La gestione degli incidenti informatici
v1.2 del 20/12/2005

Executive Summary

Questo documento contiene una serie di suggerimenti per ridurre i danni provocati da un'intrusione informatica, velocizzare la sua scoperta e su come comportarsi nell'eventualità che si verifichi.

Si consiglia:

- eseguire backup periodici del sistema e conservare copia delle distribuzioni del sistema operativo e delle patch applicate;
- eseguire controlli di integrità dei file system;
- conservare copia di tutti i log su di una macchina dedicata;
- preparare un archivio di tool adatti a rivelare lo stato del sistema.

Introduzione

Questo documento segue abbastanza fedelmente le linee guida riportate in [1], fatte salve le necessarie modifiche per tenere conto delle specifiche situazioni a cui si rivolge e gli aggiornamenti resi necessari dall'evolvere della situazione tecnologica.

Anche se il documento è strutturato come un "ricettario", va tenuto presente che l'argomento trattato, vista la sua variabilità, non è facilmente organizzabile in uno schema rigido come quello esposto. In altri termini: non si tratta di consigli validi in assoluto, ma solo nella maggior parte dei casi. È sempre all'amministratore del sistema, con la sua conoscenza della realtà locale, che spetta la decisione finale sull'opportunità o meno di certe misure.

In tutto il documento seguente, salvo esplicita indicazione contraria, quando si fa riferimento a problemi di "sicurezza", ci si riferisce sempre alla sicurezza *informatica*.

Operazioni preliminari

Per essere in grado di rispondere efficacemente ad una situazione di improvvisa emergenza quale quella provocata da un incidente di sicurezza, è ovviamente necessario avere predisposto in anticipo una serie di procedure e strumenti che permettano di capire che cosa sta succedendo e come porvi rimedio.

Backup

Tutti i sistemi critici devono essere salvati periodicamente su di un supporto non raggiungibile da eventuali intrusi: né fisicamente, né via rete.

Le copie di backup e le procedure di ripristino vanno verificate periodicamente.

Con la stessa cura delle copie di salvataggio vanno anche conservati i supporti utilizzati per installare il sistema operativo (ad es. le distribuzioni di Linux) e le eventuali patch, in modo da essere sempre in grado di ripristinare il sistema all'ultima configurazione stabile. Va posta particolare attenzione alle patch, i cui file sorgente (ad es. rpm) devono essere salvati su di un supporto stabile (ad es. cd-rom) e non eliminati dopo la loro applicazione.

È necessario eseguire periodicamente dei test sull'integrità dei file di sistema (si consiglia **samhain** [8]). È fondamentale che i database delle checksum non siano modificabili dall'eventuale intruso (ad es. **samhain** permette di conservarli su di un'altra macchina).

File di log centralizzati

È importante che i file di log di tutte le macchine critiche siano conservati anche su di una macchina dedicata e accuratamente protetta. In questo modo, anche se l'intruso li dovesse eliminare sul sistema attaccato, rimarrebbe possibile cercare di ricostruire l'accaduto.

Nel caso di sistemi unix, per inviare copia dei log su di un sistema remoto bisogna aggiungere delle righe come la seguente in **`syslog.conf`**:

```
*.info;mail.none;authpriv.none      @loghost
```

dove **`loghost`** è il nome della macchina remota.

Per ridurre i rischi di attacchi di DoS al server di log, il suo **syslogd** deve essere configurato in modo da accettare input solo dalle macchine servite.

Uno strumento che può essere di aiuto per esaminare i file di log, anche in tempo reale, è **wots** [10].

Mantenersi informati

È fondamentale essere al corrente delle ultime vulnerabilità, in modo da poter decidere se i propri sistemi debbano o meno essere aggiornati.

Riteniamo che come ragionevole compromesso tra completezza di informazione e tempo disponibile siano sufficienti i "Technical Cyber Security Alerts" pubblicati da US-CERT [2]

Nel caso si desideri una maggiore informazione (e si abbia il tempo disponibile) esistono numerosi portali specificamente dedicati al problema ([3]-[7]). Una mailing list ha il vantaggio che non richiede una partecipazione "attiva".

