

Mengenal Apache Web Server

Apache merupakan aplikasi web server yang paling populer digunakan di internet, untuk memudahkan kita menginstall paket httpd maka akan kita install menggunakan yum, adapun caranya adalah sebagai berikut

Instalasi httpd

```
yum install php-mysql php-cli php-common php httpd mysql-server
```

Dengan menjalankan perintah diatas kita baru saja menginstall paket httpd beserta paket-paket pendukungnya (php,mysql), adapun file konfigurasi httpd pada CentOS Linux terdapat di /etc/httpd/httpd.conf, dengan instalasi standar diatas apache dapat langsung kita jalankan dengan menggunakan konfigurasi default, dan untuk menjalankan service httpd pada CentOS dapat kita lakukan dengan menjalankan perintah berikut

```
/etc/init.d/httpd start
```

untuk memmatikanya

```
/etc/init.d/httpd stop
```

atau kita juga dapat menggunakan perintah berikut

```
/sbin/service httpd start
```

atau untuk menghentikan

```
/sbin/service httpd stop
```

apabila tidak terdapat pesan kesalahan (error) berarti service httpd telah berjalan, untuk mengecek proses httpd pada sistem kita dapat menjalankan perintah berikut

```
ps ax | grep httpd
```

dengan hasil output berikut

```
ps ax | grep httpd
 3929 ?        Ss      0:00 /usr/sbin/httpd
 3930 ?        S       0:00 /usr/sbin/httpd
 3931 ?        S       0:00 /usr/sbin/httpd
 3932 ?        S       0:00 /usr/sbin/httpd
 3933 ?        S       0:00 /usr/sbin/httpd
 3934 ?        S       0:00 /usr/sbin/httpd
 3936 ?        S       0:00 /usr/sbin/httpd
 3937 ?        S       0:00 /usr/sbin/httpd
 3938 ?        S       0:00 /usr/sbin/httpd
 3944 pts/0    S+      0:00 grep httpd
```

kemudian kita juga dapat mengecek apakah httpd telah berjalan (listen) dengan perintah berikut

```
netstat -pln | grep httpd
```

jika terdapat tampilan berikut berarti proses httpd telah berjalan dengan benar dan sudah siap digunakan

```
tcp          0          0 :::80                :::* LISTEN
3929/httpd
```

dengan output di atas berarti httpd telah berjalan dan telah siap untuk digunakan

Konfigurasi httpd

File konfigurasi httpd secara default tersimpan di `/etc/httpd/conf/httpd.conf`, berikut adalah beberapa parameter penting dalam konfigurasi apache :

ServerRoot "/etc/httpd" : Merupakan direktori utama dari apache, disini akan diletakan file-file konfigurasi dan juga log dari apache

Listen 80 : Merupakan port yang akan digunakan oleh httpd untuk berkomunikasi dengan client

DocumentRoot "/var/www/html" : Tempat dokumen web kita disimpan

Secara default konfigurasi httpd seharusnya sudah dapat langsung digunakan, sehingga kita hanya perlu menambahkan opsi-opsi yang dirasa perlu kita tambahkan.

Langkah selanjutnya adalah uji coba, untuk mencobanya kita dapat membuat file php yang akan menginformasikan web server yng kita gunakan, untuk memudahkan berikan nama filenya dengan `info.php` yang isinya script php berikut

```
<?
phpinfo ()
?>
```

kemudian kita simpan pada `/var/www/html`, untuk melakukan uji coba maka buka browser dan arahkan ke alamat http server kita, misalkan `http://10.55.1.50`, atau apabila webserver yang kita konfigurasi sudah mempunyai domain maka kita bisa mengaksesnya dengan mengarahkan ke alaman URL `http://server.domain/info.php`, sebagai contoh disini webserver sudah mempunyai domain `ugos.com`, maka kita dapat mengaksesnya dengan `http://ugos.com/info.php` maka kita akan mendapati tampilan berikut



