VoIP performance with IPsec in IPv4-IPv6 transition networks

ROMAN YASINOVSKYY, ALEXANDER L. WIJESINHA, RAMESH KARNE

Towson University, Maryland, USA
{ryasinovskyy, awijesinha, rkarne}@towson.edu

Keywords: VoIP, IPsec, IPv6, Teredo, 6to4

We conduct experiments in a LAN environment to determine the impact of IPsec on VoIP performance during the IPv4 to IPv6 transition period. VoIP performance with IPsec is measured in the presence of varying background traffic with IPv4, IPv6 and 6to4. To study the effect of NAT traversal, we also use Teredo. The IPsec scenarios that are evaluated include no-security (in which traffic bypasses IPsec), network-to-network (in which traffic is tunneled between IPsec gateways), and client-to-network (in which traffic is tunneled between the client and IPsec gateway). We use the popular Openswan implementation of IPsec and focus on ESP with the authentication option. The measures used for evaluating VoIP performance are delta (packet inter-arrival time), jitter, packet loss, throughput, and Mean Opinion Score (MOS). We also determine the time for the IPsec key exchange and call set up using SIP. Our results indicate that VoIP performance with IPsec in IPv4-IPv6 transition networks is not significantly different from that in today's IPv4 networks.
In particular, we find that 1) performance with IPv4, IPv6, or 6to4 is similar; 2) the overhead due to NAT traversal with Teredo is comparable to that when using NAT with 6to4 on the edge device; 3) performance degrades significantly when the amount of background traffic exceeds network capacity regardless of whether IPv4, IPv6, 6to4, or Teredo is used.

1. Introduction

VoIP continues to grow in popularity due to its low cost and convenience. While VoIP is primarily used today over IPv4 networks, VoIP calls in the next-generation Internet are likely to be placed between devices on IPv6 networks. However, the IPv4 to IPv6 transition is expected to last for several years due to the vast base of installed IPv4 networks. During this period, communication between many IPv6 networks will only be possible using existing IPv4 connectivity. The recommended interim measure to address this issue is 6to4 encapsulation [1]. 6to4 enables IPv6 traffic to be carried over IPv4 transit networks with minimal changes to the current infrastructure. Unfortunately, most IPv4 networks employ NAT, and 6to4 does not work with NAT unless the NAT box also serves as a 6to4 router. The situation is more complex if multiple NATs are traversed.

While Teredo [2,3] offers a solution to the NAT traversal problem, it requires additional infrastructure including servers and relays, as well as Teredo-aware clients. Teredo is proposed and supported by Microsoft, with Vista and Windows7 having Teredo enabled by default; there is also a Linux implementation [4]. Thus, a study of VoIP performance in IPv4-IPv6 transition networks needs to consider 6to4 as well as Teredo.

In this paper, we conduct experiments in a test LAN to evaluate the impact of IPsec with 6to4 and Teredo on VoIP performance. Although other approaches to VoIP security exist, IPsec VPNs were used in our study since they are frequently used to protect all IP traffic in IPv4 networks. The basic question we address in this research is the following. To what extent does the overhead added by an IPsec VPN impact VoIP performance during the IPv4 to IPv6 transition period? To assess this impact, we evaluate VoIP performance in an IPsec VPN over IPv4, IPv6, and 6to4, and when Teredo is used for NAT traversal. We also measure the time to complete the IPsec key exchange and for call set up using SIP.

The experiments with IPv4, IPv6 and 6to4 use a test LAN with 6to4, and IPsec gateways that allow VoIP quality to be evaluated using three IPsec scenarios: no-security in which traffic bypasses IPsec; network-to-network in which traffic is tunneled between IPsec gateways; and client-to-network in which traffic is tunneled between the client and IPsec gateway. All IPsec experiments used ESP tunnel mode with the authentication option [5]. To evaluate the effect of NAT on VoIP performance with IPsec, we configured a 6to4 gateway to serve as a NAT box, and also set up a server enabling Teredo to be used for NAT traversal by the clients.

In our experiments, VoIP traffic is transmitted through Linux routers on the test LAN together with data traffic at various rates. Congestion is introduced by using a 100 Mbps transit network to carry traffic from a gigabit Ethernet. VoIP performance is then studied by measuring values of delta (packet inter-arrival time), jitter, packet loss, and throughput using Wireshark. Finally, voice quality is estimated by computing the MOS (Mean Opinion Score).

The values of delta (packet inter-arrival time) reflect delay in the network, but do not estimate the actual end-to-end delay. Auxiliary measurements we conducted to estimate the end-to-end delay in our network showed

that it is well within the commonly accepted 150 ms limit except when the network is unstable at very high loads. Also, since Wireshark was run on a separate machine and not on the softphones, the measured values do not consider jitter buffer effects, and decoding and decryption delays at the receiver prior to playback. Thus, it is possible that the values of the reported measures might not represent the actual voice quality experienced by the receiver. To address this issue, we computed an average MOS based on MOS values assigned by human listeners. Since this average MOS correlated well with the MOS by using values measured by Wireshark, we believe that the measured values accurately reflect actual call quality.

The main contributions of this paper are results demonstrating that 1) the popular IPsec-based VPN technology used in IPv4 networks today can continue to be used during the IPv4 to IPv6 transition period with no significant impact on VoIP quality; 2) the additional overhead due to 6to4 or Teredo processing has a negligible effect on VoIP quality with IPsec if the network capacity is not exceeded. This paper is an extended version of [6]. The differences are the inclusion of VoIP performance measurements with IPsec when Teredo is used for NAT traversal; and the determination of delays for VPN establishment, and for user registration and call set up via SIP.

The rest of this paper is as follows: In Section 2, we briefly discuss related work. In Section 3, we describe the test network, and in Section 4, we present the results. In Section 5, we present the conclusion.

2. Related work

In a previous study on IPsec with IPv6 using real traffic [7], hosts with an Intel Pentium II 450 MHz processor and 128 MB memory running Free BSD 2.2.8, and routers with an Intel Pentium III 500 MHz processor were used. Their study compared the end-to-end throughput for IPv4 and IPv6 without IPsec, with only AH, with only ESP, and with both AH and ESP. The application used for the study was digital video. The experiments showed that for large amounts of data, the use of authentication and encryption reduces the throughput by 1/9. In this case, the throughput was about 10 Mbps for UDP and 6 Mbps for TCP. Their study demonstrated the feasibility of securely transmitting video using IPsec over IPv6 with ordinary hardware. However, their study does not apply to VoIP and it did not specifically consider IPsec scenarios that are common to today's VPNs using modern implementations on Linux systems that are popular today.