Strumenti

Per gli strumenti di network intrusion detection si rimanda allo specifico documento.

Di seguito sono elencati alcuni strumenti, tra quelli di pubblico dominio, ritenuti particolarmente utili.

La scelta, tra le centinaia esistenti, non pretende di essere la migliore in assoluto, ma solo un punto di partenza da personalizzare a seconda delle diverse realtà:

- f*ile integrity checking*: **samhain** [8];
- analisi dei file di log : **wots** [9];
- controllo presenza *rootkit*: **chkrootkit** [10];
- *sniffer*: **tcpdump** [11] e **ethereal** [12]**;**
- controllo porte aperte: **lsof** (unix) [13] e **Active Ports** (windows) [14];
- port scanner: **nmap** [15].
- mini distribuzione linux (per copie salvataggio): **tomsrtbt** [16];

Gestione dell'intrusione

Spesso in una macchina compromessa le utilità di sistema (ad es. **ps**, **netcat**, **ls**, **find**) sono state modificate dall'intruso in modo da non mostrare alcuni processi, utenti o file. Programmi come **chkrootkit** [10] o di *file integrity checking* [8] possono aiutare nella rilevazione di questo tipo di compromissione. *È dunque indispensabile utilizzare sempre versioni sicuramente "originali" delle utilità di sistema*, recuperate da una copia di salvataggio anteriore alla compromissione.

Raccolta informazioni preliminari

Una volta rilevata un'intrusione, per determinare le modalità e l'entità della compromissione è necessario raccogliere quante più informazioni possibili sullo stato del sistema. Alcune lo devono essere *prima* dell'isolamento del sistema:

- connessioni di rete;
- processi attivi;
- utenti attivi;
- file aperti.

lsof e **Active Ports** sono molto utili per questo tipo di rilevazioni (cfr. Appendice).

Isolamento del sistema

Una volta raccolte le informazioni preliminari è necessario "isolare" il sistema compromesso, per evitare l'estendersi dei danni e per poterlo studiare in una situazione controllata. Alcune volte l'intruso, per cautelarsi, provvede ad installare delle procedure che cancellano le sue tracce (o addirittura l'intero sistema) nel caso di perdita di connettività di rete: può quindi essere consigliabile spegnere la macchina agendo sul pulsante di alimentazione.

Ricerca di altri sistemi compromessi

È probabile che il sistema compromesso non sia unico nella rete locale, specialmente se esistono altre macchine con lo stesso tipo di configurazione (ad es. stessa versione del sistema operativo e stessi servizi attivi).

La ricerca di altri sistemi compromessi sulla stessa rete locale può essere facilitata dall'esame dei log dei router e dei sistemi di *network intrusion detection*. Ad esempio un sistema come **argus** [17] può facilmente dare risposta alle domande seguenti, di grande aiuto per cercare di capire che cosa possa essere successo (cfr. Appendice):

- quali nodi si sono collegati al nodo compromesso?
- quanti di questi si sono collegati ad altri nodi della LAN?
- quali connessioni (interne ed esterne) sono state fatte dal sistema dopo la compromissione?

Backup

Dopo lo spegnimento, controllato o meno, la macchina va riavviata in modalità *stand-alone,* per evitare l'esecuzione di eventuali procedure automatiche installate dall'intruso.

Prima di qualsiasi altra operazione va eseguito un backup completo del sistema, anche per eventuali fini legali, meglio se in doppia copia e su di un supporto non modificabile.

Identificazione del metodo di attacco e delle azioni dell'intruso

L'identificazione del metodi di attacco è spesso molto difficile se non si dispongono dei file di log *esterni* al nodo compromesso. Infatti una delle prime azioni dell'intruso è la cancellazione delle sue tracce dai file di log del sistema attaccato.

I log del router e di eventuali sistemi di *Network Intrusion Detection* possono essere utili se l'attacco ha sfruttato una vulnerabilità di un qualche servizio di rete.