System	Linux centos 2.6.18-128.el5 #1 SMP Wed Jan 21 10:44:23 EST 2009 i686
Build Date	Apr 7 2009 08:01:33
Configure Command	'./configure' '--build=i686-redhat-linux-gnu' '--host=i686-redhat-linux-gnu' '--target=i386-redhat-linux-gnu' '--program-prefix=' '--prefix=/usr' '--exec-prefix=/usr' '--bindir=/usr/bin' '--sbindir=/usr/sbin' '--sysconfdir=/etc' '--datadir=/usr/share' '--includedir=/usr/include' '--libdir=/usr/lib' '--libexecdir=/usr/libexec' '--localstatedir=/var' '--sharedstatedir=/usr/com' '--mandir=/usr/share/man' '--infodir=/usr/share/info' '--cache-file=../config.cache' '--with-libdir=lib' '--with-config-file-path=/etc' '--with-config-file-scan-dir=/etc/php.d' '--disable-debug' '--with-pic' '--disable-rpath' '--without-pear' '--with-bz2' '--with-curl' '--with-exec-dir=/usr/bin' '--with-freetype-dir=/usr' '--with-png-dir=/usr' '--enable-gd-native-ttf' '--without-gdbm' '--with-gettext' '--with-gmp' '--with-iconv' '--with-jpeg-dir=/usr' '--with-openssl' '--with-png' '--with-pspell' '--with-ldap-dir=/usr' '--with-pcre-regex=/usr' '--with-zlib' '--with-layout=GNU' '--enable-exif' '--enable-ftp' '--enable-magic-quotes' '--enable-sockets' '--enable-syssem' '--enable-sysvshm' '--enable-sysvmsg' '--enable-track-vars' '--enable-trans-sid' '--enable-yp' '--enable-wddx' '--with-kerberos' '--enable-ucd-snmp-hack' '--with-unixODBC=shared,/usr' '--enable-memory-limit' '--enable-shmop' '--enable-calendar' '--enable-dbx' '--enable-dio' '--with-mime-magic=/usr/share/file/magic.mime' '--without-sqlite' '--with-libxml-dir=/usr' '--with-xml' '--with-system-tzdata' '--with-apxs2=/usr/sbin/apxs' '--without-mysql' '--without-gd' '--without-odbc' '--disable-dom' '--disable-dba' '--without-unixODBC' '--disable-pdo' '--disable-xmlreader' '--disable-xmlwriter'
Server API	Apache 2.0 Handler
Virtual Directory Support	disabled
Configuration File (php.ini)	/etc/php.ini

Membuat Alias

Alias dapat di gambarkan sebagai nama lain dari nama direktori sebenarnya, sebagai contoh apabila kita mempunyai webserver dengan opsi konfigurasi *DocumentRoot* `"/var/www/html"` ketika kita mengakses alamat 127.0.0.1 dengan browser maka yang ditampilkan adalah isi dari direktori `/var/www/html`, permasalahan akan timbul ketika kita menginstall aplikasi lain yang letaknya tidak berada dibawah direktori `/var/www/html`, misalkan saja kita menginstall phpMyAdmin dan direktori hasil instalasinya berada di `/usr/share/phpmyadmin`, maka untuk mengakses direktori phpmyadmin kita membutuhkan alias untuk mengaksesnya

untuk membuat alias edit konfigurasi httpd yang berada di `/etc/httpd/conf/httpd.conf` dan tambahkan baris opsi untuk alias, berikut adalah contoh pembuatan alias.

```
Alias /phpadmin "/usr/share/phpMyAdmin-2.11.5.2-all-languages/"
```

```
<Directory "/usr/share/phpMyAdmin-2.11.5.2-all-languages">
    Options Indexes MultiViews
    AllowOverride None
    Order allow,deny
    Allow from all
</Directory>
```

Untuk mengaksesnya, kita gunakan browser dan arahkan ke alamat `http://server/phpadmin`, misalkan alamat server kita mempunyai domain `ugos.com` maka alamatnya akan menjadi `http://ugos.com/phpadmin`

