Kako instalirati Apache-SSL server?

Dobrica Pavlinušić, *dpavlin@rot13.org* verzija 1.0, 24. srpnja 2000.

Kako do sigurnih `https` veza na nesigurnom Internetu.

Sadržaj

- 1998-07-05: Dobrica Pavlinušić <dpavlin@rot13.org> v0.0
- 1998-07-10: dpavlin v0.1 (1st public)
- 1998-07-29: dpavlin v0.2 (apache-1.3.0 podrška, ispravljene greške)
- 1998-08-26: dpavlin v0.21 dodane reference na mod_ssl
- 1998-09-11: dpavlin v0.3 vlastiti certifikati za duže od 30 dana

- 1998-11-19: dpavlin v0.4 iso8859-2, mod_ssl instalacija
- 1999-07-17: Matej Vela <mvela@public.srce.hr> v0.41 pretvoren u Linuxdoc
- 1999-07-23: dpavlin v0.5 mnoge promjene, ostavljen samo mod_ssl, kako dodati dodatne module (mod_rewrite, mod_auth_mysql) kod kompajliranja
- 1999-09-21: dpavlin v0.6 čudne greške kod kompajliranja Apache-a napokon objašnjene
- 2000-07-24: dpavlin v1.0 promjena e-maila, kako natjerati IE da prihvati napravljeni certifikat, update

1 Gdje se nalazi zadnja verzija ovoga dokumenta?

Originalna lokacija ovoga dokumenta je promjenjena u `<http://dokumentacija.linux.hr/Apache-SSL.html>`. Ovaj dokument je dio ldpHR projekta. Ostali formati nalaze se na adresi `<http://dokumentacija.linux.hr/>`.

2 Predgovor

Apache SSL je bio preko tri godine jedini način dobivanja secure `http` servera (`https`). Međutim, pojavio se mod_ssl koji se odlikuje čistijim kodom i **mnogo** boljom dokumentacijom (koja ovaj mini-KAKO čini skoro nepotrebnim :-). Prva verzija ovoga dokumenta nastala iz potrebe da se objasne i približe korisnicima tajne Apache-SSL patcheva koji su imali svoje probleme i bubice. mod_ssl je, sa druge strane, jednostavan za instaliranje, radi veoma dobro, i bez problema surađuje sa ostalim modulima. Zbog toga ovaj dokumet od verzije 0.5 više ne spominje originalne Apache-SSL patch-eve.

3 Gdje naći potrebne datoteke?

3.1 mod_ssl

Otići na `<http://www.modssl.org/>` i skinuti zadnju verziju (ovdje će biti riječi o verziji mod_ssl-2.3.6-1.3.6)

3.2 OpenSSL biblioteka

Prijašnje verzije SSL dodataka za Apache su se oslanjale na SSLeay biblioteku, dok noviji mod_ssl koristi OpenSSL biblioteku (ovdje je riječ o verziji openssl-0.9.3a).

Možete je naći na `<http://www.openssl.org/>` ili u hrvatskoj na `<ftp://ftp.linux.hr/pub/openssl/source/>`.

3.3 Apache

Naravno, potreban Vam je i sam Apache web server. Njega možete naći na `<http://www.apache.org/dist/>`. Pazite na to da skinete odgovarajuću verziju (tj. onu za koju je predviđen modul).

4 Kompajliranje

4.1 OpenSSL

Otpakirajte arhivu OpenSSL-a, uđite u njezin direktorij i pokrenite kompajrianje korištenjem:

```
> tar xvfz openssl-0.9.3a.tar.gz
> cd openssl-0.9.3a
> sh config no-idea -fPIC
> make
```

Nakon dosta vremena (skoro najdužeg), kompajliranje bi trebalo završiti.

4.2 Apache i mod_ssl

Otpakirajte arhivu Apache-a korištenjem:

```
> tar xvfz apache_1.3.6.tar.gz
```

Samo prevođenje i kompiliranje programa će uslijediti nakon što konfigurirate mod_ssl.

```
> tar xvfz mod_ssl-2.3.6-1.3.6.tar.gz
> cd mod_ssl-2.3.6-1.3.6
> ./configure --with-apache=../apache_1.3.6 \
        --with-ssl=../openssl-0.9.3a \
        --prefix=/usr/local/apache \
        --enable-shared=ssl \
        --enable-module=rewrite \
        --add-module=../mod_auth_mysql.c
```

Zadnja opcija (–add-module) je dodana samo zato da bi se pokazalo kako dodati neki vanjski modul ako kompajlirate mod_ssl sa Apache-om. Ukoliko Vam ne treba autorizacija u Apache-u korištenjem MySQL baze, slobodno izostavite –add-module.

