Mobile VPN Schemes: Technical Analysis and Experiments

Daouda Ahmat, Mahamat Barka, Damien Magoni

Mobile VPN Schemes: Technical Analysis and Experiments

Daouda Ahmat[1,2], Mahamat Barka[2], and Damien Magoni[3]

[1] Virtual University of Chad, Chad
daouda.ahmat@gmail.com
[2] University of N'Djamena, Chad
mahamat.barka@gmail.com
[3] University of Bordeaux – LaBRI, France
magoni@labri.fr

Abstract. A new class of Virtual Private Networks (VPN), which supports both security and mobility, has recently emerged. Called mobile VPN, these systems provide not only secure tunnels but also session continuity mechanisms despite location change or connection disruptions. These mechanisms enable secure sessions to survive in dynamic/mobile environments without requiring a renegotiation of security keys during the session resumption phase. In this paper, we compare four open-source mobile VPNs in terms of functionality and performance.

Key words: mobile VPN, resilient session, seamless resumption.

1 Introduction

A Virtual Private Network (VPN) provides increased security between two remote entities that exchange data through untrusted networks such as the Internet [1]. VPN systems prevent against various attacks such eavesdropping or replay. However, traditional VPNs fail to support the mobility of users. Indeed, network failures automatically break-up secure tunnels and involve a subsequent renegotiation that is then necessary to reestablish broken tunnels. Such a negotiation involves expensive computational operations in order to restore tunnels as well as transport and application layers connections. This phase of negotiation causes not only significant latency but also presents risks of Man-In-The-Middle attacks. Traditional VPNs can not therefore effectively operate in dynamic and/or mobile environments.

Mobile devices and dynamic environments become pervasive in the Internet. However, for instance traditional VPN infrastructures do not support session continuity result from location change or network reconfiguration. After each connection disruption, key renegotiation process is needed to restore broken tunnel. In order to address network failures resulting from location changes or network reconfigurations, several solutions were proposed in the literature [2–7]
.

2 Mobile VPN Technologies

Up to now, several mobile VPN solutions have been proposed in various research papers. In this section, we describe some leading examples of mobile VPN systems and technical concepts in this area proposed in litterature.

2.1 N2N

In opposition to most dynamic VPN systems, these systems have the advantage to be fairly scalable and to have ability to communicate across NAT and firewalls. Decentralized P2P VPN are flexible and self-organizing infrastructures that enable users to create their own secure networks upon an untrusted network. A layer 2 peer-to-peer VPN (see figure 1), called N2N [2], and ELA [8] topologies are very similar despite the fact that N2N is based on the OSI layer 2 whereas ELA is based on the OSI layer 3. However the use of super nodes in N2N limits its full scalability as these nodes have a more important role than the other nodes and thus they can weaken the overall strength of the N2N network and may even break its connectivity if they fail.

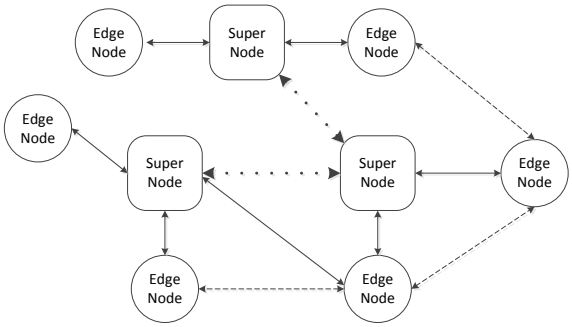

Fig. 1. Example of N2N topology.

Freelan [9] is a multi-platform and open-source peer-to-peer VPN that abstracts a LAN over the Internet. Based over the UDP protocol, the FreeLAN Secure Channel Protocol (FSCP) is designed to be secure and efficient, and it tries to reduce the network overhead. In addition, Freelan systems can be configured to act according to a client/server, peer-to-peer or hybrid model whichever suits best.