Konfigurasi Virtual Hosts

Virtual hosts merupakan salah satu fitur dari apache webserver sehingga seolah-olah server apache berada pada banyak mesin, misalkan kita mengakses `http://ugos.com` dan `http://testing.ugos.com` yang keduanya berada pada mesin yang sama, meskipun berada pada mesin yang sama kedua alamat situs tersebut akan memberikan hasil output yang berbeda, berikut adalah contoh konfigurasi virtual hosts pada apache.

```
<VirtualHost testing.ugos.com:80>
    ServerAdmin webmaster@ugos.com
    DocumentRoot /var/www/vdomain/testing
    ServerName testing.ugos.com
    ErrorLog logs/testing.ugos.com-error_log
    CustomLog logs/testing.ugos.com-access_log common

    <Directory "/var/www/vdomain/testing">
        Options Indexes MultiViews
        AllowOverride None
        Order allow,deny
        Allow from all
    </Directory>
</VirtualHost>
```

Konfigurasi ssl di httpd

Secure Socket Layer (SSL) merupakan metode koneksi terekripsi sehingga relatif lebih aman, berikut adalah cara yang dapat kita lakukan untuk mengaktifkan koneksi https di erb server.

Langkah pertama adalah kita membutuhkan modul `ssl` dan juga `openssl`, apabila kedua aplikasi tersebut belum terinstall maka kita dapat menginstallnya menggunakan tool `yum`

```
yum install mod_ssl openssl
```

Selanjutnya adalah meng-generate sertifikasi ssl yang nantinya akan kita pasang di webserver

```
openssl genrsa -out ca.key 1024
```

```
# Generate CSR
```

```
openssl req -new -key ca.key -out ca.csr
```

```
# Generate Self Signed Key
```

```
openssl x509 -req -days 365 -in ca.csr -signkey ca.key -out ca.crt
```

Setelah proses generate kunci (key) ssl maka selanjutnya adalah mengkopikan key ssl tersebut ke direktori /etc/pki/tls

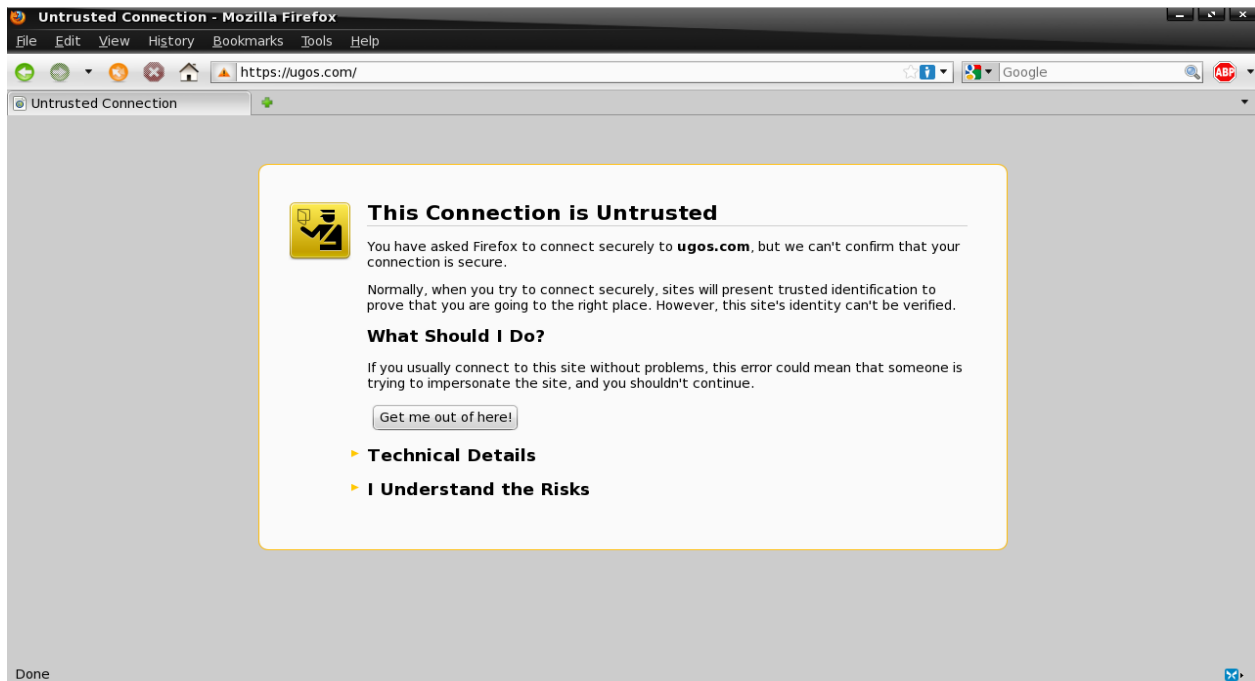
```
mv ca.crt /etc/pki/tls/certs
mv ca.key /etc/pki/tls/private/ca.key
mv ca.csr /etc/pki/tls/private/ca.csr
```

