPsec implementations, so that they can communicate where no NA(P)T is present. This implies that an IPsec-NAT compatibility solution MUST be backwards-compatible with IPsec as defined in [\[RFC2401\]](#) and IKE as defined in [\[RFC2409\]](#). In addition, it SHOULD be able to detect the presence of a NA(P)T, so that NA(P)T traversal support is only used when necessary. This implies that it MUST be possible to determine that an existing IKE implementation does not support NA(P)T traversal, so that a standard IKE conversation can occur, as described in [\[RFC2407\]](#), [\[RFC2408\]](#), and [\[RFC2409\]](#). Note that while this implies initiation of IKE to port 500, there is no requirement for a specific source port, so that UDP source port 500 may or may not be used.

Security An IPsec-NAT compatibility solution MUST NOT introduce additional IKE or IPsec security vulnerabilities. For example, an acceptable solution must demonstrate that it introduces no new denial of service or spoofing vulnerabilities. IKE MUST be allowed to re-key in a bi-directional manner as described in [\[RFC2408\]](#).

[4.](#) Existing solutions

[4.1.](#) IPsec tunnel mode

In a limited set of circumstances, it is possible for an IPsec tunnel mode implementation, such as that described in [\[DHCP\]](#), to traverse NA(P)T successfully. However the requirements for successful traversal are sufficiently limiting that a more general solution is needed:

- [1] IPsec ESP. IPsec ESP tunnels do not cover the outer IP header within the authentication hash, and so will not suffer hash invalidation due to address translation. IPsec tunnels also need not be concerned about checksum invalidation.
- [2] No address validation. Most current IPSEC tunnel mode implementations do not perform source address validation so that incompatibilities between IKE identifiers and source addresses will not be detected. This introduces security vulnerabilities as described in [Section 5](#).
- [3] "Any to Any" SPD entries. IPsec tunnel mode clients can negotiate "any to any" SPDs, which are not invalidated by address translation. This effectively precludes use of SPDs for filtering of allowed tunnel traffic.
- [4] Single client operation. With only a single client behind a NAT, there is no risk of overlapping SPDs. Since the NAT will not need to arbitrate between competing clients, there is also no risk of re-key mis-translation, or improper incoming SPI or cookie de-multiplexing.
- [5] No fragmentation. When certificate authentication is used, IKE fragmentation can be encountered. This can occur when certificate chains are used, or even when exchanging a single certificate if the key size, or size of other certificate fields (such as the distinguished name and other OIDs) is large enough. However, when pre-shared keys are used for authentication, fragmentation is less likely.
- [6] Active sessions. Most VPN sessions typically maintain ongoing traffic flow during their lifetime, so that UDP port mappings are less likely be removed due to inactivity.

[4.2](#). RSIP

RSIP, described in [\[RSIP\]](#) and [\[RSIPFrame\]](#), includes mechanisms for IPsec traversal, as described in [\[RSIPsec\]](#). By enabling host-gateway communication, RSIP addresses issues of IPsec SPI de-multiplexing as well as SPD overlap. It is thus suitable for use in enterprise as well as home networking scenarios. By enabling hosts behind a NAT to share the external IP address of the gateway, this approach is compatible with protocols including embedded IP addresses.

By tunneling IKE and IPsec packets, RSIP avoids changes to the IKE and IPsec protocols, although major changes are required to host IKE and IPsec implementations to retrofit them for RSIP-compatibility. It is thus compatible with all existing protocols (AH/ESP) and modes

(transport and tunnel).

In order to handle de-multiplexing of IKE re-keys, RSIP requires floating of the IKE source port, as well as re-keying to the floated port. As a result, interoperability with existing IPsec implementations is not assured.

RSIP does not satisfy the deployment requirements for a IPsec-NAT compatibility solution because an RSIP-enabled host requires a corresponding RSIP-enabled gateway in order to establish an IPsec SA with another host. Since RSIP requires changes only to clients and routers and not to servers, it is less difficult to deploy than IPv6. However, for vendors, implementation of RSIP requires a substantial fraction of the resources required for IPv6 support. Thus, RSIP solves a "transitional" problem on a long-term time scale, which is not useful.

