Internet Protocol Security in Virtual Private Networks Connectivity

Ajaz AHMED
Newports Institute of Communications and Economics, Karachi

Muhammad Owais Zameer KHAN
Newports Institute of Communications and Economics, Karachi.

Sumaira Yousuf KHAN
University Of Karachi, Karachi

Talat Sharatat RAHMANI
NUST, Karachi

Muhammad JAMIL
University of Karachi.

Abstract

Internet Protocol Security (IPSec) is a widely deployed mechanism for implementing Virtual Private Networks (VPNs). IPSec, proposed by Internet Engineering Task Force (IETF), is a popular tunneling technology adopted for the Virtual Private Network It provides the network layer additional security functionality to transform multiple segments of a private network into one. This paper evaluates the performance overheads associated with IPSec. We not only explore implementation issues of IPSec over popular operating systems, such as Windows and Linux, but also provide performance evaluation of the proposed algorithms by a series of experiments.

Keyword

Internet Protocol Security (IPSec), Virtual Private Networks (VPNs), Internet Engineering Task Force (IETF)

1 Introduction

Authentication, authorization & accounting (AAA) for security provides a useful framework for the reasoning and measuring of the security capability of computer systems. Authentication provides a way for the identification of users. Authorization provides users privileges in executing specified tasks. Accounting concerns the auditing of user activities. There is a strong demand for tunneling technologies for the Virtual Private Network (VPN), where the Virtual Private Network (VPN) is an extension of a private network that encompasses network links across public networks (e.g., Internet). Within a VPN, authorized users are allowed to access various network-related services and resources. The Point-to-Point Tunneling Protocol (PPTP), the Layer 2 Tunneling Protocol (L2TP), and the IP Security Protocol (IPSec) are now the main tunneling technologies Adopted for VPN. IPSec was proposed by the Internet Engineering Task Force (IETF) to provide the network layer additional security functionality. A certain degree of communication security against eavesdropping, repudiation, and spoofing is provided. Although IPSec introduces a good flexibility in the security enhancement of higher-layer protocols, the setup of IPSec-based secured tunnels tends to be static in the configuration and restricted in a machine-to-machine fashion. While user mobility has become an important feature for many systems, technologies that provide users a lower cost and flexible way in joining a VPN are in a strong demand. An example approach in this direction is the adopting of the Ensemble system for group communication by having a single

shared encryption key for the entire VPN. The major problem for this approach is the vulnerability of the secured tunnels when some malicious user obtains the shared key.

Beside work on secured tunnels, researchers have exploited various approached for user and IP mobility with security support in the past decade. In particular proposed to modify an IPSec implementation to provide mobility, where a user must carry the same machine to have mobility support. We are interested in the extension of the IPSec implementations to have secured tunnels for users who might move around dynamically without carrying the same machine. With the proposed algorithm, the user mobility could be achieved. We shall not only identify threats against IPSec but also provide a lower-cost and flexible way in joining a virtual private network, compared to a full.

Scaled Public Key Infrastructure. An AAA-based negotiation procedure will be proposed for the implementations. The implementation of the proposed algorithm is simple and highly portable. We shall consider the implementation issues of IPSec over popular operating systems, such as Windows and Linux. The overheads of IPSec under different enabled security mechanisms would also be evaluated by a series of experiments.

With the identification of security objectives, we could better point out the security enhancement features that IPSec provides and explore more complex and advanced services over IPSec. Tradeoffs between performance/overheads and security objectives could be better understood. IPSec encrypts higher-layer protocols by having an authentication header (AH) and an encapsulated security payload (ESP). IPSec can be configured into two operating modes tunnel mode and transport mode with different security functionalities.

2 Literature Review

IPSec is a technology standard for implementing security features in Internet Protocol (IP) networking. IPsec network protocols support encryption and authentication. IPsec is most commonly used in so-called "tunnel mode" with a Virtual Private Network (VPN). However, IPsec also supports a "transport mode" for direct connection between two computers. Technically, IPsec functions at the network layer (Layer 3) of the OSI model. IPsec is supported in Microsoft Windows (Win2000 and newer versions) as well as most forms of Linux / Unix.

