

# Routing IPsec through NAT Gateways using Transport Mode

Jeremy Wright  
Arizona State University  
jlwright1@asu.edu

**Abstract**—Transport Mode IPsec provides a strongly secured means to connect two hosts together. Either the Authentication Header (AH) and Encapsulated Security Protocol can be used to implement varying levels of authentication and confidentiality depending on the communication need. However, when communicating on an IPv4 network, one must consider Network Address Translation (NAT) devices translating local IP addresses to global IP addresses. Without extra consideration this breaks end-to-end authentication. This paper discusses a UDP based solution to end-to-end confidential communication using IPsec transport mode.

**Index Terms**—Transport Mode, Tunnel Mode, IPsec, NAT-T, UDP

## I. INTRODUCTION

IPsec is a layer 3 end-to-end security protocol for communicating over insecure channels [1], [2]. As a layer 3 protocol IPsec, offers a key advantage over security protocols at higher levels of abstractions such as SSL/TLS. Application level security solutions require applications to be designed with security in mind. IPsec secures the underlying transport providing security services to any application. For some applications this is an enormous advantage, especially for established applications which cannot be extended to incorporate new security models. In this paper we will describe how to establish an end-to-end secure connection between two hosts as shown in Figure 1.

## II. IPSEC

IPsec defines two modes of operation transport mode, and tunnel mode. Primarily, transport mode is used to establish host to host communication, while tunnel mode is used to link two networks together, forming 1 larger network [3]. IPsec defines 2 header types the Authentication Header, and Encapsulated Security Protocol Header. Each header

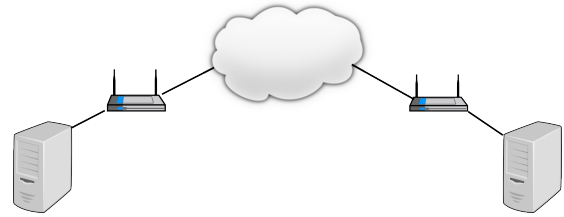


Fig. 1. Example Network with two NAT gateways

uses a HMAC based hash to preserve integrity of various fields within the packet, however when using NAT the integrity is broken by NAT mutating source and destination fields [4]. Thus each header authenticates different fields for its intended use case.

As shown in Figure 3, the Authentication Header provides no confidentiality since its default mode does not support encryption. Notice that the application developer could encrypt their own data, but this defeats the goal of IPsec providing a complete security model independent of the application.

Encapsulated Security Protocol Header (Figure 4) however, does provide confidentiality via an encrypted data payload. This packet additionally provides mutable fields excluded from the HMAC calculation. However this alone is insufficient to traverse two NAT gateways and simultaneously provide authentication and confidentiality. IPv6 networks NAT is not necessary since all addresses are globally routable. Within IPv4, NAT is required, but breaks end-to-end security since the NAT cannot change the source or destination fields, and also preserve the integrity of the packet. Tunnel mode provides a method for dealing with this, but does not fit all use cases.

Tunnel Mode uses the ESP header (Figure 5) to

0	1	2	3
0 1 2 3 4 5 6 7	0 1 2 3 4 5 6 7	0 1 2 3 4 5 6 7	0 1 2 3 4 5 6 7
Next Header		Payload Len	Reserved
Security Parameters Index			
Sequence Number			
Integrity Check Value			

Fig. 2. IPsec: Authentication Header

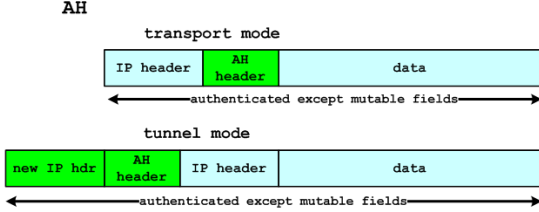


Fig. 3. Authentication Header [5]

wrap the desired IP packet. The encapsulated IP package is encrypted, and the outer header defines mutable, and immutable fields available for NAT [1]. While this addresses the NAT issue, it ignores the fact that tunnel mode is primarily intended for network to network connections.

