

Kapitel 9

Apache



Fels II

9.1 LAMP

LAMP är ett känt begrepp när man använder servrar med fri mjukvara. Det står för Linux, Apache, Mysql och PHP. P:et kan även betyda Perl eller Python om man föredrar de språken, men PHP är det vanligaste. Denna bok handlar mest om L&A, det kanske kommer en fortsättning om M&P senare. Apache är den populäraste webbservern på Internet. Enligt Netcraft körs Apache på 56% av nätets webbserverar (augusti 2010). Den kanske inte är den allra snabbaste, men är pålitlig och går att konfigurera för de flesta behov.

Apache började utvecklas 1995 som patchar till en äldre server från NCSA. A Patchy Server. Licensen för Apache är inte GPL, utan en egen variant. Men man har tillgång till all källkod och det finns utvecklare över hela världen. Apache finns med i de flesta Linuxdistributioner, och kan därmed lätt installeras med yum eller apt-get. Paketet brukar heta httpd i RH-baserade distar och apache2 i Debian.

9.2 Prefork MPM

Jag ger här en informell beskrivning av Apache utifrån teckningen på föregående sida. Indianhövdingen är chefen, den httpd som ägs av root. Den öppnar port 80, skriver loggar och är förälder till de andra som ägs av apache eller www-data och som svarar på de olika GET-terna. Småindianerna gör det tunga jobbet efter att hövdingen har avlat dem.

Så här kan det se ut i processlistan:

```
[ao@einstein ~]$ ps faux|grep '[h]ttpd'|
cut -c 1-14,50-
root      2496   Sep25   0:00 /usr/sbin/httpd
apache    19265   Oct23   0:00 \_ /usr/sbin/httpd
apache    19266   Oct23   0:00 \_ /usr/sbin/httpd
apache    19267   Oct23   0:00 \_ /usr/sbin/httpd
apache    19268   Oct23   0:00 \_ /usr/sbin/httpd
apache    19269   Oct23   0:00 \_ /usr/sbin/httpd
apache    19270   Oct23   0:00 \_ /usr/sbin/httpd
apache    19271   Oct23   0:00 \_ /usr/sbin/httpd
```

```
apache 19272 Oct23 0:00 \_ /usr/sbin/httpd
apache 22332 Oct23 0:00 \_ /usr/sbin/httpd
[ao@einstein ~]$
```

Att enbart den första körs som root och de andra av lägre privilegierad användare är en del av säkerheten för en webbserver. Vilket många "förstör" genom att låta filerna ägas av apache (eller www-data som den heter på Debian).

Att ha ett antal vilande httpd beredda att svara på anrop gör att Apache kan svara snabbt. Den vanligaste MPM:en (Multi-Processing Module) heter prefork, hövdingen föder och dödar sina barn dynamiskt alltefter belastning, antalet lediga ungar brukar ligga mellan 5 och 8.

Prefork är standard MPM i Unix. Ett annat alternativ är worker som använder trådar istället för separata processer. Men eftersom prefork startar nya processer innan de behövs, och RAM är billigt, behöver man sällan byta. Standardinställningarna för antalet idle, min eller max behöver man inte heller ändra för en normalt belastad server.

9.3 Installation

Det finns olika sätt att installera Apache. Enklast kan vara att använda distributionens paket. Alltså `yum install httpd` på Red Hat och `apt-get install apache2` på Debian. Men man kan även ladda hem och kompilera själv, med:

```
tar xvf httpd-version.tar.gz; cd htt<TAB>
./configure && make && sudo make install
```

Använder du den metoden, och inte anger annat till `configure`, så hamnar alla filer under `/usr/local/apache2/`.

Om du däremot använder RPM-filerna till Red Hat eller CentOS så finns konfigurationsfilerna under `/etc/httpd/`, webbrotten `/var/www/html/`, binärerna finns i `/usr/sbin/`, startskript i `/etc/init.d/` och loggfilerna under `/var/log/httpd/`.

På Debian och Ubuntu finns konfigurationen i `/etc/apache2/`, webbrotten blir `/var/www/` och loggfilerna `/var/log/apache2/`.

En fördel med att använda färdiga paket är att uppgradering blir automatisk. Fördel med att kompilera själv är

att du får mer kontroll över vad som är med, men du bör hålla koll på när nya versioner kommer och kompilera om ifall säkerhetshål har upptäckts. Numera är de flesta moduler dynamiskt laddade, och servrar har mycket minne, så att ha en minimal storlek på Apache-processerna är inte lika viktigt som det var förr. Det kan dock vara intressant att prova att kompilera sin egen Apache om du aldrig har gjort det förut. Du kan ju testa det på nån mindre viktig maskin, mest för skojs skull.

