Application-Independent End-to-End Security in Shared-Link Access Networks

José C. Brustoloni and Juan A. Garay

Bell Laboratories – Lucent Technologies
600 Mountain Avenue
Murray Hill, NJ 07974, USA
{jcb,garay}@research.bell-labs.com

Abstract. ISPs now offer Internet access via cable modem or DSL, which provide much higher bandwidth than does PSTN. Higher access bandwidths allow ISP customers to exploit NAT (network address and port translation) to amortize the cost of an ISP account among multiple computers. The reduced per-computer cost may encourage airport lounges, hotels, and other businesses that serve "road warriors" to provide Internet connectivity to their clients. Unfortunately, NAT may not interoperate with IPSec, which provides application-independent security in VPNs (virtual private networks). A VPN is necessary, e.g., to connect a "road warrior" securely to a corporate Intranet via the untrusted Internet. We propose a simple DHCP extension that allows client IPSec implementations to interoperate with NAT. The resulting architecture, EASE, makes "road warrior" access easy, secure, and economical.

1 Introduction

Many ISPs (Internet service providers) still offer Internet access via PSTN (public switched telephone network) lines, which provide low bandwidth (at most 57 Kbps). Recently, however, ISPs began offering Internet access via cable modem or DSL (digital subscriber line). The latter alternatives provide much higher bandwidth (up to several Mbps) at only slightly higher price.[1]

Higher bandwidths make it practical to share a single access link and ISP account among multiple computers. Sharing is implemented by NAT (network address and port translation) [8] and reduces the per-computer Internet connectivity cost. The reduced cost may encourage businesses that serve "road warriors," such as airport lounges, hotels, and conference centers, to provide Internet connectivity to their clients ("road warriors" are people who need to work away from their offices).

However, NAT is assumed to be incompatible with IPSec (the IP security architecture) [14,15,6] and therefore unsuitable for "road warriors." IPSec provides

[1] For example, in the United States, monthly flat fees in July of 1999 were around $14 for a PSTN line and $20 for a PSTN ISP account, versus $40 for cable service and cable ISP account.

G. Pujolle et al. (Eds.): NETWORKING 2000, LNCS 1815, pp. 608–619, 2000.

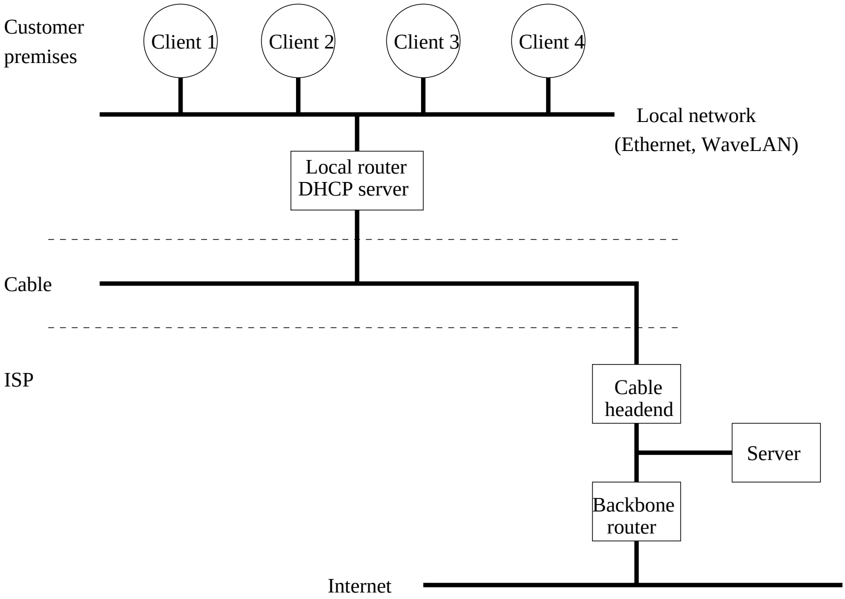


Fig. 1. The EASE architecture allows multiple clients to share easily, securely, and economically a single high-bandwidth access link and ISP account.

application-independent security in VPNs (virtual private networks). A VPN is necessary, e.g., to connect a “road warrior” securely to a corporate Intranet via the untrusted Internet.