4.3. 6to4

6to4, as described in [[RFC3056](#)] can form the basis for an IPsec-NAT traversal solution. In this approach, the NAT provides IPv6 hosts with an IPv6 prefix derived from the NAT external IPv4 address, and encapsulates IPv6 packets in IPv4 for transmission to other 6to4 hosts or 6to4 relays. This enables an IPv6 host using IPsec to communicate freely to other hosts within the IPv6 or 6to4 clouds.

While 6to4 is an elegant and robust solution where a single NA(P)T separates a client and VPN gateway, it is not universally applicable. Since 6to4 requires assignment of a routable IPv4 address to the NA(P)T, in order to allow formation of an IPv6 prefix, it is not usable where multiple NA(P)Ts exist between the client and VPN gateway. For example, a NA(P)T with a private address on its external interface, cannot be used by clients behind it to obtain an IPv6 prefix via 6to4.

While 6to4 requires little additional support from hosts that already support IPv6, it does require changes to NATs, which need to be upgraded to support 6to4. As a result, 6to4 may not be suitable for deployment in the short term.

5. Security considerations

By definition, IPsec-NAT compatibility requires that hosts and routers implementing IPsec be capable of securely processing packets whose IP headers are not cryptographically protected. A number of issues arise from this that are worth discussing.

Since IPsec AH cannot pass through a NAT, one of the side effects of providing an IPsec-NAT compatibility solution may be for IPsec ESP with null encryption to be used in place of AH where a NAT exists between the

source and destination. However, it should be noted that ESP with null encryption does not provide the same security properties as AH. For example, there are security risks relating to IP source routing that are precluded by AH, but not by ESP with null encryption.

In addition, since ESP with any transform does not protect against source address spoofing, some sort of source IP address sanity checking needs to be performed. The importance of the anti-spoofing check is not widely understood. There is normally an anti-spoofing check on the Source IP Address as part of `IPsec_{esp,ah}_input()`. This ensures that the packet originates from the same address as that was claimed within the original IKE MM and QM security associations. When a receiving host is behind a NAT, this check might not strictly be meaningful for unicast sessions, whereas in the Global Internet this check is important for tunnel-mode unicast sessions to prevent a spoofing attack described in [[AuthSource](#)].

Let us consider two hosts, A and C, both behind (different) NATs, who negotiate IPsec tunnel mode SAs to router B. Hosts A and C may have different privileges; for example, host A might belong to an employee trusted to access much of the corporate Intranet, while C might be a contractor only authorized to access a specific web site.

If host C sends a tunnel mode packet spoofing A's IP address, as the source, it is important that this packet not be accorded the privileges corresponding to A. If authentication and integrity checking is performed, but no anti-spoofing check (verifying that the originating IP address corresponds to the SPI) then host C may be allowed to reach parts of the network that are off limits. As a result, an IPsec-NAT compatibility scheme MUST provide some degree of anti-spoofing protection.

6. Normative references

- [RFC791] ISI, "Internet Protocol Specification", [RFC 791](#), September 1981.
- [RFC793] Information Sciences Institute, "Transmission Control Protocol", [RFC 793](#), September 1981.
- [RFC2119] Bradner, S., "Key words for use in RFCs to Indicate Requirement Levels", [BCP 14](#), [RFC 2119](#), March 1997.
- [RFC2401] Atkinson, R., Kent, S., "Security Architecture for the Internet Protocol", [RFC 2401](#), November 1998.
- [RFC2402] Kent, S., Atkinson, R., "IP Authentication Header", [RFC 2402](#), November 1998.

- [RFC2406] Kent, S., Atkinson, R., "IP Encapsulating Security Payload (ESP)", [RFC 2406](#), November 1998.
- [RFC2407] Piper, D., "The Internet IP Security Domain of Interpretation of ISAKMP", [RFC 2407](#), November 1998.
- [RFC2409] Harkins, D., Carrel, D., "The Internet Key Exchange (IKE)", [RFC 2409](#), November 1998.
- [RFC2663] Srisuresh, P. and Holdredge, M., "IP Network Address Translator (NAT) Terminology and Considerations", [RFC 2663](#), August 1999.
- [RFC3022] Srisuresh, P., and Egevang, K., "Traditional IP Network Address Translator (Traditional NAT)", [RFC 3022](#), January 2001.

