Guida a Ubuntu Server

Guida a Ubuntu Server

Sommario

Benvenuti nella *Guida a Ubuntu server*. Questa guida contiene informazioni su come installare e configurare diverse applicazioni server per Ubuntu a seconda delle proprie esigenze. È una guida passo-passo, orientata ai processi per configurare e personalizzare il sistema.

Riconoscimenti e licenza

Questo documento viene mantenuto dal gruppo documentazione di Ubuntu (https://wiki.ubuntu.com/DocumentationTeam). L'elenco dei collaboratori è consultabile presso *la pagina dei collaboratori*[1]

[3] https://launchpad.net/~ubuntu-core-doc

[1] ../../libs/C/contributors.xml

[2] /usr/share/ubuntu-docs/libs/C/ccbysa.xml

Indice

Lista delle tabelle

Capitolo 1. Introduzione

Benvenuti alla guida a *Ubuntu server*.

In questa guida è possibile trovare informazioni su come installare e configurare diversi applicativi server; è una guida passo-passo, orientata ai processi per configurare e personalizzare il proprio sistema.

This guide assumes you have a basic understanding of your Ubuntu system. Some installation details are covered in *Capitolo 2, Installazione [3]*, but if you need detailed instructions installing Ubuntu please refer to the *Ubuntu Installation Guide*[1].

Una versione HTML di questa guida è disponibile in rete presso *il sito web della documentazione di Ubuntu*[2] e i file HTML sono disponibili anche nel pacchetto ubuntu-serverguide. Per maggiori informazioni sull'installazione dei pacchetti, consultare il *Capitolo 3, Gestione dei pacchetti [17]*.

Se si sceglie di installare ubuntu-serverguide, è possibile visualizzare questo documento da una console digitando:

```
w3m /usr/share/ubuntu-serverguide/html/C/index.html
```

Se si sta utilizzando una versione localizzata di Ubuntu, sostituire *C* con il codice della propria lingua (per esempio *it*).

[1] https://help.ubuntu.com/10.10/installation-guide/

[2] http://help.ubuntu-it.org

1. Supporto

Esistono diverse forme di supporto per la Ubuntu Server Edition: supporto commerciale e dalla comunità. Il supporto commerciale è disponibile attraverso Canonical Ltd.: fornisce contratti di supporto a prezzi ragionevoli per postazione desktop o server. Per maggiori informazioni, consultare il *sito web di Canonical*[3].

Il supporto della comunità è fornito grazie all'impegno di singole persone, o di aziende, che desiderano rendere Ubuntu il migliore sistema operativo possibile. Il supporto viene erogato attraverso l'utilizzo di mailing list, canali IRC, forum, blog, wiki e altro. L'enorme quantità di informazioni disponibili può sembrare schiacciante, ma una valida interrogazione con un motore di ricerca può spesso fornire una risposta ai propri dubbi. Per maggiori informazioni, consultare la pagina *Ubuntu Support*[4].

[3] http://www.canonical.com/services/support
[4] http://www.ubuntu.com/support

Capitolo 2. Installazione

This chapter provides a quick overview of installing Ubuntu 10.10 Server Edition. For more detailed instructions, please refer to the *Ubuntu Installation Guide*[1].

[1] https://help.ubuntu.com/10.10/installation-guide/

1. Preparazione dell'installazione

Questa sezione spiega i diversi aspetti da considerare prima di avviare l'installazione.

1.1. Requisiti di sistema

Ubuntu 10.10 Server Edition supports two (2) major architectures: Intel x86 and AMD64. The table below lists recommended hardware specifications. Depending on your needs, you might manage with less than this. However, most users risk being frustrated if they ignore these suggestions.

Tabella 2.1. Requisiti minimi raccomandati

Tipo di installazione	RAM	Spazio disco fisso
		Installazione completa di base
Server	128 megabyte	500 megabyte

La Server Edition fornisce una base comune per tutte le tipologie di applicazioni server: ha una progettazione minimalista in grado di fornire una piattaforma per qualsiasi servizio desiderato come servizi di file e stampa, host web, email, ecc...

The requirements for UEC are slightly different for Front End requirements see *Sezione 3.2.1, «Front End Requirements» [276]* and for UEC Node requirements see *Sezione 3.2.2, «Requisiti del nodo» [277]*.

1.2. Differenze tra Server e Desktop

Esistono alcune differenze tra *Ubuntu Server Edition* e *Ubuntu Desktop Edition*. È utile ricordare però che entrambe le edizioni utilizzano i repository apt, rendendo l'installazione di un'applicazione *server* sulla Desktop Edition facile come sulla Server Edition.

Le differenze tra le due edizioni sono la mancanza dell'ambiente X nella Server Edition, il processo di installazione e diverse opzioni per il kernel.

1.2.1. Differenze del kernel

- La Server Edition utilizza lo scheduler di I/O *deadline* al posto di *CFQ* usato nella Desktop Edition.
- L'opzione di *preemption* è disabilitata nella Server Edition.
- Il timer dell'interrupt è di 100Hz nella Server Edition, 250Hz nella Desktop Edition.

Usando una versione a 64-bit di Ubuntu su processori a 64-bit non si è limitati nello spazio di indirizzamento della memoria.

To see all kernel configuration options you can look through `/boot/config-2.6.35-server`. Also, *Linux Kernel in a Nutshell*[2] is a great resource on the options available.

1.3. Effettuare una copia di backup

- Prima di installare Ubuntu Server Edition è utile creare una copia di sicurezza di tutti i dati nel sistema. Per maggiori informazioni sulle opzioni di backup, consultare il *Capitolo 18, Backup [247]*.

  Se non è la prima volta che viene installato un sistema operativo nel computer, potrebbe essere necessario ripartizionare il disco fisso per creare spazio per l'installazione di Ubuntu.

  A ogni partizionamento del disco fisso è necessario essere preparati per eventuali perdite di dati causate da errori o da malfunzionamenti nel sistema di partizionamento. I programmi usati durante l'installazione sono sicuri e usati da molti anni, ma possono anche eseguire azioni distruttive.

[2] http://www.kroah.com/lkn/

2. Installare da CD

I passi necessari per installare Ubuntu Server Edition da CD sono gli stessi per installare un qualsiasi sistema operativo da CD. Diversamente dalla *Desktop Edition*, la *Server Edition* non comprende un programma di installazione grafica, ma utilizza un processo via console.

- Per prima cosa scaricare il file ISO appropriato dal *sito web di Ubuntu*[3].
- Avviare il sistema dal CD-ROM.
- Al prompt viene chiesto di selezionare la lingua. Successivamente il processo di installazione richiede la disposizione della tastiera.
- From the main boot menu there are some additional options to install Ubuntu Server Edition. You can install a basic Ubuntu Server, or install Ubuntu Server as part of a *Ubuntu Enterprise Cloud*. For more information on UEC see *Sezione 3, «UEC» [276]*. The rest of this section will cover the basic Ubuntu Server install.
- Il programma d'installazione rileva l'hardware e configura le impostazioni di rete utilizzando il servizio DHCP. Per non utilizzare questo servizio, alla schermata successiva scegliere «Indietro» e quindi scegliere l'opzione per configurare la rete manualmente.
- Vengono chiesti il nome host e il fuso orario.
- È quindi possibile scegliere diverse opzioni di configurazione per il proprio disco fisso. Per le opzioni avanzate, consultare *Sezione 4, «Installazione avanzata» [10]*.
- Il sistema base Ubuntu viene quindi installato.
- Un nuovo utente viene configurato con accesso *root* attraverso l'uso di sudo.
- Configurato l'utente, viene richiesto di cifrare la propria directory `home`.
- Il passo successivo nel processo di installazione consiste nel decidere come aggiornare il sistema. Sono disponibili tre opzioni:
  - *Nessun aggiornamento automatico*: richiede che un amministratore si colleghi al computer e installi manualmente gli aggiornamenti.
  - *Installare automaticamente gli aggiornamenti di sicurezza*: installa il pacchetto unattended-upgrades che garantisce l'installazione di aggiornamenti di sicurezza senza l'intervento di un amministratore. Per maggiori informazioni, consultare *Sezione 5, «Aggiornamenti automatici» [25]*.
  - *Gestire il sistema con Landscape*: Landscape è un servizio a pagamento fornito da Canonical che consente di gestire diversi computer con Ubuntu installato. Per maggiori informazioni, consultare la *pagina web dedicata a Landscape*[4].
- Ora è possibile scegliere se installare o non installare diversi pacchetti per attività specifiche (tasks) (per maggiori informazioni, consultare *Sezione 2.1, «Pacchetti per attività specifiche» [7]*). È inoltre presente un'opzione per lanciare il programma aptitude per scegliere dei pacchetti specifici da installare. Per maggiori informazioni, consultare *Sezione 4, «Aptitude» [23]*.
- Infine, prima di riavviare, è necessario impostare l'orologio a UTC.

Se durante l'installazione non si è soddisfatti delle impostazioni predefinite, usare la funzione «Indietro» per visualizzare un menù d'installazione dettagliato che consente di modificare le impostazioni.

In qualsiasi momento dell'installazione è possibile leggere la guida fornita dal sistema, basta premere F1.

Once again, for detailed instructions see the *Ubuntu Installation Guide*[5].

2.1. Pacchetti per attività specifiche

Durante l'installazione della Server Edition è possibile installare dei pacchetti aggiuntivi, raggruppati per il tipo di servizio che forniscono.

- Server DNS: seleziona il server DNS BIND e la documentazione.
- Server LAMP: seleziona un server Linux/Apache/MySQL/PHP.
- Server mail: seleziona diversi pacchetti utili per un server di posta generale.
- Server OpenSSH: seleziona i pacchetti necessari per un server OpenSSH.
- Server PostgreSQL: seleziona i pacchetti client e server per il database PostreSQL.
- Server di stampa: configura il sistema come un server di stampa.
- Server file Samba: configura il sistema come server di file Samba, utile particolarmente all'interno di reti eterogenee, con sistemi Windows e Linux.
- Tomcat Java server: Installs Apache Tomcat and needed dependencies.
- Virtual Machine host: Includes packages needed to run KVM virtual machines.
- Manually select packages: Executes aptitude allowing you to individually select packages.

L'installazione di questi pacchetti è svolta utilizzando l'utilità tasksel. La grande differenza tra Ubuntu (o Debian) e le altre distribuzioni GNU/Linux è che, una volta installato, un pacchetto è configurato a valori predefiniti ragionevoli, eventualmente chiedendo le informazioni aggiuntive. Installando un "task", i pacchetti non vengono solo installati, ma anche configurati per fornire un servizio integrato.

Una volta completata l'installazione, è possibile vedere un elenco dei "task" disponibili digitando il seguente comando:

```
tasksel --list-tasks
```

L'output elenca i "task" di altre distribuzioni basate su Ubuntu come Kubuntu ed Edubuntu. È comunque possibile invocare il comando **tasksel**, che presenta un menù con i diversi "task" disponibili.

Tramite l'opzione *--task-packages* è possibile visualizzare un elenco dei pacchetti installati con ogni "task". Per esempio, per elencare i pacchetti installati con *DNS Server* digitare:

[5] https://help.ubuntu.com/10.10/installation-guide/

```
tasksel --task-packages dns-server
```

L'output del comando dovrebbe essere:

```
bind9-doc
bind9utils
bind9
```

Inoltre, se non è stato installato uno dei "task" durante il processo di installazione, ma si vuol far diventare il server LAMP un server DNS, basta inserire il CD di installazione e da un terminale digitare:

```
sudo tasksel install dns-server
```

3. Avanzamento di versione

Ci sono diversi metodi per eseguire un avanzamento da un rilascio di Ubuntu a un altro. In questa sezione vengono presentati i metodi raccomandati.

3.1. do-release-upgrade

Il metodo di avanzamento raccomandato per la Server Edition è l'utilizzo dell'utilità do-release-upgrade, installata in modo predefinito come parte del pacchetto *update-manager-core* e priva di alcuna dipendenza grafica.

I sistemi basati su Debian possono ricorrere anche al comando **apt-get dist-upgrade**. L'uso di do-release-upgrade è comunque raccomandato in quanto è in grado di gestire le modifiche necessarie alla configurazione di sistema tra i rilasci.

Per avanzare a un nuovo rilascio, da un terminale digitare:

```
do-release-upgrade
```

È anche possibile usare do-release-upgrade per avanzare a una versione di sviluppo di Ubuntu. Per fare ciò, usare l'opzione *-d*:

```
do-release-upgrade -d
```

Avanzare a una versione di sviluppo *non* è consigliato in ambienti di produzione.

4. Installazione avanzata

4.1. RAID software

Il sistema RAID è un sistema per configurare diversi dischi fissi in modo che agiscano come un unico grande disco, riducendo le probabilità di perdite di dai nel caso in cui un disco si rompa. Il sistema RAID può essere implementato sia via software (in cui il sistema operativo è a conoscenza dei dischi e li gestisce tutti) sia via hardware (tramite un sistema di controllo che fa credere al sistema operativo di avere un solo disco).

Il RAID software incluso nelle attuali versioni di Linux (e Ubuntu) è basato sul driver mdadm e funziona perfettamente, molto meglio di alcuni cosiddetti controller RAID hardware. In questa sezione viene spiegato come installare Ubuntu Server Edition utilizzando due partizioni RAID1 su due dischi fissi, uno utilizzato per */* e l'altro come *swap*.

4.1.1. Partizionamento

Seguire i passi dell'installazione fino a giungere a *Partizionamento dei dischi*, quindi:

1. Selezionare *Manuale* come metodo di partizionamento.
2. Selezionare il primo disco fisso e acconsentire alla domanda *Creare una nuova tabella delle partizioni sul dispositivo*.

   Ripetere questo passo per ogni disco da inserire nell'array RAID.
3. Selezionare lo *spazio libero* sul primo disco e quindi selezionare *Creare una nuova partizione*.
4. Selezionare la *Dimensione* della partizione: questa partizione sarà quella di *swap* e come regola generale, la dimensione della partizione di swap è solitamente il doppio della memoria RAM. Digitare la dimensione della partizione, scegliere *Primaria* e quindi *Inizio*.
5. Select the *"Use as:"* line at the top. By default this is *"Ext4 journaling file system"*, change that to *"physical volume for RAID"* then *"Done setting up partition"*.
6. Per la partizione */*, selezionare *spazio libero* sul primo drive e quindi *Crea una nuova partizione*.
7. Utilizzare il restante spazio libero sul dispositivo e scegliere *Continua*, quindi *Primaria*.
8. As with the swap partition, select the *"Use as:"* line at the top, changing it to *"physical volume for RAID"*. Also select the *"Bootable flag:"* line to change the value to *"on"*. Then choose *"Done setting up partition"*.
9. Ripetere i passi dal 3 al numero 8 per gli altri dischi e partizioni.

4.1.2. Configurare RAID

Impostate le partizioni è quindi possibile configurare gli array:

1. Nella sezione di partizionamento dei dischi, selezionare *Configurare il software RAID*.
2. Selezione *Sì* per scrivere le modifiche sul disco.

3. Choose *"Create MD device"*.
4. Per questo esempio, selezionare *RAID1*. Nel caso si stia utilizzando una diversa configurazione, scegliere la tipologia adatta (RAID0 RAID1 RAID5).

   

   Per poter usare il *RAID5* sono necessari almeno *tre* dischi. Per RAID0 oppure RAID1 solo *due*.
5. Inserire il numero dei dispositivi attivi (active), *2*, oppure il numero totale dei dischi disponibili per l'array, quindi selezionare *Continua*.
6. Inserire il numero dei dispositivi di scorta (spare), *0* come valore predefinito, quindi selezionare *Continua*.
7. Scegliere la partizione da usare: solitamente sda1, sdb1, sdc1, ecc... I numeri e le lettere solitamente corrispondono a diversi dischi fissi.

   Per la partizione di *swap* scegliere *sda1* e *sdb1*. Selezionare *Continua* per andare al passo successivo.
8. Ripetere i passi dal *tre* al *sette* per la partizione */* scegliendo *sda2* e *sdb2*.
9. Una volta completato tutto, selezionare *Terminare*.

4.1.3. Formattare

Dovrebbe essere visibile un elenco di dischi fissi e dispositivi RAID. Il passo successivo consiste nel formattare e impostare il punto di mount per i dispositivi RAID: tali dispositivi sono da considerare come dei normali dischi locali.

1. Select *"#1"* under the *"RAID1 device #0"* partition.
2. Scegliere *Usato come:*, quindi *area di swap* e infine *Preparazione di questa partizione completata*.
3. Next, select *"#1"* under the *"RAID1 device #1"* partition.
4. Choose *"Use as:"*. Then select *"Ext4 journaling file system"*.
5. Selezionare *Punto di mount* e scegliere */ - il file system root*. Modificare se necessario le altre opzioni e selezionare *Preparazione di questa partizione completata*.
6. Selezionare *Terminare il partizionamento e scrivere i cambiamenti sul disco*.

Se è stato scelto di posizionare la partizione di root nell'array RAID, il programma di installazione chiederà se avviare il sistema in modalità *degraded*. Per maggiori informazioni, consultare *Sezione 4.1.4, «RAID degraded» [11]*.

Il processo di installazione continuerà normalmente.

4.1.4. RAID degraded

Durante l'arco di vita di un computer si potrebbero verificare dei danni ai dischi. Quando si verifica un'eventualità come questa, usando il RAID software, il sistema operativo abilita la modalità *degraded* per l'array.

Se l'array è degradato ("degraded") a causa di dati rovinati, il sistema operativo, in modo predefinito, si avvierà in *initramfs* dopo 30 secondi. Una volta avviato, è possibile, entro 15 secondi, continuare il normale avvio o tentare un ripristino manuale. L'avvio in initramfs potrebbe non essere consigliato, soprattutto se si opera sul computer da remoto. Avviare il sistema in un array "degraded" può essere svolto in diversi modi:

- L'utilità dpkg-reconfigure può essere usata per configurare il comportamento predefinito e durante l'elaborazione verranno poste delle domande relative a impostazioni addizionali per l'array come monitoraggio, avvisi via email, ecc... Per riconfigurare mdadm, digitare il seguente comando:

  ```
  sudo dpkg-reconfigure mdadm
  ```

- Il processo **dpkg-reconfigure mdadm** modificherà il file di configurazione `/etc/initramfs-tools/conf.d/mdadm`. Tale file presenta il vantaggio di pre-configurare il comportamento del sistema e può essere modificato a mano:

  ```
  BOOT_DEGRADED=true
  ```

  

  Il file di configurazione può essere scavalcato utilizzando un argomento per il kernel.

- È possibile avviare il sistema in un array "degraded" utilizzando anche un argomento per il kernel:
  - When the server is booting press **Shift** to open the Grub menu.
  - Press **e** to edit your kernel command options.
  - Press the **down** arrow to highlight the kernel line.
  - Aggiungere *bootdegraded=true* alla fine della riga.
  - Premere **Ctrl**+**x** per avviare il sistema.

Una volta avviato il sistema, è possibile riparare l'array (consultare *Sezione 4.1.5, «Manutenzione RAID» [12]*) o copiare i dati importanti in un altro computer.

4.1.5. Manutenzione RAID

L'utilità mdadm può essere usata per visualizzare lo stato dell'array, aggiungere un disco all'array, rimuovere dischi, ecc...

- Per visualizzare lo stato di un array, da un terminale digitare:

  ```
  sudo mdadm -D /dev/md0
  ```

  L'opzione *-D* indica a mdadm di stampare informazioni *dettagliate* riguardo il device `/dev/md0`. Sostituire `/dev/md0` con il device RAID appropriato.

- Per visualizzare lo stato di un disco in un array:

  ```
  sudo mdadm -E /dev/sda1
  ```

L'output è molto simile al comando **mdadm -D**, regolare `/dev/sda1` per ogni disco.

- Se un disco si rompe e deve essere rimosso da un array:

```
sudo mdadm --remove /dev/md0 /dev/sda1
```

Modificare `/dev/md0` e `/dev/sda1` con il device e il disco RAID appropriati.

- Per aggiungere un nuovo disco:

```
sudo mdadm --add /dev/md0 /dev/sda1
```

Qualche volta può succedere che un disco imposti il suo stato come *difettoso* ("faulty"), anche se non presenta alcun malfunzionamento hardware. Può essere utile in questi casi rimuovere e aggiungere il disco all'array: verrà così nuovamente sincronizzato con l'array. Se il disco non riesce a sincronizzarsi con l'array, può indicare che il dispositivo sia effettivamente difettoso.

Il file `/proc/mdstat` contiene anche informazioni utili riguardo i device RAID del sistema:

```
cat /proc/mdstat
Personalities : [linear] [multipath] [raid0] [raid1] [raid6] [raid5] [raid4] [raid10]
md0 : active raid1 sda1[0] sdb1[1]
      10016384 blocks [2/2] [UU]

unused devices: <none>
```

Il seguente comando è utile per controllare lo stato di un drive di sincronizzazione:

```
watch -n1 cat /proc/mdstat
```

Premere *Ctrl+C* per fermare il comando watch.

Se è necessario sostituire un disco difettoso, una volta sostituito e sincronizzato, è necessario reinstallare grub. Per installare grub nel nuovo disco, procedere come segue:

```
sudo grub-install /dev/md0
```

Sostituire `/dev/md0` con il nome dell'array appropriato.

4.1.6. Risorse

L'argomento degli array RAID è molto complesso e vasto poiché sono disponibili molti modi diversi di configurare un array RAID. Per maggiori informazioni, consultare i seguenti collegamenti:

- *Documentazione online riguardo RAID*[6].
- *Software RAID HOWTO*[7]
- *Managing RAID on Linux*[8]

4.2. Logical Volume Manager (LVM)

Logical Volume Manger, o *LVM*, consente agli amministratori di creare volumi *logici* da uno o più dischi fissi. I volumi LVM possono essere creati sia sulle partizioni RAID software sia sulle partizioni normali presenti su un singolo disco. I volumi possono essere estesi, garantendo un'alta flessibilità al sistema nel caso cambino le necessità.

4.2.1. Panoramica

Purtroppo, la potenza e la flessibilità di LVM, comportano maggiori complicazioni. Prima di tutto è quindi necessario introdurre la terminologia adatta.

- *Gruppo di volumi (VG)*: contiene uno o più volumi logici (LV).
- *Volume logico (LV)*: è simile a una partizione in un sistema non LVM. Più volumi fisici (PV) possono creare un LV sul quale è presente il vero file system (ext3, Xfs, Jfs, ecc...).
- *Volume fisico(PV)*: il disco rigido o la partizione RAID software. Il gruppo di volumi può essere esteso aggiungendo più PV.

4.2.2. Installazione

Come esempio, in questa sezione, viene descritto come installare Ubuntu Server Edition con `/srv` montato come volume LVM. Durante l'installazione un solo volume fisico (PV) farà parte del gruppo di volumi (VG). Un altro PV verrà aggiunto dopo l'installazione come dimostrazione delle funzionalità di estensione di un VG.

Sono disponibili diverse opzioni per l'installazione LVM, *Guidato - usare l'intero disco e impostare LVM* consente di assegnare una parte dello spazio disponibile a LVM, *Guidato - usare l'intero disco e impostare LVM cifrato* o *manuale*. Attualmente l'unico metodo per configurare un sistema affinché utilizzi sia partizioni LVM che normali durante l'installazione è quello manuale.

1. Seguire i passi dell'installazione fino a giungere a *Partizionamento dei dischi*, quindi:
2. Alla finestra *Partizionamento dei dischi* scegliere *Manuale*.
3. Selezionare il disco fisso e nella schermata successiva scegliere confermare *Creare una nuova tabella delle partizioni sul dispositivo*.
4. Creare le partizioni */boot*, *swap* e */* con il file system di propria scelta.
5. Per la partizione */srv* LVM, creare una nuova partizione *Logica* e modificare *Usato come* in *volume fisico per LVM*, quindi selezionare *Preparazione di questa partizione completata*.
6. Selezionare *Configurare il Logical Volume Manager* in alto e scegliere *Sì* per scrivere le modifiche sul disco.
7. Per il *Passo di configurazione di LVM* nella schermata successiva, scegliere *Creare gruppi di volumi*. Inserire un nome per il VG come *vg01* o qualche cosa più descrittivo. Fatto ciò, selezionare la partizione configurata per LVM e scegliere *Continua*.
8. Sempre nella schermata *Passo di configurazione di LVM*, selezionare *Creare volume logico*, selezionare il gruppo di volumi appena creato e inserire un nome per il nuovo LV, per esempio

srv dato che verrà utilizzato come punto di mount per quella partizione. Scegliere la dimensione, che in questo caso può essere l'intera partizione dato che è possibile estenderla o ridurla, scegliere *Termina* per tornare alla schermata *Partizionamento dei dischi*.

9. Ora aggiungere il file system al nuovo LVM. Selezionare la partizione *LVM VG vg01, LV srv*, o in base al nome inserito, e scegliere *Usato come*. Impostare un file system selezionando */srv* come punto di mount e una volta completato, selezionare *Preparazione di questa partizione completata*.
10. Infine, selezionare *Terminare il partizionamento e scrivere i cambiamenti sul disco*, confermare le modifiche e continuare l'installazione.

Per visualizzare informazioni riguardo LVM sono disponibili diverse utilità:

- *vgdisplay*: visualizza informazioni riguardo i gruppi di volumi.
- *lvdisplay*: visualizza informazioni riguardo i volumi logici.
- *pvdisplay*: visualizza informazioni riguardo i volumi fisici.

4.2.3. Estendere i gruppi di volumi

Utilizzando l'esempio di *srv* come volume LVM, in questa sezione viene indicato come aggiungere un secondo disco fisso, come creare un volume fisico (PV), come aggiungerlo al gruppo di volumi (VG), come estendere il volume logico `srv` e infine come estendere il file system. In questo esempio viene aggiunto un secondo disco fisso chiamato `/dev/sdb`. Attenzione: assicurarsi di non avere già un device `/dev/sdb` prima di eseguire i comandi qui presentati, si potrebbero perdere i dati se eseguiti su un disco non vuoto. In questo esempio il disco fisso viene usato interamente come volume fisico (è possibile creare partizioni e usarle come diversi volumi fisici).

1. Creare il volume fisico. In un terminale digitare:

   ```
   sudo pvcreate /dev/sdb
   ```

2. Estendere il gruppo di volumi (VG):

   ```
   sudo vgextend vg01 /dev/sdb
   ```

3. Usare vgdisplay per trovare gli extent fisici (PE) liberi (PE/dimensione = dimensione da allocare). In questo esempio viene considerata una dimensione di 511 PE (equivalenti a 2GB con una dimensione di PE di 4MB) e viene utilizzato tutto lo spazio libero. Utilizzare i PE in base alle proprie disponibilità.

   Il volume logico (LV) può essere esteso in diversi modi. In questo esempio viene considerato il caso di utilizzo del PE per estendere il LV:

   ```
   sudo lvextend /dev/vg01/srv -l +511
   ```

   L'opzione *-l* consente di estendere il LV attraverso l'uso di PE. L'opzione *-L* invece, consente di estendere il LV utilizzando megabyte, gigabyte, terabyte, ecc...

4. Even though you are supposed to be able to *expand* an ext3 or ext4 filesystem without unmounting it first, it may be a good practice to unmount it anyway and check the filesystem, so that you don't mess up the day you want to reduce a logical volume (in that case unmounting first is compulsory).

   I seguenti comandi sono pensati per un file system *ext3* o *ext4*. Se si sta utilizzando un altro file system potrebbero essere disponibili altri programmi.

   ```
   sudo umount /srv
   sudo e2fsck -f /dev/vg01/srv
   ```

   L'opzione *-f* di e2fsck forza il controllo anche se il file system sembra non avere problemi.
5. Infine, ridimensionare il file system:

   ```
   sudo resize2fs /dev/vg01/srv
   ```
6. Montare la partizione e controllarne la dimensione.

   ```
   mount /dev/vg01/srv /srv && df -h /srv
   ```

4.2.4. Risorse

- Consultare la *documentazione online riguardo LVM*[9].
- Per maggiori informazioni, consultare *LVM HOWTO*[10].
- Un ottimo articolo presente su linuxdevcenter.com è *Managing Disk Space with LVM*[11].
- Per maggiori informazioni riguardo fdisk, consultarne *la pagina di manuale*[12].

Capitolo 3. Gestione dei pacchetti

Ubuntu dispone di un completo servizio di gestione dei pacchetti per l'installazione, l'aggiornamento, la configurazione e la rimozione del software. Oltre a fornire accesso a più di 24000 pacchetti software per il proprio computer, il sistema di gestione dei pacchetti è in grado di gestire le dipendenze e di verificare l'esistenza di aggiornamenti.

Per l'interazione con il sistema di gestione dei pacchetti di Ubuntu sono disponibili diversi strumenti, a partire da semplici utilità a riga di comando che possono essere usate con facilità da amministratori di sistema per attività automatizzate, fino a interfacce grafiche semplici da usare per chi si è avvicinato da poco a Ubuntu.

1. Introduzione

Il sistema di gestione dei pacchetti di Ubuntu è derivato dallo stesso sistema usato dalla distribuzione Debian GNU/Linux. I file di pacchetto contengono tutti i file, i meta-dati e le istruzioni necessari per implementare sui sistemi Ubuntu una particolare funzionalità o un'applicazione software.

Di solito, i file dei pacchetti Debian presentano l'estensione «.deb» e risiedono nei *repository*, ossia delle collezioni di pacchetti memorizzate su diversi supporti, come un disco CD-ROM o in rete. I pacchetti sono normalmente in formato binario precompilato: per questo l'installazione è veloce e non richiede la compilazione del software.

Molti pacchetti sfruttano il concetto delle *dipendenze*: dei pacchetti aggiuntivi richiesti dal pacchetto che si sta installando per poter funzionare correttamente. Per esempio, il pacchetto di sintesi vocale Festival dipende dal pacchetto libasound2 che fornisce la libreria audio ALSA necessario per la riproduzione audio. Affinché Festival possa funzionare, è necessario installare tutte le sue dipendenze. Il software di gestione dei pacchetti svolge questa operazione automaticamente.

2. dpkg

dpkg is a package manager for *Debian* based systems. It can install, remove, and build packages, but unlike other package management system's, it can not automatically download and install packages or their dependencies. This section covers using dpkg to manage locally installed packages:

- Per elencare i pacchetti installati nel sistema, da un terminale digitare:

```
dpkg -l
```

- In base a quanti pacchetto sono installati nel sistema, questo comando può generare molto output. È comunque possibile passare l'output attraverso una pipe all'applicazione grep per vedere se un particolare pacchetto è installato o meno:

```
dpkg -l | grep apache2
```

  Sostituire *apache2* con il nome di un qualsiasi altro pacchetto, parte del nome o qualsiasi altra espressione regolare.

- Per elencare i file installati da un pacchetto, in questo caso ufw, digitare:

```
dpkg -L ufw
```

- Se non si è sicuri di quale pacchetto abbia installato un file, usare il comando dpkg -S. Per esempio:

```
dpkg -S /etc/host.conf
base-files: /etc/host.conf
```

  L'output mostra che `/etc/host.conf` appartiene al pacchetto base-files.

  Molti file sono generati automaticamente durante il processo di installazione del pacchetto e benché siano nel file system, il comando **dpkg -S** potrebbe non sapere a quale pacchetto appartengono.

- Per installare un file *.deb* locale, digitare:

```
sudo dpkg -i zip_2.32-1_i386.deb
```

  Modificare `zip_2.32-1_i386.deb` con il nome del file .deb da installare.

- Per disinstallare un pacchetto:

```
sudo dpkg -r zip
```

  Disinstallare i pacchetti usando dpkg, nella maggior parte dei casi, *non* è raccomandato. È meglio usare un gestore di pacchetti in grado di gestire le dipendenze per assicurarsi che il sistema permanga sempre in una stato consistente. Per esempio, usando **dpkg -**

r, è possibile rimuovere il pacchetto zip, ma qualsiasi pacchetto che vi dipende resterà installato e potrebbe non funzionare correttamente.

Per le ulteriori opzioni di dpkg, consultare la pagina di manuale: **man dpkg**.

3. Apt-Get

Il comando apt-get è un potente strumento a riga di comando usato per operare con l'APT (*Advanced Packaging Tool*) di Ubuntu al fine di eseguire operazioni come l'installazione di nuovi pacchetti software, l'aggiornamento dei pacchetti software esistenti, l'aggiornamento dell'indice dell'elenco dei pacchetti e persino l'avanzamento di versione dell'intero sistema Ubuntu.

Essendo un semplice strumento da riga di comando, apt-get presenta agli amministratori di sistema numerosi vantaggi rispetto ad altri strumenti di gestione dei pacchetti disponibili in Ubuntu. Alcuni di questi vantaggi sono la facilità d'utilizzo mediante connessioni via terminale (SSH) e la possibilità di essere usato in script di amministrazione del sistema, magari automatizzati attraverso l'utilità di pianificazione cron.

Alcuni esempi di utilizzo tipico dell'utilità apt-get:

- **Installare un pacchetto**: l'installazione di pacchetti usando lo strumento apt-get è molto semplice. Per esempio, per installare lo scanner di rete *nmap*, digitare il seguente comando:

  ```
  sudo apt-get install nmap
  ```

- **Rimuovere un pacchetto**: la rimozione di uno o più pacchetti è altrettanto semplice e immediata. Per rimuovere il pacchetto nmap installato nell'esempio precedente, digitare il seguente comando:

  ```
  sudo apt-get remove nmap
  ```

  

  Pacchetti multipli: è possibile specificare più di un pacchetto da installare o rimuovere, separati da spazi.

  Aggiungere l'opzione *--purge* ad **apt-get remove** fa in modo che vengano rimossi anche i file di configurazione del paccahtto. Usare questa opzione con cautela se non è ciò che si vuole.

- **Aggiornare l'indice dei pacchetti**: l'indice dei pacchetti di APT è essenzialmente un database dei pacchetti disponibili dai repository definiti nel file `/etc/apt/sources.list`. Per aggiornare l'elenco locale dei pacchetti con i cambiamenti apportati di recente nei repository, digitare il comando:

  ```
  sudo apt-get update
  ```

- **Aggiornare i pacchetti**: versioni aggiornate dei pacchetti installati possono essere disponibili attraverso i repository dei pacchetti (per esempio per aggiornamenti di sicurezza). Per aggiornare il proprio sistema è necessario, prima di tutto, aggiornare l'indice dei pacchetti come spiegato sopra, quindi digitare:

  ```
  sudo apt-get upgrade
  ```

  Per informazioni sull'avanzamento a un nuovo rilascio di Ubuntu, consultare la *Sezione 3, «Avanzamento di versione» [9]*.

Le azioni del comando apt-get, come l'installazione o la rimozione di pacchetti, vengono registrate nel file di registro /var/log/dpkg.log.

Per maggiori informazioni sull'uso di APT, consultare il *Manuale utente di Debian APT*[1], oppure digitare:

```
apt-get help
```

[1] http://www.debian.org/doc/user-manuals#apt-howto

4. Aptitude

Aptitude è un'interfaccia testuale basata su menù per il sistema APT (*Advanced Packaging Tool*). Molte delle tipiche funzioni di gestione dei pacchetti, come l'installazione, la rimozione e l'aggiornamento, possono essere effettuate in Aptitude con dei comandi mappati su un solo tasto, solitamente delle lettere minuscole.

Aptitude è indicato in un ambiente non grafico per assicurare il corretto funzionamento dei comandi. È possibile avviare Aptitude eseguendo il seguente comando al prompt del terminale:

```
sudo aptitude
```

All'avvio di Aptitude, viene mostrata una barra dei menù nella parte superiore dello schermo e due riquadri sotto tale barra. Il riquadro superiore contiene delle categorie di pacchetto, come *Pacchetti nuovi* e *Pacchetti non installati*. Il riquadro inferiore contiene le informazioni relative ai pacchetti e alle categorie di pacchetto.

Usare Aptitude per la gestione dei pacchetti è relativamente chiaro e l'interfaccia utente rende le operazioni comuni semplici da eseguire. Di seguito vengono presentati alcuni esempi di funzioni comuni della gestione dei pacchetti con Aptitude:

- **Installare pacchetti**: per installare un pacchetto, localizzare il pacchetto attraverso la categoria di pacchetto "Pacchetti non installati", usando i tasti freccia sulla tastiera e il tasto **Invio**, in modo da evidenziare il pacchetto da installare. Dopo aver evidenziato il pacchetto da installare, premere il tasto **+**: la voce relativa al pacchetto assume una colorazione *verde*, per indicare che è stato contrassegnato per l'installazione. Premere quindi il tasto **g** per ricapitolare le operazioni su pacchetti. Premendo nuovamente **g**, viene richiesto di acquisire i privilegi di amministrazione per completare l'installazione. Premere quindi **Invio** per mostrare un prompt «Password:». Inserire la propria password utente per diventare root. Infine premendo **g** ancora una volta viene richiesto se scaricare il pacchetto. Premere **Invio** al prompt *Continua*: viene avviato lo scaricamento e l'installazione del pacchetto.
- **Rimuovere pacchetti**: per rimuovere un pacchetto, localizzare il pacchetto attraverso la categoria di pacchetto "Pacchetti installati", usando i tasti freccia sulla tastiera e il tasto **Invio**, in modo da evidenziare il pacchetto da rimuovere. Dopo aver evidenziato il pacchetto da rimuovere, premere il tasto **-**: la voce relativa al pacchetto assume una colorazione *rosa*, per indicare che è stato contrassegnato per la rimozione. Premere quindi il tasto **g** per ricapitolare le operazioni sui pacchetti. Premendo nuovamente **g**, viene richiesto di acquisire i privilegi di amministrazione per completare l'installazione. Premere quindi **Invio** per mostrare un prompt "Password:". Inserire la propria password utente per diventare root. Infine, premendo **g** ancora una volta, viene richiesto se scaricare il pacchetto. Premere **Invio** al prompt *Continua*: viene avviata la rimozione del pacchetto.
- **Aggiornare l'indice dei pacchetti**: per aggiornare l'indice dei pacchetti è sufficiente premere il tasto **u** e verrà richiesto di diventare amministratori per completare l'aggiornamento. Premendo **Invio** viene presentata la richiesta della password. Inserire la password del proprio utente per

assumere i privilegi di amministratore. L'aggiornamento dell'indice dei pacchetti verrà avviato. Premere **Invio** al prompt OK quando appare il dialogo per scaricare il necessario al completamento del processo.

- **Aggiornare i pacchetti**: per aggiornare i pacchetti, eseguire l'aggiornamento dell'indice dei pacchetti come spiegato precedentemente, quindi premere il tasto **U** per selezionare tutti i pacchetti con aggiornamenti. Adesso premere **g** per avere un sommario delle azioni possibili sui pacchetti. Premere nuovamente **g** per assumere i privilegi di amministratore per completare l'installazione. Premere **Invio** e inserire la password. Infine premere **g** ancora una volta per la richiesta di scaricare i pacchetti. Premere **Invio** al prompt *Continua* e l'aggiornamento dei pacchetti inizierà.

La prima colonna delle informazioni mostrate nell'elenco dei pacchetti nel riquadro superiore, indica l'attuale stato del pacchetto, utilizzando le seguenti chiavi per descrivere lo stato del pacchetto:

- **i**: pacchetto installato
- **c**: pacchetto non installato, ma nel sistema è rimasta traccia della configurazione del pacchetto
- **p**: rimosso completamente dal sistema
- **v**: pacchetto virtuale
- **B**: pacchetto non integro
- **u**: file decompressi, ma pacchetto non ancora configurato
- **C**: configurato in parte. La configurazione è fallita e necessita di essere corretta
- **H**: installato parzialmente. La rimozione è fallita e necessita di essere sistemata

Per chiudere Aptitude, è sufficiente premere il tasto **q** e confermare l'uscita. Sono disponibili molte altre funzioni dal menù di Aptitude, premendo il tasto **F10**.

5. Aggiornamenti automatici

Il pacchetto unattended-upgrades può essere usato per installare automaticamente gli aggiornamenti e può essere configurato per aggiornare tutti i pacchetti o installare solamente gli aggiornamenti di sicurezza. Per prima cosa, installare il pacchetto digitando:

```
sudo apt-get install unattended-upgrades
```

Per configurare unattended-upgrades, aprire il file `/etc/apt/apt.conf.d/50unattended-upgrades` e modificare quanto segue secondo le proprie esigenze:

```
Unattended-Upgrade::Allowed-Origins {
        "Ubuntu maverick-security";
//      "Ubuntu maverick-updates";
};
```

Alcuni pacchetti possono essere inseriti nella *blacklist* per non aggiornarli mai. Per inserire un pacchetto nella blacklist, aggiungerlo all'elenco:

```
Unattended-Upgrade::Package-Blacklist {
//      "vim";
//      "libc6";
//      "libc6-dev";
//      "libc6-i686";
};
```

I doppi slash («//») servono come commento; tutto quello che segue "//" non verrà valutato.

To enable automatic updates, edit `/etc/apt/apt.conf.d/10periodic` and set the appropriate apt configuration options:

```
APT::Periodic::Update-Package-Lists "1";
APT::Periodic::Download-Upgradeable-Packages "1";
APT::Periodic::AutocleanInterval "7";
APT::Periodic::Unattended-Upgrade "1";
```

The above configuration updates the package list, downloads, and installs available upgrades every day. The local download archive is cleaned every week.

You can read more about apt Periodic configuration options in the `/etc/cron.daily/apt` script header.

I risultati di unattended-upgrades vengono registrati in `/var/log/unattended-upgrades`.

5.1. Notifiche

Impostando *Unattended-Upgrade::Mail* nel file `/etc/apt/apt.conf.d/50unattended-upgrades`, si abilita unattended-upgrades all'invio di email all'amministratore indicando i pacchetti da aggiornare o con problemi.

Another useful package is apticron. apticron will configure a cron job to email an administrator information about any packages on the system that have updates available, as well as a summary of changes in each package.

Per installare apticron, digitare:

```
sudo apt-get install apticron
```

Una volta installato, aprire il file `/etc/apticron/apticron.conf` e impostare l'indirizzo email e altre opzioni:

```
EMAIL="root@example.it"
```

6. Configurazione

La configurazione dei repository del sistema APT (*Advanced Packaging Tool*) è memorizzata nel file di configurazione /etc/apt/sources.list. Un esempio di questo file, con le istruzioni su come aggiungere e rimuovere repository, è qui referenziato.

Questo[2] è un semplice esempio di un tipico file `/etc/apt/sources.list`.

È possibile modificare il file per abilitare o disabilitare i repository. Per esempio, per disabilitare la necessità di inserire il CD-ROM di Ubuntu ogni volta che viene effettuata un'operazione sui pacchetti, è sufficiente commenentare la riga relativa al CD-ROM, che si trova all'inizio del file:

```
# no more prompting for CD-ROM please
# deb cdrom:[Ubuntu 10.10_Maverick_Meerkat - Release i386 (20070419.1)]/ maverick main restricted
```

6.1. Repository aggiuntivi

In aggiunta ai repository dei pacchetti disponibili per Ubuntu e supportati ufficialmente, esistono repository aggiuntivi mantenuti dalla comunità che aggiungono migliaia di pacchetti che possono essere installati. Due dei più popolari sono i repository *universe* e *multiverse*. Questi repository non sono ufficialmente supportati da Ubuntu, ma proprio perché sono mantenuti dalla comunità di norma forniscono pacchetti che possono essere installati sul proprio computer senza rischi.

I pacchetti nel repository *multiverse* presentano spesso problemi di licenza che non gli permettono di essere distribuiti con un sistema operativo gratuito e potrebbero essere illegali in alcuni paesi.

Né il repository *universe* né quello *multiverse* contengono pacchetti supportati ufficialmente. In particolare, potrebbero non esserci aggiornamenti di sicurezza per tali pacchetti.

Sono disponibili molte altre sorgenti di pacchetti, alcune delle quali offrono solo un pacchetto, come nel caso di sorgenti di pacchetto fornite dallo sviluppatore di una singola applicazione. L'utilizzo di sorgenti di pacchetto non standard è rischioso, pertanto è necessario prestare la massima attenzione. È opportuno controllare la sorgente e i pacchetti in modo accurato prima di effettuare una qualsiasi installazione, poiché alcune sorgenti di pacchetto, e i rispettivi pacchetti, potrebbero rendere il sistema instabile e non funzionante sotto certi aspetti.

I repository *universe* e *multiverse*, in modo predefinito, sono abilitati, ma se si desidera disabilitarli è possibile modificare il file `/etc/apt/sources.list` e commentare le seguenti righe:

```
deb http://archive.ubuntu.com/ubuntu maverick universe multiverse
deb-src http://archive.ubuntu.com/ubuntu maverick universe multiverse
```

[2] ../sample/sources.list

```
deb http://us.archive.ubuntu.com/ubuntu/ maverick universe
deb-src http://us.archive.ubuntu.com/ubuntu/ maverick universe
deb http://us.archive.ubuntu.com/ubuntu/ maverick-updates universe
deb-src http://us.archive.ubuntu.com/ubuntu/ maverick-updates universe

deb http://us.archive.ubuntu.com/ubuntu/ maverick multiverse
deb-src http://us.archive.ubuntu.com/ubuntu/ maverick multiverse
deb http://us.archive.ubuntu.com/ubuntu/ maverick-updates multiverse
deb-src http://us.archive.ubuntu.com/ubuntu/ maverick-updates multiverse

deb http://security.ubuntu.com/ubuntu maverick-security universe
deb-src http://security.ubuntu.com/ubuntu maverick-security universe
deb http://security.ubuntu.com/ubuntu maverick-security multiverse
deb-src http://security.ubuntu.com/ubuntu maverick-security multiverse
```

7. Riferimenti

La maggior parte di quanto discusso in questo capitolo è disponibile nella pagine man, molte delle quali sono reperibili anche in rete.

- The *InstallingSoftware*[3] Ubuntu wiki page has more information.
- For more dpkg details see the *dpkg man page*[4].
- The *APT HOWTO*[5] and *apt-get man page*[6] contain useful information regarding apt-get usage.
- See the *aptitude man page*[7] for more aptitude options.
- La pagina *riguardo i repository*[8] della documentazione italiana, contiene maggiori informazioni su come aggiungere repository.

Capitolo 4. Rete

Le reti consistono in due o più dispositivi, come computer, stampanti e altri equipaggiamenti correlati, connessi tramite un cavo fisico oppure tramite collegamenti senza fili, con lo scopo di condividere e distribuire informazioni tra di loro.

Questa sezione fornisce informazioni generali e specifiche sulle reti (creare, modificare e gestire reti), compresa una panoramica sui concetti delle reti e discussioni dettagliate dei più comuni protocolli di rete.

1. Configurare la rete

Ubuntu è corredato da una serie di utilità grafiche per la configurazione dei dispositivi di rete. Questa sezione è diretta agli amministratori di server e si focalizza sulla gestione della rete da riga di comando.

1.1. Ethernet Interfaces

Ethernet interfaces are identified by the system using the naming convention of *ethX*, where *X* represents a numeric value. The first Ethernet interface is typically identified as *eth0*, the second as *eth1*, and all others should move up in numerical order.

1.1.1. Identify Ethernet Interfaces

To quickly identify all available Ethernet interfaces, you can use the ifconfig command as shown below.

```
ifconfig -a | grep eth
eth0      Link encap:Ethernet  HWaddr 00:15:c5:4a:16:5a
```

Another application that can help identify all network interfaces available to your system is the lshw command. In the example below, lshw shows a single Ethernet interface with the logical name of *eth0* along with bus information, driver details and all supported capabilities.

```
sudo lshw -class network
  *-network
       description: Ethernet interface
       product: BCM4401-B0 100Base-TX
       vendor: Broadcom Corporation
       physical id: 0
       bus info: pci@0000:03:00.0
       logical name: eth0
       version: 02
       serial: 00:15:c5:4a:16:5a
       size: 10MB/s
       capacity: 100MB/s
       width: 32 bits
       clock: 33MHz
       capabilities: (snipped for brevity)
       configuration: (snipped for brevity)
       resources: irq:17 memory:ef9fe000-ef9fffff
```

1.1.2. Ethernet Interface Logical Names

Interface logical names are configured in the file `/etc/udev/rules.d/70-persistent-net.rules`. If you would like control which interface receives a particular logical name, find the line matching the interfaces physical MAC address and modify the value of *NAME=ethX* to the desired logical name. Reboot the system to commit your changes.

```
SUBSYSTEM=="net", ACTION=="add", DRIVERS=="?*", ATTR{address}=="00:15:c5:4a:16:5a", ATTR{dev_id}=="
SUBSYSTEM=="net", ACTION=="add", DRIVERS=="?*", ATTR{address}=="00:15:c5:4a:16:5b", ATTR{dev_id}=="
```

1.1.3. Ethernet Interface Settings

ethtool is a program that displays and changes Ethernet card settings such as auto-negotiation, port speed, duplex mode, and Wake-on-LAN. It is not installed by default, but is available for installation in the repositories.

```
sudo apt-get install ethtool
```

The following is an example of how to view supported features and configured settings of an Ethernet interface.

```
sudo ethtool eth0
Settings for eth0:
        Supported ports: [ TP ]
        Supported link modes:   10baseT/Half 10baseT/Full
                                100baseT/Half 100baseT/Full
                                1000baseT/Half 1000baseT/Full
        Supports auto-negotiation: Yes
        Advertised link modes:  10baseT/Half 10baseT/Full
                                100baseT/Half 100baseT/Full
                                1000baseT/Half 1000baseT/Full
        Advertised auto-negotiation: Yes
        Speed: 1000Mb/s
        Duplex: Full
        Port: Twisted Pair
        PHYAD: 1
        Transceiver: internal
        Auto-negotiation: on
        Supports Wake-on: g
        Wake-on: d
        Current message level: 0x000000ff (255)
        Link detected: yes
```

Changes made with the ethtool command are temporary and will be lost after a reboot. If you would like to retain settings, simply add the desired ethtool command to a *pre-up* statement in the interface configuration file `/etc/network/interfaces`.

The following is an example of how the interface identified as *eth0* could be permanently configured with a port speed of 1000Mb/s running in full duplex mode.

```
auto eth0
iface eth0 inet static
pre-up /usr/sbin/ethtool -s eth0 speed 1000 duplex full
```

Although the example above shows the interface configured to use the *static* method, it actually works with other methods as well, such as DHCP. The example is meant to

demonstrate only proper placement of the *pre-up* statement in relation to the rest of the interface configuration.

1.2. IP Addressing

The following section describes the process of configuring your systems IP address and default gateway needed for communicating on a local area network and the Internet.

1.2.1. Temporary IP Address Assignment

For temporary network configurations, you can use standard commands such as ip, ifconfig and route, which are also found on most other GNU/Linux operating systems. These commands allow you to configure settings which take effect immediately, however they are not persistent and will be lost after a reboot.

To temporarily configure an IP address, you can use the ifconfig command in the following manner. Just modify the IP address and subnet mask to match your network requirements.

```
sudo ifconfig eth0 10.0.0.100 netmask 255.255.255.0
```

To verify the IP address configuration of eth0, you can use the ifconfig command in the following manner.

```
ifconfig eth0
eth0      Link encap:Ethernet  HWaddr 00:15:c5:4a:16:5a
          inet addr:10.0.0.100  Bcast:10.0.0.255  Mask:255.255.255.0
          inet6 addr: fe80::215:c5ff:fe4a:165a/64 Scope:Link
          UP BROADCAST RUNNING MULTICAST  MTU:1500  Metric:1
          RX packets:466475604 errors:0 dropped:0 overruns:0 frame:0
          TX packets:403172654 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:1000
          RX bytes:2574778386 (2.5 GB)  TX bytes:1618367329 (1.6 GB)
          Interrupt:16
```

To configure a default gateway, you can use the route command in the following manner. Modify the default gateway address to match your network requirements.

```
sudo route add default gw 10.0.0.1 eth0
```

To verify your default gateway configuration, you can use the route command in the following manner.

```
route -n
Kernel IP routing table
Destination     Gateway         Genmask         Flags Metric Ref    Use Iface
10.0.0.0        0.0.0.0         255.255.255.0   U     1      0        0 eth0
0.0.0.0         10.0.0.1        0.0.0.0         UG    0      0        0 eth0
```

If you require DNS for your temporary network configuration, you can add DNS server IP addresses in the file `/etc/resolv.conf`. The example below shows how to enter two DNS servers to `/etc/resolv.conf`, which should be changed to servers appropriate for your network. A more lengthy description of DNS client configuration is in a following section.

```
nameserver 8.8.8.8
nameserver 8.8.4.4
```

If you no longer need this configuration and wish to purge all IP configuration from an interface, you can use the ip command with the flush option as shown below.

```
ip addr flush eth0
```

Flushing the IP configuration using the ip command does not clear the contents of `/etc/resolv.conf`. You must remove or modify those entries manually.

1.2.2. Dynamic IP Address Assignment (DHCP Client)

To configure your server to use DHCP for dynamic address assignment, add the *dhcp* method to the inet address family statement for the appropriate interface in the file `/etc/network/interfaces`. The example below assumes you are configuring your first Ethernet interface identified as *eth0*.

```
auto eth0
iface eth0 inet dhcp
```

By adding an interface configuration as shown above, you can manually enable the interface through the ifup command which initiates the DHCP process via dhclient.

```
sudo ifup eth0
```

To manually disable the interface, you can use the ifdown command, which in turn will initiate the DHCP release process and shut down the interface.

```
sudo ifdown eth0
```

1.2.3. Static IP Address Assignment

To configure your system to use a static IP address assignment, add the *static* method to the inet address family statement for the appropriate interface in the file `/etc/network/interfaces`. The example below assumes you are configuring your first Ethernet interface identified as *eth0*. Change the *address*, *netmask*, and *gateway* values to meet the requirements of your network.

```
auto eth0
iface eth0 inet static
address 10.0.0.100
netmask 255.255.255.0
```

```
gateway 10.0.0.1
```

By adding an interface configuration as shown above, you can manually enable the interface through the ifup command.

```
sudo ifup eth0
```

To manually disable the interface, you can use the ifdown command.

```
sudo ifdown eth0
```

1.2.4. Loopback Interface

The loopback interface is identified by the system as *lo* and has a default IP address of 127.0.0.1. It can be viewed using the ifconfig command.

```
ifconfig lo
lo        Link encap:Local Loopback
          inet addr:127.0.0.1  Mask:255.0.0.0
          inet6 addr: ::1/128 Scope:Host
          UP LOOPBACK RUNNING  MTU:16436  Metric:1
          RX packets:2718 errors:0 dropped:0 overruns:0 frame:0
          TX packets:2718 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:0
          RX bytes:183308 (183.3 KB)  TX bytes:183308 (183.3 KB)
```

By default, there should be two lines in `/etc/network/interfaces` responsible for automatically configuring your loopback interface. It is recommended that you keep the default settings unless you have a specific purpose for changing them. An example of the two default lines are shown below.

```
auto lo
iface lo inet loopback
```

1.3. Name Resolution

Name resolution as it relates to IP networking is the process of mapping IP addresses to hostnames, making it easier to identify resources on a network. The following section will explain how to properly configure your system for name resolution using DNS and static hostname records.

1.3.1. DNS Client Configuration

To configure your system to use DNS for name resolution, add the IP addresses of the DNS servers that are appropriate for your network in the file `/etc/resolv.conf`. You can also add an optional DNS suffix search-lists to match your network domain names.

Below is an example of a typical configuration of `/etc/resolv.conf` for a server on the domain "*example.com*" and using two public DNS servers.

```
search example.com
nameserver 8.8.8.8
nameserver 8.8.4.4
```

The *search* option can also be used with multiple domain names so that DNS queries will be appended in the order in which they are entered. For example, your network may have multiple sub-domains to search; a parent domain of *example.com*, and two sub-domains, *sales.example.com* and *dev.example.com*.

If you have multiple domains you wish to search, your configuration might look like the following.

```
search example.com sales.example.com dev.example.com
nameserver 8.8.8.8
nameserver 8.8.4.4
```

If you try to ping a host with the name of *server1*, your system will automatically query DNS for its Fully Qualified Domain Name (FQDN) in the following order:

1. server1**.example.com**
2. server1**.sales.example.com**
3. server1**.dev.example.com**

If no matches are found, the DNS server will provide a result of *notfound* and the DNS query will fail.

1.3.2. Static Hostnames

Static hostnames are locally defined hostname-to-IP mappings located in the file `/etc/hosts`. Entries in the `hosts` file will have precedence over DNS by default. This means that if your system tries to resolve a hostname and it matches an entry in /etc/hosts, it will not attempt to look up the record in DNS. In some configurations, especially when Internet access is not required, servers that communicate with a limited number of resources can be conveniently set to use static hostnames instead of DNS.

The following is an example of a `hosts` file where a number of local servers have been identified by simple hostnames, aliases and their equivalent Fully Qualified Domain Names (FQDN's).

```
127.0.0.1 localhost
127.0.1.1 ubuntu-server
10.0.0.11 server1 vpn server1.example.com
10.0.0.12 server2 mail server2.example.com
10.0.0.13 server3 www server3.example.com
10.0.0.14 server4 file server4.example.com
```

In the above example, notice that each of the servers have been given aliases in addition to their proper names and FQDN's. *Server1* has been mapped to the name *vpn*, *server2* is referred to as *mail*, *server3* as *www*, and *server4* as *file*.

1.3.3. Name Service Switch Configuration

The order in which your system selects a method of resolving hostnames to IP addresses is controlled by the Name Service Switch (NSS) configuration file `/etc/nsswitch.conf`. As mentioned in the previous section, typically static hostnames defined in the systems `/etc/hosts` file have precedence over names resolved from DNS. The following is an example of the line responsible for this order of hostname lookups in the file `/etc/nsswitch.conf`.

```
hosts:          files mdns4_minimal [NOTFOUND=return] dns mdns4
```

- **files** first tries to resolve static hostnames located in `/etc/hosts`.
- **mdns4_minimal** attempts to resolve the name using Multicast DNS.
- **[NOTFOUND=return]** means that any response of *notfound* by the preceding *mdns4_minimal* process should be treated as authoritative and that the system should not try to continue hunting for an answer.
- **dns** represents a legacy unicast DNS query.
- **mdns4** represents a Multicast DNS query.

To modify the order of the above mentioned name resolution methods, you can simply change the *hosts:* string to the value of your choosing. For example, if you prefer to use legacy Unicast DNS versus Multicast DNS, you can change the string in `/etc/nsswitch.conf` as shown below.

```
hosts:          files dns [NOTFOUND=return] mdns4_minimal mdns4
```

1.4. Bridging

Il "bridging" di molteplici interfacce è una configurazione avanzata, ma utile in diversi scenari. Uno di questi scenari può consistere nel configurare un bridge con molteplici interfacce di rete e usare un firewall per filtrare il traffico tra due segmenti della rete. Un altro scenario consiste nell'usare un bridge su un sistema con una sola interfaccia per permettere alle macchine virtuali accesso diretto alla rete esterna. L'esempio che segue prende in considerazione quest'ultimo scenario.

Prima di configurare un bridge è necessario installare il pacchetto bridge-utils. In un terminale digitare:

```
sudo apt-get install bridge-utils
```

Configurare il bridge modificando il file `/etc/network/interfaces`:

```
auto lo
iface lo inet loopback

auto br0
iface br0 inet static
```

```
address 192.168.0.10
network 192.168.0.0
netmask 255.255.255.0
broadcast 192.168.0.255
gateway 192.168.0.1
bridge_ports eth0
bridge_fd 9
bridge_hello 2
bridge_maxage 12
bridge_stp off
```

Inserire i valori appropriati per la propria interfaccia di rete.

Riavviare la rete per abilitare il bridge sull'interfaccia:

```
sudo /etc/init.d/networking restart
```

La nuova interfaccia dovrebbe ora essere funzionante. L'applicazione brctl fornisce utili informazioni riguardo lo stato del bridge, controlla le interfacce che compongono il bridge, ecc... Per maggiori informazioni, consultare la pagina di manuale: **man brctl**.

1.5. Risorse

- The *Ubuntu Wiki Network page*[1] has links to articles covering more advanced network configuration.
- The *interfaces man page*[2] has details on more options for `/etc/network/interfaces`.
- The *dhclient man page*[3] has details on more options for configuring DHCP client settings.
- For more information on DNS client configuration see the *resolver man page*[4]. Also, Chapter 6 of O'Reilly's *Linux Network Administrator's Guide*[5] is a good source of resolver and name service configuration information.
- For more information on *bridging* see the *brctl man page*[6] and the Linux Foundation's *Net:Bridge*[7] page.

2. TCP/IP

Il protocollo TCP/IP (Transmission Control Protocol e Internet Protocol) è un insieme standard di protocolli sviluppato nella seconda metà degli anni '70 dalla DARPA (Defence Advanced Research Project Agency) con lo scopo di permettere la comunicazione tra diversi tipi di computer e di reti di computer. TCP/IP è il motore di Internet, ecco perché è l'insieme di protocolli di rete più diffuso al mondo.

2.1. Introduzione a TCP/IP

I due protocolli che compongono TCP/IP, interagiscono con differenti aspetti di una rete. L'*Internet Protocol*, la parte "IP" di TCP/IP, è un protocollo privo di connessione che interagisce solamente con il routing dei pacchetti attraverso la rete, usando l'*IP Datagram* come unità di base delle informazioni che consiste in un'intestazione seguita da un messaggio. Il *Transmission Control Protocol*, la parte "TCP" di TCP/IP, consente agli host della rete di stabilire le connessioni che possono essere usate per scambiare dati. Inoltre, garantisce che i dati tra le connessioni siano consegnati correttamente e nello stesso ordine in cui sono stati inviati.

2.2. Configurazione di TCP/IP

La configurazione del protocollo TCP/IP è composta da vari elementi che debbono essere impostati modificando gli appropriati file di configurazione oppure adottando soluzioni quali un server DHCP (Dynamic Host Configuration Protocol); tale server provvede ad assegnare automaticamente le corrette impostazioni di configurazione TCP/IP ai client della rete. Questi valori di configurazione debbono essere impostati correttamente per consentire al sistema Ubuntu di operare adeguatamente in rete.

