Application Layer Security Issues and Its Solutions

Raghavendra K[1], Sumith Nireshwalya [2]

[1]Dept of CS&Engg,P.A.C.E
raghu356 @gmail.com

[2]Dept of CS&Engg, S.I.T
sumith.srinath@gmail.com

Abstract: Every layer of communication has its own unique security challenges. The application layer communication is a very weak link in terms of security because that the application layer Application Layer security is a growing area of concern for developers, designers, quality assurance specialist and programmers. In this paper we discuss about various security aspects at application layer and its solution.

I INTRODUCTION

OSI REFERENCE MODEL

The OSI, or Open System Interconnection, model defines a networking framework for implementing protocols in seven layers. Control is passed from one layer to the next, starting at the application layer in one station, and proceeding to the bottom layer, over the channel to the next station and back up the hierarchy[1].

Application layer

This layer supports application and end-user processes. Everything at this layer is application-specific. This layer provides application services for file transfers, e-mail, and other network software services.

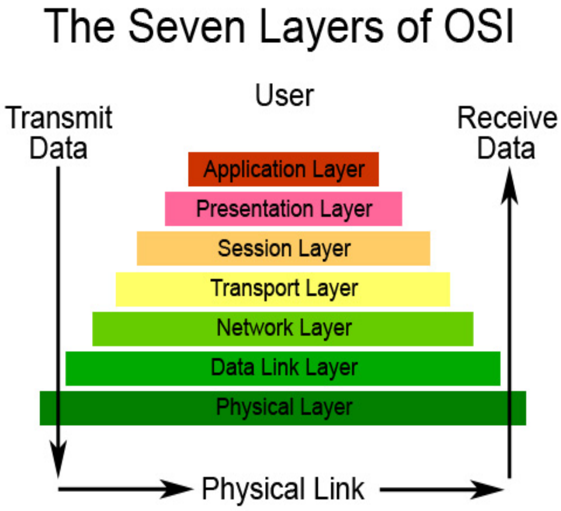

Figure 1: The Seven Layers of OSI

Telnet and FTP are applications that exist entirely in the application level. Tiered application architectures are part of this layer.

Presentation layer

This layer provides independence from differences in data representation by translating from application to network format, and vice versa. The presentation layer works to transform data into the form that the application layer can accept.

Session layer

This layer establishes, manages and terminates connections between applications. The session layer sets up, coordinates, and terminates conversations, exchanges, and dialogues between the applications at each end. It deals with session and connection coordination.

Transport layer

This layer provides transparent transfer of data between end systems, or hosts, and is responsible for end-to-end error recovery and flow control. It ensures complete data transfer.

Network layer

This layer provides switching and routing technologies, creating logical paths, known as virtual circuits, for transmitting data from node to node. Routing and forwarding are functions of this layer, as well as addressing, internetworking, error handling, congestion control and packet sequencing.

Data Link layer

At this layer, data packets are encoded and decoded into bits. It furnishes transmission protocol knowledge and management and handles errors in the physical layer, flow control and frame synchronization. The data link layer is divided into two sub layers: The Media Access Control (MAC) layer and the Logical Link Control (LLC) layer.

Physical layer

This layer conveys the bit stream - electrical impulse, light or radio signal through the network at the electrical and mechanical level. It provides the hardware means of sending and receiving data on a carrier, including defining cables, cards and physical aspects. Fast Ethernet, RS232, and ATM are protocols with physical layer components.

II. THE PROBLEM

Suppose that we apply good security through the underlying layers(1 and 2 layer), with physical isolation (layer 1),private VLANs (layer two), and firewalls with tight packet filter policies (layers 2 and 3). But then we are deficient on our application layer security (layer seven, and often layers six and five), using unpatched server software and poorly written application and script code. Since the vulnerabilities lie within the application, in a pure seven-layer model we would be hard pressed to defend against this at the lower levels.

III. WHAT HAPPENS AT APPLICATION LAYER?

Occupying the top-end of the stack, the Application Layer is the most open ended of all of the layers, and can be considered the catchall for any issues not addressed within the other six layers. Taking a more narrow view from a protocol perspective, user-oriented protocols such as naming (DNS, WINS), file-transfer (HTTP, FTP),messaging(SMTP, TOC/OSCAR[used by AIM]), and access (Telnet, RDP) all fall within the Application Layer in a more strict interpretation that views even higher level functions as outside the model completely. For the purposes of information security, the Application Layer can be considered the realm where user interaction is obtained and high-level functions operate above the network layer. These high level functions access the network from either a client or server perspective, with peer-based systems filling both functions simultaneously[6].

