A SIP based VOIP to avoid Vulnerabilities in designing VOIP network in Enterprise

K.Subhash Bhagavan #1, Kirankumar.P #2, MVSS Nagendranath#3,

#1 Student, Sasi Institute of Technology and Engineering, Tadepalligudem, W.G(dt)

#2 Asst. professor, Sasi Institute of Technology and Engineering, Tadepalligudem, W.G(dt)

#2 Assoc. professor,HOD, Sasi Institute of Technology and Engineering, Tadepalligudem, W.G(dt)

#1Subhash.kommina@gmail.com,#2kiran@sasi.ac.in,hodcse@gmail.com

Abstract— The increasing demand of VoIP and its support to internet made it as a mainstream and being implemented with a large number of service providers and enterprise networks. The integration of security standards with SIP based VoIP we need to check the effects of firewall and VPN techniques which should maintain quality to the business environment. The main goal is to understand the capabilities and to identify gaps in addressing the vulnerabilities in present VoIP systems. The specific problem like (Denial of Service (DoS) and Service Abuse) are major vulnerabilities considered during implementation of VoIP systems in enterprise. In this paper, we address the issue of denial of service attacks and its vulnerabilities which targeting the hardware and software of voice over IP servers or by misusing specific signaling protocol features. As a signaling protocol we investigate here the Session Initiation Protocol.

Index Terms—voice over Internet Protocol (VoIP), Session Initiation Protocol (SIP), Denial of Service (DoS).

I INTRODUCTION

VOICE OVER Internet Protocol (VoIP) is one of the fastest growing Internet applications. VoIP is a technology that allows users to make telephone calls using a broadband Internet connection instead of an analog phone line. VoIP holds great promise for lowering the cost of telecommunications and increasing the flexibility for both businesses and individuals. VoIP leverages existing IP-based packet-switched networks to replace the circuit-switched networks used for voice communications since the invention of the telephone as shown in figure 1.

In an open environment such as the Internet, mounting an attack on a telephony server is, however, much simpler. This is due to the fact that voice over IP (VoIP) services are based on standardized and open technologies using servers reachable through the Internet, implemented in software and provided often over general-purpose computing hardware. Therefore, such services can suffer from similar security threats as HTTP-based services. Instead of generating thousands of costly voice calls, the attacker can easily send thousands of

VoIP invitations in a similar manner to attacks on Web servers. These attacks are simple to mount and, with flat rate Internet access, are also cheap.

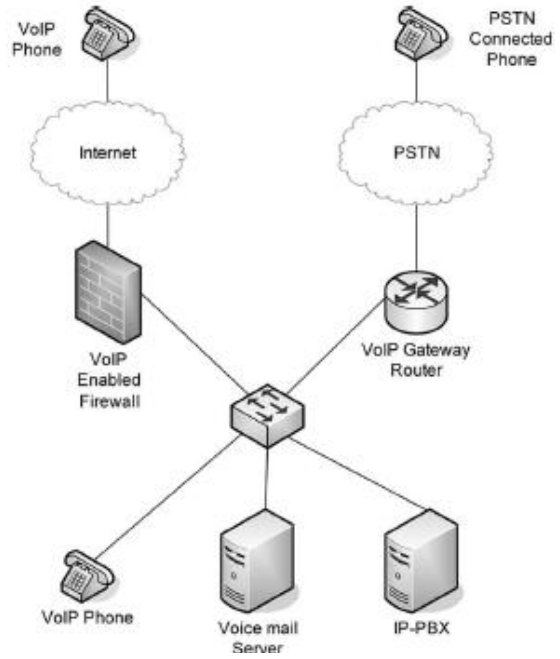

Fig. 1. (a) Typical VoIP network structure.

Denial-of-Service (DoS) attacks are explicit attempts to disable a target thereby preventing legitimate users from making use of its services. DoS attacks continue to be the main threat facing network operators. As telephony services move to Internet Protocol (IP) networks and Voice over IP (VoIP) becomes more prevalent across the world, the Session Initiation Protocol (SIP) infrastructure components, which form the core of VoIP deployments, will become targets in order to disrupt communications, gain free services, or simply to make a statement. Since DoS attacks are attempts to disable the functionality of the target, as opposed to gaining operational control, they are much more difficult to defend against than traditional invasive exploits, and are practically impossible to eliminate.

