

Drupal sintesi dell'ambiente

Ldfa 1 second ago

premessa

[Drupal](#) è un [CMS](#) con aspirazioni di [web framework](#).

Come tutti i CMS, Drupal crea le pagine richieste dal consultatore in modo dinamico. In altre parole, i contenuti informativi del sito non sono scritti in file html presenti nel file system del server web, ma in tabelle di una base dati, che vengono utilizzate per creare una pagina html *al volo*. In tal modo l'autore del sito può gestire i contenuti d'interesse tramite una interfaccia utente di supporto, senza fare ricorso ad una serie di attività *sistemistiche*.

Sviluppato in php, distribuito con licenza GNU GPL, ha una flessibilità di sviluppo, configurazione e utilizzo superiore ad altri CMS.

D'altro canto questa flessibilità si riflette in una maggior quantità di parametri da tenere sotto controllo. Per questo motivo quando si lavora con questo ambiente si tende ad utilizzare:

- sistemi configurati appositamente per ospitare siti sviluppati in Drupal;
- in alternativa, avere l'utilizzo della shell di sistema per effettuare le attività di configurazione e manutenzione necessarie.

Questa sintetica introduzione a Drupal si prefigge lo scopo di presentare i principali aspetti sistemistici relativi alla gestione di server con Drupal a bordo.

Drupal può essere installato in sistemi Linux, Windows e Mac OS X. Supporta web server Apache o IIS, e DB MySQL o PostgreSQL.

In questa nota si tratterà di sistemi Linux CentOS v.6.x, con Apache e MySQL

cosa si installa nel sistema

In sintesi servono:

- web server Apache;
- DB server MySQL;
- PHP;
- alcune utilità di sistema;
- Drush;
- Drupal.

[Drush](#) è un programma ausiliario, da shell, specializzato nella manutenzione di siti Drupal. Non è indispensabile, ma è decisamente comodo da utilizzare.

Tra l'altro, Drush è in grado di effettuare le attività necessarie ad allineare due siti installati in host diversi (si pensi a test e produzione). Per fare una cosa del genere deve essere in grado di effettuare login sul sistema remoto tramite protocollo OpenSSH utilizzando chiavi RSA. Per questo motivo di solito i sistemi che ospitano Drupal hanno OpenSSH abilitato e funzionante.

In pratica, una installazione da scratch, in CentOS, si può [esemplificare](#) come segue:

```
# install Apache, MySQL, php (& libraries), utils
yum install httpd
yum install mysql-server mysql
yum install php php-mysql php-mbstring php-xml php-gd php-pear
yum install wget unzip sudo bind-utils vixie-cron rsync xinetd

# get drush, install it, using drush download & install drupal and its (common) modules
wget http://ftp.drupal.org/files/projects/drush-7.x-5.9.tar.gz
tar xvf drush-7.x-5.9.tar.gz
drush dl drupal --drupal-project-rename=drupal
drush dl pathologic
drush dl pathauto
...
wget https://github.com/SU-SWS/open_framework/archive/7.x-2.1.tar.gz
tar xvf 7.x-2.1.tar.gz
```

La sequenza presentata serve per avere idea dei servizi, librerie, moduli che normalmente si installano in un sistema Drupal. Non è esaustiva, in quanto vi sono attività collaterali di configurazione. Ad esempio per Apache, è necessario modificare il file di configurazione e aprire nel firewall le

porte necessarie.

Riguardo Apache, una nota. Per gestire più domini in un host con un singolo IP, è d'uso ricorrere alla sua capacità di configurare `NameVirtualHost`.

A questo fine sono possibili diversi modi di *costruire* la configurazione predetta. Una di queste consiste nell'attivare la configurazione in `/etc/httpd/conf/httpd.conf`, e nell'aggiungere in coda a questo file una direttiva del tipo:

```
Include vhosts/*.conf
```

Con questa direttiva si forza Apache ad includere nella sua configurazione tutti i file presenti nella directory `/etc/httpd/vhost`, il cui nome termina con `.conf`.

A questo punto si è in grado di avere un file di configurazione per ogni nuovo host virtuale. Ad esempio possiamo avere i file di nomi `www.sito1.org.conf` e `www.sito2.org.conf`.

Se sarà necessario configurare un terzo sito basterà approntare il contenuto del file `www.sito3.org.conf` e riavviare Apache.

struttura del file system di drupal

Usualmente Drupal si installa nella directory:

```
/var/www/html/drupal
```

La directory predetta la indicheremo nel seguito come `[drupal_root]`. Come si vede, è omogenea con la gestione di altri siti nello stesso web server, in quanto è allocata nella sua directory dati: `/var/www/html`.

Da Drupal versione 7.x (la attuale di produzione), una singola installazione dell'ambiente, è in grado di gestire più siti.

Per fare ciò, l'installazione ha la seguente struttura di file system:

directory	scopo
<code>[drupal_root]</code>	codice php del core di drupal. Ad esclusione delle successive, le subdirectory presenti sono comunque parte del core
<code>[drupal_root]/sites</code>	directory dati di drupal
<code>[drupal_root]/sites/all</code>	quanto installato in questa directory è a disposizione di tutti i siti drupal
<code>[drupal_root]/sites/default</code>	file di configurazione di base da cui partire per un nuovo sito
<code>[drupal_root]/sites/sito1</code>	identificazione di <code>sito1</code> e suoi file specifici (modulii e temi, il contenuto è in base dati). Di fianco vi saranno gli altri siti: <code>sito2</code> <code>sito3</code> <i>Nota:</i> oltre la directory, ogni sito avrà associato un proprio DB.