I tipici elementi di configurazione di TCP/IP e i loro scopi sono i seguenti:

- **Indirizzo IP**: l'indirizzo IP è una stringa d'identificazione unica, espressa da quattro numeri decimali compresi tra zero (0) e duecentocinquantacinque (255), separati da punti; ciascuno dei quattro numeri rappresenta otto (8) bit dell'indirizzo per una lunghezza totale di trentadue (32) bit per l'indirizzo completo. Questo formato è detto *notazione decimale a punti*.
- **Maschera di rete**: la maschera di rete (o semplicemente *netmask*) è una maschera locale di bit, ovvero un insieme di indicatori che separano la porzione di un indirizzo IP che indica la rete dai bit che indicano la *sotto-rete*. Ad esempio, in una rete di classe C, la maschera di rete standard è 255.255.255.0 che serve a mascherare i primi tre byte dell'indirizzo IP, consentendo all'ultimo byte dell'indirizzo IP di essere disponibile per specificare gli host della sotto-rete.
- **Indirizzo di rete**: l'indirizzo di rete è dato dai byte che comprendono la parte di rete di un indirizzo IP. Per esempio, l'host 12.128.1.2 in una rete di classe A deve usare 12.0.0.0 come indirizzo di rete, dove dodici (12) è il primo byte dell'indirizzo IP (la parte di rete), e gli zeri (0) nei rimanenti tre byte indicano tutti i possibili valori degli host. Un host di rete che ha un indirizzo IP 192.168.1.100 deve invece usare un indirizzo di rete di 192.168.1.0, nel quale i primi tre byte specificano la rete di classe C 192.168.2 e lo zero (0) per tutti i possibili valori degli host nella rete.

- **Indirizzo di broadcast**: l'indirizzo di broadcast è un indirizzo IP che permette di inviare dei dati di rete simultaneamente a tutti gli host di una data sotto-rete piuttosto che a uno specifico host. L'indirizzo broadcast generale di base per una rete IP è 255.255.255.255, ma questo indirizzo broadcast non può essere usato per inviare un messaggio broadcast a tutti gli host presenti in internet perché i router lo bloccherebbero. Un indirizzo di broadcast appropriato è quello che indica una specifica sotto-rete. Per esempio, in una rete privata di classe C, 192.168.1.0, l'indirizzo broadcast è 192.168.1.255. I messaggi broadcast sono di norma prodotti dai protocolli di rete come il protocollo per la risoluzione degli indirizzi (ARP, Address Resolution Protocol) e il protocollo delle informazioni di instradamento (RIP, Routing Information Protocol).
- **Indirizzo del gateway**: l'indirizzo del gateway è l'indirizzo IP attraverso il quale una particolare rete, o un host su una rete, può essere raggiunta. Se un host di rete desidera comunicare con un altro host di rete, senza essere localizzato nelle stessa rete, allora deve essere usato un *gateway*. In molti casi l'indirizzo del gateway coincide con quello di un router della medesima rete che ha il compito di far transitare il traffico ad altre reti o host, come Internet. L'impostazione del valore dell'indirizzo del gateway deve essere corretta, altrimenti il sistema non è in grado di raggiungere gli host che non si trovano sulla rete cui appartiene.
- **Indirizzo del server dei nomi**: l'indirizzo del server dei nomi rappresenta l'indirizzo IP del sistema DNS (Domain Name Service) che traduce il nome host della rete in un indirizzo IP reale. Esistono tre livelli di indirizzo del server dei nomi che possono essere specificati in ordine di precedenza: il server dei nomi *primario*, quello *secondario* e il *terziario*. Affinché il sistema possa tradurre i nomi host in indirizzi IP, è necessario specificare degli indirizzi validi per i server dei nomi che è possibile utilizzare all'interno della configurazione TCP/IP del sistema. Nella maggior parte dei casi, questi indirizzi vengono forniti dal proprio fornitore di servizio Internet, ma ne sono disponibili anche di gratuiti e liberamente utilizzabili, come i server di terzo livello di Verizon con indirizzi IP da 4.2.2.1 a 4.2.2.6.

Gli indirizzi IP, le maschere di rete, gli indirizzi di rete, gli indirizzi di broadcast e gli indirizzi di gateway sono tipicamente determinati attraverso appropriate direttive nel file `/etc/network/interfaces`. Gli indirizzi di server dei nomi sono tipicamente specificati attraverso le direttive *nameserver* nel file `/etc/resolv.conf`. Per maggiori informazioni, consultare rispettivamente le pagine di manuale di sistema per `interfaces` e `resolv.conf`, usando i seguenti comandi da digitare al prompt di un terminale:

Accedere alla pagina di manuale di sistema per `interfaces` con il seguente comando:

```
man interfaces
```

Accedere alla pagina di manuale di sistema per `resolv.conf` con il seguente comando:

```
man resolv.conf
```

2.3. Instradamento IP

L'instradamento IP è un modo per indicare e scoprire percorsi in una rete TCP/IP attraverso i quali inviare dati. L'instradamento utilizza un insieme di *tabelle di instradamento (routing)* per dirigere i pacchetti di dati in una rete dalla loro sorgente avanti fino alla destinazione, spesso attraverso molti nodi di rete intermediari chiamati *router*. Esistono due forme primarie di instradamento IP: l'*instradamento statico* e l'*instradamento dinamico*.

L'instradamento statico comporta l'aggiunta manuale di rotte IP nella tabella di instradamento del sistema, attività che viene fatta modificando la tabella di instradamento con il comando route. L'instradamento statico presenta molti vantaggi rispetto quello dinamico, come la semplicità di implementazione per piccole reti, la predicibilità (la tabella di instradamento è scritta a priori, quindi la rotta è sempre la stessa ogni volta che viene utilizzata) e il basso carico di lavoro sugli altri router e nodi di rete dovuto all'assenza di un protocollo di instradamento dinamico. In ogni caso, l'instradamento statico presenta anche degli svantaggi. Per esempio, è limitato a piccole reti e non è facilmente espandibile. L'instradamento statico fallisce completamente se si prova ad adattarlo ai ritardi della rete e le perdite lungo la rotta per la natura statica della rotta stessa.

L'instradamento dinamico serve nelle grandi reti con molte possibili rotte IP tra una sorgente e una destinazione. Fa uso di protocolli di instradamento speciali, come il protocollo di informazione dell'instradamento (RIP, Router Information Protocol) che gestisce le correzioni automatiche nella tabella di instradamento rendendo possibile l'instradamento dinamico. Ci sono molti vantaggi rispetto l'instradamento statico, come l'adattamento alle dimensioni superiori e l'abilità di adattarsi agli errori e alle perdite lungo le rotte della rete. Inoltre, necessita di una minore configurazione manuale delle tabelle di instradamento, dato che i router comunicano tra di loro la relativa esistenza e le possibili rotte. Questo tratto caratteristico elimina anche la possibilità di introdurre inesattezze nelle tabelle di instradamento causate da errori umani. In ogni caso, l'instradamento dinamico non è perfetto e presenta alcuni svantaggi come, l'aumento della complessità e del carico di lavoro dovuto alle comunicazioni dei router della rete, dei quali non può beneficiare subito l'utente finale che comunque consuma banda di rete.

2.4. TCP e UDP

TCP è un protocollo basato sulla connessione, che offre correzione d'errore e che garantisce la consegna dei dati attraverso ciò che è conosciuto come *controllo di flusso*. Il controllo di flusso determina quando il flusso di uno stream di dati debba essere fermato e i pacchetti di dati inviati in precedenza debbano essere reinviati a causa di problemi come *collisioni*, assicurando quindi la completa e accurata consegna dei dati. TCP è tipicamente usato nello scambio di informazioni importanti come transazioni di database.

UDP (User Datagram Protocol), al contrario, è un protocollo *senza connessione* che raramente tratta della trasmissione dei dati importanti a causa della mancanza del controllo di flusso o di un altro metodo che garantisca la consegna affidabile dei dati. UDP è normalmente usato in applicazioni come lo streaming audio e video, in cui risulta considerevolmente più veloce del protocollo TCP, data la

mancanza di correzione d'errore e del controllo di flusso, e in cui la perdita di alcuni pacchetti non è generalmente un evento catastrofico.

2.5. ICPM

ICMP (Internet Control Messaging Protocol) è un'estensione di IP (Internet Protocol), come definito nell'RFC (Request For Comments) numero 792; ICPM supporta pacchetti di rete contenenti messaggi di controllo, di errore e di informazione. ICMP è usato da applicazioni di rete come l'utilità ping, che consente di determinare la disponibilità di un host o un'interfaccia di rete. Esempi di alcuni dei messaggi di errore restituiti da ICMP utili sia agli host e interfacce di rete che ai router sono *Destination Unreachable* e *Time Exceeded*.

2.6. Demoni

I demoni sono speciali applicazioni di sistema che, tipicamente, sono in continua esecuzione sullo sfondo, attendendo dagli altri programmi richieste relative a funzioni da essi fornite. Molti demoni hanno a che fare con la rete e molti di questi in esecuzione sullo sfondo nei sistemi Ubuntu forniscono delle funzionalità legate alla rete. Alcuni esempi di questi demoni di rete includono *httpd* (Hyper Text Transport Protocol Daemon), che fornisce funzionalità di server web; *sshd* (Secure SHell Daemon), che fornisce funzionalità di login e trasferimento file sicuro da remoto; *imapd* (Internet Message Access Protocol Daemon), che fornisce servizi di email.

2.7. Risorse

- There are man pages for *TCP*[8] and *IP*[9] that contain more useful information.
- Inoltre, consultare il RedBook di IBM: *TCP/IP Tutorial and Technical Overview*[10].
- Un'altra utile risorsa è il libro *TCP/IP Network Administration*[11].

3. DHCP (Dynamic Host Configuration Protocol)

Il DHCP (Dynamic Host Configuration Protocol) è un servizio di rete che consente di assegnare automaticamente le impostazioni per agli host da un server, senza la necessità di dover configurare manualmente ogni singolo host nella rete. I computer configurati per essere client DHCP non hanno alcun controllo sulle impostazioni che ricevono dal server DHCP e la configurazione è trasparente all'utente del computer.

Le impostazioni comuni fornite da un server DHCP a un client includono:

- Indirizzo IP e maschera di rete
- DNS
- WINS

Un server DHCP può fornire anche altre proprietà di configurazione come:

- Nome dell'host
- Nome del dominio
- Gateway predefinito
- Server NTP (Network Time Protocol)
- Server di stampa

Il vantaggio di utilizzare DHCP è che i cambiamenti apportati alla rete, per esempio una modifica dell'indirizzo del server DNS, devono essere apportati solamente al server DHCP, mentre tutti gli host della rete vengono riconfigurati quando i client DHCP interrogano il server DHCP. Come ulteriore vantaggio, risulta anche molto semplice integrare nuovi computer nella rete, senza la necessità di controllare la disponibilità di un indirizzo IP. I conflitti nell'allocazione degli indirizzi IP sono quindi notevolmente ridotti.

Le impostazioni di configurazione sono fornite da un server DHCP usando due metodi:

Indirizzo MAC
: Questo metodo comporta l'utilizzo di DHCP per identificare l'indirizzo hardware univoco di ogni scheda di rete collegata alla rete, così da fornire in modo continuato una configurazione costante ogni volta che il client DHCP avanza una richiesta al server DHCP usando quel particolare dispositivo di rete.

Spazio degli indirizzi
: Questo metodo comporta la definizione di un insieme o intervallo di indirizzi IP (a volte indicati come pool) con cui configurare dinamicamente i client DHCP in base all'ordine di arrivo delle richieste (la prima che arriva è la prima servita, disciplina FIFO). Dopo un determinato periodo, se il client DHCP non è presente in rete, la configurazione scade e viene reinserita nello spazio di indirizzi per poter essere riutilizzata.

Ubuntu comprende sia un server che un client DHCP. Il server è dhcpd (dynamic host configuration protocol daemon). Il client fornito è dhclient e dovrebbe essere installato su tutti i computer che

necessitano di essere configurati automaticamente. Entrambi i programmi sono facili da installare e da configurare e vengono lanciati automaticamente all'avvio del sistema.

3.1. Installazione

A un prompt di terminale, inserire il seguente comando per installare dhcpd:

```
sudo apt-get install dhcp3-server
```

È necessario modificare il file /etc/dhcp3/dhcpd.conf secondo le proprie necessità e per avere una configurazione particolare.

È inoltre necessario modificare il file /etc/default/dhcp3-server per specificare le interfacce su cui stare in ascolto. Di base è impostata l'interfaccia eth0.

I messaggi di dhcpd vengono inviati nel syslog, consultare quindi i relativi messaggi per quelli di diagnostica.

3.2. Configurazione

Il messaggio di errore con cui si conclude l'installazione potrebbe essere fuorviante, ma i passi seguenti consentono di configurare il servizio.

Nella maggior parte dei casi si vuole assegnare un indirizzo IP in modo casuale. Questo può essere ottenuto con impostazioni come le seguenti:

```
# Sample /etc/dhcpd.conf
# (add your comments here)
default-lease-time 600;
max-lease-time 7200;
option subnet-mask 255.255.255.0;
option broadcast-address 192.168.1.255;
option routers 192.168.1.254;
option domain-name-servers 192.168.1.1, 192.168.1.2;
option domain-name "miodominio.example";

subnet 192.168.1.0 netmask 255.255.255.0 {
range 192.168.1.10 192.168.1.100;
range 192.168.1.150 192.168.1.200;
}
```

Come risultato si ottiene che il server DHCP fornisce a un client un indirizzo IP nell'intervallo 192.168.1.10 ~ 192.168.1.100 oppure 192.168.1.150 ~ 192.168.1.200. Se il client non richiede uno specifico intervallo di tempo, la durata di "affitto" di un indirizzo IP è di 600 secondi; in caso contrario il valore massimo (consentito) è di 7200 secondi. Il server inoltre "avvisa" il client di utilizzare 255.255.255.0 come maschera di sottorete, 192.168.1.255 come indirizzo di broadcast, 192.168.1.254 come gateway e 192.168.1.1 e 192.168.1.2 come server DNS.

Se è necessario specificare un server WINS per i client Windows, è necessario includere l'opzione netbios-name-servers, per esempio

```
option netbios-name-servers 192.168.1.1;
```

Le impostazioni di configurazione per dhcpd sono prese dal mini-HOWTO di DHCP, il quale può essere trovato *qui*[12].

3.3. Riferimenti

- The *dhcp3-server Ubuntu Wiki*[13] page has more information.
- For more `/etc/dhcp3/dhcpd.conf` options see the *dhcpd.conf man page*[14].
- Inoltre consultare le *DHCP FAQ*[15]

[12] http://www.tldp.org/HOWTO/DHCP/index.html

4. Sincronizzazione del tempo con NTP

Questa sezione descrive i metodi per mantenere l'ora esatta del proprio computer, utile per i server, ma non necessario (o desiderabile) per computer desktop.

NTP è un protocollo TCP/IP per sincronizzare l'ora attraverso la rete: un client richiede l'ora corrente a un server e usa questa per impostare il proprio orologio.

Oltre questa semplice descrizione, c'è molta complessità. Esistono diversi livelli di server NTP, con i server di primo livello collegati a orologi atomici (solitamente via GPS) e i server dei livelli due e tre che dividono il carico delle richieste attraverso Internet. Inoltre, il software dei client è molto più complesso di quanto si possa immaginare: deve gestire i ritardi nella comunicazione e regolare l'ora in modo da non compromettere tutti i processi in esecuzione sul server.

Ubuntu ha due metodi per impostare automaticamente l'orologio: ntpdate e ntpd.

4.1. ntpdate

Ubuntu dispone di ntpdate e viene eseguito all'avvio per configurare l'orologio in base al server NTP di Ubuntu. L'orologio di un server potrebbe comunque cambiare tra un riavvio e l'altro, anche di un fattore considerevole, ed è pertanto consigliato occasionalmente regolare l'ora manualmente. Il metodo più semplice per fare questo è quello di indicare a cron di eseguire ntpdate ogni giorno. Con i privilegi di root e un editor di testo, creare un file chiamato `/etc/cron.daily/ntpdate` con il seguente contenuto:

```
ntpdate -s ntp.ubuntu.com
```

Il file `/etc/cron.daily/ntpdate` deve essere eseguibile.

```
sudo chmod 755 /etc/cron.daily/ntpdate
```

4.2. ntpd

ntpdate è uno strumento abbastanza semplice, può regolare l'ora una sola volta al giorno in un'unica modifica, mentre il demone di ntp (ntpd) è molto più raffinato. Calcola lo spostamento dell'orologio del proprio sistema e lo regola continuamente, così non ci sono mai grandi modifiche che possono portare a file di registro inconsistenti. Il costo di questo è un leggero uso di processore e memoria, ma trascurabili per un server moderno.

To install ntpd, from a terminal prompt enter:

```
sudo apt-get install ntp
```

4.3. Modificare i server NTP

In both cases above, your system will use Ubuntu's NTP server at *ntp.ubuntu.com* by default. This is OK, but you might want to use several servers to increase accuracy and resilience, and you may want

to use time servers that are geographically closer to you. to do this for ntpdate, change the contents of `/etc/cron.daily/ntpdate` to:

```
ntpdate -s ntp.ubuntu.com pool.ntp.org
```

And for ntpd edit `/etc/ntp.conf` to include additional server lines:

```
server ntp.ubuntu.com
server pool.ntp.org
```

You may notice *pool.ntp.org* in the examples above. This is a really good idea which uses round-robin DNS to return an NTP server from a pool, spreading the load between several different servers. Even better, they have pools for different regions - for instance, if you are in New Zealand, so you could use *nz.pool.ntp.org* instead of *pool.ntp.org*. Look at *http://www.pool.ntp.org/* for more details.

You can also Google for NTP servers in your region, and add these to your configuration. To test that a server works, just type:

```
sudo ntpdate ntp.server.name
```

4.4. Riferimenti

- See the *Ubuntu Time*[16] wiki page for more information.
- *Supporto NTP*[17]
- *FAQ e HOWTO per NTP*[18]

Capitolo 5. Amministrazione remota

There are many ways to remotely administer a Linux server. This chapter will cover one of the most popular OpenSSH.

1. Server OpenSSH

1.1. Introduzione

Questa sezione introduce una serie di strumenti per il controllo remoto di computer e per il trasferimento di dati tra i computer in rete di nome *OpenSSH*. Vengono spiegate alcune delle possibili impostazioni del server OpenSSH e come modificarne la configurazione in Ubuntu.

OpenSSH è una versione libera della famiglia di protocolli e strumenti SSH (Secure SHell) per il controllo remoto di un computer o per il trasferimento di file tra computer. Gli strumenti tradizionali usati per svolgere queste funzioni, come telnet o rcp, sono insicuri e quando utilizzati trasmettono la password dell'utente in chiaro. OpenSSH fornisce un demone server e degli strumenti lato client per facilitare operazioni di controllo remoto e trasferimento di file in sicurezza e con crittografia, sostituendo in modo completo gli strumenti tradizionali.

Il componente server di OpenSSH, sshd, è in ascolto continuo per le connessioni in arrivo dei client, qualunque sia lo strumento usato sui client. Quando avviene una richiesta di connessione, per mezzo di sshd viene impostata la corretta connessione in base allo strumento utilizzato dal client. Per esempio, se il computer remoto sta effettuando una connessione con l'applicazione client ssh, il server OpenSSH imposta, dopo l'autenticazione, una sessione di controllo remoto. Se un utente remoto si connette a un server OpenSSH con scp, il demone server OpenSSH inizializza, dopo l'autenticazione, una procedura di copia sicura di file tra il server e il client. OpenSSH permette l'utilizzo di diversi metodi di autenticazione, inclusi password semplice, chiave pubblica e ticket Kerberos.

1.2. Installazione

L'installazione delle applicazioni server e client di OpenSSH è semplice. Per installare l'applicazione client OpenSSH sui sistemi Ubuntu, usare questo comando al prompt di un terminale:

```
sudo apt-get install openssh-client
```

Per installare l'applicazione server di OpenSSH e i relativi file di supporto, usare questo comando al prompt di un terminale:

```
sudo apt-get install openssh-server
```

È possibile scegliere di installare il pacchetto openssh-server durante il processo di installazione della Server Edition.

1.3. Configurazione

È possibile configurare il comportamento predefinito dell'applicazione server di OpenSSH, sshd, modificando il file `/etc/ssh/sshd_config`. Per maggiori informazioni riguardo le direttive di

configurazione usate in questo file, consultare l'appropriata pagina di manuale inserendo, a un prompt di terminale, il seguente comando:

```
man sshd_config
```

All'interno del file di configurazione di sshd sono presenti diverse direttive per controllare impostazioni riguardo la comunicazione o i mezzi di autenticazione. Di seguito vengono riportati degli esempi di direttive di configurazione che è possibile modificare modificando il file `/etc/ssh/sshd_config`.

Prima di modificare il file di configurazione, è consigliato fare una copia del file originale e proteggerla dalla scrittura, così da avere le impostazioni originali come riferimento ed eventualmente riusarle se necessario.

Copiare il file `/etc/ssh/sshd_config` e proteggerlo da scrittura, con il seguente comando, digitando a un prompt di terminale:

```
sudo cp /etc/ssh/sshd_config /etc/ssh/sshd_config.original
sudo chmod a-w /etc/ssh/sshd_config.original
```

Quelli che seguono sono esempi delle direttive di configurazione che è possibile cambiare:

- Per impostare OpenSSH in modo da restare in ascolto sulla porta TCP 2222 invece che sulla predefinita porta TCP 22, cambiare la direttiva Port come segue:

  Port 2222

- Per consentire l'utilizzo in sshd di credenziali di accesso basate su chiave pubblica, aggiungere o modificare la riga:

  PubkeyAuthentication yes

  Nel file `/etc/ssh/sshd_config` assicurarsi che la riga non sia commentata.

- Per far sì che il server OpenSSH mostri il contenuto del file `/etc/issue.net` come un banner di pre-accesso, aggiungere o modificare la riga:

  Banner /etc/issue.net

  Nel file `/etc/ssh/sshd_config`.

Dopo aver apportato dei cambiamenti al file `/etc/ssh/sshd_config`, salvarlo e, per rendere effettivi i cambiamenti, riavviare il demone sshd usando il seguente comando:

```
sudo /etc/init.d/ssh restart
```

Per poter adattare il comportamento dell'applicazione server alle proprie necessità, sono disponibili molte altre direttive di configurazione per sshd. Se però l'unico metodo per

accedere a un server è ssh, è necessario prestare molta attenzione. Un qualsiasi errore nella configurazione di sshd attraverso `/etc/ssh/sshd_config` può precludere l'accesso al server dopo il suo riavvio oppure impedire l'avvio stesso di sshd a causa di una errata direttiva di configurazione. Perciò è necessaria molta attenzione nella modifica di questo file su un server remoto.

1.4. Chiavi SSH

Le *chiavi* SSH consentono l'autenticazione tra due host senza la necessità di una password. La'utenticazione con chiave SSH utilizza due chiavi, una *privata* e una *pubblica*.

Per generare le chiavi, in un terminale, digitare:

```
ssh-keygen -t dsa
```

Vengono generate le chiavi usando un'autenticazione *DSA*. Durante questo processo viene chiesto di inserire un password, premere semplicemente *Invio* quando chiesto di creare la chiave.

La chiave *pubblica* viene salvata, in modo predefinito, nel file `~/.ssh/id_dsa.pub`, mentre quella *privata* in `~/.ssh/id_dsa`. Ora, copiare il file `id_dsa.pub` nell'host remoto e aggiungere il suo contenuto al file `~/.ssh/authorized_keys` digitando:

```
ssh-copy-id NOME_UTENTE@HOST_REMOTO
```

Infine, controllare i permessi del file `authorized_keys`: solo l'utente autenticato dovrebbe avere i permessi di lettura e scrittura. Nel caso non fossero corretti, modificarli:

```
chmod 600 .ssh/authorized_keys
```

Dovrebbe essere possibile ora collegarsi via SSH all'host senza l'utilizzo di una password.

1.5. Riferimenti

- *Ubuntu Wiki SSH*[1] page.
- *Sito web di OpenSSH*[2]
- *Pagina wiki di OpenSSH avanzato*[3]

2. Puppet

Puppet is a cross platform framework enabling system administrators to perform common tasks using code. The code can do a variety of tasks from installing new software, to checking file permissions, or updating user accounts. Puppet is great not only during the initial installation of a system, but also throughout the system's entire life cycle. In most circumstances puppet will be used in a client/server configuration.

This section will cover installing and configuring puppet in a client/server configuration. This simple example will demonstrate how to install Apache using Puppet.

2.1. Installazione

To install puppet, in a terminal on the *server* enter:

```
sudo apt-get install puppetmaster
```

On the *client* machine, or machines, enter:

```
sudo apt-get install puppet
```

2.2. Configurazione

Prior to configuring puppet you may want to add a DNS *CNAME* record for *puppet.example.com*, where *example.com* is your domain. By default puppet clients check DNS for puppet.example.com as the puppet server name, or *Puppet Master*. See *Capitolo 7, DNS (Domain Name Service) [94]* for more DNS details.

If you do not wish to use DNS, you can add entries to the server and client `/etc/hosts` file. For example, in the puppet server's `/etc/hosts` file add:

```
127.0.0.1 localhost.localdomain localhost puppet
192.168.1.17 meercat02.example.com meercat02
```

On each puppet client, add an entry for the server:

```
192.168.1.16 meercat.example.com meercat puppet
```

Replace the example IP addresses and domain names above with your actual server and client addresses and domain names.

Now setup some resources for apache2. Create a file `/etc/puppet/manifests/site.pp` containing the following:

```
package {
```

```
    'apache2':
        ensure => installed
}

service {
    'apache2':
        ensure => true,
        enable => true,
        require => Package['apache2']
}
```

Next, create a node file `/etc/puppet/manifests/nodes.pp` with:

```
node 'meercat02.example.com' {
   include apache2
}
```

Replace *meercat02.example.com* with your actual puppet client's host name.

The final step for this simple puppet server is to restart the daemon:

```
sudo /etc/init.d/puppetmaster restart
```

Now everything is configured on the puppet server, it is time to configure the client.

First, configure the puppet agent daemon to start. Edit `/etc/default/puppet`, changing *START* to yes:

```
START=yes
```

Then start the service:

```
sudo /etc/init.d/puppet start
```

Back on the puppet server sign the client certificate by entering:

```
sudo puppetca --sign meercat02.example.com
```

Check `/var/log/syslog` for any errors with the configuration. If all goes well the apache2 package and it's dependencies will be installed on the puppet client.

This example is *very* simple, and does not highlight many of Puppet's features and benefits. For more information see *Sezione 2.3, «Risorse» [53]*.

2.3. Risorse

- See the *Official Puppet Documentation*[4] web site.

- Also see *Pulling Strings with Puppet*[5].
- Another source of additional information is the *Ubuntu Wiki Puppet Page*[6].

Capitolo 6. Autenticazione di rete

Questa sezione spiega i vari protocolli di autenticazione di rete.

1. Server OpenLDAP

LDAP è un acronimo per "Lightweight Directory Access Protocol", una versione semplificata del protocollo X.500. La directory impostata in questa sezione sarà usata per l'autenticazione. LDAP può comunque essere usato in diversi modi: autenticazione, directory condivisa (per i client mail), rubrica indirizzi. ecc...

Per descrivere LDAP velocemente, tutte le informazioni vengono archiviate in una struttura ad albero. Con OpenLDAP si ha la libertà di scegliere lo sviluppo dell'albero delle directory (il "Directory Information Tree", DIT). Per iniziare, si prende un esempio di un albero basilare con due nodi al di sotto della radice.

- Il nodo «People» è dove i propri utenti vengono salvati
- Il nodo «Groups» è dove i propri gruppi vengono salvati

Prima di iniziare, è necessario determinare quale sarà la radice della propria directory LDAP. In modo predefinito, l'albero sarà determinato dal proprio FQDN (Fully Qualified Domain Name), se il domino è "example.com" (usato in questo esempio), la radice sarà "dc=example,dc=com".

1.1. Installazione

Per prima cosa, installare il demone server OpenLDAP slapd e ldap-utils, un pacchetto contenente le utilità di gestione LDAP:

```
sudo apt-get install slapd ldap-utils
```

By default slapd is configured with minimal options needed to run the slapd daemon.

The configuration example in the following sections will match the domain name of the server. For example, if the machine's Fully Qualified Domain Name (FQDN) is ldap.example.com, the default suffix will be *dc=example,dc=com*.

1.2. Popolare LDAP

OpenLDAP uses a separate directory which contains the *cn=config* Directory Information Tree (DIT). The *cn=config* DIT is used to dynamically configure the slapd daemon, allowing the modification of schema definitions, indexes, ACLs, etc without stopping the service.

The backend *cn=config* directory has only a minimal configuration and will need additional configuration options in order to populate the frontend directory. The frontend will be populated with a "classical" scheme that will be compatible with address book applications and with Unix Posix accounts. Posix accounts will allow authentication to various applications, such as web applications, email Mail Transfer Agent (MTA) applications, etc.

Affinché le applicazioni esterne possano autenticarsi via LDAP, è necessario che siano configurate a tal fine. Per come fare, fare riferimento alla documentazione di ogni singola applicazione.

Remember to change *dc=example,dc=com* in the following examples to match your LDAP configuration.

First, some additional schema files need to be loaded. In a terminal enter:

```
sudo ldapadd -Y EXTERNAL -H ldapi:/// -f /etc/ldap/schema/cosine.ldif
sudo ldapadd -Y EXTERNAL -H ldapi:/// -f /etc/ldap/schema/nis.ldif
sudo ldapadd -Y EXTERNAL -H ldapi:/// -f /etc/ldap/schema/inetorgperson.ldif
```

Next, copy the following example LDIF file, naming it `backend.example.com.ldif`, somewhere on your system:

```
# Load dynamic backend modules
dn: cn=module,cn=config
objectClass: olcModuleList
cn: module
olcModulepath: /usr/lib/ldap
olcModuleload: back_hdb

# Database settings
dn: olcDatabase=hdb,cn=config
objectClass: olcDatabaseConfig
objectClass: olcHdbConfig
olcDatabase: {1}hdb
olcSuffix: dc=example,dc=com
olcDbDirectory: /var/lib/ldap
olcRootDN: cn=admin,dc=example,dc=com
olcRootPW: secret
olcDbConfig: set_cachesize 0 2097152 0
olcDbConfig: set_lk_max_objects 1500
olcDbConfig: set_lk_max_locks 1500
olcDbConfig: set_lk_max_lockers 1500
olcDbIndex: objectClass eq
olcLastMod: TRUE
olcDbCheckpoint: 512 30
olcAccess: to attrs=userPassword by dn="cn=admin,dc=example,dc=com" write by anonymous auth by self
olcAccess: to attrs=shadowLastChange by self write by * read
olcAccess: to dn.base="" by * read
olcAccess: to * by dn="cn=admin,dc=example,dc=com" write by * read
```

Change *olcRootPW: secret* to a password of your choosing.

Now add the LDIF to the directory:

```
sudo ldapadd -Y EXTERNAL -H ldapi:/// -f backend.example.com.ldif
```

The frontend directory is now ready to be populated. Create a `frontend.example.com.ldif` with the following contents:

```
# Create top-level object in domain
dn: dc=example,dc=com
objectClass: top
objectClass: dcObject
objectclass: organization
o: Example Organization
dc: Example
description: LDAP Example

# Admin user.
dn: cn=admin,dc=example,dc=com
objectClass: simpleSecurityObject
objectClass: organizationalRole
cn: admin
description: LDAP administrator
userPassword: secret

dn: ou=people,dc=example,dc=com
objectClass: organizationalUnit
ou: people

dn: ou=groups,dc=example,dc=com
objectClass: organizationalUnit
ou: groups

dn: uid=john,ou=people,dc=example,dc=com
objectClass: inetOrgPerson
objectClass: posixAccount
objectClass: shadowAccount
uid: john
sn: Doe
givenName: John
cn: John Doe
displayName: John Doe
uidNumber: 1000
gidNumber: 10000
userPassword: password
gecos: John Doe
loginShell: /bin/bash
homeDirectory: /home/john
shadowExpire: -1
shadowFlag: 0
shadowWarning: 7
shadowMin: 8
shadowMax: 999999
shadowLastChange: 10877
mail: john.doe@example.com
postalCode: 31000
l: Toulouse
o: Example
mobile: +33 (0)6 xx xx xx xx
```

```
homePhone: +33 (0)5 xx xx xx xx
title: System Administrator
postalAddress:
initials: JD

dn: cn=example,ou=groups,dc=example,dc=com
objectClass: posixGroup
cn: example
gidNumber: 10000
```

In questo esempio sono stati impostati la struttura della directory, un utente e un gruppo. In altri esempi potrebbe essere possibile notare, in ogni voce, l'elemento *objectClass: top*, ma dato che è il comportamento predefinito, non è necessario inserirlo esplicitamente.

Add the entries to the LDAP directory:

```
sudo ldapadd -x -D cn=admin,dc=example,dc=com -W -f frontend.example.com.ldif
```

We can check that the content has been correctly added with the ldapsearch utility. Execute a search of the LDAP directory:

```
ldapsearch -xLLL -b "dc=example,dc=com" uid=john sn givenName cn

dn: uid=john,ou=people,dc=example,dc=com
cn: John Doe
sn: Doe
givenName: John
```

Una semplice spiegazione:

- *-x*: non usa il metodo di autenticazione, predefinito, SASL.
- *-LLL*: disabilita la stampa di informazioni sullo schema LDIF.

1.3. Further Configuration

L'albero *cn=config* può essere manipolato usando le utilità presenti nel pacchetto ldap-utils. Per esempio:

- Usare ldapsearch per visualizzare l'albero, inserendo la password dell'amministratore impostata durante l'installazione o la riconfigurazione:

```
sudo ldapsearch -LLL -Y EXTERNAL -H ldapi:/// -b cn=config dn

SASL/EXTERNAL authentication started
SASL username: gidNumber=0+uidNumber=0,cn=peercred,cn=external,cn=auth
SASL SSF: 0
dn: cn=config
```

```
dn: cn=module{0},cn=config

dn: cn=schema,cn=config

dn: cn={0}core,cn=schema,cn=config

dn: cn={1}cosine,cn=schema,cn=config

dn: cn={2}nis,cn=schema,cn=config

dn: cn={3}inetorgperson,cn=schema,cn=config

dn: olcDatabase={-1}frontend,cn=config

dn: olcDatabase={0}config,cn=config

dn: olcDatabase={1}hdb,cn=config
```

The output above is the current configuration options for the *cn=config* backend database. Your output may be vary.

- Come esempio per modificare un albero *cn=config*, aggiungere un altro attributo all'indice usando il comando ldapmodify:

```
sudo ldapmodify -Y EXTERNAL -H ldapi:///
```

```
SASL/EXTERNAL authentication started
SASL username: gidNumber=0+uidNumber=0,cn=peercred,cn=external,cn=auth
SASL SSF: 0
dn: olcDatabase={1}hdb,cn=config
add: olcDbIndex
olcDbIndex: uidNumber eq

modifying entry "olcDatabase={1}hdb,cn=config"
```

Una volta completata la modifica, premere *Ctrl+D* per uscire dall'utilità:

- ldapmodify è anche in grado di leggere le modifica da un file. Copiare e incolalre quanto segue in un file chiamato `uid_index.ldif`:

```
dn: olcDatabase={1}hdb,cn=config
add: olcDbIndex
olcDbIndex: uid eq,pres,sub
```

Eseguire ldapmodify:

```
sudo ldapmodify -Y EXTERNAL -H ldapi:/// -f uid_index.ldif
```

```
SASL/EXTERNAL authentication started
SASL username: gidNumber=0+uidNumber=0,cn=peercred,cn=external,cn=auth
SASL SSF: 0
modifying entry "olcDatabase={1}hdb,cn=config"
```

Qusto metodo è molto utile per applicare grandi modifiche.

- Adding additional *schemas* to slapd requires the schema to be converted to LDIF format. The `/etc/ldap/schema` directory contains some schema files already converted to LDIF format as demonstrated in the previous section. Fortunately, the slapd program can be used to automate the conversion. The following example will add the *dyngroup.schema*:

  1. Creare un file di conversione `schema_convert.conf` contenente le seguenti righe:

     ```
     include /etc/ldap/schema/core.schema
     include /etc/ldap/schema/collective.schema
     include /etc/ldap/schema/corba.schema
     include /etc/ldap/schema/cosine.schema
     include /etc/ldap/schema/duaconf.schema
     include /etc/ldap/schema/dyngroup.schema
     include /etc/ldap/schema/inetorgperson.schema
     include /etc/ldap/schema/java.schema
     include /etc/ldap/schema/misc.schema
     include /etc/ldap/schema/nis.schema
     include /etc/ldap/schema/openldap.schema
     include /etc/ldap/schema/ppolicy.schema
     ```

  2. Creare una directory temporanea in cui salvare l'output:

     ```
     mkdir /tmp/ldif_output
     ```

  3. Utilizzando slapcat. convertire il file schema in LDIF:

     ```
     slapcat -f schema_convert.conf -F /tmp/ldif_output -n0 -s "cn={5}dyngroup,cn=schema,cn=config
     ```

     Adjust the configuration file name and temporary directory names if yours are different.
     It may be worthwhile to keep the `ldif_output` directory around in case you want to add additional schemas in the future.

     

     The *"cn={5}"* index number may change according to the configuration ordering. To find out the correct number execute the following:

     ```
     slapcat -f schema_convert.conf -F /tmp/ldif_output -n 0 | grep dyngroup
     ```

     Replace *dyngroup* with the appropriate schema name.

4. Edit the `/tmp/cn\=dyngroup.ldif` file, changing the following attributes:

```
dn: cn=dyngroup,cn=schema,cn=config
...
cn: dyngroup
```

Rimuovere le seguenti righe dalla fine del file:

```
structuralObjectClass: olcSchemaConfig
entryUUID: 10dae0ea-0760-102d-80d3-f9366b7f7757
creatorsName: cn=config
createTimestamp: 20080826021140Z
entryCSN: 20080826021140.791425Z#000000#000#000000
modifiersName: cn=config
modifyTimestamp: 20080826021140Z
```

I valori degli attributi possono variare, basta solo assicurarsi che gli attributi siano rimossi.

5. In fine, usando l'utilità ldapadd, aggiugnere il nuovo schema alla directory:

```
sudo ldapadd -Y EXTERNAL -H ldapi:/// -f /tmp/cn\=dyngroup.ldif
```

There should now be a *dn: cn={4}dyngroup,cn=schema,cn=config* entry in the cn=config tree.

1.4. LDAP Replication

LDAP diventa spesso un servizio altamente critico all'interno della rete e diversi sistemi possono dipendere su LDAP per autenticazione, autorizzazioni, configurazione, ecc... In questi casi è molto utile impostare un sistema ridondante attraverso l'uso della replicazione.

Replication is achieved using the *Syncrepl* engine. Syncrepl allows the changes to be synced using a *consumer*, *provider* model. A provider sends directory changes to consumers.

1.4.1. Provider Configuration

The following is an example of a *Single-Master* configuration. In this configuration one OpenLDAP server is configured as a *provider* and another as a *consumer*.

1. First, configure the provider server. Copy the following to a file named `provider_sync.ldif`:

```
# Add indexes to the frontend db.
dn: olcDatabase={1}hdb,cn=config
changetype: modify
add: olcDbIndex
olcDbIndex: entryCSN eq
-
```

```
add: olcDbIndex
olcDbIndex: entryUUID eq

#Load the syncprov and accesslog modules.
dn: cn=module{0},cn=config
changetype: modify
add: olcModuleLoad
olcModuleLoad: syncprov
-
add: olcModuleLoad
olcModuleLoad: accesslog

# Accesslog database definitions
dn: olcDatabase={2}hdb,cn=config
objectClass: olcDatabaseConfig
objectClass: olcHdbConfig
olcDatabase: {2}hdb
olcDbDirectory: /var/lib/ldap/accesslog
olcSuffix: cn=accesslog
olcRootDN: cn=admin,dc=example,dc=com
olcDbIndex: default eq
olcDbIndex: entryCSN,objectClass,reqEnd,reqResult,reqStart

# Accesslog db syncprov.
dn: olcOverlay=syncprov,olcDatabase={2}hdb,cn=config
changetype: add
objectClass: olcOverlayConfig
objectClass: olcSyncProvConfig
olcOverlay: syncprov
olcSpNoPresent: TRUE
olcSpReloadHint: TRUE

# syncrepl Provider for primary db
dn: olcOverlay=syncprov,olcDatabase={1}hdb,cn=config
changetype: add
objectClass: olcOverlayConfig
objectClass: olcSyncProvConfig
olcOverlay: syncprov
olcSpNoPresent: TRUE

# accesslog overlay definitions for primary db
dn: olcOverlay=accesslog,olcDatabase={1}hdb,cn=config
objectClass: olcOverlayConfig
objectClass: olcAccessLogConfig
olcOverlay: accesslog
olcAccessLogDB: cn=accesslog
olcAccessLogOps: writes
olcAccessLogSuccess: TRUE
# scan the accesslog DB every day, and purge entries older than 7 days
olcAccessLogPurge: 07+00:00 01+00:00
```

2. The AppArmor profile for slapd will need to be adjusted for the accesslog database location. Edit `/etc/apparmor.d/usr.sbin.slapd` adding:

```
  /var/lib/ldap/accesslog/ r,
  /var/lib/ldap/accesslog/** rwk,
```

Then create the directory, reload the apparmor profile, and copy the `DB_CONFIG` file:

```
sudo -u openldap mkdir /var/lib/ldap/accesslog
sudo -u openldap cp /var/lib/ldap/DB_CONFIG /var/lib/ldap/accesslog/
sudo /etc/init.d/apparmor reload
```

Using the *-u openldap* option with the sudo commands above removes the need to adjust permissions for the new directory later.

3. Edit the file and change the *olcRootDN* to match your directory:

```
olcRootDN: cn=admin,dc=example,dc=com
```

4. Next, add the LDIF file using the ldapadd utility:

```
sudo ldapadd -Y EXTERNAL -H ldapi:/// -f provider_sync.ldif
```

5. Restart slapd:

```
sudo /etc/init.d/slapd restart
```

The *Provider* server is now configured, and it is time to configure a *Consumer* server.

1.4.2. Consumer Configuration

1. On the *Consumer* server configure it the same as the *Provider* except for the *Syncrepl* configuration steps.

   Add the additional schema files:

```
sudo ldapadd -Y EXTERNAL -H ldapi:/// -f /etc/ldap/schema/cosine.ldif
sudo ldapadd -Y EXTERNAL -H ldapi:/// -f /etc/ldap/schema/nis.ldif
sudo ldapadd -Y EXTERNAL -H ldapi:/// -f /etc/ldap/schema/inetorgperson.ldif
```

   Also, create, or copy from the provider server, the `backend.example.com.ldif`

```
# Load dynamic backend modules
dn: cn=module,cn=config
objectClass: olcModuleList
cn: module
olcModulepath: /usr/lib/ldap
olcModuleload: back_hdb

# Database settings
```

```
dn: olcDatabase=hdb,cn=config
objectClass: olcDatabaseConfig
objectClass: olcHdbConfig
olcDatabase: {1}hdb
olcSuffix: dc=example,dc=com
olcDbDirectory: /var/lib/ldap
olcRootDN: cn=admin,dc=example,dc=com
olcRootPW: secret
olcDbConfig: set_cachesize 0 2097152 0
olcDbConfig: set_lk_max_objects 1500
olcDbConfig: set_lk_max_locks 1500
olcDbConfig: set_lk_max_lockers 1500
olcDbIndex: objectClass eq
olcLastMod: TRUE
olcDbCheckpoint: 512 30
olcAccess: to attrs=userPassword by dn="cn=admin,dc=example,dc=com" write by anonymous auth by
olcAccess: to attrs=shadowLastChange by self write by * read
olcAccess: to dn.base="" by * read
olcAccess: to * by dn="cn=admin,dc=example,dc=com" write by * read
```

And add the LDIF by entering:

```
sudo ldapadd -Y EXTERNAL -H ldapi:/// -f backend.example.com.ldif
```

2. Do the same with the `frontend.example.com.ldif` file listed above, and add it:

```
sudo ldapadd -x -D cn=admin,dc=example,dc=com -W -f frontend.example.com.ldif
```

The two severs should now have the same configuration except for the *Syncrepl* options.

3. Now create a file named `consumer_sync.ldif` containing:

```
#Load the syncprov module.
dn: cn=module{0},cn=config
changetype: modify
add: olcModuleLoad
olcModuleLoad: syncprov

# syncrepl specific indices
dn: olcDatabase={1}hdb,cn=config
changetype: modify
add: olcDbIndex
olcDbIndex: entryUUID eq
-
add: olcSyncRepl
olcSyncRepl: rid=0 provider=ldap://ldap01.example.com bindmethod=simple binddn="cn=admin,dc=exa
 credentials=secret searchbase="dc=example,dc=com" logbase="cn=accesslog"
 logfilter="(&(objectClass=auditWriteObject)(reqResult=0))" schemachecking=on
 type=refreshAndPersist retry="60 +" syncdata=accesslog
-
add: olcUpdateRef
```

```
olcUpdateRef: ldap://ldap01.example.com
```

You will probably want to change the following attributes:

- *ldap01.example.com* to your server's hostname.
- *binddn*
- *credentials*
- *searchbase*
- *olcUpdateRef:*

4. Add the LDIF file to the configuration tree:

```
sudo ldapadd -c -Y EXTERNAL -H ldapi:/// -f consumer_sync.ldif
```

The frontend database should now sync between servers. You can add additional servers using the steps above as the need arises.

Il demone slapd invia, in modo predefinito, informazioni di registro a `/var/log/syslog`. Se qualche cosa *non* dovesse funzionare controllare quel file per eventuali errori e informazioni su come poterli risolvere. Inoltre, assicurarsi che tutti i server possano raggiungere quel FQDN (Fully Qualified Domain Name). Questo viene configurato nel file `/etc/hosts` in questo modo:

```
127.0.0.1 ldap01.example.com ldap01
```

.

1.5. Impostare ACL

L'autenticazione richiede accesso al campo della password che non dovrebbe essere accessibile in modo predefinito. Inoltre, affinché gli utenti possa cambiare la loro password usando **passwd** o altre utilità, *shadowLastChange* deve essere accessibile una volta che l'utente si è autenticato.

To view the Access Control List (ACL) for the *cn=config* tree, use the ldapsearch utility:

```
sudo ldapsearch -c -Y EXTERNAL -H ldapi:///  -LLL -b cn=config olcDatabase=config olcAccess
```

```
SASL/EXTERNAL authentication started
SASL username: gidNumber=0+uidNumber=0,cn=peercred,cn=external,cn=auth
SASL SSF: 0
dn: olcDatabase={0}config,cn=config
olcAccess: {0}to * by dn.exact=gidNumber=0+uidNumber=0,cn=peercred,cn=external
 ,cn=auth manage by * break
```

To see the ACL for the frontend tree enter:

```
sudo ldapsearch -c -Y EXTERNAL -H ldapi:///  -LLL -b cn=config olcDatabase={1}hdb olcAccess
```

1.6. TLS e SSL

Durante la fase di autenticazione a un server OpenLDAP, è raccomandato usare una sessione cifrata. Questo può essere ottenuto usando TLS (Transport Layer Security) o SSL (Secure Sockets Layer).

The first step in the process is to obtain or create a *certificate*. Because slapd is compiled using the gnutls library, the certtool utility will be used to create certificates.

1. First, install gnutls-bin by entering the following in a terminal:

   ```
   sudo apt-get install gnutls-bin
   ```
2. Next, create a private key for the *Certificate Authority* (CA):

   ```
   sudo sh -c "certtool --generate-privkey > /etc/ssl/private/cakey.pem"
   ```
3. Create a `/etc/ssl/ca.info` details file to self-sign the CA certificate containing:

   ```
   cn = Example Company
   ca
   cert_signing_key
   ```
4. Now create the self-signed CA certificate:

   ```
   sudo certtool --generate-self-signed --load-privkey /etc/ssl/private/cakey.pem \
    --template  /etc/ssl/ca.info --outfile /etc/ssl/certs/cacert.pem
   ```
5. Make a private key for the server:

   ```
   sudo sh -c "certtool --generate-privkey > /etc/ssl/private/ldap01_slapd_key.pem"
   ```

   

   Replace *ldap01* in the filename with your server's hostname. Naming the certificate and key for the host and service that will be using them will help keep filenames and paths straight.
6. To sign the server's certificate with the CA, create the `/etc/ssl/ldap01.info` info file containing:

   ```
   organization = Example Company
   cn = ldap01.example.com
   tls_www_server
   encryption_key
   signing_key
   ```
7. Create the server's certificate:

   ```
   sudo certtool --generate-certificate --load-privkey /etc/ssl/private/ldap01_slapd_key.pem \
   ```

```
--load-ca-certificate /etc/ssl/certs/cacert.pem --load-ca-privkey /etc/ssl/private/cakey.pem \
--template /etc/ssl/ldap01.info --outfile /etc/ssl/certs/ldap01_slapd_cert.pem
```

Una volta ottenuto un certificato, la chiave e si è installato il tutto, usare ldapmodify per aggiungere le nuove opzioni di configurazione:

```
sudo ldapmodify -Y EXTERNAL -H ldapi:///
```

```
Enter LDAP Password:
dn: cn=config
add: olcTLSCACertificateFile
olcTLSCACertificateFile: /etc/ssl/certs/cacert.pem
-
add: olcTLSCertificateFile
olcTLSCertificateFile: /etc/ssl/certs/ldap01_slapd_cert.pem
-
add: olcTLSCertificateKeyFile
olcTLSCertificateKeyFile: /etc/ssl/private/ldap01_slapd_key.pem

modifying entry "cn=config"
```

Adjust the `ldap01_slapd_cert.pem`, `ldap01_slapd_key.pem`, and `cacert.pem` names if yours are different.

Aprire il file `/etc/default/slapd` e togliere il commento dall'opzione *SLAPD_SERVICES*:

```
SLAPD_SERVICES="ldap:/// ldapi:/// ldaps:///"
```

Garantire accesso al certificato all'utente *openldap*:

```
sudo adduser openldap ssl-cert
sudo chgrp ssl-cert /etc/ssl/private/ldap01_slapd_key.pem
sudo chmod g+r /etc/ssl/private/ldap01_slapd_key.pem
```

Se la directory `/etc/ssl/private` e il file `/etc/ssl/private/server.key` hanno permessi diversi, modificare i comandi secondo le proprie esigenze.

In fine, riavviare slapd:

```
sudo /etc/init.d/slapd restart
```

Il demone slapd dovrebbe ora essere in ascolto per le connessioni LDAPS e dovrebbe essere in grado di usare STARTTLS nella fase di autenticazione.

Se il server non dovesse avviarsi, controllare in /var/log/syslog. Se sono presenti degli errori come «main: TLS init def ctx failed: -1», potrebbe esserci un problema di configurazione.

Controllare che il certificato sia firmato dalla stessa autorità dei file configurati e che il gruppo ssl-cert abbia permessi di lettura sulla chiave privata.

1.6.1. Replicazione TLS

Se è stato impostato Syncrepl tra i server è utile cifrare il traffico di replicazione usando *TLS* (Transport Layer Security). Per maggiori informazioni su come impostare la replicazione, consultare *Sezione 1.4, «LDAP Replication» [62]*.

Assuming you have followed the above instructions and created a CA certificate and server certificate on the *Provider* server. Follow the following instructions to create a certificate and key for the *Consumer* server.

1. Create a new key for the Consumer server:

```
mkdir ldap02-ssl
cd ldap02-ssl
certtool --generate-privkey > ldap02_slapd_key.pem
```

Creating a new directory is not strictly necessary, but it will help keep things organized and make it easier to copy the files to the Consumer server.

2. Next, create an info file, `ldap02.info` for the Consumer server, changing the attributes to match your locality and server:

```
country = US
state = North Carolina
locality = Winston-Salem
organization = Example Company
cn = ldap02.salem.edu
tls_www_client
encryption_key
signing_key
```

3. Create the certificate:

```
sudo certtool --generate-certificate --load-privkey ldap02_slapd_key.pem \
 --load-ca-certificate /etc/ssl/certs/cacert.pem --load-ca-privkey /etc/ssl/private/cakey.pem \
 --template ldap02.info --outfile ldap02_slapd_cert.pem
```

4. Copy the `cacert.pem` to the directory:

```
cp /etc/ssl/certs/cacert.pem .
```

5. The only thing left is to copy the `ldap02-ssl` directory to the Consumer server, then copy `ldap02_slapd_cert.pem` and `cacert.pem` to `/etc/ssl/certs`, and copy `ldap02_slapd_key.pem` to `/etc/ssl/private`.

6. Once the files are in place adjust the *cn=config* tree by entering:

```
sudo ldapmodify -Y EXTERNAL -H ldapi:///

Enter LDAP Password:
dn: cn=config
add: olcTLSCACertificateFile
olcTLSCACertificateFile: /etc/ssl/certs/cacert.pem
-
add: olcTLSCertificateFile
olcTLSCertificateFile: /etc/ssl/certs/ldap02_slapd_cert.pem
-
add: olcTLSCertificateKeyFile
olcTLSCertificateKeyFile: /etc/ssl/private/ldap02_slapd_key.pem

modifying entry "cn=config"
```

7. As with the Provider you can now edit `/etc/default/slapd` and add the *ldaps:///* parameter to the *SLAPD_SERVICES* option.

Now that *TLS* has been setup on each server, once again modify the *Consumer* server's *cn=config* tree by entering the following in a terminal:

```
sudo ldapmodify -Y EXTERNAL -H ldapi:///

SASL/EXTERNAL authentication started
SASL username: gidNumber=0+uidNumber=0,cn=peercred,cn=external,cn=auth
SASL SSF: 0

dn: olcDatabase={1}hdb,cn=config
replace: olcSyncrepl
olcSyncrepl: {0}rid=0 provider=ldap://ldap01.example.com bindmethod=simple binddn="cn=ad
 min,dc=example,dc=com" credentials=secret searchbase="dc=example,dc=com" logbas
 e="cn=accesslog" logfilter="(&(objectClass=auditWriteObject)(reqResult=0))" s
 chemachecking=on type=refreshAndPersist retry="60 +" syncdata=accesslog starttls=yes

modifying entry "olcDatabase={1}hdb,cn=config"
```

Se il nome host del server LDAP non corrisponde al FQDN (Fully Qualified Domain Name) nel certificato, potrebbe essere necessario modificare il file `/etc/ldap/ldap.conf` e aggiungere le seguenti opzioni TLS:

```
TLS_CERT /etc/ssl/certs/ldap02_slapd_cert.pem
TLS_KEY /etc/ssl/private/ldap02_slapd_key.pem
TLS_CACERT /etc/ssl/certs/cacert.pem
```

Riavviare slapd su ogni server:

```
sudo /etc/init.d/slapd restart
```

1.7. Autenticazione LDAP

Una volta ottenuto un server LDAP funzionante, i pacchetti auth-client-config e libnss-ldap consentono di configurare facilmente un client Ubuntu affinché utilizzi l'autenticazione LDAP. Per installare questi pacchetti, digitare:

```
sudo apt-get install libnss-ldap
```

Durante la fase di installazione un dialogo richiederà i dettagli relativi alla connessione al server LDAP.

Nel caso si commetta un errore durante l'inserimento delle informazioni, è possibile rieseguire la configurazione tramite il comando:

```
sudo dpkg-reconfigure ldap-auth-config
```

I risultati della configurazione possono essere visualizzati nel file `/etc/ldap.conf`. Se il server richiede delle opzioni non contemplate durante la fase di configurazione. modificare il file secondo le proprie esigenze.

Ora che libnss-ldap è configurato, abilitare il profilo auth-client-config LDAP digitando:

```
sudo auth-client-config -t nss -p lac_ldap
```

- *-t*: modifica solamente `/etc/nsswitch.conf`.
- *-p*: nome del profilo da abilitare, disabilitare, ecc...
- *lac_ldap*: il profilo auth-client-config parte del pacchetto ldap-auth-config.

Utilizzando l'utilità pam-auth-update, configurare il sistema affinché utilizzi LDAP per l'autenticazione:

```
sudo pam-auth-update
```

Dal menù pam-auth-update, scegliere LDAP e qualsiasi altro metodo di autenticazione necessario.

Ora dovrebbe essere possibile eseguire l'accesso utilizzando le credenziali presenti nella directory LDAP.

Se LDAP viene utilizzato per archiviare utenti Samba, è necessario configurare il server affinché utilizzi l'autenticazione via LDAP. Per maggiori informazioni, consultare *Sezione 2, «Samba e LDAP» [75]*.

1.8. Gestire utenti e gruppi

Il pacchetto ldap-utils contiene diverse utilità per la gestione di directory, ma le molte opzioni necessarie possono rendere queste utilità di difficile utilizzo. Il pacchetto ldapscripts contiene degli script configurabili per gestire facilmente utenti e gruppi LDAP.

Per installare il pacchetto, da un terminale:

```
sudo apt-get install ldapscripts
```

Aprire il file `/etc/ldapscripts/ldapscripts.conf` e togliere il commento o aggiungere quanto segue in base alle proprie esigenze:

```
SERVER=localhost
BINDDN='cn=admin,dc=example,dc=com'
BINDPWDFILE="/etc/ldapscripts/ldapscripts.passwd"
SUFFIX='dc=example,dc=com'
GSUFFIX='ou=Groups'
USUFFIX='ou=People'
MSUFFIX='ou=Computers'
GIDSTART=10000
UIDSTART=10000
MIDSTART=10000
```

Creare il file `ldapscripts.passwd` per consentire l'accesso autenticato alla directory:

```
sudo sh -c "echo -n 'secret' > /etc/ldapscripts/ldapscripts.passwd"
sudo chmod 400 /etc/ldapscripts/ldapscripts.passwd
```

Sostituire «secret» con la password dell'amministratore LDAP.

Le applicazioni in ldapscripts sono ora pronte per la gestione delle directory. Quelli che seguono sono degli esempi sull'utilizzo di questi script:

- Creare un nuovo utente:

  ```
  sudo ldapadduser mario example
  ```

  Viene creato un utente con UID *mario* e imposta il gruppo primario (GID) dell'utente a *example*

- Cambiare la password di un utente:

  ```
  sudo ldapsetpasswd mario
  Changing password for user uid=mario,ou=People,dc=example,dc=com
  New Password:
  New Password (verify):
  ```

- Eliminare un utente:

  ```
  sudo ldapdeleteuser mario
  ```

- Aggiungere un gruppo:

  ```
  sudo ldapaddgroup qa
  ```

- Eliminare un gruppo:

```
sudo ldapdeletegroup qa
```

- Aggiungere un utente a un gruppo:

```
sudo ldapaddusertogroup george qa
```

  Dovrebbe essere possibile visualizzare un attributo *memberUid* per il gruppo *qa* con un valore di *mario*.

- Rimuovere un utente da un gruppo:

```
sudo ldapdeleteuserfromgroup george qa
```

  L'attributo *memberUid* dovrebbe ora essere rimosso dal gruppo *qa*.

- Lo script ldapmodifyuser consente di aggiungere, rimuovere o replicare gli attributi di un utente. Lo script utilizza la stessa sintassi dell'utilità ldapmodify. Per esempio:

```
sudo ldapmodifyuser george
# About to modify the following entry :
dn: uid=george,ou=People,dc=example,dc=com
objectClass: account
objectClass: posixAccount
cn: george
uid: george
uidNumber: 1001
gidNumber: 1001
homeDirectory: /home/george
loginShell: /bin/bash
gecos: george
description: User account
userPassword:: e1NTSEF9eXFsTFcyWlhwWkF1eGUybVdFWHZKRzJVMjFTSG9vcHk=

# Enter your modifications here, end with CTRL-D.
dn: uid=george,ou=People,dc=example,dc=com
replace: gecos
gecos: Mario Rossi
```

  L'utente *gecos* dovrebbe ora essere «Mario Rossi».

- Un'altra utile caratteristica di ldapscripts è il sistema dei modelli. I modelli consentono di personalizzare gli attributi di un utente, gruppo e degli oggetti macchina. Per esempio, per abilitare il modello *user* aprire il file `/etc/ldapscripts/ldapscripts.conf` modificando:

```
UTEMPLATE="/etc/ldapscripts/ldapadduser.template"
```

  Diversi *esempi* sono disponibili nella directory `/etc/ldapscripts`. Copiare o rinominare il file `ldapadduser.template.sample` in `/etc/ldapscripts/ldapadduser.template`:

```
sudo cp /usr/share/doc/ldapscripts/examples/ldapadduser.template.sample /etc/ldapscripts/ldapaddu
```

Modificare il modello per aggiungere gli attributi desiderati. In questo esempio viene creato un nuovo utente con una *objectClass* di *inetOrgPerson*:

```
dn: uid=<user>,<usuffix>,<suffix>
objectClass: inetOrgPerson
objectClass: posixAccount
cn: <user>
sn: <ask>
uid: <user>
uidNumber: <uid>
gidNumber: <gid>
homeDirectory: <home>
loginShell: <shell>
gecos: <user>
description: User account
title: Employee
```

Notice the <*ask*> option used for the *ssn* value. Using <ask> will configure ldapadduser to prompt you for the attribute value during user creation.

Sono presenti molti altri script nel pacchetto. Per un elenco completo usare il comando: **dpkg -L ldapscripts | grep bin**

1.9. Risorse

- The *OpenLDAP Ubuntu Wiki*[1] page has more details.
- Per maggiori informazioni, consultare *il sito web di OpenLDAP*[2]
- Anche se un po' datata, un'ottima fonte di informazioni riguardo LDAP è *LDAP System Administration*[3] di O'Reilly.
- Il libro di Packt, *Mastering OpenLDAP*[4], è un'ottima fonte che copre anche le nuove versioni di OpenLDAP.
- Per maggiori informazioni su auth-client-config, consultare la pagina di manuale: **man auth-client-config**.
- Per maggiori informazioni riguardo il pacchetto ldapscripts, consultare le pagine di manuale: **man ldapscripts**, **man ldapadduser**, **man ldapaddgroup**, ecc...

2. Samba e LDAP

Questa sezione descrive come configurare Samba affinché utilizzi LDAP per informazioni sugli account e autenticazione di utenti, gruppi e computer. Si presume sia disponibile una directory OpenLDAP installata e funzionante e che il server sia configurato per l'utilizzo dell'autenticazione. Per maggiori informazioni sulla configurazione di OpenLDAP, consultare *Sezione 1, «Server OpenLDAP» [56]* e *Sezione 1.7, «Autenticazione LDAP» [71]*; per l'installazione e la configurazione di Samba, consultare *Capitolo 17, Reti Windows [224]*.

2.1. Installazione

Sono necessari tre pacchetti per interagire con Samba attraverso LDAP: samba, samba-doc e smbldap-tools. Per installarli, digitare:

```
sudo apt-get install samba samba-doc smbldap-tools
```

Il pacchetto smbldap-tools non è necessario, ma a meno di non avere un altro pacchetto o degli script personalizzati, è necessario disporre di un metodo per la gestione di utenti, gruppi e account.