Contributions of this paper. This paper proposes a simple DHCP (dynamic host configuration protocol) [7,1] extension that enables client IPSec implementations to fully interoperate with NAT, making “road warrior” connectivity easy, secure, and economical. The proposed extension is implemented in the EASE architecture, as illustrated in Fig. 1.

EASE uses a *shared-link* access network, where multiple hosts may be dynamically connected to a local network (e.g., Ethernet or WaveLAN). A local router connects the local network to an ISP via a shared high-bandwidth link (e.g., cable, DSL, or T1 line). EASE provides easy connectivity because its local router incorporates a DHCP server, which automatically provides to dynamically connected client hosts the necessary networking configuration (e.g., IP address and default router). The local router also implements NAT, reducing the per-host Internet connectivity cost.

Security is the biggest hurdle in an architecture such as EASE’s. Because client hosts are connected to a local network, an airport lounge that adopts such an architecture might allow, for example, a passenger to forge or snoop on another passenger’s packets. Preventing forgery and snooping requires authentication and encryption, respectively. Clients may use IPSec to obtain the required end-to-end security (authentication and/or encryption) without modifications to applications. Our proposed DHCP extension makes it possible for IPSec to interoperate with NAT, achieving easy, secure, and economical connectivity.

Related work. There are many alternatives to IPSec. For example, SSH (Secure Shell) [17] and SSL (Secure Sockets Layer) [10] implement security in the application layer. SSH provides security for applications such as logging into a remote computer, executing commands in the remote computer, and transferring files between the local and the remote computers. SSL was introduced by Netscape and is widely used for Web applications. Unlike such protocols, IPSec implements security at the network layer and has the advantage of being application-independent.

PPTP (Point-to-Point Tunneling Protocol) [23] was introduced by Microsoft and allows "road warrior" users to connect with corporate Intranets via the Internet. PPTP can make long-distance calls into corporate Intranets unnecessary (a local call into an ISP suffices). PPTP can also spare corporate Intranets the cost of access routers. IPSec can provide similar benefits in tunnel mode. However, IPSec is vendor-independent, seemingly more secure [3], and, unlike PPTP, can also provide end-to-end security (in transport mode, as explained in Section 3).

The DHCP extension proposed in this paper can reduce access costs and might allow access services to be provided on a complimentary basis. In another paper [2], we describe how businesses that serve "road warriors" may provide Internet access to their clients and charge for such access.

Organization of the paper. The rest of this paper is organized as follows. Sections 2, 3, and 4 discuss in greater detail DHCP, IPSec, and NAT, respectively. Section 5 describes VPN masquerade, a NAT implementation that provides limited interoperation with IPSec. Section 6 describes our proposed DHCP extension, which builds on VPN masquerade to provide full interoperability and backward compatibility. Section 7 presents a summary and final remarks.

2 DHCP

This section describes DHCP in greater detail. DHCP is a protocol that allows *client* hosts to obtain configuration parameters from *server* hosts. In the EASE architecture, client hosts use DHCP to obtain and maintain their networking configuration, including client IP address, network address mask, broadcast address, and IP addresses of the router, DNS (domain name system) server, NTP (network time protocol) server, and (possibly) the line printer server assigned to the client. DHCP is what makes EASE easy to use: Clients can, for example, simply connect their laptops to the Ethernet or WaveLAN in an airport lounge or conference room, reboot the computer, and automatically be ready to access the Internet.

DHCP is layered on top of UDP. DHCP clients and servers use UDP ports 68 and 67, respectively. DHCP packets have a format similar to that of BOOTP

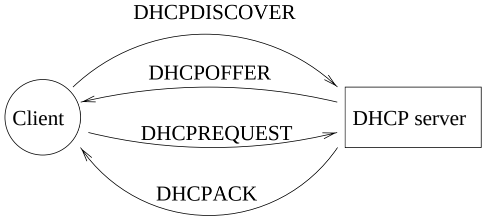


Fig. 2. Clients automatically obtain their networking configuration from a DHCP server.

