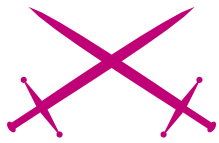


Strumenti per il test di un'infrastruttura VoIP



Attacco

Luca Leone, Nicola Mondinelli, Pierpaolo Palazzoli, Matteo Valenza



Grado di difficoltà



La tecnologia che rende possibile una conversazione telefonica attraverso il traffico IP più comunemente denominata VOIP (Voice Over IP) è sempre più utilizzata sia da aziende che da privati.

Se per le prime l'utilizzo può risultare comodo soprattutto per le connessioni telefoniche tra le varie filiali e il telelavoro dei propri dipendenti, per i secondi serve principalmente per slegarsi dalla telefonia tradizionale e trovare delle tariffe più convenienti.

Infatti molti ISP stanno prontamente introducendo offerte molto vantaggiose dal punto di vista dei costi di chiamata verso le linee telefoniche in tutto il mondo.

Questo nuovo approccio alla comunicazione telefonica ha creato sicuramente un nuovo mercato IT e dei servizi legati ad esso, ma ha anche introdotto dei problemi relativi alla sicurezza non presenti nella telefonia tradizionale.

Quello che era un semplice e *stupido* telefono analogico auto-alimentato dalla centrale telefonica viene sostituito da un terminale *intelligente* con un sistema operativo con molte altre funzionalità più evolute.

La stessa immagine può essere utilizzata per tutta l'infrastruttura telefonica a partire dai cavi concludendo con il PBX.

La comunicazione telefonica avviene tramite protocolli di segnalazione della conversazione (SIP, H323, IAX, ...) e del trasporto dei

dati (RTP, IAX ...), spesso vengono utilizzati attraverso comunicazioni in *chiaro* e sistemi di autenticazione deboli.

Come si può intuire vengono introdotti nuovi e numerosi fattori rispetto alla tecnologia precedente ed in questo articolo verranno analizzati diversi approcci all'analisi di sicurezza di un sistema VOIP.

Più nel dettaglio saranno trattati: scansione dell'infrastruttura, controllo delle interfacce di gestione, sniffing delle comunicazioni e dell'autenticazione e denial of service.

Dall'articolo imparerai...

- Vulnerabilità principali di una infrastruttura VOIP,
- Tool di analisi e auditing sui protocolli SIP e IAX,
- Metodologia di analisi del rischio.

Cosa dovresti sapere...

- Basi di Networking,
- Basi della struttura del TCP/IP,
- Basi di Network auditing.

Si potranno così individuare alcuni dei limiti del proprio sistema ed effettuare una corretta analisi del rischio.

Tool e strumenti

Gli strumenti utilizzabili nell'analisi di un'infrastruttura VoIP sono numerosi. Una buona selezione di quelli free e open source trovate facendo riferimento alla sezione in rete.

Verranno trattati alcuni di questi programmi principalmente per verificare e testare:

- Servizi attivi,
- Interfacce di gestione terminali e PBX,
- Autenticazione,
- Intercettazione telefonate,
- DoS.

Scansione servizi attivi

Grazie a nmap si possono scansionare host remoti ed individuare così apparati VoIP, tramite l'opzione -sU si possono verificare vari servizi attivi in ascolto su porte UDP alle quali si appoggiano solitamente protocolli di servizio VoIP come SIP e IAX v.2.

SMAP

Addentrando nello specifico del protocollo SIP uno strumento utile per lo scansionamento risulta SMAP ottenuto dalla fusione dall'approccio utilizzato da nmap e sipsak.

SMAP inviando varie richieste SIP ai vari apparati presenti in rete riesce, grazie ad un database di fingerprint noti, a identificare modello e OS utilizzati.

Il programma può essere scaricato in formato tar.gz e una volta scompattato l'archivio, si dovrà compilare il sorgente semplicemente tramite il *Makefile* fornito.

L'utilizzo è molto semplice, vedi Listato 1.

Come indicato dall'ideatore di questo progetto la precisione all'interno di una rete LAN può risultare buona ma se la scansione viene effettuata dietro NAT e firewall le informazioni ottenute non sono del tutto attendibili.