### III. NAT-T TRAVERSAL

RFC3947 provides a mechanism to use IPsec transport mode through a NAT gateway [6]. However as noted in section 4 of this document, this doesn't fully deal with the problem as it pushes

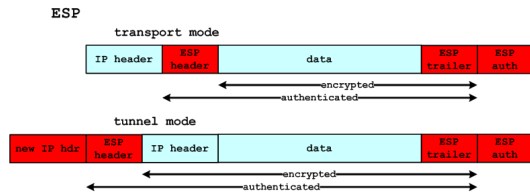


Fig. 4. Encapsulated Security Protocol Header [5]

0	1	2	3
0 1 2 3 4 5 6 7	0 1 2 3 4 5 6 7	0 1 2 3 4 5 6 7	0 1 2 3 4 5 6 7
Security Parameters Index			
Sequence Number			
Payload Data (Variable)			
Padding (Variable)			
Padding Length	Next Header		
Integrity Check Value			

Fig. 5. IPsec: Encapsulated Security Payload Header

0		1				2				3				UDP Header	Outer UDP Encapsulated Data																		
0	1	2	3	4	5	6	7	8	9	10	11	12	13			14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31
Source Port																Destination Port																ESP Header	
Length																Checksum																	
Security Parameters Index																																Client's UDP	
Sequence Number																																	
Source Port																Destination Port																	
Length																Checksum																	
Application Data																																	
00000000000000000000000000 (Padding)																																	
Padding Length								Next Header																									
Integrity Check Value																																	

Fig. 6. UDP Encapsulated Packet

requirements into the router software to function properly. "The best approach is simply to move the IKE traffic off port 500 as soon as possible to avoid any IPsec-aware NAT special casing [6]". This seems like a prime area for race conditions, and contention issues as multiple services queue for port 500 access. An alternative approach at the cost of some additional overhead is to encapsulate the IPsec payload into a UDP packet.

### IV. VPN WITH UDP ENCAPSULATION

UDP enables us to encapsulate the clients data in a standard packet type routable by any standard router. Any NAT in the transport path can mutate the fields of the outer packet without disrupting the integrity of the inner encapsulated data. We can use the ESP header to provide confidentiality and authentication as in Figure 6.

To establish our tunnel, we first must establish where the encapsulation packing and unpacking occurs. For the network defined in Figure 1, we have 3 options.

- 1) Encapsulate packets in the switch.
- 2) Encapsulate packets in the user's operating system.
- 3) Encapsulate packets in a userspace software application running on the endpoint.

For this use case we will assume the client is in a coffee shop or some similar public network, hence the client cannot modify routing information in the switch. Finally, the client wants to connect back to a desktop in a home office. For this use case, item 3 is most fitting. The user can start the vpn software, enter credentials and the software will establish the tunnel to the remote endpoint.

Since UDP will establish the tunnel we can use IPsec's transport mode. This provides a clean division of responsibilities. IPsec establishes security.

UDP establishes the tunnel. IPsec protects all higher level protocols (layers 4,5,6) without modification. Using an outer UDP packet, our software application encapsulates VPN bound packets like Figure 6. The outer IPv4 packet is free to be manipulated by any NAT within the pipeline. Whereas RFC 3947 requires a NAT discovery phase, this encapsulated packet behaves as an ordinary IPv4 datagram. When the remote host receives the outer IPv4 packet, it peels off the IPv4 and Outer UDP headers to reveal a fully authenticated ESP datagram. The application software then resumes routing the IPsec packet to its destination. This defines our end-to-end secure channel.

#### A. Routing our encapsulated header

To route our header we will assume each router in Figure 1 provides NAT services. The sender builds a datagram destined for the remote host, say an instant message professing the user's affinity for kittens (Figure 7). The application layer assembles the message as a UDP packet and sends the datagram over its socket. Our VPN software receives the packets since it's destined for the VPN secure connection.

The VPN software assembles an ESP header and encrypts the UDP packet (Figure 8). The IPsec packet is routable at this point, but will not traverse a NAT. To circumvent the address translation, the VPN software builds a UDP packet to encapsulate the ESP header (Figure 8). The UDP packet is then encapsulated into the IPv4 packet to route to the final network. The NAT is free to mutate it as required to get it to the destination. Per RFC 6864 the identification field is not used [7]. DSCP, and ECN similarly are not used. The local router receives the packet and modifies the source address in the outer IPv4 header. Since the outer IPv4 packet is not included in the encapsulated ESP header's hash, integrity is not disrupted. This is what prevented us from using the ESP header natively.

