Linux Apache SSL PHP/FI frontpage mini-HOWTO

Marcus Faure, *marcus@faure.de* v1.1, Luglio 1998

Questo documento tratta la costruzione di un **server web multiestensione** che supporti contenuti web dinamici tramite il linguaggio di script **PHP/FI**, la trasmissione di dati sicuri basata sulle **SSL** di Netscape, l'esecuzione sicura dei **CGI** e le **estensioni per un server Frontpage** Microsoft. Traduzione di Marina Sturino (mcrastell@mbox.queen.it).

Indice

1 Introduzione

Prima che iniziate a leggere: non sono madrelingua, cosicché in questo documento ci saranno probabilmente errori di ortografia o grammaticali. Sentitevi incoraggiati a informarmi degli errori.

1.1 Descrizione dei componenti

Il server web che voi sperate di avere dopo aver letto questo HOWTO è composto da più parti, il sorgente originale Apache con alcune (o meglio molte) patch ed alcuni eseguibili esterni. Raccomando di usare le versioni del software che ho provato, le quali probabilmente si compileranno senza grandi problemi e dando come risultato un demone abbastanza stabile. Se siete coraggiosi potete cercare di compilare tutte le ultime novità proposte, ma non biasimatemi se qualcosa va male ;-) Tuttavia potete riferirmi altre configurazioni di lavoro per includerle nelle versioni future di questo documento. Tutti questi passi sono stati testati su una versione Linux 2.0.35, cosicché questo HOWTO è piuttosto specifico per Linux, ma potrete usarlo altrettanto bene per altri sistemi Unix.

Non è necessario che compiliate tutti gli elementi. Ho cercato di strutturare questo HOWTO in modo che possiate saltare le parti che non vi interessano.

Questo documento non è un manuale dell'utente né per Apache, né per SSL o PHP/FI né tantomeno per Frontpage. Il suo scopo primario è di salvaguardare i fornitori di servizi web da molti mal di testa quando installano i loro server e di dare il mio piccolo contributo alla comunità Linux.

PHP è un linguaggio di scripting che supporta pagine in HTML dinamico. Esso è strutturato come le SSI di Apache ma di gran lunga più complesso ed ha dei moduli database per molti db popolari. Per il PHP sono necessarie le librerie GD.

SSL è un'implementazione del Secure Socket Layer di Netscape che permette connessioni sicure su reti insicure, per esempio per trasmettere numeri di carte di credito tramite form sul web.

Frontpage è uno strumento WYSIWYG per fare pagine web che usa alcune specifiche estensioni per il server chiamate webbot. Molte persone pensano che Frontpage sia "cool" perché permette di creare form e web di discussione senza avere conoscenze sull'HTML e su CGI. Tramite uno strumento di pubblicazione guidata sul web, evita all'autore di dover aggiornare il suo sito via ftp. Se volete supportare Frontpage ma non volete installare un server windows, le estensioni per il server Apache sono la vostra soluzione.

1.2 Configurazioni di lavoro

Penso che questo documento sia stato scaricato più di 100 volte da quando l'ho pubblicato ma ho ricevuto solo piccoli commenti. In particolare, nessuno mi ha indicato altre configurazioni di lavoro. Per me le configurazioni di lavoro sono:

- Linux 2.0.31, Apache 1.2.4, PHP 2.0.0, SSL 0.8.0, fp 98 3.0.3 (*)
- Linux 2.0.33, Apache 1.2.5, PHP 2.0.1, SSL 0.8.0, fp 98 3.0.3 (*)
- Linux 2.0.35, Apache 1.2.6, PHP 3, SSL 0.8.0, fp 98 3.0.4

(*) la versione 3.03 3.6 (non è consigliata).

1.3 Storia

v0.0 Aprile '98: Anteprima

v1.0 Giugno '98: Ora usando Apache 1.2.6, aggiornata la sezione frontpage, correzioni minori

v1.1 Giugno '98: Versione ristrutturata in SGML

Potete trovare l'ultima versione di questo documento sul sito: `http://www.faure.de`

2 Installazione dei componenti

2.1 Preparazione

Voi avete bisogno di:

- Apache 1.2.6 `http://www.apache.org/dist/apache_1_2_6.tar.gz`
- Estensioni PHP/FI
  `http://php.iquest.net/files/download.phtml?/files/php-2.01.tar.gz`
- Libreria GD `http://siva.cshl.org/gd/gd.html`
- SSL 0.8.0
  `ftp://ftp.ox.ac.uk/pub/crypto/SSL/SSLeay-0.8.0.tar.gz`
- Patch SSL per Apache 1.2.6
  `ftp://ftp.ox.ac.uk/pub/crypto/SSL/apache_1.2.6+ssl_1.17.tar.gz`
- Estensoni server per frontpage 98 e script di installazione
  `http://www.rtr.com/fpsupport/download.htm`

Potete prelevare i sorgenti che volete. Decomprimete Apache, PHP, GD, e SSL in `/usr/src`. Decomprimete la patch SSL in `/usr/src/apache_1.2.6`.