2.2 IPSec + Mobile IP

Mobile VPN systems based on both IPsec [10] and Mobile IP [11] have been proposed several times such as in [12], [7], [13], [14] and [15], in order to attempt to overcome the inherent mobility drawbacks of traditional VPNs. Nevertheless,

as explained in [16], many problems arise from the combination between IPsec and MobileIP. In order to overcome these problems, a model has been proposed which is based on the use of two HAs (Home Agents) - internal HA and external HA - and two FAs (Foreign Agents) - internal FA and external FA - by Vaarala *et al.* in [17]. However, this model imposes the use of three imbricated tunnels({x-MIP{GW{i-MIP{original packet}}}}), as shown in figure 2.

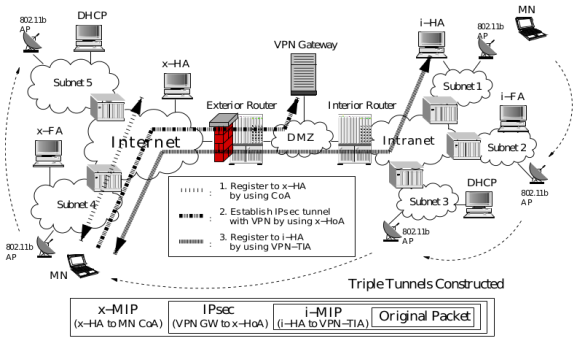

Fig. 2. IETF Mobile VPN (source: IETF).

In addition, a IPsec-based mobile VPN requires n tunnels (n security layers) when there are n IPsec hops between the source and destination entities. Therefore, the imbricated tunnels in such VPN systems have a negative impact on their network performances (i.e., throughput, overhead, etc).

In order to address IPsec mobility inherent issues, several improved schemes based on the IPsec architecture have been proposed by Eronen *et al.* in [4], [18], or [19]. Based on security extensions to MOBIKE [4], the solution described in [19] combines secure connectivity and Mobile IPv4. This approach resolves considerably the issues notified in [16] such as overhead, NAT traversal or mobility problems due to the combination of IPsec and Mobile IPv4. These solutions are however not free of scalability issues and network overhead that they inherit from IPsec and Mobile IP.

Based upon the NEMO architecture [20], the mobile VPN scheme presented in [21] provides secure connectivity between vehicles for public transportation. In other words, this model provides secure vehicle to vehicle (V2V) communications as well as secured communications between passengers in the same (or in a different) vehicle. As the above mobile VPN solutions, this current model is designed to use the best properties of MOBIKE and Mobile IP.

The dynamic VPN approach proposed in [22] enables to use alternately IPsec in Full-Mesh mode or in Hub mode with a centralized IPsec Gateway. The first mode is only used when routing problems occur. This architecture extends MOBIKE in order to support dynamic tunnels. However, this model is not designed to support mobility.

Another proposal leveraging MOBIKE is presented by Migault in [23], where they propose an alternative End-to-End security (E2E) architecture based on their own MOBIKEX protocol, which extends the MOBIKE mobility and multihoming features to multiple interfaces and to the transport mode of IPsec. Based on a topology organized in communities, peer-to-peer (P2P) mobile VPN systems have also been proposed such as ELA [8] or N2N [24].

2.3 HIP-based Mobile VPN

The Host Identity Protocol (HIP) [3, 6] is an architecture that provides both mobility and multihoming services. HIP introduces a new name space that enables the separation between the host identity, called Host Identity Tag or HIT, and the host location, as shwon in figure 3. Each HIP host is uniquely identified by the public key of its public/private key pair. When a mobile node changes its point of network attachment, its IP address is then changed and the new IP address will be communicated to its correspondent hosts. However, in addition to remaining at an experiment stage for some years, HIP introduces a new layer between the transport and the network layers in the OSI stack. This implies that the host's operating system must be modified in order to use HIP although a user-space implementation does exist.

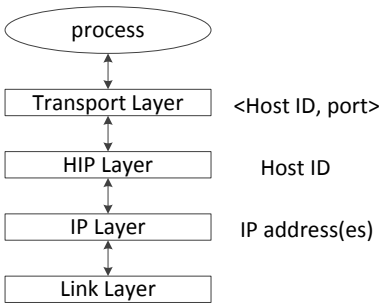

Fig. 3. HIP layer within the TCP/IP stack.

