Servizi centralizzati
v1.2 (20/12/05)

1. Premessa

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 ovviamente sempre all’amministratore del sistema, con la sua conoscenza della realtà ed esigenze locali, che spetta la decisione finale sull’opportunità o meno di certe misure.

2. Criteri generali

Le macchine destinate alla gestione di servizi centralizzati, per ovvi motivi, vanno protette con particolare cura. Oltre alle solite:

- mantenere un sistema di backup, verificandone periodicamente l’efficienza;
- evitare la trasmissione di password in chiaro, ad esempio utilizzando **ssh**, SSL/TLS e VPN;
- bloccare porte e servizi non necessari e controllare periodicamente il sistema con uno scanner esterno;
- rimuovere il software non strettamente necessario e account e gruppi installati per default e non utilizzati;
- mantenere i sistemi operativi e il software applicativo aggiornati, istallando sollecitamente le patch di sicurezza fornite dai distributori;
- utilizzare strumenti per la verifica dell’integrità dei file di sistema, mantenendo aggiornati i loro database (ad esempio **...**);
- proteggere i sistemi da *virus* e *worm*;

vanno realizzate almeno le seguenti specifiche norme di sicurezza:

- limitare il numero di servizi critici per macchina allo scopo di limitare interazioni — non sempre facilmente prevedibili — tra i servizi e limitare i danni in caso di compromissione;
- consentire l’accesso interattivo solo ai gestori, solo via SSH e solo da alcuni nodi;
- limitare l’accesso fisico;
- salvare i log su una macchina esterna (Log Server), possibilmente dedicata, sottoponendoli ad analisi periodica, meglio se con l’aiuto di strumenti apposito.

Maggiori dettagli sono disponibili nel documento “La gestione degli incidenti informatici”.

La seguente è una possibile distribuzione:

- Server DNS primario
- Server DNS secondario, Log server

- Server web
- Server NAT, Server DHCP
- Mail Transport Agent, Local Delivery Agent
- Server ftp.

A queste è consigliabile aggiungere almeno una macchina per scansioni e network intrusion detection, da collegare, ad esempio, ad una porta in mirroring mode dello switch cui è connesso anche il router di collegamento con l'esterno.

3. Server di posta elettronica

3.1 Mail Transport Agent (MTA)

Si consiglia l'uso di uno dei seguenti software (in ambiente Unix)

- **sendmail**
- **postfix**

Entrambi, se correttamente configurati, sono in grado di garantire un buon livello di sicurezza. Devono essere configurati in modo da:

- bloccare l'"*unauthorized relaying*";
- permettere l'autenticazione degli utenti per la spedizione di mail, utilizzando anche la porta 587: essenziale per consentire l'uso del server da remoto e preliminare all'installazione di meccanismi di verifica del mittente (ad es. SPF o DomainKey);
- permettere di autenticarsi in maniera sicura;
- utilizzare un sistema antivirus centralizzato.

postfix è sicuramente più semplice da configurare. È stato progettato fin dall'inizio secondo criteri di massima sicurezza e probabilmente è la scelta migliore se non si ha bisogno delle maggiori funzionalità di **sendmail**.

Si raccomanda di permettere l'invio della posta soltanto tramite il server MTA autorizzato, chiudendo sul firewall o router perimetrale la porta TCP 25 in uscita a tutti gli altri host.

Anche se non rientra direttamente fra le richieste per la sicurezza, consigliamo l'installazione di un sistema anti-spam (ad es. **spamassassin**).

3.2 Local Delivery Agent (LDA)

La configurazione ottimale prevede di mantenere su macchine separate le funzioni di MTA e LDA; in ogni caso si consiglia di permettere l'accesso alle mailbox solo tramite Mail User Agent (MUA) bloccando il login con shell interattiva agli utenti.

I sistemi di gestione delle mailbox consigliati sono:

- **cyrus**
- **imap-uw**

Entrambi permettono meccanismi di autenticazione che non trasmettono la password in chiaro (ad esempio utilizzando IMAPS o CRAM-MD5). **cyrus** ha una migliore storia dal punto di vista della sicurezza, però è più difficile da configurare e le mailbox sono in un formato non standard.

3.2.1 Accesso via WEB

L'accesso deve avvenire solo attraverso SSL/TLS.

Un rischio da tenere presente è che questo tipo di accesso può essere impiegato (e spesso lo viene) da postazioni pubbliche (ad es. internet café) dove il browser potrebbe essere configurato in modo da registrare quanto digitato dall'utente. Un'altra possibilità è che il browser memorizzi le informazioni di login in una cache, che l'utente, molto probabilmente, si dimenticherà di cancellare a fine sessione. Per ridurre l'impatto di questo tipo di rischi è molto importante istruire gli utenti e richiedere che la password della mailbox sia diversa dalle loro altre password.

3.3 Mail Client

I mail client devono essere mantenuti aggiornati alle ultime patch di sicurezza ed usati sempre in unione con un antivirus locale.

4. Server DNS

Il name server primario è uno dei servizi più critici ed è quindi molto importante che sia installato su di una macchina dedicata.

Il processo deve girare senza i privilegi di root (è possibile nelle ultime versioni di **named**) o in un ambiente *chroot*.

Gli *zone-transfer* vanno consentiti solamente verso i name server secondari.

5. Server DHCP

Gli indirizzi dovrebbero essere assegnati solamente dopo che l'utilizzatore del nodo ha comunicato il MAC Address al Servizo Calcolo.

Gli indirizzi assegnati dovrebbero essere su di una rete nascosta.

6. Server WEB

Un server web è per sua natura difficilmente difendibile; per evitare quindi che la sua compromissione abbia conseguenze più gravi, incidendo su altri servizi, si consiglia di installarlo su di una macchina dedicata.