Nel caso che l'intruso abbia utilizzato la macchina come server (ad es. IRC BOT) o per scansioni di altri nodi o vi abbia installato uno *sniffer,* i file di log prodotti possono essere di aiuto. Generalmente vengono nascosti in directory in posti insoliti (ad es. in **`/dev`**) o con nomi che possono passare inosservati ad un esame superficiale (ad es. spazio o …).

Comunicazione dell'intrusione

Si ricorda che l'intrusione informatica è un reato e che quindi va denunciata alle autorità competenti (ad es. la Polizia Postale).

A parte gli obblighi di legge, è comunque importante segnalare l'incidente al referente locale per la sicurezza. Informazioni sul responsabile della rete da cui proviene l'attaccante sono ottenibili con il comando **whois** (cfr. Appendice).

In ogni caso sono utili le seguenti informazioni:

- estremi del contatto;
- data e ora dell'incidente;
- precisione del clock della macchina;
- origine e vittima dell'attacco;
- altri siti coinvolti;
- eventuali file di log

Ulteriori informazioni sono disponibili sul server web di GARR-CERT (**http://www.cert.garr.it/**).

Recupero dall'incidente

La maniera più sicura per eliminare tutte le possibili *backdoor* lasciate dall'intruso – software di sistema opportunamente modificato, nuovi account, nuovi servizi, ecc. ecc. -- è una completa reinstallazione del sistema, seguita dall'applicazione di tutte le *patch* di sicurezza ritenute necessarie.

A reinstallazione avvenuta, e comunque in ogni caso, è necessario cambiare *tutte* le password, controllando, nello stesso tempo, che non siano stati aggiunti (o riabilitati) utenti.

Nel caso si riutilizzino i file di configurazione in uso prima della compromissione, questi vanno attentamente esaminati per controllare la presenza di modifiche destinate a permettere il ritorno dell'intruso. Ad esempio:

- abilitazione di nuovi servizi;
- programmi suid che permettono ottenere i privilegi di root;
- script automatici via **cron**;
- esportazione di volumi via **nfs**.

I dischi che non sono formattati durante la reinstallazione vanno esaminati alla ricerca di programmi *suid* (ad esempio utilizzando il comando **find**).

Bibliografia

[1] K-P. Kossakowski et al., *Responding to Intrusions* (CMU/SEI-SIM-006), Software Engineering Institute, Carnegie Mellon University (1999). Reperibile all'indirizzo: http://www.sei.cmu.edu/pub/documents/sims/pdf/sim006.pdf

[2] http://www.us-cert.gov/cas/signup.html

[3] CIAC: http://www.ciac.org/ciac/

[4] SANS Institute : http://www.sans.org/

[5] ICAT Vulnerability Search Engine: http://icat.nist.gov/icat.cfm

[6] SecurityFocus: http://www.securityfocus.com

[7] Sysman: http://sysman.na.infn.it/

[8] http://www.la-samhna.de/samhain/ (http://www.la-samhna.de/samhain/library/scanners.html per un confronto tra alcuni sistemi esistenti)

[9] http://www.hpcc.uh.edu/~tonyc/tools/

[10] http://www.chkrootkit.org/

[11] http://www.tcpdump.org/

[12] http://www.ethereal.com/

[13] ftp://vic.cc.purdue.edu/pub/tools/unix/lsof/

[14] http://www.protect-me.com/freeware.html

[15] http://www.insecure.org/nmap

[16] http://www.toms.net/rb/

[17] http://www.qosient.com/argus/

Controllo dei file di log

Il seguente è un esempio di file di configurazione di **wots** (le righe che cominciano con # sono commenti).

Il primo campo della riga è un'espressione *regexp* (perl), il secondo l'azione da intraprendere nel caso che sia trovata nella riga del file.