3. Methodology

IPSec involves many component technologies and encryption methods. Yet IPSec's operation can be broken down into five main steps. The five steps are summarized as follows:

Step1 Interesting traffic initiates the IPSec process—Traffic is deemed interesting when the IPSec security policy configured in the IPSec peers starts the IKE process.

Step 2 IKE phase one—IKE authenticates IPSec peers and negotiates IKE SAs during this phase, setting up a secure channel for negotiating IPSec SAs in phase two.

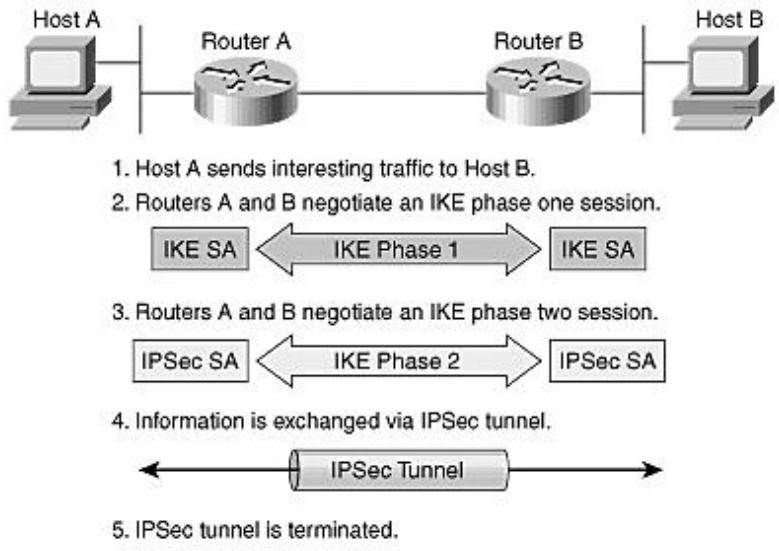
Step 3 IKE phase two—IKE negotiates IPSec SA parameters and sets up matching IPSec SAs in the peers.

Step 4 Data transfer—Data is transferred between IPSec peers based on the IPSec parameters and keys stored in the SA database.

Step 5 IPSec tunnel termination—IPSec SAs terminate through deletion or by timing out.

Step 1: Defining Interesting Traffic

Determining what type of traffic is deemed interesting is part of formulating a security policy for use of a VPN. The policy is then implemented in the configuration interface for each particular IPSec peer. For example, in Cisco routers and PIX Firewalls, access lists are used to determine the traffic to encrypt. The access lists are assigned to a crypto policy such that permit statements indicate that the selected traffic must be encrypted, and deny statements can be used to indicate that the selected traffic must be sent unencrypted. With the Cisco Secure VPN Client, you use menu windows to select connections to be secured by IPSec. When interesting traffic is generated or transits the IPSec client, the client initiates the next step in the process, negotiating an IKE phase one exchange.



Step 2: IKE Phase One

The basic purpose of IKE phase one is to authenticate the IPSec peers and to set up a secure channel between the peers to enable IKE exchanges. IKE phase one performs the following functions:

- Authenticates and protects the identities of the IPSec peers
- Negotiates a matching IKE SA policy between peers to protect the IKE exchange
- Performs an authenticated Diffie-Hellman exchange with the end result of having matching shared secret keys
- Sets up a secure tunnel to negotiate IKE phase two parameters

IKE phase one occurs in two modes:

- Main mode
- Aggressive mode

Main Mode

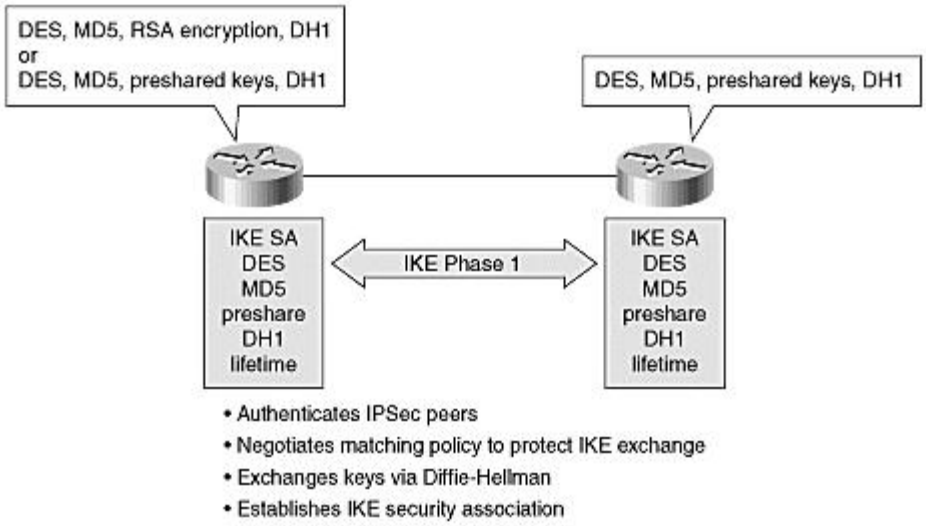
Main mode has three two-way exchanges between the initiator and receiver.