The overheads of an IPSec VPN server with IPv4, and performance improvements are studied in [8,9]. The studies use Openswan, were mainly concerned with the overhead due to the IKE/ISAKMP key exchange, and show that it is much larger than the ESP overhead. In [10], performance of voice and video in an IPsec VPN for videoconferencing is analyzed and it is concluded that the VPN cannot meet QoS requirements under heavy loads. Studies have also examined the IPsec overhead with IPv4 for email and Web applications [11], and Web servers with IPv4 and IPv6 [12]. The performance of 6to4 without IPsec for TCP traffic is evaluated in [13] and it is found that the additional overhead due to tunneling is minimal.

An evaluation of IPsec with 6to4 is done in [14], but the study does not address VoIP performance. In [15], the authors describe the implementation of an IPsec VPN using IPv6, discuss the tradeoffs, and perform testing. VoIP performance over IPv6 and IPv4 without IPsec or 6to4 is compared in [16], and it is shown that the difference in VoIP call quality due to the different IP versions is negligible. In [17], the impact of IPv6 on SIP is studied considering both 6to4 and Teredo. Our study is similar, but also takes varying levels of background traffic into account. The focus in [18] is on comparing 3G UMTS network performance over IPv6 with IPv4 and tunneled IPv6 for multimedia systems; it does not deal with VoIP performance over IPv6 with Teredo or 6to4.

The main difference between this study and the previous studies is that we focus on VoIP performance with IPsec in IPv4-IPv6 transition networks. To this end, we study VoIP performance in IPsec VPNs over IPv4, IPv6 and 6to4, and by using Teredo for NAT traversal. VoIP performance is measured by making calls using softphones and sending the VoIP traffic and variable amounts of other UDP data traffic through a LAN with several routers.

When VoIP traffic passes through a VPN tunnel, all IP payloads carrying the voice traffic (including UDP and RTP headers), the ESP trailer, and the message authentication code are encrypted. These fields (excluding the authentication code field) plus the ESP header can also be authenticated. Furthermore, the inner IP header is also encrypted and could optionally be authenticated. However, since there is no IPsec protection within the network sites, a protocol such as SRTP [19] would be needed for end-to-end VoIP security.

3. Network and experimental setup

Fig. 1 shows the test LAN for the network-to-network IPsec scenario. In this scenario, router #1 and router #4 act as both 6to4 and IPsec gateways. Router #4 can also serve as a NAT box. The test LAN for the client-to-network scenario is the same except that IPsec is enabled at client #1 instead of router #1. To test Teredo, NAT is enabled on router #1 and router #4, and a Teredo server is deployed on the network between router #2 and router #3.

Calls using Linphones [20] (softphones) are made between the two clients. We use Linphones for consistency and convenience as they exhibited stable behavior and were easy to configure with either IP version. MGEN [21] is used to generate UDP background traffic. One cli-

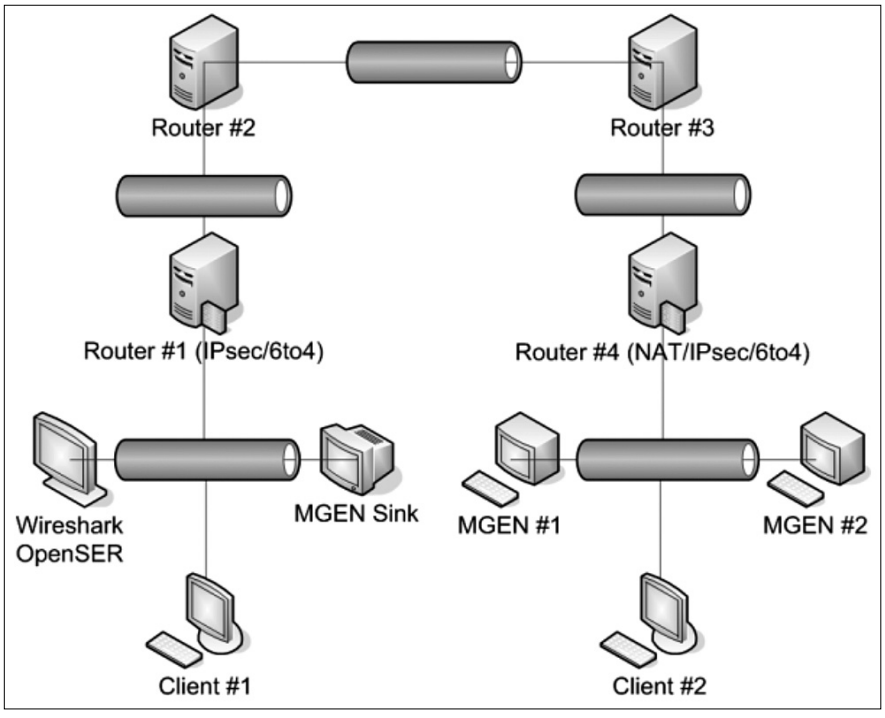
ent (client #2) and the traffic generators (MGEN#1 and MGEN #2) are located on a gigabit Ethernet. During a call, the VoIP data consisting of 20 ms voice packets generated by a Linphone is collected for 2 minutes (i.e., 2-minute conversations) and the results for the second minute only are used (to eliminate any startup effects).

We use the Openswan implementation of IPsec with IKEv1 [22] since it is presently used in most VPNs. The newer Strongswan implementation with IKEv2 [23] addresses several security issues with IKEv1 but has yet to be widely deployed.

In the network-to-network scenario for example, the background traffic (MGEN UDP data at rates of 50, 100, 150 and 200 Mbps) first passes through the 6to4 router #4. It then becomes IPsec ESP traffic and passes through three 100 Mbps networks connected by 2 routers (router #3 and router #2 respectively) before entering the destination network via router #1 that acts as an IPsec/6to4 endpoint gateway as shown in the figure. Although packet loss is possible in this network, packets cannot arrive out of order. When IPsec processing is necessary, it is always done first. For IPv6 packets, this is followed by 6to4 processing.