(boot protocol) [4]; the same format is used both for client requests and server responses [7].

When a DHCP client boots, it broadcasts in the local network a DHCP packet of type DHCPDISCOVER, as shown in Fig. 2. This causes one or more DHCP servers to send to the client a packet of type DHCPOFFER. The client then broadcasts a DHCPREQUEST packet, specifying the selected server in the packet's "server identifier" option. That server then commits the configuration by replying DHCPACK to the client.

If the local network does not contain an active DHCP server, the local router (or another BOOTP or DHCP agent) *relays* DHCP client requests to the appropriate DHCP server. The agent marks its address in the packet's `giaddr` (gateway IP address) field. This allows the actual DHCP server to return the packet to the agent, instead of attempting to reply directly to the client. The agent then returns the reply to the client, using the client's MAC address.

DHCP supports three mechanisms for client IP address allocation: *manual*, *automatic*, or *dynamic*. In manual allocation, each client's IP address is assigned by the network administrator, and DHCP is used only for centralizing such configuration. In automatic allocation, the DHCP server itself selects a permanent IP address for each client. Finally, in dynamic allocation, the DHCP server assigns an IP address to a client only for the period of time specified in the DHCP packet's "IP address lease time" option. To keep its IP address, the client must initiate another DHCPREQUEST before the lease expires. Clients may also explicitly release an IP address by sending to the server a DHCPRELEASE packet. The DHCP server may reuse client IP addresses after they are expired or released. EASE uses DHCP's dynamic allocation.

3 IP Security

As mentioned in the previous section, EASE uses DHCP for automatic networking configuration of, for example, client laptops dynamically connected to the Ethernet or WaveLAN in an airport lounge, hotel, or conference room. Because in such applications a client might easily snoop on another client's packets, EASE uses IP Security (IPSec) to provide the necessary security. This section summarizes the IPSec fundamentals.

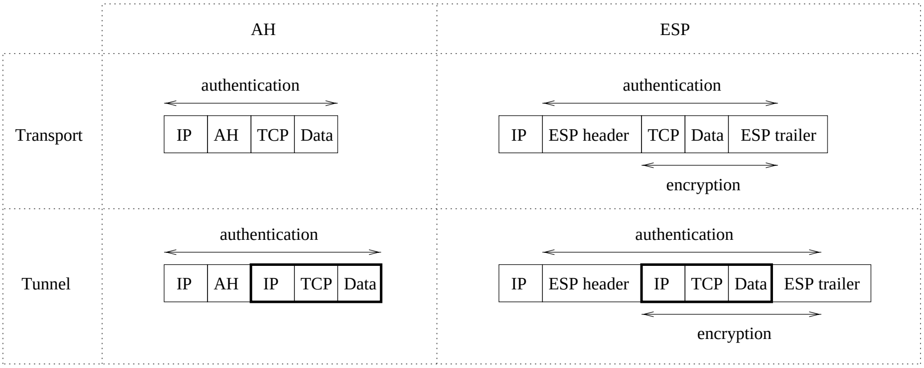


Fig. 3. IPSec packet format depends on protocol (AH or ESP) and mode (transport or tunnel). The portion of the packet that is authenticated or encrypted is different for AH or ESP. The encapsulated packet is shown in bold.

IPSec is an Internet standard from the IETF IPSec Working Group [16]. IPSec is a mandatory part of the next-generation IP protocol (IPv6 [5]), but most existing IPSec implementations assume current-generation IP (IPv4). IPsec operates at the network layer and therefore is independent of the transport- (e.g., TCP or UDP) or application-layer protocol (e.g., HTTP, FTP, or TELNET). IPSec is essentially an *encapsulation* protocol, namely, one that defines the syntax and semantics of placing one packet inside another. IPSec defines two protocols, AH (Authentication Header) [14] and ESP (Encapsulating Security Payload) [15]. AH can provide authentication of packet origin, proof of integrity of packet data, and protection against packet replay. ESP can provide, in addition to AH's services, encryption of packet data and limited traffic flow confidentiality.

