Studying the Security in VoIP Networks

A.Alseqyani, I.Mkwawa and L.Sun

Centre for Security, Communications and Network Research,
Plymouth University, Plymouth, UK
e-mail: info@cscan.org

Abstract

Voice over IP (VoIP) technology has radically increased due to its advantages such as flexibility and low cost. On the other hand, the security issue in VoIP network has become an important field in order to name and mitigate the several types of attacks including call hijacking, eavesdropping and denial of service (DoS).

This research paper discusses security in VoIP networks by giving a brief background about the VoIP structure and protocols before setting up a testbed in order to conduct different kinds of attack by running some security tools and analyzes the network behavior under these attacks. The results of experiments will be presented and commented in this paper. Finally, a proposed solution to block the flooding attack will be discussed before conclude the main points which found in these experiment.

Keywords

VoIP, network, security, flooding

1 Introduction

Voice-over-Internet Protocol (VoIP) technology is combined of different protocols and devices using the IP network to transfer the voice and video calls instead of the traditional PSTN (Finneran, 2008). It has several advantages such as flexibility, low cost and simplification although of its many problems regarding the security issue. VoIP components including VoIP clients or terminals, gateway to connect different VoIP networks, gatekeeper which is responsible for address translation and access control and the IP network. One of the main protocols in VoIP is Session Initiation Protocol (SIP) which is a signaling protocol covers all session issues such as initiating, modifying and terminating (Handley et al, 1999). This paper will cover the different attacks that could affect VoIP network in addition to the multiple solutions for these attacks and finally, some experiments will be conducted to evaluate the VoIP threats.

2 VoIP Testbed

A VoIP tested as shown in figure 1 is set up to investigate the VoIP security issue. It consists of a VoIP server, two SIP clients and an attacker which explained below.

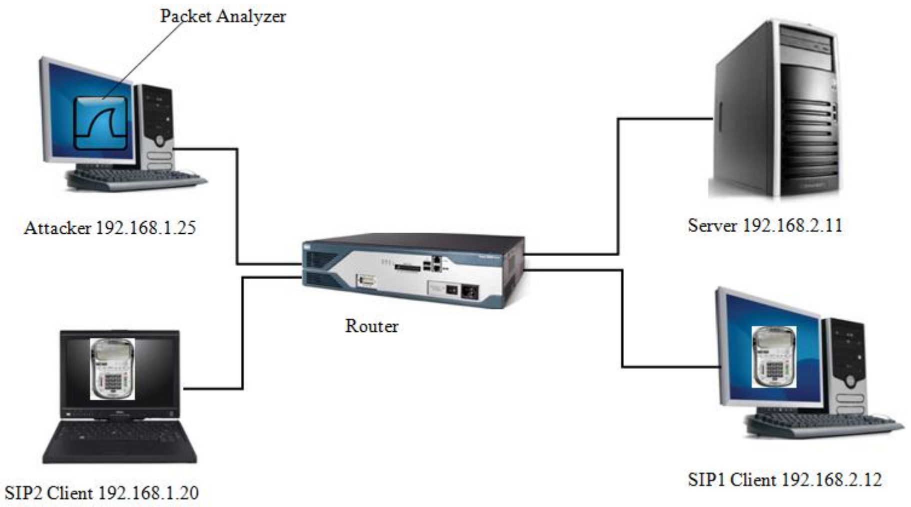


Figure 1: VoIP TestBed

- Asterisk Server: this server connects the different endpoints in the network through performing multiple tasks such as registering the VoIP clients before start calls and forwarding the call from one part to another. In this network, the server has an IP address: 192.186.2.11 and it works on Linux environment. The server hosting the asterisk has properties which are Intel Pentium 4, a CPU of 3 GHZ.
- SIP1: it is one of the two clients in the network. It is a PC with the IP address 192.168.2.12 and has SIP 1 phone works through X-lite program. The x-lite program is free software which allows VoIP clients making the voice and video calls over IP network by using Session Initiation Protocol (SIP) (Xlite, 2010). SIP 1 has the following details: a username as 1000 and password 1234 and it operates on Windows XP with a processor Intel Pentium 4, a CPU of 2.4 GHz and 1 GB memory RAM.
- SIP2: Another client in the network which is a laptop with the IP address 192.168.1.20 has SIP 2 phone works via x-lite program to initiate and receive calls. SIP 2 has specific properties such as a username as 2000 and a password 1234 and it operates on Windows Vista with a processor Intel Core 2 Duo, a CPU of 2.17 GHz and 3 GB memory RAM.
- The attacker: The attacker device is a personal computer and it has an IP address 192.168.1.25 and it will be used to attack the network by installing the tools on it and run these tools. To capture and analyze the traffic in the network, a famous program called Wireshark will be used. This device operates on Linux with a processor Intel Pentium 4, a CPU of 2.2 GHz and 3 GB memory RAM
- The gateway is used to connect the VoIP network clients together or with other clients in different networks. Its IP address is 192.168.2.9.