In order to provide both security and mobility, HIP has been extended in two subsequent proposals: Hi3 [25] and SPEAR [26]. Previously known as P2P SIP-over-HIP (p2pship), SPEAR was originally designed for SIP-based communication applications (i.e., SIP proxy), allowing users to make peer-to-peer voice / video calls, without the help of a centralized SIP infrastructure. It now supports various protocols and applications. HIP is used as data transport, making the connections secure and enabling features such as mobility and multihoming.

3 Evaluation

In this section, we present a functional analysis as well as the experiment environment and the results of the evaluation of our approach MUSeS [5] and three state-of-art solutions, namely: N2N, HIP and MOBIKE.

3.1 Functional Analysis

Table 1 describes the technical comparison between MOBIKE, N2N, HIP and MUSeS. Indeed, all these systems are based on UDP to exchange information in both handshake and re-handshake steps. The MUSeS middleware has the smallest number of exchanged packets for these two phases of communication. While both HIP and MUSeS proceed through direct connection, MOBIKE is based on indirect secure connection (through an IPSec gateway) and N2N is based on triangular negotiation in both handshake and re-handshake phases. For the two first systems, mobility is limited. This means that only *N2N Edge Node* and *MOBIKE client* can be really mobile.

Table 1. Comparison of the evaluated mobile VPN systems.

Mobile VPN system	MOBIKE	N2N	HIP	MUSeS
Handshake packets	8	3	3	2
Re-handshake packets	6	3	3	2
Secure connection mode	indirect [1]	triangular [2]	direct	direct
Mobility supported by systems	limited [3]	limited [c]	√	√
Implementations	StrongSwan [27]	n2n [2]	OpenHIP [28]	MUSeS [29]

Security Analysis: A Mobile VPN enables, in one hand, to secure communication and to keep open application sessions during location change. In other hand, mobile VPN is free to session key renegotiation in the resumption phase. These two technical properties are needed in secure mobile environments. Despite their interesting properties, the systems that operate in autonomous and mobile environments are constantly subject to some security challenges such as DoS and replay attacks.

To prevent replay attacks, MUSeS packets are built by adding sequence number to their headers. In other words, each packet is separately identified by its sequence number added to its header. Thus, when a packet is replayed, it will be automatically detected and subsequently it will be destroyed.

In the resumption phase, a Re-hello is generated and then sent in order to restore interrupted session without using session key renegotiation mechanism. However, Re-hello packet could be replayed because it does not contain a sequence number in order to detect replay attack. Thus, a malicious user that has infiltrated the network could then send a succession of Re-hello packets with the aim of perpetrating Denial-of-Service (DoS) attacks. In addition, the receiver

peer cannot determine which oacket is the last one received among other received packets, otherwise this problem could be solved easily. In concrete terms, on receiving Re-hello, the receiver peer processes it in order to resolve and, the challenge and before finishing, it receives another, again another, etc. Finally, the target peer will be saturated by a flooding of Re-hello requests. Furthermore, HIP could be vulnerable to DoS attacks in the resumption phase as shown in the paper analyzing HIP protocol security [30].

To address this security issue, MUSeS assigns a timestamp when sending to each Re-hello packet in order to recognize the freshest request among received requests. In this way, the MUSeS system tries to prevent DoS attacks that use an uninterrupted sequence of Re-hello packets. Due to their mobility, flexibility and autonomy, P2P-based VPN systems are unfortunately not totally invulnerable to intrusion of malicious users. Indeed, in fully decentralized P2P networks, each peer can join and leave the network at any time and usually without any authentication. In our system, authentication is guaranteed by using challenge messages.

Typically, to authenticate a peer over the network, a node encrypts a random challenge message and sends it to its corresponding peer. On receiving this message, the corresponding peer decrypts it and sends the same message to the initiator peer. Thus, the initiator peer ascertains the identity of the corresponding peer.

