Analysis of Security Features in 5 Layer Internet Model

Suhas J Manangi
Department of Computer Engineering
National Institute of Technology
Karnataka
Surathkal, India

Parul Chaurasia
Department of Computer Engineering
National Institute of Technology
Karnataka
Surathkal, India

Mahendra Pratap Singh
Department of Computer Engineering
National Institute of Technology
Karnataka
Surathkal, India

Abstract: ***The Internet was originally conceived as an open, loosely linked computer network that would facilitate the free exchange of data. Security concept is relatively newer concept to that of TCP/IP suite or Layered Internet Model. But over the years security functionalities in Internet Model have evolved much faster than the model itself. Multiple security functionalities are available at each layer offering redundant services. This paper gives a brief description of all the security issues and available security features at each layer. Also proposes that combining and moving basic data security features to Transport Layer provides efficient end to end security thus to avoid redundancy in security services and to balance trade-off between security and performance. And other necessary network security features can be divided across layers.***

Key Words: Internet Model, Layered Architecture, SSL, SSH, TLS, IP Sec, Network Security, Cryptography, Security Threats.

I. INTRODUCTION

This paper deals with only security issues regarding 5 Layer Internet Model but not all aspects of Information Security. Generic issues to be taken care in Internet Model are: Authentication, Access Control, Data Confidentiality, Data Integrity, Nonrepudiation, Service and Network Availability. Today's Internet mainly suffers from these: Release of Message Contents, Traffic Analysis, Masquerade, Replay Attacks, Modification of Messages, and Denial of Service. And some of the common mechanisms to overcome above are: Encipherment, Digital Signature, Access Control, Data Integrity, Authentication Exchange, Traffic Padding, Routing Control and Notarization. This paper gives a brief about existing security protocols at each layer, and analyses on the drawbacks. And also proposes changes that should be made in Internet Architecture to balance tradeoff between security and performance. This paper mainly concentrates on data security in Internet Model.

II. OVERVIEW ON SECURITY PROTOCOLS

The Internet community has developed application-specific security mechanisms in a number of application areas, including electronic mail (S/MIME, PGP), client/server (Kerberos), Web access (Secure Sockets Layer), and others. However, users have some security concerns that cut across protocol layers. For example, an enterprise can run a secure, private TCP/IP network by disallowing links to untrusted sites, encrypting packets that leave the premises and authenticating packets that enter the premises. By implementing security at the IP level, an organization can ensure secure networking not only for applications that have security mechanisms but also for the many security-ignorant applications. The following gives brief description about security features at each layer. Figure 1, figure 2 and figure 3 show available security protocols at application layer, transport layer and network layer.

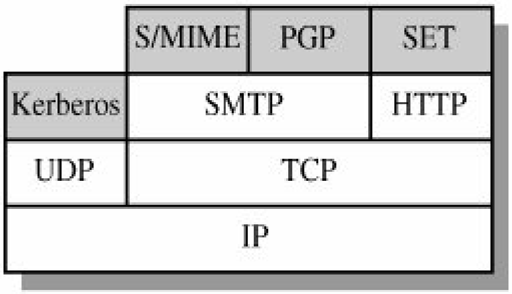

Figure 1: Security at Application Layer

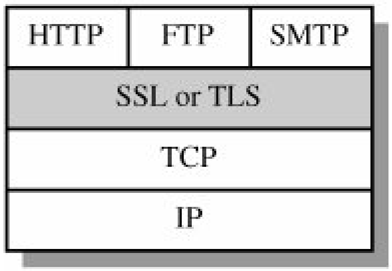

Figure 2: Security at Transport Layer

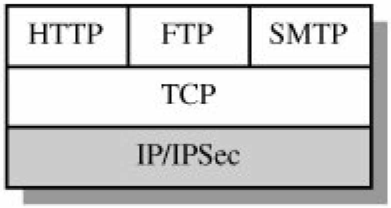

Figure 3: Security at Network Layer

A. Application Layer

SSH (Secure Shell) [1] [2]: SSH was originally defined for UNIX but now available for all platforms, it provides an authenticated and encrypted path to the shell or operating system command interpreter. SSH replace UNIX utilities such as Telnet, rlogin, and rsh for remote access. SSH protects against spoofing attacks and modification of data in communication. SSH protocol involves negotiation between local and remote sites for encryption algorithm (for example, DES, IDEA, AES) and authentication (including public key and Kerberos).