2.2. Configurare OpenLDAP

Affinché Samba possa usare OpenLDAP come un *backend passdb*, gli oggetti utente nella directory necessitano di ulteriori attributi. Questa sezione assume che si voglia configurare Samba come controller di dominio Windows NT e verranno aggiunti gli oggetti e gli attributi LDAP necessari.

- gli attributi Samba sono definiti nel file `samba.schema`, parte del pacchetto samba-doc. Il file schema necessita di essere decompresso e copiato in `/etc/ldap/schema`. Da un terminale, digitare:

  ```
  sudo cp /usr/share/doc/samba-doc/examples/LDAP/samba.schema.gz /etc/ldap/schema/
  sudo gzip -d /etc/ldap/schema/samba.schema.gz
  ```

- È necessario aggiungere lo schema *samba* all'albero *cn=config*. La procedura per aggiungere un nuovo schema a slapd è spiegata in *Sezione 1.3, «Further Configuration» [59]*.

  1. Creare un file di configurazione chiamato `schema_convert.conf`, o simile, contenente quanto segue:

     ```
     include /etc/ldap/schema/core.schema
     include /etc/ldap/schema/collective.schema
     include /etc/ldap/schema/corba.schema
     include /etc/ldap/schema/cosine.schema
     include /etc/ldap/schema/duaconf.schema
     include /etc/ldap/schema/dyngroup.schema
     include /etc/ldap/schema/inetorgperson.schema
     include /etc/ldap/schema/java.schema
     include /etc/ldap/schema/misc.schema
     include /etc/ldap/schema/nis.schema
     ```

```
include /etc/ldap/schema/openldap.schema
include /etc/ldap/schema/ppolicy.schema
include /etc/ldap/schema/samba.schema
```

2. Creare una directory temporanea in cui salvare l'output:

```
mkdir /tmp/ldif_output
```

3. Usare quindi slapcat per convertire i file schema:

```
slapcat -f schema_convert.conf -F /tmp/ldif_output -n0 -s "cn={12}samba,cn=schema,cn=config"
```

Modificare i percorsi e i nomi dei file in base alla proprie esigenze.

4. Edit the generated `/tmp/cn\=samba.ldif` file by removing *{XX}* at the top of the file, where *"{XX}"* is the index number in curly braces:

```
dn: cn=samba,cn=schema,cn=config
...
cn: samba
```

Rimuovere le seguenti righe dalla fine del file:

```
structuralObjectClass: olcSchemaConfig
entryUUID: b53b75ca-083f-102d-9fff-2f64fd123c95
creatorsName: cn=config
createTimestamp: 20080827045234Z
entryCSN: 20080827045234.341425Z#000000#000#000000
modifiersName: cn=config
modifyTimestamp: 20080827045234Z
```

I valori degli attributi possono variare, basta solo assicurarsi che gli attributi siano rimossi.

5. In fine, usando l'utilità ldapadd, aggiugnere il nuovo schema alla directory:

```
ldapadd -x -D cn=admin,cn=config -W -f /tmp/cn\=samba.ldif
```

Dovrebbe ora esserci una voce *dn: cn={X}misc,cn=schema,cn=config* nell'albero "cn=config", in cui "X" è lo schema successivo in sequenza.

- Copiare e incollare quanto segue in un file chiamato `samba_indexes.ldif`:

```
dn: olcDatabase={1}hdb,cn=config
changetype: modify
add: olcDbIndex
olcDbIndex: uidNumber eq
olcDbIndex: gidNumber eq
olcDbIndex: loginShell eq
olcDbIndex: uid eq,pres,sub
```

```
olcDbIndex: memberUid eq,pres,sub
olcDbIndex: uniqueMember eq,pres
olcDbIndex: sambaSID eq
olcDbIndex: sambaPrimaryGroupSID eq
olcDbIndex: sambaGroupType eq
olcDbIndex: sambaSIDList eq
olcDbIndex: sambaDomainName eq
olcDbIndex: default sub
```

Usando l'utilità ldapmodify, caricare i nuovi indici:

```
ldapmodify -x -D cn=admin,cn=config -W -f samba_indexes.ldif
```

Se tutto è andato a buon fine, dovrebbe essere possibile visualizzare gli indici utilizzando ldapsearch:

```
ldapsearch -xLLL -D cn=admin,cn=config -x -b cn=config -W olcDatabase={1}hdb
```

- Ora, configurare il pacchetto smbldap-tools in base al proprio ambiente di lavoro. Il pacchetto è dotato di uno script di configurazione che richiede l'impostazione delle opzioni necessarie. Per eseguire lo script:

```
sudo gzip -d /usr/share/doc/smbldap-tools/configure.pl.gz
sudo perl /usr/share/doc/smbldap-tools/configure.pl
```

Una volta risposto a tutte le domande, dovrebbero esserci dei file `/etc/smbldap-tools/smbldap.conf` e `/etc/smbldap-tools/smbldap_bind.conf`. Questi sono generati dallo script di configurazione e nel caso siano stati commessi degli errori durante l'esecuzione dello script è possibile aprirli e modificarli.

- Lo script smbldap-populate aggiunge gli utenti, i gruppi e gli oggetti LDAP necessari a Samba. È utile creare un file LDIF di backup con slapcat prima di eseguire il comando:

```
sudo slapcat -l backup.ldif
```

- Ottenuta la copia di sicurezza, eseguire smbldap-populate digitando:

```
sudo smbldap-populate
```

È possibile creare un file LDIF contenente i nuovi oggetti Samba eseguendo il comando **sudo smbldap-populate -e samba.ldif**. In questo modo, è possibile visualizzare le modifiche per assicurarsi che tutto sia corretto.

La directory LDAP ora ha le informazioni necessarie sul dominio per autenticare gli utenti Samba.

2.3. Configurare Samba

Sono disponibili diversi metodi per configurare Samba, per maggiori informazioni consultare *Capitolo 17, Reti Windows [224]*. Per configurare Samba all'uso di LDAP, modificare il file di configurazione principale di Samba, `/etc/samba/smb.conf`, commentando l'opzione *passdb backend* e aggiungendo quanto segue:

```
#   passdb backend = tdbsam

# LDAP Settings
   passdb backend = ldapsam:ldap://hostname
   ldap suffix = dc=example,dc=com
   ldap user suffix = ou=People
   ldap group suffix = ou=Groups
   ldap machine suffix = ou=Computers
   ldap idmap suffix = ou=Idmap
   ldap admin dn = cn=admin,dc=example,dc=com
   ldap ssl = start tls
   ldap passwd sync = yes
...
   add machine script = sudo /usr/sbin/smbldap-useradd -t 0 -w "%u"
```

Riavviare samba per abilitare le nuove impostazioni:

```
sudo restart smbd
sudo restart nmbd
```

Samba ora necessita di conoscere la password di amministrazione di LDAP. Da un terminale, digitare:

```
sudo smbpasswd -w secret
```

Sostituire *secret* con la password di amministrazione di LDAP.

Se sono già presenti degli utenti in LDAP e si vuole che possano autenticarsi attraverso Samba, è necessario che abbiano degli attributi come definiti nel file `samba.schema`. Aggiungere gli attributi Samba agli utenti esistenti usando l'utilità smbpasswd, sostituendo *NOME_UTENTE* con un nome utente reale:

```
sudo smbpasswd -a NOME_UTENTE
```

Viene chiesta la password dell'utente.

Per aggiungere un nuovo utente, gruppo o account macchina, usare le utilità dal pacchetto smbldap-tools. Alcuni esempi:

- Per aggiungere un nuovo utente a LDAP con attributi Samba, digitare quanto segue, sostituendo "NOME_UTENTE" con un nome utente reale:

```
sudo smbldap-useradd -a -P NOME_UTENTE
```

L'opzione *-a* aggiunge gli attributi Samba, *-P* chiama l'utilità smbldap-passwd dopo aver creato l'utente consentendo di inserire la password per l'utente.

- Per rimuovere un utente dalla directory:

```
sudo smbldap-userdel NOME_UTENTE
```

L'utilità smbldap-userdel è dotata anche di un'opzione *-r* per rimuovere la directory home dell'utente.

- Per aggiungere un gruppo, usare smbldap-groupadd, sostituendo "NOME_GRUPPO" con il nome di un gruppo esistente:

```
sudo smbldap-groupadd -a NOME_GRUPPO
```

Come per smbldap-useradd, l'opzione *-a* aggiunge gli attributi Samba.

- Per aggiungere un utente a un gruppo, usare smbldap-groupmod:

```
sudo smbldap-groupmod -m NOME_UTENTE NOME_GRUPPO
```

Assicurarsi di sostituire *NOME_UTENTE* con un utente reale. Inoltre, con l'opzione *-m* è possibile aggiungere più di un utente alla volta, elencandoli come valori *separati da virgola*.

- smbldap-groupmod può essere usato anche per rimuovere un utente da un gruppo:

```
sudo smbldap-groupmod -x NOME_UTENTE NOME_GRPPO
```

- L'utilità smbldap-useradd può anche aggiungere degli account macchina:

```
sudo smbldap-useradd -t 0 -w NOME_UTENTE
```

Sostituire *NOME_UTENTE* con il nome della workstation. L'opzione *-t 0* crea un account macchina immediatamente, *-w* indica di creare l'utente come account macchina. Notare che l'opzione *add machine script* in `/etc/samba/smb.conf` è stata modificata per usare smbldap-useradd.

Sono disponibili molte altre utilità nel pacchetto smbldap-tools. Per maggiori informazioni, consultare la pagina di manuale.

2.4. Risorse

- Sono disponibili anche diversi documenti riguardo LDAP e Samba, come nella «*Samba HOWTO Collection*[5]».
- In particolare, consultare la sezione *passdb*[6].
- Another good site is *Samba OpenLDAP HOWTO*[7].

- Inoltre, per maggiori informazioni riguardo smbldap-tools consultare le pagina man: **man smbldap-useradd**, **man smbldap-groupadd**, **man smbldap-populate**, ecc...
- Also, there is a list of *Ubuntu wiki*[8] articles with more information.

3. Kerberos

Kerberos è un sistema di autenticazione di rete basato sul principio di un "agente" terzo fidato. Le altre due parti sono l'utente e il servizio a cui l'utente vuole autenticarsi. Non tutti i serivizi e le applicazioni possono usare Kerberos, ma quelle che ne sono in grado, consentono di portare la rete a essere un SSO (Single Sign On).

Questa sezione spiega come installare e configurare un server Kerberos, fornendo alcuni esempi di configurazione.

3.1. Panoramica

Se si è nuovi di Kerberos, ci sono alcuni termini che è bene comprendere prima di procedere. Molti di questi termini potrebbero essere simili ad altri concetti di altri ambienti più familiari.

- *Principal*: qualsiasi utente, computer e servizio fornito da server deve essere difinito come "Kerberos Principal".
- *Istanze*: usate dai principal di servizio e da quelli amministrativi.
- *Realms*: il "reame" di controllo fornito dall'installazione di Kerberos. Di solito il dominio DNS convertito in maiuscolo (EXAMPLE.IT).
- *Key Distribution Center* (KDC): consiste di tre parti, un database di tutti i principal, il server di autenticazione e il server che garantisce i ticket. Per ogni reame deve esserci almeno un KDC.
- *Ticket Granting Ticket* (TGT): emesso dallo "Authentication Server" (AS), il "Ticket Granting Ticket" è cifrato con la password dell'utente ed è quindi noto solo all'utente e al KDC.
- *Ticket Granting Server* (TGS): emette i ticket su richiesta dei client.
- *Ticket:*: conferma l'identità dei due principal. Uno è l'utente e l'altro il servizio richiesto. Il ticket stabilisce una chiave di cifratura usata per garantire la sicurezza della comunicazione durante la fase di autenticazione.
- *File keytab*: sono file estratti dal KDC e contengono le chiavi di cifratura per un servizio o un host.

Per riassumenre, un reame ha almeno un KDC, preferibilmente due per ridondanza, che contiene un database di principal. Quando un utente accede in una workstation configurata per l'autenticazione Kerberos, il KDC emette un TGT (Ticket Granting Ticket). Se le informazioni fornite dall'utente corrispondono, l'utente viene autenticato e può richiedere i ticket per i servizi Kerberos da un TGS (Ticket Granting Server). I ticket consentono all'utente di autenticarsi al servizio senza dover fornire nome utente e password.

3.2. Server Kerberos

3.2.1. Installazione

Prima di installare il server Kerberos, è necessario disporre di un server DNS configurato per il proprio dominio. Dato che il reame Kerberos corrisponde al dominio, questa sezione utilizza il dominio *example.com* configurato in *Sezione 2.3, «Server primario» [97]*.

Kerberos, inoltre, è un protocollo basato sul tempo. Se l'ora locale tra il client e il server differisce di più di 5 minuti, le workstation non potranno autenticarsi. Per correggere questo problema, tutti gli host dovrebbe sincronizzare il proprio orario usando il protocollo NTP (*Network Time Protocol*). Per maggiori informazioni, consultare *Sezione 4, «Sincronizzazione del tempo con NTP» [46]*.

Il primo passo nell'installare Kerberos consiste nell'installare i pacchetti krb5-kdc e krb5-admin-server. In un terminale, digitare:

```
sudo apt-get install krb5-kdc krb5-admin-server
```

Alla fine dell'installazione viene chiesto di fornire un nome per i server Kerberos e Admin del reame, che potrebbero essere anche lo stesso server.

Creare il reame con l'utilità kdb5_newrealm:

```
sudo krb5_newrealm
```

3.2.2. Configurazione

Le domande poste durate l'installazione sono usate per impostare il file `/etc/krb5.conf`. Per modificare la configurazione del KDC, modificare il file e riavviare il demone krb5-kdc.

1. Ora che il KDC è in esecuzione, è necessario avere un utente amministratore. È raccomandato usare un nome utente diverso da quello usato giornalmente per le normali operazioni al computer. Usando l'utilità kadmin.local da un terminale:

   ```
   sudo kadmin.local
   Authenticating as principal root/admin@EXAMPLE.COM with password.
   kadmin.local: addprinc steve/admin
   WARNING: no policy specified for steve/admin@EXAMPLE.COM; defaulting to no policy
   Enter password for principal "steve/admin@EXAMPLE.COM":
   Re-enter password for principal "steve/admin@EXAMPLE.COM":
   Principal "steve/admin@EXAMPLE.COM" created.
   kadmin.local: quit
   ```

   Nell'esempio precedente *steve* è il *Principal*, */admin* è una *Instance* e *@EXAMPLE.COM* indica il realm. Il Principal *"giornaliero"* è *steve@EXAMPLE.COM* e dovrebbe avere i diritti di un utente normale.

   

   Sostituire *EXAMPLE.COM* e *steve* con il proprio reame e il nome utente dell'amministratore.

2. Il nuovo utente amministratore necessita dei permessi ACL (Access Control List) corretti, configurati tramite il file `/etc/krb5kdc/kadm5.acl`:

   ```
   steve/admin@EXAMPLE.COM        *
   ```

Questa voce garantisce a *steve/admin* la possibilità di eseguire qualsiasi operazione su tutti i principal nel reame.

3. Riavviare krb5-admin-server affinché le nuove ACL abbiano effetto:

```
sudo /etc/init.d/krb5-admin-server restart
```

4. Il nuovo utente può essere provato con l'utilità kinit:

```
kinit steve/admin
steve/admin@EXAMPLE.COM's Password:
```

Una volta inserita la password, usare l'utilità klist per visualizzare le informazioni riguardo il TGT (Ticket Granting Ticket):

```
klist
Credentials cache: FILE:/tmp/krb5cc_1000
        Principal: steve/admin@EXAMPLE.COM

  Issued           Expires          Principal
Jul 13 17:53:34  Jul 14 03:53:34  krbtgt/EXAMPLE.COM@EXAMPLE.COM
```

Potrebbe essere necessario aggiungere una voce nel file `/etc/hosts` per il KDC. Per esempio:

```
192.168.0.1   kdc01.example.com       kdc01
```

Sostituire *192.168.0.1* con l'indirizzo del KDC.

5. Affinché i client possano determinare il corretto KDC per il reame, sono necessari dei record DNS SRV. Aggiungere quanto segue al file `/etc/named/db.example.com`:

```
_kerberos._udp.EXAMPLE.COM.     IN SRV 1  0 88  kdc01.example.com.
_kerberos._tcp.EXAMPLE.COM.     IN SRV 1  0 88  kdc01.example.com.
_kerberos._udp.EXAMPLE.COM.     IN SRV 10 0 88  kdc02.example.com.
_kerberos._tcp.EXAMPLE.COM.     IN SRV 10 0 88  kdc02.example.com.
_kerberos-adm._tcp.EXAMPLE.COM. IN SRV 1  0 749 kdc01.example.com.
_kpasswd._udp.EXAMPLE.COM.      IN SRV 1  0 464 kdc01.example.com.
```

Sostituire *EXAMPLE.COM*, *kdc01* e *kdc02* con il nome del proprio dominio, il KDC primario e quello secondario.

Consultare *Capitolo 7, DNS (Domain Name Service) [94]* per le istruzioni sulla configurazione di DNS.

Il reame Kerberos è ora pronto per autenticare i client.

3.3. KDC secondario

Una volta ottenuto un KDC all'interno della rete, è utile avere anche un KDC secondario nel caso in cui quello primario non fosse più disponibile.

1. Per prima cosa installare il pacchetto e quando vengono chiesti i nomi di Kerberos e Admin, inserire il nome del KDC primario:

   ```
   sudo apt-get install krb5-kdc krb5-admin-server
   ```

2. Una volta installato il pacchetto, creare il KDC secondario. Da un terminale, digitare:

   ```
   kadmin -q "addprinc -randkey host/kdc02.example.com"
   ```

   

   Una volta eseguiti i comandi kadmin viene chiesto la propria password *NOME_UTENTE/ADMIN@EXAMPLE.COM*.

3. Estrarre il file *keytab_*

   ```
   kadmin -q "ktadd -k keytab.kdc02 host/kdc02.example.com"
   ```

4. Dovrebbe esserci un file `keytab.kdc02` nella directory corrente, spostare il file in `/etc/krb5.keytab`:

   ```
   sudo mv keytab.kdc02 /etc/krb5.keytab
   ```

   

   Se il percorso a `keytab.kdc02` è diverso, modificarlo in base al proprio caso.

   È possibile elencare tutti i principal presenti in un file Keytab, utile durante la risoluzione dei problemi, con l'utilità klist:

   ```
   sudo klist -k /etc/krb5.keytab
   ```

5. Dovrebbe esserci un file `kpropd.acl` in ogni KDC che presenti tutti i KDC del reame. Per esempio, sia sul KDC primario che secondario, creare un file `/etc/krb5kdc/kpropd.acl`:

   ```
   host/kdc01.example.com@EXAMPLE.COM
   host/kdc02.example.com@EXAMPLE.COM
   ```

6. Creare un database vuoto nel *KDC secondario*:

   ```
   sudo kdb5_util -s create
   ```

7. Avviare il demone kpropd che resterà in ascolto per le connessioni dall'utilità kprop. kprop è usato per trasferire i file di dump:

   ```
   sudo kpropd -S
   ```

8. Da un terminale dal *KDC primario*, creare un file di dump del database principale:

```
sudo kdb5_util dump /var/lib/krb5kdc/dump
```

9. Estrarre il *keytab* del KDC primario e copiarlo in `/etc/krb5.keytab`:

```
kadmin -q "ktadd -k keytab.kdc01 host/kdc01.example.com"
sudo mv keytab.kdc01 /etc/krb5.keytab
```

Assicurarsi che ci sia un *host* per *kdc01.example.com* prima di estrarre il keytab.

10. Usando l'utilità kprop eseguire il push del database sul KDC secondario:

```
sudo kprop -r EXAMPLE.COM -f /var/lib/krb5kdc/dump kdc02.example.com
```

Dovrebbe essere visualizzato un messaggio di *SUCCEEDED* se la propagazione è andata a buon fine. Se si è verificato un errore, per maggiori informazioni, controllare `/var/log/syslog` sul KDC secondario.

Potrebbe esser utile creare anche un'attività cron per aggiornare periodicamente il database sul KDC secondario. Per esempio, il comando seguente esegue il push del database ogni ora:

```
# m h  dom mon dow   command
0 * * * * /usr/sbin/kdb5_util dump /var/lib/krb5kdc/dump && /usr/sbin/kprop -r EXAMPLE.COM -f /
```

11. Sempre nel *KDC secondario*, creare un file *stash* in cui salvare la chiave principale di Kerberos:

```
sudo kdb5_util stash
```

12. Avviare il demone krb5-kdc sul KDC secondario:

```
sudo /etc/init.d/krb5-kdc start
```

Il *KDC secondario* dovrebbe ora essere in grado di emettere i ticket per il reame. È possibile verificare ciò fermando il demone krb5-kdc sul KDC primario e usando kinit per richiedere un ticket. Se tutto funziona correttamente, si dovrebbe ricevere un ticket dal KDC secondario.

3.4. Client Kerberos Linux

Questa sezione spiega come configurare un sistema Linux come un client Kerberos consentendo l'accesso a qualsiasi servizio Kerberos ad accesso effettuato correttamente da parte degli utenti.

3.4.1. Installazione

Per autenticarsi in un reame Kerberos sono necessari i pacchetti krb5-user e libpam-krb5 oltre ad altri non strettamente necessari, ma che semplificano molto la gestione. Per installare questi pacchetti, digitare:

```
sudo apt-get install krb5-user libpam-krb5 libpam-ccreds auth-client-config
```

Il pacchetto auth-client-config consente una semplice configurazione dell'autenticazione PAM per diverse sorgenti e libpam-ccreds memorizza le credenziali di autenticazione consentendo di effettuare l'accesso anche se il KDC non è disponibile. Questo pacchetto è utile anche per i computer portatili che possono autenticarsi su reti aziendali, ma devono essere in grado di farlo anche al di fuori della rete.

3.4.2. Configurazione

Per configurare il client, in un terminale digitare:

```
sudo dpkg-reconfigure krb5-config
```

Viene quindi chiesto di inserire il nome del reame Kerberos. Inoltre, se non si dispone di un DNS configurato con i record *SRV* di Kerberos, viene richiesto il nome dell'host del KDC e del server amministrativo.

Il comando dpkg-reconfigure aggiunge delle voci al file `/etc/krb5.conf` del proprio reame. Dovrebbero essere disponibili delle voci simili alle seguenti:

```
[libdefaults]
        default_realm = EXAMPLE.COM
...
[realms]
        EXAMPLE.COM = }
                kdc = 192.168.0.1
                admin_server = 192.168.0.1
        }
```

Per avviare la configurazione, richiedere un ticket usando l'utilità kinit. Per esempio:

```
kinit steve@EXAMPLE.COM
Password for steve@EXAMPLE.COM:
```

Una volta ottenuto un ticket, i dettagli possono essere visualizzati usando klist:

```
klist
Ticket cache: FILE:/tmp/krb5cc_1000
Default principal: steve@EXAMPLE.COM

Valid starting     Expires            Service principal
07/24/08 05:18:56  07/24/08 15:18:56  krbtgt/EXAMPLE.COM@EXAMPLE.COM
        renew until 07/25/08 05:18:57


Kerberos 4 ticket cache: /tmp/tkt1000
```

```
klist: You have no tickets cached
```

Usare auth-client-config per configurare il modulo libpam-krb5 affinché richieda un ticket durante la fase di accesso:

```
sudo auth-client-config -a -p kerberos_example
```

Una volta autenticati con successo, si dovrebbe ricevere un ticket.

3.5. Risorse

- Per maggiori informazioni riguardo Kerberos, consultare il sito di *Kerberos presso il MIT*[9].
- The *Ubuntu Wiki Kerberos*[10] page has more details.
- Il libro *Kerberos: The Definitive Guide*[11] di O'Reilly è un ottimo punto di riferimento per impostare un server Kerberos.
- Inoltre, è possibile chiedere informazioni nel canale IRC *#ubuntu-server* su *Freenode*[12].

4. Kerberos e LDAP

Sostituire un database principale di Kerberos tra due server può essere complicato e aggiunge un ulteriori database all'interno della rete. Il server Kerberos può comunque essere configurato per utilizzare una directory LDAP come database principale. In questa sezione viene descritto come configurare un server Kerberos, primario e secondario, affinché utilizzi OpenLDAP come database principale.

4.1. Configurare OpenLDAP

Per prima cosa è necessario caricare lo *schema* all'interno del server OpenLDAP collegato ai KDC primario e secondario. I successivi passi qui descritti hanno come presupposto la presenza di un server LDAP di replica configurato tra due server. Per maggiori informazioni su come impostare un server OpenLDAP, consultare *Sezione 1, «Server OpenLDAP» [56]*.

È inoltre richiesto per configurare OpenLDAP all'uso di connessioni TLS e SSL, in modo che il traffico tra il KDC e il server LDAP sia cifrato. Per maggiori informazioni, consultare *Sezione 1.6, «TLS e SSL» [67]*.

- Per caricare lo schema all'interno del server LDAP, installare su tale server il pacchetto krb5-kdc-ldap. Da un terminale, digitare:

  ```
  sudo apt-get install krb5-kdc-ldap
  ```

- Estrarre il file `kerberos.schema.gz`:

  ```
  sudo gzip -d /usr/share/doc/krb5-kdc-ldap/kerberos.schema.gz
  sudo cp /usr/share/doc/krb5-kdc-ldap/kerberos.schema /etc/ldap/schema/
  ```

- Lo schema *kerberos* deve essere aggiunto all'albero *cn=config*. La procedura per aggiungere un nuovo schema a slapd è descritta anche in *Sezione 1.3, «Further Configuration» [59]*.

  1. Creare un file di configurazione chiamato `schema_convert.conf`, o simile, contenente quanto segue:

     ```
     include /etc/ldap/schema/core.schema
     include /etc/ldap/schema/collective.schema
     include /etc/ldap/schema/corba.schema
     include /etc/ldap/schema/cosine.schema
     include /etc/ldap/schema/duaconf.schema
     include /etc/ldap/schema/dyngroup.schema
     include /etc/ldap/schema/inetorgperson.schema
     include /etc/ldap/schema/java.schema
     include /etc/ldap/schema/misc.schema
     include /etc/ldap/schema/nis.schema
     include /etc/ldap/schema/openldap.schema
     include /etc/ldap/schema/ppolicy.schema
     include /etc/ldap/schema/kerberos.schema
     ```

2. Creare una directory temporanea in cui salvare i file LDIF:

```
mkdir /tmp/ldif_output
```

3. Usare quindi slapcat per convertire i file schema:

```
slapcat -f schema_convert.conf -F /tmp/ldif_output -n0 -s "cn={12}kerberos,cn=schema,cn=confi
```

Modificare i percorsi e i nomi dei file in base alla proprie esigenze.

4. Modificare il file `/tmp/cn\=kerberos.ldif` generato sistemando i seguenti attributi:

```
dn: cn=kerberos,cn=schema,cn=config
...
cn: kerberos
```

Rimuovere le seguenti righe dalla fine del file:

```
structuralObjectClass: olcSchemaConfig
entryUUID: 18ccd010-746b-102d-9fbe-3760cca765dc
creatorsName: cn=config
createTimestamp: 20090111203515Z
entryCSN: 20090111203515.326445Z#000000#000#000000
modifiersName: cn=config
modifyTimestamp: 20090111203515Z
```

I valori degli attributi possono variare, basta solo assicurarsi che gli attributi siano rimossi.

5. Caricare il nuovo schema con ldapadd:

```
ldapadd -x -D cn=admin,cn=config -W -f /tmp/cn\=kerberos.ldif
```

6. Aggiungere un indice per l'attributo *krb5principalname*:

```
ldapmodify -x -D cn=admin,cn=config -W
Enter LDAP Password:
dn: olcDatabase={1}hdb,cn=config
add: olcDbIndex
olcDbIndex: krbPrincipalName eq,pres,sub

modifying entry "olcDatabase={1}hdb,cn=config"
```

7. Infine, aggiornare le ACL (Access Control Lists):

```
ldapmodify -x -D cn=admin,cn=config -W
Enter LDAP Password:
dn: olcDatabase={1}hdb,cn=config
replace: olcAccess
```

```
olcAccess: to attrs=userPassword,shadowLastChange,krbPrincipalKey by dn="cn=admin,dc=exampl
 e,dc=com" write by anonymous auth by self write by * none
-
add: olcAccess
olcAccess: to dn.base="" by * read
-
add: olcAccess
olcAccess: to * by dn="cn=admin,dc=example,dc=com" write by * read

modifying entry "olcDatabase={1}hdb,cn=config"
```

La directory LDAP è ora pronta come database principale per Kerberos.

4.2. Configurazione KDC primario

Configurato OpenLDAP, è necessario configurare KDC.

- Installare i pacchetti necessari. In un terminale, digitare:

```
sudo apt-get install krb5-kdc krb5-admin-server krb5-kdc-ldap
```

- Modificare `/etc/krb5.conf` aggiungendo le seguenti opzioni all'interno delle sezioni appropriate:

```
[libdefaults]
        default_realm = EXAMPLE.COM

...

[realms]
        EXAMPLE.COM = {
                kdc = kdc01.example.com
                kdc = kdc02.example.com
                admin_server = kdc01.example.com
                admin_server = kdc02.example.com
                default_domain = example.com
                database_module = openldap_ldapconf
        }

...

[domain_realm]
        .example.com = EXAMPLE.COM


...

[dbdefaults]
        ldap_kerberos_container_dn = dc=example,dc=com

[dbmodules]
        openldap_ldapconf = {
```

```
        db_library = kldap
        ldap_kdc_dn = "cn=admin,dc=example,dc=com"

        # this object needs to have read rights on
        # the realm container, principal container and realm sub-trees
        ldap_kadmind_dn = "cn=admin,dc=example,dc=com"

        # this object needs to have read and write rights on
        # the realm container, principal container and realm sub-trees
        ldap_service_password_file = /etc/krb5kdc/service.keyfile
        ldap_servers = ldaps://ldap01.example.com ldaps://ldap02.example.com
        ldap_conns_per_server = 5
}
```

Modificare *example.com*, *dc=example,dc=com*, *cn=admin,dc=example,dc=com* e *ldap01.example.com* con i valori corretti del dominio, dell'oggetto e del server LDAP della propria rete.

- Usare l'utilità kdb5_ldap_util per creare il reame:

```
sudo kdb5_ldap_util -D cn=admin,dc=example,dc=com create -subtrees dc=example,dc=com -r EXAMPLE.C
```

- Creare un file stash della password utilizzata per l'associazione al server LDAP. Questa password è usata con le opzioni *ldap_kdc_dn* e *ldap_kadmin_dn* nel file `/etc/krb5.conf`:

```
sudo kdb5_ldap_util -D cn=admin,dc=example,dc=com stashsrvpw -f /etc/krb5kdc/service.keyfile cn=a
```

- Copiare il certificato della CA dal server LDAP:

```
scp ldap01:/etc/ssl/certs/cacert.pem .
sudo cp cacert.pem /etc/ssl/certs
```

Modificare il file `/etc/ldap/ldap.conf` affinché utilizzi il certificato:

```
TLS_CACERT /etc/ssl/certs/cacert.pem
```

Il certificato deve anche essere copiato nel KDC secondario per consentire la connessione ai server LDAP utilizzando LDAPS.

Ora è possibile aggiungere i principal Kerberos al database LDAP che verranno copiati su tutti gli altri server LDAP di replica. Per aggiungere un principal utilizzando l'utilità kadmin.local, digitare:

```
sudo kadmin.local
Authenticating as principal root/admin@EXAMPLE.COM with password.
kadmin.local:  addprinc -x dn="uid=steve,ou=people,dc=example,dc=com" steve
WARNING: no policy specified for steve@EXAMPLE.COM; defaulting to no policy
Enter password for principal "steve@EXAMPLE.COM":
Re-enter password for principal "steve@EXAMPLE.COM":
Principal "steve@EXAMPLE.COM" created.
```

Dovrebbero ora essere aggiunti all'oggetto utente *uid=steve,ou=people,dc=example,dc=com* gli attributi krbPrincipalName, krbPrincipalKey, krbLastPwdChange e krbExtraData. Per verificare che all'utente venga emesso un ticket, utilizzare le utilità kinit e klist.

Se l'oggetto utente è già stato creato, è necessario usare l'opzione *-x dn="..."* per aggiungere gli attributi Kerberos, altrimenti verrà creato un nuovo oggetto *principal* nel sottoalbero del reame.

4.3. Configurazione KDC secondario

La configurazione di un KDC secondario utilizzando il backend LDAP è molto simile alla configurazione tramite l'utilizzo del database Kerberos.

- Installare i pacchetti necessari. In un terminale digitare:

```
sudo apt-get install krb5-kdc krb5-admin-server krb5-kdc-ldap
```

- Modificare il file `/etc/krb5.conf` affinché utilizzi il backend LDAP:

```
[libdefaults]
        default_realm = EXAMPLE.COM

...

[realms]
        EXAMPLE.COM = {
                kdc = kdc01.example.com
                kdc = kdc02.example.com
                admin_server = kdc01.example.com
                admin_server = kdc02.example.com
                default_domain = example.com
                database_module = openldap_ldapconf
        }

...

[domain_realm]
        .example.com = EXAMPLE.COM

...

[dbdefaults]
        ldap_kerberos_container_dn = dc=example,dc=com

[dbmodules]
        openldap_ldapconf = {
                db_library = kldap
                ldap_kdc_dn = "cn=admin,dc=example,dc=com"

                # this object needs to have read rights on
```

```
                # the realm container, principal container and realm sub-trees
                ldap_kadmind_dn = "cn=admin,dc=example,dc=com"

                # this object needs to have read and write rights on
                # the realm container, principal container and realm sub-trees
                ldap_service_password_file = /etc/krb5kdc/service.keyfile
                ldap_servers = ldaps://ldap01.example.com ldaps://ldap02.example.com
                ldap_conns_per_server = 5
        }
```

- Creare il file stash per la password di associazione LDAP:

```
sudo kdb5_ldap_util -D cn=admin,dc=example,dc=com stashsrvpw -f /etc/krb5kdc/service.keyfile cn=a
```

- Dal *KDC primario*, copiare il file di stash della *chiave primaria* (`/etc/krb5kdc/.k5.EXAMPLE.COM`) nel KDC secondario. Accertarsi di copiare tale file utilizzando una connessione cifrata come scp o su un supporto fisico.

```
sudo scp /etc/krb5kdc/.k5.EXAMPLE.COM steve@kdc02.example.com:~
sudo mv .k5.EXAMPLE.COM /etc/krb5kdc/
```

Ricordarsi di sostituire *EXAMPLE.COM* con il reame in uso.

- Infine. avviare il demone krb5-kdc:

```
sudo /etc/init.d/krb5-kdc start
```

All'interno della propria rete sono quindi disponibili dei KDC ridondanti che assieme ai server LDAP ridondanti permettono l'autenticazione degli utenti anche nel caso in cui un server LDAP, un server Kerberos o uno server LDAP e un server Kerberos non siano più disponibili.

4.4. Risorse

- Maggiori informazioni possono essere trovate nella *Kerberos Admin Guide*[13].
- For more information on kdb5_ldap_util see *Section 5.6*[14] and the *kdb5_ldap_util man page*[15].
- Another useful link is the *krb5.conf man page*[16].
- Also, see the *Kerberos and LDAP*[17] Ubuntu wiki page.

Capitolo 7. DNS (Domain Name Service)

Il DNS (Domain Name Service) è un servizio Internet che mappa gli indirizzi IP e i nomi di dominio univoci (FQDN) tra di loro facendo in modo di non dover ricordare gli indirizzi IP. I computer che eseguono DNS sono chiamati *server dei nomi*. Ubuntu è dotato di BIND (Berkley Internet Naming Daemon), il più diffuso programma usato per mantenere un server dei nomi su Linux.

1. Installazione

A un promt di terminale, inserire il seguente comando per installare dns:

```
sudo apt-get install bind9
```

Un pacchetto molto utile per eseguire test e risolvere i problemi di DNS è "dnsutils". Per installare dnsutils digitare quanto segue:

```
sudo apt-get install dnsutils
```

2. Configurazione

BIND9 può essere configurato in diversi modi tra cui: come cache per server dei nomi, master principale e master secondario.

- Quando configurato come un server dei nomi cache, BIND9 troverà la risposta alle interrogazioni sui nomi e la archivierà.
- Come server primario, BIND9 legge i dati per una zona da un file ed è autoritativo per quella zona.
- Nella configurazione come server secondario, BIN9 ottiene i dati della zona da un altro server dei nomi per quella zona.

2.1. Panoramica

I file di configurazione di DNS sono archiviati nella directory `/etc/bind`, il file di configurazione principale è `/etc/bind/named.conf`.

La riga *include* specifica il nome del file contenente le opzioni DNS, la riga *directory* nel file `/etc/bind/named.conf.options` indica a DNS dove cercare i file. Tutti i file usati da BIND sono presenti in questa directory.

Il file `/etc/bind/db.root` descrive i server dei nomi "radice" nel mondo. Questi server cambiano col tempo, quindi il file `/etc/bind/db.root` deve essere aggiornato ogni tanto, procedura che viene svolta, solitamente, con gli aggiornamenti al pacchetto bind9. La sezione *zone* definisce un server principale ed è archiviata in un file indicato dall'opzione *file*.

È possibile configurare lo stesso server sia come server dei nomi cache, master primario e secondario. Un server può ricoprire il ruolo di "Start of Authority" (SOA) per una zona, fornendo allo stesso tempo servizi di server secondario per un'altra zona e di cache per gli host della LAN.

2.2. Server dei nomi cache

La configurazione predefinita comporta l'utilizzo come server di cache. È necessario solamente aggiungere gli indirizzi IP dei server DNS del proprio ISP. De-commentare e modificare quanto segue nel file `/etc/bind/named.conf.options`:

```
forwarders {
               1.2.3.4;
               5.6.7.8;
          };
```

Sostituire *1.2.3.4* e *5.6.7.8* con gli indirizzi IP del server di nomi attuale.

Per abilitare la nuova configurazione è necessario riavviare il server DNS. Da un terminale, digitare:

```
sudo /etc/init.d/bind9 restart
```

Per maggiori informazioni su come eseguire test su un server cache DNS, consultare *Sezione 3.1.2, «dig» [101]*.

2.3. Server primario

In questa sezione, BIND9 viene configurato come server primario per il dominio *example.com*. Basta sostituire *example.com* con il proprio FQDN (Fully Qualified Domain Name).

2.3.1. File zona forward

Per aggiungere una zona DNS a BIND9, trasformando BIND9 in un server primario, la prima cosa da fare è modificare il file `/etc/bind/named.conf.local`:

```
zone "example.com" {
 type master;
        file "/etc/bind/db.example.com";
};
```

Prendere un file zona esistente come modello per creare il file `/etc/bind/db.example.com`:

```
sudo cp /etc/bind/db.local /etc/bind/db.example.com
```

Modificare il file `/etc/bind/db.example.com` cambiando *localhost.* nel FQDN del proprio server, lasciando il "." alla fine. Modificare *127.0.0.1* con l'indirizzo IP del server di nomi e *root.localhost* con un indirizzo email valido, ma con un "." al posto del simbolo "@", anche in questo caso lasciando il "." alla fine.

Inoltre, creare una voce *A* per *ns.example.com*. Il server dei nomi in questo esempio:

```
;
; BIND data file for local loopback interface
;
$TTL    604800
@       IN      SOA     ns.example.com. root.example.com. (
                              2         ; Serial
                         604800         ; Refresh
                          86400         ; Retry
                        2419200         ; Expire
                         604800 )       ; Negative Cache TTL
;
@       IN      NS      ns.example.com.
@       IN      A       127.0.0.1
@       IN      AAAA    ::1
ns      IN      A       192.168.1.10
```

È necessario incrementare il numero *Serial* ogni volta che vengono apportate modifiche al file zona. Se vengono eseguite molteplici modifiche prima di riavviare BIND, incrementare il valore solo una volta.

Ora è possibile aggiungere voci DNS alla fine del file zona. Per maggiori informazioni, consultare la *Sezione 4.1, «Tipi di record comuni» [105]*.

Molti amministratori usano come valore per "Serial" la data dell'ultima modifica, come *2007010100* in cui si ha AAAAMMGGSS (dove *SS* è il valore "Serial").

Modificato il file zona, è necessario riavviare BIND9 affinché le modiche vengano applicate.

```
sudo /etc/init.d/bind9 restart
```

2.3.2. File zona reverse

Una volta configurata la zona e la risoluzione dei nomi con un indirizzo IP, è necessaria anche una zona *Reverse*. Una zona "Reverse" consente a DNS di trasformare un indirizzo in un nome.

Modificare il file /etc/bind/named.conf.local aggiungendo quanto segue:

```
zone "1.168.192.in-addr.arpa" {
        type master;
        notify no;
        file "/etc/bind/db.192";
};
```

Sostituire *1.168.192* con i primi tre valori dell'indirizzo della rete che si sta usando. Inoltre, chiamare il file zona `/etc/bind/db.192` in modo appropriato, in modo tale che rispecchi il primo ottetto della propria rete.

Creare il file `/etc/bind/db.192`:

```
sudo cp /etc/bind/db.127 /etc/bind/db.192
```

Quindi modificare `/etc/bind/db.192` cambiando le stesse opzioni di `/etc/bind/db.example.com`:

```
;
; BIND reverse data file for local loopback interface
;
$TTL    604800
@       IN      SOA     ns.example.com. root.example.com. (
                              2         ; Serial
                         604800         ; Refresh
                          86400         ; Retry
                        2419200         ; Expire
                         604800 )       ; Negative Cache TTL
```

```
;
@       IN      NS      ns.
10      IN      PTR     ns.example.com.
```

The *Serial Number* in the Reverse zone needs to be incremented on each change as well. For each *A record* you configure in `/etc/bind/db.example.com` you need to create a *PTR record* in `/etc/bind/db.192`.

Dopo aver creato il file zona "reverse", riavviare BIND9:

```
sudo /etc/init.d/bind9 restart
```

2.4. Server secondario

Una volta configurato un *server primario*, un *server secondario* è necessario per mantenere la disponibilità del dominio nel caso in cui quello primario non fosse più disponibile.

Per prima cosa, nel server primario ("Primary Master"), deve essere consentita la zona "transfer". Aggiungere l'opzione *allow-transfer* alle definizioni delle zone "Forward" e "Reverse" in `/etc/bind/named.conf.local`:

```
zone "example.com" {
        type master;
 file "/etc/bind/db.example.com";
        allow-transfer { 192.168.1.11; };
};

zone "1.168.192.in-addr.arpa" {
        type master;
        notify no;
        file "/etc/bind/db.192";
 allow-transfer { 192.168.1.11; };
};
```

Sostituire *192.168.1.11* con l'indirizzo IP del server di nomi secondario.

Quindi, in quello secondario ("Secondary Master"), installare il pacchetto bind9 come fatto per il server primario, quindi modificare il file `/etc/bind/named.conf.local` e aggiungere le seguetni dichiarazioni per le zone "Forward" e "Reverse":

```
zone "example.com" {
 type slave;
        file "db.example.com";
        masters { 192.168.1.10; };
};

zone "1.168.192.in-addr.arpa" {
```

```
 type slave;
        file "db.192";
        masters { 192.168.1.10; };
};
```

Sostituire *192.168.1.10* con l'indirizzo IP del server dei nomi primario.

Riavviare BIND9 nel server secondario:

```
sudo /etc/init.d/bind9 restart
```

In `/var/log/syslog` si dovrebbe vedere qualche cosa:

```
slave zone "example.com" (IN) loaded (serial 6)
slave zone "100.18.172.in-addr.arpa" (IN) loaded (serial 3)
```

Una zona è trasferita solamente se *Serial Number* del server primario è maggiore di quello del server secondario.

The default directory for non-authoritative zone files is `/var/cache/bind/`. This directory is also configured in AppArmor to allow the named daemon to write to it. For more information on AppArmor see *Sezione 4, «AppArmor» [121]*.

3. Risoluzione problemi

Questa sezione descrive i metodi per determinare le cause dei problemi che si possono verificare con DNS e BIND9.

3.1. Test

3.1.1. resolv.conf

Il primo passo per verificare BIND9 consiste nell'aggiungere l'indirizzo IP del server di nomi in un risolutore di host. Il server dei nomi primario dovrebbe essere configurato così come un altro host per verificare il tutto. Modificare il file `/etc/resolv.conf` e aggiungere quanto segue:

```
nameserver 192.168.1.10
nameserver 192.168.1.11
```

Potrebbe essere necessario aggiungere anche l'indirizzo IP del server di nomi secondario nel caso in cui il primario non fosse più disponibile.

3.1.2. dig

Se è stato installato il pacchetto dnsutils, è possibile configurare l'utilità di ricerca DNS dig:

- Una volta installato BIND9 usare dig sull'interfaccia di loopback per assicurarsi che sia in ascolto sulla porta 53. Da un terminale digitare:

  ```
  dig -x 127.0.0.1
  ```

  L'output del comando dovrebbe essere simile al seguente:

  ```
  ;; Query time: 1 msec
  ;; SERVER: 192.168.1.10#53(192.168.1.10)
  ```

- Se BIND9 è stato configurato come un server di *cache*, eseguire "dig" su un dominio esterno per verificare il tempo dell'interrogazione:

  ```
  dig ubuntu.com
  ```

  Prestare attenzione al tempo dell'interrogazione verso la fine dell'output:

  ```
  ;; Query time: 49 msec
  ```

  Dopo una seconda esecuzione del comando si dovrebebro vedere dei miglioramenti:

  ```
  ;; Query time: 1 msec
  ```

3.1.3. ping

Per dimostrare come le applicazioni utilizzino i DNS per interpretare un nome host, usare l'utilità ping per inviare una richiesta eco ICMP. Da un terminale digitare:

```
ping example.com
```

In questo modo si verifica che il server dei nome sia in grado di interpretare il nome *ns.example.com* in un indirizzo IP. L'output del comando dovrebbe essere simile a quanto segue:

```
PING ns.example.com (192.168.1.10) 56(84) bytes of data.
64 bytes from 192.168.1.10: icmp_seq=1 ttl=64 time=0.800 ms
64 bytes from 192.168.1.10: icmp_seq=2 ttl=64 time=0.813 ms
```

3.1.4. named-checkzone

Un ottimo modo per provare i propri file zona consiste nell'usare l'utilità named-checkzone installata con il pacchetto bind9. Questa utilità consente di verificare che la configurazione sia corretta prima di riavviare BIND9 e consentendo di apportare delle modifiche.

- Per provare il file zona "Forward", in un terminale, digitare quanto segue:

  ```
  named-checkzone example.com /etc/bind/db.example.com
  ```

  Se tutto è stato configurato correttamente, si dovrebbe vedere un output simile a questo:

  ```
  zone example.com/IN: loaded serial 6
  OK
  ```

- Analogamente, per verificare il file zona "Reverse", digitare quanto segue:

  ```
  named-checkzone example.com /etc/bind/db.192
  ```

  L'output dovrebbe essere simile a quanto segue:

  ```
  zone example.com/IN: loaded serial 3
  OK
  ```

Il valore *Serial* del proprio file zona probabilmente sarà diverso.

3.2. Registrazione

BIND9 dispone di diverse configurazione per la registrazione degli eventi. Le due opzioni principali sono: *channel* che configura dove vengono salvate le registrazioni e l'opzione *category* che determina quali informazioni registrare.

Se non viene configurata alcuna opzione di registrazione, quella predefinita è:

```
logging {
     category default { default_syslog; default_debug; };
     category unmatched { null; };
};
```

Questa sezione descrive come configurare BIND9 affinché invii i messaggi di *debug* relativi alle interrogazioni DNS in un file diverso.

- Per prima cosa è necessario configurare un canale per specificare quale a quale file inviare i messaggi. Modificare quindi il file `/etc/bind/named.conf.local` e aggiungere quanto segue:

  ```
  logging {
      channel query.log {
          file "/var/log/query.log";
          severity debug 3;
      };
  };
  ```

- Configurare una categoria per inviare tutte le interrogazioni DNS al file:

  ```
  logging {
      channel query.log {
          file "/var/log/query.log";
          severity debug 3;
      };
      category queries { query.log; };
  };
  ```

  

  L'opzione *debug* può essere impostata tra 1 e 3. Se non viene specificato alcun livello, viene considerato quello predefinito, cioè 1.

- Dato che il demone *named* viene eseguito come l'utente *bind*, è necessario creare il file `/var/log/query.log` e modificarne il proprietario:

  ```
  sudo touch /var/log/query.log
  sudo chown bind /var/log/query.log
  ```

- Prima che il demone named possa scrivere nel nuovo file di registrazione, il profilo AppArmor deve esser aggiorato. Per prima cosa modificare `/etc/apparmor.d/usr.sbin.named` e aggiungere:

  ```
  /var/log/query.log w,
  ```

  Quindi ricaricare il profilo:

  ```
  cat /etc/apparmor.d/usr.sbin.named | sudo apparmor_parser -r
  ```

  Per maggiori informazioni riguardo AppArmor, consultare la *Sezione 4, «AppArmor» [121]*.

- Riavviare BIND9 affinché le modifiche abbiano effetto:

```
sudo /etc/init.d/bind9 restart
```

Dovrebbe essere possibile vedere il file `/var/log/query.log` riempirsi con le informazioni relative alle interrogazioni. Per maggio informazioni sulle opzioni di registrazione di BIND9, consultare la *Sezione 4.2, «Ulteriori informazioni» [105]*.

4. Riferimenti

4.1. Tipi di record comuni

Questa sezione descrive i più comuni tipi di record DNS.

- Record *A*: mappa un indirizzo IP con un nome host.

```
www      IN    A      192.168.1.12
```

- Record *CNAME*: usato per creare un alias di un record "A" esistente. Non è possibile creare un record CNAME che punti a un altro record CNAME.

```
web     IN    CNAME  www
```

- Record *MX*: usato per definire dove dovrebbero essere inviate le email. Deve puntare a un record A, non a uno CNAME.

```
        IN    MX  1   mail.example.com.
mail    IN    A       192.168.1.13
```

- Record *NS*: usato per definire quali server dispongono di copie di una zona. Deve puntare a un record A, non a un CNAME. Qui vengono definiti i server primario e secondario.

```
        IN    NS     ns.example.com.
 IN    NS     ns2.example.com.
ns      IN    A      192.168.1.10
ns2 IN    A      192.168.1.11
```

4.2. Ulteriori informazioni

- Il *DNS HOWTO*[1] dispone di maggiori informazioni sulla configurazione di BIND9.
- Per un approfondimento di *DNS* e BIND9, consultare *Bind9.net*[2].
- *DNS and BIND*[3] è un libro molto comune giunto ormai alla quinta edizione.
- Un ottimo posto per richiedere assistenza riguardo BIND9, e per partecipare nella comunità di Ubuntu Server, è il canale IRC *#ubuntu-server* su *freenode*[4].
- Consultare anche la *documentazione di bind9 online*[5].

Capitolo 8. Sicurezza

La sicurezza deve essere sempre considerata come uno degli aspetti più importanti durante l'installazione, lo sviluppo e l'uso di un sistema. Anche se un'installazione base di Ubuntu offre un livello di sicurezza sufficientemente elevato per l'utilizzo immediato su Internet, è importante avere una buona conoscenza della sicurezza del proprio sistema in base a come verrà usato in produzione.

This chapter provides an overview of security related topics as they pertain to Ubuntu 10.10 Server Edition, and outlines simple measures you may use to protect your server and network from any number of potential security threats.

1. Gestione utenti

L'amministrazione degli utenti è una parte critica per il mantenimento di un sistema sicuro. Utenti poco esperti con privilegi di amministrazione spesso sono la causa della compromissione di sistemi. Pertanto, è importante capire come proteggere il proprio server tramite delle semplici ed efficaci tecniche di gestione degli account utente.

1.1. Dove è l'utente root?

Gli sviluppatori di Ubuntu hanno deciso di disattivare in modo predefinito l'account di amministrazione (root) in tutte le installazioni di Ubuntu. Questo non significa che l'account root sia stato eliminato o che non sia più accessibile, è stata impostata una password che non corrisponde ad alcun possibile valore codificato, pertanto, l'accesso come root non è direttamente possibile.

Gli utenti sono incoraggiati a utilizzare lo strumento sudo per svolgere i compiti di amministrazione di sistema. Lo strumento sudo permette a un utente autorizzato di elevare temporaneamente i propri privilegi usando la propria password, invece di dover conoscere direttamente la password di root. Questo semplice, ma efficace, metodo cerca di fornire responsabilità per tutte le azioni degli utenti e dà all'amministratore un controllo granulare sulle azioni che un utente può eseguire con tali privilegi.

- Se per qualche ragione è necessario abilitare l'account root, basta assegnargli semplicemente una password:

  ```
  sudo passwd
  ```

  Il programma «sudo» chiederà di inserire la propria password e successivamente di inserirne una nuova per l'account root:

  ```
  [sudo] password for NOME_UTENTE: (inserire la propria password)
  Inserire nuova password UNIX: (inserire una nuova password per root)
  Reinserire la nuova password UNIX: (reinserire la nuova password per root)
  passwd: password aggiornata correttamente
  ```

- Per disabilitare l'account root, utilizzare la seguente sintassi per passwd:

  ```
  sudo passwd -l root
  ```

- Per maggiori informazioni riguardo sudo, consultarne il manuale:

  ```
  man sudo
  ```

In modo predefinito, l'utente iniziale creato dall'installazione di Ubuntu è un membro del gruppo «admin» ed è stato aggiunto al file `/etc/sudoers` come utente autorizzato all'utilizzo di sudo. Per autorizzare altri utenti ai pieni poteri amministrativi di root attraverso l'uso del comando sudo, è sufficiente aggiungerli al gruppo «admin».

1.2. Aggiungere e rimuovere utenti

Il processo per la gestione di utenti e gruppi locali è molto intuitivo e differisce poco dalla maggior parte degli altri sistemi GNU/Linux. Ubuntu e altre distribuzioni basate su Debian, incoraggiano l'utilizzo del pacchetto «adduser» per la gestione degli utenti.

- Per aggiungere un nuovo utente, utilizzare i seguenti comandi e seguire le istruzioni per impostare all'account una password e fornire le caratteristiche identificabili come nome, cognome, numero di telefono, ecc...

  ```
  sudo adduser NOME_UTENTE
  ```

- Per eliminare un utente e il suo gruppo principale, digitare:

  ```
  sudo deluser NOME_UTENTE
  ```

  Quando si elimina un account utente non viene rimossa la sua cartella home. È decisione dell'amministratore se rimuoverla o no in base alle proprie scelte.

  Ricordare che, se non sono state prese le necessarie precauzioni, ogni nuovo utente aggiunto successivamente con gli stessi UID/GID del precedente proprietario della cartella, avrà accesso a tale cartella.

  È possibile modificare questi valori UID/GID con qualcosa di più appropriato, come per esempio l'account root, e spostare la cartella per evitare futuri conflitti:

  ```
  sudo chown -R root:root /home/NOME_UTENTE/
  sudo mkdir /home/archived_users/
  sudo mv /home/NOME_UTENTE /home/archived_users/
  ```

- Per bloccare o sbloccare temporaneamente l'account di un utente, utilizzare, rispettivamente, i seguenti comandi:

  ```
  sudo passwd -l NOME_UTENTE
  sudo passwd -u NOME_UTENTE
  ```

- Per aggiungere o rimuovere un gruppo personalizzato, utilizzare, rispettivamente, i seguenti comandi:

  ```
  sudo addgroup NOME_GRUPPO
  sudo delgroup NOME_GRUPPO
  ```

- Per aggiungere un utente a un gruppo, digitare:

  ```
  sudo adduser NOME_UTENTE NOME_GRUPPO
  ```

1.3. Sicurezza dei profili utente

Quando viene creato un nuovo utente, l'applicazione «adduser» crea una nuova directory chiamata `/home/NOME_UTENTE`. Il profilo predefinito è modellato secondo i contenuti presenti nella directory `/etc/skel` che contiene tutti i profili di base.

Se il proprio server ospiterà più utenti, è necessario prestare la massima attenzione alle autorizzazioni delle home degli utenti, al fine di garantirne la riservatezza. In modo predefinito, in Ubuntu, le home degli utenti sono create con permessi di lettura e di esecuzione per tutti gli utenti. Questo significa che tutti gli utenti possono visualizzare e accedere al contenuto delle home degli altri utenti, cosa che potrebbe non essere soddisfacente per il proprio ambiente.

- Per verificare i permessi attuali della home degli utenti, utilizzare il seguente comando:

  ```
  ls -ld /home/NOME_UTENTE
  ```

  Il seguente output mostra che la directory `/home/NOME_UTENTE` è accessibile in lettura da parte di tutti gli utenti:

  ```
  drwxr-xr-x 2 nomeutente nomeutente 4096 2007-10-02 20:03 nomeutente
  ```

- È possibile rimuovere il permesso in lettura da tutti con il seguente comando:

  ```
  sudo chmod 0750 /home/NOME_UTENTE
  ```

  

  Alcuni amministratori utilizzano anche l'opzione per la modifica ricorsiva (-R) di tutte le sotto-cartelle e file della home, ma questo non è necessario e potrebbe inoltre causare degli effetti indesiderati. Modificare i permessi alla cartella principale è più che sufficiente per prevenire degli accessi non autorizzati.

  Un modo più efficiente potrebbe essere quello di modificare direttamente le impostazioni predefinite dell'applicazione adduser sui permessi da assegnare alle home degli utenti appena creati. È sufficiente modificare la variabile `DIR_MODE`, nel file `/etc/adduser.conf`, secondo le proprie esigenze.

  ```
  DIR_MODE=0750
  ```

- Dopo aver corretto opportunamente i permessi di accesso alle directory home come descritto precedentemente, verificare il risultato con il seguente comando:

  ```
  ls -ld /home/NOME_UTENTE
  ```

  Il risultato qui sotto mostra come i permessi di lettura per tutti gli altri utenti siano stati rimossi:

  ```
  drwxr-x--- 2 nomeutente nomeutente 4096 2007-10-02 20:03 nomeutente
  ```

1.4. Politica delle password

Una severa politica delle password è uno dei più importanti aspetti della sicurezza di un sistema. Le più frequenti violazioni di un sistema avvengono tramite attacchi di forza bruta con degli elenchi di parole che statisticamente possono comprende delle parole chiavi utilizzate come password. Se si vuole di offrire un qualsiasi tipo di accesso remoto utilizzando la propria password locale, assicurarsi che la complessità della stessa superi dei limiti minimi di adeguatezza, di impostare delle password con durate massime e controllare frequentemente i propri sistemi di autenticazione.

1.4.1. Lungezza minima di una password

By default, Ubuntu requires a minimum password length of 6 characters, as well as some basic entropy checks. These values are controlled in the file `/etc/pam.d/common-password`, which is outlined below.

```
password        [success=2 default=ignore]      pam_unix.so obscure sha512
```

If you would like to adjust the minimum length to 8 characters, change the appropriate variable to min=8. The modification is outlined below.

```
password        [success=2 default=ignore]      pam_unix.so obscure sha512 min=8
```

1.4.2. Scadenza delle password

Quando vengono creati dei nuovi utenti è possibile impostare una durata minima e massima per le loro password, obbligando gli stessi a modificarla alla scadenza.

- Per visualizzare facilmente lo stato attuale di un account utente, utilizzare il seguente comando:

  `sudo chage -l NOME_UTENTE`

  L'output seguente mostra informazioni interessanti sull'account dell'utente, in particolare che non ci sono politiche applicate:

  ```
  Ultimo cambio della password : gen 20, 2008
  Scadenza della password : mai
  Inattività della password : mai
  Scadenza dell'account : mai
  Numero minimo di giorni tra i cambi di password : 0
  Numero massimo di giorni tra i cambi di password : 99999
  Giorni di preavviso prima della scadenza della password : 7
  ```

- Per impostare uno qualsiasi di questi campi, utilizzare il seguente comando e seguire le istruzioni:

  `sudo chage NOME_UTENTE`

Quello che segue è un esempio di come sia possibile modificare manualmente la data di scadenza dell'account (-E) al 31/01/2008 (inserirla nel formato mm/gg/aaaa o nel formato aaaa/mm/gg), l'età minima della password (-m) a 5 giorni, l'età massima (-M) a 90 giorni, il periodo di inattività (-I) a 5 giorni dopo la scadenza della password e un avvertimento (-W) di 14 giorni prima della scadenza delle password.

```
sudo chage -E 01/31/2011 -m 5 -M 90 -I 30 -W 14 username
```

- Per verificare le modifiche, utilizzare lo stesso comando di prima:

```
sudo chage -l NOME_UTENTE
```

Il seguente output mostra i cambiamenti effettuati sull'account:

```
Ultimo cambio della password : gen 20, 2008
Scadenza della password : apr 19, 2008
Inattività della password : mag 19, 2008
Scadenza dell'account : gen 31, 2008
Numero minimo di giorni tra i cambi di password : 5
Numero massimo di giorni tra i cambi di password : 90
Giorni di preavviso prima della scadenza della password : 14
```

1.5. Ulteriori considerazioni sulla sicurezza

Molte applicazioni usano meccanismi di autenticazione alternativi che possono essere facilmente trascurati anche da esperti amministratori di sistema. Pertanto, è importante comprendere e controllare come avviene l'autenticazione degli utenti e come accedono ai servizi e alle applicazioni sul proprio server.

1.5.1. Accesso SSH per gli utenti disabilitati

Disattivando o bloccando l'account di un utente non impedisce che quest'ultimo riesca a effettuare l'accesso al server se precedentemente utilizzava una chiave pubblica RSA; saranno ancora in grado di ottenere l'accesso al server senza la necessità della password. Controllare sempre se nella directory home degli utenti sono presenti dei file che permettano questo tipo di autenticazione SSH, come per esempio `/home/nomeutente/.ssh/authorized_keys`.

Eliminare o rinominare la directory `.ssh/` nella home degli utenti per prevenire future autenticazioni SSH.

Assicurarsi di controllare qualsiasi connessione SSH stabilita dagli utenti disabilitati, dato che potrebbero esserci connessioni aperti in entrata o in uscita. Terminare tutte quelle che vengono trovate.