The HIP protocol is designed to be resistant to Denial of Service (DoS) and Man in the Middle (MitM) attacks, and when used with ESP enabled, it provides DoS and MitM protection to upper layer protocols, such as TCP and UDP.

In N2N and MOBIKE however, there can be no secure tunnels without a N2N-Super-Node or a MOBIKE-GW. In other words, when N2N-Super-Nodes and MOBIKE-GWs are unavailable, any secure communication is then impossible. The HIP protocol and MUSeS do not suffer from these impairments.

Although MUSeS offers a solid security mechanism in remote communication between two peers, there is, however, a security weak point in local applicative connections. Unlike communication between two MUSeS peers, local communication between MUSeS and applications is not secured. This means that an unauthorized user application launched by a malicious user should establish connection with a remote honest MUSeS or eavesdrop exchanged traffic between MUSeS and local applications. This untrusted communication should cause security issues.

On the one hand, to prevent external malicious processes to connect to MUSeS, only local applications are authorized to connect to MUSeS by loopback address. On the other hand, only the root user can catch, by using *tcpdump* or *wireshark*, local traffic passed through from local applications to the MUSeS middleware and conversely. Therefore, plain text data exchanged between local applications and MUSeS are protected.

Mobility Analysis: MOBIKE and N2N are both based upon permanent virtual addresses in order to identify separately mobile nodes. However, when N2N-Super-Node and MOBIKE-Gateway (MOBIKE-GW) change their network

points of attachment, any mobility would be possible. Thus, these systems have limited mobility. Indeed, MOBIKE authorizes only the mobility for initiators. However, in addition to mobility, MOBIKE supports also multi-homing for initiators. This means that MOBIKE mobile nodes can have several network interfaces and use them in order to support network link breakdown. In opposition to MOBIKE, all two endpoints of a N2N tunnel keep up mobility.

HIP introduces an interesting scheme of mobility and multi-addressing over IPv4 and IPv6 networks and it is designed to work in a NAT-less environment. Indeed, the HIP hosts do not change identities during location changes; this implies network addresses changes. Each HIP host is identified by its public key that is self-certified, called *Host Identity* (HI). Thus, when a mobile node changes its IP address, it notifies its currently active peers by sending a control packet containing its new location. When correspondent peers change simultaneous their location, the previous notifying method fails and a deadlock will occur. However, HIP introduces a *rendezvous* mechanism in order to address this simultaneous mobility issue. Unlike previous mobility methods, MUSeS proposes a new mobility scheme based on identifiers provided by a DHT infrastructure [31].

Each MUSeS host is identified separately by a name and an address defined as coordinates taken from the hyperbolic plane.

An *Interruption Detection* mechanism is introduced by MUSeS to detect failures and to subsequently activate the SRM module. SRM is based on keepalive messages which are periodically sent. Thus, when network failures occur within lower layers, communication will be temporarily interrupted and failures will be confined within SRM and hidden to higher layers. Due to these properties, mobility is transparent to both user applications running over MUSeS middleware and all the other MUSeS modules, except the SRM component. Therefore, loopback connections established between MUSeS and local applications survive to networks failures despite network attachment point change events, for instance.

3.2 Performance Analysis

In order to assess those four VPN technologies in a mobility scenario, we have used a tool called Network Emulator For Mobile Universes (NEmu), developed by Vincent Autefage and presented in [32].

An experiment has been carried out with the above implementation in a dynamic environment composed of one mobile node. A mobile node inside this environment has the ability to leave one network (one virtual router) in order to join another one (another virtual router). This event causes a network failure during the move until a possible subsequent reconnection. This disruption is transparent for the application and it does not prevent the MUSeS system from continuing to run despite the fact that the mobile node is disconnected for a moment. Technically, in our experimentation, the node mobility consists in causing an artificial failure on a virtual network interface. We disconnect a virtual wire from a virtual switch and reconnect it on another virtual switch. The MUSeS system hides this network change not only to the user's application but also to the remote corresponding node.