3 VoIP Security Tools

3.1 Nmap

The Network Mapper (Nmap) is designed to do scanning and security auditing in the small and wide networks (Nmap, 2010). It can come up with different types of information about the network such as available hosts, opened ports, type of operating system, IP addresses and MAC addresses. This tool supports different kinds of scan including TCP SYN Scan, UDP Scans and TCP ACK Scan.

3.2 Cain and Abel

Cain and Abel program is used for password recovery and it has many features including network sniffing for password recovery, recording VoIP conversations, analyzing routing protocols, Brute-Force attacks and using dictionary to crack encrypted passwords (Montoro, 2010). It used in hacking purposes because it has the ability to extract the conversations files and save them in wav files and support **different kinds of codecs such as MS-GSM, G711 uLaw, GSM.**

SIPp

This tool is used to examine SIP proxies, SIP servers and SIP phones. It has many features such as dealing with media traffic, describing calls flow, establishing calls and releasing calls with INVITE messages (SIPp, 2010).

3.3 Wireshark

Wireshark is a tool used by the network managers to capture the traffic and analyze it in order to discover and solve network problems. Between all the packet analyzer programs, Wireshark is the most common open source program used in the network administration (Lamping et al, 2010). Among its many uses, network troubleshooting, security problem testing and debugging protocol implementations.

3.4 Fail2ban

This tool will be installed in the server device to help preventing invite flooding. It has many features such as high configuration, supporting FAM/Gamin, Client/Server architecture and multi threaded (fail2ban, 2010). Fail2ban scans server log files and detects the possible attacking actions after X times of attempts (X can be changed each time) from the same IP address. After detecting the attack, it will ban this IP address for a specific period of time determined in the tool file by the network administrator.

4 Experiments and Results Analysis

4.1 Scanning

This experiment is a very essential step before attacking the network. It helps the attackers to take a general overview about the infrastructure, weak devices and IP addresses as it will be seen below. One of scanning methods is using the efficient tool Nmap. This command below will be used

```
nmap -O -P0 192.168.1.1-254
```

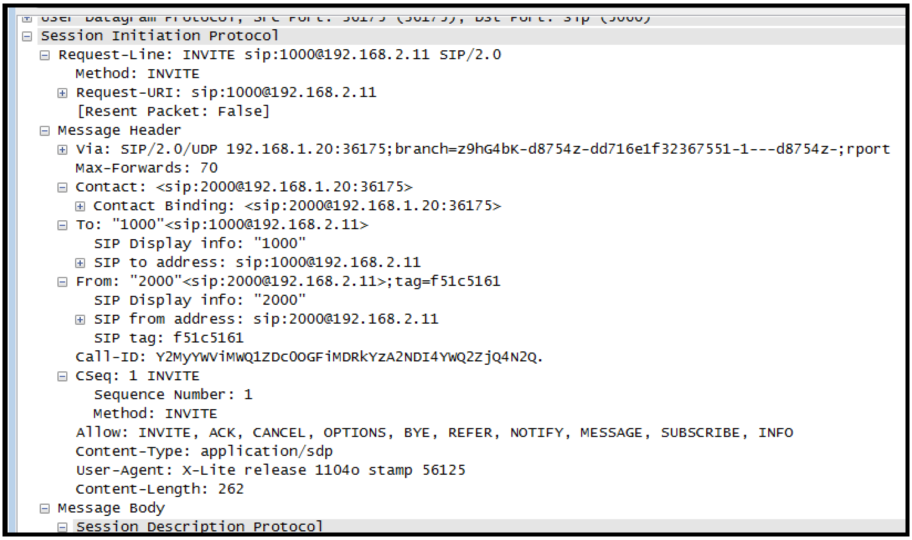
This command will scans all the IP addresses in the network from 1 to 254 and the result will list all network devices with their characteristics. The scan result of 192.168.1.9 is shown in figure 2 and it contains many pieces of information such as the MAC address, open ports (22, 111, and 6000) and the operating system which used.

```
Nmap scan report for 192.168.1.9
Host is up (0.00024s latency).
Not shown: 997 closed ports

PORT     STATE SERVICE
22/tcp   open  ssh
111/tcp  open  rpcbind
6000/tcp open  X11

MAC Address: 00:10:4B:B6:2E:AD (3com)
Device type: general purpose
Running: Linux 2.4.X
OS details: Linux 2.4.9 - 2.4.18 (likely embedded)
Network Distance: 1 hop
```

