Routing IPsec through NAT Gateways using Transport Mode

Jeremy Wright
Arizona State University
jlwrigh1@asu.edu

***Abstract*—Transport Mode IPsec provides a strongly secured packet header called the Authentication Header (AH). This header provides integrity checking on all fields, and encapsulated data, but this packet cannot freely traverse a Network Address Translator (NAT). One alternative is to reduce the fields covered by the integrity checker to allow NAT to mutate fields as necessary. This however has other issues. Instead this paper proposes using Generic Routing Encapsulation to encapsulate the stronger AH.**

***Index Terms*—Transport Mode, Tunnel Mode, IPsec**

I. Introduction

IPsec is a layer 3 end-to-end security protocol for communicating over insecure channels [1], [2]. Being a layer 3 protocol is 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 doesn't require the application to have any knowledge of security. For some application types 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 to hosts over an insecure channel (Figure 3).

II. IPsec

IPsec defines two modes of operation transport mode, and tunnel mode. Each mode has it's own header type. Transport mode uses the Authentication Header (AH). Tunnel mode uses the Encapsulated Security Protocol Header (ESP). Transport mode connects two endpoints, whereas tunnel mode is typically used for connecting two networks together to form a larger virtual LAN [3]. IPsec meets the 4 tenants of security: authentication, non-repudiation, integrity, and confidentiality. However, when using NAT the integrity is broken by NAT mutating source and destination fields [4].

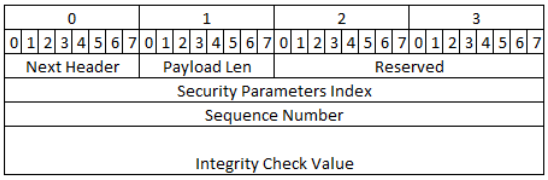

Fig. 1. IPsec: Authentication Header

Transport mode encrypts the entire IP packet, but keeps the routing information in plain text. The routing information is incorporated into the hash value protecting it from tampering. Since the routing information cannot be changed this causes problems for NAT traversal in IPv4 networks. 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 simultaneously change the source or destination fields, and 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 2) to 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.

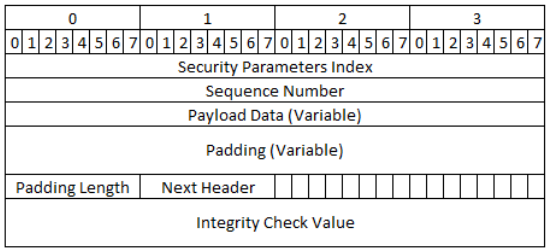

Fig. 2. IPsec: Encapsulated Security Payload Header

III. NAT-T Traversal

RFC3947 provides a mechanism to use IPsec transport mode through a NAT gateway [5]. However as noted in section 4 of this document, this standard doesn't fully deal with the problem as it pushes requirements into the router software in order 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 [5]". 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 use Generic Routing Encapsulation (GRE).

IV. VPN with Generic Routing Encapsulation

GRE is an encapsulation protocol defined in RFC2784. GRE can route any OSI Layer 3 packet such as IPsec [6]. GRE creates a point to point connection between two nodes. To establish our GRE tunnel, we first must establish where the encapsulation packing and unpacking occurs. For the network defined in Figure 3, 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.

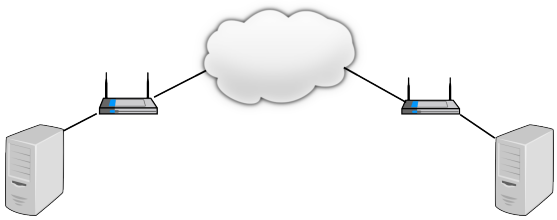

Fig. 3. Example Network with two NAT gateways

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 GRE will establish the tunnel we can use IPsec's transport mode. This provides a clean division of responsibilities. IPsec establishes security. GRE establishes the tunnel. IPsec protects all higher level protocols (layers 4,5,6) without modification. Using GRE, our software application encapsulates VPN bound packets like Figure 4. 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 GRE headers to reveal a fully authenticated AH IPsec header. 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 3 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 5). 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 AH header and encrypts the UDP packet (Figure 6). The IPsec packet is routable at this point, but will not traverse a NAT. To circumvent the address translation, the VPN software builds a GRE packet to encapsulate the AH header (Figure 7). The GRE packet is then encapsulated into the IPv4 packet to route to the final network. This IPv4 header becomes the "outer" IP packet, and 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

