

Published on [ONLamp.com](http://www.onlamp.com/) (<http://www.onlamp.com/>)
http://www.onlamp.com/pub/a/bsd/2003/07/03/FreeBSD_Basics.html
[See this](#) if you're having trouble printing code examples

HTTP Proxy-ji

by [Dru Lavigne](#)
07/03/2003

U [proslom clanku](#), spoznali smo prednosti koje se dobijaju koriscenjem proxy-ja. U danasnjem clanku, usredsredicu se na HTTP proxy-je. Pogledacemo neke od HTTP proxy-ja dostupnih u kolekciji portova i videcemo koji proxy-ji odgovaraju kojim potrebama.

Ako ste imalo upoznati sa HTTP proxy-jima, vasa prva misao ce verovatno biti [Squid](#), odlicni HTTP proxy. Posto vec postoji dosta dobrih clanaka i tutorijala na temu koriscenja i konfigurisanja Squid-a, njega cu preskociti u ovom serijalu. Za one koji su razocarani, dacu vam par URL-a:

- Odlicni serijal o Squid-u od [Jennifer Vesperman](#)
- "[Podesavanje Squid-a na FreeBSD-u](#)" na Daemonnews
- [Squid Konfiguracioni Prirucnik](#)

Squid je primer veoma konfigurabilnog HTTP proxy-ja koji se moze uklopiti u veoma velike mreze. Ovo je veoma dobro ako ste administrator veoma velike mreze, ali pravo ubistvo ako jednostavno zelite da bezbedno surfujete sa vaseg FreeBSD kompjutera ili da primenite polise u maloj kucnoj mrezi. Razmisljajuci kao korisnik, koje su to iritirajuce stvari koje dolaze sa web surfovanjem? Sledece stvari prvo padaju na pamet:

- Pop-up prozori
- Reklame
- Kolacici
- [Webbugs](#)
- Java apleti
- Shockwave introi
- Brzina, ili manjak brzine

U zavisnosti od web citaca koji koristite, neke od ovih iritirajucih stvari se mogu direktno spreciti. Ostale zahtevaju da instalirate dodatni proxy softver. Pocnimo sa pregledom nekih od najcesce koriscenih web citaca, a onda cemo precizno na odgovarajuce proxy-je.

Opcije Web Citaca

U toku pisanja ovog clanka, ovo su zadnja (slobodna) izdanja tri popularna web citaca:

- mozilla-1.3_1,2
- opera-6.12
- linux-netscape-navigator-4.8

Imajte na umu da nove opcije dolaze sa novim verzijama, tako da one opcije koje sada nedostaju se mogu pojaviti kasnije u novim verzijama. Isto tako, svaki web citac ima "Preferences" odeljak, tako da ako vas citac nije ovde naveden, pregledajte taj odeljak da vidite koje su opcije dostupne.

Sto se tice ovih citaca, odeljak Preferences se nalazi u Edit meniju za Netscape i Mozillu, i u File meniju za Operu. Videcete velike razlike u dostupnim opcijama izmenju Netscape i Mozilla ili Opera citaca. Ovo je zbog toga sto je ovo starija verzija Netscape citaca.

Sva tri citaca poseduju odgovarajuca podesavanja koja vam omogucavaju upravljanje kolacicima. Omogucavaju i da odobrite ili zabranite Java i JavaScript. i na kraju, ako imate sporu Internet vezu i dosta prostora na disku, moci cete da poboljsate brzinu tako sto cete prilagoditi podesavanja za kes u svakom citacu.

