VPN Development Based on IPSec Protocols Suit

Munthir B. Tuieb, Khalid K. Jabbar
Computer Science Department, College of Education, Al- Mustansiriyah University

ABSTRACT

Virtual Private Network (VPN) can be used over existing networks, such as the Internet; it can facilitate the secure transfer of sensitive data across public networks. This paper presents a method for VPN development based on IPSec protocols suit. The development method depends on existing packages which leads to reduce development time, which is the strongest most important criteria adopted in the process of development and Component Based Software Engineering (CBSE). Since IPSec work at network layer, so there is no need to change the security according to applications. An authentication is provided through PreShared Key technique.

The proposed method implemented using Java programming language and Linux operating system. Our proposed method depends on IPSec implementation in Linux kernel which provides Advanced Encryption Standard (AES) security feature, by focusing on the performance and design of the implementation. Furthermore, the existing paper focused on the same IPSec issues and overcomes them.

1- INTRODUCTION

There are three strategies can be used to achieve privacy which are private network, hybrid networks and virtual private networks. The cost is the major drawback of private and hybrid networks. An organization uses several leased lines when it connects several sites to create private Wide Area Networks (WANs), that means a high monthly fee, therefore WANs are expensive. The solution is using the Internet for both private and public networks. Virtual Private Network (VPN) is a technology allows organization using global Internet for both communications[1]. A VPN is used to transmit data privately and securely over an unsecured and public network infrastructure[2]. IP packets don’t provide the security inherently. As a result, receiving of IP packet isn’t guaranteed from the claimed sender, it contains the original data of the sender, or wasn’t sniffed when transit[3]. IPSec (abbreviated for IP Security) is used to protect IP datagrams, is often used in VPNs. It provides a method to specify the packet to protect, how that packet will be protected, and receiver of that packet. IPSec offers two protocols: Encapsulating Security Payload (ESP) or Authentication Header (AH) , for both IPv4 and IPv6. Internet Key Exchange (IKE) protocol is used by IPSec to establish and refresh the necessary cryptographic parameters for AH and ESP protocols[4].

T. Tsai identifies how IPSec affects the business strategy for Taiwan Internet Server Provider, using X company as an example [5]. A. Kundu proposes an extension to Multi-Layer IPSec protocol, on need basis the intermediate nodes provide access to upper layer header.
The Implementation of extended Multi-Layer IPSec uses Linux kernel of IPSec [6]. Adrian Hoban tests the potential performance that can be obtained by creating an AES-NI-GCM implementation using the new instructions within the Linux kernel cryptographic framework. AES-NI-GCM implemented using Assembly code. The data presented to demonstrate an AES-NI based on IPSec suit on Linux, using modern Intel microarchitecture running on Intel processors can deliver incredible improvements of IPSec performance over previous generations of silicon [7]. C. Shue, M. Gupta and S. Myers focus on evaluating of IPSec servers performance in many setting for client and show that IPSec is not scale as native TCP/IP implementation. A strong case is made from that analysis for optimizing the performance. In their paper, they focus on IKE optimization [8]. A. Alsheel and A. Almogren have covered the main aspects of IPSec protocol, including its relationship with other protocols such as Internet Key Exchange (IKE) and ISAKMP. They have identified IPSec architecture which includes the main components they have introduces its protocols AH and ESP and their differences. They have seen how IPSec implementation can be adopted into the TCP/IP stack in two forms as native or shim. They have dinged into Strong Swan one of the best IPSec implementation at Linux systems [9]. R. Hassan, A. Ahmed, N. Othman and S. Sami introduced a new spoofing defense mechanism to eliminate spoofing threat that happens when using IPSec in transport mode to secure IPv6 over IPv4 tunnel. A new mechanism work based on IPSec protocol ESP, it uses the padding area of ESP to write the IPv6 source address of the packet that will travel across IPv4 region [10].

The proposed development introduces some of IPSec problems and the solution for them. This paper proposes a method to enhance IPSec authentication through rekeying of Pre Shared Key technique. As well as introducing IPSec implementation inside Linux environment.