ESP encryption and authentication are then applied by router #4 to IPv6 packets carrying the VoIP data with the addition of an ESP header and an outer IPsec header (tunnel mode). The resulting packets are then prefixed with an IPv4 header (6to4 encapsulation) and forwarded to the destination through the intermediate IPv4 networks and routers. At the destination network, router #1 decapsulates the received 6to4 packet, and does IPsec authentication and decryption before forwarding the IPv6 voice and data traffic to their respective destinations.

Figure 1.
Test LAN with IPSec/6to4 the case of a site-to-site VPN



Wireshark [24] running on the destination network captures the voice traffic delivered to the client by port mirroring at the switch and reports values of delta, jitter, packet loss, and throughput that are used to measure VoIP performance. The machine running Wireshark doubles as an OpenSER SIP server [25] for setting up the calls.

The specifics of hardware, software and MGEN traffic used for the experiments are as follows:

Hardware: Router/Server/MGEN: Dell Optiplex GX260 (Pentium 4, 2.4 GHz, 512 Mb RAM, Intel PRO/1000, 3Com 10/100); Client: Dell Optiplex GX270 (Pentium 4, 2.4 GHz, 2048 Mb RAM, 3Com 10/100); Switches: Cisco Catalyst 2950, Netgear GS108 (1000), Netgear FS308 (10/100), Trendnet TE100-S55E (10/100).

Software: CentOS 5 (2.6.18-92.1.22) (Routers, SIP Server, NTP Server, Wireshark), Windows XP (SP3) (Generators + Sink), Fedora 10 (2.6.27.9-159) (Clients), Linphone 2.1.1-1 (ITU-G.711 codec), Wireshark 1.0.3, MGEN 4.2b4, OpenSER 1.3.4-1, Openswan 2.6.14-1.

MGEN Background Traffic: n streams are used to generate 5n Mbps of background traffic, where n=10, 20, 30, 40.

4. Results

Each experiment is run several times and the results shown are averages over three runs.

A) Delta (Packet Inter-arrival Time)

We consider the maximum (max) and mean values of delta (shown in *Fig. 2*) and its relative frequency distribution.

In general, it is not possible to directly relate the value of delta to the actual delay.

Max Delta: When there is no background traffic, there is no packet loss and max delta is about 40 ms for all four IPsec scenarios with either IP version or 6to4 encapsulation. When there is 50 Mbps of background traffic, there is still no packet loss, and max delta increases slightly, but again the differences due to IPsec scenario, IP version and 6to4 encapsulation are insignificant.

When background traffic is at 100 or 150 Mbps, larger increases in max delta are seen but the values are not significantly different for the no-security, and client-to-network scenarios with either IP version or 6to4 encapsulation. When background traffic is increased to 200 Mbps, delta for some packets exceeds 100 ms for no-security with 6to4 and client-to-network with either IP version.

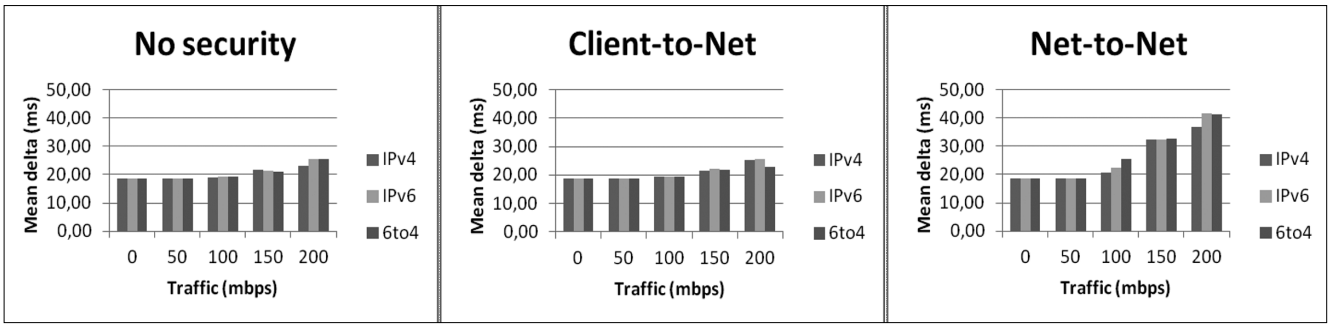


Figure 2. Mean delta

For the network-to-network scenario, it is possible for some packets to have extremely large delta values at 100 Mbps with 6to4, at 150 and 200 Mbps with IPv4, and at 200 Mbps with IPv6 (max delta exceeds 800 ms in these cases). These large delta values for the network-to-network scenario at high background traffic rates are due to packet loss and increased delays.

Mean Delta: In contrast to max delta, the values of mean delta are stable. For the no-security and client-to-network scenarios, there is very little difference in mean delta values with either IP version or 6to4 encapsulation, and it varies from 19-26 ms (the mean is 26 ms at 200 Mbps for no-security with IPv6 and 6to4, and for client-to-net with IPv4 and IPv6). For the network-to-network scenario at background traffic rates of 100 Mbps or less, mean delta values are similar to those for the other scenarios and there is at most a small difference in values with either IP version or 6to4 encapsulation.

At a background traffic rate of 150 Mbps, mean delta values are 33 ms, and at 200 Mbps it is 37 ms with IPv4 and 42 ms with IPv6 and 6to4. For the no-security and client-to-network cases, the standard deviation of delta varies from 8-16 ms when the background traffic is increased from 0-200 Mbps, and there is little difference in the standard deviation with either IP version or 6to4 encapsulation. For the network-to-network scenario, there is more variability in the delta values: at 0 and 50 Mbps, the standard deviation is 8 ms with either IP version or 6to4 encapsulation; at 100 Mbps it is 11 ms with IPv4 and IPv6, and 24 ms with 6to4; at 150 Mbps it is 21 ms with IPv6 and 6to 4 and 29 ms with IPv4; and at 200 Mbps it is 30 ms for 6to4 and approximately 40 ms for IPv4 and IPv6.