Hemsidan för webbservern Apache är <http://httpd.apache.org>. Där kan du hitta såväl källkod som aktuell dokumentation.

9.4 Start/stopp av Apache

Det finns olika sätt att starta och stoppa Apache. För att starta automatiskt vid boot av Red Hat kan du skriva `chkconfig httpd on`. På Debian startar Apache automatiskt om det är installerat. Har du kompilerat själv så finns skriptet `apachectl`, till det kan du ange `start`, `stop`, `restart` och `graceful`. Med `graceful` startar du om snällt, alltså låter föräldern vänta tills alla barnen är klara med sina uppgifter. Till Red Hat och CentOS finns `/etc/init.d/httpd` som tar samma argument. I Debian och Ubuntu heter motsvarande skript `/etc/init.d/apache2`. Det går även att använda kommandot `service httpd restart`, både i Red Hat och Debian.

9.5 Moduler

Apache är uppbyggd av många moduler. Ett fåtal av dem behövs alltid och måste vara inkompilerade i programmet. Alla övriga är frivilliga och kan både vara inkompilerade eller laddas dynamiskt från konfigurationsfilerna. Att ha dem som dynamiska moduler är ofta att föredra, och det är det vanligaste om du installerar från distributionens paket.

9.6 Vad Apache gör

Apache omvandlar en URL till information, oftast på något vis från filer.

Mer detaljerat så skickar klienten, webbläsaren, en HTTP-förfrågan. Ofta med kommandot GET och en sökväg. Plus lite övriga detaljer, som vilka språk användaren föredrar och vilken version av HTTP (0.9, 1.0, 1.1) som klienten förstår.

URL betyder Uniform Resource Locator. Kort och gott webbadress. Till exempel:

```
http://einstein.df.lth.se/GLAN/frihet.html
```

Den kan delas upp i protokoll, `://`, datornamn, `/`, katalog och fil. När Apache på datorn `einstein` får HTTP-anropet: `GET /GLAN/frihet.html`

(och `Host: einstein.df.lth.se` på en annan rad utifall den har flera namn) så översätter Apache sökvägen ”på Internet” till en sökväg i filsystemet. I detta fall filen:

```
/var/www/html/GLAN/frihet.html
```

Apache skickar denna fil tillbaka till klienten. Denna fil är ren text, men med HTML-taggar för att beskriva strukturen. Ifall dokumentet innehåller bilder `` medför var och en av dessa en ny HTTP-förfrågan med GET från webbläsaren. Webbservern skickar filerna en och en tillbaka. Och Firefox, Epiphany, Chrome eller vad nu surfaren kör, gör om alla filerna till en webbsida.

Det var den enkla varianten, med statiska sidor och enstaka bilder. Vissa sidor skapas istället varje gång besökaren tittar på dem, för det kan servern köra PHP (eller andra språk). Då kan filen på servern innehålla programkod som i sig är oförändrad, men ger olika resultat beroende på andra faktorer. Då måste Apache kunna köra koden, för det används modulen `mod_php`. Andra sätt att generera dynamiskt innehåll är CGI och SSI, men de är inte lika populära numera.

En annan sak som Apache kan sköta är autentisering (authentication) och auktorisation (authorization). Den förstnämnda är att visa vem man är, den andra att tillåta eller neka åtkomst. Se mer om detta i kapitel 15.3 om säkerhet.

För att krångla till sakerna mera kan Apache följa symboliska länkar, skriva om URL:er till andra URL:er, köra vis-

sa saker som olika användare, skapa egna sidor med status, visa enbart en sidas ändringsdata eller HTML-huvud eller neka åtkomst. Och inte minst skicka status 404: File not found.

9.7 Andra alternativ

Apache är lite av webbservrarnas Volvo. Vanlig, pålitlig och relativt flexibel. Men varken det snabbaste eller bränslesnålaste alternativet på marknaden.

Jag nämner kort några andra webbservrar, alla är fria och används ganska ofta.

- Lighttpd är släppt med BSD-licens. Den är liten, säker och snabb. Ofta används den som komplement till Apache, eller ensam. Både YouTube och TPB kör på Lighttpd.
- Nginx uttalas Engine-X. Den är rysk och ganska populär. Har BSD-liknande licens. Den används ofta som omvänd proxy, även för SSL. Ca 11% av alla sajter kör Nginx.
- thttpd är också en liten och snabb server. En fördel är att det är lätt att begränsa bandbredd, för t.ex. bilder, med den.