Network Working Group
Internet-Draft
Expires: September 27, 2004

P. Eronen
Nokia
H. Tschofenig
Siemens
March 29, 2004

Simple Mobility and Multihoming Extensions for IKEv2 (SMOBIKE)
draft-eronen-mobike-simple-00.txt

Status of this Memo

Copyright Notice

Abstract

This document describes how existing NAT Traversal functionality could be leveraged to better support mobility and multihoming for IKEv2. The purpose is not to specify a finished solution, but rather to provide input for discussions in the MOBIKE WG. In particular, this draft raises questions to what extent the complexity present in the two other MOBIKE protocol proposals is actually necessary. These questions are not answered in this document, but are to be discussed in the MOBIKE WG.

1. Introduction

IKEv2 NAT Traversal, defined in [4] and [5], allows IPsec to work through NATs. The main functional components of NAT Traversal are the following:

- Presence of NAT is detected during the IKEv2 IKE_SA_INIT exchange using NAT_DETECTION_SOURCE_IP and NET_DETECTION_DESTINATION_IP Notify payloads.

- ESP packets are encapsulated in UDP.

- The peer behind the NAT sends NAT-keepalive packets to keep NAT mappings alive if no normal traffic has been sent for some time.

- The peer that is not behind the NAT dynamically updates the other peer's address to recover from changes in NAT mappings. That is, IKEv2 and ESP packets are sent to the IP address and port from which the last valid authenticated packet from the other end was received.

The last functionality also allows a form of mobility: in typical corporate VPN gateway scenario, the client can move (that is, change its IP address) while keeping the VPN connection up as long as it was behind a NAT when the IKEv2 SA was originally established. This is because from the gateways point of view, a change in the client's IP address is indistinguishable from a change in NAT mappings.

Figure 1 shows the standard VPN scenario where the mobility enhancements of IPsec could be deployed. In this scenario the IKEv2 initiator establishes an IPsec tunnel to the VPN GW. After the tunnel establishment the the IKEv2 initiator changes its IP address. Several reasons could be responsible for this address change, such as interface switching, DHCP lease time expiry, IPv6 privacy extensions, or mobility. In Figure 1 we focus on the mobility case. As a result the SAD entries have to be updated on both nodes (typically via extensions to the PF_KEY interface). No update for the IKE-SA is required since it is not bound to an IP address. A NAT might be along the path between the two IKEv2 peers.

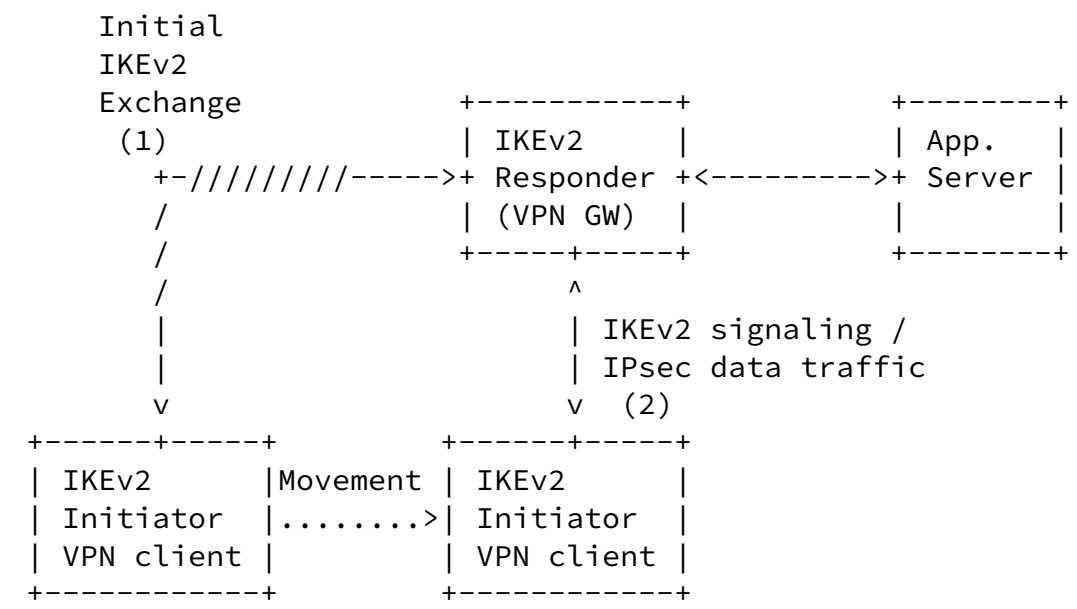

Figure 1: Mobility in VPN scenario

It might be worth noting that mobility and static multi-homing are different with respect to their requirements. This draft is also applicable to static multi-homing to a large extent.