**IP Security (IPSec)**

IPSec secures Layer three IP communications. Employ a set of protocols and technologies such as Authentication Header (AH), Encapsulating security Payload (ESP), Internet Key Exchange (IKE), and others into a complete system that provides confidentiality and authenticity of IP data [11]. IPSec operates in one of two different modes, the transport mode or the tunnel mode. In transport mode, IPSec protects upper layer protocols. That is, the protection is applied to the payload of the IP packet, and the IP header is left intact. Typically this mode is used when end to end security is desired [12]. When IPSec tunnel mode is used, IPSec encrypts the IP header and the payload, the complete IP packet is protected and a new IP header is inserted to route from one IPSec endpoint to another [13]. Here the entire packet is protected. IPSec uses two protocols to provide traffic security Authentication Header (AH) and Encapsulating Security Payload (ESP). AH provides authentication and integrity, which protect against data tampering. AH also provides optional anti-replay protection, which protects against unauthorized retransmission of packets. AH does not protect the data’s confidentiality [14] [15]. ESP header is designed to provide a mix of security services.

ESP can be used to provide confidentiality, data origin authentication, connectionless integrity, an anti-replay service (a form of partial sequence integrity), and (limited) traffic flow confidentiality. ESP adds a header and trailer to packet [16]. A Security Association (SA) is a simplex “connection” that affords security services to the traffic carried by it. Security services are afforded to an SA by the use of AH, or ESP, but not both. If both AH and ESP protection is applied to a traffic stream, then two (or more) SAs are created to afford protection to the traffic stream. For two-way IPSec communications, two SAs are needed, one for each direction (i.e. inbound and outbound traffic) [17].

**IPSec Problems**

There are some problems using IPSec, this paper introduces a method to overcome these problems. First of all, Operating System (OS) vulnerabilities such as Domain Name System (DNS) attack. This attack can be utilizes by viruses or malware putting in the Operating System. The virus can alter the IP address of the honest peer with the wrong peer when DNS resolved. Therefore, wear using Linux OS to overcome this problem because this OS decrease effective of viruses for the reason the Linux OS does not support auto-run. Also there is no control by any applications on the system, because the kernel is responsible for the administration. Another issue, Since IPSec suite has many components and many protocols which made the developing of the system more difficult. In proposed method this issue is overcome by depending on existing packages such as Raccoon, in order to reduce the development time and simplified the development operation.

One of IPSec problems is that designer of IPSec uses algorithms to provide security which has already been cracked. This increases risks of security, especially when the administrator of network uses those algorithms unknowingly. Instead of more sophisticated, modern ones which are already available [18]. The proposed method used strong security algorithm (Rijndael Algorithm) to develop a system which is depends on long key and speed of execution. The proposed method is enhancing the authentication of IPSec using Pre Shared Key (PSK) technique depending on dynamic rekeying.

**Rijndael Algorithm**

The method depends on an Application Programming Interface (API) that provided by Linux kernel to fund cryptography services. There are many cryptographic services provided by API such as hashes, ciphers, random number generation and compression. The API provides both asynchronous and synchronous calling
semantics and is common to applications of kernel mode to use. In this paper the implementation of algorithms uses a plug-in model as framework of cryptographic. To implement API the application must call `crypto_register_alg()` function which points to a structure of `crypto_alg` (Rijndael) to be used. The Rijndael structure is defined to be used as cryptographic algorithm of the implementation. Rijndael is specified using `cra_name` member which founds in the structure of `crypto_alg`. Many plug-ins may simultaneously exist and have the same functionality. The method makes a call to a certain implementation by requesting the name of implementation. `cra_driver_name` member is provided by the structure `crypto_alg` which is used in our method to match the definition of names. When algorithm name is called by multiple implementations at the same time, then our method depends on `cra_priority` member to choose the implementation of cryptographic framework to be run[7].

Pre-Shared Key (PSK)
PSK is a shared secret which was previously shared between the two parties using some secure channel before it needs to be used. This algorithm works depending on IP address of peer, email address of peer or the domain name. It also requires password which must be the same at two computers. Authentication hash in aggressive mode of IKE is sending as response to the client that establishes IPSec session. This hash may capture using a sniffer, such as tcpdump, since this hash isn't encrypted then it possible start brute force attack or dictionary against this hash to recover the PSK. The static password can be cracked. So our proposed method is using dynamic password. This method depends on generating random password which is changing with the time. The proposed method enhance the authentication of IPSec, even the password cracked this password will change in later session. PSK takes places before SA negotiated, as a result, if PSK isn’t satisfied then SA negotiation will not begin. So IPSec session will not begin. This method makes IPSec attack so hard, because even if password that acts as gate to IPSec session cracked, the cracker can’t use the cracked password in next session. IKECrack is a tool available to do this job. The proposed method proves its immunity to crack using this tool.