AH and ESP can be used either in *transport* or *tunnel* mode. Transport mode provides end-to-end security between the packet's source and destination. In contrast, tunnel mode *encapsulates* packets and thus provides security between the nodes where the packet is encapsulated and decapsulated (these can be any nodes, e.g. routers, on the path between the packet's source and destination). In EASE, a "road warrior" client might use, for example, transport mode to download (via FTP) a document from a supplier's server. On the other hand, a client would use tunnel mode to connect to an IPSec gateway into the Intranet of the client's employer.

The packet layout depends on the protocol and mode, as shown in Fig. 3. In IPv4, AH and ESP are identified by values 51 or 50 in the IP header's protocol field, respectively. AH and ESP insert a header between the IP header and the upper-layer header (in transport mode) or the encapsulated IP datagram (in tunnel mode). ESP also appends a packet trailer. Note that, in tunnel mode, the IP header may have source and destination IP addresses different from those of the encapsulated packet.

AH's authentication covers the whole IP datagram, as illustrated in Fig. 3. In contrast, ESP's authentication skips the IP header and the final part of the ESP trailer (which contains the authentication data). ESP's encryption skips both IP and ESP header and the final part of the ESP trailer.

IPSec peers negotiate what security services to implement (e.g., authentication and/or encryption) and what algorithms and keys to use. In addition to MD5 [19] and SHA [20] for authentication and DES [21] for encryption, IPSec implementations may support other algorithms. The choice of services, algorithms, and keys is called a *security association* (SA). The framework for SA negotiation is defined by ISAKMP (Internet Security Association and Key Management Protocol) [22]. ISAKMP is layered on top of UDP and uses UDP port 500 both for source and destination. IPSec's negotiation is more specifically defined by IKE (Internet Key Exchange) [12]. An IPSec packet's SA is uniquely identified by the protocol (AH or ESP) and destination IP address in the IP header, in conjunction with the SPI (Security Parameters Index, a 32-bit field) in the AH or ESP header.

4 NAT

Although DHCP makes EASE's configuration easy and IPSec makes EASE's communication secure, EASE would still be impractical if a separate ISP account were necessary for each EASE client. Because NAT allows EASE clients to share a single ISP account, NAT makes EASE convenient and economical. This section explains how NAT works.

NAT allows local hosts to use each a *private* IP address. Private addresses were reserved by the Internet Assigned Numbers Authority (IANA) for non-exclusive private use [25]. Private addresses spare EASE installations of the burden of obtaining globally unique IP addresses for each client. Dynamically connected clients obtain from EASE's DHCP server locally unique private IP addresses. When EASE clients need to communicate with other hosts on the Internet, they must (at least temporarily) use a *global* IP address (which is globally unique and, therefore, routable). EASE obtains global IP addresses from the ISP.

NAT is implemented in the local router between the local network and the ISP and provides the necessary translations between private and global addresses. NAT uses the upper-layer (e.g. TCP or UDP) port number to distinguish packets of the various local hosts[2]. In *outgoing* traffic (packets sent to the ISP), NAT modifies each packet header's *private* source (IP address, port number) to a *global* source (IP address, port number). NAT maintains in a *translation table* the one-to-one correspondence between private and global (IP address, port

[2] When NAT was originally proposed [8], it used a pool of global addresses and thus might not require port translation. For economic reasons, however, it became more usual to use a single global IP address (or a small number of global IP addresses), along with port translation.

number) pairs. When NAT receives corresponding *incoming* traffic (packets received from the ISP), NAT modifies the packet header's destination from global to private (IP address, port number).

Some application-layer protocols, e.g. FTP (File Transfer Protocol) [24], may include in packet payloads IP addresses and possibly port numbers. Such addresses and port numbers must also be translated. Therefore, for each such protocol, NAT includes an Application Level Gateway (ALG) that provides the necessary translations.