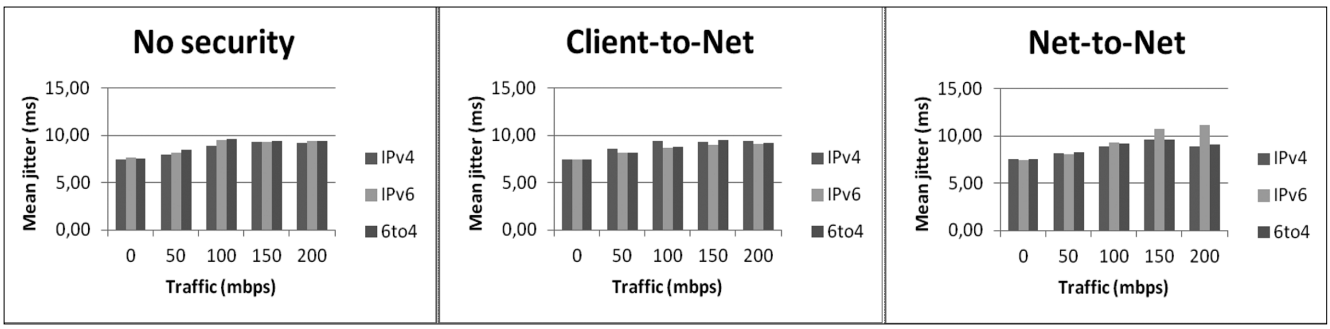
Relative Frequency Distribution: The relative frequency distribution of delta provides more details concerning the actual values of delta that are obtained. At background traffic rates of 0 and 50 Mbps, for all IPsec scenarios with either IP version or 6to4 encapsulation, approximately 70-80% of packets have delta values between 0-24 ms and the rest have delta values between 25-49 ms. The same is true at 100 Mbps, for the no-security and client-to-network scenarios, with either IP version or 6to4 encapsulation, except that a very small number (less than 1%) of packets have delta values between 50-74 ms.

For all three IPsec scenarios with either IP version or 6to4 encapsulation at 150 and 200 Mbps, at most 6% of packets have delta values between 50-74% and a very small number (at most 1.5%) have delta values of 75 ms or more. For the network-to-network scenario, the delta distribution has more variability: with either IP version or 6to4 encapsulation at 0 and 50 Mbps, the delta distribution is similar to the other scenarios; at the higher rates of background traffic, the distribution is also similar to the other scenarios except that the percentages of packets having delta values respectively between 50-74 ms and 75 ms or more increases (for instance, the percentage of packets having delta values between 50-74 ms varies from about 5-30% and the percentage of packet having delta values of 75 ms or more varies from about 5-15%.

B) Jitter

Max Jitter: Max jitter ranges from 13 ms for IPv4 with no security to 24 ms at 150 Mbps for the network-to-network scenario with IPv4. In the case of IPv6, max jitter ranges from 13 ms for the client-to-network or network-to-network scenarios with no background traffic to 21 ms for the network-to-network scenario with background traffic at 200 Mbps. With 6to4, max jitter varies from 13 ms

Figure 3. Mean jitter



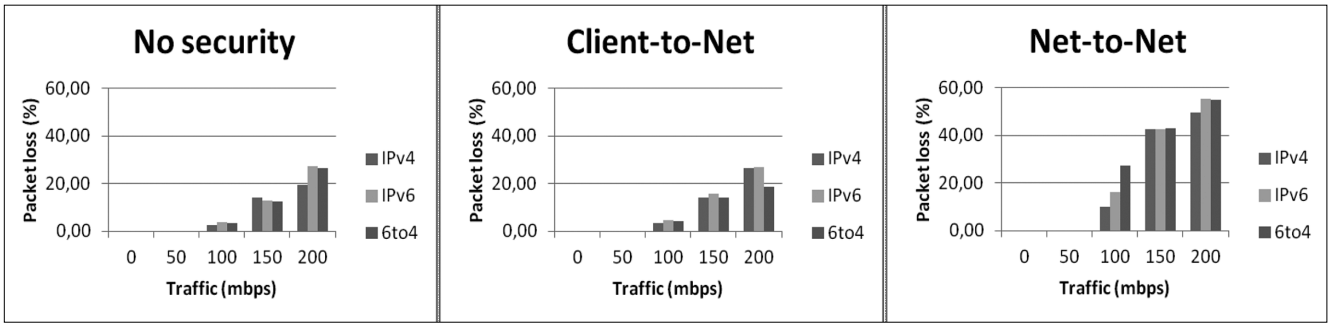


Figure 4. Packet loss

with no security and no background traffic to 26 ms for the network-to-network scenario with background traffic at 100 Mbps. However, max jitter sometimes reached 27 ms even with no security and no background traffic. Thus, it is important to examine both maximum and mean jitter.

Mean Jitter: Mean jitter values are shown in *Fig. 3.* The values range from 7-10 ms for IPv4, from 7-11 ms for IPv6, and from 7-10 ms with 6to4. The results show that mean jitter is not affected significantly by IPsec and 6to4 processing.

C) Packet loss

Packet loss percentages are shown in *Fig. 4.* Wireshark calculates these percentages by using RTP sequence numbers to determine missing packets. We note that there is no packet loss when background traffic is at 0 or 50 Mbps for all IPsec scenarios with either IP version or 6to4 encapsulation. At 100 Mbps of background traffic, packet loss with IPv4 varies from 2% for no-security to 10% for the network-to-network scenario. For IPv6, the range is from 4-16%, and for 6to4, it is from 4-27%. The highest packet loss percentage is 55% for the network-to-network scenario at 200 Mbps with either IPv6 or 6to4.

D) MOS

The maximum MOS of 4.41 is obtained with 0 or 50 Mbps of background traffic regardless of the IPsec scenario and regardless of whether IPv4, IPv6 or 6to4 encapsulation is used. At 100 Mbps of background traffic, MOS values are good with a slight drop for the client-to-network and network-to-network IPsec scenarios with either IP version or 6to4 encapsulation. As expected, at 150 or 200 Mbps of background traffic, the MOS drops to unacceptable levels.

E) Throughput

The throughput is the number of bits transferred per second considering only the voice packets. Throughput is shown in *Fig. 5.* If there is no packet loss, it is easily verified that the expected throughput with 20 ms voice packets is 0.4*s, where s is the total size in bytes of the voice packets including all headers and data. The packet size with IPv4 is 218 bytes, and with IPv6 and 6to4 is 238 bytes (since there are 160 bytes of voice data, 12 bytes of RTP header, 8 bytes of UDP header, either 40 bytes of IPv6 header or 20 bytes of IPv4 header, and 18 bytes of Ethernet header plus trailer). This gives expected throughput rates of 87.2 kbps with IPv4, and 95.2 kbps with IPv6 or 6to4.