Da notare che **wots** può analizzare più di un file contemporaneamente

```
# nome del file da analizzare
from /var/log/messages

##################################
# righe da ignorare (ntpd, named e ssh)
/xntpd/                        ignore
/named-xfer\[.+\]:/            ignore
/ssh.*Generating.*key/         ignore

# queste righe vengono evidenziate
/auth failure/                 echo=red;bold,mail,exec=alarm.sh
/tftpd/                        echo=white;on_red
/telnet|ftp/                   echo=magenta;bold
/rlogin|rexec|rsh|remsh/       echo=red;on_green

# tutte le altre righe vengono mostrate
/./                            echo

# altro file da analizzare
from /var/log/authlog
##################################
/generating.+RSA key/   ignore
/./                     echo=red
```

rootkit

Con il nome di *rootkit* si indicano quei pacchetti che permettono all'intruso di mascherare le sue tracce. Nel caso più semplice contengono versioni modificate delle principali utility di sistema che non segnalano la presenza di alcuni processi e file, che permettono di riottenere accesso privilegiato al sistema e che facilitano la cancellazione delle tracce dell'attacco dai file di log.

I file di configurazione sono contenuti in directory opportunamente 'mascherate', ad es. '...', o con nomi apparentemente di sistema, ad es. `/dev/pty00`.

Uno dei rootkit più diffusi, ad esempio, consiste in

- **chfn**, **chsh**, **passwd** modificati per permettere di diventare *root*;
- **du**, **find**, **ls** modificati per nascondere alcuni file e directory (indicati nei file di configurazione);
- **ifconfig** modificato per non mostrare il flag di modo promiscuo (indicatore della presenza di uno sniffer);
- vari sniffer;
- **login** modificato per permettere login come *root*;
- **netstat**: modificato per nascondere particolari connessioni;
- **ps**, **top** modificati per nascondore certi processi;
- **syslogd** modificato per non scrivere su syslog certe stringhe;
- **tcpd** modificato per permettere l'accesso senza login da alcuni host;
- **wted**, **z2** per modificare **utmp**, **wtmp** e **lastlog**.

Questo tipo di rootkit viene facilmente individuato da un programma come **samhain**.

L'altro tipo di rootkit (ad es. **Knark**, **Adore**, **Rtkit**), invece, si basa sulla modifca del *kernel* del sistema attaccato, lasciando quindi inalterate le utility: è proprio il kernel che non fornisce loro le informazioni che l'intruso vuole nascondere. Questo tipo di rootkit vine, di solito, rivelato con dei programmi specializzati (ad es **chkrootkit**), che però vanno tenuti costantemente aggiornati.

Controllo connessioni di rete e processi

netstat ed **lsof** permettono di verificare quali connessioni di rete sono attive e quali processi le hanno aperte.

Questo è un esempio tipico di output di **netstat**

```
# netstat -a
Proto   Recv-Q   Send-Q    Local Address    Foreign Address         State
tcp          0        0       *:sunrpc           *:*               LISTEN
tcp          0        0       *:auth             *:*               LISTEN
tcp          0        0       *:ssh              *:*               LISTEN
tcp          0       20    host:ssh         pcc.es:4325      ESTABLISHED
udp          0        0       *:syslog           *:*
udp          0        0       *:sunrpc           *:*
udp          0        0       *:2345             *:*
```

lsof permette di scoprire quale processo è in ascolto sulla porta 2345 (un'informazione che **netstat** non fornisce)

```
# lsof -i | grep 2345
nc       12112     root    3u   inet    0x01437018     0t0  UDP *:2345
```

lsof permette anche di scoprire quali sono i file aperti da un processo. Nell'esempio seguente si cercano i file aperti da un processo che si chiama 'ps' (un nome spesso utilizzato per mascherare processi illegittimi:

```
# lsof | grep ps
ps       28637     root    cwd  VDIR   31,0x5000   3072   10240  /dev/ttyq2
```