Mode Algorithms
These algorithms are mainly used to specify the mode which defines the way to provide the security by IPSec protocols (AH and/or ESP). Racoon tool is used for development which save development time. Using this tool to provide the development to IPSec protocols which are Authentication Header (AH) and Encapsulating Security Payload (ESP), the next algorithm represent authentication header.

Algorithm 1: Authentication Header
Step1: Read data (Input)
Step2: Is data length even THEN
Add AH and exchange the contents of Next Header field and Protocol field of IP header.
ELSE
Add zeros as padding to make data even then GOTOStep2.
Step3: Set authentication data field empty.
Step4: Calculate authentication data.
Step5: Put the result in authentication data field (Output).
Step6: END.
The following algorithm represents Encapsulating Security Payload:

Algorithm 2: Encapsulating Security Payload
Step1: Read data (Input).
Step2: Is data length even THEN GOTO Step4 ELSE GOTO Step3
Step3: Add zeros as padding
Step4: Add ESP trailer and exchange contents of Next Header field and Protocol field.
Step5: Encrypt Transport layer data with ESP trailer using Rijndael algorithm
Step6: Add ESP header.
Step7: Calculate authentication data.
.Step8: Put the result of authentication at the end of packet (Output)
Step9: End.

Implementation
The proposed method implemented as an application runs by Linux, the application depends on security services that are found in the kernel of Linux. This will omit the transformations between applications and operating system, as a result this will enhance the speed of execution. Also the wrong settings can affect the performance of the application. The application is developed using Java, because it is an abstract computer defined by a specification. Java specification omits implementation details that are not part of the Java virtual machine’s specification. The main reason for this omission is to not unnecessarily constrain the creativity of implementers. Figure 1 shows testing for implementation of the application using Kame tool.

Figure 1: Checking application work using Kame tool

Results
The central processing unit (CPU) is typically the hardware component most affected by IPSec. The main drawback of IPSec is the CPU overhead, because
IPSec provides its security to required packets as well as to unwanted ones. The paper made IPSec application-dependent when it operates on fourth layer using Transport Mode to overcome processing unwanted ones. IPSec implementation typically provides protection for all traffic. In some cases, this may not be advisable because of performance reasons. The number of packets are reduced in the system, this leads to reduce IPSec processing, packet processing and the overload. Encrypting traffic that does not need protection or is already protected (e.g., encrypted by another application) can be a significant waste of resources. Two modes are used in this paper, with each connection type there is appropriate mode. This can affect the security of the system such as using Transport mode for connection between computers in different networks is less secure than Tunnel mode. At remote network the IP layer may has vulnerability could be passed across IPSec tunnel into the corporate network. It is doable to guarantee that this isn’t happening, but higher support results is expensive. By using Transport Mode the drives of network don’t appear to remote workers, the network is shielding against threats such as worms. The method enhances the security of IPSec protocol suite by using additional authentication technique. Pre-Shared Key is authentication technique that uses in the paper which depends on rekeying to reduce attack happening. The security comes from many resources, first of all, Operating System (Linux) that prevents auto-run and another application controlling and security packages. Wrapping algorithm (Rijndael) isn’t cracked anymore. The original protocol suit provides strong security as well as enhancing authentication using rekeying of Pre-Shared Key. The above factors make application provides strong security which overcomes some attack such as DNS attack, sniffer and brute force attack. Tcpdump and Kame tools are used to test robust of application security.

Conclusions
The proposed method provides additional level of authentication to that is provided by IPSec protocol, this leads to enhance the security. The proposed method depends on kernel of operating system to provide the security; this reduces time of execution, because it eliminates the time taken for transformation between operating system and application. The method prevents a normal user from work unless having (Root) permission, this will provide optimal security. The viruses that can exploit system vulnerability are less affective, since the system is developed with Linux operating system.

References