Limitare l'accesso SSH solo agli utenti che ne hanno il diritto. Per esempio, è possibile creare un gruppo chiamato «sshlogin» e aggiungere il nome del gruppo alla voce `AllowGroupsvarname> nel file /etc/ssh/sshd_config.`

```
AllowGroups sshlogin
```

Dopo aver aggiunto gli utenti con diritto di accesso SSH al gruppo «sshlogin», riavviare il server SSH.

```
sudo adduser NOME_UTENTE sshlogin
sudo /etc/init.d/ssh restart
```

1.5.2. Autenticazione utenti su database esterno

La maggior parte delle reti aziendali richiedono un servizio di autenticazione e di controllo degli accessi centralizzato per tutte le risorse di sistema. Se il server è stato configurato per gestire l'autenticazione attraverso database esterni, assicurarsi di disabilitare gli account utente sia esternamente che internamente, in questo modo l'autenticazione locale di riserva non è più possibile.

2. Sicurezza della console

Come con qualsiasi sistema di protezione che viene usato per proteggere il proprio server. è molto difficile difendersi dai rischi imprevisti, causati da qualcuno con accesso fisico all'interno dell'ambiente di lavoro, come furti di dischi fissi, mancanza di corrente e via dicendo. Perciò, è necessario considerare la sicurezza della console come un componente della sicurezza totale del sistema. Una porta bloccata può fermare un malintenzionato o può rallentare un ladro esperto, ed è quindi utile prendere delle precauzioni anche a livello della console.

Le seguenti istruzioni consentiranno di proteggere il proprio server da problemi che potrebbero portare serie conseguenze.

2.1. Disabilitare il Ctrl+Alt+Canc

Qualsiasi persona con accesso fisico alla tastiera può semplicemente premere **Ctrl**+**Alt**+**Canc** per riavviare il server senza eseguire l'accesso. Qualcuno può sempre scollegare la presa della corrente, ma per lo meno è da evitare l'uso di questa combinazione di tasti su un server in produzione. In questo modo un malintenzionato è costretto a utilizzare altre strategie per riavviare un server e consente di non riavviarlo accidentalmente.

- To disable the reboot action taken by pressing the **Ctrl**+**Alt**+**Delete** key combination, comment out the following line in the file `/etc/init/control-alt-delete.conf`.

```
#exec shutdown -r now "Control-Alt-Delete pressed"
```

3. Firewall

3.1. Introduzione

Il kernel Linux include il sottosistema *Netfilter* usato per manipolare o decidere la sorte del traffico di rete diretto all'interno o attraverso un server. Tutte le moderne soluzioni firewall per Linux si basano su questo sistema di filtraggio dei pacchetti.

Il sistema di filtraggio dei pacchetti del kernel non è di grande utilità per gli amministratori senza un'interfaccia nello spazio utente per gestirlo. Questo è il compito di iptables. Quando un pacchetto raggiunge il proprio server, esso è gestito affidato al sottosistema Netfilter per l'accettazione, la manipolazione oppure il rifiuto secondo quanto stabilito da regole fornite al sottosistema dallo spazio utente attraverso iptables. Quindi, iptables è tutto ciò che è necessario per gestire il proprio firewall, a patto che si abbia la dimestichezza necessaria; sono comunque disponibili molte altre applicazioni per semplificare tale attività.

3.2. ufw - Firewall non complicato

L'applicazione predefinita in Ubuntu per la configurazione di un firewall è ufw. Sviluppato per semplificare la configurazione di iptables, ufw offre un modo semplice per creare un firewall basato su protocolli IPv4 e IPv6.

ufw, in modo predefinito, è inizialmente disabilitato. Dal manuale di ufw si legge:

«ufw is not intended to provide complete firewall functionality via its command interface, but instead provides an easy way to add or remove simple rules. It is currently mainly used for host-based firewalls (ufw non ha lo scopo di implementare tutte le funzionalità di un firewall tramite la sua interfaccia di comandi, ma invece cerca di facilitare l'aggiunta o la rimozione di semplici regole. È usato principalmente per dei firewall host-based)»

Seguono degli esempi sull'uso di ufw:

- Per prima cosa, è necessario abilitare ufw. In un terminale digitare:

```
sudo ufw enable
```

- Per aprire una porta (in questo caso la porta di SSH):

```
sudo ufw allow 22
```

- Le regole possono anche essere aggiunte usando un formato *a numeri*:

```
sudo ufw insert 1 allow 80
```

- Analogamente, per chiudere una porta aperta:

```
sudo ufw deny 22
```

- Per eliminare una regola, usare «delete» seguito dalla regola:

```
sudo ufw delete deny 22
```

- È anche possibile consentire l'accesso da host o da reti specifici a una porta. Il seguente esempio consente accesso SSH dall'host 192.168.0.2 a qualsiasi indirizzo IP su questo host:

```
sudo ufw allow proto tcp from 192.168.0.2 to any port 22
```

  Sostituire 192.168.0.2 con 192.168.0.0/24 per consentire accesso SSH da tutta la sotto-rete.

- Aggiungendo l'opzione *--dry-run* a un comando *ufw* è possibile visualizzare il risultato delle regole, ma senza applicarle. Per esempio, questo è quello che verrebbe applicato nel caso venisse aperta la porta HTTP:

```
sudo ufw --dry-run allow http
```

```
*filter
:ufw-user-input - [0:0]
:ufw-user-output - [0:0]
:ufw-user-forward - [0:0]
:ufw-user-limit - [0:0]
:ufw-user-limit-accept - [0:0]
### RULES ###

### tuple ### allow tcp 80 0.0.0.0/0 any 0.0.0.0/0
-A ufw-user-input -p tcp --dport 80 -j ACCEPT

### END RULES ###
-A ufw-user-input -j RETURN
-A ufw-user-output -j RETURN
-A ufw-user-forward -j RETURN
-A ufw-user-limit -m limit --limit 3/minute -j LOG --log-prefix "[UFW LIMIT]: "
-A ufw-user-limit -j REJECT
-A ufw-user-limit-accept -j ACCEPT
COMMIT
Rules updated
```

- È possibile disabilitare ufw con il comando:

```
sudo ufw disable
```

- Per visualizzare lo stato del firewall usare:

```
sudo ufw status
```

- Per informazioni più dettagliate usare:

```
sudo ufw status verbose
```

- Per visualizzare il formato *a numeri*:

```
sudo ufw status numbered
```

Se la porta che si vuole aprire o chiudere è definita in `/etc/services`, è possibile usare il nome della porta al posto del numero. In questo esempio si sostituisce *22* con *ssh*.

Questa è una breve introduzione all'utilizzo di ufw. Per maggiori informazioni, consultare le pagine man di ufw.

3.2.1. Integrazione delle applicazioni con ufw

Le applicazioni che aprono delle porte possono includere un profilo ufw in cui vengono descritte le porte necessarie all'applicazione per funzionare correttamente. I profili vengono salvati in `/etc/ufw/applications.d` e possono essere modificati se le porte predefinite sono cambiate.

- Per visualizzare quali applicazioni hanno un profilo installato, in un terminale digitare:

```
sudo ufw app list
```

- Usare un profilo di un'applicazione è simile al consentire il traffico attraverso una porta:

```
sudo ufw allow Samba
```

- È disponibile anche una sintassi più estesa:

```
ufw allow from 192.168.0.0/24 to any app Samba
```

Sostituire *Samba* e *192.168.0.0/24* con il profilo dell'applicazione da usare e l'intervallo di indirizzi della propria rete.

Non è necessario specificare il *protocollo* per l'applicazione, dato che queste informazioni sono contenute nel profilo. Notare che il nome dell'*applicazione* sostituisce il numero della *porta*.

- Per visualizzare i dettagli riguardo quali porte, protocolli, ecc... sono definiti per un'applicazione, digitare:

```
sudo ufw app info Samba
```

Non tutte le applicazioni che richiedono l'apertura di una porta hanno un profilo ufw, ma se è stato creato un profilo per un'applicazione e lo si vuole includere nel pacchetto, segnalare un bug su *Launchpad*[1].

[1] https://launchpad.net/

3.3. IP masquerading

Il compito dell'IP masquerading è di consentire a quei computer della rete forniti di indirizzi IP privati e non instradabili, di accedere a Internet tramite il computer che opera il masquerading. Il traffico che va dalla rete privata verso Internet deve essere manipolato per ottenere risposte che siano re-instradabili al computer che ne ha fatto richiesta. Per ottenere questo risultato, il kernel deve modificare l'indirizzo IP *sorgente* di ciascun pacchetto affinché tali risposte vengano re-instradate a esso invece che all'indirizzo IP privato che ha fatto la richiesta, procedura impossibile da eseguire su Internet. Linux fa uso del *tracciamento della connessione* (conntrack) per tenere traccia di quale connessione appartenga a quale computer e di conseguenza per re-instradare ciascun pacchetto di risposta. Il traffico in uscita dalla rete privata viene quindi "mascherato" per simulare l'uscita dalla macchina gateway Ubuntu. Nella documentazione Microsoft questo processo è indicato come condivisione delle connessioni internet (Internet Connection Sharing).

3.3.1. Masquerading con ufw

L'IP masquerading può essere ottenuto utilizzando regole ufw personalizzate. Questo è possibile dato che il backend attuale per ufw è iptables-restore con i file delle regole posizionati in `/etc/ufw/*.rules`. Questi file possono essere usati per aggiungere vecchie regole di iptables usate senza ufw e regole maggiormente legate al gateway o al bridge.

Le regole sono divise in due file diversi, regole da eseguire prima delle regole a riga di comando di ufw e regole da eseguire dopo ufw.

- Per prima cosa, è necessario abilitare l'inoltro dei pacchetti modificando due file di configurazione. In `/etc/default/ufw` modificare *DEFAULT_FORWARD_POLICY* in «ACCEPT»:

  ```
  DEFAULT_FORWARD_POLICY="ACCEPT"
  ```

  Quindi modificare il file `/etc/ufw/sysctl.conf` de-commentando:

  ```
  net/ipv4/ip_forward=1
  ```

  Similmente, per abilitare l'inoltro con IPv6 de-commentare:

  ```
  net/ipv6/conf/default/forwarding=1
  ```

- Ora verranno aggiunte delle regole al file `/etc/ufw/before.rules`. Le regole predefinite configurano solamente la tabella *filter* e per abilitare il masquerading è necessario configurare la tabella *nat*. Aggiungere all'inizio del file, subito dopo i commenti dell'intestazione, quanto segue:

  ```
  # regole tabella nat
  *nat
  :POSTROUTING ACCEPT [0:0]
  ```

```
# Inoltro traffico da eth1 attraverso eth0
-A POSTROUTING -s 192.168.0.0/24 -o eth0 -j MASQUERADE

# non cancellare la riga 'COMMIT' o queste tabelle di regole non saranno elaborate
COMMIT
```

I commenti non sono necessari, ma è buona pratica documentare le proprie configurazioni. Inoltre, quando si modificano i file *rules* in `/etc/ufw`, assicurarsi che queste righe siano sempre le ultime in ogni tabella modificata:

```
# non eliminare la riga 'COMMIT' o queste tabelle di regole non saranno elaborate
COMMIT
```

Per ogni *tabella* è necessario un *COMMIT*. In questi esempi sono mostrate solamente le tabelle *nat* e *filter*, ma è possibile aggiungere regole per le tabelle *raw* e *mangle*.

Nell'esempio precedente, sostituire *eth0*, *eth1* e *192.168.0.0/24* con le interfacce appropriate e con l'intervallo di indirizzi corretto.

- Infine, disattivare e riattivare ufw per applicare le modifiche:

```
sudo ufw disable && sudo ufw enable
```

L'IP masquerading ora dovrebbe essere abilitato. È possibile aggiungere regole FORWARD aggiuntive al file `/etc/ufw/before.rules`. È utile che queste regole addizionali vengano aggiunte alla catena *ufw-before-forward*.

3.3.2. Masquerading con iptables

È possibile anche utilizzare iptables per abilitare il masquerading.

- Similmente a ufw, il primo passo per abilitare l'inoltro di pacchetti con IPv4 è quello di modificare il file `/etc/sysctl.conf` e de-commentare la seguente riga:

```
net.ipv4.ip_forward=1
```

Per abilitare l'inoltro con IPv6, de-commetare:

```
net.ipv6.conf.default.forwarding=1
```

- Quindi, eseguire il comando sysctl per abilitare le nuove impostazioni nel file di configurazione:

```
sudo sysctl -p
```

- L'IP masquerading può essere ottenuto con una sola regola di iptables, che può cambiare leggermente in base alla configurazione della propria rete:

```
sudo iptables -t nat -A POSTROUTING -s 192.168.0.0/16 -o ppp0 -j MASQUERADE
```

Il comando precedente assume che lo spazio di indirizzi privato sia 192.168.0.0/16 e che il dispositivo collegato a Internet sia ppp0. La sintassi del comando è la seguente:

- -t nat: regola viene inserita nella tabella nat
- -A POSTROUTING: la regola viene accodata (-A) alla catena POSTROUTING
- -s 192.168.0.0/16: la regola si applica al traffico originato dallo spazio di indirizzi specificato
- -o ppp0: la regola si applica al traffico instradato attraverso l'interfaccia di rete specificata
- -j MASQUERADE: il traffico che soddisfa questa regola viene "saltato" (-j sta per jump) alla destinazione MASQUERADE per essere manipolato come descritto in precedenza

- Inoltre, ogni catena nella tabella "filter" (la tabella predefinita e dove avvengono la maggior parte dei filtri sui pacchetti) ha una *politica* predefinita di ACCEPT, ma se si sta creando un firewall in aggiunta a un dispositivo gateway, è possibile aver impostato le politiche DROP e REJECT, nel cui caso il traffico "masqueraded" deve essere consentito attraverso la catena FORWARD affinché la regola precedente possa funzionare:

```
sudo iptables -A FORWARD -s 192.168.0.0/16 -o ppp0 -j ACCEPT
sudo iptables -A FORWARD -d 192.168.0.0/16 -m state --state ESTABLISHED,RELATED -i ppp0 -j ACCEPT
```

  I precedenti comandi consentiranno a tutte le connessioni della propria rete locale accesso a Internet e a tutto il traffico relativo a queste connessioni di ritornare ai computer che lo hanno originato.
- Per fare in modo che il masquerading sia abilitato al riavvio, modificare il file `/etc/rc.local` e aggiungere qualsiasi dei comandi utilizzati precedentemente. Per esempio, aggiungere il primo comando senza filtro:

```
iptables -t nat -A POSTROUTING -s 192.168.0.0/16 -o ppp0 -j MASQUERADE
```

3.4. Registri

I registri del firewall sono molto utili per riconoscere gli attacchi, migliorare le regole del firewall e per verificare attività inusuali nella propria rete. È necessario includere regole di registrazione per fare in modo che vengano eseguite le registrazioni e queste devono essere inserite prima di qualsiasi regola terminante applicabile (un regola con un obiettivo che decide il destino di un pacchetto, come ACCEPT, DROP o REJECT).

Se si sta usando ufw è possibile attivare la registrazione con il seguente comando:

```
sudo ufw logging on
```

Per disabilitare la registrazione in ufw, sostituire, nel comando precedente, *on* con *off*.

Se è in uso iptables al posto di ufw, digitare:

```
sudo iptables -A INPUT -m state --state NEW -p tcp --dport 80 -j LOG --log-prefix "NEW_HTTP_CONN: "
```

Una richiesta sulla porta 80 dal computer locale, genererebbe, in dmesg, una traccia simile:

```
[4304885.870000] NUOV_CONN_HTTP: IN=lo OUT= MAC=00:00:00:00:00:00:00:00:00:00:00:00:08:00 SRC=127.0
```

Il registro precedente appare anche nei file `/var/log/messages`, `/var/log/syslog` e `/var/log/kern.log`. Questo comportamento può essere cambiato, modificando in modo appropriato il file `/etc/syslog.conf` oppure installando e configurando ulogd e facendo uso della destinazione ULOG al posto di LOG. Il demone ulogd è un server nello spazio utente in ascolto per le istruzioni di registro del kernel specifiche dei firewall; è possibile salvare i registri su qualsiasi file o perfino in un database come PostgreSQL o MySQL. Per dare un significato ai registri del firewall è possibile utilizzare delle applicazioni di analisi dei reigistri come fwanalog, fwlogwatch o lire.

3.5. Altri strumenti

Esistono diversi strumenti per "costruire" un firewall completo senza alcuna conoscenza di iptables. Per chi preferisce un'interfaccia grafica:

- *Firestarter*[2] è molto usato e facile da utilizzare.
- *fwbuilder*[3] è molto potente e ha un aspetto che può risultare familiare agli amministratori che hanno utilizzato un firewall commerciale come Checkpoint FireWall-1.

Per chi preferisce uno strumento a riga di comando con file di configurazione in semplice testo:

- *Shorewall*[4] è una soluzione molto potente per configurare un firewall di livello avanzato per qualsiasi rete.
- *ipkungfu*[5] dovrebbe fornire un firewall funzionante "out of the box", senza alcuna configurazione e consente di configurare firewall avanzati modificando semplici file di configurazione ben documentati.
- *fireflier*[6] è progettato per essere un'applicazione firewall per il desktop. È costituito da un server (fireflier-server), da un client grafico (GTK o QT) e si comporta come diverse applicazioni firewall interattive per Windows.

3.6. Riferimenti

- La pagina wiki relativa a *Ubuntu Firewall*[7] contiene informazioni sullo sviluppo di ufw.
- Inoltre, la pagina di manuale di ufw contiene molte informazioni utili: **man ufw**.
- Per maggiori informazioni sull'uso di iptables, consultare *packet-filtering-HOWTO*[8].
- Il *nat-HOWTO*[9] contiene ulteriori dettagli sul masquerading.
- The *IPTables HowTo*[10] in the Ubuntu wiki is a great resource.

4. AppArmor

AppArmor è un'implementazione del «Linux Security Module» per il controllo degli accessi vincolante basato sul nome. AppArmor racchiude individualmente i programmi in un insieme di file e capacità posix 1003.1e draft.

AppArmor è installato e caricato in modo predefinito e utilizza i *profili* di un'applicazione per determinare quali file e permessi siano necessari all'applicazione. Alcuni pacchetti installano i propri profili e ulteriori profili possono essere trovati nel pacchetto apparmor-profiles.

Per installare il pacchetto apparmor-profiles, in un terminale digitare:

```
sudo apt-get install apparmor-profiles
```

I profili di AppArmor dispongono di due modalità di esecuzione:

- Apprendimento (complaining/learning): le violazioni del profilo sono consentite e vengono registrate. Utile per verificare e sviluppare nuovi profili.
- Esecutiva (enforced/confined): obbliga a rispettare la politica del profilo e registra le violazioni.

4.1. Utilizzare AppArmor

Il pacchetto apparmor-utils contiene utilità a riga di comando che è possibile usare per modificare la modalità di esecuzione di AppArmor, trovare lo stato di un profilo, creare nuovi profili, ecc...

- apparmor_status è utilizzata per visualizzare lo stato attuale dei profili AppArmor.

  ```
  sudo apparmor_status
  ```

- aa-complain posiziona un profilo nella modalità *apprendimento*.

  ```
  sudo aa-complain /percorso/al/binario
  ```

- aa-enforce posiziona un profilo nella modalità *esecutiva*.

  ```
  sudo aa-enforce /percorso/al/bianrio
  ```

- Nella directory `/etc/apparmor.d` sono archiviati tutti i profili di AppArmor ed è possibile, da qui, modificare la *modalità* di tutti i profili.

  Usare il seguente comando per impostare tutti i profili nella modalità apprendimento:

  ```
  sudo aa-complain /etc/apparmor.d/*
  ```

  Per impostare tutti i profili nella modalità esecutiva:

  ```
  sudo aa-enforce /etc/apparmor.d/*
  ```

- apparmor_parser è utilizzata per caricare un profilo all'interno del kernel. Può essere usata anche per ricaricare profili attraverso l'opzione *-r*. Per caricare un profilo:

```
cat /etc/apparmor.d/profile.name | sudo apparmor_parser -a
```

  Per ricaricare un profilo:

```
cat /etc/apparmor.d/profile.name | sudo apparmor_parser -r
```

- `/etc/init.d/apparmor` può essere usato per *ricaricare* tutti i profili:

```
sudo /etc/init.d/apparmor reload
```

- La directory `/etc/apparmor.d/disable` può essere usata con l'opzione apparmor_parser -R per *disabilitare* un profilo.

```
sudo ln -s /etc/apparmor.d/profile.name /etc/apparmor.d/disable/
sudo apparmor_parser -R /etc/apparmor.d/profile.name
```

  Per *riabilitare* un profilo disabilitato, rimuovere il collegamento simbolico al profilo in `/etc/apparmor.d/disable/`, quindi caricare il profilo usando l'opzione *-a*.

```
sudo rm /etc/apparmor.d/disable/profile.name
cat /etc/apparmor.d/profile.name | sudo apparmor_parser -a
```

- È possibile disabilitare AppArmor e scaricare il modulo del kernel attraverso i seguenti comandi:

```
sudo /etc/init.d/apparmor stop
sudo update-rc.d -f apparmor remove
```

- Per riabilitare AppArmor:

```
sudo /etc/init.d/apparmor start
sudo update-rc.d apparmor defaults
```

Sostituire *profile.name* con il nome del profilo da modificare e sostituire anche `/percorso/eseguibile/` con il percorso all'eseguibile. Per esempio, per il comando ping, usare `/bin/ping`

4.2. Profili

I profili di AppArmor sono dei semplici file di testo posizionati in `/etc/apparmor.d/`. Questi file vengono nominati con il percorso completo all'eseguibile del profilo, sostituendo «/» con «.». Per esempio, `/etc/apparmor.d/bin.ping` è il profilo AppArmor del comando `/bin/ping`.

Esistono due principali tipologie di regole usate nei profili:

- *Voci di percorso*: specificano a quali file nel file system un'applicazione può accedere.

- *Voci di capacità*: determinano quali privilegi un processo può utilizzare.

Per un esempio, consultare `/etc/apparmor.d/bin.ping`:

```
#include <tunables/global>
/bin/ping flags=(complain) {
  #include <abstractions/base>
  #include <abstractions/consoles>
  #include <abstractions/nameservice>

  capability net_raw,
  capability setuid,
  network inet raw,

  /bin/ping mixr,
  /etc/modules.conf r,
}
```

- *#include <tunables/global>*: asserzioni di inclusione da altri file. Consente di usare un file comune con le asserzioni di inclusione per molteplici applicazioni.
- */bin/ping flags=(complain)*: percorso al programma con profilo, impostandone la modalità ad *apprendimento*.
- *capability net_raw,*: consente all'applicazione di accedere alla capacità CAP_NET_RAW Posix.1e.
- */bin/ping mixr,*: consente all'applicazione accesso in lettura e in esecuzione al file.

Dopo aver modificato un profilo, è necessario ricaricarlo. Per maggiori informazioni, consultare *Sezione 4.1, «Utilizzare AppArmor» [121]*.

4.2.1. Creare un profilo

- *Progettare un piano di verifica*: cercare di pensare a come l'applicazione dovrebbe essere eseguita. Il piano di verifica dovrebbe essere diviso in tanti piccoli casi d'uso, ognuno dei quali dovrebbe avere una breve descrizione e un elenco dei passi da compiere.

  Alcuni casi standard da verificare sono:

  - Avvio del programma.
  - Arresto del programma.
  - Ricaricamento del programma.
  - Verifica di tutti i comandi supportati dallo script init.
- *Generare il nuovo profilo*: usare aa-genprof per generare un nuovo profilo. Da un terminale:

  ```
  sudo aa-genprof eseguibile
  ```

  Per esempio:

```
sudo aa-genprof slapd
```

- Affinché il proprio nuovo profilo venga incluso nel pacchetto apparmor-profiles, segnalare un bug su *Launchpad* riguardo il pacchetto *AppArmor*[11]:
  - Includere la pianificazione e le casistiche del test.
  - Allegare il nuovo profilo al bug.

4.2.2. Aggiornare i profili

Quando il programma si comporta stranamente, messaggi di audit vengono inviati ai file di registro. Il programma aa-logprof può essere usato per analizzare i file di registro per i messaggi di audit di AppArmor, per controllarli e per aggiornare i profili. Da un terminale:

```
sudo aa-logprof
```

4.3. Riferimenti

- Per le opzioni avanzate di configurazione, consultare la *AppArmor Administration Guide*[12]
- Per maggiori informazioni su come usare AppArmor con altri rilasci di Ubuntu, consultare la *documentazione della comunità italiana*[13].
- La pagina *OpenSUSE AppArmor*[14] è un'altra introduzione ad AppArmor.
- Un buon posto per chiedere assistenza riguardo AppArmor, e per partecipare nella comunità di Ubuntu Server, è il canale IRC *#ubuntu-server* su *freenode*[15].

5. Certificati

Una delle più comuni forme di crittografia odierna è la crittografia a *chiave pubblica*. Questo tipo di crittografia utilizza una *chiave pubblica* e una *chiave privata*. Il sistema funziona *cifrando* le informazioni usando la chiave pubblica che possono solo essere *decifrate* con la chiave privata.

L'utilizzo più comune della crittografia a chiave pubblica è nella cifratura del traffico delle applicazioni attraverso una connessione SSL (Secure Socket Layer) o TLS (Transport Layer Security), per esempio configurando Apache affinché fornisca *HTTPS*, il protocollo HTTP via SSL. Questo consente di cifrare il traffico utilizzando un protocollo che non fornisce nativamente una cifratura.

Un *certificato* è un metodo di distribuzione di una *chiave pubblica* e di altre informazioni riguardo un server e l'organizzazione che ne è responsabile. I certificati possono essere firmati digitalmente a un'*Autorità di Certificazione* o CA. Una CA è un'entità fidata che conferma la veridicità delle informazioni contenute nel certificato.

5.1. Tipologie dei certificati

Per configurare un server sicuro affinché usi la crittografia a chiave pubblica, nella maggior parte dei casi, è necessario inviare la richiesta del certificato (compresa la chiave pubblica), una prova di esistenza della propria società e il pagamento a una CA. La CA verifica la richiesta e la propria identità e quindi invia un certificato per il proprio server. In alternativa, è possibile creare il proprio certificato *auto-firmato*.

I certificati auto-firmati non dovrebbero essere usati in ambienti di produzione.

Continuando l'esempio di HTTPS, un certificato CA firmato dispone di caratteristiche che un certificato auto-firmato non ha:

- I browser, solitamente, riconoscono automaticamente il certificato e consentono l'attivazione di una connessione sicura senza chiedere nulla all'utente.
- Quando una CA emette un certificato, garantisce l'identità dell'organizzazione che fornisce la pagina web al browser.

La maggior parte dei browser web, e dei computer che supportano SSL, dispongono di un elenco di CA i cui certificati sono accettati automaticamente. Se un browser incontra un certificato la cui CA non è presente nell'elenco, il browser chiede all'utente di accettare o rifiutare la connessione. Inoltre, altre applicazioni possono generare un messaggio di errore quando viene usato un certificato auto-firmato.

Il processo per ottenere un certificato da una CA è molto semplice. Un piccolo promemoria:

1. Creare un coppia di chiavi pubblica e privata.
2. Creare una richiesta per un certificato basato su chiave pubblica. La richiesta del certificato contiene informazioni riguardo il server a la società che lo ospita.

3. Inviare la richiesta, con una fotocopia di un documento di identità, a una CA. Non è possibile consigliare quale autorità di certificazione scegliere. La decisione potrebbe essere basata su esperienze passate, esperienze di amici o colleghi o per un fattore economico.

   Una volta scelta la CA, è necessario seguire le istruzioni fornite dal CA per ottenere il certificato.
4. Una volta che la CA ha verificato l'identità del richiedente, invierà un certificato digitale.
5. Installare questo certificato sul proprio server sicuro e configurare le applicazioni appropriate affinché usino il certificato.

5.2. Generare una CSR (Certificate Signing Request)

Sia che si stia ottenendo un certificato da una CA sia che si auto-firmi il proprio, il primo passo consiste nel generare una chiave di cifratura.

Se il certificato verrà usato da servizi come Apache, Postfix, Dovect, ecc..., è solitamente indicato usare una chiave priva di passphrase.

Questa sezione indica come generare una chiave dotata di passphrase e una priva di passphrase. La chiave priva di passphrase verrà impiegata per generare un certificato che può essere usato da diversi servizi.

Avere in esecuzione i servizi senza una passphrase è conveniente poiché non vi è la necessità di digitare la passphrase a ogni avvio del servizio, ma non è molto sicuro in quanto se la chiave viene compromessa, verrà compromesso anche il server.

Per generare le *chiavi* per la CSR (Certificate Signing Request), eseguire in un terminale il seguente comando:

```
openssl genrsa -des3 -out server.key 1024
```

```
Generating RSA private key, 1024 bit long modulus
.....................++++++
.................++++++
unable to write 'random state'
e is 65537 (0x10001)
Enter pass phrase for server.key:
```

È ora necessario inserire una passphrase. Per una maggiore sicurezza, dovrebbe contenere almeno 8 caratteri. La lunghezza minima con l'opzione «-des3» è di 4 caratteri. Dovrebbe includere numeri o segni di punteggiatura e non dovrebbe essere una parola reperibile in un vocabolario. Ricordarsi che una passphrase differenzia tra minuscole e maiuscole.

Digitare nuovamente la passphrase per la verifica. Una volta digitata correttamente, la chiave per il server viene generata e archiviata nel file `server.key`.

Creare la chiave insicura, quella priva di passphrase, e scambiare i nomi delle chiavi:

```
openssl rsa -in server.key -out server.key.insecure
mv server.key server.key.secure
mv server.key.insecure server.key
```

La chiave insicura è ora chiamata `server.key` ed è possibile usare questo file per generare la CSR senza passphrase.

Per creare il CSR, eseguire il seguente comando:

```
openssl req -new -key server.key -out server.csr
```

It will prompt you enter the passphrase. If you enter the correct passphrase, it will prompt you to enter Company Name, Site Name, Email Id, etc. Once you enter all these details, your CSR will be created and it will be stored in the `server.csr` file.

È ora possibile inviare il file della CSR alla CA che lo utilizzerà per creare il certificato finale. È comunque possibile creare un certificato auto-firmato utilizzando questa CSR.

5.3. Creare un certificato auto-firmato

Per creare un certificato auto-firmato, eseguire da un terminale il seguente comando:

```
openssl x509 -req -days 365 -in server.csr -signkey server.key -out server.crt
```

Il comando precedente chiederà la passphrase. Una volta digitata correttamente, il certificato viene creato e sarà disponibile nel file `server.crt`.

Se il server deve essere utilizzato in ambito commerciale, è necessario un certificato emesso da una CA. Non è raccomandato utilizzare un certificato auto-firmato.

5.4. Installare il certificato

È possibile installare il file `server.key` e quello del certificato `server.crt`, o il file del certificato fornito dalla CA, eseguendo, in un terminale, i seguenti comandi:

```
sudo cp server.crt /etc/ssl/certs
sudo cp server.key /etc/ssl/private
```

Ora basta configurare le applicazioni che possono usare la crittografia a chiave pubblica affinché utilizzino i file del *certificato* e della *chiave*. Per esempio, Apache può fornire HTTPS, Dovecot può fornire IMAPS e POP3S ecc...

5.5. Autorità di Certificazione

Se i servizi all'interno della propria rete richiedono più di un certificato auto-firmato, potrebbe essere utile impostare una *Autorità di Certificazione* personale. Usando certificati firmati dalla propria CA,

consente ai vari servizi che usano tali certificati di fidarsi di altri servizi che fanno uso di certificati emessi dalla stessa CA.

1. Per prima cosa, creare le directory che conterranno il certificato della CA e i file relativi

   ```
   sudo mkdir /etc/ssl/CA
   sudo mkdir /etc/ssl/newcerts
   ```

2. La CA necessita di alcuni altri file per funzionare correttamente: uno per tenere traccia dell'ultimo numero seriale usato (ogni certificato deve avere un numero univoco) e l'altro per registrare quali certificati sono stati emessi:

   ```
   sudo sh -c "echo '01' > /etc/ssl/CA/serial"
   sudo touch /etc/ssl/CA/index.txt
   ```

3. Il terzo file è il file di configurazione della CA. Benché non strettamente necessario, è molto utile quando vengono emessi certificati multipli. Aprire il file `/etc/ssl/openssl.cnf` e nella sezione *[CA_default]* modificare:

   ```
   dir             = /etc/ssl/             # Dove viene salvato tutto
   database        = $dir/CA/index.txt     # File indice del database
   certificate     = $dir/certs/cacert.pem # Il certificato della CA
   serial          = $dir/CA/serial        # Il numero seriale corrente
   private_key     = $dir/private/cakey.pem# La chiave privata
   ```

4. Creare il certificato auto-firmato principale:

   ```
   openssl req -new -x509 -extensions v3_ca -keyout cakey.pem -out cacert.pem -days 3650
   ```

   Viene chiesto di inserire i dettagli del certificato.

5. Installare il certificato principale e la chiave:

   ```
   sudo mv cakey.pem /etc/ssl/private/
   sudo mv cacert.pem /etc/ssl/certs/
   ```

6. È ora possibile firmare i certificati. La prima cosa necessaria è una CSR (Certificate Signing Request), consultare *Sezione 5.2, «Generare una CSR (Certificate Signing Request)» [126]*. Una volta ottenuta, digitare quanto segue per generare un certificato firmato dalla CA:

   ```
   sudo openssl ca -in server.csr -config /etc/ssl/openssl.cnf
   ```

   Inserita la passoword della chiave CA, viene chiesto di firmare il certificato e di generare quello nuovo. Dovrebbe quindi essere visibile l'output della generazione del certificato stesso.

7. There should now be a new file, `/etc/ssl/newcerts/01.pem`, containing the same output. Copy and paste everything beginning with the line: -----*BEGIN CERTIFICATE*----- and continuing through the line: ----*END CERTIFICATE*----- lines to a file named after the hostname of the

server where the certificate will be installed. For example `mail.example.com.crt`, is a nice descriptive name.

Tutti i certificati successivi saranno chiamati `02.pem`, `03.pem`, ecc...

Sostituire *mail.example.it.crt* con un nome descrittivo appropriato al proprio caso.

8. In fine, copiare il nuovo certificato nell'host e configurare le applicazioni al suo uso. La posizione predefinita per l'installazione dei certificati è `/etc/ssl/certs`, consentendo così a molteplici servizi di usare lo stesso certificato senza complicare inutilmente i permessi.

   Per le applicazioni che possono essere configurate all'uso di un certificato di una CA, è necessario copiare il file `/etc/ssl/certs/cacert.pem` nella directory `/etc/ssl/certs/` di ogni server.

5.6. Riferimenti

- Per ulteriori informazioni sull'utilizzo della crittografia, consultare lo *SSL Certificates HOWTO*[16].
- Il sito *The PKI Page*[17] contiene un elenco di autorità di certificazione.
- La pagina Wikipedia *HTTPS*[18] dispone di ulteriori informazioni riguardo HTTPS.
- Per maggiori informazioni riguardo *OpenSSL*, consultare il *sito web di OpenSSL*[19].
- Inoltre, il libro *Network Security with OpenSSL*[20] di O'Reilly è un ottimo punto di riferimento.

6. eCryptfs

eCryptfs è un file system crittografico POSIX-conforme per Linux. Disponendosi al di sopra del livello del file system normale, *eCryptfs* è in grado di proteggere i file indipendentemente dal file system sottostante, dal tipo di partizione, ecc...

Durante la fase di installazione è disponibile un'opzione per cifrare l'intera partizione `/home` in grado di configurare tutto il necessario per cifrare e montare la partizione.

As an example, this section will cover configuring `/srv` to be encrypted using *eCryptfs*.

6.1. Usare eCryptfs

Per prima cosa, installare i pacchetti necessari. In un terminale digitare:

```
sudo apt-get install ecryptfs-utils
```

Montare la partizione da cifrare:

```
sudo mount -t ecryptfs /srv /srv
```

Vengono chiesti alcuni dettagli su come ecryptfs dovrebbe cifrare i dati.

Per verificare che i file in `/srv` siano veramente cifrati, copiare la directory `/etc/default` in `/srv`:

```
sudo cp -r /etc/default /srv
```

Smontare `/srv` e cercare di visualizzare un file:

```
sudo umount /srv
cat /srv/default/cron
```

Montare `/srv` utilizzando ecryptfs per poter visualizzare nuovamente i dati.

6.2. Montare automaticamente le partizioni cifrate

È possibile montare un file system ecryptfs in diversi modi all'avvio. Questo esempio fa uso di un file `/root/.ecryptfsrc` contenente le opzioni di mount e un file, salvato su una chiave USB, contenente la passphrase.

Creare il file `/root/.ecryptfsrc` contenente:

```
key=passphrase:passphrase_passwd_file=/mnt/usb/passwd_file.txt
ecryptfs_sig=5826dd62cf81c615
ecryptfs_cipher=aes
ecryptfs_key_bytes=16
```

```
ecryptfs_passthrough=n
ecryptfs_enable_filename_crypto=n
```

Modificare il campo *ecryptfs_sig* con la firma presente in `/root/.ecryptfs/sig-cache.txt`.

Creare il file `/mnt/usb/passwd_file.txt` per la passphrase:

```
passphrase_passwd=[secrets]
```

Aggiungere quanto necessario in `/etc/fstab`:

```
/dev/sdb1       /mnt/usb        ext3    ro      0 0
/srv /srv ecryptfs defaults 0 0
```

Assicurarsi che il dispositivo USB venga montato prima della partizione cifrata.

Finally, reboot and the `/srv` should be mounted using *eCryptfs*.

6.3. Altre utilità

Il pacchetto ecryptfs-utils contiene diverse utilità:

- *ecryptfs-setup-private*: crea una directory `~/Private` per contenere informazioni cifrate. Questa utilità può essere eseguita da utenti senza alcun tipo di privilegio all'interno del sistema per creare una piccola zona privata dove salvare dati.
- *ecryptfs-mount-private* e *ecryptfs-umount-private*: monta e smonta la directory `~/Private` degli utenti.
- *ecryptfs-add-passphrase*: aggiunge una nuova passphrase al portachiavi.
- *ecryptfs-manager*: gestisce gli oggetti eCryptfs come le chiavi.
- *ecryptfs-stat* consente di visualizzare le meta informazioni di ecryptfs relative a un file.

6.4. Riferimenti

- For more information on *eCryptfs* see the *Launchpad project page*[21].
- There is also a *Linux Journal*[22] article covering *eCryptfs*.
- Also, for more ecryptfs options see the *ecryptfs man page*[23].
- The *eCryptfs Ubuntu Wiki*[24] page also has more details.

Capitolo 9. Monitoraggio

1. Panoramica

Il monitoraggio di server e servizi essenziali è un aspetto importante dell'amministrazione di sistema. La maggior parte dei servizi di rete vengono monitorati per controllarne prestazioni, disponibilità oppure entrambi. Questa sezione descrive l'installazione e la configurazione di Nagios per il monitoraggio mirato alla disponibilità dei servizi e di Munin per il monitoraggio delle prestazioni.

Gli esempi in questa sezione utilizzano due server con nome host *server01* e *server02*. Il server chiamato *server01* viene configurato con Nagios per monitorare i servizi sul server stesso e su *server02*. Inoltre, viene configurato anche munin per raccogliere informazioni dalla rete. Utilizzando il pacchetto munin-node, *server02* viene configurato per inviare informazioni a *server01*.

Questi semplice esempi dovrebbe permettere di monitorare server aggiuntivi e servizi all'interno della rete.

2. Nagios

2.1. Installazione

Per prima cosa, su *server01* installare il pacchetto nagios. In un terminale digitare:

```
sudo apt-get install nagios3 nagios-nrpe-plugin
```

Viene chiesto di inserire una password per l'utente *nagiosadmin*. Le credenziali vengono salvate nel file `/etc/nagios3/htpasswd.users`. Per modificare la password dell'utente *nagiosadmin* o per aggiungere altri utenti, usare il comando htpasswd, parte del pacchetto apache2-utils.

Per esempio, per modificare la password dell'utente *nagiosadmin* digitare:

```
sudo htpasswd /etc/nagios3/htpasswd.users nagiosadmin
```

Per aggiungere un utente:

```
sudo htpasswd /etc/nagios3/htpasswd.users steve
```

Su *server02* installare il pacchetto nagios-nrpe-server. Da un terminale su server02 inserire:

```
sudo apt-get install nagios-nrpe-server
```

NRPE consente di eseguire controlli locali sugli host remoti. Esistono anche altri metodi per eseguire questo attraverso l'uso di altri plugin o controlli di Nagios.

2.2. Panoramica della configurazione

Esistono diverse directory contenenti file di configurazione e di controllo di Nagios.

- `/etc/nagios3`: contiene i file di configurazione per le operazioni del demone nagios, i file CGI, host, ecc...
- `/etc/nagios-plugins`: contiene i file di configurazione per i controlli del servizio.
- `/etc/nagios`: sull'host remoto contiene i file di configurazione di nagios-nrpe-server.
- `/usr/lib/nagios/plugins/`: contiene i file binari dei controlli. Per visualizzare le opzioni di un controllo, usare l'opzione *-h*.

  Per esempio: **/usr/lib/nagios/plugins/check_dhcp -h**

Esistono moltissimi controlli che è possibile eseguire tramite Nagios su un qualsiasi host. In questo esempio Nagios viene configurato per controlla lo spazio su disco, DNS e un gruppo di host MySQL. Il controllo DNS avviene su *server02* e il gruppo di host MySQL include sia *server01* che *server02*.

Consultare *Sezione 1, «HTTPD - Server web Apache2» [141]* per informazioni su Apache, *Capitolo 7, DNS (Domain Name Service) [94]* su DNS e *Sezione 1, «MySQL» [160]* su MySQL.

Inoltre, vi sono alcuni termini che una volta descritti, aiuteranno a rendere più semplice la comprensione di Nagios:

- *Host*: un server, una workstation o un dispositivo di rete che viene monitorato.
- *Gruppo di host*: un gruppo di host simili. Per esempio potrebbe essere possibile raggruppare tutti i server web, i server di file, ecc...
- *Servizio*: il servizio che viene monitorato sull'host come HTTP, DNS, FTP, ecc...
- *Gruppo di servizi*: consente di raggruppare servizi simili. Utile, per esempio, per raggruppare più servizi HTTP.
- *Contatto*: una persona da notificare quando si verifica un evento. Nagios può essere configurato per inviare email, SMS, ecc...

Come impostazione predefinita, Nagios è configurato per controllare HTTP, spazio su disco, SSH, gli utenti attuali, i processi e il carico sul *localhost*. Inoltre, è in grado di controllare attraverso il comando ping il *gateway*.

Installazioni di Nagios di grosse dimensioni possono essere complesse da configurare ed è quindi utile partire con una configurazione piccola, uno o due host, prima di aumentare le dimensioni.

2.3. Configurazione

- 1. Creare un file di configurazione *host* per *server02*. In un terminale digitare:

```
sudo cp /etc/nagios3/conf.d/localhost_nagios2.cfg /etc/nagios3/conf.d/server02.cfg
```

Nei comandi precedenti e in quelli che seguono, sostituire *"server01"*, *"server02"*, *172.18.100.100* e *172.18.100.101* con i nomi host e gli indirizzi IP dei propri server.

2. Modificare il file `/etc/nagios3/conf.d/server02.cfg`:

```
define host{
        use                     generic-host            ; Name of host template to use
        host_name               server02
        alias                   Server 02
        address                 172.18.100.101
}

# check DNS service.
define service {
        use                             generic-service
        host_name                       server02
        service_description             DNS
        check_command                   check_dns!172.18.100.101
```

```
}
```

3. Riavviare il demone nagios per abilitare la nuova configurazione:

```
sudo /etc/init.d/nagios3 restart
```

- 1. Aggiungere una definizione di servizio per il controllo MySQL aggiungendo quanto segue al file `/etc/nagios3/conf.d/services_nagios2.cfg`:

```
# check MySQL servers.
define service {
        hostgroup_name                  mysql-servers
        service_description             MySQL
        check_command                   check_mysql_cmdlinecred!nagios!secret!$HOSTADDRESS
        use                             generic-service
        notification_interval           0 ; set > 0 if you want to be renotified
}
```

2. A *mysql-servers* hostgroup now needs to be defined. Edit `/etc/nagios3/conf.d/hostgroups_nagios2.cfg` adding:

```
# MySQL hostgroup.
define hostgroup {
        hostgroup_name  mysql-servers
                alias           MySQL servers
                members         localhost, server02
        }
```

3. Il controllo di Nagios necessita di autenticarsi con MySQL. Per aggiungere un utente *nagios* a MySQL inserire:

```
mysql -u root -p -e "create user nagios identified by 'secret';"
```

È necessario aggiungere l'utente *nagios* a tutti gli host deò gruppo *mysql-servers* hostgroup.

4. Riavviare nagios per iniziare il controllo dei server MySQL.

```
sudo /etc/init.d/nagios3 restart
```

- 1. Infine configurare NRPE affinché controlli lo spazio su disco su *server02*.

Sul *server01* aggiungere il controllo del servizio al file `/etc/nagios3/conf.d/server02.cfg`:

```
# NRPE disk check.
define service {
        use                             generic-service
        host_name                       server02
        service_description             nrpe-disk
        check_command                   check_nrpe_1arg!check_all_disks!172.18.100.101
```

```
}
```

2. Su *server02* modificare il file `/etc/nagios/nrpe.cfg`:

   ```
   allowed_hosts=172.18.100.100
   ```

   E nella sezione dove sono definiti i comandi, aggiungere:

   ```
   command[check_all_disks]=/usr/lib/nagios/plugins/check_disk -w 20% -c 10% -e
   ```

3. Infine, riavviare nagios-nrpe-server:

   ```
   sudo /etc/init.d/nagios-nrpe-server restart
   ```

4. Riavviare, su *server01*, nagios:

   ```
   sudo /etc/init.d/nagios3 restart
   ```

Dovrebbe essere possibile visualizzare l'host e i controlli nei file CGI di Nagios. Per accedere a questi file, in un browser web inserire l'indirizzo http://server01/nagios3. Vengono richiesti password e nome utente dell'utente *nagiosadmin*.

2.4. Riferimenti

Questa sezione ha fornito una panoramica preliminare delle caratteristiche di Nagios, i pacchetti nagios-plugins-extra e nagios-snmp-plugins contengono molti altri controlli.

- Per maggiori informazioni, consultare il sito web di *Nagios*[1].
- In particolare, consultare la *documentazione in rete*[2].
- Sono disponibili anche molti *libri*[3] riguardo Nagios e il monitoraggio di rete:
- The *Nagios Ubuntu Wiki*[4] page also has more details.

3. Munin

3.1. Installazione

Prima di installare Munin su *server01*, è necessario installare apache2. La configurazione predefinita è sufficiente per poter eseguire un server munin. Per maggiori informazioni, consultare *Sezione 1, «HTTPD - Server web Apache2» [141]*.

Installare, su *server01*, munin. In un terminale, inserire:

```
sudo apt-get install munin
```

Su *server02*, installare il pacchetto munin-node:

```
sudo apt-get install munin-node
```

3.2. Configurazione

Su *server01* modificare il file `/etc/munin/munin.conf` aggiungendo l'indirizzo IP di *server02*:

```
## First our "normal" host.
[server02]
       address 172.18.100.101
```

Sostituire *server02* e *172.18.100.101* con il nome host e con l'indirizzo IP del proprio server.

Successivamente, configurare munin-node su *server02*. Modificare il file `/etc/munin/munin-node.conf` per consentire l'accesso al *server01*:

```
allow ^172\.18\.100\.100$
```

Sostituire *^172\.18\.100\.100$* con l'indirizzo IP del proprio server munin.

Riavviare munin-node su *server02* per applicare le modifiche:

```
sudo /etc/init.d/munin-node restart
```

Infine, in un browser, inserire l'indirizzo *http://server01/munin* per visualizzare grafici che rappresentano le informazioni dal pacchetto *munin-plugins* standard per disco, rete, processi e sistema.

Poiché è una nuova installazione, potrebbe impiegare un po' di tempo affinché i grafici visualizzino qualche cosa di utile.

3.3. Plugin aggiuntivi

Il pacchetto munin-plugins-extra contiene controlli per le prestazioni e per servizi come DNS, DHCP, Samba e altri. Per installare il pacchetto, in un terminale inserire:

```
sudo apt-get install munin-plugins-extra
```

Assicurarsi di installare il pacchetto sia sul server che su tutti i nodi.

3.4. Riferimenti

- Per maggiori informazioni, consultare il sito web di *Munin*[5].
- In particolare, la pagina relativa *alla documentazione*[6] contiene informazioni su maggiori plugin, sulla scrittura di plugin, ecc..
- È anche disponibile un libro in tedesco da Open Source Press: *Munin Graphisches Netzwerk- und System-Monitoring*[7].
- Another resource is the *Munin Ubuntu Wiki*[8] page.

Capitolo 10. Server web

Un server web è un programma interattivo che accetta richieste HTTP da client, noti come browser web, e invia loro risposte HTTP insieme ad altri dati opzionali, di solito pagine web come documenti HTML e oggetti collegati (immagini, ecc.).

1. HTTPD - Server web Apache2

Apache è il server web più utilizzato nei sistemi Linux. I server web sono usati per inviare le pagine web richieste da un client, che solitamente esege le richieste attraverso un browser web come Firefox, Opera o Mozilla.

Gli utenti inseriscono un URL (Uniform Resource Locator) per contattare un server web conoscendo il suo FQDN (Fully Qualified Domain Name, nome di dominio non ambiguo, ndt) e un percorso per la risorsa richiesta. Per esempio, per vedere la pagina principale del *sito web di Ubuntu*[1], l'utente deve inserire solamente il FQDN. Per richiedere informazioni specifiche sul *supporto a pagamento*[2], l'utente deve inserire il FQDN seguito da un percorso.

Il protocollo più utilizzato per il trasferimento delle pagine web è l'HTTP (Hyper Text Transfer Protocol). Sono anche supportati protocolli come HTTPS (Hyper Text Transfer Protocol over Secure Sockets Layer) e FTP (File Transfer Protocol), un protocollo per caricare e scaricare file dalla rete.

I server web Apache vengono comunemente usati in combinazione con il motore di database MySQL, il linguaggio di script per la pre-elaborazione dell'ipertesto PHP (Pre-processor Hyper Text) e altri noti linguaggi di script come Python e Perl. Questa configurazione viene denominata LAMP (Linux, Apache, MYSQL e Perl/Phyton/PHP) e costituisce una piattaforma robusta e potente per lo sviluppo e l'installazione di applicazioni basate sul web.

1.1. Installazione

Il server web Apache2 è disponibile in Ubuntu 10.04. Per installare Apache2:

- Al prompt di un terminale, eseguire il seguente comando:

```
sudo apt-get install apache2
```

1.2. Configurazione

La configurazione di Apache2 avviene scrivendo delle *direttive* in semplici file di testo. Queste *direttive* sono suddivise tra i seguenti file e directory:

- *apache2.conf*: il principale file di configurazione di Apache2. Contiene impostazioni *globali* per Apache2.
- *conf.d*: contiene file di configurazione che si applicano *globalmente* ad Apache2. Altri pacchetti che usano Apache2 per fornire contenuti possono aggiungere file o collegamenti simbolici in questa directory.
- *envvars*: file dove vengono impostate le variabili *d'ambiente* di Apache2.

[1] http://www.ubuntu-it.org

[2] http://www.ubuntu.com/support/paid

- *httpd.conf*: storicamente il file di configurazione principale di Apache2, il cui nome deriva dal demone httpd. Il file può essere usato per configurazioni *utente* che hanno effetto globalmente su Apache2.
- *mods-available*: questa directory contiene file di configurazione per caricare e configurare *moduli*. Non tutti i moduli hanno file di configurazione specifici.
- *mods-enabled*: contiene *collegamenti simbolici* ai file in `/etc/apache2/mods-available`. Quando viene creato un collegamento simbolico a un modulo di configurazione, viene abilitato al successivo riavvio di apache2.
- *ports.conf*: contiene le direttive che determinano su quali porte TCP Apache2 sta in ascolto.
- *sites-available*: questa directory contiene i file di configurazione per i *Virtual Hosts* di Apache2. Questi consentono di configurare Apache2 affinché venga utilizzato per siti multipli con configurazioni separate.
- *sites-enabled*: come mods-enabled, `sites-enabled` contiene collegamenti simbolici alla directory `/etc/apache2/sites-available`. Quando viene creato un collegamento simbolico di un file di configurazione nella directory sites-available, il sito configurato sarà attivo al riavvio di Apache2.

Altri file di configurazione possono essere aggiunti attraverso la direttiva *Include* e caratteri speciali possono essere usati per aggiungere molti altri file di configurazione. Una qualsiasi direttiva può essere posizionata in uno qualsiasi di questi file di configurazione. Modifiche ai file principali di configurazione vengono riconosciute solo con un riavvio di Apache2.

Il server legge anche un file contenente i tipi MIME dei documenti, il nome di questo file è impostato in *TypesConfig* e il valore predefinito è `/etc/mime.types`

1.2.1. Impostazioni di base

Questa sezione descrive i parametri di configurazione fondamentali del server Apache2. Per maggiori informazioni, consultare la *documentazione di Apache2*[3].

- Apache2 è dotato di una configurazione predefinita adatta agli host virtuali: è configurato con un singolo host virtuale (attraverso l'uso della direttiva *VirtualHost*) che può essere modificato oppure usato così com'è nel caso si disponga di un solo sito web oppure usato come modello per aggiungere altri host virtuali. Se lasciato così, l'host virtuale predefinito verrà usato come sito predefinito o come il sito che gli utenti vedranno se l'URL inserito non corrisponde alla direttiva *ServerName* in uno qualsiasi dei file personalizzati. Per modificare l'host virtuale, modificare il file `/etc/apache2/sites-available/default`.

Le direttive impostate per un host virtuale si applicano solamente a quel particolare host. Se una direttiva è impostata all'interno del server e non è definita nelle impostazioni dell'host virtuale, vengono utilizzate le impostazioni predefinite. Per esempio, è possibile impostare un indirizzo email per il webmaster e non definirne alcuno per per gli host virtuali.

[3] http://httpd.apache.org/docs/2.2/

Per configurare un nuovo host virtuale o un nuovo sito, copiare quel file nella stessa directory con un nome a scelta. Per esempio:

```
sudo cp /etc/apache2/sites-available/default /etc/apache2/sites-available/mionuovosito
```

Modificare il file per configurare il nuovo sito usando alcune delle direttive descritte di seguito.

- La direttiva *ServerAdmin* specifica a quale indirizzo email il sistema deve indirizzare la posta destinata agli amministratori. Il valore predefinito è «webmaster@localhost». Quest'impostazione deve essere modificata con l'indirizzo che è stato assegnato all'utente (nel caso sia l'amministratore). Se il sito presenta dei problemi, Apache2 mostrerà un messaggio di errore indicante l'indirizzo a cui deve essere segnalato il problema. Questa direttiva è presente nel file /etc/apache2/sites-available del proprio sito.
- La direttiva *Listen* specifica la porta, e opzionalmente l'indirizzo IP, su cui Apache2 dovrebbe essere in ascolto. Se l'indirizzo IP non è specificato, Apache2 ascolta tutti gli indirizzi IP assegnati alla macchina. Il valore predefinito per la direttiva Listen è 80. Modificare questo valore, in 127.0.0.1:80 per fare in modo che Apache2 ascolti solo l'interfaccia di loopback e non sia disponibile verso internet, in 81 per modificare la porta di ascolto o lasciare il valore predefinito per il normale funzionamento. Questa direttiva può essere trovata e modifica in un file specifico: `/etc/apache2/ports.conf`
- La direttiva *ServerName* è opzionale e specifica il FQDN a cui il proprio sito risponde. L'host virtuale predefinito non ha la direttiva ServerName impostata, cosicché risponderà a tutte le richieste che non corrispondono alla direttiva ServerName in un altro host virtuale. Se si è i proprietari del dominio "ubunturocks.com" e si vuole ospitare tale dominio su un server Ubuntu, il valore della direttiva ServerName nel file di configurazione dell'host virtuale dovrebbe essere "ubunturocks.com". Aggiungere quindi questa direttiva al nuovo file di configurazione creato precedentemente (`/etc/apache2/sites-available/mionuovosito`).

  Potrebbe essere necessario che il proprio sito risponda anche alle richieste per "www.ubunturocks.com", dato che molti utenti ritengono corretto inserire il prefisso "www". Per ottenere questo, usare la direttiva *ServerAlias*: è possibile usare anche caratteri speciali con la direttiva ServerAlias.

  Per esempio, la seguente configurazione farà in modo che il proprio sito risponda a qualsiasi richiesta il cui dominio termina con *.ubunturocks.com*.

  ```
  ServerAlias *.ubunturocks.com
  ```

- La direttiva *DocumentRoot* specifica dove Apache2 deve cercare i file che compongono il sito. Il valore predefinito è /var/www. Nessun sito viene configurato qui, ma se si viene tolto il commento alla direttiva *RedirectMatch* in `/etc/apache2/apache2.conf` le richieste verranno reindirizzate a /var/www/apache2-default dove è presente il sito predefinito Apache2. Cambiare questo valore nel file virtual host del proprio sito, e ricordare di creare quella directory se necessaria.

La directory /etc/apache2/sites-available **non** viene analizzata da Apache2. I collegamenti simbolici in /etc/apache2/sites-enabled puntano ai siti disponibili.

Abilitare il nuovo *VirtualHost* utilizzando l'utilità a2ensite e riavviare Apache2:

```
sudo a2ensite mionuovosito
sudo /etc/init.d/apache2 restart
```

Assicurarsi di sostituire *mionuovosito* con una nome più descrittivo per il VirtualHost. Un metodo molto utilizzato consiste nel definire il nome del file secondo la direttiva *ServerName* dell'host virtuale.

Allo stesso modo, usare l'utilità a2dissite per disabilitare i siti. Questo può rivelarsi utile per diagnosticare problemi di configurazione con molteplici host virtuali:

```
sudo a2dissite mionuovosito
sudo /etc/init.d/apache2 restart
```

1.2.2. Impostazioni predefinite

Questa sezione si occupa delle impostazioni predefinite del server Apache2. Per esempio, se viene aggiunto un host virtuale, le impostazioni modificate dell'host virtuale hanno precedenza rispetto quelle dell'host. Per una direttiva non definita, viene utilizzato il valore predefinito.

- *DirectoryIndex* è la pagina predefinita proposta dal server alle richieste dell'indice di una directory, specificate attraverso l'uso di una barra (/) come suffisso al nome della directory.

  For example, when a user requests the page http://www.example.com/this_directory/, he or she will get either the DirectoryIndex page if it exists, a server-generated directory list if it does not and the Indexes option is specified, or a Permission Denied page if neither is true. The server will try to find one of the files listed in the DirectoryIndex directive and will return the first one it finds. If it does not find any of these files and if *Options Indexes* is set for that directory, the server will generate and return a list, in HTML format, of the subdirectories and files in the directory. The default value, found in `/etc/apache2/mods-available/dir.conf` is "index.html index.cgi index.pl index.php index.xhtml index.htm". Thus, if Apache2 finds a file in a requested directory matching any of these names, the first will be displayed.
- La direttiva *ErrorDocument* consente di specificare un file da usare per errori specifici. Per esempio, se un utente cerca di accedere a una risorsa che non esiste, si verifica un errore 404 e viene visualizzato il file predefinito `/usr/share/apache2/error/HTTP_NOT_FOUND.html.var` . Il file non è presente nella DocumentRoot, ma c'è una direttiva Alias in `/etc/apache2/apache2.conf` che redirige le richieste alla directory /error verso `/usr/share/apache2/error/`.

  Per un elenco delle direttive predefinite ErroDocument, usare il seguente comando:

  ```
  grep ErrorDocument /etc/apache2/apache2.conf
  ```

- Il server, in modo predefinito, scrive il registro dei trasferimenti in `/var/log/apache2/access.log`. È possibile modificare questo valore per ogni singolo sito nella configurazione dell'host virtuale tramite la direttiva *CustomLog* oppure basta ometterla per utilizzare il valore predefinito indicato in `/etc/apache2/apache2.conf`. È anche possibile specificare il file in cui registrare gli errori attraverso la direttiva *ErrorLog* il cui valore predefinito è `/var/log/apache2/error.log`. Questi sono mantenuti separati dal registro dei trasferimenti per semplificare la risoluzione di problemi che possono incorrere col server Apache2. Inoltre, è anche possibile specificare il *LogLevel* (il valore predefinito è "warn") e il *LogFormat* (per il valore predefinito, consultare `/etc/apache2/apache2.conf`).
- Alcune opzioni vengono specificate per directory piuttosto che per server, come la direttiva *Options*. Una stanza "Directory" è racchiusa tra tag in stile XML:

```
<Directory /var/www/mionuovosito>
...
</Directory>
```

La direttiva *Options* all'interno della stanza "Directory" accetta uno o più dei seguenti valori (tra gli altri) separati da spazi:

- **ExecCGI**: consente l'esecuzione di script CGI. Questi script non vengono eseguiti se l'opzione non è selezionata.

La maggior parte dei file non dovrebbe venir eseguita come script CGI, potrebbe essere molto pericoloso. Gli script CGI dovrebbero essere mantenuti in una directory separata, al di fuori della propria DocumentRoot e solo questa directory dovrebbe avere l'opzione ExecCGI impostata. Questo è il comportamento predefinito in Ubuntu e la posizione per gli script CGI è `/usr/lib/cgi-bin`.

- **Includes**: consente le inclusioni lato server. Le inclusioni lato server consentono ai file HTML di *includere* altri file. Questa non è un'opzione comune. Per maggiori informazioni, consultare *the Apache2 SSI HOWTO*[4].
- **IncludesNOEXEC**: consente inclusioni lato server, ma disabilita i comandi *#exec* e *#include* negli script CGI.
- **Indexes**: visualizza un elenco formattato dei contenuti della directory se non esiste alcun *DirectoryIndex* (come index.html) nella directory richiesta.

Per motivi di sicurezza, quest'opzione non dovrebbe essere impostata e soprattutto non su DocumentRoot. Abilitare questa opzione con molta cautela solo su alcune directory e nel caso in cui si voglia visualizzare l'intero contenuto della directory.

- **Multiview**: supporta visualizzazioni multiple in base al contenuto, quest'opzione è disabilitata in modo predefinito per ragioni di sicurezza. Per maggiori informazioni, consultare *la documentazione di Apache2*[5].
- **SymLinksIfOwnerMatch**: segue i collegamenti simbolici solamente se il file di arrivo o la directory hanno gli stessi proprietari del collegamento.

1.2.3. Impostazioni di httpd

Questa sezione espone alcune delle configurazioni di base del demone httpd.