2.2 Aggiungere PHP

Fate `cd` in `/usr/src/gd1.2` e scrivete `make`. Ciò costruirà la libreria GD `libgd.a`che dovrà essere copiata in `/usr/lib`. Ora fate `cd` in `php_2.0.1` e avviate `./install`. Le domande importanti sono:

```
Would you like to compile PHP/FI as an Apache module? [yN] y
Are you compiling for an Apache 1.1 or later server? [Yn] y
Are you using Apache-Stronghold? [yN] y
Does your Apache server support ELF dynamic loading? [yN] y
Apache include directory (which has httpd.h)? [/usr/local/include/apache] /usr/src/apache_1.2.6/src
Would you like to build an ELF shared library? [yN] y
Additional directories to search for .h files []: /usr/src/gd1.2
Would you like the bundled regex library? [yN] n
```

Come le estensioni frontpage, phtml introduce un problema di sicurezza perché gira con lo uid del server web. Assicuratevi di attivare il modo sicuro (safe mode) in `src/php.h` e restringete il percorso di ricerca a un valore di sicurezza. Ci sono alcune altre opzioni in `php.h` che potreste voler modificare. Se siete molto preoccupati riguardo il problema della sicurezza, compilate php come un CGI. Tuttavia ciò causerà una diminuzione delle prestazioni e non sarà così forte come la versione a moduli.

Digitate `make` per compilare il file. Quando la compilazione è terminata, copiare `mod_php.*` e `lib_php.a` in `/usr/src/apache_1.2.6/src`. Aggiungete la riga:

```
Module php_module mod_php.o
```

alla fine di `/usr/src/apache_1.2.6/src/Configuration`, aggiungete

```
-lphp -lm -lgdbm -lgd
```

alla `EXTRA_LIBS` nello stesso file,

```
application/x-httpd-php phtml
```

nell'`Apache mime.types` e

```
AddType  application/x-httpd-php .phtml
```

nell'Apache `srm.conf`.

Potete anche voler aggiungere `index.phtml` alla `DirectoryIndex` in questo file, cosicché il file `index.phtml` sia caricato automaticamente quando si richiama la sua directory.

2.3 Aggiungere SSL

Fate `cd /usr/src/SSL-0.8.0; ./Configure linux-elf; make; make rehash`. Questo creerà le librerie richieste da Apache. Potete digitare `make test` per verificare la compilazione. Ora dovete applicare la patch ad Apache. È importante che la applichiate prima della patch per frontpage, altrimenti frontpage non funzionerà. Fate poi `cd` in `/usr/src/apache_1.2.6/src` e digitate `patch < /usr/src/apache_1.2.6/SSLpatch`. Impostate `SSL_BASE=/usr/src/SSLeay-0.8.0` in `Configuration`. Assicuratevi che `Module proxy_module` sia disabilitato altrimenti Apache non compilerà. Se avete la necessità di avere un proxy, optate per Squid `squid.nlanr.net`

Ora date `make certificate` per generare `SSLconf/conf/httpsd.pem`.

2.4 Aggiungere frontpage

Rinominate il file `fp30.linux.tar.Z` come `fp30.linux.tar.gz` altrimenti lo script di installazione non lo troverà. Fate partire `./fp_install` per copiare i file delle estensioni in `/usr/local/frontpage`. `zcat` è solitamente invocato come `/usr/bin/zcat` Ora potete applicare la patch per frontpage. Fate `cd` in `/usr/src/apache_3.2.6/src` e digitate `patch < /usr/src/frontpage/version3.0/apache-fp/fp-patch-apache_1.2.5`. Questo creerà i file `mod_frontpage.*`, apporterà delle modifiche in `Configuration`, ecc. La patch 1.2.5 funzionerà sia con Apache 1.2.5 che 1.2.6. Saltate la parte riguardo l'installazione del web, potrete farlo successivamente.