7. Informative references

- [RFC2661] Townsley, W., Valencia, A., Rubens, A., Pall, G., Zorn, G., and Palter, B., "Layer Two Tunneling Protocol L2TP", [RFC 2661](#), August 1999.
- [RFC2960] Stewart, R., et al., "Stream Control Transmission Protocol", [RFC 2960](#), October 2000.
- [RFC3056] Carpenter, B., Moore, K., "Connection of IPv6 Domains via IPv4 Clouds", [RFC 3056](#), February 2001.
- [RFC3309] Stone, J., Stewart, R., Otis, D., "Stream Control Transmission Protocol (SCTP) Checksum Change", [RFC 3309](#), September 2002.
- [RSIPFrame] Borella, M., Lo, J., Grabelsky, D., Montenegro, G., "Realm Specific IP: A Framework", [RFC 3102](#), October 2001.
- [RSIP] Borella, M., Grabelsky, D., Lo, J., Taniguchi, K., "Realm Specific IP: Protocol Specification", [RFC 3103](#), October 2001.
- [RSIPsec] Montenegro, G., Borella, M., "RSIP Support for End-to-End IPsec", [RFC 3104](#), October 2001.
- [DHCP] Patel, B., Aboba, B., Kelly, S., Gupta, V., "DHCPv4 Configuration of IPsec Tunnel Mode", [RFC 3456](#), January 2003.
- [AuthSource] Kent, S., "Authenticated Source Addresses", IPsec WG Archive (<ftp://ftp.ans.net/pub/archive/IPsec>), Message-Id: <v02130517ad121773c8ed@[128.89.0.110]>, January 5, 1996.

[AddIP] Stewart, R., et al., "Stream Control Transmission Protocol (SCTP) Dynamic Address Reconfiguration", Internet draft (work in progress), [draft-ietf-tsvwg-addip-sctp-06.txt](#), September 2002.

Acknowledgments

Thanks to Steve Bellovin of AT&T Research, Michael Tuexen of Siemens, Peter Ford of Microsoft, Ran Atkinson of Extreme Networks and Daniel Senie for useful discussions of this problem space.

Authors' Addresses

Bernard Aboba
Microsoft Corporation
One Microsoft Way
Redmond, WA 98052

Phone: +1 425 706 6605
Fax: +1 425 936 7329
EMail: bernarda@microsoft.com

William Dixon
Microsoft Corporation
One Microsoft Way
Redmond, WA 98052

Phone: +1 882 8080
EMail: wdixon@microsoft.com

Full Copyright Statement

Copyright (C) The Internet Society (2003). All Rights Reserved.

This document and translations of it may be copied and furnished to others, and derivative works that comment on or otherwise explain it or assist in its implementation may be prepared, copied, published and distributed, in whole or in part, without restriction of any kind, provided that the above copyright notice and this paragraph are included on all such copies and derivative works. However, this document itself may not be modified in any way, such as by removing the copyright notice or references to the Internet Society or other Internet organizations, except as needed for the purpose of developing Internet standards in which case the procedures for copyrights defined in the Internet Standards process must be followed, or as required to translate it into languages other than English. The limited permissions granted above are perpetual and will not be revoked by the Internet Society or its successors or assigns. This document and the information contained

herein is provided on an "AS IS" basis and THE INTERNET SOCIETY AND THE INTERNET ENGINEERING TASK FORCE DISCLAIMS ALL WARRANTIES, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO ANY WARRANTY THAT THE USE OF THE INFORMATION HEREIN WILL NOT INFRINGE ANY RIGHTS OR ANY IMPLIED WARRANTIES OF MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE."

Intellectual Property Statement

The IETF takes no position regarding the validity or scope of any intellectual property or other rights that might be claimed to pertain to the implementation or use of the technology described in this document or the extent to which any license under such rights might or might not be available; neither does it represent that it has made any effort to identify any such rights. Information on the IETF's procedures with respect to rights in standards-track and standards-related documentation can be found in [BCP-11](#). Copies of claims of rights made available for publication and any assurances of licenses to be made available, or the result of an attempt made to obtain a general license or permission for the use of such proprietary rights by implementors or users of this specification can be obtained from the IETF Secretariat.

The IETF invites any interested party to bring to its attention any copyrights, patents or patent applications, or other proprietary rights which may cover technology that may be required to practice this standard. Please address the information to the IETF Executive Director.

Expiration Date

This memo is filed as [<draft-ietf-ipsec-nat-reqts-04.txt>](#), and expires September 22, 2003.