Figure 2: General Characteristics of host 192.168.1.9

4.2 Eavesdropping

Eavesdropping is one of the famous methods that used by attackers to target the network and collect useful information about the VoIP system. A Wireshark tool will be used here to capture traffic and analyze it during connection between two clients. Figure 3, which captured during a call between two endpoints with the IP addresses 192.168.1.20 and 192.168.2.12, shows different types of important data that could be easily obtained. There is much information appears in the figure below including the IP addresses and usernames for the two endpoints in addition to the port number. The type of media that used during the session is also one of the essential information that can be used from the attacker. All of this information is very necessary to be known before different methods of attacks could happen to any target



```
⊞ User Datagram Protocol, Src Port: 36175 (36175), Dst Port: sip (5060)
⊟ Session Initiation Protocol
  ⊟ Request-Line: INVITE sip:1000@192.168.2.11 SIP/2.0
      Method: INVITE
    ⊞ Request-URI: sip:1000@192.168.2.11
      [Resent Packet: False]
  ⊟ Message Header
    ⊞ Via: SIP/2.0/UDP 192.168.1.20:36175;branch=z9hG4bK-d8754z-dd716e1f32367551-1---d8754z-;rport
      Max-Forwards: 70
    ⊟ Contact: <sip:2000@192.168.1.20:36175>
      ⊞ Contact Binding: <sip:2000@192.168.1.20:36175>
    ⊟ To: "1000"<sip:1000@192.168.2.11>
        SIP Display info: "1000"
      ⊞ SIP to address: sip:1000@192.168.2.11
    ⊟ From: "2000"<sip:2000@192.168.2.11>;tag=f51c5161
        SIP Display info: "2000"
      ⊞ SIP from address: sip:2000@192.168.2.11
        SIP tag: f51c5161
      Call-ID: Y2MyYWViMWQ1ZDc0OGFiMDRkYzA2NDI4YWQ2ZjQ4N2Q.
    ⊟ CSeq: 1 INVITE
        Sequence Number: 1
        Method: INVITE
      Allow: INVITE, ACK, CANCEL, OPTIONS, BYE, REFER, NOTIFY, MESSAGE, SUBSCRIBE, INFO
      Content-Type: application/sdp
      User-Agent: X-Lite release 1104o stamp 56125
      Content-Length: 262
  ⊟ Message Body
    ⊟ Session Description Protocol
```

Figure 3: Important Information can be obtained from SIP Message using Wireshark

4.3 INVITE Flooding

This experiment covered affect of INVITE flooding Attack on the Asterisk server. SIPp tool has been used to generate INVITE messages. The numbers of INVITE messages started from 0 per second (no flooding) and increased by 5000 each time until 50000 INVITE messages per second. The number of passed calls, failed calls and the delay are calculated each time. This command which used to send INVITE flooding has been run from the attacker device to the server:

Sipp –sn uac –r N –rp M IP address

It will run SIPp with embedded client (uac) scenario at the rate N calls per M seconds.

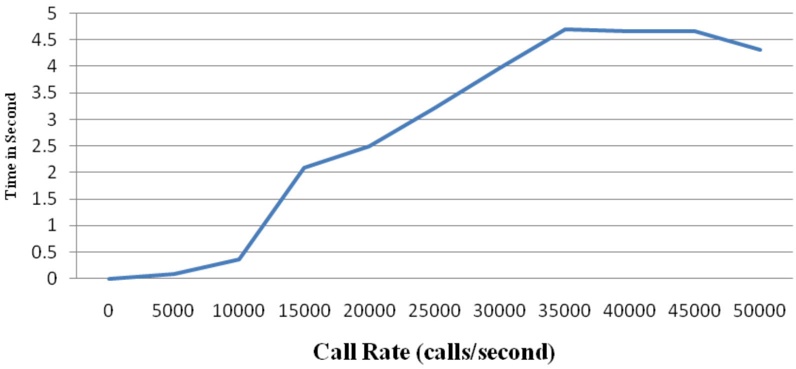


Figure 4 Call Delay during INVITE Flood

The call delay will be calculated in this experiment by using *Wireshark* to capture the traffic during make calls. This delay is referred to the difference in time between initiating the call and receiving this call from the other client.