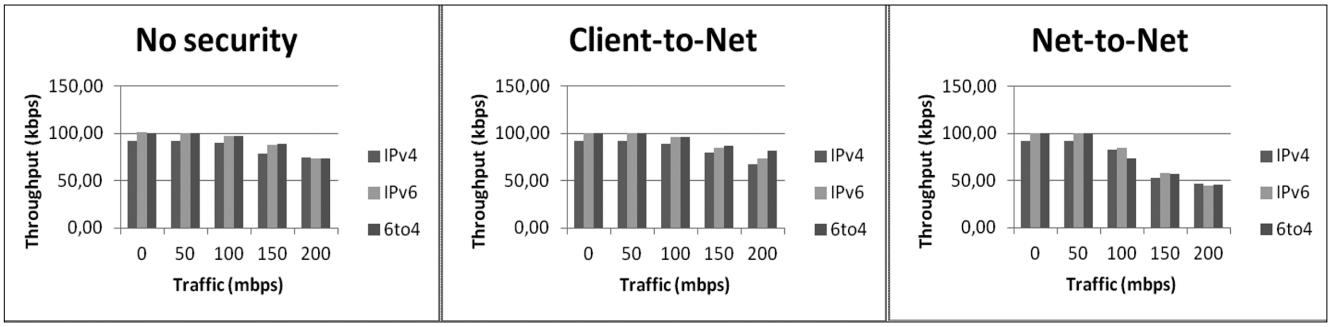
As expected, the measured and expected throughput is similar with background traffic rates up to 50 Mbps but differ when the rates are 100 Mbps or higher. This is because there is packet loss and increased delays at higher background traffic rates, which lowers the number of packets received per second. In general, the throughput for all IPsec scenarios is about the same for a given rate of background traffic with either IP version or 6to4. This implies that the additional overhead due to the extra headers and processing with IPsec and IPv6 or 6to4 does not significantly affect the voice throughput.

Note that size of a packet captured by Wireshark does include the 4-byte CRC at the end of an Ethernet packet. It computes the throughput by multiplying the number of packets received during the measurement interval by the observed packet size.

F) NAT with 6to4 or Teredo

To determine the additional overhead on the IPsec/6to4 router when it is using NAT to handle traffic from IPv4

Figure 5. Throughput



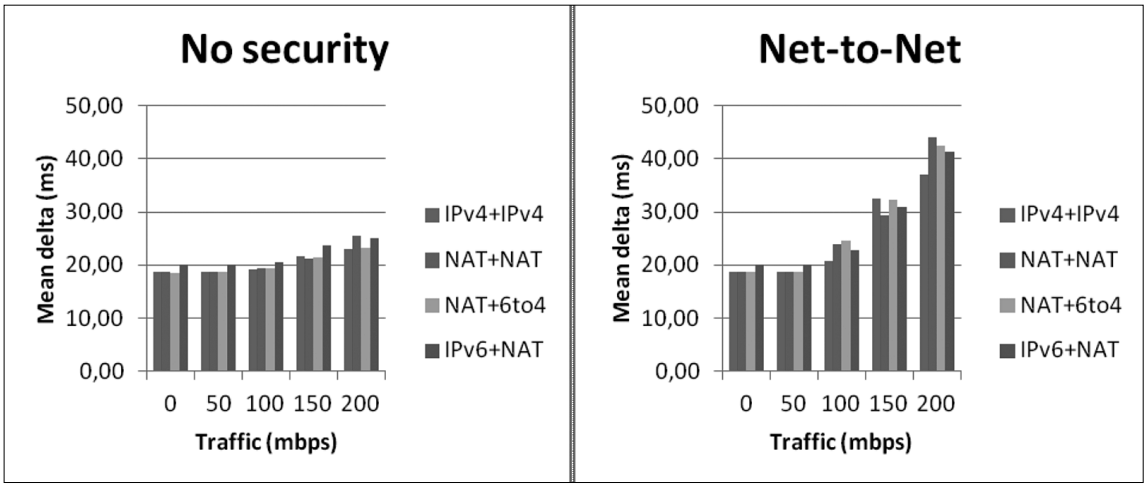


Figure 6. Mean delta (NAT with 6to4)

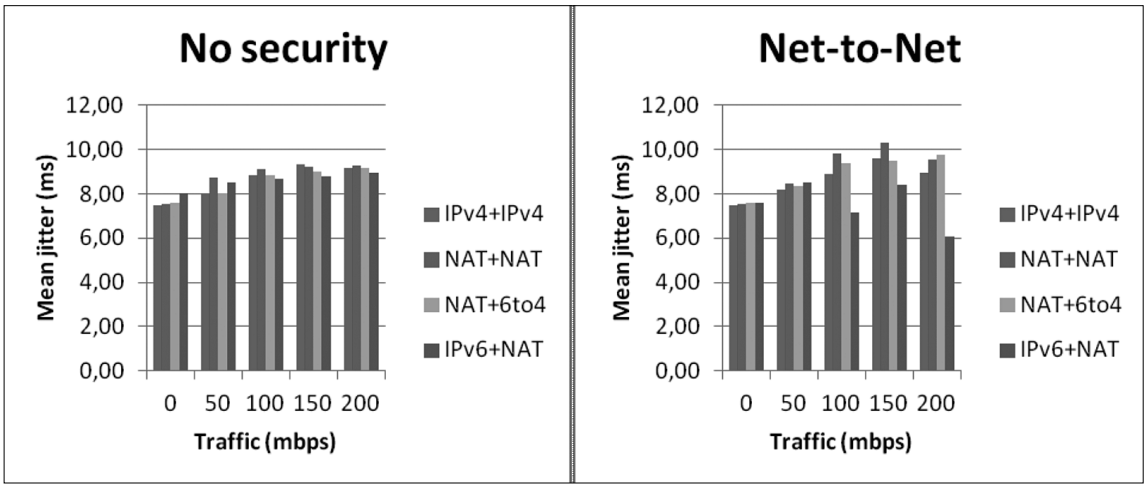


Figure 7. Mean jitter (NAT with 6to4)