3 Mettere tutto assieme

3.1 Moduli Apache da cercare

I moduli che io uso in aggiunta a SSL, PHP e frontpage sono:

```
Module env_module          mod_env.o
Module config_log_module   mod_log_config.o
Module mime_module         mod_mime.o
Module negotiation_module  mod_negotiation.o
Module dir_module          mod_dir.o
Module cgi_module          mod_cgi.o
Module asis_module         mod_asis.o
Module imap_module         mod_imap.o
Module action_module       mod_actions.o
```

```
Module alias_module        mod_alias.o
Module rewrite_module      mod_rewrite.o
Module access_module       mod_access.o
Module auth_module         mod_auth.o
Module anon_auth_module    mod_auth_anon.o
Module digest_module       mod_digest.o
Module expires_module      mod_expires.o
Module headers_module      mod_headers.o
Module browser_module      mod_browser.o
```

3.2 Rendere più sicuri i CGI

Se siete un ISP (internet service provider) (probabilmente lo siete se leggete questo) vorrete migliorare la sicurezza. Le utility suexec vi permetteranno di farlo; eseguiranno i CGI sotto lo UID del webowner invece di eseguirli con l'UID del server web. Andate in `/usr/src/apache_1.2.6/support` e date `make suexec`. Poi date `chmod 4711 suexec` e copiatelo nella locazione specificata in `../src/httpd.h` che solitamente è `/usr/local/etc/httpd/sbin/suexec`. Se questo percorso vi sembra un po' criptico - come è parso a me - editate `httpd.h` e impostate il percorso a un valore per voi più comodo.

3.3 Compilare ed installare il demone del server

Entrate in `/usr/src/apache_1.2.6/src` ed editate `Configuration` per impostare tutti i moduli che volete includere nel vostro demone apache. Quando l'avete fatto, avviate `./Configure` e `make.` Questo è l'ultimo (ed il più complicato) passo per la configurazione che vi passerà per le mani. Se ha successo, copiate `httpsd` in `/usr/sbin.` Il demone è piuttosto grande, tenetelo in considerazione quando assemblate il vostro server web. Create la directory `/var/httpd` con le sottodirectory `cgi-bin`, `conf`, `htdocs`, `icons`, `virt1`, `virt2` e `logs`. In `/usr/src/apache_1.2.6/conf` editate `access.conf-dist`, `mime.types` e `srm.conf-dist` per adattarli alle vostre necessità e copiateli in `var/httpd/conf/access.conf`. Copiate il file `httpsd.pem` che avete creato con `make certificate` in `/var/httpd/conf`. Usate il seguente `httpd.conf`:

```
ServerType standalone
Port 80
Listen 80
Listen 443
User wwwrun
Group wwwrun
ServerAdmin webmaster@yourhost.com
ServerRoot /var/httpd
ErrorLog logs/error_log
TransferLog logs/access_log
PidFile logs/httpd.pid
ServerName www.yourhost.com
MinSpareServers 3
MaxSpareServers 20
StartServers 3

SSLCACertificatePath /var/httpd/conf
SSLCACertificateFile /var/httpd/conf/httpsd.pem
SSLCertificateFile /var/httpd/conf/httpsd.pem
SSLLogFile /var/httpd/logs/ssl.log
```

```
<VirtualHost www.virt1.com>
SSLDisable
ServerAdmin webmaster@virt1.com
DocumentRoot /var/httpd/virt1
ScriptAlias /cgi-bin/ /var/httpd/virt1/cgi-bin/
ServerName www.virt1.com
ErrorLog logs/virt1-error.log
TransferLog logs/virt1-access.log
User virt1admin
Group users
</VirtualHost>

<VirtualHost www.virt1.com:443>
ServerAdmin webmaster@virt1.com
DocumentRoot /var/httpd/virt1
ScriptAlias /cgi-bin/ /var/httpd/virt1/cgi-bin/
ServerName www.virt1.com
ErrorLog logs/virt1-ssl-error.log
TransferLog logs/virt1-ssl-access.log
User virt1admin
Group users
SSLCACertificatePath /var/httpd/conf
SSLCACertificateFile /var/httpd/conf/httpsd.pem
SSLCertificateFile /var/httpd/conf/httpsd.pem
SSLLogFile /var/httpd/logs/virt1-ssl.log
SSLVerifyClient 0
SSLFakeBasicAuth
</VirtualHost>

<VirtualHost www.virt2.com>
SSLDisable
ServerAdmin webmaster@virt2.com
DocumentRoot /var/httpd/virt2
ScriptAlias /cgi-bin/ /var/httpd/virt2/cgi-bin/
ServerName www.virt2.com
ErrorLog logs/virt2-error.log
TransferLog logs/virt2-access.log
</VirtualHost>
```

A seconda dei moduli compilati in esso, non tutte le direttive sono disponibili. Potete ottenere una lista delle direttive disponibili digitando `httpsd -h`.