On the receiving end, the remote NAT accepts the outer IPv4 packet and translates the destination IP to an internal IP address via its internal tables. The remote host receives the datagram, and peels off the IPv4 and UDP headers. It uses the UDP key as internal information. The ESP header is verified for integrity (which verifies the fields and data). This integrity check is a key benefit of ESP. Lastly,

srcPort 80	dstPort 8080
6592 bytes	0
I like kittens	

Fig. 7. UDP Packet the user wishes to send via VPN

0	1	2	3	
Version	IHL	DSCP	ECN	Total length
Identification	Flags	Fragmentation Offset		
Time To Live	Protocol	Header Checksum		
	Source IP Address	Destination IP Address		
		Options		
Source Port 0x8080		Destination Port 0x80		
Length = 1024 bytes		Checksum = 0x0000		
Security Parameters Index = Internal Used Data				
Sequence Number = 1000				
[7UC9i3]bPZYURXBaphXXdudjHK43HNfuhXjW/TXkDjxgXedikvveBzn+1tLf0byBddk3wVTzFnozwbG3i/dsoo9E7kwj0yIuntzgpKNvFLu1XdhTbqJR46Z8N081Mj63tQUtE8CAU3yxuNDUIE+5CCd0kKOU RdIbxzMVcpAuSC0nFolmCfayjV3tg/eVQowjmbqLOoCkdhxXZYF4AtZILScWgiYj9rKXNeZD95VBSEcC2rGjFIVjTYE8doUgZ				
000000000000000000000000 (Padding)				
Padding Length	Next Header			
Integrity Check = 74ceb880da7d6394c2e1bfbd373c9edbd80b4aa1				

Fig. 8. IPv4 Sending the UDP Packet

the data gram is decrypted, and the resulting UDP packet is routed up to the application layer.

#### V. CONCLUSION

Traditionally, when configuring IPsec, one would use Tunnel Mode to connect to large networks together creating a larger virtual LAN. For remote users connecting to a publicly accessible server, transport mode may be more useful. However to create a secure connection between two end users in a peer-to-peer fashion, one can look at defining a new protocol based on UDP, to route their custom secure clients through Network Address Translators. Since this approach operates at Layer 3, it has the added benefit of providing secure communications to applications without existing security. Their traffic is successfully encapsulated into the ESP, and UDP headers to arrive at their destination, securely.

#### REFERENCES

- [1] K. Seo and S. Kent. (). Security architecture for the internet protocol, [Online]. Available: <https://tools.ietf.org/html/rfc4301> (visited on 03/15/2014).
- [2] *OSI protocols*, in *Wikipedia, the free encyclopedia*, Page Version ID: 593705554, Mar. 14, 2014. [Online]. Available: [http://en.wikipedia.org/w/index.php?title=OSI\\_protocols&oldid=593705554](http://en.wikipedia.org/w/index.php?title=OSI_protocols&oldid=593705554) (visited on 03/15/2014).

- [3] (). IPsec overview part two: modes and transforms > tunnel and transport modes, [Online]. Available: <http://www.ciscopress.com/articles/article.asp?p=25477> (visited on 03/15/2014).
- [4] K. B. Egevang and P. Srisuresh. (). Traditional IP network address translator (traditional NAT), [Online]. Available: <http://tools.ietf.org/html/rfc3022> (visited on 03/12/2014).
- [5] (). Ipsec.png, [Online]. Available: <http://www.deepsh.it/pix/ipsec.png> (visited on 03/18/2014).
- [6] V. V. <vvolpe@cisco.com>. (). Negotiation of NAT-Traversal in the IKE, [Online]. Available: <http://tools.ietf.org/html/rfc3947> (visited on 03/12/2014).
- [7] J. T. <touch@isi.edu>. (). Updated specification of the IPv4 ID field, [Online]. Available: <http://tools.ietf.org/html/rfc6864> (visited on 03/16/2014).