The open-ended nature of the Application Layer may present threats.[3] Some of the threats can be summarized as follows:

- One of the prime threats at the Application Layer is poor or nonexistent security design of the basic function of an application.
- Some applications may insecurely handle sensitive information by placing it in publicly accessible files or encoding it in “hidden” areas which are trivially displayed, such as in the HTML code of a web form.
- Programs may have well-known backdoors or shortcuts that bypass otherwise secure controls and provide unauthorized access.
- Applications with weak or no authentication are prime targets for unauthorized use and abuse over the network.
- The TFTP protocol is extensively used for booting of diskless workstations and network device management, but does not require any sort of username or password authentication to use its file access ability, giving an intruder possible access to configuration and access information without challenge other than the need to guess filenames.
- Applications may rely upon untrustworthy channels to establish identity or set privilege.
- Overly complex access controls may seem to protect access but fail to prevent unauthorized activity due to poorly understood or written access rules.

IV. PROVIDING SECURITY AT THE APPLICATION LAYER

The following steps are been in practice to make the application layer safer.

A. Use methods from applications

From the higher levels outside of the model, user input is a significant threat from both deliberate and accidental standpoints. Users may provide unexpected input into the application environment, which if not handled properly could lead to crashes or other unexpected behavior. The unsuspecting hapless user may cause his application to crash or otherwise fail. A malicious user may be able to use bugs and program flaws to attack and gain access to resources or data. Some of the most prevalent controls at the application layer relate to strong design practices in application design and implementation.

Applications should make use of the secure facilities available to them in the lower network layers, carefully check incoming and outgoing data, and assume that communications can and will be subject to attack, requiring the use of strong authentication and encryption to validate and protect data as it travels across the network. Applications should also implement their own security controls, allowing for fine-grained control of privilege to access resources and data, ideally using a mechanism that is straightforward and strikes a balance between usability and effectiveness.

Detailed logging and audit capability should be a standard feature of any application that handles sensitive or valuable data. Testing and review is also critical as a control for the application layer. Given the wide variety of both problems and solutions, standards and practices will not be able to capture all possible twists and turns in the application environment. Developers will often have conflicting motivations and agendas regarding their applications, and in a structured programming environment, mandated code security review and application security testing are critical parts of a secure Software Development Life Cycle (SDLC).

B. Use hardware security

On the hardware front, Intrusion Detection Systems (IDS) can observe data traffic for known profiles of network activity that can indicate probes for vulnerable applications or an imminent or ongoing attack, as well as detecting the presence of undesirable application traffic[8].

Many current host-based firewall systems also include the means to control the access of applications to the network. This control is useful in preventing the unauthorized or covert use of network resources by local programs, as well as providing the conventional layer three and four control functions of a firewall. Many also include basic IDS functionality as well.

C. Role of radius protocol

Authentication, Authorization and Accounting (AAA) protocols such as TACACS [TACACS] and RADIUS [RADIUS] were initially deployed to provide dial-up PPP [PPP] and terminal server access.[5]Over time, with the growth of the Internet and the introduction of new access technologies, including wireless, DSL, Mobile IP and Ethernet, routers and network access servers (NAS) have increased in complexity and density, putting new demands on AAA protocols. RADIUS provides the following advantages:

- Tight security : RADIUS allows user information to be stored on one host, minimizing the risk of security loopholes.

- Flexibility: Using modifiable "stubs," RADIUS can be adapted to work with existing security systems and protocols. The
- RADIUS server may be adapted to your network, rather than adjusting your network to work with RADIUS.
- Simplified management: Security information is stored in text files at a central location, the RADIUS server. Adding new users to the database or modifying existing user information can be easily accomplished by editing these text files.
- Extensive logging capabilities: RADIUS provides extensive audit trail capabilities, referred to as RADIUS accounting. Information collected in a log file can be analyzed for security purposes, or used for billing.

Security aspect is better handled in RADIUS version 2.0 which provides SecurID.[9] The SecurID authentication is based on Security Dynamics' token technology, which authenticates users using a patented time-synchronization method.The RADIUS 2.0 server can forward some or all authentication requests to a SecurID ACE/Server running on the same host as the RADIUS server.

D .Role of diameter protocol

All data delivered by the diameter protocol is in the form of an Attribute-Value Pair. Some of these AVP values are used by the Diameter protocol itself, while others deliver data associated with particular applications that employ Diameter. AVPs may be added arbitrarily to Diameter messages, so long as the required AVPs are included and AVPs that are explicitly excluded are not included. A security association which is an association between two endpoints in a Diameter session is used for security. It allows the endpoints to communicate with integrity and confidentially, even in the presence of relays and/or proxies.