Note that DHCP and NAT give to EASE clients a degree of anonymity. In a hotel, for example, a given private IP address could at any time be allocated to any guest, and the hotel's global IP address could be simultaneously used by all guests. This anonymity is usually advantageous.

Most NAT implementations do not support IPSec. In fact, it is widely believed that IPSec cannot interoperate with NAT [6]. The next section shows, however, that under certain conditions, some partial interoperation is possible. In Section 6 we show how to achieve full IPSec functionality with NAT.

5 VPN Masquerade

Several difficulties suggest that interoperation of IPSec with NAT is not possible. AH's authentication covers the entire packet, including source and destination IP addresses. When NAT translates an address, it would need to adjust AH's authentication data correspondingly. Unfortunately, that is not possible, because NAT does not (and should not) have access to the authentication key. In contrast, ESP's authentication does not cover the IP header. However, ESP interoperation with NAT can still be problematic in transport mode: When NAT translates the source or destination IP address, it would need to adjust the TCP or UDP checksum correspondingly. (TCP and UDP checksums are calculated over the packet's IP "pseudo-header," TCP or UDP header, and data. The pseudo-header includes the source and destination IP addresses.) However, because the checksum is encrypted (along with the rest of the TCP or UDP header and data) but NAT does not have access to the encryption key, NAT would be unable to make the necessary adjustment. Another problem with both AH and ESP is that, unlike TCP and UDP, they do not use "port numbers" that NAT could modify and use for demultiplexing incoming traffic.

"VPN masquerade" [11] is a patch for Linux that, unlike other NAT implementations, does support IPSec, but only for the case of ESP in tunnel mode (and not ESP in transport mode or AH). NAT is possible in this case because, in tunnel mode, the IP pseudo-header of the encapsulated packet is unaffected by NAT's address translations, and therefore no adjustments are necessary in encapsulated checksums.[3]

VPN masquerade does not attempt to translate TCP or UDP port numbers of encapsulated packets, which may be authenticated and/or encrypted. Instead,

[3] In this case, however, the anonymity provided by NAT is lost, as the private IP address is sent unchanged in the encapsulated packet.

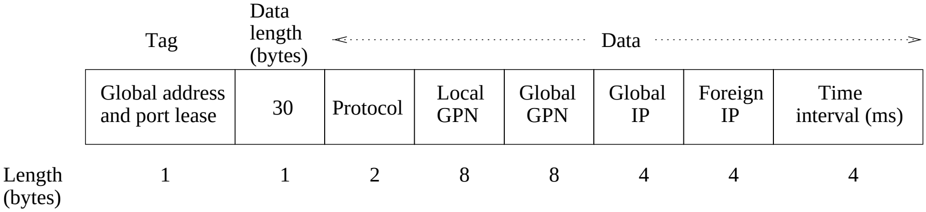
VPN masquerade resorts to a number of heuristics for demultiplexing incoming packets. The heuristics may fail only when two or more local hosts communicate with the same remote node, but even then the probability of failure is low. When the heuristics fail, an incoming packet may be forwarded to the wrong local host. This vulnerability could be used in denial-of-service attacks, but does not compromise integrity or privacy any more than snooping on the local network would.

VPN masquerade treats ISAKMP packets as a special case: UDP packets with source and destination port 500 do not have the port translated. If the packet is outgoing, VPN masquerade writes down in the translation table the local and foreign addresses and the "initiator cookie," a 64-bit field present in all ISAKMP packets during a negotiation. The local address is private, while the foreign address is global and corresponds to a node outside the local network, with which the local host wishes to communicate. The initiator cookie is randomly selected by the local host. When VPN masquerade receives an incoming ISAKMP packet, it forwards the packet to the local address that corresponds to the packet's foreign address and initiator cookie. The packet may be incorrectly forwarded if more than one local host is negotiating with the same foreign node using the same initiator cookie.

VPN masquerade uses the foreign address and the SPI field in the ESP header to demultiplex incoming ESP packets. However, VPN masquerade has to determine the corresponding local address by inspection, because the portion of ISAKMP packets that specifies SPI values is encrypted. Additionally, if more than one local host chooses the same incoming SPI for communicating with the same foreign host, VPN masquerade may not demultiplex incoming packets correctly.