LockFile: la direttiva LockFile imposta il percorso al file di lock utilizzato quando il server viene compilato con USE_FCNTL_SERIALIZED_ACCEPT o USE_FLOCK_SERIALIZED_ACCEPT. Deve essere conservato nel disco locale. Questo valore dovrebbe essere lasciato invariato a meno che la directory di log non sia localizzata su una condivisione NFS. In questo caso, il valore dovrebbe essere modificato con una posizione sul disco locale e una directory accessibile solamente dall'utente root.

PidFile: la direttiva PidFile imposta il file in cui il server registra il proprio «pid». Questo file dovrebbe essere leggibile solamente dall'utente root. Nella maggior parte dei casi può essere lasciata invariata.

User: la direttiva User imposta lo «userid» utilizzato dal server in modo tale che risponda alle richieste. Questa impostazione determina l'accesso al server. Qualsiasi file non accessibile a questo utente è inaccessibile anche a chi cerca di visitare il sito. Il valore predefinito è www-data.

A meno che non sia estremamente necessario, non impostare mai la direttiva «User» a root. Utilizzare root con «User» può creare una falla nella sicurezza del server Web.

La direttiva Group è simile alla direttiva User, imposta il gruppo a cui il server è tenuto rispondere. Il gruppo predefinito è anche www-data.

1.2.4. Moduli di Apache2

Apache2 è un server modulare: solo le funzionalità basilari sono incluse nel server principale. È possibile estendere le funzionalità del server attraverso dei moduli che vengono caricati all'interno di Apache2. Un piccolo insieme di moduli è incluso nel server durante la compilazione: se il server è compilato per caricare i moduli dinamicamente, gli stessi moduli possono essere compilati separatamente e aggiungi quando necessario utilizzando la direttiva LoadModule; altrimenti è necessario ricompilare Apache2 per aggiungere o rimuovere i moduli.

La versione di Ubuntu consente il caricamento dinamico dei moduli. Le direttive di configurazione possono essere incluse in base alla presenza di un particolare modulo racchiudendole in un blocco tipo: <*IfModulo*> block.

È quindi possibile installare moduli aggiuntivi di Apache2 e usarli con il server web. Per esempio, per installare il modulo *MySQL Authentication*, in un terminale digitare quanto segue:

```
sudo apt-get install libapache2-mod-auth-mysql
```

Per altri moduli, consultare la directory `/etc/apache2/mods-available`.

Usare l'utilità a2enmod per abilitare un modulo:

```
sudo a2enmod auth_mysql
sudo /etc/init.d/apache2 restart
```

Allo stesso modo, a2dismod disabiliterà un modulo:

```
sudo a2dismod auth_mysql
sudo /etc/init.d/apache2 restart
```

1.3. Configurazione HTTPS

Il modulo mod_ssl aggiunge un'importante caratteristica al server Apache2, l'abilità di cifrare le comunicazioni. In questo modo, quando il browser utilizza la cifratura SSL per le comunicazioni, il prefisso «https://» verrà inserito nell'URL (Uniform Resource Locator) nella barra degli indirizzi.

Il modulo mod_ssl è disponibile nel pacchetto apache2-common. Per abilitare il modulo mod_ssl, eseguire il seguente comando in un terminale:

```
sudo a2enmod ssl
```

Esiste un file di configurazione HTTPS predefinito in `/etc/apache2/sites-available/default-ssl`. Affinché Apache2 possa fornire connessioni HTTPS, sono necessari un *certificato* e una *chiave*. La configurazione HTTPS predefinita utilizza un certificato e una chiave generati attraverso ssl-cert, utili in fase di test, ma da sostituire con con una versione specifica per il sito o il server. Per maggiori informazioni su come generare una chiave e su come procurarsi un certificato, consultare *Sezione 5, «Certificati» [125]*

Per configurare l'HTTPS per Apache2, digitare quanto segue:

```
sudo a2ensite default-ssl
```

Le directory `/etc/ssl/certs` e `/etc/ssl/private` sono le posizioni predefinite. Se si installa il certificato e la chiave in un'altra directory assicurarsi di modificare *SSLCertificateFile* e *SSLCertificateKeyFile* appropriatamente.

Con l'HTTPS configurato, riavviare il servizio per abilitare le nuove impostazioni:

```
sudo /etc/init.d/apache2 restart
```

In base a come è stato ottenuto il certificato, potrebbe essere necessario inserire una passphrase quando viene avviato Apache2.

È possibile accedere alle pagine del server sicuro digitando «https://nome_host/url/» nel barra degli indirizzi del proprio browser.

1.4. Riferimenti

- La *documentazione di Apache2*[6] contiene informazioni dettagliate riguardo le direttive di configurazione di Apache2. Inoltre, per la documentazione ufficiale di Apache2, consultare il pacchetto apache2-doc.
- Per maggiori informazioni riguardo SSL, consultare la *documentazione di Mod SSL*[7].
- Il libro *Apache Cookbook*[8] di O'Reilly è un'ottima risorsa per informazioni su specifiche configurazioni di Apache2.
- Per domande relative alla versione di Ubuntu di Apache2, chiedere nel canale IRC *#ubuntu-server* sul server *freenode.net*[9].
- Una buona risorsa riguardo PHP e MySQL può essere trovata nella *documentazione online*[10].

2. PHP5 - Linguaggio di scripting

PHP è un linguaggio di script universale pensato per lo sviluppo web. Uno script PHP può essere inserito direttamente nel codice HTML. Questa sezione spiega come installare e configurare PHP5 in sistemi Ubuntu con Apache2 e MySQL.

Questa sessione da per scontato che Apache2 e il server MySQL siano installati e configurati. Per maggiori informazioni sull'installazione e sulla configurazione dei due server, consultare la rispettiva documentazione presenti in questo documento.

2.1. Installazione

PHP5 è disponibile su Ubuntu linux.

- Per installare PHP5 è possibile digitare, in un terminale, quanto segue:

  ```
  sudo apt-get install php5 libapache2-mod-php5
  ```

  È possibile eseguire script di PHP5 dalla riga di comando installando il pacchetto php5-cli. Per installare php5-cli è sufficiente eseguire il seguente comando al prompt del terminale:

  ```
  sudo apt-get install php5-cli
  ```

  È possibile inoltre eseguire gli script di PHP5 senza installare il modulo PHP5 di Apache. Per fare ciò, è sufficiente installare il pacchetto php5-cgi digitando il seguente comando al prompt del terminale: **sudo apt-get install php5-cgi**

  Per usare MySQL con PHP5 è necessario installare il pacchetto php5-mysql. Per installare php5-mysql, eseguire il seguente comando al prompt del terminale:

  ```
  sudo apt-get install php5-mysql
  ```

  Allo stesso modo, per usare PostgreSQL con PHP5 è necessario installare il pacchetto php5-pgsql. Per installare php5-pgsql, eseguire il seguente comando al prompt del terminale:

  ```
  sudo apt-get install php5-pgsql
  ```

2.2. Configurazione

Una volta installato PHP5, è possibile eseguire gli script di PHP5 dal browser web. Se il pacchetto php5-cli è installato, è possibile eseguire gli script PHP5 dal prompt dei comandi.

Il server web Apache2 è configurato, in modo predefinito, per eseguire gli script di PHP5. In altre parole, il modulo PHP5 quando viene installato, viene abilitato automaticamente nel server web Apache2. Verificare che i file `/etc/apache2/mods-enabled/php5.conf` e `/etc/apache2/mods-`

`enabled/php5.load` esistano. Se non dovessero esistere, è possibile abilitare il modulo usando il comando **a2enmod**.

Una volta installati i pacchetti relativi a PHP5 e abilitato il modulo PHP5 per Apache2, è necessario riavviare il server web Apache2 per poter eseguire gli script PHP5. Per riavviare il server web, al prompt del terminale, digitare:

```
sudo /etc/init.d/apache2 restart
```

2.3. Test

Per verificare l'installazione, è possibile eseguire la funzione «phpinfo» di PHP5 come segue:

```
<?php
  phpinfo();
?>
```

È sufficiente copiare il contenuto precedente in un file, come `phpinfo.php`, e salvarlo nella directory **DocumentRoot** del server web Apache2. Una volta puntato il browser web all'indirizzo `http://hostname/phpinfo.php`, dovrebbero venir visualizzati i valori di molti parametri di configurazione di PHP5.

2.4. Riferimenti

- Per ulteriori informazioni, consultare la documentazione di *php.net*[11].
- Esistono diversi libri su PHP. O'Reilly dispone di due ottimi libri: *Learning PHP 5*[12] e *PHP Cook Book*[13].
- Consultare anche la *documentazione online*[14].

3. Squid - Server proxy

Squid è un potente proxy cache server che fornisce servizi proxy e cache per HTTP (Hyper Text Transport Protocol), FTP (File Transfer Protocol) e molti altri protocolli di rete. Squid può implementare servizi di caching e proxy anche per richieste SSL (Secure Sockets Layer), caching per ricerche di DNS (Domain Name Server) e fornire un caching trasparente. Squid supporta molti protocolli per il caching come ICP (Internet Cache Protocol), HTCP (Hyper Text Caching Protocol), CARP (Cache Array Routing Protocol) e WCCP (Web Cache Coordination Protocol).

Il server Squid è una valida soluzione per le necessità di caching e proxy, scala dall'utilizzo in un piccolo ufficio fino alla grande impresa, fornendo, attraverso il protocollo SNMP (Simple Network Management Protocol), un meccanismo di controllo e monitoraggio dei parametri critici molto accurato. Nella selezione di un computer da utilizzare come procy Squid dedicato, o come server cache, assicurarsi che il sistema sia equipaggiato con una grande quantità di memoria fisica, dal momento che Squid mantiene un cache in memoria per aumentare le prestazioni.

3.1. Installazione

Per installare il server Squid, da terminale digitare:

```
sudo apt-get install squid
```

3.2. Configurazione

La configurazione di Squid avviene attraverso la modifica di alcune direttive presenti nel file `/etc/squid/squid.conf`. Gli esempi che seguono descrivono alcune delle direttive che possono essere modificate. Per maggiori informazioni sulla configurazione di Squid consultare la sezione «Riferimenti».

Prima di modificare il file di configurazione, è utile farne una copia e proteggerla dalla scrittura così, in caso di necessità, è possibile utilizzare il file originale.

Copiare il file `/etc/squid/squid.conf` e proteggerlo dalla scrittura utilizzando i seguenti comandi:

```
sudo cp /etc/squid/squid.conf /etc/squid/squid.conf.original
sudo chmod a-w /etc/squid/squid.conf.original
```

- Per impostare il server Squid affinché stia in ascolto sulla porta 8888 invece che sulla porta predefinita 3128, modificare la direttiva http_port:

  ```
  http_port 8888
  ```

- Modificare la direttiva visible_hostname per dare a Squid uno specifico hostname. Questo nome non deve essere necessariamente il nome del computer. Nell'esempio seguente è impostato a *weezie*

```
visible_hostname weezie
```

- Using Squid's access control, you may configure use of Internet services proxied by Squid to be available only users with certain Internet Protocol (IP) addresses. For example, we will illustrate access by users of the 192.168.42.0/24 subnetwork only:

  Aggiungere quanto segue alla **fine** della sezione ACL del file `/etc/squid/squid.conf`:

  ```
  acl fortytwo_network src 192.168.42.0/24
  ```

  Quindi aggiungere quanto segue all'**inizio** della sezione http_access del file `/etc/squid/squid.conf`:

  ```
  http_access allow fortytwo_network
  ```

- Utilizzando il sistema di controllo degli accessi di Squid, è possibile configurare l'utilizzo di alcuni servizi internet in proxy con Squid in alcune fasce orarie: L'esempio seguente descrive come consentire agli utenti l'accesso al servizio dalle 9:00 alle 17:00 dal lunedì al venerdì che utilizza la sotto rete 10.1.42.0/42:

  Aggiungere quanto segue alla **fine** della sezione ACL del file `/etc/squid/squid.conf`:

  ```
  acl biz_network src 10.1.42.0/24
  acl biz_hours time M T W T F 9:00-17:00
  ```

  Quindi aggiungere quanto segue all'**inizio** della sezione http_access del file `/etc/squid/squid.conf`:

  ```
  http_access allow biz_network biz_hours
  ```

Una volta apportate le modifiche al file `/etc/squid/squid.conf`, salvarlo e, per rendere effettivi i cambiamenti, riavviare squid utilizzando il comando:

```
sudo /etc/init.d/squid restart
```

3.3. Riferimenti

Sito web di Squid[15]

Ubuntu Wiki Squid[16] page.

[15] http://www.squid-cache.org/

[16] https://help.ubuntu.com/community/Squid

4. Ruby on Rails

Ruby on Rails è un ambiente web open source, per sviluppare applicazioni web che si avvalgono di database. È ottimizzato per la produttività sostenibile del programmatore dato che richiede di scrivere codice favorendo le convenzioni piuttosto che le configurazioni.

4.1. Installazione

Prima di installare Rails è necessario installare Apache e MySQL. Per installare il pacchetto Apache fare riferimento alla *Sezione 1, «HTTPD - Server web Apache2» [141]*, per MySQL fare riferimento alla *Sezione 1, «MySQL» [160]*.

Una volta installati Apache e MySQL, è possibile installare il pacchetto Ruby on Rails.

Per installare i pacchetti base di Ruby, digitare in un terminale il seguente comando:

```
sudo apt-get install rails
```

4.2. Configurazione

Modificare il file di configurazione `/etc/apache2/sites-available/default` per impostare i propri domini.

La prima cosa da cambiare è la direttiva *DocumentRoot*:

```
DocumentRoot /percorso/applicazione/rails/publicc
```

Successivamente, modificare <Directory "/percorso/applicazione/rails/public">:

```
<Directory "/percorso/applicazione/rails/public">
        Options Indexes FollowSymLinks MultiViews ExecCGI
        AllowOverride All
        Order allow,deny
        allow from all
        AddHandler cgi-script .cgi
</Directory>
```

È utile anche abilitare il modulo mod_rewrite di Apache. Per abilitare il modulo mod_rewrite, digitare il seguente comando in un terminale:

```
sudo a2enmod rewrite
```

Infine, è necessario modificare i proprietari delle directory `/percorso/applicazione/rails/public` e `/percorso/applicazione/rails/tmp` con il proprietario usato per eseguire il processo Apache:

```
sudo chown -R www-data:www-data /percorso/applicazione/rails/public
sudo chown -R www-data:www-data /percorso/applicazione/rails/tmp
```

Il server è ora pronto per le applicazioni Ruby on Rails.

4.3. Riferimenti

- Per ulteriori informazioni, consultare il *sito web di Ruby on Rails*[17].
- Anche *Agile Development with Rails*[18] è un'ottima risorsa.
- Another place for more information is the *Ruby on Rails Ubuntu Wiki*[19] page.

5. Apache Tomcat

Apache Tomcat è un "contenitore" web che consente di servire Java Servlets e applicazioni web JSP (Java Server Pages).

The Tomcat 6.0 packages in Ubuntu support two different ways of running Tomcat. You can install them as a classic unique system-wide instance, that will be started at boot time will run as the tomcat6 unpriviledged user. But you can also deploy private instances that will run with your own user rights, and that you should start and stop by yourself. This second way is particularly useful in a development server context where multiple users need to test on their own private Tomcat instances.

5.1. Installazione globale

Per installare il server Tomcat, digitare al prompt quanto segue:

```
sudo apt-get install tomcat6
```

In questo modo verrà installato il server Tomcat con un'applicazione web predefinita che visualizza una semplice pagina "It works".

5.2. Configurazione

I file di configurazione di Tomcat possono essere trovati in `/etc/tomcat6`. In questa sezione verranno spiegate solo alcune modifiche, per maggiori informazioni, consultare la *documentazione di Tomcat 6.0*[20].

5.2.1. Modificare la porta predefinita

Tomcat 6.0 esegue un connettore HTTP sulla porta 8080 e un connettore AJP sulla porta 8009; potrebbe essere utile modificare queste porte per evitare conflitti con altri server all'interno del sistema. Per fare questo, basta modificare le seguenti righe nel file `/etc/tomcat6/server.xml`:

```
<Connector port="8080" protocol="HTTP/1.1"
               connectionTimeout="20000"
               redirectPort="8443" />
...
<Connector port="8009" protocol="AJP/1.3" redirectPort="8443" />
```

5.2.2. Cambiare la JVM usata

Tomcat viene eseguito preferibilmente con OpenJDK-6, quindi con la JVM di Sun e infine con altre JVM. Se sono installate diverse JVM, è possibile impostare quale usare modificando la variabile JAVA_HOME nel file `/etc/default/tomcat6`:

[20] http://tomcat.apache.org/tomcat-6.0-doc/index.html

```
JAVA_HOME=/usr/lib/jvm/java-6-sun
```

5.2.3. Dichiarare utenti e ruoli

Nomi utente, password e ruoli (gruppi) possono essere definiti in un contenitore Servlet. Con Tomcat 6.0 questo è fatto nel file `/etc/tomcat6/tomcat-users.xml`:

```
<role rolename="admin"/>
<user username="tomcat" password="s3cret" roles="admin"/>
```

5.3. Usare le applicazioni web standard di Tomcat

Tomcat dispone di applicazioni web che è possibile installare per documentarsi, per l'amministrazione o solo per dimostrazione.

5.3.1. Documentazione di Tomcat

Il pacchetto tomcat6-docs contiene la documentazione di Tomcat 6.0 sotto forma di applicazione web a cui è possibile accedere all'indirizzo "http://IL_PROPRIO_SERVER:8080/docs". È possibile installare il pacchetto attraverso il seguente comando:

```
sudo apt-get install tomcat6-docs
```

5.3.2. Applicazioni web amministrative di Tomcat

Il pacchetto tomcat6-admin contiene due applicazioni web che possono essere usate per amministrare il server Tomcat attraverso un'interfaccia web. È possibile installarle attraverso il seguente comando:

```
sudo apt-get install tomcat6-admin
```

La prima applicazione è il cosiddetto *manager*, a cui è possibile accedere dall'indirizzo "http://IL_PROPRIO_SERVER:8080/manager/html". È principalmente usata per ottenere informazioni sul server e riavviare le applicazioni web.

L'accesso al *manager* è protetto: è necessario definire un utente con il ruolo di "manager" nel file `/etc/tomcat6/tomcat-users.xml` prima di potervi accedere:

La seconda applicazione è l'*host-manager* a cui è possibile accedere attraverso l'indirizzo "http://IL_PROPRIO_SERVER:8080/host-manager/html". È possibile usarla per creare host virtuali dinamicamente.

Anche l'accesso all'applicazione *host-manager* è protetto: è necessario definire un utente con il ruolo di "admin" nel file `/etc/tomcat6/tomcat-users.xml` prima di potervi accedere.

Per motivi di sicurezza, l'utente tomcat6 non può scrivere nella directory `/etc/tomcat6` e alcune di queste applicazioni di amministrazione (produzione delle applicazioni, creazione di host virtuali)

necessitano di accesso in scrittura in tale directory. Per poter usare queste caratteristiche, eseguire i seguenti comandi per dare agli utenti del gruppo tomcat6 i permessi necessari:

```
sudo chgrp -R tomcat6 /etc/tomcat6
sudo chmod -R g+w /etc/tomcat6
```

5.3.3. Applicazioni web di esempio

Il pacchetto tomcat6-examples contiene due applicazioni web che possono essere usate per verificare o dimostrare le Servlet o le caratteristiche di JSP e sono accessibile dall'indirizzo "http://IL_PROPRIO_SERVER:8080/examples". Per installarle, usare il seguente comando:

```
sudo apt-get install tomcat6-examples
```

5.4. Usare istanze private

Tomcat è spesso usato in ambienti di sviluppo e di test dove usare una singola istanza all'interno del sistema non risulta molto utile ai molteplici utenti che sfruttano il sistema. I pacchetti di Tomcat 6.0 sono dotati di strumenti che facilitano la creazione di istanze dedicate a ogni singolo utente, consentendo, all'interno del sistema, di eseguire (senza i privilegi di root) istanze private e separate usando però sempre le librerie di sistema.

È possibile eseguire le istanze globali e private in parallelo, basta solo che non usino le stesse porte TCP.

5.4.1. Installare il supporto alle istanze private

È possibile installare tutto il necessario per eseguire istanze private attraverso il seguente comando:

```
sudo apt-get install tomcat6-user
```

5.4.2. Creare un'istanza privata

È possibile creare un'istanza privata attraverso il seguente comando:

```
tomcat6-instance-create mia-istanza
```

In questo modo verrà creata una nuova directory `mia-istanza` con tutte le sottodirectory e gli script necessari. Sarà poi possibile installare le librerie comuni nella sottodirectory `lib/` e sviluppare le proprie applicazioni in `webapps/`. Non vi è alcuna applicazione predefinita in questa directory.

5.4.3. Configurare un'istanza privata

I file di configurazione di Tomcat per un'istanza privata sono disponibili nella sottodirectory `conf/`. È necessario modificare, per esempio, il file `conf/server.xml` per modificare le porte predefinite

usate dall'istanza privata di Tomcat per evitare conflitti con altre istanze che potrebbero essere in esecuzione.

5.4.4. Avviare e fermare un'istanza privata

È possibile avviare un'istanza privata utilizzando il seguente comando (si presuppone che l'istanza sia posizionata nella directory `mia-istanza`):

```
mia-istanza/bin/startup.sh
```

Controllare la sottodirectory `logs/` per la presenza di errori. Se si nota un errore del tipo "*java.net.BindException: Address already in use<null>:8080*", significa che la porta in uso è già utilizzata ed è necessario modificarla.

Per fermare un'istanza, usare il seguente comando (si presuppone che l'istanza sia posizionata nella directory `mia-istanza`):

```
mia-istanza/bin/shutdown.sh
```

5.5. Riferimenti

- Per ulteriori informazioni, consultare il *sito web di Apache Tomcat*[21].
- Il libro *Tomcat: The Definitive Guide*[22] è un'ottima risorsa per creare siti web con Tomcat.
- Per ulteriori libri, consultare la pagina *Tomcat Books*[23].
- Also, see the*Ubuntu Wiki Apache Tomcat*[24] page.

Capitolo 11. Database

Ubuntu fornisce due dei più popolari server database:

- MySQL™
- PostgreSQL

Questi sono disponibili nel repository "main". La seguente sezione descrive come installare e configurare questi database.

1. MySQL

MySQL è un robusto database SQL multi-thread e multi-utente. È concepito per funzionare in situazioni critiche, sistemi a elevato carico e anche per essere inserito in sistemi embedded.

1.1. Installazione

Per installare MySQL, eseguire il seguente comando dal terminale:

```
sudo apt-get install mysql-server
```

Durante l'installazione viene chiesto di inserire un password per l'utente root di MySQL.

Una volta completata l'installazione, il server MySQL dovrebbe avviarsi automaticamente. È possibile digitare i seguenti comandi in un terminale per controllare se il server è in esecuzione:

```
sudo netstat -tap | grep mysql
```

L'output del comando precedente dovrebbe essere:

```
tcp        0      0 localhost:mysql         *:*                     LISTEN      2556/mysqld
```

Se il server non funziana correttamente, è possibile digitare il seguente comando per avviarlo:

```
sudo /etc/init.d/mysql restart
```

1.2. Configurazione

You can edit the `/etc/mysql/my.cnf` file to configure the basic settings -- log file, port number, etc. For example, to configure MySQL to listen for connections from network hosts, change the *bind-address* directive to the server's IP address:

```
bind-address            = 192.168.0.5
```

Sostituire 192.168.0.5 con l'indirizzo appropriato.

Dopo aver apportato una modifica al file `/etc/mysql/my.cnf`, il demone mysql deve essere riavviato:

```
sudo /etc/init.d/mysql restart
```

Per modificare la password di *root* di MySQL, in un terminale digitare:

```
sudo dpkg-reconfigure mysql-server-5.1
```

Il demone mysql viene fermato e viene chiesto di inserire la nuova password.

1.3. Risorse

- Per maggiori informazioni, consultare *il sito web di MySQL*[1].
- Lo *MySQL Handbook* è disponibile nel pacchetto mysql-doc-5.0. Per installarlo, in un terminale, digitare:

  ```
  sudo apt-get install mysql-doc-5.0
  ```

  La documentazione è in formato HTML, per visualizzarlo inserire **file:///usr/share/doc/mysql-doc-5.0/refman-5.0-en.html-chapter/index.html** nella barra degli indirizzi del proprio browser.
- Per informazioni generali riguardo SQL, consultare *Using SQL Special Edition*[2] di Rafe Colburn.
- Ulteriori informazioni sono disponibili nella *documentazione online*[3].

2. PostgreSQL

PostgreSQL è un database relazionale orientato agli oggetti che presenta le caratteristiche di un database commerciale tradizionale e anche miglioramenti dei sistemi DBMS di prossima generazione.

2.1. Installazione

Per installare PostgreSQL, eseguire il seguente comando dal terminale:

```
sudo apt-get install postgresql
```

Una volta che l'installazione è completata, è possibile configurare il server PostgreSQL a seconda delle proprie esigenze, sebbene la configurazione predefinita sia abbastanza buona.

2.2. Configurazione

La connessione via TCP/IP, per configurazione predefinita, è disabilitata. PostgreSQL supporta diversi metodi di autenticazione lato client, quello predefinito per postgres e gli utenti locali è IDENT. Per maggiori informazioni, fare riferimento alla *guida di amministrazione di PostgreSQL*[4].

L'esempio seguente assume che si vogliano abilitare le connessioni TCP/IP e si voglia usare il metodo MD5 per l'autenticazione lato cliet. I file di configurazione di PostgreSQL sono presenti nella directory `/etc/postgresql/<version>/main`: se è installata la versione 8.4 di PostgreSQL, i file di configurazione sono nella directory `/etc/postgresql/8.4/main`.

Per configurare l'autenticazione *ident*, aggiungere le seguenti voci al file `/etc/postgresql/8.4/main/pg_ident.conf`.

Per abilitare le connessioni TCP/IP, modificare il file `/etc/postgresql/8.4/main/postgresql.conf`

Localizzare la riga *#listen_addresses = 'localhost'* e modificarla in:

```
listen_addresses = 'localhost'
```

Per consentire ad altri computer di collegarsi al server PostgreSQL, sostituire "localhost" con l'*indirizzo IP* del server.

Tutti gli altri parametri possono essere modificati, ma bisogna sapere cosa si sta facendo. Per maggiori informazioni, consultare la documentazione di PostgresSQL o fare riferimento ai file di configurazione.

Ora che è possibile collegarsi al server PostgreSQL, è necessario impostare una password per l'utente *postgres*. In un terminale, eseguire il seguente comando per connettersi al modello di database predefinito di PostgreSQL:

[4] http://www.postgresql.org/docs/8.4/static/admin.html

```
sudo -u postgres psql template1
```

Il comando precedente connette al database PostreSQL *template1* come l'utente *postgres*. Una volta collegati al server PostgreSQL, si sarà al prompt SQL. È possibile eseguire il seguente comando SQL al prompt psql per configurare la password per l'utente *postgres*.

```
ALTER USER postgres with encrypted password 'TUA_PASSWORD';
```

Configurata la password, modificare il file `/etc/postgresql/8.4/main/pg_hba.conf` affinché venga usata l'autenticazione *MD5* con l'utente *postgres*:

```
local   all         postgres                          md5
```

Infine, riavviare il servizio PostgreSQL per inizializzare la nuova configurazione. In un terminale, digitare quanto segue per riavviare PostgreSQL:

```
sudo /etc/init.d/postgresql-8.4 restart
```

La configurazione precedente non è completa. Per maggiori informazioni sulla configurazione di altri parametri, fare riferimento alla *guida di amministrazione di PostgreSQL*[5].

2.3. Risorse

- Come detto precedentemente, la *guida di amministrazione*[6] è un'ottima risorsa. La guida è anche disponibile nel pacchetto postgresql-doc-8.4. Per installare il pacchetto, eseguire il seguente comando in un terminale:

  ```
  sudo apt-get install postgresql-doc-8.4
  ```

  Per visualizzare la guida, inserire il seguente URI **file:///usr/share/doc/postgresql-doc-8.4/html/index.html** nella barra degli indirizzi del browser web.
- Per informazioni generali riguardo SQL, consultare *Using SQL Special Edition*[7] di Rafe Colburn.
- Per maggiori informazioni, consultare anche la *documentazione online riguardo PostgreSQL*[8].

Capitolo 12. Applicazioni LAMP

1. Panoramica

Le installazioni LAMP (Linux + Apache + MySQL + PHP) sono molto diffuse sui server Ubuntu ed esistono moltissime applicazioni open source scritte utilizzando questa infrastruttura. Alcune di queste applicazioni sono: wiki, CMS e software di gestione come phpMyAdmin.

One advantage of LAMP is the substantial flexibility for different database, web server, and scripting languages. Popular substitutes for MySQL include PostgreSQL and SQLite. Python, Perl, and Ruby are also frequently used instead of PHP.

L'installazione tradizionale della maggior parte delle applicazioni *LAMP* consiste nel:

- Scaricare un archivio contenente il codice sorgente dell'applicazione.
- Estrarre l'archivio in una directory accessibile a un server web.
- Depending on where the source was extracted, configure a web server to serve the files.
- Configurare l'applicazione affinché si colleghi al database.
- Eseguire uno script o spostarsi su una pagina dell'applicazione per installare il database necessario all'applicazione.
- Completati i passi precedenti, o dei passi simili, è possibile utilizzare l'applicazione.

Esistono anche alcuni svantaggi con questo approccio: i file delle applicazioni non sono organizzati all'interno del file system in modo standard causando confusione sul dove è stata installata l'applicazione. Inoltre, l'aggiornamento dell'applicazione risulta essere complicato: quando viene rilasciata una nuova versione, è necessario ripetere gli stessi passi per l'installazione.

Molte applicazioni *LAMP* sono comunque disponibili all'interno dei repository di Ubuntu e si installano come tutte le altre normali applicazioni. In base però all'applicazione, potrebbe essere necessario apportare alcune configurazioni in più una volta installate.

Questa sezione espone come installare e configurare le applicazioni wiki MoinMoine MediaWiki e il programma per gestire MySQL phpMyAdmin.

Un wiki è un sito web che permette ai visitatori di aggiungere, cancellare e modificare facilmente i contenuti. La facilità di interazione e delle operazioni possibili, rende il wiki uno strumento efficace per la scrittura collaborativa di massa. Il termine wiki è anche riferito al software collaborativo.

2. Moin Moin

MoinMoin è un motore per wiki scritto in Python, basato sul motore «PikiPiki» e rilasciato sotto licenza GNU GPL.

2.1. Installazione

Per installare MoinMoin, eseguire il seguente comando al prompt:

```
sudo apt-get install python-moinmoin
```

È necessario installare anche il server web apache2. Per installare apache-2, consultare la sottosezione *Sezione 1.1, «Installazione» [141]* della sezione *Sezione 1, «HTTPD - Server web Apache2» [141]* .

2.2. Configurazione

Per configurare per la prima volta un wiki, chiamato per esempio *mywiki*, eseguire i seguenti comandi:

```
cd /usr/share/moin
sudo mkdir mywiki
sudo cp -R data mywiki
sudo cp -R underlay mywiki
sudo cp server/moin.cgi mywiki
sudo chown -R www-data.www-data mywiki
sudo chmod -R ug+rwX mywiki
sudo chmod -R o-rwx mywiki
```

È ora necessario configurare MoinMoin affinché identifichi il nuovo wiki *mywiki*. Per configurare MoinMoin, aprire il file `/etc/moin/mywiki.py` e modificare la riga:

```
data_dir = '/org/mywiki/data'
```

in

```
data_dir = '/usr/share/moin/mywiki/data'
```

Inoltre, al di sotto dell'opzione *data_dir*, aggiungere *data_underlay_dir*:

```
data_underlay_dir='/usr/share/moin/mywiki/underlay'
```

Se il file `/etc/moin/mywiki.py` non esiste, è necessario copiare il file `/etc/moin/moinmaster.py` nel file `/etc/moin/mywiki.py` ed eseguire la modifica descritta precedentemente.

Se il nome del wiki è *my_wiki_name*, è necessario inserire nel file `/etc/moin/farmconfig.py` questa riga «("my_wiki_name", r".*")» subito dopo la riga «("mywiki", r".*")».

Una volta configurato MoinMoin per trovare il wiki chiamato *mywiki*, è necessario configurare apache2 in modo che gestisca anche i wiki.

Aggiungere le seguenti righe nel file `/etc/apache2/sites-available/default` all'interno della sezione «<VirtualHost *>»

```
### moin
  ScriptAlias /mywiki "/usr/share/moin/mywiki/moin.cgi"
  alias /moin_static184 "/usr/share/moin/htdocs"
  <Directory /usr/share/moin/htdocs>
  Order allow,deny
  allow from all
  </Directory>
### end moin
```

Modificare *"moin_static184"* nella riga *alias* precedente con la versione di moinmoin installata.

Una volta configurato apache2, è necessario riavviarlo. Per riavviare il server web apache2, digitare:

```
sudo /etc/init.d/apache2 restart
```

2.3. Verifica

Per verificare se l'applicazione wiki funziona, è sufficiente aprire con un browser web il seguente URL:

```
http://localhost/mywiki
```

È possibile eseguire il comando di prova del wiki aprendo con un browser web il seguente URL:

```
http://localhost/mywiki?action=test
```

Per ulteriori dettagli, consultare il sito web di *MoinMoin*[1].

2.4. Riferimenti

- Per maggiori informazioni, consultare il *wiki di MoinMoin*[2].
- Also, see the *Ubuntu Wiki MoinMoin*[3] page.

[1] http://moinmo.in/

3. MediaWiki

MediaWiki è un software per wiki scritto con il linguaggio PHP ed è in grado di utilizzare database come MySQL o PostgreSQL per l'archiviazione dei dati.

3.1. Installazione

Prima di installare MediaWiki è necessario installare Apache2, il linguaggio PHP5 e un sistema di database. MySQL o PostgreSQL sono i più comuni, sceglierne uno in base alle proprie necessità. Per le istruzioni su come installarli, fare riferimento alle relative sezioni all'interno di questa guida.

Per installare MediaWiki, eseguire il seguente comando al prompt:

```
sudo apt-get install mediawiki php5-gd
```

Per maggiori informazioni sulle funzionalità di MediaWiki, consultare il pacchetto mediawiki-extensions.

3.2. Configurazione

Il file di configurazione di Apache `mediawiki.conf` per MediaWiki è installato nella directory `/etc/apache2/conf.d/`. Da questo file, per poter accedere all'applicazione MediaWiki, è utile togliere il commento alla seguente riga.

```
# Alias /mediawiki /var/lib/mediawiki
```

Una volta tolto il commento alla riga precedente, riavviare il server Apache e accedere a MediaWiki utilizzando il seguente URL:

```
http://localhost/mediawiki/config/index.php
```

Consultare la sezione «Checking environment...» presente in quella pagina. È possibile risolvere molti problemi leggendola attentamente.

Once the configuration is complete, you should copy the `LocalSettings.php` file to `/etc/mediawiki` directory:

```
sudo mv /var/lib/mediawiki/config/LocalSettings.php /etc/mediawiki/
```

You may also want to edit `/etc/mediawiki/LocalSettings.php` adjusting:

```
ini_set( 'memory_limit', '64M' );
```

3.3. Estensioni

Le estensioni aggiungono nuove funzionalità a MediaWiki e forniscono agli amministratori del wiki e agli utenti l'abilità di personalizzare MediaWiki in base alle loro necessità.

È possibile scaricare estensioni per MediaWiki come un archivio o direttamente dal repository Subversion copiandolo nella directory `/var/lib/mediawiki/extensions` directory. Alla fine del file aggiungere la seguente riga: `/etc/mediawiki/LocalSettings.php`.

```
require_once "$IP/extensions/ExtentionName/ExtentionName.php";
```

3.4. Riferimenti

- Per maggiori informazioni, consultare il *sito web di MediaWiki*[4].
- La *MediaWiki Administrators' Tutorial Guide*[5] contiene molte informazioni per i nuovi amministratori di MediaWiki.
- Also, the *Ubuntu Wiki MediaWiki*[6] page is a good resource.

4. phpMyAdmin

phpMyAdmin è un'applicazione LAMP sviluppata appositamente per amministrare server MySQL. Scritta in PHP e accessibile attraverso un browser web, fornisce un'interfaccia grafica per svolgere attività di amministrazione su un database.

4.1. Installazione

Prima di poter installare phpMyAdmin, è necessario poter accedere al database MySQL o dallo stesso host in cui phpMyAdmin è installato o da un host accessibile via rete (per maggiori informazioni, consultare *Sezione 1, «MySQL» [160]*). Da un terminale, digitare:

```
sudo apt-get install phpmyadmin
```

Al prompt dei comandi, scegliere quale server web configurare per phpMyAdmin. Nel resto di sezione sezione viene utilizzato Apache2.

All'interno di un browser, nella barra degli indirizzi, scrivere *http://NOMESERVER/phpmyadmin*, sostituendo *NOMESERVER* con il vero nome dell'host. Alla schermata di accesso, scrivere *root* per *username* o un altro utente MySQL e digitare MySQL per la password.

Una volta effettuato l'accesso, è possibile modificare la password di *root*, creare utenti e creare ed eliminare database, tabelle, ecc...

4.2. Configurazione

I file di configurazione di phpMyAdmin sono posizionati in `/etc/phpmyadmin`. Il file principale di configurazione è `/etc/phpmyadmin/config.inc.php` e contiene le opzioni globali di phpMyAdmin.

Per utilizzare phpMyAdmin per l'amministrazione di un database MySQL presente in un altro server, modificare le seguenti opzioni nel file `/etc/phpmyadmin/config.inc.php`:

```
$cfg['Servers'][$i]['host'] = 'db_server';
```

Sostituire *db_server* con il vero nome del server in cui è presente il database remoto oppure con il suo indirizzo IP. Inoltre, assicurarsi che l'host in cui è presente phpMyAdmin possa accedere al database remoto.

Una volta configurato, terminare e ricominciare la sessione di phpMyAdmin per poter accedere al nuovo server.

I file `config.header.inc.php` e `config.footer.inc.php` vengono usati per aggiungere un'intestazione e un pedice HTML a phpMyAdmin.

Un altro importante file di configurazione è `/etc/phpmyadmin/apache.conf`, un collegamento simbolico al file `/etc/apache2/conf.d/phpmyadmin.conf`, usato per configurare Apache2 affinché

visualizzi phpMyAdmin. Nel file sono presenti le direttive per il caricamento di PHP, i permessi per la directory, ecc... Per maggiori informazioni sulla configurazione di Apache2, consultare *Sezione 1, «HTTPD - Server web Apache2» [141]*.

4.3. Riferimenti

- La documentazione di phpMyAdmin è installata automaticamente con il pacchetto ed è possibile accedervi dal collegamento *phpMyAdmin Documentation* (un punto di domanda) al di sotto del logo di phpMyAdmin. La documentazione ufficiale può anche essere visualizzata direttamente dal *sito di phpMyAdmin*[7].
- Inoltre, il libro *Mastering phpMyAdmin*[8] è un'ottima fonte per reperire ulteriori informazioni.
- A third resource is the *phpMyAdmin Ubuntu Wiki*[9] page.

Capitolo 13. Server di file

Se si dispone di più di un computer su una singola rete, a un certo punto potrebbe essere necessario condividere dei file tra questi computer. In questa sezione viene spiegato come installare e configurare servizi come FTP, NFS e CUPS.

1. Server FTP

FTP (File Transfer Protocol) è un protocollo TCP per inviare e scaricare file tra computer. Opera con un modello client/server in cui la parte server è chiamata *demone FTP* e resta in ascolto di eventuali richieste FTP da parte di clienti remoti. Alla ricezione di una richiesta, gestisce l'autenticazione e attiva la connessione. Per la durata della sessione esegue i comandi inviati dal client FTP.

L'accesso a un server FTP può essere gestito in due modi:

- Anonimo
- Con autenticazione

Nella modalità "Anonymous", i client remoti possono accedere al server FTP usando l'account predefinito "anonymous" o "ftp" usando come password un indirizzo email. Nella modalità "Authenticated", un utente deve avere un account e una password. L'accesso alle directory e ai file nel server FTP dipende dai permessi definiti per l'account usato per l'accesso. Come regola generale, il demone FTP nasconde la directory root del server FTP e la modifica con la directory home di FTP, nascondendo così il resto del file system dalle sessioni remote.

1.1. vsftpd - Installazione del server FTP

vsftpd è un demone FTP, facile da installare e configurare. Per installare vsftpd, eseguire il seguente comando:

```
sudo apt-get install vsftpd
```

1.2. Configurazione anonima di FTP

By default vsftpd is *not* configured to only allow anonymous download. If you wish to enable anonymous download edit `/etc/vsftpd.conf` changing:

```
anonymous_enable=Yes
```

During installation a *ftp* user is created with a home directory of `/srv/ftp`. This is the default FTP directory.

If you wish to change this location, to `/srv/files/ftp` for example, simply create a directory in another location and change the *ftp* user's home directory:

```
sudo mkdir /srv/files/ftp
sudo usermod -d /srv/files/ftp ftp
```

Applicate le modifiche, riavviare vsftpd:

```
sudo restart vsftpd
```

Finally, copy any files and directories you would like to make available through anonymous FTP to `/srv/files/ftp`, or `/srv/ftp` if you wish to use the default.

1.3. Configurazione FTP per utenti autenticati

By default vsftpd is configured to authenticate system users and allow them to download files. If you want users to be able to upload files, edit `/etc/vsftpd.conf`:

```
write_enable=YES
```

Riavviare vsftpd:

```
sudo restart vsftpd
```

Ora, quando gli utenti accedono via FTP, il loro punto di partenza sarà la propria directory *home*, dove potranno scaricare e caricare file e creare directory.

Gli utenti anonimi, come impostazione predefinita, non possono caricare alcun file su un server FTP. Per modificare questo comportamento, togliere il commento alla seguente riga e riavviare vsftpd:

```
anon_upload_enable=YES
```

Abilitare il caricamento anonimo di file via FTP può compromettere la sicurezza del sistema. È sconsigliato abilitare il caricamento anonimo su server collegati direttamente a Internet.

Il file di configurazione è composto da diversi parametri di configurazione, le cui informazioni sono disponibili nel file stesso. In alternativa, è possibile fare riferimento alla pagina man (**man 5 vsftpd.conf**).

1.4. FTP sicuro

All'interno del file di configurazione `/etc/vsftpd.conf` di vsftpd, sono presenti molte opzioni per rendere il programma più sicuro. Per esempio, togliendo il commento a quanto segue, gli utenti possono essere limitati all'utilizzo solo della propria directory personale:

```
chroot_local_user=YES
```

È anche possibile limitare un particolare gruppo di utenti all'utilizzo delle sole directory personali:

```
chroot_list_enable=YES
chroot_list_file=/etc/vsftpd.chroot_list
```

Tolto il commento alle opzioni precedenti, creare un file `/etc/vsftpd.chroot_list` con l'elenco degli utenti, uno per riga, quindi riavviare vsftpd:

```
sudo restart vsftpd
```

Inoltre, il file `/etc/ftpusers` contiene un elenco di utenti a cui è *negato* l'accesso FTP. L'elenco comprende gli utenti root, daemon, nobody, ecc... Per disabilitare l'accesso FTP ad altri utenti, aggiungerli semplicemente a questo elenco.

Il protocollo FTP può essere cifrato utilizzando *FTPS*. A differenza di *SFTP*, che è una sessione FTP all'interno di una connessione cifrata con *SSH*, *FTPS* è FTP su SSL (Secure Socket Layer). La principale differenza consiste nel fatto che gli utenti SFTP devono avere un account *shell* sul sistema al posto di una shell *nologin*. Fornire però una shell a tutti gli utenti potrebbe non essere sempre applicabile, come nei casi si servizio di host web.

Per configurare *FTPS*, modificare il file `/etc/vsftpd.conf` aggiungendo:

```
ssl_enable=Yes
```

Inoltre, notare anche le opzioni relative al certificato e alla chiave:

```
rsa_cert_file=/etc/ssl/certs/ssl-cert-snakeoil.pem
rsa_private_key_file=/etc/ssl/private/ssl-cert-snakeoil.key
```

Queste opzioni, come impostazioni predefinita, hanno impostati la chiave e il certificato forniti dal pacchetto ssl-cert. In un ambiente di produzione questi dovrebbero essere sostituiti con un certificato e una chiave generati per lo specifico host. Per maggiori informazioni sui certificati, consultare *Sezione 5, «Certificati» [125]*.

Riavviare vsftpd e gli utenti non-anonimi utilizzeranno *FTPS*:

```
sudo restart vsftpd
```

Per consentire accesso FTP agli utenti dotati di una shell `/usr/sbin/nologin`, ma non dispongono di accesso shell, modificare il file `/etc/shells` aggiungendo *nologin*:

```
# /etc/shells: valid login shells
/bin/csh
/bin/sh
/usr/bin/es
/usr/bin/ksh
/bin/ksh
/usr/bin/rc
/usr/bin/tcsh
/bin/tcsh
/usr/bin/esh
/bin/dash
/bin/bash
/bin/rbash
```

```
/usr/bin/screen
/usr/sbin/nologin
```

Questo è necessario poiché, in modo predefinito, vsftpd utilizza PAM per l'autenticazione e i file di configurazione `/etc/pam.d/vsftpd` contiene:

```
auth    required        pam_shells.so
```

Il modulo *shells* di PAM limita l'accesso alle shell indicate nel file `/etc/shells`.

La maggior parte dei client FTP può essere configurata per utilizzare connessioni FTPS. Il client a riga di comando lftp è in grado di utilizzare FTPS.

1.5. Riferimenti

- Per maggiori informazioni, consultare il *sito web di vsftpd*[1].
- Per maggiori informazioni sulle opzioni disponibili in `/etc/vsftpd.conf`, consultare la *pagina di manuale di vsftpd.conf*[2].
- L'articolo *FTPS vs. SFTP: What to Choose*[3] di CodeGuru ha molte informazioni riguardo FTPS e SFTP.
- Ulteriori informazioni possono essere trovate anche nella *documentazione online*[4].

2. NFS (Network File System)

NSF permette a un sistema di condividere file e directory con altri attraverso una rete. Utilizzando NFS, utenti e programmi possono accedere ai file presenti su sistemi remoti come se fossero dei file locali.

Alcuni dei principali benefici forniti da NFS sono:

- Le workstation locali utilizzano meno spazio su disco perché i dati comuni possono essere memorizzati su una singola macchina, pur rimanendo accessibili agli altri attraverso la rete.
- Gli utenti non devono avere diverse directory home su ciascuna macchina in rete. Le directory home possono risiedere sul server NFS ed essere rese disponibili attraverso la rete.
- I dispositivi di archiviazione come dischi floppy, unità CD-ROM e USB possono essere utilizzate dagli altri computer della rete. Questo può ridurre il numero di unità per supporti rimovibili presenti nella rete.

2.1. Installazione

Per installare il server NFS, inserire il comando seguente a un prompt di terminale:

```
sudo apt-get install nfs-kernel-server
```

2.2. Configurazione

È possibile configurare le directory da esportare aggiungendole al file `/etc/exports`. Per esempio:

```
/ubuntu *(ro,sync,no_root_squash)
/home *(rw,sync,no_root_squash)
```

È possibile sostituire * con uno qualsiasi dei formati per i nomi di host. È necessario rendere la dichiarazione dei nomi di host più specifica possibile per impedire l'accesso di sistemi indesiderati ai mount NFS.

Per avviare il server NFS, è possibile eseguire il seguente comando a un prompt di terminale:

```
sudo /etc/init.d/nfs-kernel-server start
```

2.3. Configurazione client NFS

Utilizzare il comando mount per montare una directory NFS condivisa da un'altra macchina, digitando un comando simile al seguente a un prompt di terminale:

```
sudo mount esempio.nomehost.it:/ubuntu /locale/ubuntu
```

Il punto di mount `/locale/ubuntu` deve esistere. Non ci dovrebbero essere né file, né sottodirectory all'interno di `/locale/ubuntu`.

Un modo alternativo per montare una condivisione NFS da un'altra macchina consiste nell'aggiungere una riga al file `/etc/fstab`. Questa riga deve contenere il nome dell'host del server NFS, la directory esportata dal server e la directory sulla macchina locale dove montare la condivisione NFS.

La sintassi generale per la riga nel file `/etc/fstab` è come segue:

```
esempio.nomehost.it:/ubuntu /locale/ubuntu nfs rsize=8192,wsize=8192,timeo=14,intr
```

Se si hanno problemi nel montare la condivisione NFS, assicurarsi che il pacchetto nfs-common sia installato sul client. Per installare nfs-common, digitare il seguente comando al prompt del terminale:

```
sudo apt-get install nfs-common
```

2.4. Riferimenti

FAQ di NFS per Linux[5]

Documentazione online riguardo NFS[6]

[5] http://nfs.sourceforge.net/

[6] https://help.ubuntu.com/community/NFSv4Howto

3. CUPS - Server di stampa

Il sistema primario e i servizi di stampa di Ubuntu sono gestiti da **Common UNIX Printing System** (CUPS). Questo è un sistema di stampa liberamente disponibile e altamente portabile ed è diventato il nuovo standard per la stampa in molte distribuzioni Linux.

CUPS gestisce lavori e code di stampa, fornisce la stampa in rete tramite l'utilizzo del protocollo IPP (Internet Printing Protocol) e al tempo stesso offre supporto a una nutrita schiera di stampanti, dalle quelle a matrice di punti a quelle al laser (comprese tutte quelle nel mezzo). CUPS supporta anche il PPD (PostScript Printer Detection) e il rilevamento automatico delle stampanti di rete; inoltre fornisce un semplice strumento di amministrazione e configurazione basato sul web.

3.1. Installazione

Per installare CUPS nel proprio computer Ubuntu, basta usare sudo con il comando apt-get e fornire i pacchetti da installare come primo parametro. Un'installazione completa di CUPS ha molte dipendenze di pacchetti, ma possono essere specificati tutti nella stessa riga di comando. Digitare quello che segue al prompt del terminale per installare CUPS:

```
sudo apt-get install cups
```

Dopo essersi autenticati con la propria password utente, i pacchetti dovrebbero essere scaricati e installati. Completato questo processo, il server CUPS viene avviato automaticamente.

Per la risoluzione dei problemi, è possibile accedere alle registrazioni degli errori attraverso il file `/var/log/cups/error_log`. Se non vengono mostrate informazioni sufficienti per risolvere i problemi incontrati, è possibile incrementare la prolissità delle registrazioni del server CUPS modificando la direttiva **LogLevel** nel file di configurazione dal valore predefinito "info" a "debug" oppure "debug2", che registra tutto. Se vengono apportate ulteriori modifiche, ricordarsi di ripristinare i valori iniziali una volta risolto il problema per evitare di ritrovarsi file di registrazione di notevoli dimensioni

3.2. Configurazione

Il comportamento del server CUPS viene configurato attraverso le direttive contenute nel file `/etc/cups/cupsd.conf`. Il file di configurazione di CUPS segue la stessa sintassi del file di configurazione primario del server HTTP Apache. In questo modo, l'utente che ha familiarità con la modifica del file di configurazione di Apache si sentirà a suo agio nella modifica del file di configurazione di CUPS. Di seguito vengono presentati alcuni esempi di impostazioni che potrebbe essere opportuno cambiare fin da subito.

Prima di modificare il file di configurazione, è opportuno creare un copia del file originale e proteggerla da scrittura, in modo da avere le impostazioni originali come riferimento e per riusarle in caso di necessità.

Copiare il file `/etc/cups/cupsd.conf` e proteggerlo dalla scrittura con i seguenti comandi, inseriti a un prompt di terminale.

```
sudo cp /etc/cups/cupsd.conf /etc/cups/cupsd.conf.original
sudo chmod a-w /etc/cups/cupsd.conf.original
```

- **ServerAdmin**: per configurare l'indirizzo email dell'amministratore del server CUPS, modificare il file di configurazione `/etc/cups/cupsd.conf` con un editor di testo e aggiungere o modificare la riga *ServerAdmin*. Per esempio, se si è amministratori del server CUPS e il proprio indirizzo email è "mario@example.net", modificare la riga ServerAdmin in questo modo:

  ```
  ServerAdmin mario@example.net
  ```

- **Listen**: in modo predefinito, su Ubuntu, l'installazione del server CUPS resta in ascolto solamente sull'interfaccia di loopback all'indirizzo IP *127.0.0.1*. Per poter fare in modo che il server CUPS ascolti sull'indirizzo IP della rete, è necessario specificare un nome host, l'indirizzo IP oppure una coppia indirizzo IP/porta con l'aggiunta di una direttiva «Listen». Per esempio, se il server CUPS è all'interno di una rete locale all'indirizzo IP *192.168.10.250* e si desidera renderlo accessibile ad altri sistemi in questa sotto-rete, è necessario modificare il file `/etc/cups/cupsd.conf` e aggiungere una direttiva «Listen» in questo modo:

  ```
  Listen 127.0.0.1:631 # Listen esistente per loopback
  Listen /var/run/cups/cups.sock # socket Listen esistente
  Listen 192.168.10.250:631 # Listen sull'interfaccia LAN, porta 631 (IPP)
  ```

  Nell'esempio precedente, è possibile rendere un commento o rimuovere il riferimento all'indirizzo di loopback (127.0.0.1) se non si desidera che cupsd resti in ascolto su quell'interfaccia, ma che invece resti in ascolto solo sull'interfaccia Ethernet della LAN (Local Area Network). Per abilitare l'ascolto su tutte le interfacce di rete a cui un certo host è collegato, inclusa quella di loopback, è possibile creare una voce Listen per l'host *socrates* come segue:

  ```
  Listen socrates:631 # Listen su tutte le interfacce dell'host "socrates"
  ```

  oppure omettendo la direttiva Listen e utilizzando quella *Port*, come in:

  ```
  Port 631 # Listen sulla porta 631 di tutte le interfacce
  ```

Per ulteriori esempi di direttive di configurazione nel file di configurazione del server CUPS, consultare la pagina manuale associato inserendo il comando seguente a un prompt di terminale:

```
man cupsd.conf
```

Ogni volta che vengono apportati cambiamenti al file di configurazione `/etc/cups/cupsd.conf`, è necessario riavviare il server CUPS digitando il comando seguente a un prompt di terminale:

```
sudo /etc/init.d/cups restart
```

3.3. Interfaccia web

CUPS può essere configurato e monitorato utilizzando un'interfaccia web disponibile all'indirizzo *http://localhost:631/admin*. L'interfaccia web può anche essere usata per svolgere tutte le attività di gestione della stampante.

Per svolgere le attività di amministrazione attraverso l'interfaccia web, è necessario avere l'account root abilitato sul server o aver eseguito l'autenticazione con un utente nel gruppo *lpadmin*. Per motivi di sicurezza, CUPS non autentica gli utenti provi di password.

Per aggiungere un utente al gruppo *lpadmin*, eseguire il seguente comando in un terminale:

```
sudo usermod -aG lpadmin username
```

Maggiore documentazione è disponibile nella scheda *Documentation/Help* dell'interfaccia web.

3.4. Riferimenti

Sito Web di CUPS[7]

Documentazione online riguardo cups[8]

[7] http://www.cups.org/

[8] https://help.ubuntu.com/community/cups

Capitolo 14. Servizi email

Il processo per portare una email da una persona a un'altra all'interno di una rete o attraverso internet, comporta l'utilizzo di diversi sistemi che cooperano tra loro. Ognuno di questi sistemi deve essere configurato correttamente. Colui che spedisce una email utilizza un *Mail User Agent* (MUA), o client email, per spedire il messaggio attraverso uno o più *Mail Transfer Agents* (MTA), l'ultimo dei quali lo consegnerà a un *Mail Delivery Agent* (MDA) per la consegna nella casella di posta del destinatario, che la preleverà utilizzando un client email attraverso un server POP3 o IMAP.

1. Postfix

Postfix è il Mail Transfer Agent (MTA) predefinito di Ubuntu. Cerca di essere facile da amministrare e sicuro ed è compatibile con l'MTA sendmail. Questa sezione espone come installare e configurare postfix e anche come configurare un server SMTP utilizzando un collegamento sicuro (per l'invio di email in sicurezza).

Questa guida non spiega come configurare *Virtual Domains* di Postfix. Per informazioni sui Virtual Domains e altre configurazioni avanzate, consultare *Sezione 1.7.3, «Riferimenti» [189]*.

1.1. Installazione

Per installare postfix eseguire il seguente comando:

```
sudo apt-get install postfix
```

Premere "Invio" quando il processo di installazione pone delle domande, la configurazione verrà effettuata in dettaglio al passo successivo.

1.2. Configurazione di base

Per configurare postfix, eseguire il seguente comando:

```
sudo dpkg-reconfigure postfix
```

Viene visualizzata l'interfaccia utente. In ogni schermata selezionare i seguenti valori:

- Internet Site
- mail.example.com
- steve
- mail.example.com, localhost.localdomain, localhost
- No
- 127.0.0.0/8 [::ffff:127.0.0.0]/104 [::1]/128 192.168.0.0/24
- 0
- +
- tutti

Replace mail.example.com with the domain for which you'll accept email, 192.168.0.0/24 with the actual network and class range of your mail server, and steve with the appropriate username.

A questo punto è utile decidere quale formato usare per la mailbox. Postfix, come impostazione predefinita, utilizza **mbox** come formato. Invece di modificare il file di configurazione, è possibile usare il comando **postconf** per configurare tutti i parametri di postfix che vengono salvati nel file

`/etc/postfix/main.cf`. Per riconfigurare un particolare parametro, è sempre possibile eseguire il comando precedente o modificare il file.

Per configurare la casella di posta per **Maildir**:

```
sudo postconf -e 'home_mailbox = Maildir/'
```

Questo posizionerà le nuove mail in /home/*NOME_UTENTE*/Maildir e sarà quindi necessario configurare il proprio MDA (Mail Delivery Agent) affinché utilizzi lo stesso percorso.

1.3. Autenticazione SMTP

SMTP-AUTH consente a un client di identificarsi attraverso un meccanismo di autenticazione (SASL). TLS (Transport Layer Security) dovrebbe essere usato per cifrare il processo di autenticazione. Una volta autenticato, il server SMTP consentirà ai client di scaricare le email.

1. Configurare Postfix per SMTP-AUTH usando SASL (Dovecot SASL):

```
sudo postconf -e 'smtpd_sasl_type = dovecot'
sudo postconf -e 'smtpd_sasl_path = private/auth-client'
sudo postconf -e 'smtpd_sasl_local_domain ='
sudo postconf -e 'smtpd_sasl_security_options = noanonymous'
sudo postconf -e 'broken_sasl_auth_clients = yes'
sudo postconf -e 'smtpd_sasl_auth_enable = yes'
sudo postconf -e 'smtpd_recipient_restrictions = permit_sasl_authenticated,permit_mynetworks,re
sudo postconf -e 'inet_interfaces = all'
```

La configurazione *smtpd_sasl_path* è un percorso relativo alla directory di Postfix.

2. Ora, ottenere un certificato digitale per TLS (per maggiori informazioni consultare *Sezione 5, «Certificati» [125]*). In questo esempio viene usata anche una Autorità di Certificazione (CA). Per informazioni su come generare un certificato CA, consultare *Sezione 5.5, «Autorità di Certificazione» [127]*.

È possibile ottenere un certificato da un'autorità di certificazione, ma diversamente dai client web, i client SMTP non pongono problemi riguardo i certificati auto-firmati ed è quindi possibile crearne uno. Per maggiori informazioni, consultare *Sezione 5.3, «Creare un certificato auto-firmato» [127]*.

3. Ottenuto un certificato, configurare Postfix affinché fornisca cifratura TLS per le mail in entrate e in uscita:

```
sudo postconf -e 'smtpd_tls_auth_only = no'
sudo postconf -e 'smtp_use_tls = yes'
sudo postconf -e 'smtpd_use_tls = yes'
```

```
sudo postconf -e 'smtp_tls_note_starttls_offer = yes'
sudo postconf -e 'smtpd_tls_key_file = /etc/ssl/private/server.key'
sudo postconf -e 'smtpd_tls_cert_file = /etc/ssl/certs/server.crt'
sudo postconf -e 'smtpd_tls_loglevel = 1'
sudo postconf -e 'smtpd_tls_received_header = yes'
sudo postconf -e 'smtpd_tls_session_cache_timeout = 3600s'
sudo postconf -e 'tls_random_source = dev:/dev/urandom'
sudo postconf -e 'myhostname = mail.example.com'
```

4. Se si sta usando la propria *Autorità di Certificazione* per firmare il certificato, digitare:

   `sudo postconf -e 'smtpd_tls_CAfile = /etc/ssl/certs/cacert.pem'`

   Again, for more details about certificates see *Sezione 5, «Certificati» [125]*.

Una volta eseguiti tutti i comandi, Postfix è configurato per SMTP-AUTH ed è stato creato un certificato auto-firmato per la cifratura TLS.

Ora, il file `/etc/postfix/main.cf` dovrebbe essere simile a *questo*[1].

La configurazione iniziale di postfix è completa, eseguire il seguente comando per riavviarne il demone:

```
sudo /etc/init.d/postfix restart
```

Postfix supports SMTP-AUTH as defined in *RFC2554*[2]. It is based on *SASL*[3]. However it is still necessary to set up SASL authentication before you can use SMTP-AUTH.

1.4. Configurare SASL

Postfix supporta due implementazioni SASL: Cyrus SASL e Dovecot SASL. Per abilitare Dovecot SASL è necessario installare il pacchetto dovecot-common. In un terminale digitare:

```
sudo apt-get install dovecot-common
```

Modificare quindi il file `/etc/dovecot/dovecot.conf`. Nella sezione *auth default* de-commentare l'opzione *socket listen* e modificare come di seguito:

```
socket listen {
  #master {
    # Master socket provides access to userdb information. It's typically
    # used to give Dovecot's local delivery agent access to userdb so it
    # can find mailbox locations.
    #path = /var/run/dovecot/auth-master
    #mode = 0600
```

[1] ../sample/postfix_configuration

[2] http://www.ietf.org/rfc/rfc2554.txt

[3] http://www.ietf.org/rfc/rfc2222.txt

```
    # Default user/group is the one who started dovecot-auth (root)
    #user =
    #group =
  #}
  client {
    # The client socket is generally safe to export to everyone. Typical use
    # is to export it to your SMTP server so it can do SMTP AUTH lookups
    # using it.
    path = /var/spool/postfix/private/auth-client
    mode = 0660
    user = postfix
    group = postfix
  }
}
```

In order to let Outlook clients use SMTP-AUTH, in the *auth default* section of /etc/dovecot/dovecot.conf add *"login"*:

```
mechanisms = plain login
```

Una volta configurato Dovecot, riavviarlo:

```
sudo /etc/init.d/dovecot restart
```

1.5. Postfix-Dovecot

Un'altra opzione per configurare Postfix per SMTP-AUTH consiste nell'usare il pacchetto dovecot-postfix. Questo pacchetto installa Dovecot e configura Postfix sia all'uso di autenticazione SASL che come MDA (Mail Delivery Agent) e configura Dovecot per IMAP, IMAPS, POP3 e POP3S.