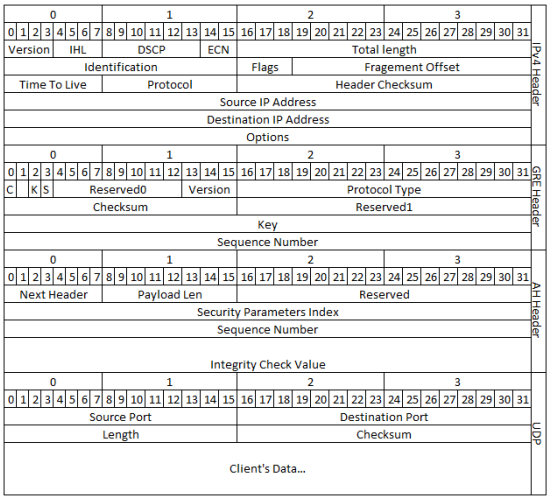

Fig. 4. GRE encapsulated IPsec packet

Fig. 5. UDP Packet the user wishes to send via VPN

the packet and modifies the source address in the outer IPv4 header. Since the outer IPv4 packet is not included in the encapsulated AH header's hash, integrity is not disrupted. This is what prevented us from using the AH 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 GRE headers. It uses the GRE key as internal information. The AH header is verified for integrity (which verifies the fields and data). This integrity check is a key benefit of AH, over ESP. Lastly, the data gram is decrypted, and the resulting UDP packet is routed up to the application layer.

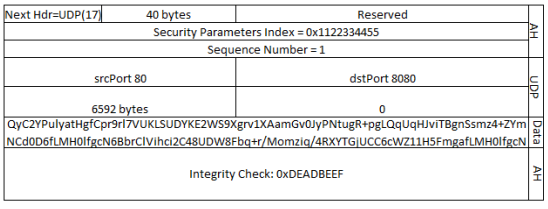

Fig. 6. AH encapsulated UDP Packet (Encrypted)

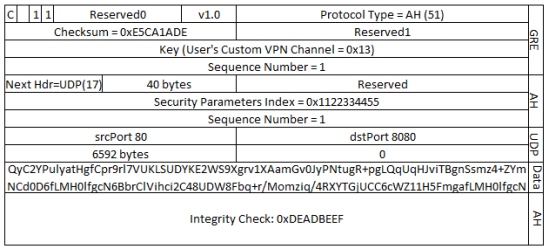

Fig. 7. AH encapsulated in GRE Packet

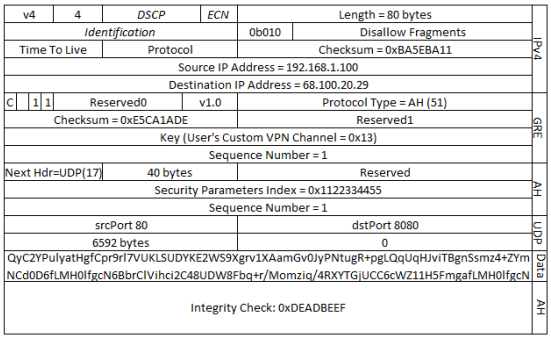

Fig. 8. IPv4 Sending the GRE Packet

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 the GRP, 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 AH, and GRE 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 (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] 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).

[6] S. Hanks, D. Meyer, D. Farinacci, and P. Traina. (). Generic routing encapsulation (GRE), [Online]. Available: http://tools.ietf.org/html/rfc2784 (visited on 03/15/2014).

[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).