Figure4 shows that the delay time increases as the number of call rate increases until it is almost being stable from 35,000 to 50,000 calls per second. The normal delay when no flooding attack is generated (at 0 calls per second) was about 2.96 millisecond (0.00296 second). These results appeared because of the effect of the attack on the SIP proxy server which causes it unable to process the call as the number of calls increase. Note that the Asterisk server works with Intel Pentium 4 and its CPU is 3 GHZ and therefore, its performance is expected to be better if powerful server with more features had been used.

Regarding the number of passed calls, 100 calls will be done at each call rate to give more accurate results. As it is shown from figure 5, the Number of calls forwarded by the Server decreases from 100 calls at no flooding (0 calls/sec) until it reaches about 20 passed calls at 50000 calls/sec. The Asterisk server has fixed features which can afford handling certain numbers of clients per specific period of time. After that, the server has to drop some messages to be able to process the other which make the caller sometimes starts to get a timeout message from the phone saying that the number is unavailable now

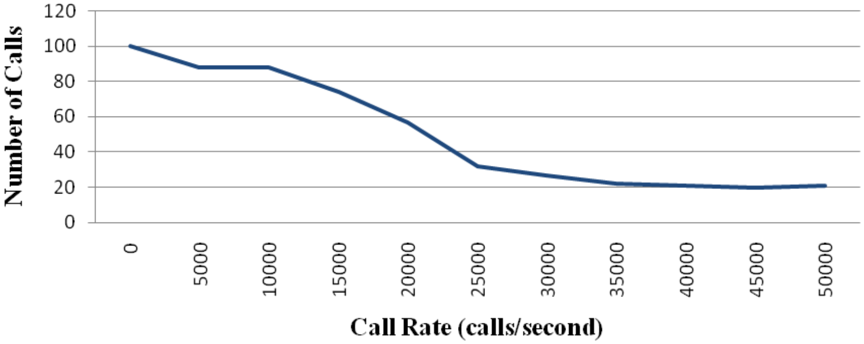


Figure 5: Passed Calls

4.4 INVITE Flooding Mitigation

In this experiment, a tool called Fail2ban has been used to run in the Asterisk server during the flooding attack. This tool requires determining the source and destination IP to block the flooding attack and ban the attacker for specific period of time. SIPp is used to launch the flooding attack from the attacker device through this command

```
Sipp –sn uac –r N –rp M –i IP address1 IP address2
```

Where IP address1 is the attacker IP address (192.168.1.25) and IP address2 is the server IP address (192.168.2.11). Fail2ban parameters have been set to the following: Maximum number of attempts: 5 and Ban period of time: 300 second. By setting these parameters, any IP address trying to attack asterisk server for 5 times is banned for 300 seconds. To do this, Fail2ban checks the log files and determine any suspicious traffic.

Figure 6 shows the difference in delay where it is clearly appeared that the mitigation reduced the delay because the VoIP clients had a period of time to make calls without any flooding attack can affect the service. For example, the delay at 40000 calls per second before using fail2ban was 4.65 second while it dramatically decreased to 0.69 second after applying the solution. The difference in delay can be noticed each time the call rate is increased.