Figures 4 to 7 show the evolution of the throughput between the two corresponding applications over time. For all systems, the network interruption happens at the 40^{th} second after the start of the experiment and the throughput instantly drops to zero in the time intervals $[40s\ ;\ 60s[$. This means that the disruption duration is 20 seconds. The connectivity is reestablished at the network and CLOAK level at the 60^{th} second. However, due to latency, the throughput remains at zero after the 60^{th} second until the effective recovery. This latency varies from one system to another. Indeed, whereas MUSeS middleware has a latency of 3 seconds (see Figure 7), MOBIKE, N2N and HIP protocols have respectively latencies of 12 seconds (see Figure 6), 51 seconds (see Figure 4) and 13 seconds (see Figure 5).

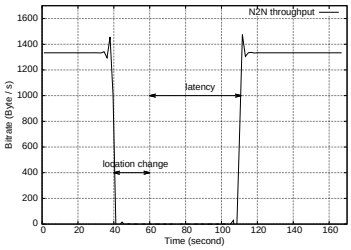

Fig. 4. N2N resumption latency.

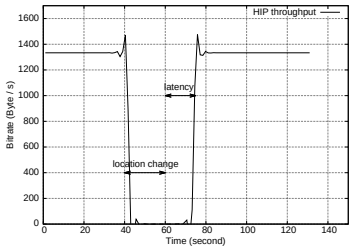

Fig. 5. HIP resumption latency.

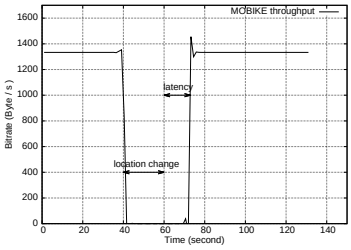

Fig. 6. MOBIKE resumption latency.

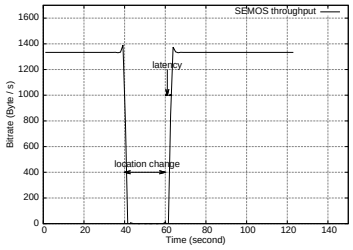

Fig. 7. MUSeS resumption latency.

4 Conclusion

Disrupted networks, due to both poor-quality devices and technical skills deficiency, are pervasive in developing countries, particularly in Africa. In theses areas, secured resilient sessions are needed to overcome both security and performance issues inherent to connection disruptions. In this paper, we have presented four open-source mobile VPN solutions, have provided a detailed technical analysis of those systems and have compared them in terms of functionality and

performance. Results show that MUSeS is a competitive solution for providing secure and mobile communications.

References

[1] T. Berger. Analysis of current vpn technologies. In *The First Int'l Conf. on Availability, Reliability and Security, ARES 2006.*, page 8 pp., 2006.

[2] Luca Deri and Richard Andrews. N2N. `http://www.ntop.org/products/n2n/`.

[3] R. Moskowitz and P. Nikander. Host identity protocol (hip) architecture. IETF RFC 4423, 2006.

[4] P. Eronen. IKEv2 Mobility and Multihoming Protocol (MOBIKE). IETF RFC 4555, 2006.

[5] Daouda Ahmat and Damien Magoni. Muses: Mobile user secured session. In *5th IFIP Wireless Days Int'l Conf.*, Dublin, Irland.

[6] Andrei Gurtov. *Host Identity Protocol (HIP): Towards the Secure Mobile Internet*. Wiley, 2008.

[7] Jim Binkley. An Integrated IPsec and Mobile-IP for FreeBSD. *Technical Report*, pages 01–10, 2001.

[8] S. Aoyagi, M. Takizawa, M. Saito, H. Aida, and H. Tokuda. ELA: A Fully Distributed VPN System over Peer-to-Peer Network. *Symp. on Applications and the Internet*, pages 89–92, 2005.

[9] J. Kauffmann. The freelan secure channel protocol. `https://github.com/ereOn/libfscp/blob/1.0/fscp.txt`, 2011.

[10] S. Kent and K. Seo. Security Architecture for the Internet Protocol. IETF RFC 4301, 2005.