Kontrola iskakucih prozora je novija opcija, tako da ne postoji u ovoj verziji Netscape citaca. u Operi, kliknite na General da nadjete podesavanja za zabranu iskakucih prozora. Mozilla ide korak dalje tako sto blokira sve iskacuce prozore ili u zavisnosti od sajta do sajta. Da onemogucite sve iskacuce prozore, idite do Privacy & Security->Popup Windows i procitajte upozorenje o racvanju. Alternativno, kada naidjete na stranicu sa iritirajucim iskacucim prozorima, jednostavno kliknite na stranicu desnim klikom i izaberite "Reject popup windows from this site."

bfilter

Pogledajmo sada sta neke aplikacije u kolekciji portova mogu da urade da uvecaju opcije koje su vec dostupne u vasem omiljenom web citacu. Pocecu sa [bfilter](#)-om. Ovaj HTTP proxy ne samo da kontrolise iskacuce prozore, vec i zaustavlja one dosadne svetlucave reklame i obecava da onemoguci webbugs. Da bi instalirali ovaj port, postanite superuser i:

```
# cd /usr/ports/net/bfilter
# make install clean
```

Port ce instalirati aplikaciju u `/usr/local/bin/bfilter` i konfiguracioni fajl u `/usr/local/etc/bfilter/config`. Kada se kompajliranje zavrshi, izadjite iz superuser naloga i ukucajte `bfilter` da bi startovali proxy. Onda proverite da li proxy slusa za zahteve:

```
$ sockstat -4
USER      COMMAND  PID    FD PROTO  LOCAL ADDRESS          FOREIGN ADDRESS
dlavigne  bfilter  20336  3  tcp4   127.0.0.1:8080        *:*
```

Primiticete da `bfilter` slusa na protu 8080 na loopback adresi. Ako procitate komentare u njegovom konfiguracionom fajlu, videcete da 127.0.0.1 znaci da sluca HTTP zahteve na svim interfejsima. Ako zelite da slusate na samo jednom interfejsu, odredite njegovu IP adresu u konfiguracionom fajlu.

`bfilter` nije transparentni proxy, sto znaci da cete morati da konfigurisete vas web citac da bi mogao da koristi proxy. Idite u Preferences odeljak vases web citaca i trebalo bi da nadjete podesavanje koja se odnose na proxy. Ukucajte IP adresu i broj porta koji koristi `bfilter`. U mom primeru, `bfilter` radi na istoj masini kao i moj citac, tako da koristim 127.0.0.1 kao IP adresu i 8080 za broj porta. Ako pokrecete `bfilter` na posebnom kompjuteru, promenite IP adresu u konfiguracionom fajlu koja ce predstavljati IP adresu mrezne kartice prikacene na vasu unutrašnju mrežu. Onda podesite citace na kompjuterima u vasoj mrezi da koriste IP adresu u njihovim podesavanjima za proxy u Preferences. `bfilter` ima i fajl sa pravilima, koji se nalazi u `/usr/local/etc/bfilter/rules`. Meni su i podrazumevana pravila radial besprekorno u hvatanju iskakucih prozora i reklama. Ako trazite proxy koji je lak za koriscenje i koji radi onako kako ga instalirate bez dodatnih podesavanja, `bfilter` moze biti dobro resenje.

middleman

Jos jedan HTTP proxy koji uzivam da koristim je [middleman](#). Kao i `bfilter`, radi cim ga instalirate, ali ono sto ovaj proxycini interesantnim su dodatne opcije koje omogucavaju da na zavodljiv nacin naucite vise o HTTP i sta se desava iza scene svaki put kada posetite web stranicu.

Prvo instalirajmo port:

```
# cd /usr/ports/www/middleman
# make install clean
```

Primitite da ce ime instalirane aplikacije biti `/usr/local/bin/mman`. Trebate znati i ime podrazumevanog konfiguracionog fajla da bi ste mogli da pokrenete aplikaciju. Ako samo ukucate `mman`, dobicete help fajl. Umesto toga, use the `c` ili config-file dodatak da pokrenete proxy:

```
# mman -c /usr/local/etc/mman.xml
```

Nalazim da proxy mora biti pokrenut kao superuser. Ne zaboravite da pogledate port na kojem `mman` slusa i podesite proxy podesavanja u vasim web sitacima:

```
sockstat -4
USER      COMMAND   PID    FD PROTO  LOCAL ADDRESS    FOREIGN ADDRESS
root      mman      575    0 tcp4    127.0.0.1:8080   *:*
```