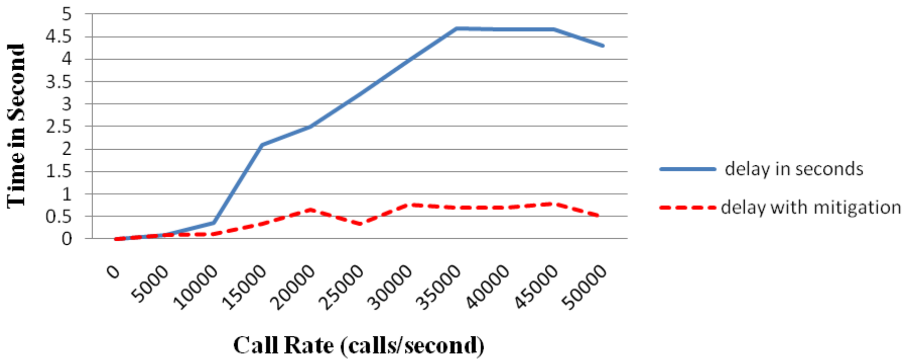


Figure 6: Difference in Delay

Regarding the number of passed calls that succeeded in arriving at the destination, figure 7 shows that about 98% of the calls have been processed by the server which gives an indication about the positive effect of the proposed solution. After applying this solution, the server does not suffer from the flooding attack most of the time because the tool bans the attacker for a specific period of time after detecting the attack which gives VoIP clients the opportunity to successfully make calls. The number of failed calls decreased significantly to the lowest level. For example, at the call rate of 35000 calls per second, no failed calls recorded and all the generated calls were succeeded. The number of passed calls is very high because the possibility of initiating the calls outside the ban period is very low. This happens due that the time which the server takes before banning the attacker is almost lower than 31 second which represents the time out period which ensures the server to forward calls in this period. This number may change at different situation such as increasing the number of clients in the network or decreasing the time period of attacker ban.

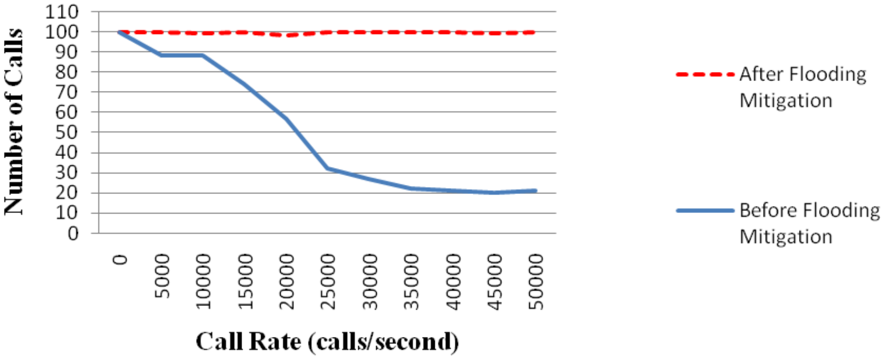


Figure 7: Passed Calls after and before mitigation

5 Conclusion

Security in Voice over IP (VoIP) networks is an important issue that should be taken in account from both users and providers. In order to discuss VoIP security, this paper introduced different types of attacks such as eavesdropping and INVITES flooding.

These experiments also have revealed clearly that VoIP network is vulnerable to different attacks especially DoS flooding attack which has high impact on the VoIP clients and Asterisk server. As we have seen from flooding test, the attack produces more delay when one client contacts the other in a VoIP network because the server could not afford the huge number of INVITE requests and therefore, the number of failed calls increased each time the call rate is increased.

To achieve high level of protection, different countermeasures can be used to detect or reduce the VoIP threats. These solutions include firewall, Network Address Translation (NAT), intrusion detection system (IDS) and encryption technique. However experimental outcomes have shown that the proposed solution using fail2ban tool that used to detect flooding attack can reduce the effect of this attack. These results depend on the ban period time and number of clients in the network.

Further studies are needed to address this topic in the future because none of the proposed solution can present a complete countermeasure for all problems and with time, new threats will be appeared and need to be solved efficiently. But generally the VoIP system will depend in the future on the security part. If the threats in VoIP technology could be solved, the deployment of VoIP will be easier and could replace the traditional phone system.

6 Reference

Fail2ban, 2010, [online] available at: http://www.fail2ban.org/wiki/index.php/Main_Page accessed 25 August 2010

Finneran, M 2008,” Voice over WLANs”, Page 161-162 Elsevier Inc, America

Handley M, Schulzrinne H, Schooler E, Rosenberg J March 1999, “SIP: Session Initiation Protocol” [online] available at: http://www.ietf.org/rfc/rfc2543.txt accessed 25 August 2010

X-lite, 2010, Counter Path Corporation, [online] available at http://www.counterpath.com/x-lite.html accessed 25 August 2010

Nmap, 2010, Network Mapper, [online] available at : http://nmap.org/ accessed 23 August 2010

Montoro, M 2010, OXID.IT, [online] available at: http://www.oxid.it/cain.html accessed 23 August 2010

SIPP, 2010, [online] available at: http://sipp.sourceforge.net/ accessed 23 August 2010