Interfacce di gestione
Potremmo immaginarci internet

senza motori di ricerca? Senza strumenti in grado di mettere ordine nell'enorme calderone di informazio-

ni che rimbalzano in giro per il mondo? La nascita di Google, Altavista, Yahoo (e simili) è un passo necessa-

Listato 1. Utilizzo smap in rete

```
smap [ options ] <ip | ip/mask | host>
$ ./smap 192.168.100.0/24
smap 0.4.0-cvs <hscholz@raisdorf.net> http://www.wormulon.net/
Host 192.168.100.1:5060: (ICMP OK) SIP enabled
Host 192.168.100.2:5060: (ICMP OK) SIP timeout
Host 192.168.100.3:5060: (ICMP timeout) SIP enabled
...
Host 192.168.100.254:5060: (ICMP OK) SIP enabled
      Asterisk PBX (unknown version)
256 hosts scanned, 10 ICMP reachable, 3 SIP enabled
Grazie all'opzione -o si possono utilizzare i fingerprint contenuti nel file
      fingerprint.db
(smap riconoscimento fingerprint)
$ ./smap -o 192.168.100.1
smap 0.4.0-cvs <hscholz@raisdorf.net> http://www.wormulon.net/
Host 192.168.100.1:5060: (ICMP OK) SIP enabled
AVM FRITZ!Box Fon Series firmware: 14.03.(89|90)
1 hosts scanned, 1 ICMP reachable, 51SIP enabled
```

Listato 2: Dump del traffico udp

```
dimebag SIPcrack-0.1 # tcpdump -s 0 -w net-capture.txt udp -i eth0
tcpdump: listening on eth0, link-type EN10MB (Ethernet), capture size 65535
      bytes
237 packets captured
474 packets received by filter
0 packets dropped by kernel
```

Listato 3. Filtraggio del traffico perso con tcpdump

```
dimebag SIPcrack-0.1 # ./sipdump -d sip-logins.dump -f net-capture.txt
SIPdump 0.1 ( MaJoMu | www.remote-exploit.org )
-----
* Using tcpdump data file 'net-capture.txt' for sniffing
* Starting to sniff with filter 'tcp or udp'
* Adding 192.168.123.92:50195 <-> 192.168.123.99:50451 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: '201') to dump
      file
* Exiting, sniffed 1 logins
```

Listato 4. Sipcrack in azione

```
dimebag SIPcrack-0.1 # ./sipcrack -w fifosipcrack -d sip-logins.dump
SIPcrack 0.1 ( MaJoMu | www.remote-exploit.org )
-----
* Reading and parsing dump file...
* Found Accounts:
Num Server      Client          User Algorithm Hash / Password
1 192.168.123.99 192.168.123.92 201 MD5      dfc9979f98f0c546
                                         c08dc3073dda1cc1
* Select which entry to crack (1 - 1): 1
* Generating static MD5 hash...e71899168871bb8929ff6c25aab955b2
* Starting bruteforce against user '201' (MD5 Hash: 'dfc9979f98f0c546c08dc30
      73dda1cc1')
* Loaded wordlist: 'fifosipcrack'
* Tried 25 passwords in 0 seconds
* Found password: '1234'
* Updating 'sip-logins.dump'...done
```



rio per lo sviluppo di una realtà così complessa e ricca come è quella del mondo digitale.

Anche se non è il tema di quest'articolo è giusto ricordare che questi motori di ricerca per avere un database il più possibile completo sono al lavoro notte e giorno con strumenti di *data harvesting* conosciuti sotto il nome di *spider* (anche *crawler*, *bot*) che incessantemente

sondano la rete alla ricerca di input. Questi script non fanno altro che recuperare delle URI dalla rete, prevalentemente pagine web, analizzarne il contenuto e catalogarlo nel database del motore di ricerca. La URI può essere fornita direttamente dagli sviluppatori oppure recuperata in maniera ricorsiva partendo dagli hyperlink presenti in altre pagine web precedentemente analizzate.

In questo modo è possibile catalogare milioni di siti in poco tempo.

Per qualcuno questo fatto potrebbe risultare inquietante, per alcuni sbalorditivo per altri ancora... sfruttabile.