- First exchange The algorithms and hashes used to secure the IKE communications are agreed upon in matching IKE SAs in each peer.
- Second exchange This exchange uses a Diffie-Hellman exchange to generate shared secret keying material used to generate shared secret keys and to pass nonces, which are random numbers sent to the other party, signed, and returned to prove their identity.
- Third exchange This exchange verifies the other side's identity. The identity value is the IPSec peer's IP address in encrypted

form. The main outcome of main mode is matching IKE SAs between peers to provide a protected pipe for subsequent protected ISAKMP exchanges between the IKE peers. The IKE SA specifies values for the IKE exchange: the authentication method used, the encryption and hash algorithms, the Diffie-Hellman group used, the lifetime of the IKE SA in seconds or kilobytes, and the shared secret key values for the encryption algorithms. The IKE SA in each peer is bidirectional.

Aggressive Mode

In the aggressive mode, fewer exchanges are done and with fewer packets. In the first exchange, almost everything is squeezed into the proposed IKE SA values, the Diffie-Hellman public key, a nonce that the other party signs, and an identity packet, which can be used to verify the initiator's identity through a third party. The receiver sends everything back that is needed to complete the exchange. The only thing left is for the initiator to confirm the exchange. The weakness of using the aggressive mode is that both sides have exchanged information before there is a secure channel. Therefore, it is possible to sniff the wire and discover who formed the new SA. However, aggressive mode is faster than main mode.



Step 3: IKE Phase Two

The purpose of IKE phase two is to negotiate IPSec SAs to set up the IPSec tunnel. IKE phase two performs the following functions:

- Negotiates IPSec SA parameters protected by an existing IKE SA
- Establishes IPSec security associations
- Periodically renegotiates IPSec SAs to ensure security
- Optionally performs an additional Diffie-Hellman exchange

IKE phase 2 has one mode, called quick mode. Quick mode occurs after IKE has established the secure tunnel in phase one. It negotiates a shared IPSec policy, derives shared secret keying material used for the IPSec security algorithms, and establishes IPSec SAs. Quick mode exchanges nonce that provide replay protection. The nonce are used to generate new shared secret key material and prevent replay attacks from generating bogus SAs. Quick mode is also used to renegotiate a new IPSec SA when the IPSec SA lifetime expires. Base

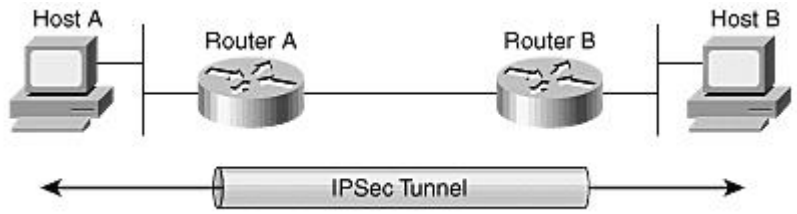
quick mode is used to refresh the keying material used to create the shared secret key based on the keying material derived from the DiffieHellman exchange in phase one.

Perfect Forward Secrecy

If perfect forward secrecy (PFS) is specified in the IPSec policy, a new Diffie-Hellman exchange is performed with each quick mode, providing keying material that has greater entropy (key material life) and thereby greater resistance to cryptographic attacks. Each DiffieHellman exchange requires large exponentiations, thereby increasing CPU use and exacting a performance cost.