You may or may not want to run IMAP, IMAPS, POP3, or POP3S on your mail server. For example, if you are configuring your server to be a mail gateway, spam/virus filter, etc. If this is the case it may be easier to use the above commands to configure Postfix for SMTP-AUTH.

Per installare il pacchetto, in un terminale digitare:

```
sudo apt-get install dovecot-postfix
```

Il server mail dovrebbe essere funzionante, anche se è possibile modificarne ulteriormente la configurazione. Il pacchetto, per esempio, utilizza il certificato e la chiave presi dal pacchetto ssl-cert, ma con un server in produzione dovrebbero essere usati un certificato e una chiave generati appositamente per l'host. Per maggiori informazioni, consultare *Sezione 5, «Certificati» [125]*.

Ottenuto un certificato personalizzato e una chiave per l'host, modificare le seguenti opzioni nel file `/etc/postfix/main.cf`:

```
smtpd_tls_cert_file = /etc/ssl/certs/ssl-mail.pem
smtpd_tls_key_file = /etc/ssl/private/ssl-mail.key
```

Riavviare Postfix:

```
sudo /etc/init.d/postfix restart
```

1.6. Test

La configurazione di SMTP-AUTH è completa ed è ora necessario provarla.

Per verificare se SMTP-AUTH e TLS funzionano correttamente, eseguire il seguente comando:

```
telnet mail.example.com 25
```

Una volta stabilita la connessione al server mail Postfix, digitare:

```
ehlo mail.example.com
```

Se, tra tutte le righe, viene visualizzato anche questo, allora funziona correttamente. Digitare **quit** per uscire.

```
250-STARTTLS
250-AUTH LOGIN PLAIN
250-AUTH=LOGIN PLAIN
250 8BITMIME
```

1.7. Risoluzione problemi

Questa sezione descrive alcuni metodi comuni per determinare la cause dei problemi che potrebbero verificarsi.

1.7.1. Evitare l'uso di chroot

Il pacchetto postfix di Ubuntu viene installato, in modo predefinito e per ragioni di sicurezza, all'interno di un ambiente *chroot*.

Per terminate l'operazione chroot, localizzare la seguente riga nel file `/etc/postfix/master.cf`:

```
smtp      inet  n       -       -       -       -       smtpd
```

e modicarlo come segue:

```
smtp      inet  n       -       n       -       -       smtpd
```

È necessario riavviare Postfix affinché utilizzi la nuova configurazione. In un terminale, digitare:

```
sudo /etc/init.d/postfix restart
```

1.7.2. File di registro

Postfix invia tutti i messaggi di registrazione in `/var/log/mail.log`. I messaggi di errore e gli avvisi possono andar persi nell'output della registrazione normale, per questo vengono anche registrati in `/var/log/mail.err` e `/var/log/mail.warn` rispettivamente.

Per visualizzare in tempo reale i messaggi che vengono registrati, usare il comando tail -f:

```
tail -f /var/log/mail.err
```

Il livello di dettaglio delle registrazioni può essere incrementato. Di seguito vengono riportate alcune opzioni di configurazione per aumentare i dettagli di registrazione in alcune delle aree descritte precedentemente.

- Per aumentare la registrazione delle attività *TLS*, impostare l'opzione *smtpd_tls_loglevel* a un valore compreso tra 1 e 4.

  ```
  sudo postconf -e 'smtpd_tls_loglevel = 4'
  ```
- Se si riscontrano problemi nell'inviare o nel ricevere email da uno specifico dominio, è possibile aggiungere tale dominio al parametro *debug_peer_list*.

  ```
  sudo postconf -e 'debug_peer_list = problem.domain'
  ```
- È possibile incrementare il livello di registrazione di qualsiasi demone Postfix modificando il file `/etc/postfix/master.cf` e aggiungendo *-v* subito dopo la voce. Per esempio, modificare la voce *smtp*:

  ```
  smtp unix  - - - - - smtp -v
  ```

  

  È importante notare che dopo aver apportato una delle modifiche alla registrazione, il processo Postfix deve essere riavviato affinché riconosca la nuova configurazione: **sudo /etc/init.d/postfix reload**
- Per incrementare il livello di informazioni registrate durante la risoluzione di problemi con *SASL*, è possibile impostare le seguenti opzioni nel file `/etc/dovecot/dovecot.conf`

  ```
  auth_debug=yes
  auth_debug_passwords=yes
  ```

  

  Proprio come Postfix, modificando la configurazione di Dovecot il processo deve essere ricaricato: **sudo /etc/init.d/dovecot reload**.

  

  Alcune delle opzioni precedenti possono aumentare drasticamente la quantità di informazioni inviata ai file di registrazione. Ricordarsi di ripristinare il livello di

registrazione al valore predefinito dopo aver corretto il problema, quindi ricaricare il demone appropriato affinché la configurazione abbia effetto.

1.7.3. Riferimenti

Amministrare un server Postfix può essere un compito molto complicato e potrebbe essere necessario richiedere aiuto alla comunità.

Un ottimo punto per richiedere assistenza riguardo Postfix, e per partecipare nella comunità di Ubuntu Server, è il canale IRC *#ubuntu-server* su *freenode*[4]. È anche possibile lasciare un messaggio in uno dei tanti *forum*[5].

Per informazioni dettagliate riguardo Postfix, gli sviluppatori Ubuntu consigliano il libro *The Book of Postfix*[6].

In fine, il *sito web di Postfix*[7] dispone di ottima documentazione riguardo le diverse opzioni di configurazione.

Also, the *Ubuntu Wiki Postifx*[8] page has more information.

[4] http://freenode.net

[5] http://www.ubuntu.com/support/community/webforums

[6] http://www.postfix-book.com/

[7] http://www.postfix.org/documentation.html

[8] https://help.ubuntu.com/community/Postfix

2. Exim4

Exim4 è un MTA (Message Transfer Agent) sviluppato dalla "University of Cambridge" per essere usato sui sistemi Unix collegati a Internet. Exim può essere installato al posto di sendmail, anche se la configurazione di exim è diversa da quella di sendmail.

2.1. Installazione

Per installare exim4, eseguire il seguente comando:

```
sudo apt-get install exim4
```

2.2. Configurazione

Per configurare Exim4, eseguire il seguente comando:

```
sudo dpkg-reconfigure exim4-config
```

Viene visualizzata l'interfaccia che consente di configurare molti dei parametri. Per esempio, in Exim4 i file di configurazione sono divisi in molti piccoli file, per averli tutti raggruppati in un unico file, è possibile farlo attraverso questa interfaccia.

Tutti i parametri configurati tramite l'interfaccia utente vengono salvati nel file `/etc/exim4/update-exim4.conf.conf`. Per eseguire nuovamente la configurazione è sufficiente rieseguire l'assistente alla configurazione o modificare il file con un qualsiasi editor di testo. Una volta configurato, è possibile usare il seguente comando per creare il file di configurazione master:

```
sudo update-exim4.conf
```

Il file di configurazione principale è generato e archiviato in `/var/lib/exim4/config.autogenerated`.

Per nessun motivo modificare il file `/var/lib/exim4/config.autogenerated`. È aggiornato automaticamente ogni volta che viene eseguito il comando **update-exim4.conf**

Per avviare il demone Exim4, eseguire il seguente comando:

```
sudo /etc/init.d/exim4 start
```

2.3. Autenticazione SMTP

Questa sezione descrive come configurare Exim4 affinché usi SMTP-AUTH con TLS e SASL.

Il primo passo è quello di creare un certificato da usare con TLS. In un terminale, digitare quanto segue:

```
sudo /usr/share/doc/exim4-base/examples/exim-gencert
```

Ora è necessario configurare Exim4 per l'utilizzo di TLS modificando il file `/etc/exim4/conf.d/main/03_exim4-config_tlsoptions` e aggiungendo quanto segue:

```
MAIN_TLS_ENABLE = yes
```

È ora necessario configurare Exim4 affinché utilizzi saslauthd per l'autenticazione. Modificare il file `/etc/exim4/conf.d/auth/30_exim4-config_examples` e de-commentare le sezioni *plain_saslauthd_server* e *login_saslauthd_server*:

```
 plain_saslauthd_server:
   driver = plaintext
   public_name = PLAIN
   server_condition = ${if saslauthd{{$auth2}{$auth3}}{1}{0}}
   server_set_id = $auth2
   server_prompts = :
   .ifndef AUTH_SERVER_ALLOW_NOTLS_PASSWORDS
   server_advertise_condition = ${if eq{$tls_cipher}{}{}{*}}
   .endif
#
 login_saslauthd_server:
   driver = plaintext
   public_name = LOGIN
   server_prompts = "Username:: : Password::"
   # don't send system passwords over unencrypted connections
   server_condition = ${if saslauthd{{$auth1}{$auth2}}{1}{0}}
   server_set_id = $auth1
   .ifndef AUTH_SERVER_ALLOW_NOTLS_PASSWORDS
   server_advertise_condition = ${if eq{$tls_cipher}{}{}{*}}
   .endif
```

Infine, aggiornare la configurazione di Exim4 e riavviare il servizio:

```
sudo update-exim4.conf
sudo /etc/init.d/exim4 restart
```

2.4. Configurare SASL

Questa sezione descrive come configurare saslauthd per fornire l'autenticazione per Exim4.

Per prima cosa è necessario installare il pacchetto sasl2-bin. In un terminale, digitare quando segue:

```
sudo apt-get install sasl2-bin
```

Per configurare saslauthd, modificare il file "/etc/default/saslauthd" e impostare START=no a:

```
START=yes
```

Affinché Exim4 possa usare il servizio saslauth, l'utente *Debian-exim* deve far parte del gruppo *sasl*:

```
sudo adduser Debian-exim sasl
```

Ora avviare il servizio saslauthd:

```
sudo /etc/init.d/saslauthd start
```

Exim4 è ora configurato con il supporto a SMTP-AUTH con l'uso dell'autenticazione TLS e SASL.

2.5. Riferimenti

- Per maggiori informazioni, consultare *exim.org*[9].
- È anche disponibile un *libro su Exim4*[10].
- Another resource is the *Exim4 Ubuntu Wiki* [11] page.

3. Server Dovecot

Dovecot è un Mail Delivery Agent progettato per garantire la sicurezza. Supporta la maggior parte dei formati di caselle di posta: mbox o maildir. Questa sezione espone come configurarlo come server imap o pop3.

3.1. Installazione

Per installare dovecot, in un terminale, digitare:

```
sudo apt-get install dovecot-imapd dovecot-pop3d
```

3.2. Configurazione

Per configurare dovecot è possibile modificare il file `/etc/dovecot/dovecot.conf`. È possibile scegliere il protocollo da usare, che può essere pop3, pop3s (pop3 sicuro), imap and imaps (imap sicuro). Una descrizione di questi protocolli va oltre lo scopo di questa guida. Per maggiori informazioni, fare riferimento agli articoli su Wikipedia relativi a *POP3*[12] e *IMAP*[13].

IMAPS e POP3S sono più sicuri dei semplici IMAP e POP3 poiché utilizzano la cifratura SSL per connettersi. Una volta scelto il protocollo, modificare la seguente riga nel file `/etc/dovecot/dovecot.conf`:

```
protocols = pop3 pop3s imap imaps
```

Quindi, scegliere la mailbox che si desidera usare. Dovecot supporta i formati **maildir** e **mbox**, che sono i formati di mailbox più comunemente usati. Entrambi hanno i propri vantaggi, discussi sul *sito web di Dovecot*[14].

Una volta scelta la tipologia della casella di posta, modificare il file `/etc/dovecot/dovecot.conf` e cambiare la seguente riga:

```
mail_location = maildir:~/Maildir # (per maildir)
oppure
mail_location = mbox:~/mail:INBOX=/var/spool/mail/%u # (per mbox)
```

È necessario configurare l'MTA (Mail Transport Agent) per trasferire le mail ricevute in questo tipo di casella di posta se è differente da quella impostata.

Una volta configurato, riavviare il demone dovecot per provare le impostazioni:

[12] http://en.wikipedia.org/wiki/POP3
[13] http://en.wikipedia.org/wiki/Internet_Message_Access_Protocol
[14] http://wiki.dovecot.org/MailboxFormat

```
sudo /etc/init.d/dovecot restart
```

Se è stato abilitato imap o pop3, è possibile provare a eseguire l'accesso con i comandi **telnet localhost pop3** o **telnet localhost imap2**. Se viene visualizzata una schermata simile alla seguente, l'installazione è stata eseguita con successo:

```
telnet localhost pop3
Trying 127.0.0.1...
Connected to localhost.localdomain.
Escape character is '^]'.
+OK Dovecot ready.
```

3.3. Configurazione di Dovecot SSL

Per configurare dovecot affinché utilizzi SSL, è possibile modificare il file `/etc/dovecot/dovecot.conf` e cambiare le seguenti righe:

```
ssl_cert_file = /etc/ssl/certs/ssl-cert-snakeoil.pem
ssl_key_file = /etc/ssl/private/ssl-cert-snakeoil.key
ssl_disable = no
disable_plaintext_auth = no
```

È possibile ottenere il certificato SSL da un'entità di certificazione oppure è possibile creare il proprio certificato auto-firmato. Quest'ultima opzione è valida per le email, dato che i client SMPT solitamente non danno grossi problemi a riguardo. Per maggiori informazioni su come creare un certificato SSL, consultare *Sezione 5, «Certificati» [125]*. Una volta creato, sono disponibili una chiave e un certificato sotto forma di file, copiarli nella posizione puntata all'interno del file `/etc/dovecot/dovecot.conf`.

3.4. Configurazione del firewall per un server email

Per accedere al server mail da un altro computer, è necessario configurare il firewall affinché consenta i collegamenti al server sulle porte necessarie.

- IMAP - 143
- IMAPS - 993
- POP3 - 110
- POP3S - 995

3.5. Riferimenti

- Per maggiori informazioni, consultare il *sito web di Dovecot*[15].
- Also, the *Dovecot Ubuntu Wiki*[16] page has more details.

4. Mailman

Mailman è un programma open source per la gestione di discussioni elettroniche e newsletter. Molte mailing list open source (incluse tutte le mailing list di *Ubuntu*[17]) utilizzano Mailman come software. È molto potente e facile da installare.

4.1. Installazione

Mailman dispone in un'interfaccia web sia per gli amministratori che per gli utenti, utilizzando un server mail esterno per inviare e ricevere le email e si integra perfettamente con i seguenti server mail:

- Postfix
- Exim
- Sendmail
- Qmail

Viene descritto come installare Mailman, il server web Apache e il server mail Postfix o Exim. Per installare Mailman con un server mail diverso, fare riferimento alla sezione «Riferimenti».

È necessario installare solamente un server mail e Postfix è il Mail Transfer Agent predefinito di Ubuntu.

4.1.1. Apache2

To install apache2 you refer to *Sezione 1.1, «Installazione» [141]* for details.

4.1.2. Postfix

Per le istruzioni su come installare e configurare Postfix, consultare *Sezione 1, «Postfix» [183]*

4.1.3. Exim4

Per installare Exim4, consultare refer to *Sezione 2, «Exim4» [190]*.

Once exim4 is installed, the configuration files are stored in the `/etc/exim4` directory. In Ubuntu, by default, the exim4 configuration files are split across different files. You can change this behavior by changing the following variable in the `/etc/exim4/update-exim4.conf` file:

```
dc_use_split_config='true'
```

4.1.4. Mailman

Per installare Mailman, in un terminale, digitare il seguente comando:

[17] http://lists.ubuntu.com

```
sudo apt-get install mailman
```

Questo copia i file di installazione nella directory /var/lib/mailman, gli script CGI nella directory /usr/lib/cgi-bin/mailman, crea l'utente *list* e il gruppo *list*. Il proprietario del processo mailman sarà l'utente creato.

4.2. Configurazione

Questa sezione ha come presupposto l'avvenuta installazione di mailman, apache2 e di postfix o exim4. Ora è solo necessario configurarle.

4.2.1. Apache2

Un file di esempio di Apache è disponibile con Mailman ed è localizzato in `/etc/mailman/apache.conf`. Affinché Apache possa utilizzare il file di configurazione è necessario copiarlo in `/etc/apache2/sites-available`:

```
sudo cp /etc/mailman/apache.conf /etc/apache2/sites-available/mailman.conf
```

In questo modo verrà configurato un nuovo *VirtualHost* per il sito di amministrazione di Mailman. Ora è necessario abilitare la configurazione e riavviare Apache:

```
sudo a2ensite mailman.conf
sudo /etc/init.d/apache2 restart
```

Mailman utilizza apache2 per eseguire gli script CGI. Gli script CGI di mailman sono installati all'interno della directory /usr/lib/cgi-bin/mailman e l'URL di mailman risulta quindi "http://hostname/cgi-bin/mailman/". È possibile apportare cambiamenti al file `/etc/apache2/sites-available/mailman.conf` per modificarne il comportamento.

4.2.2. Postfix

Per l'integrazione di Postfix, verrà associato il dominio "lists.example.com" con le seguenti mailing list. Sostituire *lists.example.com* con il proprio dominio.

È possibile usare il comando postconf per aggiungere la configurazione necessaria in `/etc/postfix/main.cf`:

```
sudo postconf -e 'relay_domains = lists.example.com'
sudo postconf -e 'transport_maps = hash:/etc/postfix/transport'
sudo postconf -e 'mailman_destination_recipient_limit = 1'
```

Controllare che in `/etc/postfix/master.cf` sia presente quanto segue:

```
mailman   unix  -       n       n       -       -       pipe
  flags=FR user=list argv=/usr/lib/mailman/bin/postfix-to-mailman.py
```

```
  ${nexthop} ${user}
```

Invoca lo script *postfix-to-mailman.py* quando viene ricevuta una mail in una lista.

Associare il dominio "lists.example.com" a Mailman con la mappa dei metodi "transport". Modificare il file `/etc/postfix/transport`:

```
lists.example.com      mailman:
```

Ora è necessario far generare a Postfix la mappa "transport" digitando, in un terminale:

```
sudo postmap -v /etc/postfix/transport
```

Riavviare Postfix per abilitare le nuove configurazioni:

```
sudo /etc/init.d/postfix restart
```

4.2.3. Exim4

Una volta installato Exim4, è possibile avviare il server Exim digitando, in un terminale, il seguente comando:

```
sudo /etc/init.d/exim4 start
```

Affinché mailman funzioni con Exim4, è necessario configurare Exim4. Come già spiegato, Exim4 utilizza molteplici file di configurazione di diverse tipologia (per maggiori informazioni, fare riferimento al *sito web di Exim*[18]). Per poter eseguire mailman, è necessario aggiungere un nuovo file di configurazione alle seguenti tipologie di configurazione:

- Main
- Transport
- Router

Exim quindi crea un file di configurazione principale ordinando tutti i file di configurazione: l'ordine di questi file è molto importante.

4.2.4. Main

Tutti i file di configurazione appartenenti al tipo main sono archiviati nella directory `/etc/exim4/conf.d/main/`. È possibile aggiungere il seguente contenuto a un nuovo file di configurazione chiamato `04_exim4-config_mailman`:

```
# start
# Home dir for your Mailman installation -- aka Mailman's prefix
# directory.
# On Ubuntu this should be "/var/lib/mailman"
```

[18] http://www.exim.org

```
# This is normally the same as ~mailman
MM_HOME=/var/lib/mailman
#
# User and group for Mailman, should match your --with-mail-gid
# switch to Mailman's configure script. Value is normally "mailman"
MM_UID=list
MM_GID=list
#
# Domains that your lists are in - colon separated list
# you may wish to add these into local_domains as well
domainlist mm_domains=hostname.com
#
# -=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=
#
# These values are derived from the ones above and should not need
# editing unless you have munged your mailman installation
#
# The path of the Mailman mail wrapper script
MM_WRAP=MM_HOME/mail/mailman
#
# The path of the list config file (used as a required file when
# verifying list addresses)
MM_LISTCHK=MM_HOME/lists/${lc::$local_part}/config.pck
# end
```

4.2.5. Transport

Tutti i file di configurazione appartenenti al tipo transport sono archiviati nella directory `/etc/exim4/conf.d/transport/`. È possibile aggiungere il seguente contenuto a un nuovo file di configurazione chiamato `40_exim4-config_mailman`:

```
  mailman_transport:
   driver = pipe
   command = MM_WRAP \
               '${if def:local_part_suffix \
                    {${sg{$local_part_suffix}{-(\\w+)(\\+.*)?}{\$1}}} \
                    {post}}' \
               $local_part
    current_directory = MM_HOME
    home_directory = MM_HOME
    user = MM_UID
    group = MM_GID
```

4.2.6. Router

Tutti i file di configurazione appartenenti al tipo router sono archiviati nella directory `/etc/exim4/conf.d/router/`. È possibile aggiungere il seguente contenuto a un nuovo file di configurazione chiamato `101_exim4-config_mailman`:

```
  mailman_router:
```

```
driver = accept
require_files = MM_HOME/lists/$local_part/config.pck
local_part_suffix_optional
local_part_suffix = -bounces : -bounces+* : \
                    -confirm+* : -join : -leave : \
                    -owner : -request : -admin
transport = mailman_transport
```

L'ordine dei file di configurazione main e transport può essere qualsiasi. L'ordine dei file di configurazione del tipo router deve essere lo stesso. Questo particolare file deve apparire prima del file `200_exim4-config_primary`. Questi file contengono le stesse informazioni, ma il primo ha la precedenza. Per maggiori informazioni fare riferimento alla sezione «Riferimenti».

4.2.7. Mailman

Una volta installato mailman, è possibile avviarlo usando il seguente comando:

```
sudo /etc/init.d/mailman start
```

Creare quindi la mailing list predefinita. Per crearla, eseguire il seguente comando:

```
sudo /usr/sbin/newlist mailman
```

```
Enter the email address of the person running the list: bhuvan at ubuntu.com
Initial mailman password:
To finish creating your mailing list, you must edit your /etc/aliases (or
equivalent) file by adding the following lines, and possibly running the
`newaliases' program:

## mailman mailing list
mailman: "|/var/lib/mailman/mail/mailman post mailman"
mailman-admin: "|/var/lib/mailman/mail/mailman admin mailman"
mailman-bounces: "|/var/lib/mailman/mail/mailman bounces mailman"
mailman-confirm: "|/var/lib/mailman/mail/mailman confirm mailman"
mailman-join: "|/var/lib/mailman/mail/mailman join mailman"
mailman-leave: "|/var/lib/mailman/mail/mailman leave mailman"
mailman-owner: "|/var/lib/mailman/mail/mailman owner mailman"
mailman-request: "|/var/lib/mailman/mail/mailman request mailman"
mailman-subscribe: "|/var/lib/mailman/mail/mailman subscribe mailman"
mailman-unsubscribe: "|/var/lib/mailman/mail/mailman unsubscribe mailman"

Hit enter to notify mailman owner..

#
```

Postfix o Exim4 sono stati configurati per riconoscere tutte le email di mailman ed è ora obbligatorio creare le nuove voci in `etc/aliases`. Se sono state apportate modifiche ai file di configurazione, assicurarsi di riavviare tali servizi prima di continuare.

Exim4 non utilizza gli alias precedenti per inoltrare le mail a Mailman dato che usa un approccio di tipo *discover*. Per eliminare gli alias quando viene creato l'elenco, è possibile aggiungere la riga *MTA=None* nel file di configurazione di Mailman `/etc/mailman/mm_cfg.py`.

4.3. Amministrazione

Si assume che sia stata fatta un'installazione di base. Gli script cgi di mailman si trovano nella directory /usr/lib/cgi-bin/mailman/. Mailman fornisce uno strumento di amministrazione basato sul web, per accedere alla relativa pagina, aprire con il browser il seguente url:

http://hostname/cgi-bin/mailman/admin

La mailing list di base, *mailman*, comparirà in questa schermata. Facendo clic sul nome della mailing list, verrà richiesta la password di autenticazione. Se viene inserita la password corretta sarà possibile modificare le preferenze di amministrazione di questa mailing list. È possibile creare una nuova mailing list usando l'utilità a riga di comando (**/usr/sbin/newlist**). In alternativa, è possibile creare una nuova mailing list usando l'interfaccia web.

4.4. Utenti

Mailman fornisce un'interfaccia web per gli utenti. Per accedere a questa pagina, indirizzare il browser web al seguente URL:

http://hostname/cgi-bin/mailman/listinfo

La mailing list predefinita, *mailman*, compare a schermo. Facendo clic sul nome, viene presentato il modulo di iscrizione. È possibile inserire il proprio indirizzo email, il nome (opzionale) e la password per completare l'iscrizione. Viene così inviata una email di invito all'indirizzo specificato. È possibile seguire le istruzioni contenute nell'email per completare l'iscrizione.

4.5. Riferimenti

GNU Mailman - Manuale di installazione[19]

HOWTO - Using Exim 4 and Mailman 2.1 together[20]

Also, see the *Mailman Ubuntu Wiki*[21] page.

[19] http://www.list.org/mailman-install/index.html
[20] http://www.exim.org/howto/mailman21.html
[21] https://help.ubuntu.com/community/Mailman

5. Filtrare le email

Uno dei più grandi problemi oggigiorno con le email è lo Unsolicited Bulk Email (UBE). Conosciuto anche come SPAM, questi messaggi possono essere virus e altre forme di malware. Secondi alcuni rapporti, questi messaggi compongono la maggior parte del traffico di email su Internet.

This section will cover integrating Amavisd-new, Spamassassin, and ClamAV with the Postfix Mail Transport Agent (MTA). Postfix can also check email validity by passing it through external content filters. These filters can sometimes determine if a message is spam without needing to process it with more resource intensive applications. Two common filters are opendkim and python-policyd-spf.

- Amavisd-new è un "wrapper" che può chiamare qualsiasi programma di filtraggio per rilevare la posta indesiderata, virus, ecc...
- Spamassassin utilizza moti meccanismi diversi per filtrare le email in base al contenuto del messaggio.
- ClamAV è un antivirus open source.
- opendkim implements a Sendmail Mail Filter (Milter) for the DomainKeys Identified Mail (DKIM) standard.
- python-policyd-spf abilita il controllo Sender Policy Framework (SPF) con Postfix.

Il processo di elaborazione è il seguente:

- Un messaggio email viene accettato da Postfix.
- The message is passed through any external filters opendkim and python-policyd-spf in this case.
- Amavisd-new quindi elabora il messaggio.
- ClamAV analizza il messaggio. Se contiene un virus, Postfix rifiuta il messaggio.
- I messaggi puliti vengono poi analizzati da Spamassassin per verificare che non sia indesiderato. Spamassassin aggiunge quindi una riga X-Header per consentire ad Amavisd-new di analizzare ulteriormente il messaggio.

Per esempio, se un messaggio ha un punteggio spam di oltre 50, questo può essere scartato automaticamente senza nemmeno farlo arrivare al ricevente. Un altro metodo per gestire i messaggi con una segnalazione, è quello di lasciarli arrivare al Mail User Agent (MUA) consentendo all'utente di gestirli come meglio crede.

5.1. Installazione

Per maggiori informazioni sull'installazione e la configurazione di Postfix, consultare *Sezione 1, «Postfix» [183]*.

Per installare le restanti applicazioni, in un terminale, digitare:

```
sudo apt-get install amavisd-new spamassassin clamav-daemon
```

```
sudo apt-get install opendkim python-policyd-spf
```

Esistono dei pacchetti opzionali che si integrano con Spamassassin per rilevare più efficientemente la posta indesiderata:

```
sudo apt-get install pyzor razor
```

Oltre alle applicazioni per il filtraggio, sono necessarie le utilità di compressioni per elaborare alcuni allegati delle email.

```
sudo apt-get install arj cabextract cpio lha nomarch pax rar unrar unzip zip
```

If some packages are not found, check that the *multiverse* repository is enabled in `/etc/apt/sources.list`

If you make changes to the file, be sure to run **sudo apt-get update** before trying to install again.

5.2. Configurazione

Ora è necessario configurare il tutto affinché i programmi funzionino assieme e vengano filtrate le email.

5.2.1. ClamAV

Il comportamento predefinito di ClamAV soddisferà le proprie necessità. Per le altre opzioni di ClamAV, controllare i file di configurazioni presenti in `/etc/clamav`.

Aggiungere l'utente *clamav* al gruppo *amavis* affinché Amavisd-new possa avere accesso per analizzare i file:

```
sudo adduser clamav amavis
```

5.2.2. Spamassassin

Spamassassin rileva automaticamente i componenti opzionali e ne fa uso se sono presenti. Ciò significa che non c'è alcuna necessità di configurare pyzor e razor.

Modificare `/etc/default/spamassassin` per attivare il demone Spamassassin daemon. Cambiare *ENABLED=0* in:

```
ENABLED=1
```

Ora avviare il demone:

```
sudo /etc/init.d/spamassassin start
```

5.2.3. Amavisd-new

Per prima cosa, attivare il rilevamento spam e antivirus in Amavisd-new modificando `/etc/amavis/conf.d/15-content_filter_mode`:

```
use strict;

# You can modify this file to re-enable SPAM checking through spamassassin
# and to re-enable antivirus checking.

#
# Default antivirus checking mode
# Uncomment the two lines below to enable it
#

@bypass_virus_checks_maps = (
   \%bypass_virus_checks, \@bypass_virus_checks_acl, \$bypass_virus_checks_re);


#
# Default SPAM checking mode
# Uncomment the two lines below to enable it
#

@bypass_spam_checks_maps = (
   \%bypass_spam_checks, \@bypass_spam_checks_acl, \$bypass_spam_checks_re);

1;  # insure a defined return
```

rifiutare lo spam e rinviarlo al mittente può essere una cattiva idea, dato che l'indirizzo solitamente è fasullo. Modificare quindi `/etc/amavis/conf.d/20-debian_defaults` per impostare *$final_spam_destiny* a "D_DISCARD" piuttosto che "D_BOUNCE":

```
$final_spam_destiny       = D_DISCARD;
```

Per indicare più messaggi come indesiderati, è possibile utilizzare anche questa opzione:

```
$sa_tag_level_deflt = -999; # add spam info headers if at, or above that level
$sa_tag2_level_deflt = 6.0; # add 'spam detected' headers at that level
$sa_kill_level_deflt = 21.0; # triggers spam evasive actions
$sa_dsn_cutoff_level = 4; # spam level beyond which a DSN is not sent
```

Se il *nome host* del server è diverso dal record MX del dominio è necessario impostare manualmente l'opzione *$myhostname*. Inoltre, se il server riceve email da diversi domini, è necessario personalizzare l'opzione *@local_domains_acl*. Modificare il file `/etc/amavis/conf.d/50-user`:

```
$myhostname = 'mail.example.com';
@local_domains_acl = ( "example.com", "example.org" );
```

Una volta configurato, Amavisd-new deve essere riavviato:

```
sudo /etc/init.d/amavis restart
```

5.2.3.1. Whitelist DKIM

Amavisd-new può essere configurato per inserire automaticamente in una *whitelist* gli indirizzi da domini dotati di "Domain Keys" valide. Nel file `/etc/amavis/conf.d/40-policy_banks` sono disponibili alcuni domini preconfigurati.

L'aggiunta di un dominio nella whitelist è possibile in diversi modi:

- *'example.com' => 'WHITELIST',*: inserisce nella whitelist qualsiasi indirizzo dal dominio "example.com".
- *'.example.com' => 'WHITELIST',*: inserisce nella whitelist qualsiasi indirizzo da qualsiasi *sotto dominio* di "example.com" con una firma valida.
- *'.example.com/@example.com' => 'WHITELIST',*: inserisce nella whitelist i sotto domini di "example.com" che utilizzano una firma del dominio superiore *example.com*.
- *'./@example.com' => 'WHITELIST',*: adds addresses that have a valid signature from "example.com". This is usually used for discussion groups that sign their messages.

A domain can also have multiple Whitelist configurations. After, editing the file restart amavisd-new:

```
sudo /etc/init.d/amavis restart
```

In questo contesto, una volta aggiunto un dominio alla whitelist, il messaggio non verrà più filtrato dall'anti-virus o dal filtro anti-spam. Questo potrebbe essere o meno un comportamento indesiderato per un dominio.

5.2.4. Postfix

Per l'integrazione con Postfix, in un terminale, digitare quanto segue:

```
sudo postconf -e 'content_filter = smtp-amavis:[127.0.0.1]:10024'
```

Ora modificare il file `/etc/postfix/master.cf` e aggiungere quanto segue alla fine:

```
smtp-amavis     unix    -       -       -       -       2       smtp
        -o smtp_data_done_timeout=1200
        -o smtp_send_xforward_command=yes
        -o disable_dns_lookups=yes
        -o max_use=20

127.0.0.1:10025 inet    n       -       -       -       -       smtpd
        -o content_filter=
        -o local_recipient_maps=
```

```
-o relay_recipient_maps=
-o smtpd_restriction_classes=
-o smtpd_delay_reject=no
-o smtpd_client_restrictions=permit_mynetworks,reject
-o smtpd_helo_restrictions=
-o smtpd_sender_restrictions=
-o smtpd_recipient_restrictions=permit_mynetworks,reject
-o smtpd_data_restrictions=reject_unauth_pipelining
-o smtpd_end_of_data_restrictions=
-o mynetworks=127.0.0.0/8
-o smtpd_error_sleep_time=0
-o smtpd_soft_error_limit=1001
-o smtpd_hard_error_limit=1000
-o smtpd_client_connection_count_limit=0
-o smtpd_client_connection_rate_limit=0
-o receive_override_options=no_header_body_checks,no_unknown_recipient_checks
```

Aggiungere anche le seguenti righe dopo il servizio di trasporto "*pickup*":

```
 -o content_filter=
 -o receive_override_options=no_header_body_checks
```

In questo modo si eviteranno i messaggi generati per segnalare lo spam che viene classificato come spam.

Infine riavviare Postfix:

```
sudo /etc/init.d/postfix restart
```

Il filtraggio sul contenuto per lo spam e il rilevamento di virus sono ora abilitati.

5.2.5. Amavisd-new and Spamassassin

When integrating Amavisd-new with Spamassassin, if you choose to disable the bayes filtering by editing `/etc/spamassassin/local.cf` and use cron to update the nightly rules, the result can cause a situation where a large amount of error messages are sent to the *amavis* user via the amavisd-new cron job.

There are several ways to handle this situation:

- Configure your MDA to filter messages you do not wish to see.
- Change `/usr/sbin/amavisd-new-cronjob` to check for *use_bayes 0*. For example, edit `/usr/sbin/amavisd-new-cronjob` and add the following to the top before the *test* statements:

```
egrep -q "^[ \t]*use_bayes[ \t]*0" /etc/spamassassin/local.cf && exit 0
```

5.3. Test

Per prima cosa, verificare che Amavisd-new SMTP sia in ascolto:

```
telnet localhost 10024
Trying 127.0.0.1...
Connected to localhost.
Escape character is '^]'.
220 [127.0.0.1] ESMTP amavisd-new service ready
^]
```

Nell'intestazione dei messaggi che passano attraverso il filtraggio del contenuto, dovrebbe essere visibile:

```
X-Spam-Level:
X-Virus-Scanned: Debian amavisd-new at example.com
X-Spam-Status: No, hits=-2.3 tagged_above=-1000.0 required=5.0 tests=AWL, BAYES_00
X-Spam-Level:
```

L'output potrebbe variare, ma l'aspetto importante è la presenza delle voci *X-Virus-Scanned* e *X-Spam-Status*.

5.4. Risoluzione problemi

Il miglior metodo per comprendere cosa non funzioni correttamente è controllare i file di registro.

- Per istruzioni sulle registrazioni di Postfix, consultare *Sezione 1.7, «Risoluzione problemi» [187]*.
- Amavisd-new fa uso di Syslog per inviare i messaggi verso `/var/log/mail.log`. Il livello di dettaglio può essere aumentato aggiungendo l'opzione *$log_level* in `/etc/amavis/conf.d/50-user` e impostando il valore da 1 a 5.

  ```
  $log_level = 2;
  ```

  

  Quando il livello dei messaggi di registro di Amavisd-new viene aumentato, viene aumentato automaticamente anche quello di Spamassassin.
- Il livello di messaggi di ClamAV può invece essere aumentato modificando il file `/etc/clamav/clamd.conf` e impostando la seguente opzione:

  ```
  LogVerbose true
  ```

  ClamAV, in modo predefinito, invia i messaggi verso `/var/log/clamav/clamav.log`.

Dopo aver cambiato le impostazioni di registrazione di un'applicazione, ricordarsi di riavviare il servizio. Una volta risolto il problema, è buona norma ripristinare il livello di registrazioni originale.

5.5. Riferimenti

Per maggiori informazioni sul filtraggio mail, consultare i seguenti indirizzi:

- *Documentazione di Amavisd-new*[22]

- *Documentation ClamAV*[23] e il *wiki di ClamAV*[24]
- *Wiki di Spamassassin*[25]
- *Sito web di Pyzor*[26]
- *Sito web di Razor*[27]
- *DKIM.org*[28]
- *Postfix Amavis New*[29]

È possibile anche porre le proprie domande nel canale IRC *#ubuntu-server* su *freenode*[30].

[30] http://freenode.net

Capitolo 15. Applicazioni per conversazioni

1. Panoramica

In questa sezione viene discusso come installare e configurare un server IRC (ircd-irc2) e come installare e configurare Jabber, un server di messaggistica istantanea.

2. Server IRC

Nei repository di Ubuntu sono disponibili molti server Internet Relay Chat, ma in questa sezione viene descritto come installare e configurare il server IRC ircd-irc2.

2.1. Installazione

Per installare ircd-irc2, eseguire il seguente comando in un terminale:

```
sudo apt-get install ircd-irc2
```

I file di configurazione sono presenti nella directory `/etc/ircd`, i documenti nella directory `/usr/share/doc/ircd-irc2`.

2.2. Configurazione

Le impostazioni IRC possono essere svolte nel file di configurazione `/etc/ircd/ircd.conf`, dove è possibile impostare il nome host IRC modificando la seguente riga:

```
M:irc.localhost::Debian ircd default configuration::000A
```

Assicurarsi di aggiungere gli alias DNS per il nome host IRC. Per esempio, se il nome host IRC è irc.example.net, assicurarsi che irc.example.net possa essere risolto dal proprio DNS. Il nome host IRC non dovrebbe essere lo stesso del nome host.

I dettagli dell'amministratore possono essere configurati modificando la seguente riga:

```
A:Organization, IRC dept.:Daemon <ircd@example.irc.org>:Client Server::IRCnet:
```

Per configurare le porte IRC da ascoltare, per configurare le credenziali di Operator o l'autenticazione lato client, è necessario aggiungere delle specifiche righe nel file di configurazione. Per maggiori informazioni, fare riferimento al file di configurazione di esempio `/usr/share/doc/ircd-irc2/ircd.conf.example.gz`.

Il messaggio (banner) IRC da visualizzare nei client IRC, quando gli utenti si connettono al server, può essere impostato nel file `/etc/ircd/ircd.motd`.

Una volta apportate le necessarie modifiche al file di configurazione, riavviare il server IRC tramite il seguente comando:

```
sudo /etc/init.d/ircd-irc2 restart
```

2.3. Riferimenti

Potrebbe essere interessante controllare anche altri server IRC disponibili nei repository Ubuntu come ircd-ircu e ircd-hybrid.

- Per maggiori informazioni riguardo il server IRC, consultare le *IRCD FAQ*[1].
- Ulteriori informazioni sono disponibili nella *documentazione online*[2].

3. Server di messaggistica istantanea Jabber

Jabber è un protocollo di messaggistica molto diffuso, basato su XMPP, uno standard aperto per la messaggistica istantanea e usato da molte applicazioni. Questa sezione espone come configurare un server *Jabberd 2* in una rete locale. La configurazione può anche essere adattata per fornire servizi di messaggistica agli utenti attraverso Internet.

3.1. Installazione

Per installare jabberd2, in un terminale digitare:

```
sudo apt-get install jabberd2
```

3.2. Configurazione

A couple of XML configuration files will be used to configure jabberd2 for *Berkeley DB* user authentication. This is a very simple form of authentication. However, jabberd2 can be configured to use LDAP, MySQL, PostgreSQL, etc for for user authentication.

Aprire il file `/etc/jabberd2/sm.xml` e alla riga:

```
<id>jabber.example.com</id>
```

Sostituire *jabber.example.com* con il nome host, o altro identificativo, del proprio server.

Nella sezione <storage>, modificare <driver> in:

```
<driver>db</driver>
```

Modificare il file `/etc/jabberd2/c2s.xml` e nella sezione <*local*> cambiare:

```
<id>jabber.example.com</id>
```

Nella sezione <authreg> sistemare la sezione <module> in

```
<module>db</module>
```

Riavviare jabberd2 per abilitare le nuove impostazioni:

```
sudo /etc/init.d/jabberd2 restart
```

Dovrebbe quindi essere possibile connettersi al server utilizzando un client Jabber come Empathy.

Il vantaggio nell'uso di Berkeley DB per i dati utenti consiste nella bassa manutenzione necessaria una volta configurato. Per avere un maggiore controllo sugli account utente e le credenziali di autenticazione, è consigliato usare un altro metodo di autenticazione.

3.3. Riferimenti

- Il *sito web di Jabberd2*[3] contiene molte informazioni sulla configurazione di Jabberd2.
- For more authentication options see the *Jabberd2 Install Guide*[4].
- Ulteriori informazioni sono disponibili nella *documentazione online*[5].

Capitolo 16. Sistemi per il controllo della versione

Il controllo della versione è l'arte della gestione dell'evolversi delle informazioni. È stato a lungo uno strumento critico per i programmatori, che spendono il loro tempo apportando piccole modifiche al software per poi cancellarle il giorno seguente. Ma l'utilità del software per il controllo della versione va oltre il mondo dello sviluppo di programmi. Ovunque si incontrino persone che utilizzino il computer per gestire informazioni in continuo cambiamento c'è posto per il controllo della versione.

1. Bazaar

Bazaar è un nuovo sistema di controllo della versione sponsorizzato da Canonical, la società commerciale dietro Ubuntu. Diversamente da Subversion e CVS che supportano solamente un modello centralizzato di repository, Bazaar supporta anche un *controllo distribuito della versione*, consentendo alle persone di collaborare più efficientemente. In particolare, Bazaar è progettato per massimizzare il livello di partecipazione della comunità nei progetti open source.

1.1. Installazione

Per installare bzr, in un terminale, digitare:

```
sudo apt-get install bzr
```

1.2. Configurazione

Per introdursi a bzr, usare il comandi *whoami*:

```
$ bzr whoami 'Mario Rossi <mario.rossi@ubuntu.com>'
```

1.3. Imparare a usare Bazaar

La documentazione fornita con Bazaar è installata in /usr/share/doc/bzr/html, il tutorial è un buon punto di partenza. Il comando bzr è dotato di un sistema di aiuto integrato:

```
$ bzr help
```

Per avere maggiori informazioni riguardo il comando *foo*:

```
$ bzr help foo
```

1.4. Integrazione con Launchpad

Anche se è altamente utilizzabile come strumento dedicato, Bazaar è dotato di un'ottima integrazione con *Launchpad*[1], il sistema di sviluppo collaborativo utilizzato da Canonical, e altre comunità di progetti open source, per la gestione di Ubuntu. Per informazioni su come Bazaar possa essere usato con Launchpad per la collaborazione nei progetti open source, consultare *http://bazaar-vcs.org/LaunchpadIntegration*[2].

[1] https://launchpad.net/

[2] http://bazaar-vcs.org/LaunchpadIntegration/

2. Subversion

Subversion è un software open source per il controllo della versione. Utilizzando Subversion è possibile registrare la storia del codice sorgente e dei documenti. È in grado di gestire l'evolversi di file e directory nel tempo. Nel repository centrale viene posizionato un albero di tutti i file. Il repository è come un server di file, tranne per il fatto che si ricorda qualsiasi cambiamento apportato.

2.1. Installazione

Per accedere al repository di Subversion utilizzando il protocollo HTTP, è necessario installare e configurare un server web come Apache2, che funziona molto bene con Subversion. Fare riferimento alla sottosezione HTTP della sezione relativa ad Apache2 per installare e configurare un certificato digitale.

Per installare Subversion, in un terminale, digitare:

```
sudo apt-get install subversion libapache2-svn
```

2.2. Configurazione del server

I passi seguenti presumono siano stati installati i pacchetti elencati in precedenza. Questa sezione descrive come creare un repository con Subversion e come accedere al progetto.

2.2.1. Creare un repository con Subversion

Un repository può essere creato con il seguente comando:

```
svnadmin create /posizione/del/repository/project
```

2.2.2. Importare i file

Una volta creato il repository è possibile *import*arvi file. Per importare una directory, digitare ciò che segue al prompt del terminale:

```
svn import /percorso/della/directory/da/importare file:///percorso/del/repository/
```

2.3. Metodi di accesso

È possibile accedere (checkout) ai repository Subversion in diversi modi, sul disco locale o attraverso diversi protocolli di rete. La posizione di un repository, comunque, è sempre un URL. La tabella illustra come i diversi schemi URL vengono mappati ai diversi metodi di accesso.

Tabella 16.1. Metodi di accesso

Schema	Metodo di accesso
file://	Accesso diretto al repository (sul disco locale)

Schema	Metodo di accesso
http://	Accesso attraverso il protocollo WebDAV al server web Apache2 di Subversion
https://	Come http://, ma con cifratura SSL
svn://	Accesso attraverso un protocollo personalizzato a un server svnserve
svn+ssh://	Come svn://, ma attraverso un tunnel SSH

In questa sezione viene descritto come configurare Subversion per tutti questi metodi. Saranno descritti solo gli elementi basilari. Per maggiori informazioni, fare riferimento al *libro di svn*[3].

2.3.1. Accesso diretto al repository (file://)

Questo è il metodo di accesso più semplice. Non necessita di alcun server di Subversion in esecuzione e serve per accedere a Subversion dalla stessa macchina in cui è in esecuzione. La sintassi del comando è la seguente:

```
svn co file:///percorso/del/repository/progetto
```

o

```
svn co file://localhost/percorso/del/repository/progetto
```

Se non viene specificato l'host, è necessario utilizzare tre slash (///), due per il protocollo (in questo caso file) e uno è lo slash iniziale del percorso. Se viene specificato l'host, utilizzare due slash (//).

I permessi di accesso al repository dipendono dai permessi impostati nel file system. Se l'utente possiede i permessi di scrittura e lettura, allora potrà eseguire checkout e commit al repository.

2.3.2. Accesso con il protocollo WebDAV (http://)

To access the Subversion repository via WebDAV protocol, you must configure your Apache 2 web server. Add the following snippet between the <*VirtualHost*> and <*/VirtualHost*> elements in `/etc/apache2/sites-available/default`, or another VirtualHost file:

```
<Location /svn>
 DAV svn
 SVNPath /home/svn
 AuthType Basic
 AuthName "Your repository name"
 AuthUserFile /etc/subversion/passwd
 Require valid-user
```

[3] http://svnbook.red-bean.com/

```
</Location>
```

The above configuration snippet assumes that Subversion repositories are created under `/home/svn/` directory using **svnadmin** command. They can be accessible using **http://hostname/svn/repos_name** url.

Per importare o eseguire il "commit" di file nel proprio repository Subversion via HTTP, il repository deve essere di proprietà dell'utente del servizio HTTP. Nei sistemi Ubuntu, solitamente, l'utente del servizio HTTP è **www-data**. Per cambiare il proprietario dei file del repository, digitare il comando seguente in un terminale:

```
sudo chown -R www-data:www-data /percorso/al/repository
```

Modificando il proprietario del repository come **www-data** non sarà più possibile importare o eseguire il "commit" di file nel repository attraverso il comando **svn import file:///** come un qualsiasi utente, ma solo come **www-data**.

Creare il file `/etc/subversion/passwd` che conterrà i dettagli di autenticazione utente. Per creare un file, eseguire il seguente comando al prompt dei comandi (viene creato il file e aggiunto il primo utente):

```
sudo htpasswd -c /etc/subversion/passwd nome_utente
```

Per aggiungere ulteriori utenti, omettere l'opzione "*-c*" poiché questa opzione sostituisce i vecchio file. Usare invece questa forma:

```
sudo htpasswd /etc/subversion/password user_name
```

Verrà richiesta la password. Una volta inserita, l'utente viene aggiunto al file. Ora, per accedere al repository, digitare:

```
svn co http://servername/svn
```

La password viene trasmessa come testo in chiaro. Per evitare attacchi di tipo "password snooping", è necessario utilizzare la cifratura SSL. Per maggiori informazioni fare riferimento alla sezione successiva.

2.3.3. Accesso con protocollo WebDAV protetto da cifratura SSL (https://)

Accedere a un repository Subversion attraverso il protocollo WebDAV con cifratura SSL (https://) è simile a http://, l'unica differenza sta nel dover installare e configurare il certificato digitale nel server web Apache. Per usare SSL con Subversion, aggiungere la precedente configurazione di Apache2 al file `/etc/apache2/sites-available/default-ssl`. Per maggiori informazioni su come configurare Apache2 con SSL, consultare *Sezione 1.3, «Configurazione HTTPS» [147]*.

È possibile installare un certificato digitale emesso da un'autorità certificante o in alternativa è possibile usare un certificato auto-firmato.

I passi seguenti hanno come presupposto l'installazione di un certificato digitale all'interno del server web Apache2. Per accedere a un repository Subversion, fare riferimento alla sezione precedente. I metodi di accesso sono esattamente gli stessi tranne per il protocollo, in quanto è necessario utilizzare https://.

2.3.4. Accesso con il protocollo personalizzato (svn://)

Una volta creato il repository è possibile configurare il controllo degli accessi modificando il file `/path/to/repos/project/conf/svnserve.conf`. Per esempio, per impostare l'autenticazione, togliere i commenti alle seguenti righe presenti nel file di configurazione:

```
# [general]
# password-db = passwd
```

Dopo aver tolto i commenti alle righe precedenti, è possibile gestire la lista degli utenti nel file passwd. Modificare il file `passwd` presente nella directory e inserire il nuovo utente. La sintassi da usare è la seguente:

```
username = password
```

Per maggiori informazioni fare riferimento al file.

Per accedere a Subversion attraverso il protocollo svn://, sia dalla stessa macchina sia da un'altra macchina, avviare svnserver utilizzando il comando svnserve. La sintassi è la seguente:

```
$ svnserve -d --foreground -r /percorso/al/repository
# -d -- daemon mode
# --foreground -- run in foreground (useful for debugging)
# -r -- root of directory to serve

Per ulteriori dettagli sull'utilizzo fare riferimento a:
$ svnserve --help
```

Una volta eseguito questo comando, Subversion si mette in ascolto sulla porta predefinita (3690). Per accedere al repository del progetto, è necessario eseguire, da un terminale, il seguente comando:

```
svn co svn://hostname/project project --username nome_utente
```

In base alla configurazione del server, verrà richiesta la password. Una volta autenticati, viene eseguito il check out del codice dal repository di Subversion. Per sincronizzare il repository del progetto con la copia locale, è possibile eseguire il comando **update**. La sintassi del comando è la seguente:

```
cd DIRECTORY_DEL_PROGETTO ; svn update
```

Per maggiori informazioni sui sotto comandi di Subversion fare riferimento al manuale. Per esempio, per informazioni sul comando co (checkout), al prompt dei comandi digitare:

```
svn co help
```

2.3.5. Accesso con protocollo personalizzato a cifratura SSL (svn+ssh://)

La configurazione e le procedure sono le medesime del metodo svn:// . Per i dettagli consultare la sezione precedente. Questo passaggio prevede che sia stata seguita la procedura precedente e il server Subversion sia stato avviato con il comando svnserve.

Si suppone che il server ssh sia in esecuzione sulla macchina e che accetti connessioni in entrata. Per una conferma, provare a collegarsi alla macchina attraverso SSH. Se il login viene eseguito, tutto è configurato. In caso contrario configurare SSH.

Il protocollo svn+ssh:// è utilizzato per accedere al repository di Subversion usando la cifratura SSL. I dati che vengono trasmessi sono cifrati con questo metodo. Per accedere al repository del progetto (per esempio attraverso un checkout), utilizzare, con il comando, la sintassi seguente:

```
svn co svn+ssh://hostname/var/svn/repos/project
```

È necessario utilizzare il percorso completo (/percorso/al/repository/progetto) per accedere al repository di Subversion utilizzando questo metodo di accesso.

In base alla configurazione del server, viene richiesta la password. Utilizzare la password per il login con SSH. Una volta autenticati, viene eseguito il checkout del codice dal repository di Subverison.

3. Server CVS

CVS è un sistema di controllo della versione che è possibile utilizzare per registrare i cambiamenti al codice sorgente di un programma.

3.1. Installazione

Per installare CVS, eseguire il seguente comando in un terminale:

```
sudo apt-get install cvs
```

Una volta installato cvs, installare xinetd per avviare/fermare il server CVS. In un terminale, digitare quando segue per installare xinetd:

```
sudo apt-get install xinetd
```

3.2. Configurazione

Installato CVS, il repository verrà inizializzato automaticamente e la directory predefinita in cui viene archiviato è /var/lib/cvs. Per modificare questo percorso, eseguire il seguente comando:

```
cvs -d /your/new/cvs/repo init
```

Una volta impostato il repository iniziale, è possibile configurare xinetd per l'avvio del server CVS. È sufficiente copiare queste righe nel file `/etc/xinetd.d/cvspserver`.

```
service cvspserver
{
     port = 2401
     socket_type = stream
     protocol = tcp
     user = root
     wait = no
     type = UNLISTED
     server = /usr/bin/cvs
     server_args = -f --allow-root /var/lib/cvs pserver
     disable = no
}
```

Assicurarsi di modificare il repository nel caso in cui sia stata modificata la directory predefinita del repository (/var/lib/cvs).

Configurato xinetd, è possibile avviare il server CVS eseguendo il seguente comando:

```
sudo /etc/init.d/xinetd restart
```

Per avere la conferma che il server CVS è in esecuzione, digitare il seguente comando:

```
sudo netstat -tap | grep cvs
```

L'output del comando precedente dovrebbe essere:

```
tcp 0 0 *:cvspserver *:* LISTEN
```

A questo punto è possibile aggiungere altri utenti, nuovi progetti e gestire il server CVS.

CVS consente di aggiungere nuovi utenti indipendentemente dal sistema operativo. Il modo più semplice è utilizzare l'utente Linux per CVS, benché presenti dei problemi di sicurezza. Per maggiori informazioni, consultare il manuale di CVS.

3.3. Aggiungere progetti

Questa sezione spiega come aggiungere nuovi progetti a un repository CVS, creare la directory, aggiungervi i documenti necessari e i file sorgente. Per aggiungere un progetto al repository CVS, eseguire le seguenti istruzioni:

```
cd your/project
cvs import -d :pserver:nomeutente@nomehost.it:/var/lib/cvs -m "Importazione del mio progetto nel re
```

È possibile utilizzare la variabile d'ambiente CVSROOT per memorizzare la directory root di CVS. Una volta esportata, si può evitare di utilizzare l'opzione "-d" nel comando precedente.

La stringa *new_project*, è un tag del vendor, mentre la stringa *start* è una stringa di rilascio. Non servono a nulla in questo contesto, ma visto che CVS le richiede, vanno inserite.

Quando si aggiunge un nuovo progetto, l'utente CVS deve avere i permessi di scrittura per il repository CVS (/var/lib/cvs). In modo predefinito, il gruppo src possiede tali permessi. Basta semplicemente aggiungere l'utente a questo gruppo per permettergli di gestire progetti in un repository CVS.

4. Riferimenti

Sito web di Bazaar[4]

Launchpad[5]

Sito web di Subversion[6]

Libro su Subversion[7]

Manuale CVS[8]

Easy Bazaar Ubuntu Wiki page[9]

Ubuntu Wiki Subversion page[10]

[4] http://bazaar-vcs.org/
[5] https://launchpad.net/
[6] http://subversion.tigris.org/
[7] http://svnbook.red-bean.com/
[8] http://ximbiot.com/cvs/manual/cvs-1.11.21/cvs_toc.html
[9] https://help.ubuntu.com/community/EasyBazaar
[10] https://help.ubuntu.com/community/Subversion

Capitolo 17. Reti Windows

Spesso le reti di computer sono costituite da sistemi eterogenei e, sebbene gestire una rete composta interamente da computer con Ubuntu sarebbe certamente divertente, alcuni ambienti di rete debbono essere costituiti da sistemi Ubuntu e Microsoft® Windows® che operano insieme in armonia. Questa sezione della guida di Ubuntu introduce i principi e gli strumenti utilizzati nella configurazione di un server Ubuntu per la condivisione di risorse di rete con computer Windows.

1. Introduzione

Utilizzare Ubuntu in una rete composta da client Windows significa fornire e integrare i servizi tipici degli ambienti Windows. Questi servizi offrono supporto per la condivisione di dati e informazioni riguardo i computer e gli utenti della rete e possono essere classificati, in base alle loro funzionalità, in tre principali categorie:

- **Servizi per la condivisione di file e stampanti**. Utilizzo del protocollo SMB (Server Message Block) per agevolare la condivisione di file, cartelle, volumi e stampanti attraverso la rete.
- **Servizi di directory**. Condivisione di informazioni vitali sui computer e sugli utenti della rete con l'uso di tecnologie come LDAP (Lightweight Directory Access Protocol) e Microsoft Active Directory®.
- **Autenticazione e accesso**. Stabilire l'identità del computer o dell'utente della rete e determinare quali risorse siano accessibili al computer o all'utente tramite i permessi e i privilegi, utilizzando permessi dei file, politiche di gruppo e il servizio di autenticazione Kerberos.

Fortunatamente, i sistemi Ubuntu sono in grado di fornire queste funzionalità ai client Windows, permettendo la condivisione di risorse di rete. Uno dei componenti software principali, incluso nei sistemi Ubuntu per le operazioni di rete con Windows, è la suite SAMBA, che comprende strumenti e applicazioni per server SMB.

Questa sezione della guida server di Ubuntu è un'introduzione all'uso di Samba e a come installare e configurare i pacchetti necessari. Per maggiori informazioni e documentazione su Samba, consultare il *sito web di Samba*[1].

[1] http://www.samba.org

2. Server di file Samba

Una delle opzioni più comuni per mettere in comunicazione computer con Ubuntu e Windows, è quella di configurare Samba come server di file. Questa sezione spiega come configurare un server Samba per la condivisione di file con client Windows.

Il server viene configurato per condividere file con qualsiasi client nella rete senza dover usare una password. Se all'interno del proprio ambiente di lavoro è richiesto un maggior controllo sugli accessi, consultare *Sezione 4, «Sicurezza di un server di file e di stampa Samba» [231]*

2.1. Installazione

Per prima cosa installare il pacchetto samba. Alla riga di comando, digitare:

```
sudo apt-get install samba
```

Questo è quanto. Ora è possibile configurare Samba affinché possa condividere i file.

2.2. Configurazione

Il file principale di configurazione di Samba è localizzato in `/etc/samba/smb.conf` e dispone di molti commenti utili nella configurazione delle varie direttive.

Non tutte le opzioni disponibili sono incluse nel file di configurazione predefinito. Per maggiori informazioni, consultare la pagina man di `smb.conf` oppure «*Samba HOWTO Collection*[2]».

1. Per prima cosa, modificare le seguenti coppie chiave/valore nella sezione *[global]* del file `/etc/samba/smb.conf`:

   ```
   workgroup = ESEMPIO
   ...
   security = user
   ```

   Il parametro *security* è più avanti nella sezione [global] ed è commentato. Inoltre, modificare *ESEMPIO* in modo che rispecchi il proprio ambiente di lavoro.

2. Per la nuova directory da condividere, creare una nuova sezione verso la fine del file oppure togliere il commento a uno degli esempi:

   ```
   [share]
       comment = Condivisione file Ubuntu
       path = /srv/samba/share
       browsable = yes
       guest ok = yes
       read only = no
   ```

```
create mask = 0755
```

- *comment*: una breve descrizione della condivisione. Modificarla in base alle proprie esigenze.
- *path*: il percorso alla directory da condividere.

  Questo esempio utilizza `/srv/samba/sharename` poiché, in base alla *Filesystem Hierarchy Standard (FHS)*, */srv*[3] è la posizione in cui dovrebbero essere tenuti i file relativi ai siti. Tecnicamente, le condivisione Samba possono essere posizionate ovunque all'interno del file system, basta che i permessi siano impostati correttamente. In ogni caso, è raccomandato aderire agli standard.
- *browsable*: abilita i client Windows a esplorare la directory condivisa usando Windows Explorer.
- *guest ok*: consente ai client di connettersi alla condivisione senza dover fornire una password.
- *sola lettura:* determina se la condivisione è di sola lettura o se sono garantiti anche i privilegi di scrittura. I privilegi di scrittura sono consentiti solo quando il valore è *no*, come mostrato nell'esempio. Se il valore è *si*, allora l'accesso alla condivisione è in sola lettura.
- *create mask*: determina i permessi dei nuovi file creati.

3. Ora che Samba è configurato, è necessario creare la directory e modificarne i permessi. Da un terminale digitare:

```
sudo mkdir -p /srv/samba/share
sudo chown nobody.nogroup /srv/samba/share/
```

The *-p* switch tells mkdir to create the entire directory tree if it doesn't exist.

4. Infine, riavviare il servizio samba per abilitare la nuova configurazione:

```
sudo restart smbd
sudo restart nmbd
```

La configurazione precedente fornisce accesso completo a tutti i client nella rete locale. Per una configurazione più sicura, consultare *Sezione 4, «Sicurezza di un server di file e di stampa Samba» [231]*.

Da un client Windows dovrebbe ora essere possibile esplorare il file server Ubuntu e consultare la directory condivisa. Per verificare che tutto funzioni correttamente, provare a creare una directory da Windows.

Per creare ulteriori condivisioni basta creare delle nuove sezioni *[dir]* nel file `/etc/samba/smb.conf` e riavviare *Samba*. Assicurarsi che le directory da condividere esistano e abbiano i permessi impostati correttamente.