Mi spiego meglio: *Google passa tutto il tempo a recuperare informazioni da internet, perchè fare la fatica di cercarle anche noi, quando posso sfruttare chi l'ha già fatto per me?*

Cosa centra questo con la sicurezza nel VoIP? I vostri telefoni VoIP hanno un'interfaccia di gestione web? I vostri server VoIP hanno un'interfaccia di gestione web? I vostri telefoni o server VoIP hanno la loro interfaccia web raggiungibile da remoto, da internet?

Ok, molti di voi penseranno *chi è quello stupido che lascerebbe un'interfaccia di gestione raggiungibile da tutta internet?* Ok, pensatelo, ma mentre lo pensate fate un giro a controllare i vostri apparati, perchè non si sa mai.

La *footprinting* è una tecnica ormai molto utilizzata per individuare e raccogliere informazioni preliminari su sistemi con falle di sicurezza note o con malconfigurazioni (qualcosa come user *admin* e password *admin*), si tratta di eseguire delle ricerche mirate all'interno dei database di Google alla ricerca di stringhe di caratteri identificative di determinate interfacce di gestione che gli spider hanno trovato in rete.

Questa tecnica si è affinata ed evoluta con l'avvento di migliaia di apparati che forniscono il management via web, ed ormai con semplici ricerche è possibile individuare determinati dispositivi collegati in tutto il globo.

Esempio pratico:

```
[inurl:"NetworkConfiguration" cisco]
```

Se inserite la precedente riga (escluse le parentesi quadre) nella casella di ricerca di Google vi ritroverete a sondare il database di Google alla ricerca di (alcuni) telefoni VoIP Cisco, o meglio, della loro interfaccia di gestione.