Il processo del server web deve avere un insieme molto ridotto di privilegi, in particolare:

- deve avere accesso in sola lettura ai file pubblici e non deve avere accesso agli altri file esterni (ad esempio via link);
- i file di configurazione e di log devono essere all'esterno delle directory pubbliche;
- non usare link che puntano all'esterno dell'albero pubblico;
- i file temporanei creati dalle applicazioni web devono essere confinati in directory specifiche, all'interno delle directory pubbliche, e il loro accesso deve essere limitato ai soli processi che li hanno creati.

Per quello che riguarda la configurazione delle caratteristiche del server web si raccomanda di:

- limitare la possibilità di modifica locale dei controlli globali di accesso (ad es. i file **`.htaccess`** di Apache);
- disabilitare il listing automatico delle directory;
- limitare il più possibile l'uso di script e plug-in, e, in ogni caso, mantenerli sotto il controllo esclusivo degli amministratori (gli script sono la principale causa di vulnerabilità di un server web);
- disabilitare l'uso dei Server Side Include (SSI) perché possono permettere agli utenti di eseguire qualunque programma sul server (oltre ad avere ripercussioni negative sulle prestazioni);
- non abilitare gli upload (directory **`incoming`**) via ftp anonimo sul server web, o, se proprio non si può farne a meno, non consentire l'accesso dal server web alla directory di upload.

Per le pagine che richiedono autenticazione utlizzare solamente connessioni SSL per evitare di trasmettere dati critici in chiaro.

Nel caso si impieghino script **cgi-bin** o simili:

- cercare di evitare che lo script faccia lo *spawn* di una shell (ad esempio comando **`system`** in **perl**): se viene passata una stringa non controllata per l'esistenza di caratteri speciali possono essere eseguiti comandi arbitrari sul server;
- filtrare le stringhe di input per caratteri speciali e controllarne la lunghezza per evitare i buffer overflow;
- non utilizzare script SUID: è praticamente impossibile renderli sicuri; se uno script deve proprio girare con un UID diverso da quello di **httpd**, usare **CGIwrap** o **sbox**.

I file di log devono essere conservati ed analizzati con regolarità. Fare attenzione in particolare alle invocazioni di programmi di sistema (ad es. **rm**, **login**, **chmod**) e a URL molto lunghe (potrebbero essere indicative di tentativi di *buffer overflow*).

7. Server FTP

Un server FTP che consenta l'accesso anonimo deve essere configurato con la massima cura onde evitare che venga utilizzato da terzi per trasferimenti non autorizzati.

La root directory e le sue sottodirectory non devono essere di proprietà dell'utente ftp né del suo gruppo: la soluzione più semplice è che appartengano a root.

Il seguente è un tipico esempio di directory per ftp anonimo (tutti i file contenuti devono avere lo stesso tipo di protezione)

```
drwxr-xr-x  7 root system 512 Mar 1 15:17  ./
drwxr-xr-x 25 root system 512 Jan 4 11:30  ../
drwxr-xr-x  2 root system 512 Dec 20 15:43 bin/
drwxr-xr-x  2 root system 512 Mar 12 16:23 etc/
drwxr-xr-x 10 root system 512 Jun 5 10:54  pub/
```

L'accesso agli account di sistema deve essere disabilitato (**`/etc/ftpusers`**).

Il file di password (**`etc/passwd`**) e quello dei gruppi (**`etc/groups`**) *non devono* essere copie di quelli reali, ma contenere solo informazioni relative ai file per l'ftp anonimo.

7.1 Directory scrivibili

Le directory scrivibili pongono particolari problemi di sicurezza e vanno quindi abilitate solo in casi di effettiva necessità.

La seguente è una possibile soluzione (classico esempio di *security through obscurity*, pratica da evitare nei limiti del possibile).

Create una directory top level (**`~ftp/incoming`**) con protezione 751, in modo che l'utente ftp possa solamente eseguirvi **cd**, ma *non* listarne il contenuto:

```
drwxr-x--x 12 root system 512 Jun 5 11:23  incoming/
```

Create sottodirectory con nomi non facilmente indovinabili, noti solamente a chi deve depositare file sul server: se i nomi diventano noti non c'è più nessuna protezione contro l'abuso del server.

Le directory vanno tenute sotto costante controllo per evitare abusi.

8. Bibliografia

Parte generale

Allen et al., *Securing Network Servers*, CMU/SEI, 2000
http://www.sei.cmu.edu/pub/documents/sims/pdf/sim010.pdf
http://www.cert.org/security-improvement/modules/m10.html

Server di Posta Elettronica

NIST Special Publication 800-45, *Guidelines on Electronic Mail Security,* 2002, http://csrc.nist.gov/publications/nistpubs/index.html.

Gruppo di lavoro GARR sec-mail: http://www.garr.it/WG/sec-mail/

Server DNS

http://www.oreilly.com/catalog/dns4/chapter/ch11.html

Server Web

NIST Special Publication 800-44, *Guidelines on Securing Public Web Servers,* 2002 (http://csrc.nist.gov/publications/nistpubs/index.html).

WWW Security FAQ, 2002 (http://www.w3.org/Security/Faq/)

Securing Public Web Servers, CMU/SEI, 2001, (http://www.cert.org/security-improvement/modules/m11.html).

The Unofficial Web Hack FAQ (http://www.windowsecurity.com/whitepaper/websecurity/The_Unofficial_Web_Hack_FAQ/).

CERT advisory on CGI metacharacters http://www.cert.org/tech_tips/cgi_metacharacters.html

COAST Hotlist Content (http://www.cerias.purdue.edu/infosec/hotlist/).

Server FTP

Anonymous FTP Configuration Guidelines, CERT/CC (http://www.cert.org/tech_tips/anonymous_ftp_config.html)