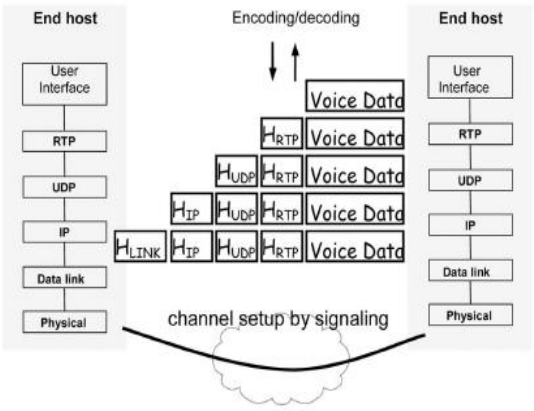

Fig. 1: (b) Voice data processing of the VoIP.

To make things worse, attackers have developed tools to coordinate distributed attacks from many separate sites, also known as distributed denial of service (DDoS) attacks. Besides launching brute force attacks by generating a large number of useless VoIP calls, attackers can use certain features of the used VoIP protocol to incur higher loads at the servers. This might involve issuing requests that must be authenticated, require database lookups by the VoIP servers, or cause an overhead at the servers in terms of saved state information or incurred calculations.

In this paper, session initiation protocol is used for investigating possibilities of launching denial of service attacks on SIP servers and proposes preventing ways which reduce the effects of such attacks. The Session Initiation Protocol (SIP) is establishing itself as the standard for VoIP services in the Internet and next generation networks. SIP is a text-based protocol designed to establish or terminate a session between two partners. The message format is similar to HTTP, with message headers and corresponding values.

II RELATED WORK

The reliable performance of SIP server is critical under DoS attacks. There has been previous effort to protect VoIP deployments from DoS threats. An early evaluation of firewalls for VoIP security

was proposed, but it lacked concrete architectural and implementation aspects. A mitigation strategy for flooding DoS attacks on media components using a dynamic pinhole filtering device that blocks all traffic not associated with a legitimate call was previously developed as part of an earlier phase of this research.

Wu, Y. et al. and Niccolini, S. et al. have applied intrusion detection and prevention mechanisms to safeguard the SIP infrastructure, while the work makes use of finite state machines to achieve similar goals. An interesting approach involving VoIP "honeypots" was proposed. Extensive work on detecting DoS attacks on IP telephony environments has been published. Although promising, none of the architectures and algorithms proposed so far offer a comprehensive DoS mitigation strategy that scales up to the performance needs and complexity of carrier-class VoIP deployments, because they are based on software solutions.

B. Bencsath et al. have empirically evaluated the SMTP servers against DoS attacks. An important work is reported, which the authors conceptually discussed the impact of different types of attacks on VoIP infrastructure. They have conceptually identified exploitable server resources, such as memory, CPU usage and bandwidth and presented abstract guidelines to ensure SIP servers' robustness under different attack scenarios. But they paid no attention to empirically analyze the performance hit of SIP servers under attack.

S. McGann et al. summarize the features of vulnerability analysis tools available for VoIP and suggest using a Virtual Private Network (VPN) solution to circumvent attacks on a SIP server. The authors did not discuss how their scheme is resilient against DoS attacks

E. Nahum et al. have experimentally evaluated the SIP proxy (OpenSER), using micro-benchmarks,and analyzed the performance of OpenSER as a function of selecting different configuration modes of the server. They have also ignored robustness analysis of SIP servers against different types of DoS attacks.

III BASIC RESOURCES

The majority of DoS attacks are based on exhausting some of a server's resources and causing the server not to operate properly due to lack of resources. With SIP servers, there are three resources needed for operation: memory, CPU and bandwidth.

Memory

A SIP server needs to copy each incoming request in its internal buffers to be able to process the message. The amount of buffered data and the time period the server is supposed to keep the buffered data varies depending on whether the server is working in a stateful or stateless mode. The size of a SIP message might range from a few hundreds of bytes up to a few thousands.