[11] C. Perkins. IP Mobility Support for IPv4. IETF RFC 3344, 2002.

[12] Motorolla. Mobile VPN, Secure Connectivity on the Move. *White paper*, 2008.

[13] T. Braun and M. Danzeisen. Secure mobile IP communication. *26th IEEE Conf. on Local Computer Networks*, pages 586–593, 2001.

[14] Heesook Choi, Hui Song, Guohong Cao, and T. La Porta. Mobile multi-layered ipsec. *24th Joint Conf. of the IEEE Computer and Communications Societies*, pages 1929–1939, 2005.

[15] R. Ruppelt, A. Pelinescu, C. Constantin, J. Floroiu, D. Sisalem, and B. Butscher. Building ALL-IP Based Virtual Private Networks in Mobile Environment. *Int. Work. on Informatic and Mobile communication over wireless LAN: Research and applications*, 2001.

[16] F. Adrangi and H. Levkowetz. Problem Statement: Mobile IPv4 Traversal of Virtual Private Network Gateways. IETF RFC 4093, 2005.

[17] S. Vaarala and E. Klovning. Mobile IPv4 Traversal across IPsec-Based VPN Gateways. IETF RFC 5265, 2008.

[18] V. Devarapalli and P. Eronen. Secure Connectivity and Mobility Using Mobile IPv4 and IKEv2 Mobility and Multihoming (MOBIKE). IETF RFC 5266, 2008.

[19] M.M. Karbasioun, M. Berenjkub, and B. Taji. Securing mobile IP communications using MOBIKE protocol. *Int'l Conf. on Telecommunications*, 2008.

[20] V. Devarapalli, R. Wakikawa, A. Petrescu, and P. Thubert. Network Mobility (NEMO) Basic Support Protocol. IETF RFC 3963, 2005.

[21] A. Petrescu and A. Olivereau. Mobile VPN and V2V NEMO for public transportation. *9th Int'l Conf. on Intelligent Transport Systems Telecommunications*, pages 63–68, 2009.

[22] K. Ishimura, T. Tamura, S. Mizuno, H. Sato, and T. Motono. Dynamic IP-VPN architecture with secure IPsec tunnels. *Symp. on Information and Telecommunication Technologies*, 2010.

[23] D. Migault, D. Palomares, E. Herbert, Wei You, G. Ganne, G. Arfaoui, and M. Laurent. E2e: An optimized ipsec architecture for secure and fast offload. In *Int'l Conf. on Availability, Reliability and Security*, pages 365–374, 2012.

[24] L. Deri and R. Andrews. N2N: A Layer Two Peer-to-Peer VPN. *Int'l Conf. on Autonomous Infrastructure, Management and Security*, pages 53–64, 2008.

[25] Andrei Gurtov, Dmitry Korzun, Andrey Lukyanenko, and Pekka Nikander. Hi3: An efficient and secure networking architecture for mobile hosts. *Comput. Commun.*, 31(10):2457–2467, 2008.

[26] Jookos and Agur. A secure peer-to-peer services overlay architecture, 2010.

[27] Andreas Steffen. StrongSwan. `http://www.strongswan.org`.

[28] Tom Henderson. OpenHIP. `http://www.openhip.org`.

[29] Daouda Ahmat. SEcure MObile Session. `http://www.labri.fr/~magoni/cape/`.

[30] Tuomas Aura, Aarthi Nagarajan, and Andrei Gurtov. Analysis of the hip base exchange protocol. In *In 10th Australasian Conf. on Information Security and Privacy (ACISP 2005)*, pages 481–494, 2005.

[31] T. Tiendrebeogo, D. Ahmat, D. Magoni, and O. Sié. Virtual connections in p2p overlays with dht-based name to address resolution. *Int'l Journal on Advances in Internet Technology*, 5(1):11–25, 2012.

[32] V. Autefage and D. Magoni. Network emulator: a network virtualization testbed for overlay experimentations. In *17th IEEE Int'l Work. on Computer-Aided Modeling Analysis and Design of Communication Links and Networks*, pages 38–42, 2012.