Ako planirate da koristite `middleman`, odvojite vreme da procitate `/usr/local/share/doc/middleman/README.html`. Ovo je jedina dokumentacija za aplikaciju, ali je veoma detaljna i sadrzi dosta interesantnih ideja kako koristiti proxy. Iako ce podrazumevana konfiguracija verovatno zadovoljiti vase potrebe, trebalo bi da pogledate i web interfejs tako sto cete ukucati `mman` u vas citac. Ovo ce vam omoguciti da pogledate:

- Aktivne konekcije
- Upise u logovima
- Konfiguraciju
- Cache podesavanja
- DNS cache
- Headers
- Connection pool

Malo o HTTP

Ako nikada niste upravljali HTTP serverom ili HTTP proxy-jem ranije, zapanjicete se kolicinom interakcije koja nastaje kad god se web sitac poveze na web server. U zanjem clanku smo napomenuli da cemo se pozivati na [HTTP RFC \(2616\)](#). Pogledajmo nakratko kako HTTP protokol radi; Ostavicu na vama da pogledate RFC da upotpunite detalje koji vas interesuju.

Uvek kada posetite web stranicu, vas citac mora da napravi poseban zahtev za svaki detalj na toj stranici. Na primer, ako ukucam slashdot.org u moj citac, videcu sledece unose u mom `mman` kesu:

- <http://images.slashdot.org:80/topics/topicgamesrts.gif>
- <http://images.slashdot.org:80/topics/topicinternet.gif>
- <http://images.slashdot.org:80/title.gif>
- <http://slashdot.org:80/>
- <http://images.slashdot.org:80/topics/topicaposx.gif>
- <http://images.slashdot.org:80/topics/topicms.gif>
- <http://images.slashdot.org:80/topics/topiccomdex.gif>
- <http://images.slashdot.org:80/slc.gif>
- <http://images.slashdot.org:80/pix.gif>
- <http://images.slashdot.org:80/topics/topicscience.gif>
- <http://images.slashdot.org:80/greendot.gif>
- <http://slashdot.org:80/favicon.ico>
- <http://images.slashdot.org:80/topics/topichardware.gif>

Primitite da je svaki GIF ili slika poseban zahtev, zato sto je svaki sacuvan kao poseban fajl na web serveru. Da bi moj citac mogao da prikaze glavnu stranicu Slashdot sajta, morao je pojedinačno da zahteva svaki od 11 .gifs, jedan .ico, i HTML stranicu koja objasnjava kako da se to sve formatira zajedno.

U HTTP-u, postoji dva tipa paketa: paket zahteva i paket odgovora. Paket zahteva uvek dolazi od web citaca. Ovo ima smisla, jer je web citac klijent i posao klijenta je da pravi zahteve. Sa druge strane, paketi odgovora uvek dolaze od web servera.

Paket zahteva web citaca sadrzi tri komponente:

- Metod
- Header
- Telo

Metoda ukazuje na to sta klijent zahteva. Metode su izlistane i objasnjene u RFC i obicno se pisu velikim slovima. Najcesci metod je **GET** metod, zato sto vas web citac zeli da "dobije" odredjenu stranicu ili sliku od web citaca. Ako pogledate vas **mman** log, ili log sa bilo kojeg HTTP proxy-ja ili HTTP servera, videcete **GET** zahteve:

```
Sat 21 16:04:43 [575] request: GET
http://images.slashdot.org:80/greendot.gif
Sat 21 16:04:43 [575] cache: create:
http://images.slashdot.org:80/greendot.gif
Sat 21 16:04:43 [575] request: GET http://images.slashdot.org:80/pix.gif
Sat 21 16:04:43 [575] cache: create: http://images.slashdot.org:80/pix.gif
Sat 21 16:04:43 [575] request: GET
http://images.slashdot.org:80/topics/topicgamesrts.gif
Sat 21 16:04:43 [575] cache: create:
http://images.slashdot.org:80/topics/topicgamesrts.gif
Sat 21 16:04:43 [575] request: GET
http://images.slashdot.org:80/topics/topiccomdex.gif
Sat 21 16:04:43 [575] cache: create:
http://images.slashdot.org:80/topics/topiccomdex.gif
<snip>
```

Ovde, **mman** je izdao **GET** zahtev u ime mog citaca, i onda ubacio kopiju zahtevanog detalja usvoj kes.