The file share named *"[share]"* and the path `/srv/samba/share` are just examples. Adjust the share and path names to fit your environment. It is a good idea to name a share after a

directory on the file system. Another example would be a share name of *[qa]* with a path of `/srv/samba/qa`.

2.3. Risorse

- Per delle configurazioni più dettagliate riguardo Samba, consultare *Samba HOWTO Collection*[4]
- La guida è disponibile anche in *formato cartaceo*[5].
- Il libro *Using Samba*[6] di O'Reilly è un'altra buona lettura.
- La pagina *su Samba*[7] della documentazione.

3. Server di stampa Samba

Un' altra configurazione molto comune di Samba è come condivisione di stampanti installate, localmente o in remoto, su un server Ubuntu. Come *Sezione 2, «Server di file Samba» [226]*, questa sezione spiega come configurare Samba affinché qualsiasi client sulla rete locale possa utilizzare le stampanti installate senza la necessità di fornire nome utente o password.

Per una configurazione più sicura, consultare *Sezione 4, «Sicurezza di un server di file e di stampa Samba» [231]*.

3.1. Installazione

Prima di installare e configurare Samba è utile prima di tutto avere un'installazione funzionante di CUPS. Per maggiori informazioni, consultare *Sezione 3, «CUPS - Server di stampa» [179]*.

Per installare il pacchetto samba, da un terminale digitare:

```
sudo apt-get install samba
```

3.2. Configurazione

Una volta installato samba, modificare il file `/etc/samba/smb.conf`. Cambiare il *workgroup* a un valore appropriato per la propria rete e impostare la chiave *security* a *share*:

```
workgroup = ESEMPIO
...
security = user
```

Nella sezione *[printers]* modificare l'opzione *guest ok* a *yes*:

```
browsable = yes
guest ok = yes
```

Una volta modificato il file `smb.conf`, riavviare Samba:

```
sudo restart smbd
sudo restart nmbd
```

La configurazione predefinita di Samba condividerà automaticamente qualsiasi stampante installata. Basta installare la stampante localmente sui client Windows.

3.3. Risorse

- Per delle configurazioni più dettagliate riguardo Samba, consultare *Samba HOWTO Collection*[8]
- La guida è disponibile anche in *formato cartaceo*[9].

- Il libro *Using Samba*[10] di O'Reilly è un'altra buona lettura.
- Per maggiori informazioni sulla configurazione di CUPS, consultare il *sito web di CUPS*[11].
- La pagina *su Samba*[12] della documentazione.

4. Sicurezza di un server di file e di stampa Samba

4.1. Modalità di sicurezza di Samba

Esistono due livelli di sicurezza disponibili al protocollo CIFS (Common Internet Filesystem): a livello *utente* e a livello *condivisione*. L'implementazione della *modalità di sicurezza* di Samba consente una maggiore flessibilità, fornendo quattro modi per implementare la sicurezza a livello utente e uno per quella a livello condivisione.

- *security = user*: richiede a client di fornire nome utente e password per collegarsi alla condivisione. Gli account di Samba sono separati da quelli di sistema, ma il pacchetto libpam-smbpass consente di sincronizzare utenti e password con il database degli utenti di Samba.
- *security = domain*: questa modalità consente al server Samba di apparire ai client Windows come «Primary Domain Controller» (PDC), «Backup Domain Controller» (BDC) oppure «Domain Member Server» (DMS). Per maggiori informazioni, consultare *Sezione 5, «Samba come controller di dominio» [236]*.
- *security = ADS*: consente al server Samba di unirsi a un dominio «Active Directory» come membro nativo. Per maggiori informazioni, consultare *Sezione 6, «Integrare Samba con Active Directory» [240]*.
- *security = server*: questa modalità non dovrebbe essere usata per motivi di sicurezza. Per maggiori informazioni, consultare la sezione *Server Security*[13] della guida di Samba.
- *security = share*: consente ai client di collegarsi alle condivisioni senza fornire nome utente e password.

La modalità di sicurezza scelta dipende dal proprio ambiente di lavoro e da cosa si vuole ottenere col server Samba.

4.2. Livello di sicurezza utente

Questa sezione spiega come riconfigurare i server di file e di stampa Samba, come spiegato in *Sezione 2, «Server di file Samba» [226]* e *Sezione 3, «Server di stampa Samba» [229]*, affinché richieda l'autenticazione.

Per prima cosa, installare il pacchetto libpam-smbpass che consente di sincronizzare gli utenti di sistema col database degli utenti di Samba:

```
sudo apt-get install libpam-smbpass
```

Se è stato scelto il task *Server Samba* durante l'installazione, il pacchetto libpam-smbpass è già installato.

Aprire il file `/etc/samba/smb.conf` e nella sezione *[share]* modificare:

```
    guest ok = no
```

Riavviare Samba affinché le nuove impostazioni abbiano effetto:

```
sudo restart smbd
sudo restart nmbd
```

Ora, collegandosi alle directory o alle stampanti condivise, verranno richiesti il nome utente e la password.

Se si sceglie di mappare un drive di rete alla condivisione, selezionare la casella di spunta «Reconnect at Logon» affinché sia possibile inserire nome utente e password solo una volta, almeno finché la password non viene cambiata.

4.3. Livello di sicurezza condivisione

Ci sono diverse opzioni disponibili per aumentare la sicurezza di ogni singola directory condivisa. Facendo uso dell'esempio *[share]*, questa sezione illustra alcune di queste opzioni.

4.3.1. Gruppi

I gruppi definiscono un insieme di computer e utenti che godono dei medesimi privilegi di accesso alle risorse condivise, offrendo un alto livello di controllo di questi accessi. Per esempio, se il gruppo *qa* contiene gli utenti *freda*, *danika* e *rob* e viene definito il secondo gruppo *support* che contiene gli utenti *danika*, *jeremy* e *vincent*, allora alcune risorse di rete impostate per concedere l'accesso al gruppo *qa* concedono automaticamente l'accesso anche agli utenti freda, danika e rob, mentre lo negano a jeremy o vincent. Dal momento che l'utente *danika* è membro di entrambi i gruppi *qa* e *support* potrà accedere a tutte le risorse condivise il cui accesso è stato concesso a entrambi i gruppi, gli altri utenti avranno accesso alle risorse esplicitamente assegnate al gruppo di appartenenza.

Samba, in modo predefinito, controlla i gruppi di sistema locali definiti in `/etc/group` per determinare quali utenti appartengono a quali gruppi. Per maggiori informazioni su come aggiungere o rimuovere gruppi, consultare *Sezione 1.2, «Aggiungere e rimuovere utenti» [108]*.

Quando si definiscono i gruppi nel file di configurazione di Samba, `/etc/samba/smb.conf`, la sintassi predefinita è quella di usare il prefisso "@" col nome del gruppo. Per esempio, per definire il gruppo *sysadmin* in una sezione del file `/etc/samba/smb.conf`, bisogna inserire il nome del gruppo come **@sysadmin**.

4.3.2. Permessi dei file

I permessi dei file definiscono i diritti che un computer o un utente ha su una particolare directory, file o insieme di file. Tali permessi possono essere definiti modificando il file `/etc/samba/smb.conf` e specificando i permessi di una condivisione definita.

Per esempio, se è stata definita una condivisione Samba chiamata *share* e si vuole dare il permesso di *sola lettura* al gruppo di utenti conosciuto come *qa*, ma si vuole concedere permesso di scrittura

sulla condivisione al gruppo *sysadmin* e all'utente *vincent*, modificare il file `/etc/samba/smb.conf` e aggiungere quanto segue al di sotto della sezione *[share]*:

```
read list = @qa
write list = @sysadmin, vincent
```

Un altro possibile permesso con Samba consente di usare i permessi *amministrativi* su una particolare risorsa condivisa. Gli utenti con permessi amministrativi possono leggere, scrivere o modificare qualsiasi informazione all'interno della risorsa per cui sono stati abilitati.

Per esempio, per concedere all'utente *melissa* permessi amministrativi all'interno dell'esempio *share*, modificare il file `/etc/samba/smb.conf` e aggiungere quanto segue al di sotto della sezione *[share]*:

```
admin users = melissa
```

Modificato il file `/etc/samba/smb.conf`, riavviare Samba affinché le modifiche abbiano effetto:

```
sudo restart smbd
sudo restart nmbd
```

Affinché *read list* e *write list* funzionino, il modello di sicurezza di Samba *non* deve essere impostato a *security = share*

Ora che Samba è stato configurato per limitare quali gruppi hanno accesso alla directory condivisa, è necessario aggiornare i permessi del file system.

Il sistema dei permessi sui file di Linux non funziona correttamente con le ACL (Access Control List) di Windows NT. In questi casi, nei server Ubuntu, sono disponibili le ACL POSIX che forniscono un controllo più fine. Per esempio, per abilitare le ACL su `/srv` con file system ext3, modificare il file `/etc/fstab` aggiungendo l'opzione *acl*:

```
UUID=66bcdd2e-8861-4fb0-b7e4-e61c569fe17d /srv  ext3    noatime,relatime,acl 0       1
```

Quindi montare nuovamente la partizione:

```
sudo mount -v -o remount /srv
```

L'esempio precedente assume che `/srv` sia in una partizione separata. Se `/srv` o qualsiasi sia il percorso di condivisione, fa parte della partizione `/`, potrebbe essere necessario riavviare il sistema.

Per uguagliare la configurazione precedente di Samba, al gruppo *sysadmin* devono essere dati i permessi di lettura, scrittura e di esecuzione su `/srv/samba/share`, al gruppo *qa* devono essere dati i permessi di lettura ed esecuzione e i file devono essere di proprietà del nome utente *melissa*. In un terminale, digitare quanto segue:

```
sudo chown -R melissa /srv/samba/share/
sudo chgrp -R sysadmin /srv/samba/share/
sudo setfacl -R -m g:qa:rx /srv/samba/share/
```

Il comando setfacl imposta i permessi di *esecuzione* a tutti i file nella directory `/srv/samba/share`. Nel caso non sia desiderato, non eseguire il comando.

Da un client Windows ora dovrebbe essere possibile notare la nuova implementazione dei permessi dei file. Per maggiori informazioni riguardo le ACL POSIX, consultare le pagine di manuale di acl e setfacl.

4.4. Profilo AppArmor Samba

Ubuntu è dotato del modulo di sicurezza AppArmor, che fornisce un controlli di acceso. Il profilo predefinito di AppArmor per Samba deve essere adattato alla propria configurazione. Per maggiori informazioni sull'uso di AppArmor, consultare *Sezione 4, «AppArmor» [121]*.

All'interno del pacchetto apparmor-profiles sono disponibili dei profili predefiniti di AppArmor per `/usr/sbin/smbd` e `/usr/sbin/nmbd`, i binari dei demoni di Samba. Per installare il pacchetto, da un terminale digitare:

```
sudo apt-get install apparmor-profiles
```

Questo pacchetto contiene profili per molti altri binari.

I profili per smbd e nmbd sono, in modo predefinito, nella modalità *complain*, consentendo a Samba di lavorare senza dover modificare il profilo e registrando solamente gli errori. Per impostare il profilo smbd in modalità *enforce* e per far funzionare Samba come di consueto, il profilo deve essere modificato per rispecchiare le directory da condividere.

Modificare il file `/etc/apparmor.d/usr.sbin.smbd` aggiungendo informazioni alla sezione *[share]* dall'esempio del server di file:

```
  /srv/samba/share/ r,
  /srv/samba/share/** rwkix,
```

Ora impostare il profilo in modalità *enforce* e ricaricarlo:

```
sudo aa-enforce /usr/sbin/smbd
cat /etc/apparmor.d/usr.sbin.smbd | sudo apparmor_parser -r
```

Dovrebbe essere possibile leggere, scrivere ed eseguire i file nella directory condivisa come di consuetudine e il binario smbd dovrebbe avere accesso solo ai file e le directory configurati. Assicurarsi di aggiungere una voce per ogni directory che viene configurata alla condivisione. Tutti gli errori verranno registrati in `/var/log/syslog`.

4.5. Risorse

- Per delle configurazioni più dettagliate riguardo Samba, consultare *Samba HOWTO Collection*[14]
- La guida è disponibile anche in *formato cartaceo*[15].
- Il libro *Using Samba*[16] di O'Reilly è un'altra buona lettura.
- Il *capitolo 18*[17] della «Samba HOWTO Collection» è dedicato alla sicurezza.
- Il libro *Using Samba*[18] di O'Reilly è un'altra buona lettura.
- La pagina *su Samba*[19] della documentazione.

5. Samba come controller di dominio

Benché non possa funzionare come un controller di dominio primario (PDC) Active Directory, un server Samba può essere configurato per apparire come un controller di dominio in stile Windows NT4. Uno dei vantaggi di questa configurazione consiste nell'abilità di centralizzare le credenziali di utenti e computer, inoltre, Samba può utilizzare diversi backend per archiviare le informazioni.

5.1. Controller di dominio primario (PDC)

Questa sezione spiega come configurare Samba come controller di dominio primario (PDC) usando il backend predefinito «smbpasswd».

1. Per prima cosa, installare Samba e libpam-smbpass per sincronizzare gli account utente digitando quanto segue in un terminale:

```
sudo apt-get install samba libpam-smbpass
```

2. Configurare Samba modificando il file `/etc/samba/smb.conf`. La variabile *security* dovrebbe essere impostata a *user* e il *workgroup* dovrebbe essere relativo alla propria organizzazione.

```
workgroup = ESEMPIO
...
security = user
```

3. Nelle sezione «Domains» aggiungere o togliere il commento a quanto segue:

```
domain logons = yes
logon path = \\%N\%U\profile
logon drive = H:
logon home = \\%N\%U
logon script = logon.cmd
add machine script = sudo /usr/sbin/useradd -N -g machines -c Machine -d /var/lib/samba -s /
```

Per non usare i profili *roaming*, non togliere il commento alle opzioni *logon home* e *logon path*.

- *domain logons*: fornisce il servizio netlogon facendo in modo che Samba si comporti come un controller di dominio.
- *logon path*: posiziona il profilo degli utenti Windows all'interno della loro directory home. È possibile anche configurare una condivisione *[profiles]* posizionando tutti i profili all'interno di una sola directory.
- *logon drive*: specifica il percorso locale della directory home.
- *logon home*: specifica la posizione della directory home.
- *logon script*: determina quale script eseguire localmente una volta che un utente ha eseguito l'accesso. Lo script deve essere all'interno della condivisione *[netlogon]*.

- *add machine script*: uno script che crea automaticamente lo *Machine Trust Account* necessario per accedere al dominio.

  In questo esempio il gruppo *machines* deve essere creato usando l'utilità addgroup. Per maggiori informazioni, consultare *Sezione 1.2, «Aggiungere e rimuovere utenti» [108]*.

4. Togliere il commento alla condivisione *[homes]* per consentire la mappatura di *logon home*:

```
[homes]
   comment = Home Directories
   browseable = no
   read only = no
   create mask = 0700
   directory mask = 0700
   valid users = %S
```

5. Quando configurato come controller di dominio, è necessario configurare una condivisione *[netlogon]*. Per abilitarla, togliere il commento a:

```
[netlogon]
   comment = Network Logon Service
   path = /srv/samba/netlogon
   guest ok = yes
   read only = yes
   share modes = no
```

Il percorso della condivisione predefinita di *netlogon* è `/home/samba/netlogon`, ma in base allo «Filesystem Hierarchy Standard» (FHS), */srv*[20] è la corretta posizione i cui dovrebbero essere tenuti i file specifici dei siti forniti dal sistema.

6. Creare la directory `netlogon` e un file `logon.cmd` per ora vuoto:

```
sudo mkdir -p /srv/samba/netlogon
sudo touch /srv/samba/netlogon/logon.cmd
```

È possibile inserire qualsiasi comando di logon Windows in `logon.cmd` per personalizzare l'ambiente del client.

7. Restart Samba to enable the new domain controller:

```
sudo restart smbd
sudo restart nmbd
```

8. Lastly, there are a few additional commands needed to setup the appropriate rights.

Con l'utente *root* disabilitato in modo predefinito, per poter inserire una workstation nel dominio, un gruppo di sistema deve essere mappato al gruppo Windows *Domain Admins*. Usando l'utilità net, da un terminale digitare:

```
sudo net groupmap add ntgroup="Domain Admins" unixgroup=sysadmin rid=512 type=d
```

Modificare *sysadmin* con un qualsiasi altro gruppo si voglia usare. Inoltre, l'utente usato per unirsi al dominio deve essere membro del gruppo *sysadmin* oltre al gruppo *admin*. Il gruppo *admin* consente l'utilizzo di sudo.

If the user does not have Samba credentials yet, you can add them with the smbpasswd utility, change the *sysadmin* username appropriately:

```
sudo smbpasswd -a sysadmin
```

Inoltre, è necessario fornire i diritti al gruppo *Domain Admins* per consentire ad *add machine script* (e altre funzioni di amministrazione) di funzionare. Per fare ciò:

```
net rpc rights grant -U sysadmin "EXAMPLE\Domain Admins" SeMachineAccountPrivilege \
SePrintOperatorPrivilege SeAddUsersPrivilege SeDiskOperatorPrivilege SeRemoteShutdownPrivilege
```

9. Dovrebbe ora essere possibile unire i client Windows al dominio come in un dominio NT4 in esecuzione su un server Windows.

5.2. Controller di dominio di backup

Con la presenza di un controller di dominio primario (PDC) all'interno delle rete è utile avere anche un controller di dominio di backup (BDC). In questo modo i client potranno autenticarsi anche nel caso in cui il PDC non sia più disponibile.

Quando si configura Samba come BDC, è necessario avere un metodo di sincronizzazione delle informazioni sugli account con il PDC. A questo scopo è possibile usare scp, rsync oppure LDAP come backend *passdb*.

Il metodo migliore per sincronizzare le informazioni sugli account consiste nell'usare LDAP, poiché entrambi i controller di dominio possono usare le stesse informazioni in tempo reale. Configurare un server LDAP potrebbe essere troppo complicato per un esiguo numero di utenti e computer. Per maggiori informazioni, consultare *Sezione 2, «Samba e LDAP» [75]*.

1. Installare samba e libpam-smbpass. Da un terminale digitare:

```
sudo apt-get install samba libpam-smbpass
```

2. Modificare il file `/etc/samba/smb.conf` e togliere il commento a quanto segue nella sezione *[global]*:

```
workgroup = ESEMPIO
...
security = user
```

3. Nella sezione *Domains* togliere il commento o aggiungere quanto segue:

```
domain logons = yes
domain master = no
```

4. Assicurarsi che un utente abbia i permessi di lettura sui file in `/var/lib/samba`. Per esempio, per consentire agli utenti del gruppo *admin* di eseguire scp sui file, digitare:

```
sudo chgrp -R admin /var/lib/samba
```

5. Sincronizzare gli account utente usando scp per copiare la directory `/var/lib/samba` dal PDC:

```
sudo scp -r NOME_UTENTE@PDC:/var/lib/samba /var/lib
```

Sostituire *NOME_UTENTE* con un nome utente valido e *PDC* con il nome host o l'indirizzo IP del controller di dominio primario.

6. Riavviare samba:

```
sudo restart smbd
sudo restart nmbd
```

È possibile verificare se il controller di dominio di backup è funzionante fermando il demone Samba sul PDC e quindi cercando di eseguire l'accesso su un client Windows all'interno del dominio.

È utile ricordare anche che se è stata configurata l'opzione *logon home* come directory sul PDC e quest'ultimo non è più disponibile, anche l'accesso al drive *home* degli utenti non lo sarà. Per questo motivo è utile configurare *logon home* affinché sia posizionato in un server di file separato da PDC e BDC.

5.3. Risorse

- Per delle configurazioni più dettagliate riguardo Samba, consultare *Samba HOWTO Collection*[21]
- La guida è disponibile anche in *formato cartaceo*[22].
- Il libro *Using Samba*[23] di O'Reilly è un'altra buona lettura.
- Il *capitolo 4*[24] della «Samba HOWTO Collection» spiega come configurare un controller di dominio primario.
- Il *capitolo 5*[25] della «Samba HOWTO Collection» spiega come configurare un controller di dominio di backup.
- La pagina *su Samba*[26] della documentazione.

6. Integrare Samba con Active Directory

6.1. Accedere a una condivisione Samba

Un altro uso di Samba consiste nell'integrarlo all'interno di una rete Windows esistente. Una volta parte di un dominio Active Directory, Samba può fornire servizi di file e stampa agli utenti AD.

Il metodo più facile per unirsi in un dominio AD consiste nell'usare Likewise-open. Per maggiori informazioni, consultare *Sezione 7, «Likewise Open» [243]*.

Una volta parte del dominio, digitare il seguente comando al prompt del terminale:

```
sudo apt-get install samba smbfs smbclient
```

Dato che i pacchetti likewise-open e samba utilizzano file `secrets.tdb` separati, è necessario creare un collegamento simbolico in `/var/lib/samba`:

```
sudo mv /var/lib/samba/secrets.tdb /var/lib/samba/secrets.tdb.orig
sudo ln -s /etc/samba/secrets.tdb /var/lib/samba
```

Aprire il file `/etc/samba/smb.conf` e modificare quanto segue:

```
   workgroup = EXAMPLE
   ...
   security = ads
   realm = EXAMPLE.IT
   ...
   idmap backend = lwopen
   idmap uid = 50-9999999999
   idmap gid = 50-9999999999
```

Riavviare samba affinché le nuove impostazioni abbiano effetto:

```
sudo restart smbd
sudo restart nmbd
```

Dovrebbe essere ora possibile accedere qualsiasi condivisione Samba da un client Windows. Assicurarsi comunque di concedere agli utenti o ai gruppi AD accesso alla directory condivisa. Per maggiori informazioni, consultare *Sezione 4, «Sicurezza di un server di file e di stampa Samba» [231]*.

6.2. Accedere a una condivisione Windows

Ora che il server Samba è parte del dominio Active Directory, è possibile accedere a qualsiasi condivisione server di Windows:

- Per montare una condivisione file di Windows, in un terminale digitare quanto segue:

```
mount.cifs //fs01.example.it/share mount_point
```

  È possibile accedere alle condivisioni su computer non facenti parte del dominio AD, ma sarà necessario fornire un nome utente e una password.

- Per montare la condivisione durante la fase di avvio, aggiungere una voce al file `/etc/fstab`, per esempio:

```
//192.168.0.5/share /mnt/windows cifs auto,username=steve,password=secret,rw 0        0
```

- Un altro modo per copiare i file da un server Windows consiste nell'usare l'utilità smbclient. Per elencare i file presenti in una condivisione Windows:

```
smbclient //fs01.example.it/share -k -c "ls"
```

- Per copiare un file da una condivisione, digitare:

```
smbclient //fs01.example.com/share -k -c "get file.txt"
```

  In questo modo si copierà il file `file.txt` nella directory corrente.

- Per copiare una file nella condivisione:

```
smbclient //fs01.example.it/share -k -c "put /etc/hosts hosts"
```

  In questo modo il file `/etc/hosts` verrà copiato in `//fs01.example.com/share/hosts`.

- L'opzione *-c* usata nei comandi precedenti consente di eseguire il comando smbclient in una sola volta. Questo è utile all'interno di script e per altre operazioni sui file. Per accedere al prompt *smb:* \>, un prompt simile a quello di FTP dove è possibile svolgere normali operazioni su file e directory, digitare:

```
smbclient //fs01.example.it/share -k
```

Sostituire tutte le occorrenze di *fs01.example.it/share*, *//192.168.0.5/share*, *username=steve,password=secret* e *file.txt* con l'indirizzo IP del proprio server, il nome nome host, il nome della condivisione, il nome del file e il nome utente e la password dell'utente a cui è consentito accedere alla condivisione.

6.3. Risorse

Per maggiori informazioni sulle opzioni di smbclient, consultare la pagina di manuale: **man smbclient**, disponibile anche *in rete*[27].

[27] http://manpages.ubuntu.com/manpages/maverick/en/man1/smbclient.1.html

La *pagina di manuale*[28] di mount.cifs contiene ulteriori informazioni.

La pagina *su Samba*[29] della documentazione.

[28] http://manpages.ubuntu.com/manpages/maverick/en/man8/mount.cifs.8.html
[29] https://help.ubuntu.com/community/Samba

7. Likewise Open

Likewise Open semplifica la configurazione necessaria per autenticare un computer Linux in un dominio Active Directory. Il pacchetto likewise-open, basato su winbind, semplifica l'integrazione dell'autenticazione di Ubuntu all'interno di una rete Windows esistente.

7.1. Installazione

Ci sono due metodi per utilizzare Likewise Open, tramite l'utilità a riga di comando likewise-open e likewise-open-gui. Questa sezione prende in esame l'utilità a riga di comando.

Per installare il pacchetto likewise-open, aprire un terminale e digitare:

```
sudo apt-get install likewise-open
```

7.2. Entrare in un dominio

L'eseguibile principale del pacchetto likewise-open è `/usr/bin/domainjoin-cli`, usato per unire il computer all'interno del dominio. Prima di poter unirsi in un dominio è necessario assicurarsi di avere:

- Accesso a un utente Active Directory con i permessi necessari per entrare nel dominio.
- Il *Fully Qualified Domain Name* (FQDN) del dominio a cui unirsi. Se il dominio AD non corrisponde a un dominio AD valido come *example.it*, potrebbe essere nella forma *nomedominio.local*.
- DNS configurati correttamente per il dominio, necessari in un ambiente AD di produzione. È necessario un DNS Microsoft in modo che i client possano determinare che il dominio Active Directory è attivo.

  Se non si dispone di un server DNS Windows all'interno della propria rete, consultare

Per entrare in un dominio, da un terminale, digitare:

```
sudo domainjoin-cli join example.it Administrator
```

Sostituire *example.it* con il nome del proprio dominio e *Administrator* con il nome utente corretto.

Viene quindi chiesta la password utente e se tutto procede correttamente viene stampato sulla console un messaggio di *successo*.

Una volta uniti al dominio, è necessario riavviare prima di tentare l'autenticazione.

Una volta introdotto un computer Ubuntu in un dominio Active Directory, è possibile autenticarsi usando un qualsiasi utente AD valido. Per effettuare l'accesso è necessario inserire il nome come "DOMINIO\NOME_UTENTE", per esempio, per collegarsi via SSH al server nel dominio, digitare:

```
ssh 'example\mario'@nomehost
```

Se si sta configurando un computer desktop, il nome utente deve essere preceduto da *dominio*\ nella finestra di accesso grafico.

Per fare in modo che likewise-open utilizzi un dominio predefinito, è possibile aggiungere quanto segue nel file `/etc/samba/lwiauthd.conf`:

```
winbind use default domain = yes
```

Riavviare quindi i demoni likewise-open:

```
sudo /etc/init.d/likewise-open restart
```

Una volta configurato per un *dominio predefinito* la parte *'dominio*\' non è più necessaria e gli utenti possono collegarsi usando solo il nome utente.

L'utilità domainjoin-cli può essere usata per lasciare il dominio. Da un terminale digitare:

```
sudo domainjoin-cli leave
```

7.3. Altre utilità

Il pacchetto likewise-open dispone di altre utilità che possono rivelarsi utili per ottenere informazioni sull'ambiente Active Directory. Queste utilità sono usate per introdurre un computer nel dominio e sono le stesse disponibili nei pacchetti samba-common e winbind:

- lwinet: fornisce informazioni riguardo la rete e il dominio,
- lwimsg: consente di interagire con il demone likewise-winbindd.
- lwiinfo: visualizza informazioni riguardo diverse parti del dominio.

Per maggiori informazioni, consultare le pagine man delle utilità.

7.4. Risoluzione problemi

- Se il client incontra dei problemi nell'unirsi al dominio, controllare che il DNS Microsoft sia elencato per primo nel file `/etc/resolv.conf`. Per esempio:

  ```
  nameserver 192.168.0.1
  ```

- Per ricevere maggiori informazioni mentre si entra in un dominio, usare l'opzione *--loglevel verbose* o *--advanced* dell'utilità domainjoin-cli:

  ```
  sudo domainjoin-cli --loglevel verbose join example.it Administrator
  ```

- Se un utente Active Directory incontra problemi nell'accedere, controllare `/var/log/auth.log` per maggiori dettagli.

- Quando viene unita una workstation Ubuntu a un dominio, è necessario modificare `/etc/nsswitch.conf` se il dominio Active Directory utilizza la sintassi *.local*. Affinché sia possibile unirsi al dominio, la voce *"mdns4"* deve essere rimosso dall'opzione *hosts*. Per esempio:

  ```
  hosts: files mdns4_minimal [NOTFOUND=return] dns mdns4
  ```

  Modificare in:

  ```
  hosts: files dns [NOTFOUND=return]
  ```

  Quindi riavviare i servizi di rete:

  ```
  sudo /etc/init.d/networking restart
  ```

  Ora dovrebbe essere possibile unirsi al dominio Active Directory.

7.5. DNS Microsoft

Quelle che seguono sono istruzioni per installare DNS su un controller di dominio Active Directory in esecuzione su Windows Server 2003, ma le istruzioni dovrebbero essere simili ad altre versioni:

- Click Start → Administrative Tools → Manage Your Server. This will open the Server Role Mangement utility.
  1. Fate clic su Aggiungi o rimuovi un ruolo
  2. Fare clic su Next
  3. Selezionare «DNS Server»
  4. Fare clic su Next
  5. Fate di nuovo clic su Successivo per procedere
  6. Selezionare, nel caso non lo sia, «Create a forward lookup zone».
  7. Fare clic su Next
  8. Assicurarsi che «This server maintains the zone» sia selezionato e fare clic su Next.
  9. Inserire il nome del dominio e fare clic su Next
  10. Fare clic su Next per «Allow only secure dynamic updates»
  11. Inserire l'indirizzo IP per i server DNS a cui inoltrare le richieste o selezionare «No, it should not forward queries» e fare clic su Next.
  12. Fare clic su Finish
  13. Fare clic su Finish

  Il DNS è ora installato e può essere ulteriormente configurato utilizzando la Microsoft Management Console.

- Ora, configurare il server affinché venga usato per le interrogazioni DNS:

1. Fare clic su Start
2. Control Panel
3. Network Connections
4. Fare clic su «Local Area Connection»
5. Fare clic su «Properties»
6. Fare doppio-clic su «Internet Protocol (TCP/IP)»
7. Inserire l'indirizzo IP del server come «Preferred DNS server»
8. Fare clic su Ok
9. Fare clic nuovamente su Ok per salvare le impostazioni

7.6. Riferimenti

Per maggiori informazioni, consultare il sito web di *Likewise*[30].

Per maggiori informazioni sulle opzioni di domainjoin-cli, consultare la pagina di manuale **man domainjoin-cli**.

Consultare anche la *documentazione della comunità*[31].

[30] http://www.likewisesoftware.com/

[31] https://help.ubuntu.com/community/LikewiseOpen

Capitolo 18. Backup

È possibile eseguire dei backup delle installazioni di Ubuntu in molti modi diversi. La fase più importante è comunque quella della *pianificazione*: di cosa eseguire il backup, dove salvarlo e come ripristinarlo.

Questa sezione descrive diversi metodi per compiere queste attività.

1. Script shell

Uno dei metodi più semplici per effettuare il backup del sistema è usare uno *script shell*. Per esempio, è possibile usare uno script per scegliere le directory da archiviare e usare queste directory come argomento per tar per creare un archivio, che può essere quindi spostato o copiato in un'altra posizione. L'archivio può anche essere creato su un disco remoto, come una condivisione *NFS*.

L'utilità tar crea un archivio di file a partire da molti file o directory. Con tar è possibile anche elaborare i file con utilità di compressione riducendo così la dimensione dell'archivio.

1.1. Semplice script shell

Il seguente script utilizza tar per creare un archivio su un file system remoto NFS. Il nome dell'archivio è determinato utilizzando delle utilità a riga di comando aggiuntive.

```
#!/bin/sh
####################################
#
# Backup to NFS mount script.
#
####################################

# What to backup.
backup_files="/home /var/spool/mail /etc /root /boot /opt"

# Where to backup to.
dest="/mnt/backup"

# Create archive filename.
day=$(date +%A)
hostname=$(hostname -s)
archive_file="$hostname-$day.tgz"

# Print start status message.
echo "Backing up $backup_files to $dest/$archive_file"
date
echo

# Backup the files using tar.
tar czf $dest/$archive_file $backup_files

# Print end status message.
echo
echo "Backup finished"
date

# Long listing of files in $dest to check file sizes.
ls -lh $dest
```

- *$backup_files*: una variabile con le directory di cui si vuole fare una copia. L'elenco va modificato come desiderato.
- *$day*: una variabile contenente il giorno della settimana (lunedì, martedì, mercoledì, ecc...) viene usata per creare un archivio per ogni giorno della settimana, consentendo di avere una cronologia di archivi di sette giorni. Esistono altri metodi per fare questo, come l'uso dell'utilità date.
- *$hostname*: variabile contenente il nome host *breve* del sistema. Usare il nome dell'host nel nome dell'archivio, consente di avere backup giornalieri di diversi sistemi in una sola directory.
- *$archive_file*: il nome completo dell'archivio.
- *$dest*: destinazione dell'archivio. La directory deve essere creata e in questo caso anche *montata* prima di eseguire lo script. Per dettagli sull'uso di *NFS*, consultare *Sezione 2, «NFS (Network File System)» [177]*.
- *messaggi:* messaggi opzionali stampati sulla console usando echo.
- *tar czf $dest/$archive_file $backup_files*: il comando tar usato per creare l'archivio.
  - *c*: crea l'archivio.
  - *z*: passa l'archivio attraverso l'utilità di compressione gzip.
  - *f*: usa un file archivio, altrimenti il comando tar invia l'output sullo STDOUT.
- *ls -lh $dest*: istruzione opzionale che stampa un elenco lungo (-*l*) in un formato leggibile (-*h*) della directory di destinazione. È utile per controllare la dimensione dell'archivo. Questa verifica non dovrebbe sostituire la verifica dell'archivio.

Questo è un semplice esempio di un backup eseguito con uno script shell ed è possibile aggiungere molte opzioni. Per maggiori informazioni sugli script shell, consultare *Sezione 1.4, «Riferimenti» [251]*.

1.2. Eseguire lo script

1.2.1. Esecuzione da terminale

Il metodo più facile per eseguire lo script di backup è quello di copiare il contenuto dello script in un file, `backup.sh` per esempio, ed eseguirlo in un terminale:

```
sudo bash backup.sh
```

È un ottimo modo per provare lo script e assicurarsi che funzioni correttamente.

1.2.2. Esecuzione con cron

L'utilità cron può essere usata per automatizzare l'esecuzione dello script. Il demone cron consente l'esecuzione di script o comandi a un determinato orario e data.

L'applicazione cron è configurata attraverso delle voci in un file `crontab` file. I file `crontab` sono separati in campi:

```
# m h dom mon dow   comando
```

- *m*: minuto di esecuzione del comando, tra 0 e 59.
- *h*: ora di esecuzione del comando, tra 0 e 23.
- *dom*: giorno del mese di esecuzione del comando.
- *mon*: il mese in cui viene eseguito il comando tra 1 e 12.
- *dow*: il giorno della settimana in cui viene eseguito il comando tra 0 e 7. La domenica può essere specificata usando sia 0 che 7.
- *comando*: il comando da eseguire.

Per aggiungere o modificare voci in un file `crontab`, dovrebbe essere usato il comando crontab -e, i contenuti di un file `crontab` possono essere visualizzati usando il comando crontab -l.

Per eseguire lo script backup.sh usando cron, in un terminale digitare quanto segue:

```
sudo crontab -e
```

Usare sudo con il comando crontab -e, modifica il crontab dell'utente *root*. Questo è necessario nel caso in cui si stiano eseguendo copie di backup di file accessibili solo dall'utente root.

Aggiungere quanto segue al file `crontab`:

```
# m h dom mon dow   command
0 0 * * * bash /usr/local/bin/backup.sh
```

Lo script backup.sh verrà eseguito ogni giorno alle 12.00 AM.

Lo script backup.sh deve essere copiato nella directory `/usr/local/bin/` affinché possa essere eseguito correttamente. Lo script può essere posizionato ovunque nel file system, basta correggere il percorso di conseguenza.

Per maggiori informazioni riguardo crontab, consultare *Sezione 1.4, «Riferimenti» [251]*.

1.3. Ripristinare l'archivio

Una volta creato un archivio, è importante verificarlo, elencandone i contenuti oppure, ed è la scelta migliore, *ripristinare* un file dall'archivio.

- Per visualizzare il contenuto di un archivio, da un terminale, digitare:

  ```
  tar -tzvf /mnt/backup/host-lunedì.tgz
  ```
- Per ripristinare un file dall'archivio in una directory diversa, digitare:

```
tar -xzvf /mnt/backup/host-lunedì.tgz -C /tmp etc/hosts
```

L'opzione *-C* di tar redirige i file estratti nella directory specificata. L'esempio precedente estrarrà il file `/etc/hosts` in `/tmp/etc/hosts`. La struttura della directory viene quindi ricreata da tar.

Notare anche che il simbolo "/" iniziale del percorso in cui ripristinare è stato tralasciato.

- Per ripristinare tutti i file presenti nell'archivio, digitare:

```
cd /
sudo tar -xzvf /mnt/backup/host-lunedì.tgz
```

In questo modo verranno sovrascritti i file attualmente presenti nel file system.

1.4. Riferimenti

- Per maggiori informazioni riguardo lo script da shell, consultare la *Advanced Bash-Scripting Guide*[1]
- Il libro *Teach Yourself Shell Programming in 24 Hours*[2] è disponibile in linea ed è un'ottima risorsa per lo script da shell.
- La pagina della *della documentazione in linea su cron*[3] contiene ulteriori dettagli sulle opzioni avanzate di cron.
- Per maggiori informazioni sulle opzioni del comando tar, consultare il *manuale in linea di tar*[4].
- La pagina inglese di Wikipedia *Backup Rotation Scheme*[5] contiene informazioni sugli schemi di backup.
- Questo script utilizza tar per creare l'archivio, ma esistono diverse altre utilità a riga di comando che possono essere usate, per esempio:
  - *cpio*[6]: usata per copiare file da e verso degli archivi.
  - *dd*[7]: parte del pacchetto coreutils. Un'utilità a basso livello in grado di copiare dati da un formato a un altro.
  - *rsnapshot*[8]: un'utilità per creare snapshot (istantanee) del file system usata per creare copie di un intero file system.

2. Rotazione degli archivi

Lo script in *Sezione 1, «Script shell» [248]* consente solamente sette archivi differenti. Per un server i cui dati non cambiano molto spesso questo può essere sufficiente, ma se il server dispone di grossi quantitativi di dati è necessario avere uno schema di rotazione degli archivi più efficiente.

2.1. Rotazione degli archivi NFS

In questa sezione lo script verrà modificato per implementare uno schema di rotazione del tipo progenitore-genitore-figlio (mensile-settimanale-giornaliero).

- La rotazione eseguirà un backup *giornaliero* dalla domenica al venerdì.
- Il sabato, viene eseguito un backup *settimanale* consentendo di avere così quattro backup settimanali al mese.
- Il backup *mensile* è eseguito il primo giorno del mese, ruotando due backup mensili se il mese è pari o dispari.

Questo è il nuovo script:

```
#!/bin/bash
####################################
#
# Backup to NFS mount script with
# grandfather-father-son rotation.
#
####################################

# What to backup.
backup_files="/home /var/spool/mail /etc /root /boot /opt"

# Where to backup to.
dest="/mnt/backup"

# Setup variables for the archive filename.
day=$(date +%A)
hostname=$(hostname -s)

# Find which week of the month 1-4 it is.
day_num=$(date +%d)
if (( $day_num <= 7 )); then
        week_file="$hostname-week1.tgz"
elif (( $day_num > 7 && $day_num <= 14 )); then
        week_file="$hostname-week2.tgz"
elif (( $day_num > 14 && $day_num <= 21 )); then
        week_file="$hostname-week3.tgz"
elif (( $day_num > 21 && $day_num < 32 )); then
        week_file="$hostname-week4.tgz"
fi
```

```
# Find if the Month is odd or even.
month_num=$(date +%m)
month=$(expr $month_num % 2)
if [ $month -eq 0 ]; then
        month_file="$hostname-month2.tgz"
else
        month_file="$hostname-month1.tgz"
fi

# Create archive filename.
if [ $day_num == 1 ]; then
 archive_file=$month_file
elif [ $day != "Saturday" ]; then
        archive_file="$hostname-$day.tgz"
else
 archive_file=$week_file
fi

# Print start status message.
echo "Backing up $backup_files to $dest/$archive_file"
date
echo

# Backup the files using tar.
tar czf $dest/$archive_file $backup_files

# Print end status message.
echo
echo "Backup finished"
date

# Long listing of files in $dest to check file sizes.
ls -lh $dest/
```

Lo script può essere eseguito attraverso gli stessi metodi descritti in *Sezione 1.2, «Eseguire lo script» [249]*.

È utile, nel caso si verifichino disastri, che il dispositivo di backup sia in una località diversa. Nello script di esempio l'unità di backup è un server che fornisce una condivisione NFS: spostare questo server in un'altra località potrebbe non essere fattibile. In base alla velocità di connessione, potrebbe essere utile considerare di copiare il backup attraverso un collegamento WAn su un server remoto.

Un'altra opzione consiste nel copiare l'archivio su di un disco esterno che può essere spostato. Poiché il prezzo dei dischi portatili esterni è sempre in diminuzione, l'utilizzo di due dischi per ogni livello dell'archivio può risultare altamente vantaggioso: in questo modo è possibile avere un disco esterno collegato al server di backup e un altro in un altra località.

2.2. Dispositivi a nastro

Invece di usare una condivisione NFS, è possibile utilizzare un dispositivo a nastro collegato al server. Un dispositivo di questo tipo semplifica la rotazione degli archivi e il portare il dispositivo stesso in un'altra locazione.

Quando viene usato un dispositivo a nastro, la parte relativa al nome del file non è necessaria in quanto la data è inviata direttamente al dispositivo, ma sono necessari alcuni comandi per manipolare il nastro. In particolare mt, un'utilità di controllo nastri magnetici parte del pacchetto cpio.

Questo è lo script modificato per l'uso di un dispositivo a nastro:

```
#!/bin/bash
####################################
#
# Backup to tape drive script.
#
####################################

# What to backup.
backup_files="/home /var/spool/mail /etc /root /boot /opt"

# Where to backup to.
dest="/dev/st0"

# Print start status message.
echo "Backing up $backup_files to $dest"
date
echo

# Make sure the tape is rewound.
mt -f $dest rewind

# Backup the files using tar.
tar czf $dest $backup_files

# Rewind and eject the tape.
mt -f $dest rewoffl

# Print end status message.
echo
echo "Backup finished"
date
```

Il nome del device predefinito per un dispositivo a nastro SCSI è `/dev/st0`, utilizzare il percorso al device appropriato per il proprio sistema.

Ripristinare i dati da un dispositivo a nastro funziona allo stesso modo di ripristinare da un file. Riavvolgere il nastro e usare il percorso del dispositivo al posto del percorso al file. Per esempio, per ripristinare il file `/etc/hosts` in `/tmp/etc/hosts`:

```
mt -f /dev/st0 rewind
tar -xzf /dev/st0 -C /tmp etc/hosts
```

3. Bacula

Bacula è un programma per eseguire backup, ripristinare e verificare i dati attraverso la rete. Esistono client Bacula per Linux, Windows, e Mac OS X rendendolo una soluzione multi-piattaforma.

3.1. Panoramica

Bacula è composto da diversi componenti e servizi usati per la gestione dei file di cui eseguire il backup e dove eseguirlo:

- Bacula Director: un servizio che controlla tutte le operazioni di backup, ripristino, verifica e di archiviazione.
- Bacula Console: un'applicazione che consente di comunicare con "Director". Sono disponibili tre versioni:
  - Versione testuale per la riga di comando.
  - Versione grafica per GNOME basata su GTK+.
  - Interfaccia wxWidgets.
- Bacula File: conosciuta anche come Bacula Client. Questa applicazione è installata nei computer di cui deve essere fatto il backup ed è responsabile dei dati richiesti dal Director.
- Bacula Storage: il programma che esegue l'archiviazione e il ripristino sul dispositivo fisico.
- Bacula Catalog: responsabile per mantenere l'indice dei file e il database di tutti i file, consentendo una facile localizzazione e ripristino. "Catalog" supporta tre diversi database: MySQL, PostgreSQL e SQLite.
- Bacula Monitor: consente di monitorare i demoni "Director", "File" e "Storage". Attualmente "Monitor" è disponibile solo come applicazione GTK+.

Questi servizi e applicazioni possono essere eseguiti su molteplici server e client oppure possono essere installati su un solo computer se deve essere eseguito il backup di un singolo disco o volume.

3.2. Installazione

Ci sono molteplici pacchetti che contengono i diversi componenti di Bacula. Per installare Bacula, in un terminale, digitare:

```
sudo apt-get install bacula
```

In modo predefinito, installando il pacchetto bacula viene usato un database MySQL per "Catalog". Se si vuole usare SQLite oppure PostgreSQL, installare bacula-director-sqlite3 o bacula-director-pgsql rispettivamente.

Durante il processo di installazione viene chiesto di fornire delle credenziali per l'*amministratore* del database e per il *proprietario* del database *bacula*. L'amministratore del database deve avere i

diritti apporpriati per poter creare un database. Per maggiori informazioni, consultare la *Sezione 1, «MySQL» [160]*.

3.3. Configurazione

I file di configurazione di Bacula sono formattati in base alle *risorse* composte da *direttive* marcate da parantesi «{}». Ogni compoenente di Bacula dispone di un file nella directory `/etc/bacula`.

I diversi componenti di Bacula devono autorizzarsi tra di loro. Questo è fatto usando la direttiva *password*. Per esempio, la risorsa password di *Storage* nel file `/etc/bacula/bacula-dir.conf` deve corrispondere alla risorsa password di *Director* nel file `/etc/bacula/bacula-sd.conf`.

In modo predefinito, il lavoro di backup chiamato *Client1* è confoigurato per archiviare il "Catalog" di Bacula. Se si intende usare il server per eseguire il backup di più di un client, è necessario modificare il nome del lavoro con qualche cosa di più descrittivo. Per fare questo, modificare il file `/etc/bacula/bacula-dir.conf`:

```
#
# Define the main nightly save backup job
#   By default, this job will back up to disk in
Job {
  Name = "BackupServer"
  JobDefs = "DefaultJob"
  Write Bootstrap = "/var/lib/bacula/Client1.bsr"
}
```

L'esempio precedente modifica il nome del lavoro in *BackupServer*, in corrispondenza del nome host del computer. Sostituire «BackupServer» con il nome host appropriato o un altro nome descrittivo.

Console può essere usato per interrogare *Director* riguardo i lavori, ma per poter usare "Console" con un utente *non-root*, l'utente deve essere nel gruppo *bacula*. Per aggiungere un utente al gruppo "bacula", in un terminale, digitare:

```
sudo adduser NOME_UTENTE bacula
```

Sostituire *NOME_UTENTE* con il vero nome utente. Inoltre, se si sta aggiungendo l'utente corrente al gruppo, è necessario terminare la sessione e rientrarvi affinché le modifiche abbiano effetto.

3.4. Backup locale

Questa sezione descrive come eseguire un backup di specifiche directory di un singolo host in un dispositivo a nastro locale.

- Per prima cosa, *Storage* deve essere configurato. Modificare `/etc/bacula/bacula-sd.conf`:

```
Device {
  Name = "Tape Drive"
  Device Type = tape
  Media Type = DDS-4
  Archive Device = /dev/st0
  Hardware end of medium = No;
  AutomaticMount = yes;               # when device opened, read it
  AlwaysOpen = Yes;
  RemovableMedia = yes;
  RandomAccess = no;
  Alert Command = "sh -c 'tapeinfo -f %c | grep TapeAlert'"
}
```

L'esempio è per un dispositivo a nastro *DDS-4*. Modificare "Media Type" e "Archive Device" affinché corrispondano al proprio hardware.

È possibile anche de-commentare uno degli altri file di esempio.

- Una volta modificato il file `/etc/bacula/bacula-sd.conf`, il demone Storage deve essere riavviato:

```
sudo /etc/init.d/bacula-sd restart
```

- Ora aggiungere una risorsa *Storage* in `/etc/bacula/bacula-dir.conf` per usare il nuovo "Device":

```
# Definition of "Tape Drive" storage device
Storage {
  Name = TapeDrive
  # Do not use "localhost" here
  Address = backupserver               # N.B. Use a fully qualified name here
  SDPort = 9103
  Password = "Cv70F6pf1t6pBopT4vQOnigDrR0v3LT3Cgkiyj"
  Device = "Tape Drive"
  Media Type = tape
}
```

La direttiva *Address* deve essere il "Fully Qualified Domain Name" (FQDN) del server. Modificare quindi *backupserver* col nome host attuale.

Inoltre, assicurarsi che la direttiva *Password* corrisponda alla stringa in `/etc/bacula/bacula-sd.conf`.

- Creare un nuovo *FileSet*, per determinare di quali directory eseguire il backup:

```
# LocalhostBacup FileSet.
FileSet {
  Name = "LocalhostFiles"
  Include {
    Options {
      signature = MD5
      compression=GZIP
```

```
    }
    File = /etc
    File = /home
  }
}
```

Questa sezione *FileSet* farà in modo di eseguire il backup della directory `/etc` e `/home`. La direttiva *Options* configura "FileSet" per creare una somma MD5 per ogni file di cui è fatto il backup e per comprimere tali file con GZIP.

- Creare una nuova sezione *Schedule* per il lavoro di backup:

```
# LocalhostBackup Schedule -- Daily.
Schedule {
  Name = "LocalhostDaily"
  Run = Full daily at 00:01
}
```

Il lavoro verrà eseguito ogni giorno alle 00.01. Sono comunque disponibili molte altre opzioni di schedulatura.

- Infine creare il *Job*:

```
# Localhost backup.
Job {
  Name = "LocalhostBackup"
  JobDefs = "DefaultJob"
  Enabled = yes
  Level = Full
  FileSet = "LocalhostFiles"
  Schedule = "LocalhostDaily"
  Storage = TapeDrive
  Write Bootstrap = "/var/lib/bacula/LocalhostBackup.bsr"
}
```

Questo lavoro crearà un backup *Full* (completo) ogni giorni sul dispositivo a nastro.

- Ogni nastro usato deve avere una *Label*. Se il nastro corrente ne è sprovvisto, Bacula invierà un'email. Per aggiungere un'etichetta a un nastro usando Console, in un terminale, digitare:

```
bconsole
```

- Al prompt di "Console" digitare:

```
label
```

- Viene quindi chiesta la risorsa *Storage*:

```
Automatically selected Catalog: MyCatalog
Using Catalog "MyCatalog"
```

```
The defined Storage resources are:
     1: File
     2: TapeDrive
Select Storage resource (1-2):2
```

- Inserire il nome del nuovo *Volume* (volume):

```
Enter new Volume name: Sunday
Defined Pools:
     1: Default
     2: Scratch
```

  Sostituire *Sunday* con l'etichetta desiderata.

- Ora selezionare *Pool*:

```
Select the Pool (1-2): 1
Connecting to Storage daemon TapeDrive at backupserver:9103 ...
Sending label command for Volume "Sunday" Slot 0 ...
```

Bacula è ora configurato per eseguire backup del host locale su un dispositivo a nastro.

3.5. Risorse

- Per maggiori informazioni sulle opzioni di configurazione di *Bacula*, consultare il *manuale di Bacula*[9]
- Il sito *di Bacula*[10] contiene le ultime notizie dello sviluppo di Bacula.
- Consultare anche la *documentazione di bacula online*[11].

Capitolo 19. Virtualizzazione

La virtualizzazione, al giorno d'oggi, viene utilizzata in diversi ambienti e situazioni. Dal punto di vista dello sviluppatore, la virtualizzazione offre un ambiente sicuro dove poter eseguire qualsiasi tipo di sviluppo, senza compromettere l'ambiente di lavoro. Per l'amministratore di sistema, è possibile usare la virtualizzazione per separare facilmente i propri servizi e spostarli in base alle richieste.

La tecnologia di virtualizzazione supportata in modo predefinito in Ubuntu è KVM, una tecnologa che sfrutta le estensioni di virtualizzazione presenti nelle CPU Intel o AMD. Per l'hardware che non è dotato di estensioni di virtualizzazione, Xen e Qemu sono le soluzioni più usate.

1. libvirt

La libreria libvirt è utilizzata per interfacciarsi con differenti tecnologie di virtualizzazione. Prima di iniziare a utilizzare libvirt è utile accertarsi che il proprio hardware supporti le estensioni di virtualizzazione necessarie per KVM. In un terminale, digitare quanto segue:

```
kvm-ok
```

Verrà stampato un messaggio che indica se la CPU *supporta* o *non supporta* la virtualizzazione hardware.

Nella maggior parte dei processori che supportano la virtualizzazione è necessario attivarla attraverso un'opzione nel BIOS.

1.1. Rete virtuale

Esistono diversi modi per consentire accesso alla rete esterna a una macchina virtuale. La configurazione di rete predefinita è *usermode*, che utilizza il protocollo SLIRP e il traffico è passato attraverso l'interfaccia dell'host verso la rete esterna.

Affinché gli host esterni possano accedere i servizi su una macchina virtuale, è necessario configurare un *bridge*. Questo consente alle interfacce virtuali di connettersi alla rete esterna attraverso l'interfaccia fisica, facendole apparire come normali host al resto della rete. Per informazioni su come impostare un bridge, consultare *Sezione 1.4, «Bridging» [37]*.

1.2. Installazione

Per installare i pacchetti necessari, da un terminale digitare:

```
sudo apt-get install kvm libvirt-bin
```

Dopo aver installato libvirt-bin, l'utente usato per la gestione delle macchine virtuali deve essere aggiunto al gruppo *libvirtd*. In questo modo, all'utente è garantito accesso alle configurazioni avanzate di rete.

In un terminale digitare:

```
sudo adduser $USER libvirtd
```

Se l'utente scelto è quello corrente, è necessario terminare la sessione e ri-accedervi affinché le modifiche abbiano effetto.

È ora possibile installare un sistema operativo *ospite*. La procedura di installazione di una macchina virtuale è la stessa di un sistema operativo normale ed è quindi necessario automatizzare la procedura oppure avere una tastiera e uno schermo collegati al computer.

Nel caso delle macchine virtuali, un'interfaccia grafica è analoga all'uso di una tastiera e di un mouse. Invece di installare un'interfaccia grafica, è possibile usare virt-viewer per connettersi alla console di una macchina virtuale via VNC. Per maggiori informazioni, consultare la *Sezione 1.6, «Visualizzatore di macchine virtuali» [265]*.

There are several ways to automate the Ubuntu installation process, for example using preseeds, kickstart, etc. Refer to the *Ubuntu Installation Guide*[1] for details.

Un altro metodo per installare una macchina virtuale Ubuntu consiste nell'usare l'applicazione ubuntu-vm-builder. ubuntu-vm-builder consente di impostare partizioni avanzate, eseguire script personalizzati post-installazione, ecc... Per maggiori informazioni, consultare *Sezione 2, «JeOS e vmbuilder» [267]*

1.3. virt-install

virt-installfa parte del pacchetto python-virtinst. Per installarlo, in un terminale, digitare:

```
sudo apt-get install python-virtinst
```

Durante l'uso di virt-install sono disponibili molte azioni, per esempio:

```
sudo virt-install -n web_devel -r 256 -f web_devel.img \ -s 4 -c jeos.iso --accelerate \ --connect=
```

- *-n web_devel*: il nome della nuova macchine virtuale usato in questo esempio sarà *web_devel*.
- *-r 256*: specifica la quantità di memoria che la macchina virtuale userà.
- *-f web_devel.img*: indica il percorso al disco virtuale che può essere un file, una partizione o un volume logico. In questo esempio è un file chiamato `web_devel.img`.
- *-s 4*: la dimensione del disco virtuale.
- *-c jeos.iso*: il file usato come CD-ROM virtuale. Il file può essere un file ISO o il percorso al device del CD-ROM nell'host.
- *--accelerate*: abilita letecnologie di accelerazione nel kernel.
- *--vnc*: esporta la console virtuale usando VNC.
- *--noautoconsole*: non si collegherà automaticamente alla console della macchina virtuale.
- *-v*: crea un ospite completamente virtualizzato.

Una volta lanciata virt-install è possibile collegarsi alla console della macchina virtuale utilizzando, localmente, un'interfaccia grafica oppure l'utilità virt-viewer.

1.4. virt-clone

L'applicazione virt-clone può essere usata per copiare una macchina virtuale in un'altra, per esempio:

[1] https://help.ubuntu.com/10.10/installation-guide/

```
sudo virt-clone -o web_devel -n database_devel -f /path/to/database_devel.img --connect=qemu:///sys
```

- *-o*: macchina virtuale originale.
- *-n*: nome della nuova macchina virtuale.
- *-f*: percorso al file, volume logico o partizione da usare per la nuova macchina virtuale.
- *--connect*: specifica a quale hypervisor collegarsi.

Usare anche le opzioni *-d* o *--debug* per risolvere i problemi che potrebbero verificarsi con virt-clone.

Sostituire *web_devel* e *database_devel* con i nomi delle macchine virtuali appropriati.

1.5. Gestire la macchina virtuale

1.5.1. virsh

Sono disponibili diverse utilità per la gestione delle macchine virtuali e di libvirt. L'utilità virsh può essere utilizzata dalla riga di comando. Alcuni esempi:

- Per elencare le macchine virtuali in esecuzione:

  ```
  virsh -c qemu:///system list
  ```
- Per avviare una macchina virtuale:

  ```
  virsh -c qemu:///system start web_devel
  ```
- Similmente, per lanciare una macchina virtuale durante l'avvio del computer:

  ```
  virsh -c qemu:///system autostart web_devel
  ```
- Riavviare una macchina virtuale con:

  ```
  virsh -c qemu:///system reboot web_devel
  ```
- Lo *stato* di una macchina virtuale può essere salvato in un file per poterlo ripristinare successivamente. Il seguente comando salva lo stato della macchina virtuale in un file nominato in base alla data.

  ```
  virsh -c qemu:///system save web_devel web_devel-022708.state
  ```

  Una volta salvata, la macchina virtuale non sarà più in esecuzione.
- Per ripristinare una macchina virtuale:

  ```
  virsh -c qemu:///system restore web_devel-022708.state
  ```
- Per arrestare una macchina virtuale:

```
virsh -c qemu:///system shutdown web_devel
```

- Per montare un CD-ROM in una macchina virtuale, digitare:

```
virsh -c qemu:///system attach-disk web_devel /dev/cdrom /media/cdrom
```

Nell'esempio precedente, sostituire *web_devel* con il nome della macchina virtuale appropriata e `web_devel-022708.state` con un nome file descrittivo.

1.5.2. Gestore macchina virtuale

Il pacchetto virt-manager contiene un'utilità grafica per gestire le macchie virtuali locali e remote. Per installare virt-manager digitare:

```
sudo apt-get install virt-manager
```

Dato che virt-manager richiede un'interfaccia grafica (GUI), è raccomandato installarlo su una workstation o una postazione di prova invece che un server di produzione. Per connettersi al servizio libvirt locale:

```
virt-manager -c qemu:///system
```

È possibile collegarsi al servizio libvirt in esecuzione su un altro host digitando, in un terminale:

```
virt-manager -c qemu+ssh://virtnode1.mydomain.com/system
```

L'esempio precedente assume che la connessione SSH tra il sistema di gestione e virtnode1.mydomain.com sia già configurata e utilizzi le chiavi SSH per l'autenticazione. Le *chiavi* SSH sono necessarie perché libvirt invia il prompt password a un altro processo. Per maggiori informazioni sulla configurazione di SSH, consultare la *Sezione 1, «Server OpenSSH» [49]*

1.6. Visualizzatore di macchine virtuali

L'applicazione virt-viewer consente di collegarsi alla console di una macchina virtuale. virt-viewer non richiede un'interfaccia grafica per interagire con la macchina virtuale.

Per installare virt-viewer, da un terminale digitare:

```
sudo apt-get install virt-viewer
```

Una volta installata e in esecuzione, è possibile connettersi alla console della macchina virtuale digitando:

```
virt-viewer -c qemu:///system web_devel
```

Analogamente a virt-manager, virt-viewer può collegarsi a un host remoto utilizzando *SSH* con chiave di autenticazione:

```
virt-viewer -c qemu+ssh://virtnode1.miodominio.it/system web_devel
```

Assicurarsi di sostituire *web_devel* con il nome corretto della macchina virtuale.

Se configurato per usare un'interfaccia di rete *bridged*, è anche possibile impostare accesso SSH alla macchina virtuale. Per maggiori informazioni, consultare *Sezione 1, «Server OpenSSH» [49]* e *Sezione 1.4, «Bridging» [37]*.

1.7. Risorse

- Per maggiori informazioni, consultare il sito web di *KVM*[2].
- Per maggiori informazioni su libvirt, consultare *il sito web di libvirt*[3]
- Il sito di *Virtual Machine Manager*[4] dispone di ulteriori informazioni riguardo lo sviluppo di virt-manager.
- È anche possibile passare nel canale IRC *#ubuntu-virt* su *freenode*[5] per discutere delle tecnologie di virtualizzazione in Ubuntu.
- Un'altra ottima risorsa è la *documentazione online*[6] riguardo KVM.

2. JeOS e vmbuilder

2.1. Introduzione

2.1.1. Cos'è JeOS

Ubuntu *JeOS* (pronunciato come la parola "juice") è una variante di della versione server di Ubuntu, configurata appositamente per le applicazioni virtuali. Non è disponibile sotto forma di file ISO per CD-ROM, ma solo come opzione:

- durante l'installazione della versione server (premere *F4* alla prima schermata per scegliere l'opzione "Installa un sistema minimale" che equivale a selezionare JeOS).
- oppure può essere generato usando "vmbuilder" come descritto di seguito.

JeOS è un'installazione di Ubuntu Server Edition con un kernel appositamente configurato che contiene gli elementi basilari necessari all'esecuzione di un ambiente virtualizzato.

Ubuntu JeOS è stato progettato per sfruttare tutte quelle tecnologie chiave, relative alle prestazioni, presenti negli ultimi prodotti di virtualizzazione di VMware. La combinazione di una ridotta dimensione e prestazioni ottimizzate, assicurano che Ubuntu JeOS Edition sia in grado di offrire un uso efficiente delle risorse server in grandi produzioni virtuali.