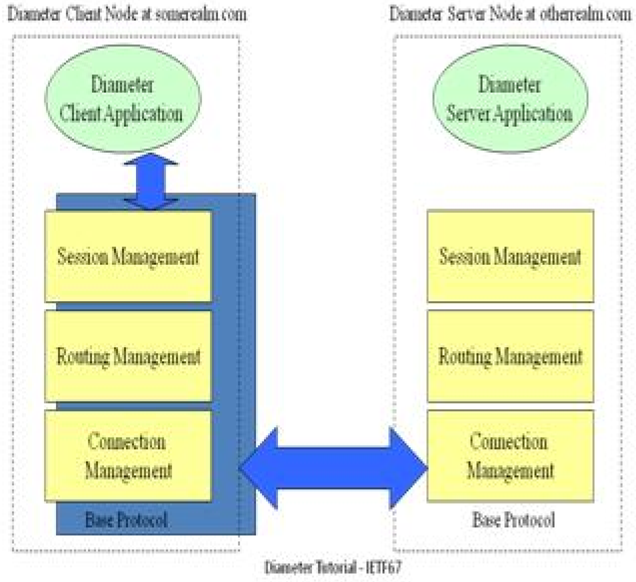
Diameter Client Node at somerealm.com
Diameter Server Node at otherrealm.com
Diameter Client Application
Diameter Server Application
Session Management
Routing Management
Connection Management
Base Protocol
Session Management
Routing Management
Connection Management
Base Protocol
Diameter Tutorial - IETF67

Figure 2: Diameter basic functionality

Diameter base protocol also provides End-to-End Security Framework. End-to-end security services include confidentiality and message origin authentication. These services are provided by supporting AVP integrity and confidentiality between two peers, communicating through agents. End-to-end security is provided via the End-to-End security extension, described in [AAACMS]. The circumstances requiring the use of end-to-end security are determined by policy on each of the peers. Security policies, which are not the subject of standardization, may be applied by next hop Diameter peer or by destination realm. For example, where TLS or IPsec transmission-level security is sufficient, there may be no need for end-to-end security.[4] Diameter requires transmission level security to be used on each connection (TLS or IPsec). Therefore, each connection is authenticated, replay and integrity protected and confidential on a per-packet basis.In addition to authenticating each connection, each connection as well as the entire session MUST also be authorized. Before initiating a connection, a Diameter Peer MUST check that its peers are authorized to act in their roles. For example, a Diameter peer may be authentic, but that does not mean that it is authorized to act as a Diameter Server advertising a set of Diameter applications.

V. COMPARATIVE STUDY

Comparative study on the above mentioned approaches are summarized in table 1.

Threats handled	**codes in application (A)**	**Hardware security (B)**	**RADIUS (C)**	**Diameter (D)**
Input validation	Yes	No	No	No
Buffer overflow	Yes	No	No	no
Configuration management	Yes	Yes	Yes	Yes
Eaves dropping/data tampering	No	No	Yes	Yes
Session hijacking, MITM	No	No	Yes	Yes
DoS	No	No	Yes	Yes
Auditing & logging	No	Yes	Yes	Yes
Query string manipulation	Yes	Yes	Yes	Yes
Level	Low	Medium	High	High

Table 1: comparative study on different approaches for securing application

VI CONCLUSION

When you incorporate security features into your application's design, implementation, and deployment, it helps to have good understanding of threats. In this paper study on various aspects of application security is been done. Its pros and cons are evaluated in the table 1 Method A and B provide to a maximum of medium security. Since our applications are changing rapidly we suggest Method C and D to be implemented to provide high security. Method D, use of diameter protocol have extensible features that can be used on any existing system with minimal change.

REFERENCES

[1]. Applying the OSI Seven Layer Network Model To Information Security- SANS institute infosec reading room
[2] Microsoft patterns & practices- chapter 2 –threats & countermeasures, 2006
[3]Communicationsecurity.html- communication security at application layer
[4]Interlink networks-introduction to diameter protocol
[5]Wikipedia- introduction to radius
[6] Aplication layer security by john rouda, july,25, 2006
[7] Diameter WebAuth: An AAA-based identity Management Framework for Web Applications by Niklas Neumann
[8] comTIA Network+ study guide by todd Lammle

ACKNOWLEDGMENT

Prof. Raghavendra K and Prof. Sumith thank Dr. Udaykumar Shenoy for his valuable suggestions that helped in the completion of the paper.

ABOUT THE AUTHORS

Prof Sumith Nireshwalya holds B.E in information science & engineering and M.Tech in Computer science & engineering. She is presently working in the department of Computer science and Engineering as assistant professor.

Prof.Raghavendra holds B.E in computer science & engineering and M.Tech in Computer networks & engineering. He is presently working in the department of Computer science and Engineering as assistant professor.