SSL (Secure Socket Layer) [3]: SSL protocol is placed between application layer and TCP layer in Internet Model. It provides server authentication, optional client authentication, and an encrypted communications channel between client and server.

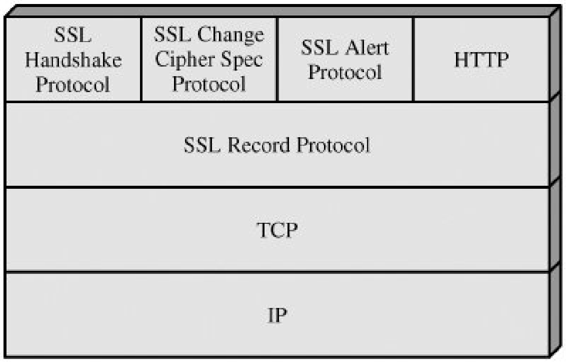

Figure 4: Placement of SSL

To use SSL, the client requests an SSL session. The server responds with its public key certificate so that the client can determine the authenticity of the server. The client returns part of a symmetric session key encrypted under the server's public key. Both the server and client compute the session key, and then they switch to encrypted communication, using the shared session key. Client and server negotiate a mutually supported suite of encryption for session encryption and hashing; possibilities include triple DES and SHA1, or RC4 with a 128-bit key and MD5. Figure 4 shows the placement and packet structure of SSL.

B. Transport Layer

TLS (Transport Layer Security) [4] [5]: The TLS protocol allows client/server applications to communicate across a network in a way designed to prevent eavesdropping and tampering. TLS provides endpoint authentication and communications confidentiality over the Internet using cryptography.

A TLS client and server negotiate a stateful connection by using a handshaking procedure. During this handshake, the client and server agree on various parameters used to establish the connection's security.

- The handshake begins when a client connects to a TLS-enabled server requesting a secure connection, and presents a list of supported CipherSuites (ciphers and hash functions).
- From this list, the server picks the strongest cipher and hash function that it also supports and notifies the client of the decision.
- The server sends back its identification in the form of a digital certificate. The certificate usually contains the server name, the trusted certificate authority (CA), and the server's public encryption key.
- The client may contact the server that issued the certificate (the trusted CA as above) and confirm that the certificate is authentic before proceeding.
- In order to generate the session keys used for the secure connection, the client encrypts a random number (RN) with the server's public key (PbK), and sends the result to the server. Only the server should be able to decrypt it (with its private key (PvK)): this is the one fact that makes the keys hidden from third parties, since only the server and the client have access to this data. The client

knows PbK and RN, and the server knows PvK and (after decryption of the client's message) RN. A third party may only know RN if PvK has been compromised.

- From the random number, both parties generate key material for encryption and decryption.

This concludes the handshake and begins the secured connection, which is encrypted and decrypted with the key material until the connection closes.

C. Network Layer

The most popular protocol used in the network layer is IP (Internet Protocol). The following are the key security risks at the Network Layer associated with the IP: IP Spoofing, Routing (RIP) attacks, Denial of Service attacks, ICMP attacks, PING Flood, Ping of Death Attack, Teardrop attack, Packet Sniffing. Most famous security protocol used along with IP protocol is IP Sec.

IP Sec [6]: Designed to address fundamental shortcomings such as being subject to spoofing, eavesdropping, and session hijacking, the IPSec protocol defines a standard means for handling encrypted data. IPSec is implemented at the IP layer. IPSec is somewhat similar to SSL, in that it supports authentication and confidentiality in a way that does not necessitate significant change either above it (in applications) or below it (in the TCP protocols). Like SSL, it was designed to be independent of specific cryptographic protocols and to allow the two communicating parties to agree on a mutually supported set of protocols. There are 2 modes of operation for IP-Sec protocol: Transport mode and Tunnel mode.

In Transport mode everything below IP header is encrypted (transport layer packet). It optionally authenticates the IP payload and selected portions of IP header.

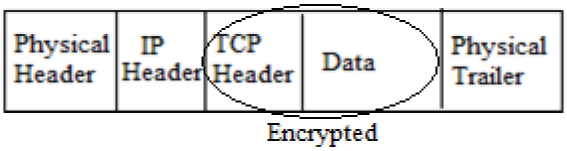

Figure 5: IP-Sec in Transport Mode

In Tunnel mode complete IP packet is encrypted. This is done by adding additional fields above IP header. The entire packet plus security fields is treated as the payload of new "outer" IP packet with a new outer IP header. The entire original, or inner, packet travels through a "tunnel" from one point of an IP network to another; no routers along the way are able to examine the inner IP header.