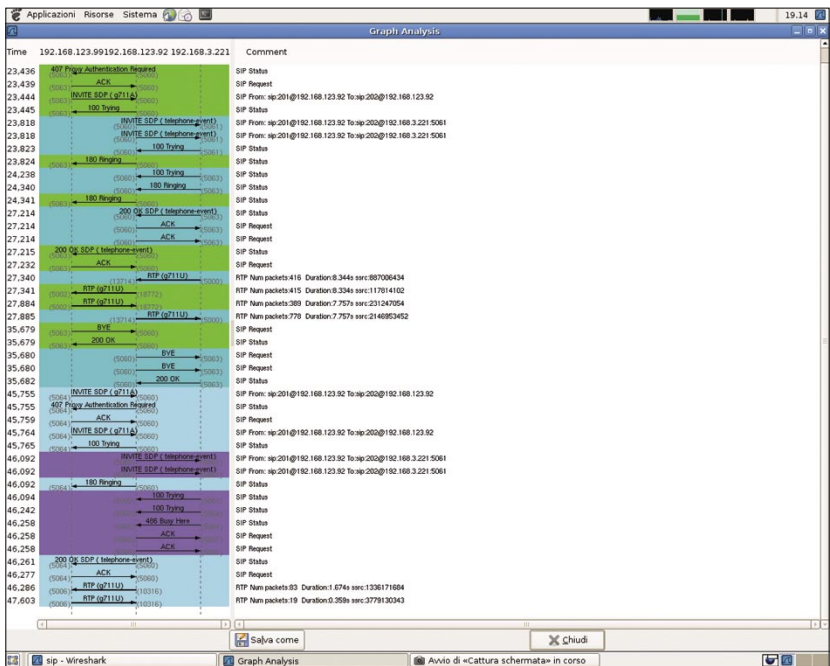


Fig. 1. Grafico chiamata Wireshark

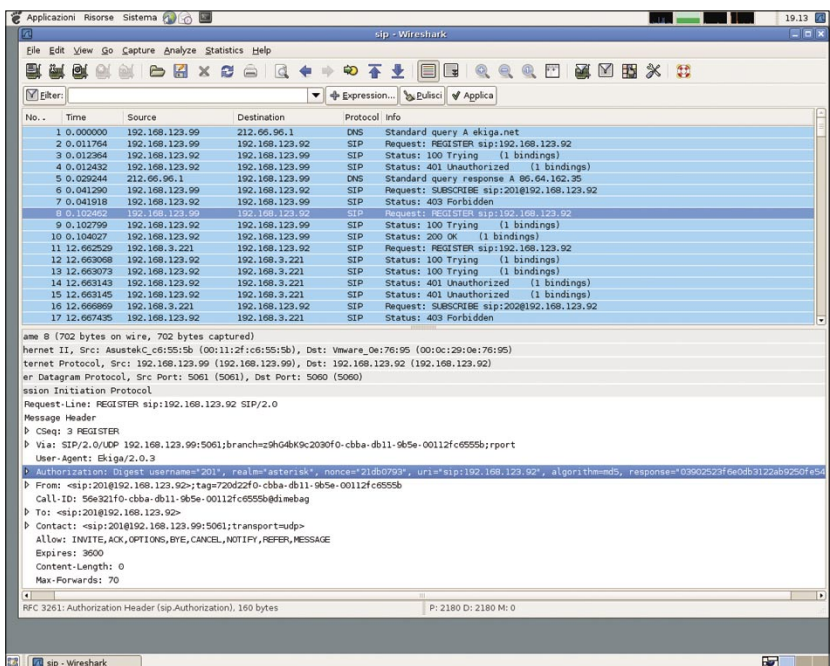


Fig. 2. Autenticazione SIP

Sorprendente, non trovate? Decine di dispositivi comodamente indicizzati. In verità è doveroso dire che questo tipo di ricerca è nota da tempo e quindi solamente poche decine di dispositivi risultano ancora raggiungibili, mentre nei mesi passati il numero era nettamente maggiore.

L'interfaccia di management Cisco però è molto povera di funzionalità.

Pensiamo cosa si potrebbe fare se l'interfaccia permettesse di fare chiamate Voip: certo noi non parteciperemmo alla conversazione ma dall'altro lato i telefoni inizierebbero a squillare per motivi ignoti. Divergente?!

E se l'interfaccia prevedesse un sistema di packet capturing (PCAP)? Potremmo essere in grado addirittura di intercettare il traffico di fonìa e scaricarlo in locale per poi analizzarlo con calma; ma interfacce così evolute non se ne trovano, chi sarebbe così sprovveduto? Assolutamente non è possibile!

```
Proxvate ["(e.g. 0114930398330)" snom].
```

Lascio alla vostra fantasia ogni ulteriore footprint che vi piacerà eseguire sulle interfacce di gestione remota: l'importante è individuare in queste pagine delle stringhe univoche che sia possibile ricercare in Google e il gioco è fatto.

Per chi progetta interfacce web è utile conoscere uno standard di configurazione per dialogare con gli *spider: robots.txt*. Questo è un file, che deve essere presente nella root directory del nostro server web nel quale si indica al *crawler* quali pagine indicizzare e quali no. Nelle interfacce di gestione web, che non è opportuno siano indicizzate è utile inserire questo file con il seguente contenuto (2 righe):

```
User-Agent: *
Disallow: /
```

Il significato spero sia immediato senza bisogno di ulteriori spiegazioni.

Autenticazione

L'autenticazione per molti apparati e client SIP è basata sul meccanismo utilizzato dal protocollo HTTP trami-

te lo schema Digest con MD5 (rfc 2617). Come si può intuire questo tipo di autenticazione è soggetta a potenziali vulnerabilità ed un esempio

Listato 5. File di configurazione voipong.conf

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

[FILTERS]
startup = "udp"
```

Listato 6. Voipong in background

```
dimebag voipong-2.0 # ./voipong
EnderUNIX VOIPONG Voice Over IP Sniffer starting...
Release 2.0, running on dimebag [Linux 2.6.18 i686]
(c) Murat Balaban http://www.enderunix.org/
dimebag voipong-2.0 #
dimebag voipong-2.0 # ./voipctl
Connected to VoIPong Management Console
System:
dimebag [Linux 2.6.18 i686]
voipong> shcall
ID      NODE1          PORT1 NODE2          PORT2 STIME          DURATION
-----
09534 192.168.123.99 05022 192.168.123.92 16260 13/02/07 17:26:32 9 seconds