Senza l'utilizzo di driver non necessari e ricorrendo solo ai pacchetti richiesti, gli ISV possono configurare il proprio SO di supporto proprio come desiderano. Inoltre, viene assicurato che gli aggiornamenti, di sicurezza o per miglioramenti, saranno limitati al minimo richiesto dallo specifico ambiente. Gli utenti che sviluppano soluzioni virtuali basate su JeOS, dovranno gestire meno aggiornamenti, e quindi una minor manutenzione, di quanto avrebbero dovuto fare con un'installazione server completa.

2.1.2. Cos'è vmbuilder

Utilizzando vmbuilder non è necessario scaricare un'immagine di JeOS: verranno scaricati i pacchetti necessari per creare una macchina virtuale adatta alle proprie esigenze. vmbuilder è uno script che automatizza la creazione di una macchina virtuale Linux. Gli hypervisor supportati attualmente sono KVM e Xen.

È possibile passare opzioni a riga di comando per aggiungere dei pacchetti, per rimuoverne, per scegliere la versione di Ubuntu, quale mirror, ecc... Su piattaforme hardware recenti dotate di molta memoria RAM, con tmpdir in `/dev/shm` o usando un tmpfs e un mirror locale, è possibile avere una macchina virtuale in meno di un minuto.

Introdotto come semplice script shell in Ubuntu 8.04 LTS, ubuntu-vm-builder era un semplice progetto per aiutare gli sviluppatori nel provare il codice scritto in una virtual machine senza dover ricominciare sempre da capo. Lo script è stato in seguito migliorato e Soren Hansen (l'autore dello script e lo specialista di virtualizzazione in Ubuntu virtualization, non il giocatore di golf) lo ha riscritto da capo per Intrepid in python con i seguenti obiettivi:

- Svilupparlo affinché possa essere usato anche da altre distribuzioni.
- Usare un meccanismo di plugin per tutte le interazioni di virtualizzazione per facilitare l'aggiunta di altri ambienti di virtualizzazione o una logica più complessa.
- Fornire un'interfaccia web facile da usare come opzione alla riga di comando.

I principi generali e i comandi restano sempre gli stessi.

2.2. Configurazione iniziale

Si presuppone che siano già stati installati e configurati libvirt e KVM sul computer che si intende usare. Per maggiori informazioni, consultare:

- *Sezione 1, «libvirt» [262]*
- La pagina relativa a *KVM*[7] nella documentazione (in inglese).

Si dà per assodato che si sappia utilizzare un editor di testo come nano oppure vi. In caso contrario, è possibile avere una panoramica dei vari editor di testo consultando *la documentazione di Ubuntu*[8]. Questa guida è stata scritta basandosi su KVM, ma il principio dovrebbe essere lo stesso anche per altre tecnologie di virtualizzazione.

2.2.1. Installare vmbuilder

Il nome del pacchetto da installare è python-vm-builder. In un terminale digitare:

```
sudo apt-get install python-vm-builder
```

Se si sta eseguendo la versione 8.04 è sempre possibile eseguire queste azioni usando la versione del pacchetto chiamata ubuntu-vm-builder; ci sono solo alcune modifiche nella sintassi da usare con il programma.

2.3. Definire una macchina virtuale

Definire una macchina virtuale con vmbuilder è molto facile, ma è necessario prendere in considerazione alcuni aspetti:

- Se si pianifica di fornire applicativi virtuali, non assumere che l'utente finale sappia come estendere la dimensione del disco secondo le proprie esigenze. Prendere quindi in considerazione l'utilizzo di dischi virtuali di grandi dimensioni per consentire agli applicativi di crescere o spiegare nella documentazione come allocare maggiore spazio. Potrebbe essere una buona idea salvare i dati in un sistema di archiviazione esterno.
- Dato che la memoria RAM è più facile da allocare in una MV, la dimensione della RAM dovrebbe essere impostata a un valore minimo sicuro per la propria applicazione.

[8] http://wiki.ubuntu-it.org/Ufficio/EditorDiTesto#powereditor

Il comando vmbuilder dispone di due parametri principali: la *tecnologia di virtualizzazione (hypervisor)* e la *distribuzione* finale. Sono disponibili molti altri parametri e tutti possono essere visualizzati con il seguente comando:

```
vmbuilder kvm ubuntu --help
```

2.3.1. Parametri base

As this example is based on KVM and Ubuntu 10.10 (Maverick Meerkat), and we are likely to rebuild the same virtual machine multiple time, we'll invoke vmbuilder with the following first parameters:

```
sudo vmbuilder kvm ubuntu --suite maverick --flavour virtual --arch i386 -o --libvirt qemu:///syste
```

Il parametro --*suite* definisce il rilascio di Ubuntu, --*flavour* specifica di usare il kernel virtuale (quello usato per generare un'immagine JeOS), --*arch* indica di usare un computer a 32 bit, -*o* indica a vmbuilder di sovrascrivere la versione precedente della macchina virtuale e --*libvirt* aggiunge la macchina virtuale risultante tra quelle disponibili nell'ambiente di virtualizzazione.

Note:

- Data la natura delle operazioni eseguite da vmbuilder, sono necessari i privilegi di root.
- Se la macchina virtuale necessita di usare più di 3GB di RAM, è utile generare una macchina a 64 bit (--arch amd64).
- Fino a Ubuntu 8.10, il kernel virtuale era generato solo per architetture a 32 bit, per definire quindi una macchina amd64 su Hardy, usare --*flavour server*.

2.3.2. Parametri di installazione di JeOS

2.3.2.1. Rete con JeOS

2.3.2.1.1. Assegnare un indirizzo IP fisso

Come applicazione che verrà messa in produzione all'interno di reti diverse, è molto difficile conoscere la struttura attuale della rete. Per semplificare la configurazione è utile procedere come solitamente procedono i venditori di hardware di rete, assegnando un indirizzo IP fisso all'interno di una classe di rete che verrà descritta all'interno della propria documentazione. Un indirizzo nell'intervallo 192.168.0.0/255 è una buona scelta.

Per ottenere questo vengono usati i seguenti parametri:

- --*ip INDIRIZZO*: indirizzo IP (il valore predefinito è dhcp se non viene specificato nulla)
- --*hostname NAME*: Set NAME as the hostname of the guest.
- --*mask VALORE*: maschera di rete (valore predefinito: 255.255.255.0)
- --*net VALORE*: indirizzo IP net (valore predefinito: X.X.X.0)
- --*bcast VALORE*: broadcast (valore predefinito: X.X.X.255)
- --*gw INDIRIZZO*: indirizzo del gateway (valore predefinito: X.X.X.1)

- --*dns INDIRIZZO*: indirizzo server dei nomi (valore predefinito: X.X.X.1)

Si dà per scontato che i valori predefiniti siano sufficiente. Il comando diventa:

```
sudo vmbuilder kvm ubuntu --suite maverick --flavour virtual --arch i386 -o --libvirt qemu:///syste
```

2.3.2.1.2. Bridging

Because our appliance will be likely to need to be accessed by remote hosts, we need to configure libvirt so that the appliance uses bridge networking. To do this add the --*bridge* option to the command:

```
sudo vmbuilder kvm ubuntu --suite maverick --flavour virtual --arch i386 -o --libvirt qemu:///syste
```

You will need to have previously setup a bridge interface, see *Sezione 1.4, «Bridging» [37]* for more information. Also, if the interface name is different change *br0* to the actual bridge interface.

2.3.2.2. Partizionamento

Il partizionamento dell'applicativo virtuale deve prendere in considerazione cosa si intende fare. Dato che molti applicativi non avranno un sistema di archiviazione separato per i dati, usare una partizione `/var` separata è una buona idea.

Per ottenere tutto questo, vmbuilder dispone dell'opzione --*part*:

```
--part PATH
  Allows you to specify a partition table in a partition file, located at PATH. Each line of the pa
  (root first):
      mountpoint size
  where  size  is  in megabytes. You can have up to 4 virtual disks, a new disk starts on a
  line with '---'.  ie :
      root 1000
      /opt 1000
      swap 256
      ---
      /var 2000
      /log 1500
```

In questo caso, creare un file di testo `vmbuilder.partition` contenente quanto segue:

```
root 8000
swap 4000
---
/var 20000
```

Notare che vengono usate immagini disco virtuali, le dimensioni inserite sono le dimensioni massime dei volumi.

Il comando diventa quindi:

```
sudo vmbuilder kvm ubuntu --suite maverick --flavour virtual --arch i386 \
        -o --libvirt qemu:///system --ip 192.168.0.100 --hostname myvm --part vmbuilder.partition
```

L'uso di "\" all'interno di un comando consente di scrivere comandi su più righe.

2.3.2.3. Utente e password

È necessario anche impostare un utente e una password predefiniti e generici da poter includere nella documentazione. Successivamente verrà presentato uno script che viene eseguito al primo accesso di un utente che tra le molte cose chiederà di modificare la password. In questo esempio viene usato come nome utente *user* e *default* come password.

Per fare questo vengono usati i seguenti parametri:

- *--user NOME_UTENTE*: imposta il nome utente da aggiungere. Valore predefinito: ubuntu.
- *--name NOME_COMPLETO*: imposta il nome completo dell'utente da aggiungere. Valore predefinito: Ubuntu.
- *--pass PASSWORD*: imposta la password dell'utente: Valore predefinito: ubuntu.

Il comando ora è il seguente:

```
sudo vmbuilder kvm ubuntu --suite maverick --flavour virtual --arch i386 \
        -o --libvirt qemu:///system --ip 192.168.0.100 --hostname myvm --part vmbuilder.partition
        --user user --name user --pass default
```

2.3.3. Installare i pacchetti richiesti

In questo esempio verrà installato un pacchetto (Limesurvey) che accede a un database MySQL ed è dotato di un'interfaccia web. Il sistema operativo dovrà quindi aver installato:

- Apache
- PHP
- MySQL
- Server OpenSSH
- Limesurvey (un'applicazione di esempio creata appositamente)

This is done using vmbuilder by specifying the --addpkg option multiple times:

```
--addpkg PKG
  Install PKG into the guest (can be specfied multiple times)
```

Purtroppo, in base al funzionamento di vmbuilder, i pacchetti che devono porre delle domande nella fase di post-installazione non sono supportati e dovrebbero essere installati successivamente quando

è possibile interagirvi. Questo è il caso di Limesurvey che verrà installato successivamente, dopo che l'utente ha eseguito l'accesso.

Altri pacchetti che pongono delle semplici domande di debconf, come mysql-server che richiede di impostare una password, possono essere installati, ma dovranno essere riconfigurati una volta eseguito l'accesso.

Se alcuni dei pacchetti che si devono installare non sono presenti nel componente "main", è necessario abilitare dei repository aggiuntivi usando le opzioni "--comp" e "--ppa":

```
--components COMP1,COMP2,...,COMPN
           A comma separated list of distro components to include (e.g. main,universe). This defaul
           to "main"
--ppa=PPA  Add ppa belonging to PPA to the vm's sources.list.
```

Limesurvey non fa parte degli archivi attualmente ed è quindi necessario specificarne l'indirizzo PPA (Personal Package Archive) così da aggiungerlo al file `/etc/apt/source.list` della macchina virtuale. Aggiungere quindi quanto segue al comando:

```
--addpkg apache2 --addpkg apache2-mpm-prefork --addpkg apache2-utils --addpkg apache2.2-common \ --
```

2.3.4. Considerazioni sulla velocità

2.3.4.1. Cache dei pacchetti

When vmbuilder creates builds your system, it has to go fetch each one of the packages that composes it over the network to one of the official repositories, which, depending on your internet connection speed and the load of the mirror, can have a big impact on the actual build time. In order to reduce this, it is recommended to either have a local repository (which can be created using apt-mirror) or using a caching proxy such as apt-proxy. The later option being much simpler to implement and requiring less disk space, it is the one we will pick in this tutorial. To install it, simply type:

```
sudo apt-get install apt-proxy
```

Una volta completata l'installazione, il proxy (vuoto) è pronto all'indirizzo "http://INDIRIZZO_MIRROR:9999" e trovaerà i repository Ubuntu sotto "/ubuntu". Affinché vmbuilder possa usarlo, è necessario usare l'opzione *--mirror*:

```
--mirror=URL  Use Ubuntu mirror at URL instead of the default, which
              is http://archive.ubuntu.com/ubuntu for official
              arches and http://ports.ubuntu.com/ubuntu-ports
              otherwise
```

Aggiungere quindi al comando:

```
--mirror http://INDIRIZZO_MIRROR:9999/ubuntu
```

The mirror address specified here will also be used in the `/etc/apt/sources.list` of the newly created guest, so it is useful to specify here an address that can be resolved by the guest or to plan on reseting this address later on.

2.3.4.2. Installare un mirror locale

Se si è in un ambiente molto grande, può aver senso creare un mirror locale dei repository di Ubuntu. Il pacchetto "apt-mirror" fornisce uno script per la gestione delle operazioni di mirror. È utile avere almeno 20GB di spazio per ogni rilascio supportato e architettura.

Il file di configurazione predefinito usato da apt-mirror è `/etc/apt/mirror.list`. Dato che è già impostato, dovrà solamente replicare l'architettura del computer locale. Se è necessario supportare altre architetture all'interno del mirror, basta duplicare le righe che iniziano con “deb”, sostituendo la parola "deb" con "/deb-{arch}", dove "arch" può essere i386, amd64, ecc... Per esempio, su architettura amd64, per avere anche gli archivi per i386:

```
deb  http://archive.ubuntu.com/ubuntu maverick main restricted universe multiverse
/deb-i386  http://archive.ubuntu.com/ubuntu maverick main restricted universe multiverse

deb  http://archive.ubuntu.com/ubuntu maverick-updates main restricted universe multiverse
/deb-i386  http://archive.ubuntu.com/ubuntu maverick-updates main restricted universe multiverse

deb http://archive.ubuntu.com/ubuntu/ maverick-backports main restricted universe multiverse
/deb-i386  http://archive.ubuntu.com/ubuntu maverick-backports main restricted universe multiverse

deb http://security.ubuntu.com/ubuntu maverick-security main restricted universe multiverse
/deb-i386  http://security.ubuntu.com/ubuntu maverick-security main restricted universe multiverse

deb http://archive.ubuntu.com/ubuntu maverick main/debian-installer restricted/debian-installer uni
/deb-i386 http://archive.ubuntu.com/ubuntu maverick main/debian-installer restricted/debian-install
```

I pacchetti dei sorgenti non sono stati inclusi nel mirror dato che non sono molto usati quanto i binari e occupano molto spazio. È comunque possibile aggiungerli facilmente all'elenco.

Una volta terminata l'operazione di duplicazione del mirror (può durare molto), è necessario configurare Apache affinché i file del mirror (in `/var/spool/apt-mirror` se non è stato modificato il valore predefinito) siano pubblicati dal proprio server Apache. Per maggiori informazioni su Apache, consultare *Sezione 1, «HTTPD - Server web Apache2» [141]*.

2.4. Pacchettizzare l'applicativo

Sono disponibili due opzioni:

- Il metodo raccomandato è quello di creare un pacchetto *Debian*. Dato che questo argomento esula da questa guida, non verrà spiegato questo metodo e si rimanda alla *Ubuntu Packaging Guide*[9]. In questo caso è anche utile creare un repository per contenere il pacchetto in modo tale

che gli aggiornamenti vengano prelevati da questo. Per ulteriori informazioni, consultare *Debian Administration*[10].

- Installare l'applicativo nella directory `/opt` come raccomandato dalle *linee guida di FHS*[11].

In questo caso viene usato Limesurvey come esempio di applicazione web per cui creare un applicativo virtuale. Come accennato precedentemente. è disponibile un pacchetto di questa applicazione attraverso gli archivi PPA (Personal Package Archive).

2.5. Utili accorgimenti

2.5.1. Configurare gli aggiornamenti automatici

Affinché il sistema sia configurato per aggiornarsi automaticamente a scadenze determinate, basta installare il pacchetto unattended-upgrades. Aggiungere quindi quanto segue al comando:

```
--addpkg unattended-upgrades
```

Dato che il pacchetto dell'applicazione è stata inserito nel PPA, il processo di aggiornamento non aggiornerà solamente il sistema, ma anche l'applicazione ogni qualvolta ci sia una versione aggiornata nel PPA.

2.5.2. Gestire gli eventi ACPI

Affinché la macchina virtuale possa gestire gli eventi come riavvio e arresto che le vengono inviati, è utile installare anche il pacchetto "acpid". Aggiungere quindi quanto segue al comando:

```
--addpkg acpid
```

2.6. Il comando finale

Ecco il comando con tutte le opzioni presentate poco sopra:

```
sudo vmbuilder kvm ubuntu --suite maverick --flavour virtual --arch i386 -o \
         --libvirt qemu:///system --ip 192.168.0.100 --hostname myvm --part vmbuilder.partition --u
         --name user --pass default --addpkg apache2 --addpkg apache2-mpm-prefork \
         --addpkg apache2-utils --addpkg apache2.2-common --addpkg dbconfig-common \
         --addpkg libapache2-mod-php5 --addpkg mysql-client --addpkg php5-cli \
         --addpkg php5-gd --addpkg php5-ldap --addpkg php5-mysql --addpkg wwwconfig-common \
         --addpkg mysql-server --addpkg unattended-upgrades --addpkg acpid --ppa nijaba \
         --mirror http://mirroraddress:9999/ubuntu
```

2.7. Risorse

Per avere maggiori informaioni, per porre qualche domanda o per lasciare dei suggerimenti, contattare l'«Ubuntu Server Team» presso:

- IRC: #ubuntu-server on freenode
- Mailing list: *ubuntu-server at lists.ubuntu.com*[12]
- Also, see the *JeOSVMBuilder Ubuntu Wiki*[13] page.

3. UEC

3.1. Panoramica

This tutorial covers UEC installation from the Ubuntu 10.10 Server Edition CD, and assumes a basic network topology, with a single system serving as the *"all-in-one controller"*, and one or more nodes attached.

From this Tutorial you will learn how to install, configure, register and perform several operations on a basic UEC setup that results in a cloud with a one controller *"front-end"* and one or several node(s) for running Virtual Machine (VM) instances. You will also use examples to help get you started using your own private compute cloud.

3.2. Prerequisiti

To deploy a minimal cloud infrastructure, you'll need at least *two* dedicated systems:

- Un'interfaccia.
- Uno o più nodi.

The following are recommendations, rather than fixed requirements. However, our experience in developing this documentation indicated the following suggestions.

3.2.1. Front End Requirements

Use the following table for a system that will run one or more of:

- Cloud Controller (CLC)
- Cluster Controller (CC)
- Walrus (the S3-like storage service)
- Storage Controller (SC)

Tabella 19.1. UEC Front End Requirements

Hardware	Minimo	Suggerito	Note
CPU	1 GHz	2 x 2 GHz	Per un'interfaccia *tutta-in-uno* è utile avere almeno un processore dual core.
Memoria	2 GB	4 GB	Per l'interfaccia Java è utile avere molta memoria disponibile.
Disco	5400 RPM IDE	7200 RPM SATA	È possibile utilizzare anche dischi più lenti, ma i tempi di avvio risulteranno più lenti.
Spazio su disco	40 GB	200 GB	40GB è lo spazio sufficiente per una singola immagine, cache, ecc...
Rete	100 Mbps	1000 Mbps	La dimensione delle immagini è di centinaia di megabyte ed è necessario copiare il tutto attraverso la rete verso i nodi.

3.2.2. Requisiti del nodo

The other system(s) are *nodes*, which will run:

- il Node Controller (NC)

Tabella 19.2. Requisiti nodo UEC

Hardware	Minimo	Suggerito	Note
CPU	Estensioni VT	VT, 64-bit, Multicore	64-bit è in grado di eseguire istanze sia i386 che amd64; Eucalyptus eseguirà solamente 1 VM per core di CPU su un nodo.
Memoria	1 GB	4 GB	Più memoria significa guest più grandi e numerosi.
Disco	5400 RPM IDE	7200 RPM SATA or SCSI	I nodi di Eucalyptus sfruttano molto i dischi, le attese di I/O possono causare cali nelle prestazioni.
Spazio su disco	40 GB	100 GB	Le immagini verranno salvate localmente.
Rete	100 Mbps	1000 Mbps	La dimensione delle immagini è di centinaia di megabyte ed è necessario copiare il tutto attraverso la rete verso i nodi.

3.3. Installare l'interfaccia Server Cloud/Cluster/Storage/Walrus

1. Scaricare l'immagine di Ubuntu 10.10 Server e masterizzarla su CD.
2. When you boot, select *"Install Ubuntu Enterprise Cloud"*. The installer will detect if any other Eucalyptus components are present.
3. You can then choose which components to install, based on your chosen *topology*[14].
4. When asked whether you want a *"Cluster"* or a *"Node"* install, select *"Cluster"*.
5. It will ask two other cloud-specific questions during the course of the install:
   - Il nome del cluster.
     - per esempio *cluster1*
   - Un insieme di indirizzi IP pubblici sulla rete che il cloud posso allocare.
     - per esempio *192.168.1.200-192.168.1.249*

3.4. Installare i Node Controller

The node controller install is even simpler. Just make sure that you are connected to the network on which the cloud/cluster controller is already running.

1. Boot from the same ISO on the node(s).
2. When you boot, select *"Install Ubuntu Enterprise Cloud"*.
3. Select *"Install Ubuntu Enterprise Cloud"*.
4. It should detect the Cluster and preselect *"Node"* install for you.
5. Confermare lo schema di partizionamento.
6. The rest of the installation should proceed uninterrupted; complete the installation and reboot the node.

3.5. Registrare i nodi

1. Nodes are the physical systems within UEC that actually run the virtual machine instances of the cloud.

   La registrazione dei componenti dovrebbe essere automatica se:

   a. Public SSH keys have been exchanged properly.
   b. The services are configured properly.
   c. The appropriate *uec-component-listener* is running.
   d. Verify Registration.

   Steps a to e should only be required if you're using the *UEC/PackageInstall*[15] method. Otherwise, if you are following this guide, these steps should already be completed automatically for you, and therefore you can skip *"a"* to *"e"*.
2. Exchange Public Keys

   The Cloud Controller's *eucalyptus* user needs to have SSH access to the Walrus Controller, Cluster Controller, and Storage Controller as the eucalyptus user.

   Install the Cloud Controller's *eucalyptus* user's public ssh key by:

   • On the target controller, temporarily set a password for the eucalyptus user:

   ```
   sudo passwd eucalyptus
   ```

   • Then, on the Cloud Controller:

   ```
   sudo -u eucalyptus ssh-copy-id -i ~eucalyptus/.ssh/id_rsa.pub eucalyptus@<IP_OF_NODE>
   ```

   • You can now remove the password of the eucalyptus account on the target controller, if you wish:

```
sudo passwd -d eucalyptus
```

3. Configurare i servizi

   Nel *Cloud Controller*:

   - Per la registrazione del *Cluster Controller*:
     - Define the shell variable CC_NAME in `/etc/eucalyptus/eucalyptus-cc.conf`
     - Define the shell variable CC_IP_ADDR in `/etc/eucalyptus/eucalyptus-ipaddr.conf`, as a space separated list of one or more IP addresses.
   - Per la registrazione del *Walrus Controller*:
     - Define the shell variable WALRUS_IP_ADDR in `/etc/eucalyptus/eucalyptus-ipaddr.conf`, as a single IP address.

   Nel *Cluster Controller*:

   - Per la registrazione dello *Storage Controller*:
     - Define the shell variable CC_NAME in `/etc/eucalyptus/eucalyptus-cc.conf`
     - Define the shell variable SC_IP_ADDR in `/etc/eucalyptus/eucalyptus-ipaddr.conf`, as a space separated list of one or more IP addresses.

4. Publish

   Now start the publication services.

   - *Walrus Controller:*

     ```
     sudo start eucalyptus-walrus-publication
     ```

   - *Cluster Controller:*

     ```
     sudo start eucalyptus-cc-publication
     ```

   - *Storage Controller:*

     ```
     sudo start eucalyptus-sc-publication
     ```

   - *Node Controller:*

     ```
     sudo start eucalyptus-nc-publication
     ```

5. Start the Listener

   Nel *Cloud Controller* e nei *Cluster Controller*, eseguire:

   ```
   sudo start uec-component-listener
   ```

6. Verificare la registrazione

   ```
   cat /var/log/eucalyptus/registration.log
   ```

```
2010-04-08 15:46:36-05:00 | 24243 -> Calling node cluster1 node 10.1.1.75
2010-04-08 15:46:36-05:00 | 24243 -> euca_conf --register-nodes returned 0
2010-04-08 15:48:47-05:00 | 25858 -> Calling walrus Walrus 10.1.1.71
2010-04-08 15:48:51-05:00 | 25858 -> euca_conf --register-walrus returned 0
2010-04-08 15:49:04-05:00 | 26237 -> Calling cluster cluster1 10.1.1.71
2010-04-08 15:49:08-05:00 | 26237 -> euca_conf --register-cluster returned 0
2010-04-08 15:49:17-05:00 | 26644 -> Calling storage cluster1 storage 10.1.1.71
2010-04-08 15:49:18-05:00 | 26644 -> euca_conf --register-sc returned 0
```

L'output sul proprio computer potrebbe essere diverso dall'esempio precedente.

3.6. Ottenere le credenziali

After installing and booting the *Cloud Controller*, users of the cloud will need to retrieve their credentials. This can be done either through a web browser, or at the command line.

3.6.1. Da un browser

1. From your web browser (either remotely or on your Ubuntu server) access the following URL:

   ```
   https://<indirizzo-ip-cloud-controller>:8443/
   ```

   

   You must use a secure connection, so make sure you use "https" not "http" in your URL. You will get a security certificate warning. You will have to add an exception to view the page. If you do not accept it you will not be able to view the Eucalyptus configuration page.
2. Use username *'admin'* and password *'admin'* for the first time login (you will be prompted to change your password).
3. Then follow the on-screen instructions to update the admin password and email address.
4. Once the first time configuration process is completed, click the *'credentials'* tab located in the top-left portion of the screen.
5. Click the *'Download Credentials'* button to get your certificates.
6. Salvare il tutto in `~/.euca`.
7. Unzip the downloaded zip file into a safe location (`~/.euca`).

   ```
   unzip -d ~/.euca mycreds.zip
   ```

3.6.2. Dalla riga di comando

- Alternatively, if you are on the command line of the *Cloud Controller*, you can run:

  ```
  mkdir -p ~/.euca
  chmod 700 ~/.euca
  ```

```
cd ~/.euca
sudo euca_conf --get-credentials mycreds.zip
unzip mycreds.zip
ln -s ~/.euca/eucarc ~/.eucarc
cd -
```

3.6.3. Estrarre e utilizzare le credenziali

Now you will need to setup EC2 API and AMI tools on your server using X.509 certificates.

1. Installare gli strumenti richiesti:

```
sudo apt-get install euca2ools
```

2. Per verificare che tutto funzioni correttamente, recuperare i dettagli di disponibilità del cluster locale:

```
. ~/.euca/eucarc
euca-describe-availability-zones verbose
AVAILABILITYZONE   myowncloud               192.168.1.1
AVAILABILITYZONE   |- vm types              free / max   cpu   ram  disk
AVAILABILITYZONE   |- m1.small              0004 / 0004   1    128     2
AVAILABILITYZONE   |- c1.medium             0004 / 0004   1    256     5
AVAILABILITYZONE   |- m1.large              0002 / 0002   2    512    10
AVAILABILITYZONE   |- m1.xlarge             0002 / 0002   2   1024    20
AVAILABILITYZONE   |- c1.xlarge             0001 / 0001   4   2048    20
```

L'output del comando precedente potrebbe essere diverso.

3.7. Install an Image from the Store

The following is by far the simplest way to install an image. However, advanced users may be interested in learning how to *Bundle their own image*[16].

The simplest way to add an image to UEC is to install it from the Image Store on the UEC web interface.

1. Access the web interface at the following URL (Make sure you specify https):

```
https://<indirizzo-ip-cloud-controller>:8443/
```

2. Enter your login and password (if requested, as you may still be logged in from earlier).
3. Click on the *Store* tab.
4. Browse available images.
5. Click on *install* for the image you want.

[16] https://help.ubuntu.com/community/UEC/BundlingImages

Once the image has been downloaded and installed, you can click on *"How to run?"* that will be displayed below the image button to view the command to execute to instantiate (start) this image. The image will also appear on the list given on the *Image* tab.

3.8. Eseguire un'immagine

Ci sono diversi modi per inizializzare un'immagine in UEC:

- Usare la riga di comando.
- Use one of the UEC compatible management tools such as *Landscape*.
- Use the *ElasticFox*[17] extension to Firefox.

Di seguito viene descritta la procedura dalla riga di comando:

1. Before running an instance of your image, you should first create a *keypair* (ssh key) that you can use to log into your instance as root, once it boots. The key is stored, so you will only have to do this once.

   Eseguire il seguente comando:

   ```
   if [ ! -e ~/.euca/mykey.priv ]; then
       mkdir -p -m 700 ~/.euca
       touch ~/.euca/mykey.priv
       chmod 0600 ~/.euca/mykey.priv
       euca-add-keypair mykey > ~/.euca/mykey.priv
   fi
   ```

   

   You can call your key whatever you like (in this example, the key is called *'mykey'*), but remember what it is called. If you forget, you can always run **euca-describe-keypairs** to get a list of created keys stored in the system.

2. È necessario consentire accesso alla porta 22 in tutte le istanze:

   ```
   euca-authorize default -P tcp -p 22 -s 0.0.0.0/0
   ```

3. È quindi possibile creare istanze delle proprie immagini registrate:

   ```
   euca-run-instances $EMI -k mykey -t m1.small
   ```

   

   If you receive an error regarding *image_id*, you may find it by viewing Images page or click *"How to Run"* on the *Store* page to see the sample command.

4. The first time you run an instance, the system will be setting up caches for the image from which it will be created. This can often take some time the first time an instance is run given that VM images are usually quite large.

   To monitor the state of your instance, run:

```
watch -n5 euca-describe-instances
```

In the output, you should see information about the instance, including its state. While first-time caching is being performed, the instance's state will be *'pending'*.

5. When the instance is fully started, the above state will become *'running'*. Look at the IP address assigned to your instance in the output, then connect to it:

```
IPADDR=$(euca-describe-instances | grep $EMI | grep running | tail -n1 | awk '{print $4}')
ssh -i ~/.euca/mykey.priv ubuntu@$IPADDR
```

6. And when you are done with this instance, exit your SSH connection, then terminate your instance:

```
INSTANCEID=$(euca-describe-instances | grep $EMI | grep running | tail -n1 | awk '{print $2}')
euca-terminate-instances $INSTANCEID
```

3.8.1. Primo avvio

The cloud-init package provides "first boot" functionality for the Ubuntu UEC images. It is in charge of taking the generic filesystem image that is booting and customizing it for this particular instance. That includes things like:

- Setting the hostname.
- Putting the provided ssh public keys into `~ubuntu/.ssh/authorized_keys`.
- Running a user provided script, or otherwise modifying the image.

Setting hostname and configuring a system so the person who launched it can actually log into it are not terribly interesting. The interesting things that can be done with cloud-init are made possible by data provided at launch time called *user-data*[18].

First, install the cloud-init package:

```
sudo apt-get install cloud-init
```

If the user-data starts with *'#!'*, then it will be stored and executed as root late in the boot process of the instance's first boot (similar to a traditional 'rc.local' script). Output from the script is directed to the console.

Per esempio, creare un file chiamato `ud.txt` che contiene quanto segue:

```
#!/bin/sh
echo ========== Hello World: $(date) ==========
echo "I have been up for $(cut -d\  -f 1 < /proc/uptime) sec"
```

Ovviare un'istanza con l'opzione *--user-data-file*:

[18] http://developer.amazonwebservices.com/connect/entry.jspa?externalID=1085

```
euca-run-instances $EMI -k mykey -t m1.small --user-data-file=ud.txt
```

Attendere che il sistema e la console siano disponibili. Per visualizzare i risultati, digitare:

```
euca-get-console-output $EMI | grep --after-context=1 Hello
========== Hello World: Mon Mar 29 18:05:05 UTC 2010 ==========
I have been up for 28.26 sec
```

L'output del proprio comando potrebbe variare.

The simple approach shown above gives a great deal of power. The user-data can contain a script in any language where an interpreter already exists in the image (#!/bin/sh, #!/usr/bin/python, #!/usr/bin/perl, #!/usr/bin/awk ...).

For many cases, the user may not be interested in writing a program. For this case, cloud-init provides *"cloud-config"*, a configuration based approach towards customization. To utilize the cloud-config syntax, the supplied user-data must start with a *'#cloud-config'*.

Per esempio, creare un file di testo chiamato `clout-config.txt` contenente:

```
#cloud-config
apt_upgrade: true
apt_sources:
- source: "ppa:ubuntu-server-edgers/server-edgers-apache "

packages:
- build-essential
- pastebinit

runcmd:
- echo ======= Hello World =====
- echo "I have been up for $(cut -d\  -f 1 < /proc/uptime) sec"
```

Creare una nuova istanza:

```
euca-run-instances $EMI -k mykey -t m1.small --user-data-file=cloud-config.txt
```

Una volta avviato il sistema, dovrebbe avere:

- Added the Apache Edgers PPA.
- Run an upgrade to get all updates available
- Installed the 'build-essential' and 'pastebinit' packages
- Printed a similar message to the script above

The *Apache Edgers PPA*, in the above example, contains the latest version of Apache from upstream source repositories. Package versions in the PPA are unsupported, and depending

on your situation, this may or may not be desirable. See the *Ubuntu Server Edgers*[19] web page for more details.

The *'runcmd'* commands are run at the same point in boot that the *'#!'* script would run in the previous example. It is present to allow you to get the full power of a scripting language if you need it without abandoning *cloud-config*.

For more information on what kinds of things can be done with cloud-config, see *doc/examples*[20] in the source.

3.9. Ulteriori informazioni

How to use the *Storage Controller*[21]

Controllare i servizi di Eucalyptus

- sudo service eucalyptus [start|stop|restart] (on the CLC/CC/SC/Walrus side)
- sudo service eucalyptus-nc [start|stop|restart] (on the Node side)

Posizione di alcuni dei file importanti:

- *File di registro:*
  - /var/log/eucalyptus
- *File di configurazione:*
  - /etc/eucalyptus
- *Database:*
  - /var/lib/eucalyptus/db
- *Chiavi:*
  - /var/lib/eucalyptus
  - /var/lib/eucalyptus/.ssh

Don't forget to source your `~/.euca/eucarc` before running the client tools.

3.10. Riferimenti

- Per informazioni sul caricamento delle istanze, consultare la *documentazione della comunità internazionale*[22].
- *Eucalyptus Project Site (forums, documentation, downloads)*[23].
- *Eucalyptus on Launchpad (bugs, code)*[24].
- *Eucalyptus Troubleshooting (1.5)*[25].
- *Register your cloud with RightScale*[26].

[20] http://bazaar.launchpad.net/~cloud-init-dev/cloud-init/trunk/files/head:/doc/examples/

[21] https://help.ubuntu.com/community/UEC/StorageController

- È anche possibile trovare aiuto nei canali IRC *#ubuntu-virt*, *#eucalyptus*e *#ubuntu-server* sul server *Freenode*[27].

3.11. Glossario

The Ubuntu Enterprise Cloud documentation uses terminology that might be unfamiliar to some readers. This page is intended to provide a glossary of such terms and acronyms.

- *Cloud* - A federated set of physical machines that offer computing resources through virtual machines, provisioned and recollected dynamically.
- *Cloud Controller (CLC)* - Eucalyptus component that provides the web UI (an https server on port 8443), and implements the Amazon EC2 API. There should be only one Cloud Controller in an installation of UEC. This service is provided by the Ubuntu eucalyptus-cloud package.
- *Cluster* - A collection of nodes, associated with a Cluster Controller. There can be more than one Cluster in an installation of UEC. Clusters are sometimes physically separate sets of nodes. (e.g. floor1, floor2, floor2).
- *Cluster Controller (CC)* - Eucalyptus component that manages collections of node resources. This service is provided by the Ubuntu eucalyptus-cc package.
- *EBS* - Elastic Block Storage.
- *EC2* - Elastic Compute Cloud. Amazon's pay-by-the-hour, pay-by-the-gigabyte public cloud computing offering.
- *EKI* - Eucalyptus Kernel Image.
- *EMI* - Eucalyptus Machine Image.
- *ERI* - Eucalyptus Ramdisk Image.
- *Eucalyptus* - Elastic Utility Computing Architecture for Linking Your Programs To Useful Systems. An open source project originally from the University of California at Santa Barbara, now supported by Eucalyptus Systems, a Canonical Partner.
- *Front-end* - Physical machine hosting one (or more) of the high level Eucalyptus components (cloud, walrus, storage controller, cluster controller).
- *Node* - A node is a physical machine that's capable of running virtual machines, running a node controller. Within Ubuntu, this generally means that the CPU has VT extensions, and can run the KVM hypervisor.
- *Node Controller (NC)* - Eucalyptus component that runs on nodes which host the virtual machines that comprise the cloud. This service is provided by the Ubuntu package eucalyptus-nc.
- *S3* - Simple Storage Service. Amazon's pay-by-the-gigabyte persistent storage solution for EC2.
- *Storage Controller (SC)* - Eucalyptus component that manages dynamic block storage services (EBS). Each 'cluster' in a Eucalyptus installation can have its own Storage Controller. This component is provided by the eucalyptus-sc package.
- *UEC* - Ubuntu Enterprise Cloud. Ubuntu's cloud computing solution, based on Eucalyptus.
- *VM* - Virtual Machine.

- *VT* - Virtualization Technology. An optional feature of some modern CPUs, allowing for accelerated virtual machine hosting.
- *Walrus* - Eucalyptus component that implements the Amazon S3 API, used for storing VM images and user storage using S3 bucket put/get abstractions.

Capitolo 20. Cluster

1. DRBD

DRDB (Distributed Replicated Block Device) replica i device a blocchi tra diversi host. La replica è trasparente alle applicazioni sul sistema host e qualsiasi device a blocchi (disco fisso, partizione, RAID, volume logico) può essere replicato.

Per utilizzare drbd, per prima cosa è necessario installare i pacchetti necessari. In un terminale digitare:

```
sudo apt-get install drbd8-utils
```

Se si sta usando il *kernel virtuale* come parte di una macchina virtuale, è necessario compilare il modulo drbd. Potrebbe anche essere più semplice installare il pacchetto linux-server nella macchina virtuale.

In questa sezione viene indicato come configurare drbd per replicare tra due host una partizione `/srv` separata con file system ext3. La dimensione della partizione non è rilevante, ma entrambe le partizioni devono avere la stessa dimensione.

1.1. Configurazione

I due host in questo esempio sono chiamati *drbd01* e *drbd02* ed è necessario configurarne la risoluzione del nome attraverso DNS o con il file `/etc/hosts`. Per maggiori informazioni, consultare *Capitolo 7, DNS (Domain Name Service) [94]*.

- Per configurare drbd, sul primo host modificare il file `/etc/drbd.conf`:

```
global { usage-count no; }
common { syncer { rate 100M; } }
resource r0 {
        protocol C;
        startup {
                wfc-timeout  15;
                degr-wfc-timeout 60;
        }
        net {
                cram-hmac-alg sha1;
                shared-secret "secret";
        }
        on drbd01 {
                device /dev/drbd0;
                disk /dev/sdb1;
                address 192.168.0.1:7788;
                meta-disk internal;
        }
        on drbd02 {
                device /dev/drbd0;
                disk /dev/sdb1;
```

```
                address 192.168.0.2:7788;
                meta-disk internal;
        }
}
```

All'interno del file `/etc/drbd.conf` sono disponibili molte opzioni, ma per questo esempio i valori predefinito sono sufficienti.

- Copiare il file `/etc/drbd.conf` sul secondo host:

```
scp /etc/drbd.conf drbd02:~
```

- Sull'host *drbd02*, spostare il file in `/etc`:

```
sudo mv drbd.conf /etc/
```

- Utilizzando l'utilità drbdadm, inizializzare l'archivio dei meta-dati. Su ogni singolo server eseguire il seguente comando:

```
sudo drbdadm create-md r0
```

- Su entrambi gli host, avviare il demone drbd:

```
sudo /etc/init.d/drbd start
```

- Sull'host *drbd01*, o su qualsiasi host primario configurato, digitare:

```
sudo drbdadm -- --overwrite-data-of-peer primary all
```

- Una volta eseguito il comando precedente, inizierà la sincronizzazione dei dati con l'host secondario. Per visualizzare l'avanzamento, su *drbd02*, digitare il seguente comando:

```
watch -n1 cat /proc/drbd
```

  Per fermare l'operazione di controllo, premere *Ctrl+c*.

- Infine, aggiungere un file system a `/dev/drbd0` e montarlo:

```
sudo mkfs.ext3 /dev/drbd0
sudo mount /dev/drbd0 /srv
```

1.2. Test

Per verificare che i dati siano effettivamente sincronizzati tra gli host, copiare alcuni file sull'host primario, *drbd01*, nella directory `/srv`:

```
sudo cp -r /etc/default /srv
```

Smontare `/srv`:

```
sudo umount /srv
```

Retrocedere il server *primario* a ruolo di *secondario*:

```
sudo drbdadm secondary r0
```

Ora, *promuovere* il server *secondario* a *primario*:

```
sudo drbdadm primary r0
```

Per completare, montare la partizione:

```
sudo mount /dev/drbd0 /srv
```

Usando *ls* dovrebbe essere possibile vedere il file `/srv/default` copiato dal precedente host *primario drbd01*.

1.3. Riferimenti

- Per maggiori informazioni riguardo DRBD, consultare il *sito web di DRBD*[1].
- La *pagina di manuale di drbd.conf*[2] contiene ulteriori informazioni riguardo le opzioni disponibili.
- Consultare anche la *pagina di manuale di drbdadm*[3].
- Ulteriori informazioni sono disponibili nella *documentazione online*[4].

Capitolo 21. VPN

Una Virtual Private Network, o *VPN*, è una connessione di rete cifrata tra due o più reti. Esistono diversi modi per creare una VPN, sia usando particolare software che delle appliance hardware dedicate. In questo capitolo verrà trattata l'installazione e la configurazione di OpenVPN per creare una VPN tra due server.

1. OpenVPN

OpenVPN fa uso di una PKI (Public Key Infrastructure) per cifrare il traffico VPN tra i nodi.
Un modo semplice di impostare una VPN attraverso OpenVPN è di connettere i client attraverso un'interfaccia bridge sul server VPN. Questa guida parte dal presupposto che un nodo VPN, in questo caso il server, presenti un'interfaccia bridge configurata. Per maggiori informazioni su come impostare un'interfaccia bridge, consultare *Sezione 1.4, «Bridging» [37]*.

1.1. Installazione

Per installare openvpn, in un terminale, digitare:

```
sudo apt-get install openvpn
```

1.1.1. Certificati server

Installato il pacchetto openvpn, è necessario creare i certificati.

Per prima cosa, copiare la directory `easy-rsa` in `/etc/openvpn`: in questo modo qualsiasi modifica fatta agli script non verrà persa in caso di aggiornamento del pacchetto. È necessario anche modificare i permessi alla directory `easy-rsa` per consentire all'utente attuale di creare file. Da un terminale, digitare:

```
sudo mkdir /etc/openvpn/easy-rsa/
sudo cp -r /usr/share/doc/openvpn/examples/easy-rsa/2.0/* /etc/openvpn/easy-rsa/
sudo chown -R $USER /etc/openvpn/easy-rsa/
```

Modificare quindi il file `/etc/openvpn/easy-rsa/vars` sistemando quanto segue al proprio ambiente:

```
export KEY_COUNTRY="IT"
export KEY_PROVINCE="Roma"
export KEY_CITY="Roma"
export KEY_ORG="Società di esempio"
export KEY_EMAIL="mario@example.com"
```

Digitare quanto segue per creare i certificati del server:

```
cd /etc/openvpn/easy-rsa/
source vars
./clean-all
./build-dh
./pkitool --initca
./pkitool --server server
cd keys
openvpn --genkey --secret ta.key
sudo cp server.crt server.key ca.crt dh1024.pem ta.key /etc/openvpn/
```

1.1.2. Certificati client

Il client VPN necessita anche di un certificato per autenticarsi sul server. Per creare il certificato, inserire quanto segue in un terminale:

```
cd /etc/openvpn/easy-rsa/
source vars
./pkitool hostname
```

Sostituire *hostname* con il nome del computer che si collega alla rete VPN.

Copiare i seguenti file nel client:

- /etc/openvpn/ca.crt
- /etc/openvpn/easy-rsa/keys/hostname.crt
- /etc/openvpn/easy-rsa/keys/hostname.key
- /etc/openvpn/ta.key

Ricordarsi di modificare il nome del file precedente affinché rispecchi il *nome host* del computer client.

È raccomandato usare un metodo sicuro per copiare il certificato e i file di chiave. L'utilità scp è una ottima scelta, ma anche copiare i file su un supporto rimovibile e poi sul client funziona in modo adeguato.

1.2. Configurazione

1.2.1. Configurazione del server

Ora configurare il server openvpn creando il file `/etc/openvpn/server.conf` a partire dal file d'esempio. In un terminare inserire:

```
sudo cp /usr/share/doc/openvpn/examples/sample-config-files/server.conf.gz /etc/openvpn/
sudo gzip -d /etc/openvpn/server.conf.gz
```

Modificare il file `/etc/openvpn/server.conf` cambiando le seguenti opzioni:

```
local 172.18.100.101
dev tap0
up "/etc/openvpn/up.sh br0"
down "/etc/openvpn/down.sh br0"
;server 10.8.0.0 255.255.255.0
server-bridge 172.18.100.101 255.255.255.0 172.18.100.105 172.18.100.200
push "route 172.18.100.1 255.255.255.0"
push "dhcp-option DNS 172.18.100.20"
push "dhcp-option DOMAIN example.com"
tls-auth ta.key 0 # This file is secret
```

```
user nobody
group nogroup
```

- *local*: è l'indirizzo IP dell'interfaccia bridge.
- *server-bridge*: necessario quando la comunicazione utilizza il "bridging". La parte *172.18.100.101 255.255.255.0* è l'interfaccia bridge e la maschera. L'intervallo degli indirizzi *172.18.100.105 172.18.100.200* indica quali indirizzi possono essere assegnati ai client.
- *push*: sono direttive per aggiungere opzioni di rete ai client.
- *user e group*: configura l'utente e il gruppo con cui viene mandato in esecuzione il demone openvpn.

Sostituire tutti gli indirizzi IP e i nomi di dominio precedenti con i valori della propria rete.

Come passo seguente, creare una coppia di script helper per aggiungere l'interfaccia *tap* al bridge. Creare `/etc/openvpn/up.sh`:

```
#!/bin/sh

BR=$1
DEV=$2
MTU=$3
/sbin/ifconfig $DEV mtu $MTU promisc up
/usr/sbin/brctl addif $BR $DEV
```

E `/etc/openvpn/down.sh`:

```
#!/bin/sh

BR=$1
DEV=$2

/usr/sbin/brctl delif $BR $DEV
/sbin/ifconfig $DEV down
```

Per renderli eseguibili:

```
sudo chmod 755 /etc/openvpn/down.sh
sudo chmod 755 /etc/openvpn/up.sh
```

Una volta configurato il server, riavviare openvpn digitando:

```
sudo /etc/init.d/openvpn restart
```

1.2.2. Configurazione del client

Installare openvpn sul client:

```
sudo apt-get install openvpn
```

Configurato il server e copiati i certificati del client nella directory `/etc/openvpn/`, creare un file di configurazione per il client copiando l'esempio. Nel computer client, da un terminale, digitare:

```
sudo cp /usr/share/doc/openvpn/examples/sample-config-files/client.conf /etc/openvpn
```

Ora modificare `/etc/openvpn/client.conf` sistemando le seguenti opzioni:

```
dev tap
remote vpn.example.com 1194
cert hostname.crt
key hostname.key
tls-auth ta.key 1
```

Sostituire *vpn.example.com* con il nome host del proprio server VPN e *hostname.** con i nomi dei file del certificato e della chiave reali.

Infine, riavviare openvpn:

```
sudo /etc/init.d/openvpn restart
```

Ora dovrebbe essere possibile connettersi alla rete LAN remota attraverso VPN.

1.3. Riferimenti

- Per maggiori informazioni, consultare il sito web di *OpenVPN*[1].
- Un'ottima risorsa è anche *OpenVPN: Building and Integrating Virtual Private Networks*[2] di Pakt (in inglese).
- Ulteriori informazioni possono essere trovate nella *documentazione online*[3].

Capitolo 22. Altre utili applicazioni

Esistono molte applicazioni sviluppate dallo Ubuntu Server Team e altre integrate all'interno della Ubuntu Server Edition che non sono molto conosciute. Questo capitolo presenta alcune di queste utili applicazioni che possono rendere l'amministrazione di un server Ubuntu, o di molti server, più facile.

1. pam_motd

Quando si esegue l'accesso con una versione server di Ubuntu, è possibile vedere dei messaggi giornalieri di informazioni (MOTD). Queste informazioni sono ricavate e visualizzate utilizzando diversi pacchetti:

- *landscape-common*: fornisce le librerie principali di landscape-client, che può essere utilizzato per la gestione di sistemi attraverso l'interfaccia web di *Landscape*. Il pacchetto comprende l'utilità / usr/bin/landscape-sysinfo che può essere usata per recuperare informazioni visualizzate attraverso il MOTD.
- *update-notifier-common:* is used to automatically update the MOTD via pam_motd module.

pam_motd executes the scripts in `/etc/update-motd.d` in order based on the number prepended to the script. The output of the scripts is written to `/var/run/motd`, keeping the numerical order, then concatenated with `/etc/motd.tail`.

È possibile aggiungere delle informazioni dinamiche per il messaggio giornaliero. Per esempio, per aggiungere informazioni meteo locali:

- Installare il pacchetto weather-util:

  ```
  sudo apt-get install weather-util
  ```

- L'utilità weather utilizza i dati METAR dalla «National Oceanic and Atmospheric Administration» e le previsioni meteo dal «National Weather Service». Per reperire informazioni locali è necessario il codice a 4 cifre ICAO. Per ottenere questo codice è possibile consultare il sito web del *National Weather Service*[1].

  Benché il «National Weather Service» sia un'agenzia governativa degli Stati Uniti d'America, stazioni meteo sono disponibili in tutti il mondo. Informazioni meteorologiche potrebbero però non essere disponibili per tutte le località al di fuori del territorio americano.

- Creare il file `/usr/local/bin/local-weather`, un semplice script per usare weather con il proprio indicatore ICAO locale:

  ```
  #!/bin/sh
  #
  #
  # Prints the local weather information for the MOTD.
  #
  #

  # Replace KINT with your local weather station.
  # Local stations can be found here: http://www.weather.gov/tg/siteloc.shtml

  echo
  weather -i KINT
  echo
  ```

- Rendere lo script eseguibile:

```
sudo chmod 755 /usr/local/bin/local-weather
```

- Next, create a symlink to `/etc/update-motd.d/98-local-weather`:

```
sudo ln -s /usr/local/bin/local-weather /etc/update-motd.d/98-local-weather
```

- Finally, exit the server and re-login to view the new MOTD.

You should now be greeted with some useful information, and some information about the local weather that may not be quite so useful. Hopefully the local-weather example demonstrates the flexibility of pam_motd.

2. etckeeper

etckeeper consente di archiviare il contenuto della directory `/etc` in un sistema di controllo della versione e si integra con apt per inviare le modifiche apportate a `/etc` quando vengono installati o aggiornati pacchetti. Utilizzare un sistema di controllo della versione per gestire la directory `/etc` è considerata una "best practice" e l'obiettivo di etckeeper è quello di rendere questo processo il più facile possibile.

Installare etckeeper digitando quanto segue in un terminale:

```
sudo apt-get install etckeeper
```

Il file di configurazione principale, `/etc/etckeeper/etckeeper.conf`, è molto semplice. L'opzione principale è quale VCS usare. Come impostazione predefinita, etckeeper utilizza bzr per il controllo della versione. Il repository viene automaticamente inizializzato (e viene eseguito il primo commit) durante l'installazione del pacchetto. È possibile annullare questo inserendo il seguente comando:

```
sudo etckeeper uninit
```

Il programma etckeeper esegue i commint delle modifiche a /etc giornalmente.
Questo comportamento può essere disabilitato usando l'opzione di configurazione AVOID_DAILY_AUTOCOMMITS. Inoltre, esegue i commit delle modifiche prima di ogni installazione di un pacchetto. Per un tracciamento delle modifiche più preciso, è consigliato eseguire i commit manualmente aggiungendovi anche un messaggio di commit:

```
sudo etckeeper commit "..Commento sulle modifiche.."
```

Utilizzando i comandi del sistema di controllo è possibile visualizzare il registro delle informazioni riguardo i file in `/etc`:

```
sudo bzr log /etc/passwd
```

Per una dimostrazione dell'integrazione col sistema di gestione dei pacchetti, installare postfix:

```
sudo apt-get install postfix
```

Completata l'installazione, tutti i file di configurazione di postfix dovrebbero essere inviati al repository:

```
Committing to: /etc/
added aliases.db
modified group
modified group-
modified gshadow
```

```
modified gshadow-
modified passwd
modified passwd-
added postfix
added resolvconf
added rsyslog.d
modified shadow
modified shadow-
added init.d/postfix
added network/if-down.d/postfix
added network/if-up.d/postfix
added postfix/dynamicmaps.cf
added postfix/main.cf
added postfix/master.cf
added postfix/post-install
added postfix/postfix-files
added postfix/postfix-script
added postfix/sasl
added ppp/ip-down.d
added ppp/ip-down.d/postfix
added ppp/ip-up.d/postfix
added rc0.d/K20postfix
added rc1.d/K20postfix
added rc2.d/S20postfix
added rc3.d/S20postfix
added rc4.d/S20postfix
added rc5.d/S20postfix
added rc6.d/K20postfix
added resolvconf/update-libc.d
added resolvconf/update-libc.d/postfix
added rsyslog.d/postfix.conf
added ufw/applications.d/postfix
Committed revision 2.
```

Per un esempio di come etckeeper tiene traccia delle modifiche manuali, aggiungere un nuovo host in `/etc/hosts`. Usando bzr è possibile visualizzare quali file sono stati modificati:

```
sudo bzr status /etc/
modified:
  hosts
```

Ora inviare le modifiche:

```
sudo etckeeper commit "nuovo host"
```

Per maggiori informazioni su bzr consultare *Sezione 1, «Bazaar» [215]*.

3. Byobu

One of the most useful applications for any system administrator is screen. It allows the execution of multiple shells in one terminal. To make some of the advanced screen features more user friendly, and provide some useful information about the system, the byobu package was created.

When executing byobu pressing the *F9* key will bring up the Configuration menu. This menu will allow you to:

- Visualizzare il menù dell'aiuto
- Change Byobu's background color
- Change Byobu's foreground color
- Toggle status notifications
- Modificare le associazioni dei tasti
- Modificare la sequenza di escape
- Create new windows
- Gestire le finestre predefinite
- Byobu currently does not launch at login (toggle on)

The *key bindings* determine such things as the escape sequence, new window, change window, etc. There are two key binding sets to choose from *f-keys* and *screen-escape-keys*. If you wish to use the original key bindings choose the *none* set.

byobu provides a menu which displays the Ubuntu release, processor information, memory information, and the time and date. The effect is similar to a desktop menu.

Using the *"Byobu currently does not launch at login (toggle on)"* option will cause byobu to be executed any time a terminal is opened. Changes made to byobu are on a per user basis, and will not affect other users on the system.

One difference when using byobu is the *scrollback* mode. Press the *F7* key to enter scrollback mode. Scrollback mode allows you to navigate past output using *vi* like commands. Here is a quick list of movement commands:

- *h*: sposta il cursore a sinistra di un carattere
- *j*: sposta il cursore in giù di una riga
- *k*: sposta il cursore in su di una riga
- *l*: sposta il cursore a destra di un carattere
- *0*: va all'inizio della riga attuale
- *$*: va alla fine della riga attuale
- *G*: va alla riga specificata (come valore predefinito va alla fine del buffer)
- */*: cerca in avanti

- *?*: cerca all'indietro
- *n*: si sposta alla corrispondenza successiva, in avanti o all'indietro

4. Riferimenti

- See the *update-motd man page*[2] for more options available to update-motd.
- L'articolo di «The Debian Package of the Day» riguardo *weather*[3], presenta molte altre informazioni.
- Per maggiori informazioni riguardo l'uso di etckeeper, consultare il *sito web di etckeeper*[4].
- The *etckeeper Ubuntu Wiki*[5] page.
- Per maggiori informazioni riguardo bzr, consultare il *sito web di bzr*[6].
- Per maggiori informazioni riguardo screen, consultare il *sito web di screen*[7].
- And the *Ubuntu Wiki screen*[8] page.
- Also, see the byobu *project page*[9] for more information.

Appendice A. Appendix

1. Reporting Bugs in Ubuntu Server Edition

While the Ubuntu Project attempts to release software with as few bugs as possible, they do occur. You can help fix these bugs by reporting ones that you find to the project. The Ubuntu Project uses *Launchpad*[1] to track its bug reports. In order to file a bug about Ubuntu Server on Launchpad, you will need to *create an account*[2].

1.1. Reporting Bugs With ubuntu-bug

The preferred way to report a bug is with the ubuntu-bug command. The ubuntu-bug tool gathers information about the system useful to developers in diagnosing the reported problem that will then be included in the bug report filed on Launchpad. Bug reports in Ubuntu need to be filed against a specific software package, thus the name of the package that the bug occurs in needs to be given to ubuntu-bug:

```
ubuntu-bug PACKAGENAME
```

For example, to file a bug against the openssh-server package, you would do:

```
ubuntu-bug openssh-server
```

You can specify either a binary package or the source package for ubuntu-bug. Again using openssh-server as an example, you could also generate the report against the source package for openssh-server, openssh:

```
ubuntu-bug openssh
```

See *Capitolo 3, Gestione dei pacchetti [17]* for more information about packages in Ubuntu.

The ubuntu-bug command will gather information about the system in question, possibly including information specific to the specified package, and then ask you what you would like to do with collected information:

```
ubuntu-bug postgresql

*** Collecting problem information

The collected information can be sent to the developers to improve the
application. This might take a few minutes.
..........
```

[1] https://launchpad.net/

[2] https://help.launchpad.net/YourAccount/NewAccount

```
*** Send problem report to the developers?

After the problem report has been sent, please fill out the form in the
automatically opened web browser.

What would you like to do? Your options are:
  S: Send report (1.7 KiB)
  V: View report
  K: Keep report file for sending later or copying to somewhere else
  C: Cancel
Please choose (S/V/K/C):
```

The options available are:

- **Send Report** Selecting Send Report submits the collected information to Launchpad as part of the the process of filing a bug report. You will be given the opportunity to describe the situation that led up to the occurrence of the bug.

  ```
  *** Uploading problem information

  The collected information is being sent to the bug tracking system.
  This might take a few minutes.
  91%

  *** To continue, you must visit the following URL:

    https://bugs.launchpad.net/ubuntu/+source/postgresql-8.4/+filebug/kc6eSnTLnLxF8u0t3e56EukFeqJ?

  You can launch a browser now, or copy this URL into a browser on another
  computer.

  Choices:
    1: Launch a browser now
    C: Cancel
  Please choose (1/C):
  ```

  If you choose to start a browser, by default the text based web browser w3m will be used to finish filing the bug report. Alternately, you can copy the given URL to a currently running web browser.

- **View Report** Selecting View Report causes the collected information to be displayed to the terminal for review.

  ```
  Package: postgresql 8.4.2-2
  PackageArchitecture: all
  Tags: lucid
  ProblemType: Bug
  ProcEnviron:
    LANG=en_US.UTF-8
    SHELL=/bin/bash
  Uname: Linux 2.6.32-16-server x86_64
  Dependencies:
  ```

```
  adduser 3.112ubuntu1
  base-files 5.0.0ubuntu10
  base-passwd 3.5.22
  coreutils 7.4-2ubuntu2
...
```

After viewing the report, you will be brought back to the same menu asking what you would like to do with the report.

- **Keep Report File** Selecting Keep Report File causes the gathered information to be written to a file. This file can then be used to later file a bug report or transferred to a different Ubuntu system for reporting. To submit the report file, simply give it as an argument to the ubuntu-bug command:

```
What would you like to do? Your options are:
  S: Send report (1.7 KiB)
  V: View report
  K: Keep report file for sending later or copying to somewhere else
  C: Cancel
Please choose (S/V/K/C): k
Problem report file: /tmp/apport.postgresql.v4MQas.apport
```

```
ubuntu-bug /tmp/apport.postgresql.v4MQas.apport
```

```
*** Send problem report to the developers?
...
```

- **Cancel** Selecting Cancel causes the collected information to be discarded.

1.2. Reporting Application Crashes

The software package that provides the ubuntu-bug utility, apport, can be configured to trigger when applications crash. This is disabled by default, as capturing a crash can be resource intensive depending on how much memory the application that crashed was using as apport captures and processes the core dump.

Configuring apport to capture information about crashing applications requires a couple of steps. First, gdb needs to be installed; it is not installed by default in Ubuntu Server Edition.

```
sudo apt-get install gdb
```

See *Capitolo 3, Gestione dei pacchetti [17]* for more information about managing packages in Ubuntu.

Once you have ensured that gdb is installed, open the file `/etc/default/apport` in your text editor, and change the *enabled* setting to be **1** like so:

```
# set this to 0 to disable apport, or to 1 to enable it
# you can temporarily override this with
# sudo service apport start force_start=1
```

```
enabled=1

# set maximum core dump file size (default: 209715200 bytes == 200 MB)
maxsize=209715200
```

Once you have completed editing `/etc/default/apport`, start the apport service:

```
sudo start apport
```

After an application crashes, use the apport-cli command to search for the existing saved crash report information:

```
apport-cli

*** dash closed unexpectedly on 2010-03-11 at 21:40:59.

If you were not doing anything confidential (entering passwords or other
private information), you can help to improve the application by
reporting
the problem.

What would you like to do? Your options are:
  R: Report Problem...
  I: Cancel and ignore future crashes of this program version
  C: Cancel
Please choose (R/I/C):
```

Selecting *Report Problem* will walk you through similar steps as when using ubuntu-bug. One important difference is that a crash report will be marked as private when filed on Launchpad, meaning that it will be visible to only a limited set of bug triagers. These triagers will review the gathered data for private information before making the bug report publicly visible.

1.3. Risorse

- See the *Reporting Bugs*[3] Ubuntu wiki page.
- Also, the *Apport*[4] page has some useful information. Though some of it pertains to using a GUI.