Items in VPN masquerade's translation table associate local address, foreign address, outgoing SPI, and incoming SPI. VPN masquerade marks a translation table item "outstanding" when the first outgoing packet between the given local and foreign addresses and with the given outgoing SPI is forwarded. The incoming SPI is set to 0, as it is then unknown. At most one item with a given foreign address can be outstanding at any time. When the first packet is received from a given foreign address with a given incoming SPI, VPN masquerade forwards the packet to the local address that has an outstanding item with that foreign address. VPN masquerade then updates the item's incoming SPI and marks the item "established." If there is no outstanding item with the given foreign address, VPN masquerade multicasts the incoming packet to all local addresses that have recently had an ISAKMP negotiation with the given foreign address. However, because of validation of the cryptographic transformations, the incoming packet will be accepted only by its intended recipient, and dropped by the other multicast recipients. To prevent denial-of-service attacks, VPN masquerade expires outstanding items after a short time, and established items after a period of inactivity.

Unfortunately, VPN masquerade's scheme is susceptible to race conditions that may cause misassociations. For example, if local hosts a and b both nego-



	Tag	Data length (bytes)	Data					
	Global address and port lease	30	Protocol	Local GPN	Global GPN	Global IP	Foreign IP	Time interval (ms)
Length (bytes)	1	1	2	8	8	4	4	4

Fig. 4. Our new DHCP option allows clients to lease global IP addresses, initiator cookies, and incoming SPIs for a specified time interval. Knowledge of the global IP address allows clients to use ESP in transport mode or AH. Cookie and SPI leases prevent NAT demultiplexing errors.

tiate ISAKMP with a foreign node f at the same time, and a sends an ESP packet to f before b does, a obtains an outstanding translation table item for f. However, if f sends a packet to b before replying to a, then b's packets will be erroneously forwarded to a and vice-versa. Misassociations are eventually cleared by timeouts.

6 DHCP Extension for IPSec/NAT Interoperation

The previous section describes VPN masquerade, a Linux patch that allows ESP in tunnel mode to interoperate with NAT under certain conditions. We describe in this section a new DHCP extension that builds on that previous work so as to provide full IPSec/NAT interoperability, including AH and ESP protocols both in transport and tunnel mode.

Our DHCP extension consists of a new DHCP option, GLOBAL_ADDRESS_AND_PORT_LEASE, as illustrated in Fig. 4. DHCP clients use this option to indicate to the DHCP server that, during the specified time interval, they intend to use the specified protocol, global IP address, and private and global GPN (generalized port number) to communicate with the specified foreign IP address. GPN is interpreted according to the protocol: for TCP and UDP, GPN is the TCP or UDP port number; for ISAKMP, GPN is the initiator cookie; for AH and ESP, GPN is the incoming SPI. Clients must specify the protocol and private GPN, but may leave unspecified (that is, with value 0) the time interval, global IP address, and global GPN parameters. The server picks appropriate values for unspecified parameters, and includes them in the reply. If the foreign IP address has value 0, it is a wild card that matches any foreign IP address.

A client sends to its default router the DHCPREQUEST packet with the GLOBAL_ADDRESS_AND_PORT_LEASE option. The router incorporates a DHCP server and VPN masquerade. The DHCP server processes the option if VPN masquerade is not *nested* (that is, the router does not connect the client's network to another link whose interface has a private IP address). On the other hand, if VPN masquerade is nested (e.g., the ISP provides only private IP addresses) then the DHCP server *relays* the request to another DHCP server (e.g., located at the ISP's cable head-end or backbone router).