Total listed: 1
```

Listato 7. Console di gestione voipctl)

```
voipong> help
Commands:
help           : this one
quit          : quit management console
uptime        : Server uptime
logrotate     : rotate server's logs
setdebug [level] : set debug level to [level]
setmixflag [flag] : set mix voice flag to true or false [e.g: 1 for true,
0 for false]
shutdown     : shutdown server
rusage       : CPU usage statistics for the server
loadnets    : Reload voipongnets file
info         : General server information
shcall       : Show currently monitored calls
shrtcp       : Show currently RTCP cache
killcall [id] : end monitoring session with [id]
```



concreto può essere dato dall'utilizzo di semplici strumenti per password cracking.

Per analizzare il traffico di rete utilizzeremo Wireshark sulle connessioni UDP, grazie a questo potentissimo tool possiamo visualizzare addirittura il grafico dello scambio di pacchetti avvenuti durante una telefonata semplicemente selezionando *Statistics --> VoIP Calls --> Graph* (Fig. 1).

Per poter individuare l'autenticazione in rete basterà quindi utilizzare l'apposito filtro per il protocollo SIP ed individuare la richiesta di registrazione (Fig. 2).

Per semplificare maggiormente quest'ultima operazione di filtraggio si può utilizzare SIPcrack, un piccolo programma scritto in C per analizzare unicamente l'autenticazione SIP.

SIPcrack è un SIP protocol login cracker composto da due programmi, sipdump serve a individuare i tentativi di autenticazione in rete da un dump effettuato con tcpdump e sipcrack per recuperare le password tramite un attacco a forza bruta.

Un esempio di utilizzo può essere il seguente: eseguire la cattura di tutti i pacchetti udp sull'interfaccia *eth0* salvandoli nel file *net-capture.txt* (Listato 2).

Tramite *sipdump* filtrare i login avvenuti in rete e salvarli nel file *sip-logins.dump* (Listato 3).

Listato 8. Chiamate intercettate

```
dimebag ~ # cd /var/log/voipong/20070213/
dimebag 20070213 # ls

session-enc0-PCMU-8KHz-192.168.123.92,16260-192.168.123.99,5022.raw
session-enc0-PCMU-8KHz-192.168.123.92,19088-192.168.123.99,5026.raw
session-enc0-PCMU-8KHz-192.168.123.99,5022-192.168.123.92,16260.raw
session-enc0-PCMU-8KHz-192.168.123.99,5022-192.168.123.92,16260.wav
session-enc0-PCMU-8KHz-192.168.123.99,5026-192.168.123.92,19088.raw
session-enc0-PCMU-8KHz-192.168.123.99,5026-192.168.123.92,19088.wav
```

Listato 9. Snort rules

```
# this set are for general SIP specific flooding
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, second
s 60; classtype:attempted-dos; sid:2003193; rev:1;)

#from the rules at nextsoft.cz
#intended to catch unusual numbers of unauthorized responses from sip servers
drop ip $HOME_NET 5060 -> any any (msg:"BLEEDING-EDGE VOIP Multiple
Unauthorized SIP Responses"; content:"SIP/2.0 401
Unauthorized"; depth:24; threshold: type both, tra
ck by_src, count 5, seconds 360; classtype:attempted-dos; sid:2003194; rev:
1;)
```

Listato 10. Snort rules