Esempio di utilizzo di argus

Il comando seguente estrae dal database creato da **argus** (`argus.out`) le connessioni fatte e ricevute dal nodo **vittima** (il numero dopo il nome del nodo è la porta e i numeri seguenti sono i byte e i pacchetti in entrata e in uscita).

```
ra -r argus.out -c host VITTIMA
08/11 20:49:36 *  tcp  RETEVISION.1031   ->      VITTIMA.23    1550  1505   56659   41352   CLO
08/11 20:50:51    tcp     VITTIMA.1067   -> TECHNOTRONIC.21    26    19     141     1332    CLO
08/11 20:51:12    tcp     VITTIMA.1068  <-  TECHNOTRONIC.20    4     5      0       2451    CLO
08/11 21:14:59    tcp     VITTIMA.1071   ->       BAROLO.21    20    16     109     410     CLO
08/11 21:15:10    tcp     VITTIMA.1072  <-        BAROLO.20    6     9      0       8145    CLO
08/11 21:26:25 *  tcp RETEVISION2.1028   ->      VITTIMA.23    2405  2362   1658    146227  CLO
08/11 21:29:47    tcp     VITTIMA.1080   ->      TITANIA.23    283   239    161     13700   CLO
08/11 21:29:47    tcp     TITANIA.26862  ->      VITTIMA.113   5     5      9       36      CLO
08/11 22:40:11 *  tcp RETEVISION2.1053   ->      VITTIMA.23    4938  5342   458935  90861   CLO
08/11 23:07:27 s  tcp     VITTIMA.1143   o>         IRC1.6667  2     0      0       0       TIM
08/11 23:10:22    tcp     VITTIMA.1147  <|          IRC2.6667  1     1      0       0       RST
08/11 23:10:56 d  tcp     VITTIMA.1152   ->         IRC3.6667  169   179    1367    14136   CLO
08/11 23:10:57    tcp        IRC3.22133  ->      VITTIMA.113   5     5      13      39      CLO
08/11 23:24:23 *  tcp     VITTIMA.1168   |>         IRC4.6667  143   125    884     11665   RST
08/11 23:24:24    tcp        IRC4.4531   ->      VITTIMA.113   7     6      169     38      CLO
```

whois

Il comando **whois** (ci sono anche varianti web) permette di interrogare i vari database in cui sono conservate le informazioni di riferimento per tutti gli indirizzi ip assegnati. Purtroppo, però, spesso queste informazioni sono lacunose e obsolete.

I database sono diversi a seconda della collocazione geografica della rete:

- per le reti europee:
  `whois –h whois.ripe.net xxx.xxx.xxx.xxx`
- per le reti americane (ha anche qualche informazione sulle altre reti):
  `whois –h whois.arin.net xxx.xxx.xxx.xxx`
- per le reti asiatiche:
  `whois –h whois.apnic.net xxx.xxx.xxx.xxx`

dove `xxx.xxx.xxx.xxx` è l'indirizzo ip del nodo.

Questo è un esempio di output:

```
# whois -h whois.ripe.net 192.84.145.1

inetnum:      192.84.145.0 - 192.84.145.255
netname:      INFNET20
descr:        INFN (National Institute of Nuclear Physics)
descr:        Sezione di Firenze
country:      IT
admin-c:      RC4178-RIPE
tech-c:       RC4178-RIPE
rev-srv:      avaxfi.fi.infn.it
remarks:      Firenze INFN Department
remarks:      GARR - Italian academic and research network
remarks:      to notify any abuse cert@garr.it

route:        192.84.144.0/21
descr:        Aggregated GARR routes

person:       Roberto Cecchini
address:      INFN, Sezione di Firenze
address:      L.go E. Fermi 2
address:      I 50125 Firenze
phone:        +39 55 2307696
e-mail:       roberto.cecchini@fi.infn.it
nic-hdl:      RC4178-RIPE
```

Le informazioni utili, di solito, sono quelle sul responsabile amministrativo (righe `admin-c` e `nic-hdl`) e tecnico (righe `tech-c` e `nic-hdl`). Qualche volta, come nel caso di sopra, sono presenti righe addizionali (`remarks`) che forniscono informazioni utili proprio in caso di incidenti.