Paket odgovora web servera isto sadrzi tri komponente:

- Status
- Header
- Telo

To znaci, paket zahteva salje metod, i web server odgovara sa statusnom porukom. Status poruke su numericke, i one su izlistane u RFC. Verovatno ste nekada naisli na "404 error," **404** je statusni broj koji predstavlja "nije nadjeno." Najcesci status je **200** ili **OK**. Ako web citac izda **GET** zahtev i server nadje zahtevani resors, poslaca ga nazad zajedno sa statusom **200**. Ako ne moze da nadje zahtevani fajl, umesto toga ce poslati status **404**.

Verovatno ste primetili da i zahtevani i paketi odgovora sadrže headere i telo. Telo obicno sadrzi zahtevanu stranicu ili sliku. Znac, kada je moj web citac izdao **GET** zahtev za <http://images.slashdot.org:80/greendot.gif>, web server je nasao GIF i poslao paket odgovor sa statusom **200** i sam GIF u telu tog paketa.

Prikazivanje Headera sa **mman**

Header-i su interesantni deo HTTP paketa. Oni sadrže korisne informacije koje pomazu web citacu i web serveru da medjusobno komuniciraju. Isto tako sadrže i osteljive informacije o web serveru i web citacu. Ovo su rezultati nakon klika na Show Headers u **mman**-ovom web interfejsu:

Unfiltered

```
Type      Value
Host      mman
User-Agent Mozilla/5.0 (X11; U; FreeBSD i386; en-US; rv:1.3.1)Gecko/20030619
Accept    text/xml,application/xml,application/xhtml+xml,text/html;q=0.9,text/plain;q=0.8,video/x-mng,image/png,image/jpeg,image/gif;q=0.2,*/*;q=0.1
Accept-Language en-us,en;q=0.5
Accept-Encoding gzip,deflate,compress;q=0.9
Accept-Charset ISO-8859-1,utf-8;q=0.7,*;q=0.7
Keep-Alive 300
Proxy-Connection keep-alive
Referer   http://mman/headers
```

Filtered

```
Type      Value
Host      mman
Accept    text/xml,application/xml,application/xhtml+xml,text/html;q=0.9,text/plain;q=0.8,video/x-mng,image/png,image/jpeg,image/gif;q=0.2,*/*;q=0.1
Accept-Language en-us,en;q=0.5
Accept-Encoding gzip,deflate,compress;q=0.9
Accept-Charset ISO-8859-1,utf-8;q=0.7,*;q=0.7
Referer   http://mman/headers
User-Agent Mozilla/4.0 (compatible; MSIE 6.0; Windows NT 5.0; Q312461)
```

Zapamtite, svaki HTTP paket sadrzi header-e. Ovde vidite vrednosti koje su poslate od strane mog web citaca. **Unfiltered** sekcija sadrzi defaulte koje koristi moj web citac. Citko prikazuje moj Operativni sistem i verziju i tip web citaca koji koristim. **Filtered** prikazuje da je **mman** promenio neke od tih header-a pre nego ih je poslao do web servera. Ako mi se ne svidjaju ove nove vrednosti, mogu jednostavno da kliknem na Config, odaberem header, i da editujem, npr, **User-Agent**. Ova konfiguraciona sekcija je veoma snazna, zato sto mozete da dodajete, brisete, i modifikujete sadrzaj header-a. Nemojte ovo ipak raditi tek onako. Prvo procitajte RFC i shvatite razgranavanje pojedinih header vrednosti sa kojima ce te se uhvatiti u kostac.