Predzadnja opcija (–enable-module=rewrite) uključuje rewrite modul koji je standardni dio Apache distribucije, ali se ne uključuje u standardnoj konfiguraciji. Kako je modul veoma koristan, svakako preporučujemo njegovo uključivanje i proučavanje dokumentacije o njemu.

```
> cd ../apache_1.3.6
> make
```

4.3 Čudne greške kod kompajliranja Apache-a. Ako vam se desi da kod kompiliranja Apache-a sa dodatnim modulima dobijete pogrešku koja glasi:

```
+ doing sanity check on compiler and options
** A test compilation with your Makefile configuration
** failed. This is most likely because your C compiler
** is not ANSI. Apache requires an ANSI C Compiler, such
** as gcc. The above error message from your compiler
** will also provide a clue.
 Aborting!
```

nemojte odmah očajavati. Naime, najvjerojatnije je greška u tome da se neki od modula nije mogao kompajlirati i linkati sa nekom od biblioteka. Kako otkriti sa kojom? Otiđite u `src` poddirektorij i pokrenite `./helpers/TestCompile -v sanity`. To je zapravo shell skripta koja se pokreće, a opcija `-v` će ispistati detaljniji opis pogreške. U našem slučaju ona je bila:

```
/usr/bin/ld: cannot open -lmysqlclient: No such file or directory
```

što znači da je unutar modula u konfiguracijskom dijelu trebalo promijeniti putanju do mysqlclient biblioteke.

5 Izrada probnog certifikata

[Ovdje bi trebao doći opis što je to cetrifikat, i zašto je potreban -op.a.]

Nakon što se apache server (nadam se uspješno) kompajlira, možete napraviti svoj prvi certifikat korištenjem:

```
> make certificate TYPE=custom
```

Mislim da ćete znati što treba upisati na koje pitanje. Naročito je važno da za `Common Name` unesete točno ime servera jer Vam u protivnom certifikat neće raditi. Ako Vaš server ima više imena (CNAME-a), morate izabrati jedno za SSL veze i koristiti **samo** njega jer ćete inače morati imati onoliko certifikata koliko želite imati imena servera. Dobar primjer za takvo ime je `secure.proba.hr` ili `ssl.proba.hr`.

Drugo zanimljivo pitanje je da li želite zaštititi svoje privatne ključeve. Kako je to dobra ideja ukoliko postoji mogućnost da netko dobije pristup račualu (npr. ako je računalo koje poslužuje `https` server višekorisničko) to ima i svojih nedostataka. Svaki puta kod pokretanja servera morati ćete upisivati tu istu zaporku. To zapravo zanči da se server neće moći startati bez da netko (tko zna zaporku) ne bude pokraj njega.

Za razliku od starijih verzija, ovdje razmatrana verzija sama kreira certifikat koji Vam vrijedi 365 dana.

Ukoliko namjeravate ozbiljno koristiti `https` morati ćete kupiti pravi certifikat. Jedan od dobrih adresa je `<http://www.thawte.com/>`

6 Instaliranje servera

Nakon svega ovoga, ovo je stvarno jednostavno:

```
> make install
```

Server će biti instaliran u `/usr/local/apache`.

7 Pokretanje i isprobavanje servera

Sada možete server i pokrenuti:

```
> /usr/local/apache/bin/apachectl startssl
```

Nakon što se server pokrenuo, na portu 8080 nalazi se obična, a na 8443 `https` verzija vaših stranica. Kako Vaš certifikat (kojeg ste napravili sa `make certificate`) nije potpisala niti jedna organizacija koja se bavi izdavanjem digitalnih certifikata, morati ćete u Netscape-u odlučiti da želite prihvaćati takve certifikate prvi puta kada pristupite svojim stranicama korištenjem `https://secure.proba.hr:8443/`.