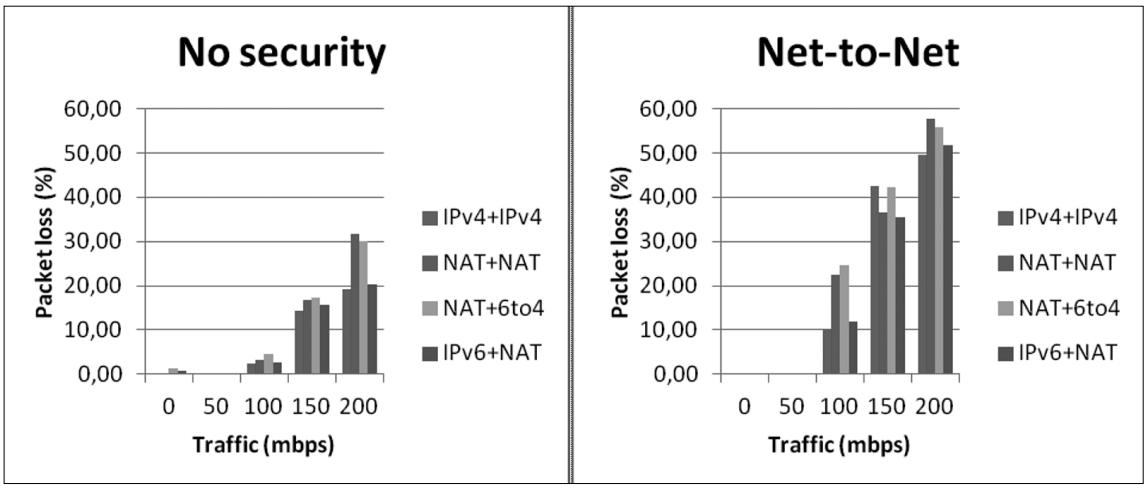


Figure 8. Packet loss (NAT with 6to4)

subnets, we repeated the previous experiments after enabling NAT. The difference in the values of all measures of interest due to NAT processing was found to be negligible regardless of the IPsec scenario. For reasons of space, we only show mean delta, mean jitter and packet loss for the no security and network-to-network IPsec scenarios with NAT *(Fig. 6-8)*.

In the figures, IPv6+NAT means the clients are IPv6 and the background traffic is IPv4 with NAT, NAT+6to4 means clients are IPv4 with NAT and background traffic is 6to4, and so on. However, as noted previously, NAT traversal over 6to4 is not possible unless the NAT box and 6to4 edge router are co-located and the NAT box is capable of performing the necessary 6to4 functions.

Teredo required the exchange of 6 pairs of router solicitation and router advertisement pairs between the Teredo client and the Teredo server over IPv4 before voice packets were exchanged over IPv6. Each solicitation message is an 89-byte IPv4 datagram that carries a UDP-encapsulated IPv6 message prefixed with a 13-byte Teredo authentication header. The IPv6 message contains an 8-byte ICMPv6 message. The advertisement message is a 125-byte IPv4 datagram that carries a UDP-encapsulated IPV6 message prefixed with a 13-

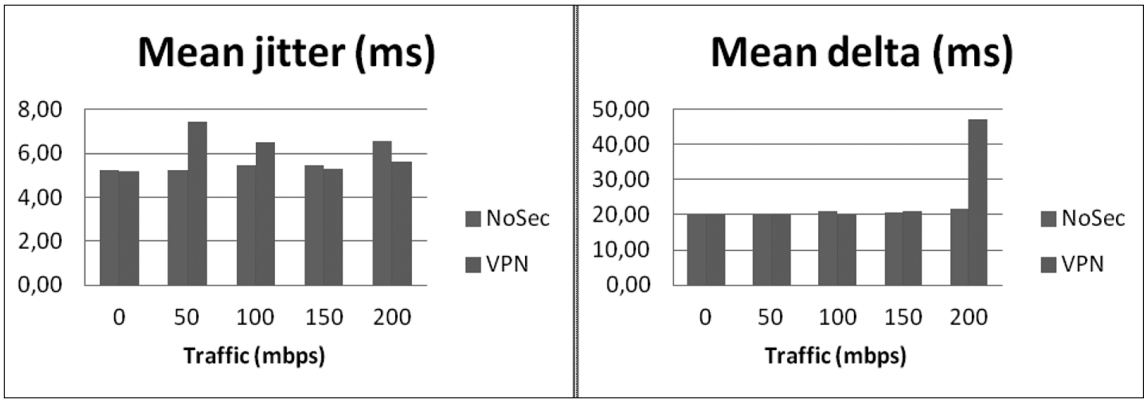


Figure 9. Mean jitter and mean delta (Teredo)

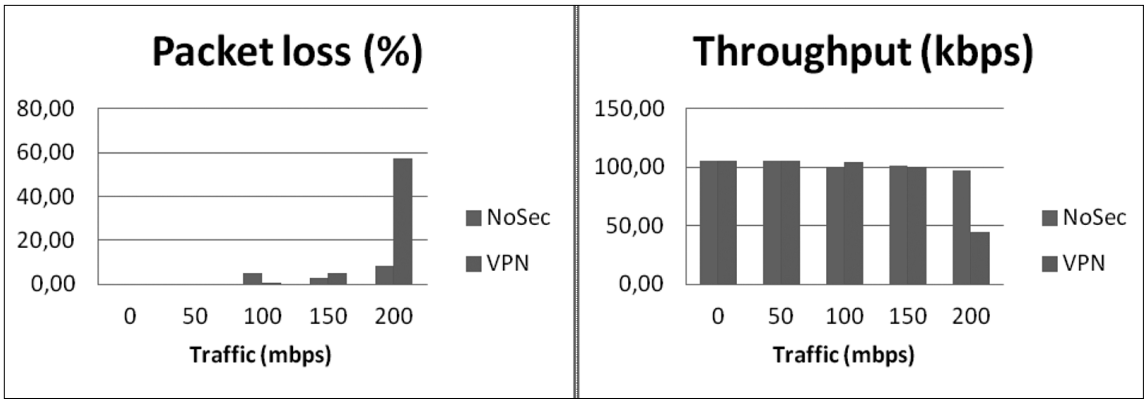


Figure 10. Packet loss and throughput (Teredo)

byte Teredo authentication header and an 8-byte Teredo indicator. The IPv6 message contains a 56-byte ICMPv6 message. Each voice packet over Teredo is a 248-byte IPv4 datagram that consists of a UDP-encapsulated IPv6 message containing 160 bytes of voice data, a 12-byte RTP header, and an 8-byte UDP header. In contrast, a 6to4 voice packet has 8 bytes less due to not requiring the extra UDP header. The only IPsec scenarios possible with Teredo are no security and network-to-network. Comparing *Fig. 2, 3 and 9* we see that that the mean jitter and mean delta values for Teredo and 6to4 are similar.