Selanjutnya cek konfigurasi ssl di /etc/httpd/conf.d/ssl.conf, dan pastikan terdapat baris opsi konfigurasi berikut pada file konfigurasi httpd

```
LoadModule ssl_module modules/mod_ssl.so
SSLEngine on
SSLCertificateFile /etc/pki/tls/certs/ca.crt
SSLCertificateKeyFile /etc/pki/tls/private/ca.key
```

Setelah opsi konfigurasi sudah benar, langkah selanjutnya adalah merestart service httpd

```
/etc/init.d/httpd restart
/sbin/service httpd restart
```



Autentikasi httpd Menggunakan .htaccess

Kadang kala kita mungkin sering membutuhkan perlindungan lebih terhadap web kita, sebagai contoh ketika kita mempunyai data-data yang penting dan hanya orang-orang yang berkepentingan yang boleh mengaksesnya, untuk itu kita dapat membuat otentikasi untuk membatasi agar tidak semua

orang dapat mengakses alamat web kita, pertama kita tentukan dulu alamat ana yang akan kita batasi aksesnya, sebagai contoh kita akan membuat alias yang hanya dapat diakses oleh user dengan autentikasi, pertama edit `/etc/httpd/conf/httpd.conf` dan tambahkan opsi berikut

```
Alias /admin "/var/www/admin"
```

```
<Directory "/var/www/admin">  
    Options Indexes MultiViews  
    AllowOverride AuthConfig  
    Order deny,allow  
    Allow from all  
</Directory>
```

setelah kita tambahkan baris diatas, selanjutnya kita buat file `.htaccess` yang kita letakan di dalam direktori yang akan kita batasi aksesnya, karena pada contoh kita akan membatasi akses ke direktori `/var/www/admin` maka kita buat file `.htaccess` pada direktori `/var/www/admin` yang isinya sebagai berikut

```
AuthType Basic  
AuthUserFile /var/www/.htpasswd  
AuthName Admin-area  
require valid-user
```

selanjutnya kita buat user yang boleh yang akan digunakan untuk melakukan otentikasi dengan menggunakan perintah `htpasswd` seperti berikut

```
htpasswd -c /var/www/.htpasswd rifqi  
New password:  
Re-type new password:  
Adding password for user rifqi
```

pada contoh ini nama pengguna yang digunakan untuk login adalah `rifqi`, kemudian masukan password untuk username `rifqi`, setelah konfigurasi selesai langkah terakhir adalah restart service `httpd`

```
/etc/inint.d/httpd restart
```

kemudian kita coba mengakses alamat yang barusaja kita batasi dengan autentikasi degan mengakses alamat `http://nama_server/admin` apabila tidak terdapat kesalahan maka seharusnya kita akan ditanyai autentikasi username dan password seperti gambar berikut