Processing of the GLOBAL_ADDRESS_AND_PORT_LEASE option is as follows. If the client fully specifies the quadruplet (protocol, global GPN, global IP address, foreign IP address), the DHCP server first checks if the quadruplet is already assigned to the client in VPN masquerade's translation table (i.e., the client is renewing its lease). If so, the DHCP server extends the validity of that translation entry for the requested time interval (picking a default value if unspecified by the client) and returns DHCPACK to the client. Otherwise, the DHCP server checks if the client's total number of items in VPN masquerade's translation table is below a certain limit. If not, the DHCP server returns DHCPNACK to the client. Otherwise, the DHCP server attempts to pick global GPN and/or global IP address (if unspecified by the client) such that the quadruplet (protocol, global GPN, global IP address, foreign IP address) does not conflict with any other current assignment. If the client-specified or server-picked quadruplet has no conflicts, the DHCP server assigns the quadruplet to the client, installs the corresponding new item in VPN masquerade's translation table, and returns DHCPACK to the client. Otherwise, the DHCP server returns DHCPNACK to the client.

In order to interoperate fully with NAT, IPSec implementations should use the new DHCP option when the source IP address is private but the destination IP address is global, or vice-versa. In such cases, NAT is necessary, and IPSec should:

1. Before using an initiator cookie in an ISAKMP negotiation, lease the local host's global IP address and cookie from the DHCP server. This prevents NAT demultiplexing errors due to two or more local hosts using the same global IP address and cookie.
2. For similar reasons, before selecting an incoming SPI in an ISAKMP negotiation, lease the incoming SPI from the DHCP server (keeping the global IP address the same as in the first step).
3. For outgoing packets, before authentication and encryption, (i) in transport mode, replace source port number by a global port number; (ii) in tunnel mode, replace encapsulated source IP address and port number by a global IP address and port number; (iii) sum to the TCP or UDP checksum (a) the difference between global and private source IP addresses, and (b) the difference between global and private source port numbers; and (iv) process any ALG that may be necessary (e.g., for FTP packets).
4. Compute a packet's AH authentication data as if the source or destination IP address (for outgoing or incoming packets, respectively) were equal to the global IP address leased in the first step.
5. For incoming packets, after authentication and decryption, (i) process any ALG that may be necessary (e.g., for FTP packets); (ii) in transport mode, replace global destination port number by the corresponding private port number; (iii) in tunnel mode, replace in decapsulated packet the global destination IP address and port number by the corresponding private address

and port number; and (iv) subtract from the TCP or UDP checksum (a) the difference between global and private destination IP addresses, and (b) the difference between global and private destination port numbers.

6. Periodically renew leases for global IP addresses, initiator cookies, incoming SPIs, and global port numbers, while needed.

Note that TCP and UDP checksum arithmetic uses 16-bit 1-complement arithmetic.

Denial of service attacks are possible because, for example, DHCP packets are not authenticated. However, local and remote hosts or gateways establish cryptographic keys and SAs through ISAKMP, identifying and authenticating each other by means other than IP addresses. Therefore, intruders cannot jeopardize the packet authentication and/or privacy provided by IPSec.

Only minimal modifications are necessary to VPN masquerade. When the DHCP server installs a new AH or ESP item in VPN masquerade's translation table, the DHCP server marks the item "established" and sets its outgoing SPI to 0, a wild card that matches any outgoing SPI for the given local and foreign addresses (thus circumventing VPN masquerade's "outstanding" marking). VPN masquerade should demultiplex AH and ESP packets according to the foreign address and incoming SPI, and translate between private and global IP addresses according to the translation table.

Clients that adopt the new DHCP option can use the previously unsupported AH protocol and/or transport mode and prevent the race conditions and demultiplexing errors discussed in the previous section; also note that the anonymity provided by NAT is now preserved. However, our solution is backward-compatible and continues to support clients that use ESP in tunnel mode and that are not updated to take advantage of the new DHCP option.

7 Conclusions

EASE is an architecture that can greatly reduce the cost of Internet access and may allow airport lounges, hotels, and conference centers to provide convenient Internet connectivity. The technology for ubiquitous Internet access is largely available: Cable and DSL provide high-bandwidth, low-cost links to the Internet; NAT allows those links to be shared; DHCP provides automatic configuration; WaveLAN connects clients without wires; and IPSec makes it all secure, regardless of the application. EASE's biggest hurdle is the interoperation of IPSec and NAT. We proposed a simple, backward-compatible DHCP extension that provides full IPSec/NAT interoperation, including the AH and ESP protocols both in transport and in tunnel mode.