Da questa struttura si comprende che se voglio installare un nuovo modulo drupal a disposizione di tutti i siti, lo dovrò allocare come sottodirectory di `drupal_root/sites/all/modules`.

Mentre, se voglio installare un nuovo modulo ad esclusivo uso di `sito1`, lo allocherò come subdirectory di `[drupal_root]/sites/sito1/modules`.

Di solito i nomi delle directory dei siti riflettono quello del dominio del sito. Ad esempio: `[drupal_root]/sites/www.nuovosito.org`.

La gestione di più siti con una sola installazione di core Drupal, evita di avere più copie del suo codice php.

In tal modo, quando è necessario effettuare un aggiornamento di minor version di Drupal (task relativamente frequente vista la continua attività di rilascio di update vuoi per bug fixing vuoi per protezione del codice in caso di scoperta di vulnerabilità), è possibile farlo scaricando il nuovo codice una sola volta.

Nota. Un aggiornamento non implica solo il caricamento del codice php aggiornato. Può essere necessario eseguire delle attività di riallineamento della struttura della base dati. Se funziona tutto senza problemi, questa attività viene effettuata interattivamente da web browser. Se sorgono problemi potrebbe essere necessario mettere mano ai comandi mysql da root.

creare un sito

Per creare un sito si eseguono i seguenti passi:

- si crea il DB per il sito;
- si crea un utente e gli si danno i diritti completi di gestione del DB per il sito;
- si crea la configurazione dell'host virtuale per Apache;

- si crea la directory per il sito;
- si copia nella directory del sito il file `[drupal_root]/settings.php`
- da web browser si accede al sito e si procede rispondendo alle domande del sistema per configurare il sito stesso.

Esemplificando:

```
# gestione del DB
mysqladmin -u root -proot_password create site_db
mysql -u root -ppassword -e GRANT SELECT, INSERT, UPDATE, DELETE, CREATE, DROP, INDEX, ALTER, LOCK
TABLES, CREATE TEMPORARY TABLES ON site_db.* TO 'site_db_user'@'localhost' IDENTIFIED BY
'site_db_password';

# gestione di Apache
cp www.nuovosito.org.conf /etc/httpd/vhosts/www.nuovosito.org.conf

# gestione di drupal
mkdir /var/www/html/drupal/sites/www.nuovosito.org
cp /var/www/html/drupal/sites/default/default.settings.php /var/www/html/drupal/sites/www.nuovosito.org
/settings.php

# usando un web browser, accedere al sito: http://www.nuovosito.org e rispondere alle domande
```

I comandi indicati sono una semplificazione in quanto, per sicurezza, sarà necessario gestire opportunamente (prima concedendoli e successivamente revocandoli) i diritti di scrittura sui file d'installazione del sito,

Il file `www.nuovosito.org.conf` ha contenuti simili ai seguenti:

```
<VirtualHost 192.168.0.11:80>
  DocumentRoot "/var/www/html/drupal"
  ServerName www.nuovosito.org
  CustomLog "logs/www.nuovosito.org.access.log" combined
  ErrorLog "logs/www.nuovosito.org.error.log"
  <Directory "/var/www/html/drupal">
    Options Indexes FollowSymLinks
    AllowOverride All
    Order allow,deny
    Allow from all
  </Directory>
</VirtualHost>
```

e viene caricato in coda dal file `/etc/httpd/conf/httpd.conf` con una direttiva del tipo: `Include vhosts/*.conf`

gestione del nome di dominio

Quando si sviluppa un sito, è necessario avere:

- un sistema di sviluppo (ad es. `dev.nuovosito.org`);
- uno di test (ad es. `staging.nuovosito.org`);
- il sito di produzione (ad es. `www.nuovosito.org`).

Ognuno di questi sistemi avrà una propria URL da cui essere raggiunto. E questo può essere un problema per i siti dinamici, perché eventuali riferimenti ad URL assolute del sito non funzionano più quando si trasporta il contenuto da un sito ad un altro.

Per evitare quanto detto si lavora utilizzando URL relative. Ma questo non è sempre possibile. In particolare è un problema la gestione delle URL di file caricati nel sito (immagini, pdf, ...). Questi file non entrano nel DB; rimangono nel file system del sito. Di conseguenza la loro URL include il nome della directory del sito.

Se ne deduce che il nome della directory del sito deve essere la stessa per tutti questi sistemi. Di solito si usa il nome del sito di produzione.

Dopo di che ogni sistema utilizzerà un link simbolico al nome del sito di produzione. Ad esempio, nel sito di sviluppo:

```
cd /var/www/html/drupal/sites
ln -s www.nuovosito.org dev.nuovosito.org
```

In tal modo riferimenti interni a `dev.nuovosito.org` sono dirottati in modo trasparente su `www.nuovosito.org`.

gli aggiornamenti

da scrivere

del contenuto

da scrivere

di moduli e temi

da scrivere

del core

da scrivere