```
#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;)
```

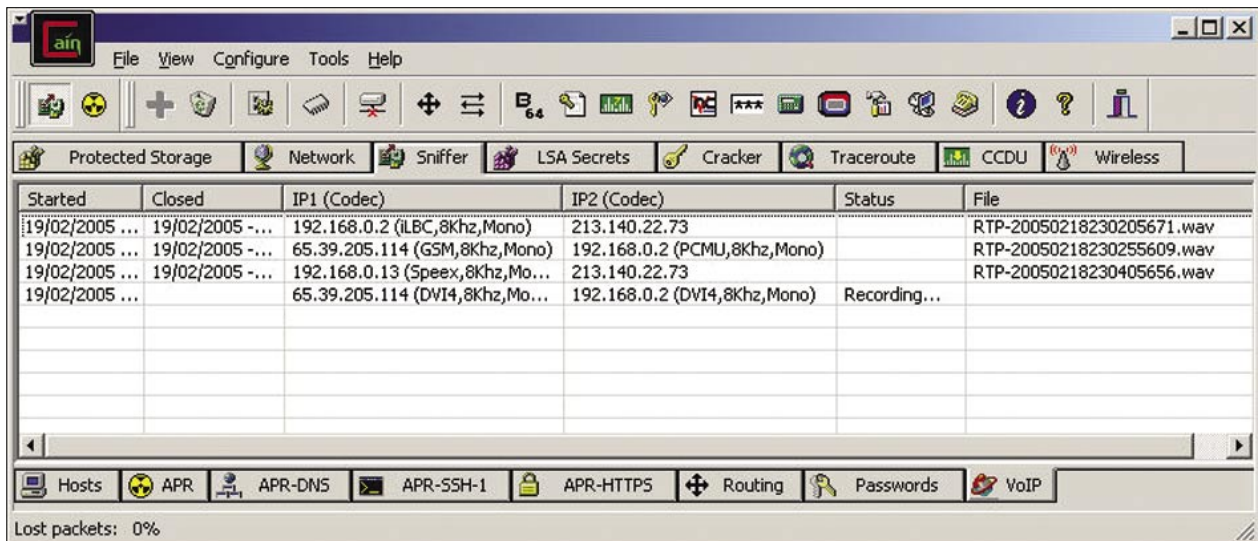


Fig. 3. Cain&Abel Sezione Voip

Creare un file pipe fifo:

```
dimebag SIPcrack-0.1 # mkfifo
fifosipcrack
```

per poter demandare l' utilizzo di wordlist provenienti da un secondo programma come ad esempio John the ripper:

```
(avvio di john the ripper)
dimebag SIPcrack-0.1
# john --incremental=alnum -stdout=8
> fifosipcrack
```

da un altro terminale si dovrà poi lanciare sipcrack passando il dump precedentemente filtrato (Listato 4).

Questo esempio deve far riflettere; è veramente molto semplice perdere le proprie credenziali in una infrastruttura voip. Soluzioni a questo problema possono essere l'appoggio a canali cifrati tramite VPN o SIP over TLS (*Transport Layer Security*).

Un' analisi simile può essere fatta del protocollo IAX v2 con autenticazione tramite l'utilizzo di MD5. Per evitare questo il protocollo permette

anche un'autenticazione a chiave pubblica e privata attraverso RSA.

Intercettazione Telefonate

La parte di comunicazione effettiva avviene tramite il protocollo RTP come si può vedere dal grafico precedente creato tramite *Wireshark*. Per evidenziare i problemi rispetto alla intercettazione sfrutteremo *Voipong*, uno sniffer di rete che intercetta le chiamate Voip effettuate mediante svariati protocolli come SIP, H323 e *Cisco's Skinny Client Protocol*. Individuando la comunicazione in chiaro veicolata dal RTP questo tool è in grado di decodificare la telefonata e di salvarla in file wav. Il progetto che può essere seguito e scaricato anche in versione live CD dal sito dello sviluppatore.

E' prevista la possibilità di estendere in moduli DSOM (*Dynamic Shared Object Modules*) la struttura dei decoder supportati, ma nativamente con la versione 2.0 sono utilizzabili i codec G711 u-law e G711 a-law che risultano essere i più adottati dai ter-

minali in LAN per questioni di qualità audio.

Per un corretto funzionamento lo sniffer necessita delle librerie libpcap e di sox per la creazione di file WAV.

Il pacchetto dopo essere stato compilato ed installato prevede la configurazione attraverso voipong.conf (Listato 5).

e il file dove vengono indicati gli obiettivi da controllare con lo sniffer chiamato voipongnets:

```
(file voipongnets)
192.168.3.0/255.255.255.0 lfp
```

dove lfp (*Least False Positive*) indica un algoritmo per identificare le chiamate voip. Per maggiori dettagli si può consultare la dettagliata documentazione on-line.

Come per un normale sniffer di rete voipong dovrà essere messo in ascolto attraverso una interfaccia di rete in grado di rilevare tutto il traffico voip. Per ottenere questo si possono intraprendere differenti strade:

- installazione sulla macchina gateway della rete voip,
- interfaccia attestata sulla porta monitor dello switch,
- interfaccia in rete condivisa tramite hub,
- arpoisoning,
- switch flooding.

Lanciando poi l'eseguibile voipong si potrà attivare in modalità background lo sniffer e tramite la console voipctl si potranno osservare le chiamate intercettate (Listato 6).

Come si può notare nel Listato 6 grazie al comando shcall è stata visualizzata una comunicazione tra l'host 192.168.123.99 alla porta udp 5022 e l'host 192.168.123.92 alla porta 16260.

Tramite console si possono visualizzare informazioni e configurare alcune opzioni per il server (Listato 7).

Per ascoltare le telefonate intercettate nel caso preso in esame ci si dovrà recare nella directory indicata nel file di configurazione alla voce outdir e recuperare i file .wav (Listato 8).

In Rete

- <http://voipsa.org/Resources/tools.php>,
- http://www.hackingvoip.com/sec_tools.html,
- <http://www.wormulon.net/index.php?archives/1125-smap-released.html> – SMAP,
- http://remote-exploit.org/codes_sipcrack.html – SIPcrack,
- <http://www.enderunix.org/voipong/> – VOIPONG,
- <http://www.hackingexposedvoip.com/tools/iaxflood.tar.gz> – IAXflood,
- <http://www.openwall.com/john/doc/EXAMPLES.shtml> – documentazione John the ripper,
- <http://www.voip-info.org/wiki/view/IAX+encryption>,
- <http://www.voip-info.org/wiki/index.php?page=Asterisk+iax+rsa+auth>.

Cenni sugli autori

Snortattack.org, portale orientato alla sicurezza, è il risultato della fusione di conoscenze e di collaborazione del team. Le tipologie di argomenti trattati spaziano su 360 gradi tutti gli ambiti di sicurezza: attacco/difesa.

Grande punto di forza è l'uso di Snort come soluzione alle innumerevoli problematiche di intrusione. Un forum e mailing list concorrono a tenere aggiornati gli utenti sulle nuove problematiche. Con *Snortattack.org*, il team, intende creare uno *Snort User Group* finalizzato alla collaborazione tra gli utenti di Snort dell'Italia e di tutto il Mondo, e la missione di trattare le problematiche di sicurezza.



Utilizzando invece Cain & Abel su una macchina Windows si può ottenere un simile risultato grazie ad uno sniffer VOIP che permette le intercettazioni.

Esso intercetta comunicazioni codificate con i seguenti codec: G711 uLaw, G771 aLaw, ADPCM, DVI4, LPC, GSM610, Microsoft GSM, L16, G729, Speex, iLBC, G722.1, G723.1, G726-16, G726-24, G726-32, G726-40, LPC-10. Selezionandolo verranno visualizzate le chiamate con il codec utilizzato e automaticamente verranno salvati i file di decodifica .wav nella directory di installazione di Cain & Abel.

Grazie a questi potentissimi strumenti possiamo renderci conto di quanto sia semplice effettuare intercettazioni telefoniche con protocolli che lavorano in chiaro senza adottare opportune tecniche di cifratura. Per ovviare a questo problema si consiglia l'utilizzo di canali VPN o del protocollo SRTP.

Analogamente lo streaming audio trasportato da IAX v2 in chiaro può essere intercettato, ma gli ideatori del protocollo stanno lavorando ad un modalità di canale cifrato tramite AES non ancora dichiarata.

DoS

Un altro difficile ostacolo da superare risulta essere il Denial of service sia per il protocollo SIP che IAX v2.

Strumenti in grado di inviare pacchetti VoIP possono essere scritti ad-hoc molto facilmente ad esempio tramite perl appoggiandosi alle librerie CPAN specifiche per il protocollo, o altrimenti sfruttando programmi come SIPBomber, IAXflood, SIPsak.

Un pacchetto di facile utilizzo e di grande potenza risulta: iaxflood. Questo programma è in grado di dare un Dos sul server voip nel caso in cui si usi protocollo iax.

L'uso è sestremamente semplice (iaxflood):