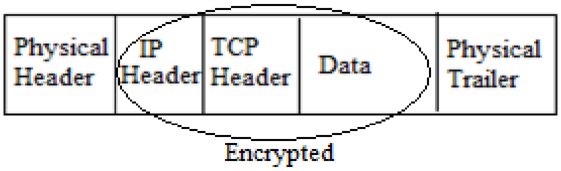

Figure 6: IP-Sec in Tunnel Mode

D. Data Link Layer

In Data Link Layer nothing much is done for data security and authentication process. But protocol is designed to take care of Content-Addressable Memory table overflow, VLAN hopping, Spanning-Tree Protocol manipulation attack, MAC address spoofing, ARP attack, Private LAN problems, and DHCP starvation.

III. ANALYSIS ON THE SECURITY PROTOCOLS

1. IP-Sec in Transport mode encrypts complete TCP/UDP packet. In case of TCP packet, each incoming packet at host is opened and then ordering and checking of checksum is carried out. Time for decryption is wasted if packet is a duplicate or with incorrect checksum. This affects performance and adds wasteful computations.
2. IP-Sec in Tunnel mode encrypts complete IP packet. Along with performance degradation caused by Transport mode, this mode has additional computation of constructing additional headers on top of the original IP packet which is encrypted now.
3. IP-Sec is almost similar in operation to that of SSL, so if IP-Sec is used over a packet which is running SSL, then additional computation needed is much more to additional security provided.
4. IP-Sec in Tunnel mode if implemented in routers and gateways, then this would provide only host to host security but not process to process or end to end security.

5. SSL needs extra messages to establish sessions and key exchange. These extra communications consumes bandwidth and communication time.
6. TLS is exactly same as SSL except padding schemes [7], inclusion/exclusion of key exchange methods and cipher suits, and MAC schemes. Thus offers completely redundant services, but generally either of SSL or TLS is used in communication.
7. With further advances in cryptography and cryptanalysis, stronger algorithms are designed and present algorithms might become obsolete. Though most protocols are designed such that they are independent of algorithms used to carry out the functionalities, yet many application level protocols are specific to methods and algorithms. In such cases, modifications are needed in protocol design and implementation else these applications become outdated and insecure. For e.g. telnet was taken over by SSH due to lack of security features in telnet.
8. Each layer has its own requirements in security, so while solving these often protocols are designed in a generic way but not to specific problems only.

Link Encryption	**End to End Encryption**
Security within Hosts	
Data exposed in sending host	Data encrypted in sending host
Data exposed in intermediate nodes	Data encrypted in intermediate nodes
Roles of user	
Applied by sending host	Applied by sending process
Invisible to user	User applies encryption
Host maintains encryption	User must find algorithm
One facility for all users	User selects encryption
Typically done in hardware	Either software or hardware implementation
All or no data encrypted	User chooses to encrypt or not, for each data item
Implementation concerns	
Requires one key per host	Requires one key per user pair
Provides node authentication	Provides user authentication

Table 1: Comparison between Link Layer security and Network Layer Security

IV. PROPOSED CHANGES IN INTERNET MODEL:

1. Clear identification of security requirements at each layer, and overlapping security requirements could be placed in only one layer.
2. Complete Data Confidentiality, Data Integrity, Authentication can be placed in Transport Layer[8][9] thus removing redundant parts in SSL, and IP Sec and other related protocols. TLS can be modified to accommodate these changes.
3. TCP and UDP can be modified to include TLS partially or completely.
4. On TCP it can have key negotiation and establishment along with connection handshake[10], and followed by secure communication. And optional header can be expanded to include communicate available cryptographic algorithms, key exchange mechanisms and other parameters required.
5. On UDP separate key establishment handshake message exchanges followed by secure communication.
6. Public Key Cryptography (Like Diffie Hellman Key Exchange [11]) can be used to establish key, and secure communication can be carried out using symmetric key cryptography.
7. Data Confidentiality, Data integrity, Authentication should be done only at Transport Layer and other layer should include only specific security needed at that layer. Like Network layers Routing attacks, or Data Link layers MAC spoofing etc.
8. Each upper layer should be able to communicate with below layers for what security features it wants from below layers. For instance if Application want no security features from TCP, there should be option to convey it to TCP layer and carry on communication with needed facility. If a application needs only data confidentiality but not data integrity or any other service then below layers should be able to provide only those services which are requested.
9. In TCP encryption should be applied only to packet part below sequence number. Thus duplicate packets can be dropped without decrypting them, this increases performance.