Acknowledgements

The authors thank Dale Blodgett, David Faucher and Dan Heer for their valuable input and many discussions on IPSec and NAT.

References

1. S. Alexander and R. Droms. “DHCP Options and BOOTP Vendor Extensions,” IETF, RFC 2132, Mar. 1997.
2. J. Brustoloni and J. Garay. “μISPs: Providing Convenient and Low-Cost High-Bandwidth Internet Access,” to appear in *Proc. 9th Intl. World Wide Web Conf.*, W3C, Amsterdam, Netherlands, May 2000.
3. Counterpane. “PPTP Crack,” available at `http://www.counterpane.com/pptp.html`.
4. W. Croft and J. Gilmore. “Bootstrap Protocol,” IETF, RFC 951, Sept. 1985.
5. S. Deering and R. Hinden. “Internet Protocol, Version 6 (IPv6) Specification,” IETF, RFC 2460, Dec. 1998.
6. N. Doraswamy and D. Harkins. “IPSec: The New Security Standard for the Internet, Intranets and Virtual Private Networks,” Prentice-Hall, 1st. ed., July 1999.
7. R. Droms. “Dynamic Host Configuration Protocol,” IETF, RFC 2131, Mar. 1997.
8. K. Egevang and P. Francis. “The IP Network Address Translator (NAT),” IETF, RFC 1631, May 1994.
9. FreeS/WAN. Homepage at `http://www.xs4all.nl/~freeswan/`.
10. A. Freier, P. Karlton and P. Kocher. “The SSL Protocol Version 3.0,” Netscape, Mar. 1996, available at `http://home.netscape.com/eng/ssl3/ssl-toc.html`.
11. J. Hardin. “Linux VPN Masquerade.” Homepage at `http://www.wolfenet.com/~jhardin/ip_masq_vpn.html`.
12. D. Harkins and D. Carrel. “The Internet Key Exchange (IKE),” IETF, RFC 2409, Nov. 1998.
13. Internet Software Consortium. Homepage at `http://www.isc.org/`.
14. S. Kent and R. Atkinson. “IP Authentication Header,” IETF, RFC 2402, Nov. 1998.
15. S. Kent and R. Atkinson. “IP Encapsulating Security Payload (ESP),” IETF, RFC 2406, Nov. 1998.
16. S. Kent and R. Atkinson. “Security Architecture for the Internet Protocol,” IETF, RFC 1825, March 1997.
17. T. König. “Ssh (Secure Shell) FAQ - Frequently asked questions,” available at `http://www.uni-karlsruhe.de/~ig25/ssh-faq/`.
18. Lucent InterNetworking Systems. Homepage at `http://www.lucent.com/dns/products/`.
19. C. Madson and R. Glenn. “The Use of HMAC-MD5-96 within ESP and AH,” IETF, RFC 2403, Nov. 1998.
20. C. Madson and R. Glenn. “The Use of HMAC-SHA-1-96 within ESP and AH,” IETF, RFC 2404, Nov. 1998.
21. C. Madson and N. Doraswamy. “The ESP DES-CBC Cipher Algorithm with Explicit IV,” IETF, RFC 2405, Nov. 1998.
22. D. Maughan, M. Schertler, M. Schneider and J. Turner. “Internet Security Association and Key Management Protocol (ISAKMP),” IETF, RFC 2408, Nov. 1998.
23. Microsoft. “Point-to-Point Tunneling Protocol (PPTP) FAQ,” available at `http://www.microsoft.com/NTServer/commserv/deployment/moreinfo/PPTPfaq.asp`.
24. J. Postel and J. Reynolds. “File Transfer Protocol,” IETF, RFC 959, Oct. 1985.
25. Y. Rekhter, B. Moskowitz, D. Karrenberg, G. J. de Groot and E. Lear. “Address Allocation for Private Internets,” IETF, RFC 1918, Feb. 1996.