```
usage: ./iaxflood sourcename
destinationname numpackets
```

Quindi sorgente, destinazione e numero di pacchetti. La sorgente e destinazione devono essere raggiungibili senza nat, quindi raggiungibili direttamente sul proprio ip.

La finalità dell'uso di questo pacchetto è di sicuro l'abbassamento della qualità del servizio fino ad arrivare al blocco dell' erogazione del servizio stesso.

Conclusione e consigli

In base all'infrastruttura si dovranno avere determinate attenzioni rivolte alla sicurezza come:

- mantenere linee di backup PSTN o ISDN per il traffico voce,

- progettare una rete di alimentazione di backup tramite UPS e switch power over ethernet per alimentare i terminali,
- esporre in internet il minor numero possibile di servizi in chiaro con autenticazione debole,
- non esporre in internet telefoni e interfacce di gestione,
- utilizzare password sicure per la gestione dei terminali,
- utilizzare VLAN all'interno della nostra rete per separare il traffico dati da quello voip,
- se è possibile utilizzare apparati che supportino la cifratura dello streaming audio SRTP,
- gestire la qualità del servizio QoS,
- utilizzo di canali cifrati per il traffico voip tramite VPN ipsec o tls,
- consentire l'accesso alle proprie risorse in internet limitandone l'utilizzo (controllo ip sorgente, ...),
- utilizzare firewall in grado di lavorare a livello applicativo SIP/IAX,
- utilizzo di Intrusion Prevention System.

Ad esempio questo ultimo approccio indicato risulta determinante per la tipologia di attacchi DoS su protocolli VOIP,essendo a livello applicativo essi risultano non intercettabili da parte di dispositivi di sicurezza a livello 3 ISO/OSI.

Come tutti ben sanno l'IDS/IPS standard de facto è Snort, IPS in modalità in-line.

Come discusso il protocollo più affetto da problematiche di sicurezza è il SIP. Esistono ormai rules (regole caricate in snort) che proteggono dagli attacchi più comuni.

Per citarne alcuni (Listato 9).

Si nota in questo stralcio di rules della bleedingthreads la volontà di proteggere la continuità di servizio, componente assolutamente necessaria in un servizio voip.

In quest'altro esempio la protezione è da una vulnerabilità nota (Listato 10).

Questa (rules della community di snort) ci protegge da possibili violazioni dannose. ●

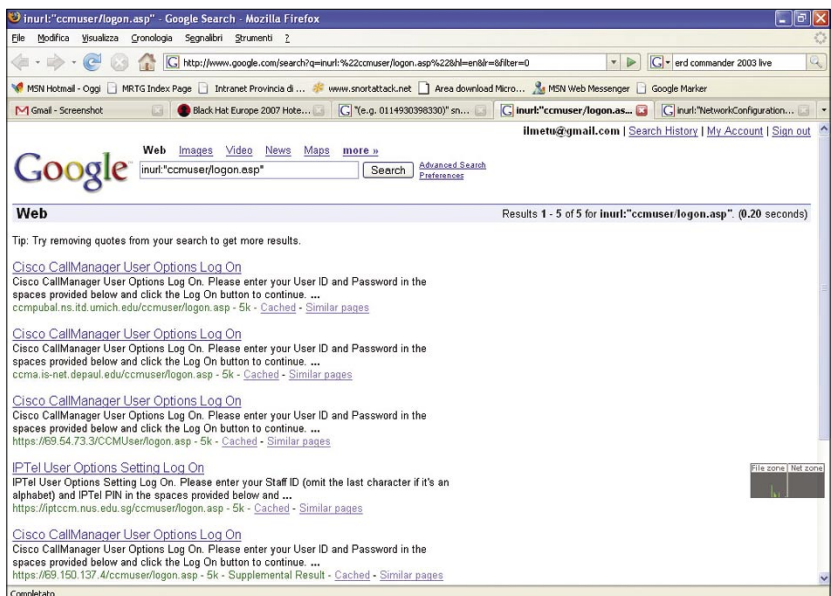


Fig. 4. Footprinting con google