Multiplekser sslh

Softver otvorenog koda nudi mnoge prednosti nad vlasničkim softverom. Zbog otvorenosti koda, možete iz njega učiti i usavršavati svoje znanje konkretnog programskog jezika. Ukoliko ste zabrinuti za sigurnost, možete provjeriti sadržava li kod neke maliciozne elemente. Mnogima je mogućnost modifikacije, odnosno mogućnost prilagodbe za vlastite potrebe najbitnija. U slučaju kojeg ćemo opisati, uz pomoć dodatnog programa, napravljena je nadgradnja i povezivanje dva nevezana programa u jednu cjelinu.

Radi se o promjeni ponašanja dva standardna servisa na poslužitelju, HTTPS i SSH, kojima je pomoću programa **sslh** omogućeno da slušaju zahtjeve na istom portu. Ovo je zgodna mogućnost kada ste iza vatrozida nad kojim nemate kontrolu, ili jednostavno nastojite sakriti SSH servis na nestandardnom portu. Za zajednički port odabrat ćemo standardni port 443, a time ćemo ujedno izbjeći i potrebu bilo kakve modifikcije na klijentu-browseru. Klijentu za SSH je lako reći da se proba spojiti na neki drugi port.

Kako sslh uspijeva vrtiti dva servisa na jednom portu? To je moguće pomoću malog trika, odnosno prirode ova dva protokola. SSH servis kod spajanja klijenta prvi "progovara" i ispisuje pozdravnu poruku i inačicu:

```
# telnet localhost 22
Trying 127.0.0.1...
Connected to localhost.
Escape character is '^]'.
SSH-2.0-OpenSSH_5.5p1 Debian-6+squeeze3
```

Kod SSL-a je situacija drugačija, u ovom slučaju klijent progovara prvi. Koristeći ovu činjenicu, **sslh** može posluživati oba protokola. Kod spajanja klijenta **sslh** "šuti" i čeka što će klijent poslati. Ukoliko se ništa ne dogodi određeno vrijeme (2s), radi se o SSH klijentu, te se klijent prepušta SSH daemonu.

Kod SSL-a klijent prvi progovara, te je jasno da će u tom slučaju biti kontaktiran HTTPS poslužitelj, a u većini slučajeva će to naravno biti apache.

Jasno je da će zbog ovih preusmjeravanja biti potrebno napraviti preinake u Apachejevom servisu HTTPS, kao i konfigurirati **sslh**. **Sslh** instaliramo na klasičan Debianovski način (ako je to uopće više potrebno navoditi):

```
# apt-get install sslh
...
Setting up sslh (1.6i-4) ...
Adding system-user for sslh daemon
Service disabled by default, please adjust the configuration to your needs
in /etc/default/sslh.
```

Sslh ima samo jednu izvršnu datoteku, konfiguracija se nalazi u **/etc/default/sslh**, a starta se sa "/etc/init.d/sslh start".

Prvo ćemo konfigurirati Apache, jer njemu uzimamo port za SSL. On više ne može slušati na portu 443, pa mu treba definirati novi slobodni port, na primjer 8080 koji se često koristi za HTTP proxy. Trebamo otvoriti datoteku **/etc/apache2/sites-available/default-ssl** (ili bilo koju drugog imena gdje vam je definiran SSL host), pa promijeniti redak VirtualHost tako da izgleda ovako:

```
<VirtualHost _default_:8080>
```

U Debianu je potrebno na još jednom jestu promijeniti konfiguraciju, a to je /etc/apache2/ports.conf:

```
#<IfModule mod_gnutls.c>
#    Listen 8080
#</IfModule>

<IfModule mod_ssl.c>
    Listen 8080
</IfModule>
```

Moduli mod_gnutls i mod_ssl su međusobno isključivi, odlučite se za jedan.

Nakon restarta apacheja, provjerite iz nekog browsera radi li HTTPS na novom portu, tako da ga dodate na kraj URL-a:

```
https://server.domena.hr:8080
```

Ukoliko je sve u redu, možemo dalje na konfiguraciju sslh-a. U **/etc/default/sslh** dodajte sljedeće (u osnovi, samo treba dodati opciju "RUN"):

```
RUN="yes"
DAEMON_OPTS="-u sslh -p 0.0.0.0:443 -s 127.0.0.1:22 -l 127.0.0.1:8080
   -P /var/run/sslh.pid"
```

Iako se iz samih parametara može zaključiti čemu služe, objasnit ćemo ih:

-p adresa i port na kojem sluša sslh, 0.0.0.0 označava sva sučelja/adrese
-s adresa i port za SSH daemon
-l adresa i port za SSL daemon
-P datoteka u koju će seupisati PID procesa sslh
-t timeout za SSH klijenta, default je 2 sekunde
-u user od kojim će se vrtiti sslh (podržano je spuštanje privilegija)

Ostaje samo pokrenuti daemon:

```
# /etc/init.d/sslh start
Starting ssl/ssh multiplexer : sslh.
# pgrep sslh
15522
```

Provjerite radi li SSL pomoću nekog browsera, ovaj put bez dodanog porta. Za SSH, klijentu je potrebno dodati opciju "-p 443":

```
server2$ ssh korisnik@server.domena.hr -p 443
The authenticity of host '[server.domena.hr]:443' can't be established.
RSA key fingerprint is 3e:6c:ac:a4:43:e0:86:fb:cf:d9:df:1f:7e:ae:8e:02.
Are you sure you want to continue connecting (yes/no)? yes
Warning: Permanently added '[server.domena.hr]:443,[161.53.X.Y]:443' (RSA)
to the list of known hosts.
korisnik@server.domena.hr's password: ********
Linux po 2.6.32-5-686-bigmem #1 SMP Sun Sep 23 10:27:25 UTC 2012 i686
server$
```

Ukoliko nešto ne radi, probajte restartati sve daemone, a pri tome provjerite jesu li se doista ugasili i oslobodili portove (naredba **lsof -i** će vam u tome pomoći).

Jedna mana ovakvog multipleksiranja je što se u logovima više neće vidjeti vanjska adresa, nego lokalna adresa. No, zato sslh ima svoje logove koje zapisuje u datoteku /var/log/auth.log. Logovi izgledaju ovako:

```
Apr  4 01:39:29 server sslh[20361]: connection from X.207.108.Y:2627 forwarded to SSL
Apr  4 01:40:55 server sslh[20934]: connection from 161.53.X.Y:34034 forwarded to SSH
```

Ukoliko ste zabrinuti za sigurnost, pristup sslh daemonu je moguće ograničiti preko poznate biblioteke **tcpd/libwrap**, odnosno njezinih datoteka **/etc/hosts_allow** i **/etc/hosts_deny.**

Više informacija možete naći naredbom man i u datoteci README na stadardnom mjestu: **/usr/share/doc/sslh**.

uto, 2013-04-09 00:50 - Marko Jukić**Kuharice:** Linux [1]
Kategorije: Software [2]
Vote: 0

No votes yet

Source URL: https://sysportal.carnet.hr/node/1246

Links
[1] https://sysportal.carnet.hr/taxonomy/term/17
[2] https://sysportal.carnet.hr/taxonomy/term/25