This document specifies how this existing functionality can be leveraged to better support mobility and multihoming. This is done by allowing the use of dynamic address updates and/or UDP encapsulation even when no NATs are detected.

The main purpose of this draft is not to specify a finished solution, but rather to provide input for discussions in the MOBIKE WG. In particular, this draft tries to present a simpler alternative to the two other proposals, [1] and [6], and raises questions to what extent the complexity in the other two proposals is actually necessary. Furthermore, the authors believe that ability to work together with NATs is a required functionality.

2. SMOBIKE Protocol

2.1 Indicating support for SMOBIKE

Implementations that support SMOBIKE indicate this by including a Vendor ID payload in the IKE_SA_INIT exchange (first two messages). The value for this payload is 30DB45C6 AF1CA28C DAC08C30 9CE062C5 (MD5 hash of the string "draft-eronen-mobike-simple").

2.2 Enabling dynamic address updates

In NAT Traversal, the peer that is not behind the NAT dynamically

updates the other peer's address to recover from changes in NAT mappings. That is, IKEv2 and ESP packets are sent to the IP address and port from which the last valid authenticated packet from the other end was received.

By sending the USE_DYNAMIC_ADDRESS_UPDATES Notify payload, a peer can instruct the other peer to dynamically update its address even when no NAT is present. Note that this enables just dynamic address updates, but does not automatically mean UDP encapsulation.

2.3 Changing addresses

The procedure when changing address depends on how UDP encapsulation is used. Note that the same procedure applies to both a mobile host moving to another address, and a multihomed host switching to another interface.

- If the host does not support UDP encapsulation (perhaps it is disabled by configuration), just start using the new address.

- If the host is currently using UDP encapsulation, and wants to keep on using it, just starting using the new address.

- If the host is unsure whether the current setting of UDP encapsulation is the right one for the new address, perform a new NAT detection.

If the current setting of UDP encapsulation is the right one, it is

sufficient to just start using the new address; the other party will update the address when it receives an authenticated packet. If there are no ESP packets to send, the host can send an empty IKEv2 Informational exchange.

In some cases, it may be necessary to either turn on UDP encapsulation (when moving to behind a NAT), or turn it off (when moving back to clear). A host can request a new NAT detection by sending an Informational exchange with NAT_DETECTION_SOURCE_IP and NAT_DETECTION_DESTINATION_IP payloads.

These payloads are processed the same way as in initial IKE_SA_INIT exchange.

2.4 UDP encapsulation without NATs

There are cases when UDP encapsulation is needed even when no NATs are present. A typical example would be a stateful firewall that performs similar filtering as a "symmetric" NAT [8], but does not change the IP addresses (and therefore is not detected by

NAT_DETECTION payloads).

A host can be configured to always use UDP encapsulation, or it can guess that UDP encapsulation might be needed if it does not receive any ESP packets. In some mobile or ad hoc networks UDP encapsulation could be always used since a route change in the network might cause packets to traverse a NAT even without end-host mobility.

The host can then force the use of UDP encapsulation by including a USE_UDP_ENCAPSULATION Notify payload in the same message as NAT_DETECTION payloads.

3. Analysis

This section presents a short analysis of how SMOBIKE handles different mobility and multihoming cases.

1. One or both of the parties are mobile and/or multi-homed. No NATs (or stateful firewalls) are present.

   SMOBIKE works fine, except in some variations of simultaneous

address change (see below). This case is typical in multihoming situations: e.g. a multihomed host talking to a single-homed host, or two multihomed gateways/hosts talking to each other.

The only case where SMOBIKE does not work is if (a) both parties change addresses simultaneously and (b) disable their old addresses before the other party has received the update. In this case, neither of the parties knows the current address of the other party, so they cannot communicate.

2. One party is stationary, single-homed, and not behind a NAT. The other party is be mobile and/or multihomed, and sometimes behind a NAT (or stateful firewall).

   SMOBIKE handles this as well. UDP encapsulation can also be switched off when it is not needed. This case is the typical case of a mobile client talking to a single-homed VPN gateway.

3. Both parties are mobile and/or multihomed, and there is a "full cone" NAT (see [8] for a description of different NAT types) between them.

   This case could be made to work (except for the special case simultaneous address change described above).