Interesantno je videti i header-e koji su poslani od strane servera. Ako ukucam ovaj URL u moj citac i stavim dve tacke izmedju reci "headers" i URL-a:

```
headers..www.mp3.com
```

Videcu ovo:

```
*Server header:*
```

```
HTTP/1.1 200 OK
Date: Sat, 21 Jun 2003 21:17:43 GMT
Server: Apache/1.3.12m1 (Unix) yas1/2.25 sw/1.7 mod_rdbcookie/1.2
      mod_mp3idver/0.12 rwh/1.1 bw/3.37 rewrite/3.3 include/3.6
Connection: close
Transfer-Encoding: chunked
Content-Type: text/html
```

Primitite da ne postoje tajne ni na serveru. Header jasno oznacava tip i verziju web server softvera koji se koristi. Ako ste odgovorni za odrzavanje web servera, zapamtite da svaki HTTP koji napusta web server otkriva da li ste ili niste instalirali zadnje web server zakrpe!

Kontrola Pristupa

`mman` podrzava i opcije koje mogu biti veoma korisne u mreznom okruzenju. Prvo, moze da primora korisnike da se autorizuju pre nego sto im se dozvoli da koriste Internet. Kliknuću na `config` i izabrati `access` i dodati polisu. Onda ce mi se otvoriti forma.

Ako ostavim sekciju za IP adresu praznom, polisa pristupa ce se primeniti na svaku IP adresu koja se poveze sa proxy-jem. Onda mogu da postavim vrednosti u poljima za ime korisnika lozinku. Pre nego sto sacuvam polisu, moram da konfiguršem koji pristup ce korisnicima biti dozvoljen kada ukucaju ispravno korisnicko ime i lozinku. Moj izbor je:

- *Web interface*: Ovo ce omoguciti korisnicima da konfigurisu proxy, zbog toga cu ovu opciju ostaviti necekiranu.
- *Proxy requests*: Ako cekiram ovu opciju, proxy ce prihvatiti zahteve web citaca koji su rucno konfigurisani da koriste IP adresu i broj porta proxy-ja.
- *CONNECT requests*: `CONNECT` je HTTP metod koji se cesto onemogucava zbog [svojih bezbednosnih rizika](#).
- *HTTP requests*: Zapamticu da izaberem ovu opciju, ili korisnici nece mozi da pristupe HTTP serverima.
- *Transparent proxying*: Ako cekiram ovu opciju, proxy ce presresti web zahteve, i u slucaju da web citac nije konfigurisan da koristi proxy. Ovo je u sustini dobra stvar za mrezu, zato sto osigurava da korisnici nece moci da zaobidju vas proxy server.
- *Allow bypassing*: `mman` sadrzi reci koje mogu biti ukljucene zajedno sa URL-om da bi se zaobisle restrikcije za odredjeni sajt. Na primer, Ako zelim da vidim iskacuce prozore za neki sajt, otkucacu ovo u mom web citacu: `bypass[f].www.mp3.com`. Ako ne zelite da korisnici zaobilaze vase filtere, nemojte selektovati ovu opciju.

Ako odlucite da kreirate vasu polisu, zapamtite da kreirate i drugu polisu koja ce vam omoguciti, kao administratoru, da konfigurisete `mman`. Ako planirate da konfigurisete `mman` na istom kompjuteru koji ima proxy softver, zadržite podrazumevane polise, ali ih stavite ispod vasih novih polisa koje se ticu vasih korisnika.

Sada, kada korisnici pokrenu svoje web citace, sam citac ce ih upitati za korisnicko ime i lozinku koje ste kreirali u svojoj polisi. Ako je upisu tacno, imace pristup Internetu, prema parametrima koje ste vi odredili u vasim polisama. Zadnja opcija koju zelim da pomenem je `limits`. Ova konfiguracija vam omogucava da kontrolisete pristup Internetu u zavisnosti od meseca, dana i sati. Na primer, mozete konfigurisati polisu koja ogranicava pristup Internetu u vremenu izmedju 9:00 do 17:00 od Ponedeljka do Petka.

Zakljucak

Cini se da smo tek zagreballi površinu `middleman` proxy servera. Mozda vas je zaintrigiralo da sami probate ovu aplikaciju. U sledecem clanku, završicu sa proxy serijalom tako sto cemo pogledati neke SMTP proxy-je.

ˆDalibor Gudzic