3.4 Aggiungere al web il supporto frontpage

Entrate in `/usr/local/frontpage/version3.0/bin` e caricate `./fpsrvadm`. Scegliete `install` e `apache-fp`. Rispondete alle domande che seguono nel seguente modo:

```
Enter server config filename: /var/httpd/conf/httpd.conf
Enter host name for multi-hosting []: www.virt2.com
```

```
Starting install, port: www.virt2.com:80, web: ""
Enter user's name []: virt2admin
Enter user's password:
Confirm password:
Creating root web
Recalculate links for root web
Install completed.
```

Lo username deve essere quello di login unix del webowner, la password non deve necessariamente corrispondere alla password del sistema. Dovete aggiungere manualmente `sendmailcommand:/usr/sbin/sendmail %r` in `/usr/local/frontpage/www.virt2.com:80.conf`, altrimenti i vostri utenti non saranno in grado di inviare email generate dal web. Ora date `kill -HUP` al vostro httpsd per far rileggere a fp la sua configurazione. Ora potete accedere a www.virt2.com con il vostro client frontpage.

In alcune circostanze `fpsrvadm` protesta se non è stata installata prima una directory root per il web. Ciò è praticamente inutile, ma dovrete farlo così da far tacere fpsrvadm.

3.5 Avviare il demone

Avviare apache con `httpsd -f /var/httpd/conf/httpd.conf.` Ora potete accedere a www.virt1.com sia attraverso http che https e ciò è veramente "cool". Naturalmente dovete pagare per ottenere un certificato reale se volete offrire servizi SSL via web o gli utenti potrebbero deridervi.

Copiate uno dei file demo della directory degli esempi php in `virt1` per testare phtml.

3.6 Alcune altre considerazioni

Non usate le estensioni di frontpage '97. Non girano sotto Linux. Quando si installano le versioni specifiche delle librerie C++, sembrano funzionare correttamente, ma i vostri logs saranno presto riempiti di `premature end of script headers` e la vostra casella postale sarà si riempirà di proteste. Non usate le estensioni frontpage '98 antecedenti la versione 3.0.2.1330. Non confondetevi, i numeri di versione sono a volte non eterogenei. Facendo un telnet alla porta 80, scrivendo get/http/1.0 (e premendo due volte invio), otterrete la versione numero 3.0.4 per frontpage.

Potete trovare un numero di versione più specifico eseguendo `/usr/local/frontpage/currentversion/exes/_vti_bin/shtml.exe -version.` Versioni più vecchie hanno dei bug cattivi che richiedono che `httpd.conf` sia scrivibile dal GID del server web. Ciò potrebbe farvi strillare se siete particolarmente preoccupati per la sicurezza. Le versioni a partire dalla 3.0.2.1330 sono più usabili.

3.7 Bug conosciuti

Quando si utilizza Recalculate Links nel client frontpage, il server inizializza un processo che consuma il 99% dei cicli della CPU e più di 10 Mb di memoria. Ma anche per i web di medie dimensioni e per macchine veloci, i client spesso ricevono un messaggio di timeout prima che l'elaborazione sia finita correttamente. Informate gli utilizzatori di frontpage di essere pazienti e di non utilizzare Recalculate Links molte volte. Equipaggiate il server con almeno 64 Mb di memoria.

A questo punto notate che funzionano sia le SSL sia frontpage, ma non contemporaneamente, questo significa che non potete né pubblicare il vostro web usando SSL, né, far uso dei webbots tramite https. Potete pubblicare il vostro web sulla porta 80 ed accedere ad esso crittati sulla porta 443, ma i vostri contatori, ecc si interromperanno. Io considero questo un bug. Il problema verrà corretto in SSL 0.9.0.

3.8 Considerazioni finali

Per coloro che pensano che il titolo di questo howto sia quasi lungo quanto il documento: avete mai sentito Meat Loaf?

O.K. lettori, per oggi avete finito. Sentitevi liberi di inviarmi le vostre critiche, la vostra gratitudine eterna, fiori, moneta elettronica, automobili, profumi, ecc.