Step 4: IPSec Encrypted Tunnel

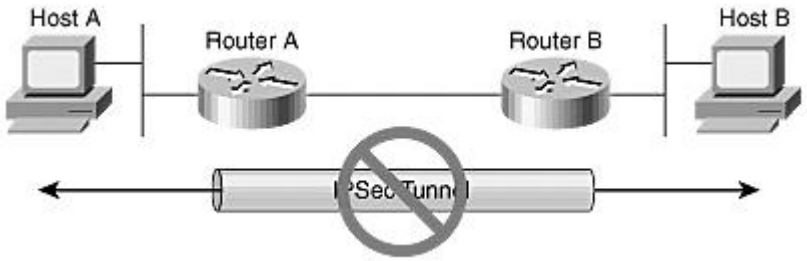
After IKE phase two is complete and quick mode has established IPSec SAs, information is exchanged by an IPSec tunnel. Packets are encrypted and decrypted using the encryption specified in the IPSec SA.



Step 5: Tunnel Termination

IPSec SAs terminate through deletion or by timing out. An SA can time out when a specified number of seconds have elapsed or when a specified number of bytes have passed through the tunnel. When the SAs terminate, the keys are also discarded. When subsequent IPSec SAs

are needed for a flow, IKE performs a new phase two and, if necessary, a new phase one negotiation. A successful negotiation results in new SAs and new keys. New SAs can be established before the existing SAs expire so that a given flow can continue uninterrupted.



IPSec Security Associations (SAs)

4 Conclusion

This paper exploits the extension of the IPSec implementations to have secured tunnels for users who might move around dynamically. Instead of having a full-scaled Public Key Infrastructure, a lower-cost and flexible way in joining a virtual private network is proposed. An AAA-based negotiation procedure is proposed for the implementations. The implementation of the proposed algorithm is simple and highly portable. We consider the implementation issues of IPSec over popular operating systems, such as Windows and Linux. The overheads of the proposed method were also evaluated by a series of experiments. For future research, we shall explore dynamic routing methods for security enhancement. Tradeoff between security enhancement and the balancing of networks workloads would be a focus of this study.

5 Recommendation

Our focus is to bridge this gap by presenting a new formal model that covers the semantics of a wide range of filtering policies including IPSec, and a sound and complete framework for analyzing IPSec policy conflicts. The verification framework utilizes OBDDs, a well-known powerful verification tool that is widely used in many fields, to represent IPSec policies and derive solid formulation of policy conflicts. Based on this framework, we developed techniques for identifying rule conflicts in IPSec policies of a single device or across multiple inter-connected devices.

6 Future Plan

Internet Protocol Security (IPSec) standard provides various flexible data protection schemes for IP networks, configuring IPSec policies manually can be extremely complex, particularly in large networks. An exhaustive analysis of policy rules in all IPSec gateways is required to discover policy conflicts and avoid serious network security threats like insecure transmission and flooding attacks. IPSec security, like any other technology, requires proper management support, including automatic conflict analysis and verification, in order to provide the required security services. Our approach is sufficiently general to be used for verifying many other filtering based security policies such as firewalls, intrusion detection systems and access control devices. We show that our implementation of these techniques in a tool called the "Security Policy Advisor" is very effective in checking real-life IPSec policies. For example, our tool was able to discover conflicts in IPSec policies that were overlooked by up to 29% of expert network administrators in our experiment. Our experiments have also shown that the average processing time in intra- and inter-policy conflict discovery is very reasonable for off-line analysis in many network configurations.

There is much more research to pursue in the automation of security policy management. Our future research plan includes.

References

[1]. KAME Project. http://www.kame.net/.

[2]. Common Criteria for Information Technology Security Evaluation (CCITSE), version 2.1.Technical Report CCIMB-99-032, National Institute of Standards and Technology, August 1999.http://csrc.nist.gov/cc/.

[3]. M. Berioli and F. Trotta. Ip mobility support for IPSecbasedvirtual private networks: an architectural solution.In IEEE Global Telecommunications Conference, 2003.

[4]. R. Bryant. Graph-based algorithms for Boolean function manipulation.
IEEE Transactions on Computers, C-35(8):677-691, August 1986.

[5]. J. Burch, E. Clarke, K. McMillan, D. Dill, and J. Hwang. Symbolicmodel checking: 1020 states and beyond. Journal of Informationand Computation, 98(2), 1992.

[6]. Cisco Systems. Configuring IPSec network security. In CiscoIOS Security Configuration Guide, Release 12.2, 2003.