CPU

After receiving a SIP message, the SIP server needs to parse the message, do some processing (e.g., authentication), and perform transaction mapping and forward the message. Depending on the content and type of the message and server policies, the actual amount of CPU resources might vary. Whereas the CPU capacity of a well engineered and configured proxy should be able to process SIP messages up to link capacity, there are many server operations that make servers block.

Bandwidth

This involves overloading the access links connecting a SIP server to the Internet to such a level as to cause congestion losses. By overloading the server's access links, one could cause the loss of SIP messages which causes longer session setup times or even the failure of session setups.

IV DOS ATTACKS AND COUNTER MEASURES

A) Memory Based Attacks:

State maintenance in SIP servers is one of easier targets for DoS attacks. Measurements indicate that a stateful server flooded with a continuous stream of requests belonging to different transactions will run out of memory very quickly. Basic attacks are:

Brute force attacks: The simplest method for mounting an attack on the memory of a SIP server is to initiate a large number of SIP sessions with different session identities.

Broken sessions: With brute force attacks, memory is only consumed for the duration of a transaction and is released afterwards. To intensify the effects of memory usage, the attackers might infer only parts of a session.

Counter Measure:

Monitoring and filtering: Similar to Web and mail servers, SIP proxies need to maintain lists of suspicious users and deny those users from establishing sessions. These lists can be established by monitoring the transactions served by the proxy and logging user behavior.

Authentication: In general, verifying the identity of a user before forwarding his/her messages would prevent malicious behavior as the user would be easily traceable. Naturally, this is only true if it is not possible for an attacker to presume the identity of a valid user.

Like HTTP, SIP uses digest authentication, which requires state maintenance at the server by storing the issued challenge. This can be misused for a broken session attack, if attackers ignore or falsely respond to authentication requests and start another session instead. A solution to this problem is usage of **predictive nonces** that allow for stateless authentication and introduce limited message integrity. The construct is based on nonces being calculated in a way that makes them valid only for validated messages within a time window. When a challenge-response pair arrives at a server, the nonce is first verified to be correct, followed by the verification of the response. This method works without any changes to the protocol.

B) CPU Attacks

Besides the processing power needed for parsing incoming SIP messages, CPU resources are required for the following tasks.

Security Checks — For verifying the identity of a user, a SIP server needs to generate a nonce and then check the credentials of the user. This checking uses hashing schemes such as MD5, which require a low calculation overhead.

Interaction with External Servers — As already indicated a SIP proxy might need to contact an external server to fetch some information or realize a service. This not only consumes processing time but also can cause the server to block and reject new incoming messages while the SIP proxy is awaiting an answer from the contacted server.

Application Execution — A SIP server might need to execute a certain application (i.e., a CPL or CGI script or some other kind of application) after receiving a request. The amount of CPU resources used depends on the application type and its complexity. If the application server is located on the same hardware platform as the SIP server, the CPU resources used for execution of the applications is no longer available for processing SIP messages.

If the application server is located on a different hardware platform, some form of remote communication between the SIP server and the application server is needed. Thus, attacks on the application server result in blocking the SIP server after requesting the execution of an application until the application server generates a reply.

Counter Measures:

Server design: The first line of defense against any DoS attacks is achieved by using well dimensioned hardware with fast CPUs, and large memory and high-speed network connection. Additionally, the software itself needs to be designed with security, speed, and attack possibility in mind. This implies deploying some or all of the following server design options:

(1) Clean and efficient implementation: Implementers need especially to use efficient and fast memory allocation schemes, event handling, and parsing mechanisms.
(2) Parallel processing: In order to avoid blocking incoming messages while the server is busy processing a message or while waiting for an answer from an external server (e.g., AAA) a SIP proxy should be implemented using threads or parallel processes with each process or thread responsible for processing one message at a time.