Fig. 10 contains packet loss and throughput for Teredo. The packet loss values for Teredo are slightly better than those for 6to4 in Fig. 4. The expected throughput value for Teredo computed using the formula given earlier is 0.4*266=106.4 kbps, and this value is close to that achieved for background traffic at 150 Mbps or less.

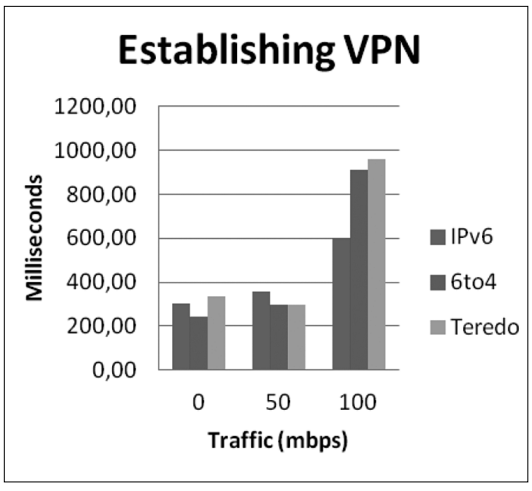


Figure 11. IPsec key exchange delay

When background traffic is at 200 Mbps, throughput in the VPN scenario drops drastically due to the high percentage of lost packets.

G) IPsec key exchange and SIP call set up times

We also determined the delays for the key exchange to initially set up the VPN using the IKEv1/ISAKMP handshake.

Fig. 11 shows that the delays during the initial key exchange are between 200-400 ms with background traffic levels of 0-50 Mbps regardless of whether IPv6, 6to4 or Terdeo is used. When the traffic level exceeds 100 Mbps, delays are unpredictable due to a backoff algorithm that doubles the delays between retries. This delay would be a factor in evaluating overall VoIP performance if the handshake is repeated several times during a call to provide additional protection against key compromise or staleness.

Fig. 12 reports the delay between sending a SIP registration request (required of all clients prior to call set up) and receiving the 200 OK message. It is seen that delays for the network-to-network scenario with no security are comparable for IPv6, 6to4, and Teredo (around 50 ms) when background traffic levels are 0-50 Mbps. With a VPN tunnel, the delays appear to be unpredictable even with background traffic at 50 Mbps.

Finally, *Fig. 13* measures the SIP call set up delay, which is the time between sending the invite message and receiving the 180 ringing message. With 0-50 Mbps of background traffic, the call set up delays are less than 20 ms for IPv6, 6to4 and Teredo with or without a VPN tunnel. When the background traffic level rises to 100 Mbps, delays are on the order of 120 ms.

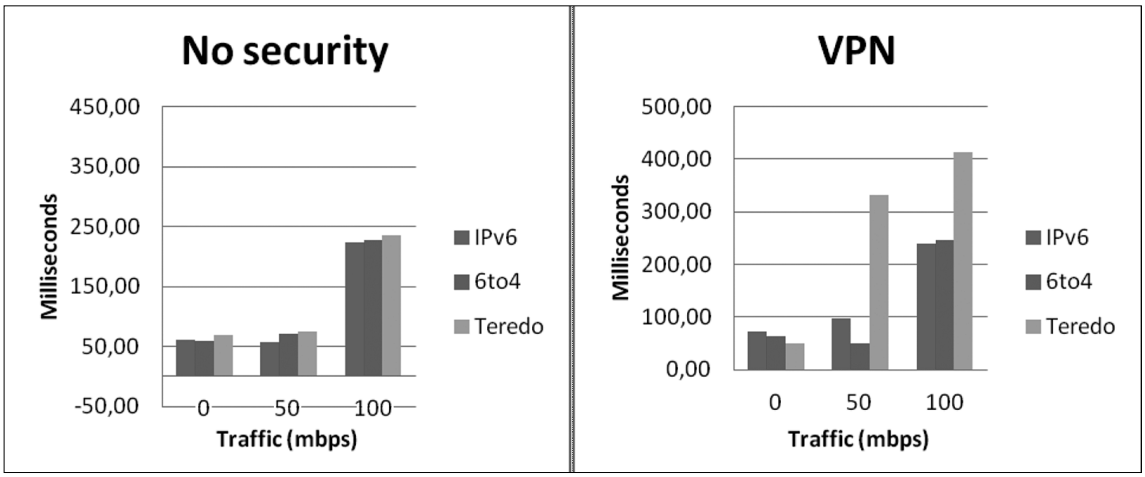


Figure 12. SIP registration delay

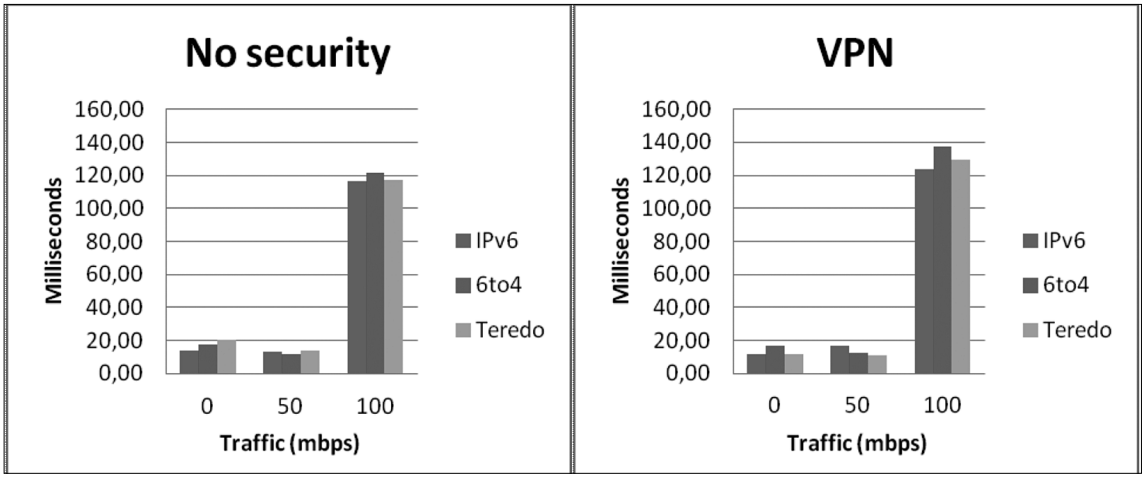


Figure 13. SIP call set up time

5. Summary

We conducted a study to evaluate VoIP performance with IPsec in IPv4, IPv6 and 6to4 networks, and also when using Teredo for NAT traversal in a test LAN. The experiments used softphones to make calls and generated background traffic to create congestion on the links and routers. The results demonstrated the feasibility of using a single Linux box to handle IPsec, 6to4 and NAT processing. It was found that voice quality is acceptable as long as the traffic does not exceed network capacity.