Važno je također napomenuti da će Inernet Explorer odbiti kontaktirati servere koji nemaju certifikat od nekog od priznatih CA-ova. Taj problem možete rješiti tako da datoteku server.crt koju ste kreirali sa `make certificate` "podmetnete" IE-u korištenjem običnog `http` protokola. Nakon toga IE će vas pitati da li da importira taj certifikat, baš kao i Netscape.

8 Konfiguriranje servera

Sada možete promijeniti port 8443 u standardni SSL port 443, a port 8080 prebaciti na standardni 80. Tako ćete istom sadržaju moći pristupati i preko `http`-a i `https`-a. To vjerojatno **nije** ono što ste željeli, ali konfiguriranjem virtualnih hostova riješiti ćete i što je zaštićeno, a što nije...

9 Mogući problemi

Uz malo sreće, sve radi kako bi i trebalo. Međutim, možda ćete naići na sljedeće probleme (popis problema uključuje i starije verzije programa i/ili modula):

1. Starije verzije apache-ssl-a su odbijale učitati certifikate kod pokretanja servera. To je doduše do danas sređeno (to se odnosi na servere verzije 1.1.3 sa SSLeay-om 0.6.x), ali nikad se ne zna. :-(

2. Ukoliko ste koristili apache 1.2.6 i SSLeay 0.8.0, neće Vam raditi cache-ing. Zbog toga dodajte

```
#define CACHE_SESSIONS          FALSE
```

   u `apache\_ssl.c` da biste isključili tu opciju. Također, morate zakomentirati sve *Cache* direktive u `SSLeay/conf/httpd.conf` datoteci. Iako bi se po source kodu moglo zaključiti da bi to trebalo raditi od verzije 0.8.0, nisam baš sasvim siguran u to. Naime, sa SSLeay-om 0.9.0b (i uključenim cache-ingom) je server pri startanju jednostavno bacio core-u.

   Kod Apache 1.3.0 servera, sa druge strane radi cache-ing ako se koristi TCP socket, a ne port. To je probano sa SSLeay-om 0.8.0.

3. Sve radi savršeno, osim što `httpsd` ne želi raditi na portu 443? Pa, zapravo potrebno je pokrenuti server kao root (inače se ne može bind-ati na portove manje od 1024). Ako još uvijek ne radi, provjerite svoje firewall rule-ove!

4. Ponekad se Apache ne želi kompajlirati zbog toga što ne može naći neki include file ili biblioteku. To je čest slučaj kada dodajete module. mod_rewrite npr. koristi ndbm.h koji se na RedHat distribuciji nalazi u `/usr/include/db1/ndbm.h` a ne u `/usr/include/ndbm.h` gdje ga on očekuje. Također, pogledajte u `mod\_auth\_mysql.c` prije kompajliranja

10 Na kraju, disclaimer, zašto ovaj dokument?

Puno sreće u korištenju Apache sa SSL-om i ukoliko imate nerazjašnjenih problema možete kontaktirati autora na *dpavlin@rot13.org*.

Također, uzmite u obzir da komercijalna rješenja koštaju desetke tisuća dolara, pa ovakva (iako potpuno funkcionalna) ipak imaju svoje bubice.

Za svoje podatke, mogućnost korištenja i eventualne gubitke zbog nemogućnosti pristupa serveru odgovarate isključivo i samo Vi (`std.disclaimer`?).

Možda se pitate zašto sam uopće išao pisati ovaj dokument? Pa, kako sam instalirao nekoliko apache-ssl servera shvatio sam (na najteži način – na vlastitoj koži) da cijela procedura nije jednostavna i bez problema. Tako je nastala prva verzija ovoga dokumeta, u želi da drugi ne moraju prolaziti isti bolan put. Tokom vremena, alati su postajali sve bolji, a ovaj dokument sve duži. Na kraju sam odustao od podržavanja Apache-SSL patch-a jer je mod_ssl bio mnogo čišće i jednostavnije rješenje. Da danas instaliram prvi Apache sa mod_ssl-om ovaj dokumet vjerojatno nikada ne bi ni nastao.

11 Adrese na mreži (gotovi binary!)

"Ma sve je to super, ali ja sam korisnik Crvenog Šeširića (RedHat) i samo trebam `rpm` da instaliram to čudo..." `<ftp://ftp.replay.com/pub/replay/pub/redhat/i386/>`