(C) Message Parsing Attacks

In order to figure out how to handle an incoming message, the server needs to parse at least part of the message and check its consistency. However, due to the free text format of the SIP protocol even a perfectly valid SIP message can be constructed in a way to hamper proper parsing. Here we give a list of possibilities how to complicate message parsing:

- An attacker can create unnecessary long messages in a simple way by adding additional headers (like informative header fields, e.g., Supported) in conjunction with a large message body. Many SIP messages may include bodies, even when they are not needed in every message. Instead of only depleting processor power, longer message also increase network utilization and memory usage.
- Poor parser implementations can be rendered inoperable by including message bodies of a size that does not match that indicated in the Content-Length header.
- Additionally, the SIP standard mandates that headers that have multiple values can be separated into individual header fields so each only contains one value. If multiple message headers of the same field are included in a message where these headers are spread all over the message, this will further complicate parsing.

Counter Measures:

One way to accomplish this is by inserting multiple informative header fields, e.g. Allow or Supported, before the routing fields. SIP as defined in RFC 3261 is a refined version of the previous standard as defined in RFC 2543. Some of the newer design decisions are made to simplify certain operations. However, any RFC 3261 compliant SIP element must be able to handle RFC 2543 messages, which can complicate processing. As such, this can be used by an attacker. Among these modifications are:

VIA headers. Via header fields contain a branch parameter. If this branch parameter does not start with the magic cookie "z9hGbK," the message is considered to be pre-RFC 3261. This would indicate a fallback to the more complex RFC 2581 message handling routine.

Missing tag field: Messages with a To and From header field, but without a tag field, need to be checked by the UAS against all ongoing transactions, thus requiring more processing overhead. Parsing attacks can be countered by an efficient implementation (e.g., by parsing only those parts needed for its correct functioning).

V CONCLUSION

VoIP has become a popular solution for voice communication in enterprises of different sizes. VoIP deployment still faces great challenges regarding malicious attacks, Dos attacks and requires numerous countermeasures to migrate these attacks in existing implementation and future development. The specific problem like (Denial of Service (DoS) and Service Abuse) are major vulnerabilities considered during implementation of VoIP systems in enterprise. Based on the challenges and resources on VoIP in this paper, we specified different kinds of attack scenarios and their counter measures. Technology to handle attacks aiming at specific VoIP protocols such as SIP protocol is implemented. Session initiation protocol is used for investigating possibilities of launching denial of service attacks on SIP servers and proposes preventing ways which reduce the effects of such attacks.

VI REFERENCES

[1] P. Mehta and S. Udani, "Overview of voice over IP", Dept. Comput. Inf. Sci., Univ. Pennsylvania, Philadelphia, PA, Rep. MS-CIS-01-31,Feb. 2001.

[2] J. Mirkovic *et al.*, *Internet Denial of Service: Attack and Defense Mechanisms*, Prentice Hall, 2005.

[3] J. Rosenberg *et al.*, "Session Initiation Protocol," RFC 3261, 2002.

[4] M. Handley *et al.*, "SIP: Session Initiation Protocol," RFC 2543 (obsoleted), Mar. 1999.

[5] J. Rosenberg and H. Schulzrinne, "Session Initiation Protocol (SIP): Locating SIP Servers," RFC 2543, June 2002.

[6] J. Rosenberg, "Request Header Integrity in SIP and HTTP Digest Using Predictive Nonces," expired Internet draft, work in progress, IETF, June 2001.draft-rosenberg-sip-http-pnonce-00.txt

[7] J. Franks *et al.*, "HTTP Authentication: Basic and Digest Access Authentication," Internet Engineering Task Force, RFC 2617, June 1999.

[8] J. Kuthan, "Comparison of Service ------+Creation Approaches for SIP," Int'l. SIP Conf. 2000, Mar. 2000.

[9] S. Axelsson, *Intrusion Detection Systems: A survey and Taxonomy*, Dept. Comput. Eng., Chalmers Univ., Goteborg, Sweden, Rep. 99-15, 2000.

[10] H. Debar, M. Dacier, and A. Wespi, "Towards a taxonomy of intrusiondetection systems," *Comput. Netw.*, vol. 31, pp. 805–822, 1999.

[11] J. Bilien, E. Eliasson, J. Orrblad, and J.O. Vatn, "Secure VoIP: Call establishment and media protection," presented at the 2ndWorkshop Secur. Voice IP, Washington, DC, Jun. 2005.