4. Both parties are mobile and/or multihomed, and there is a "restricted cone", "port restricted cone" or "symmetric" NAT [8], or a stateful firewall, between them.

   This case does not work. The NAT (or firewall) blocks the address updates sent by the party outside the NAT.

Assuming that simultaneous mobility is not very important, this analysis would seem to indicate that more complex solutions are justified only if they not only handle cases #1-#3, but also improve case #4. Not handling case #2 would in our opinion be a serious deficiency.

SMOBIKE does not support moving traffic to a new address before the

host is able to send packets using the new address as a source address. Other than that, SMOBIKE works for both "break-before-make" (host breaks its old connection before the new connection is fully established) and "make-before-break" (both connections work for a while during mobility) situations.

4. Security Considerations

In the current version, this section lists only the most important points about this protocol.

Just like normal IKEv2 NAT Traversal, SMOBIKE does not send the IP addresses inside IKEv2 payloads, only in the IP header. These addresses are not integrity protection and not authenticated. Protecting these addresses would render the protocol incompatible with NAT Traversal. This problem is already known from other areas such as Mobile IP NAT traversal.

This leads to two possible problems, also known as "transient pseudo-NAT attack" and "third party bombing".

In the "transient pseudo-NAT attack" [3], an attacker intercepts authenticated packets and changes their source IP address (and port). As a consequence the recipient will start using the incorrect peer address, and send IPsec protected data traffic to this address. If the adversary provides an address which is a blackhole then traffic sent to this address will be dropped. This represents a denial of service attack. Obviously, an attacker who can modify packets between the parties could also change e.g. the ICV field, causing the packets to be dropped. Also, the situation is self-fixing: when an unmodified packet gets through, the address is updated back to the correct one.

In "third party bombing" [2], a valid peer redirects its traffic to some third party with the intent of flooding the victim with large

amounts of traffic. For this attack to continue, the attacker has to keep sending traffic with spoofed IP address, because otherwise dead peer detection will fix the situation. Even if the MOBIKE extensions had more complex return routability checks, the attacker could claim not to support them, and use normal IKEv2 NAT Traversal for the attack instead. A regular IKEv2 dead peer exchange will also detect third party bombing.

Both attacks require that the adversary is along the data path. Note that unlike, for instance, in standard Mobile IP, IPsec protects the transmitted payloads from eavesdropping and modification, and thus the consequences of traffic redirection are different.

It might be worth mentioning that a truly secure NAT traversal can be accomplished only if the client can determine which NAT devices are actually authorized to modify the addresses, and communicate with them in secure fashion to determine the current mappings. This requires a protocol for communicating with the NAT, such as NSIS. A detailed discussion of these approaches is far beyond the scope of this document.

5. IANA Considerations

Two new Notify payloads, USE_DYNAMIC_ADDRESS_UPDATES and USE_UDP_ENCAPSULATION.

6. Acknowledgments

This draft obviously borrows many ideas from Tero Kivinen and Francis Dupont, although it ended up looking a bit different than their work.

References

[1] Dupont, F., "Address Management for IKE version 2", draft-dupont-ikev2-addrmgmt-04 (work in progress), February 2004.

[2] Dupont, F., "A note about 3rd party bombing in Mobile IPv6", draft-dupont-mipv6-3bombing-00 (work in progress), February 2004.

[3] Dupont, F. and J. Bernard, "Transient pseudo-NAT attacks or how NATs are even more evil than you believed", draft-dupont-transient-pseudonat-03 (work in progress), February 2004.

[4] Huttunen, A., Swander, B., Volpe, V., DiBurro, L. and M. Stenberg, "UDP Encapsulation of IPsec Packets",

draft-ietf-ipsec-udp-encaps-08 (work in progress), February 2004.

[5] Kaufman, C., "Internet Key Exchange (IKEv2) Protocol", draft-ietf-ipsec-ikev2-13 (work in progress), March 2004.

[6] Kivinen, T., "MOBIKE protocol", draft-kivinen-mobike-protocol-00 (work in progress), February 2004.

[7] Kivinen, T., "Design of the MOBIKE protocol", draft-kivinen-mobike-design-00 (work in progress), February 2004.

[8] 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.

Authors' Addresses

Pasi Eronen
Nokia Research Center
P.O. Box 407
FIN-00045 Nokia Group
Finland

EMail: pasi.eronen@nokia.com

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

EMail: Hannes.Tschofenig@siemens.com

Intellectual Property Statement

Disclaimer of Validity

Copyright Statement

Acknowledgment

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