V. ADVANTAGES OF PROPOSED SYSTEM
VI.

1. The proposed suggestions in Transport layer provides balanced tradeoff between performance and security, optimum security with lesser performance degradation.

2. In TCP key establishment takes place along with initial handshake hence reduces extra key establishment message exchanges.
3. Redundancy in security service is reduced thus contributing to higher performance.
4. Applications could be made free from basic authentication and communication security. Thus rate of development of applications will increase.
5. With evolving security, applications need no constant up-gradation with respect to security features since only Transport layer needs to be upgraded. Thus longer life span for applications.
6. Since security features are divided into existing layer. Changes in one feature will not affect the other layers. Thus constant development in security of each layer can be taken care.
7. Each layer can specify services it requires from below layers, thus all layers can optimize to increase performance and security.

VI. DISADVANTAGES OF PROPOSED SYSTEM:

1. The network security measures at the data link layer are complementary to higher layers thus measures to be taken to provide extra protection of the network and users, especially in the case of wireless LAN.
2. Since each layer's requirements are well defined and common requirements are shifted to only one layer. There might be possibility that one layer can affect another layer because of its shortcomings in security services.
3. Since Internet infrastructure is already established, complete redesign of security features in each layer is difficult.

VII. CONCLUSIONS

This proposed model will balance tradeoff between performance and security. It provides very high results for general transactions in Internet which are mainly concerned with data security, data authentication, and also sender-receiver authentication compared to the transactions requiring very high access control needs. Each layer has necessary security features and each layer can negotiate with below layers for customized services it needs thus optimizing layer services and increasing performance while balancing security.

REFERENCES

[1]. Daniel J. Barrett, Richard E. Silverman, and Robert G. Byrnes – SSH: The Secure Shell (The Definitive Guide), O'Reilly 2005 (2nd edition). ISBN 0-596-00895-3
[2]. Secure Shell (SSH) RFC 4250, 4251, 4252, 4253, 4344
[3]. Wagner, David; Schneier, Bruce (November 1996). "Analysis of the SSL 3.0 Protocol". The Second USENIX Workshop on Electronic Commerce Proceedings.
[4]. Dierks T. and E. Rescorla. "The Transport Layer Security (TLS) Protocol Version 1.1, RFC 4346".
[5]. *RFC* 2246: The *TLS* Protocol Version 1.0 and *RFC* 4346: The *TLS* Protocol Version 1.1
[6]. RFC : IP-Sec 2401, 2412
[7]. Stephen A. Thomas (2000). SSL and TLS essentials securing the Web. New York: Wiley. ISBN 0-471-38354-6.
[8]. Security assessment of the Transmission Control Protocol (TCP) Technical notes archive ID: 00003 Ref: TN0309 Date: February 2009
[9]. RFC: 793 TRANSMISSION CONTROL PROTOCOL
[10]. "Secure TCP - providing security functions in TCP layer" INET'95 Paper no:144
[11]. RFC 2631 – Diffie–Hellman Key Agreement Method E. Rescorla June 1999.

AUTHORS PROFILE

Suhas J Manangi is pursuing his Bachelors in Computer Engineering at National Institute of Technology Karnataka, Surathkal (India). His research interests are Computer Networks, Cryptography and Network Security.

Parul Chaurasia is pursuing her Bachelors in Computer Engineering at National Institute of Technology Karnataka, Surathkal (India). Her research interests are Algorithms, Computer Networks, Cryptography and Network Security.

Mahendra Pratap Singh received B.Tech. degree in Computer Science & engineering from UPT University, Lucknow (U.P) and M.E. degree in Computer Science & engineering from Karunya University ,Coimbatore(T.N), India. From January 2009, he is with the Department of Computer Engineering, National Institute of Technology Karnataka, Surathkal, Mangalore, working as Assistant Professor. His research interests include Information Security, Cryptography, Image Processing, Cloud Computing and Wireless Sensor Network.