The study showed that VoIP performance with IPsec is not adversely affected by the overhead due to 6to4 or Teredo. Future studies should evaluate the impact of multiple calls and the use of IPsec with both Teredo and 6to4 in a more complex test network.

Authors

ROMAN YASINOVSKYY received his B.Sc. degree in computer science and the M.Sc. degree in computer control systems and technologies from the National University of Kyiv-Mohyla Academy, Ukraine in 2002 and 2004 respectively. He received his D.Sc. degree from Towson University in 2009. His dissertation dealt with VoIP performance over IPv6. He is currently a lecturer in the Department of Computer and Information Sciences at Towson University. His research interests include operating systems security, networking, and parallel/distributed computing.

ALEXANDER L. WIJESINHA is an associate professor in the Department of Computer and Information Sciences at Towson University. He holds a Ph.D. in computer science from the University of Maryland Baltimore County, and both a M.S. in computer science and a Ph.D. in mathematics from the University of Florida. He received a B.S. in mathematics from the University of Colombo, Sri Lanka. His research interests are in computer networks including wireless networks, VoIP, network protocol adaptation for bare machines, network performance, and network security.

RAMESH K. KARNE is a professor in the Department of Computer and Information Sciences at Towson University. He obtained his Ph.D. in Computer Science from the George Mason University. Prior to that, he worked with IBM at many locations in hardware, software, and architecture development for mainframes. He also worked at the Institute of Systems Research at University of Maryland, College Park as a research scientist. His research interests are in bare machine/dispersed operating system computing.

References

[1] Carpenter, B. and Moore, K.,
"Connection of IPv6 Domains via IPv4 Clouds",
RFC 3056, February 2001.

[2] Huitema, C.,
"Teredo: Tunneling IPv6 over UDP through Network Address Translations (NATs)",
RFC 4380, December 2006.

[3] Huang, S-M., Wu, Q., and Lin, Y-B.,
"Tunneling IPv6 through NAT with Teredo Mechanism,"
In Proc. 19th International Conference on Advanced Information Networking and Applications (AINA'05), Vol. 2, pp.813–818, 2005.

[4] Miredo – Teredo for Linux and BSD,
http://www.remlab.net/miredo/

[5] Kent, S.,
"IP Encapsulating Security Payload (ESP)",
RFC 4303, December 2005.

[6] Yasinovskyy, R., Wijesinha, A., and Karne, R.,
"Impact of IPsec and 6to4 on VoIP Quality over IPv6,"
In Proc. 10th International Conference on Telecommunications (ConTEL'09), pp.235–242, 2009.

[7] Ariga, S., Nagahashi, K., Minami, M., Esaki, H., and Murai, J.,
"Performance Evaluation of Data Transmission using IPSec over IPv6 Networks,"
In Proc. INET'2000, July 2000.
http://www.isoc.org/inet2000/cdproceedings/1i/1i_1.htm

[8] Shue, C., Shin, Y., Gupta, M., and Choi, J.Y.,
"Analysis of IPSec overheads for VPN servers,"
In Proc. 1st IEEE ICNP Workshop on Secure Network Protocols, pp.25–30, 2005.

[9] Shue, C., Gupta, M., and Myers, S.A.,
"IPSec: Performance Analysis and Enhancements,"
In Proc. IEEE International Conference on Communications (ICC), Glasgow, Scotland, June 2007.

[10] Perez, J.A., Zarate, V., Montes, A., and Garcia C.,
"Quality of Service Analysis of IPSec VPNs for Voice and Video Traffic,"
In Proc. International Conference on Internet and Web Applications and Services/Advanced International Conference on Telecommunications (AICT-ICIW), pp.43–43, 2006.

[11] Hadjichristophi, G.C., Davis IV, N.J., and Midkiff, S.F.,
"IPSec overhead in wireline and wireless networks for web and email applications,"
In Proc. 22nd IEEE IPCCC, April 2003.

[12] Meenakshi, S.P. and Raghavan, S.V.,
"Impact of IPSec Overhead on Web Application Servers,"
In Proc. International Conference on Advanced Computing and Communications (AdCom), pp.652–657, 2006.

[13] Liu, L. and Gao, W.,
"Building IPsec VPN in IPv6 Based on Openswan,"
In Proc. IFIP International Conference on Network and Parallel Computing Workshops (NPC), pp.784–787, 2007.

[14] Zeadally, S. and Raicu, I.,
"Evaluating IPv4 to IPv6 transition mechanisms,"
In Proc. 10th International Conference on Telecommunications (ICT), pp.1091–1098, 2003.

[15] Mujinga M., Muyingi, H., and Rao, G.S.V.R.K.,
"IPSec overhead analysis in dual stack IPv4/IPv6 transition mechanisms,"
In Proc. ICACT, 2006.

[16] Yasinovskyy, R., Wijesinha, A., Karne, R., and Khaksari G.,
"A Comparison of VoIP Performance on IPv6 and IPv4 Networks,"
In Proc. 7th ACS/IEEE International Conference on Computer Systems and Applications (AICCSA'09), pp.603–609, 2009.

[17] Hoeher, T., Petraschek, M., Tomic, S., Hirschbichler, M.,
"Evaluating Performance Characteristics of SIP over IPv6,"
Journal of Networks, Vol. 2, No. 4, pp.40–50, 2007.

[18] Bokor, L., Kanizsai, Z., and Jeney, G.,
"Performance Evaluation of key IMS operations over IPv6-capable 3G UMTS networks,"
In Proc. 9th International Conf. on Networks, 2010.

[19] Baugher, M., McGrew, D., Naslund M., Carrara E., and Norrman K.,
"The Secure Real-time Transport Protocol (SRTP),"
RFC 3711, March 2004.

[20] Linphone,
http://www.Linphone.org/

[21] The Multi-Generator (MGEN) Version 4.2,
http://pf.itd.nrl.navy.mil/mgen/mgen.html

[22] Openswan,
http://www.Openswan.org/

[23] strongSwan,
http://www.Strongswan.org/

[24] Wireshark,
http://www.Wireshark.org/

[25] OpenSER SIP server (now OpenSIPS),
http://www.OpenSER.org/