Web server

from scratch

Odabir distribucije:

Vaša distribucija trebala bi zadovoljiti slijedeće zahtjeve:

- **Posjedovanje paketnog sustava (rpm, deb, tgz...)**
- **Redovito izdavanje patcheva za trenutne rupe u paketima**
- **Robusnost**
- **Stabilnost**
- **Budući administrator bi se trebao dobro snalaziti u distribuciji koju ćete izabrati**

Red Hat

Debian

Red Hat i Debian – najvažnija razlika očituje se korištenju paketnog sustava.

Red Hat koristi .rpm pakete dok Debian koristi .deb pakete.

Spuštanje nepotrebnih servisa

Razni programi i servisi koji na web serveru nisu potrebni, mogu vrlo brzo postati uzrok raznih problema. Uklonite sve servise koji vam nisu nužno potrebni!

Servisi prilikom boot-a mogu biti pokretani na tri načina:

- **Direktno iz init skripti (/etc/rc.d/rc.local, /etc/rc.boot/)**
- **Iz (x)inetd-a (ovo je dosta diskutabilno ali pripada ovdje)**
- **Iz skripti koje koriste SYSV init specifikacije**

Primjer /etc/rc.d/rc.local skripte na Red Hat distribuciji:

```
#!/bin/sh
echo “Hello world”
```

Pandan ovoj skripti na Debian distribuciji je direktorij /etc/rc.boot u koji možete stavljati skripte koje se trebaju izvršavati prilikom boot-a.

Na obje distribucije skripte su po defaultu najčešće prazne jer se Servisi dižu ili iz xinetd-a ili iz init skripti prema SYSV standardu.

Primjer izgleda /etc/xinetd.d direktorija na Red Hat distribuciji:

```
-rw-r--r-- 1 root root 318 Mar  4 12:27 amanda
-rw-r--r-- 1 root root 227 Mar  4 12:27 amandaidx
-rw-r--r-- 1 root root 230 Mar  4 12:27 amidxtape
-rw-r--r-- 1 root root 295 Mar  4 12:27 chargen
-rw-r--r-- 1 root root 315 Mar  4 12:27 chargen-udp
-rw-r--r-- 1 root root 295 Mar  4 12:27 daytime
-rw-r--r-- 1 root root 315 Mar  4 12:27 daytime-udp
-rw-r--r-- 1 root root 287 Mar  4 12:27 echo
-rw-r--r-- 1 root root 306 Mar  4 12:27 echo-udp
-rw-r--r-- 1 root root 317 Feb 21 07:32 rsync
-rw-r--r-- 1 root root 319 Mar  4 12:27 time
-rw-r--r-- 1 root root 315 Mar  4 12:27 time-udp
-rw-r--r-- 1 root root 362 Feb 24 14:28 wu-ftpd
```

Primjer izgleda /etc/xinetd.d/ftp fajla:

```
# default: on
# description: The wu-ftpd FTP server serves FTP connections. It uses
# normal, unencrypted usernames and passwords for authentication.
service ftp
{
        disable                 = no
        socket_type             = stream
        protocol                = tcp
        wait                    = no
        user                    = root
        server                  = /usr/sbin/in.ftpd
        server_args             = -l -a
        log_on_success          += DURATION USERID
        log_on_failure          += USERID
        nice                    = 10
}
```

Ista stvar u nekadašnjem /etc/inetd.conf izgledala bi ovako:

```
ftp     stream     tcp     nowait     root     /usr/sbin/in.ftpd -l -a
```

SYSV init

Izgled /etc/rc.d/ direktorija na Red Hat distribuciji:

```
drwxr-xr-x    2 root     root         4096 Tra 13 23:11 init.d
-rwxr-xr-x    1 root     root         3219 Srp 10  2001 rc
drwxr-xr-x    2 root     root         4096 Ožu 16 13:46 rc0.d
drwxr-xr-x    2 root     root         4096 Ožu 16 13:46 rc1.d
drwxr-xr-x    2 root     root         4096 Ožu 16 13:46 rc2.d
drwxr-xr-x    2 root     root         4096 Ožu 16 13:46 rc3.d
drwxr-xr-x    2 root     root         4096 Ožu 16 13:46 rc4.d
drwxr-xr-x    2 root     root         4096 Ožu 30 16:10 rc5.d
drwxr-xr-x    2 root     root         4096 Ožu 16 13:46 rc6.d
-rwxr-xr-x    1 root     root          240 Stu 27 14:01 rc.local
-rwxr-xr-x    1 root     root        21401 Vel  7 04:44 rc.sysinit
```

Izgled /etc/rc.d/rc3.d direktorija:

```
lrwxrwxrwx    1 root     root           15 Stu 24 23:25 K03rhnsd -> ../init.d/rhnsd
lrwxrwxrwx    1 root     root           15 Stu 24 23:25 K20rwhod -> ../init.d/rwhod
lrwxrwxrwx    1 root     root           13 Stu 28 12:07 K35smb -> ../init.d/smb
lrwxrwxrwx    1 root     root           16 Stu 24 23:27 K74ypxfrd -> ../init.d/ypxfrd
lrwxrwxrwx    1 root     root           15 Stu 24 23:18 S05kudzu -> ../init.d/kudzu
...
```

Chkconfig – program koji olakšava upravljanje init skriptama:

Chkconfig ima tri opcije:

--list daje nam ispis trenutnog stanja servisa

--add služi za dodavanje skripti u SYSV init hierarhiju direktorija

--del služi za brisanje skripti

--level osim dodavanja i brisanja skripti, možete ih i isključivati/uključivati u pojedinim runlevelima

chkconfig –list pokazuje kakvo je stanje sa našim servisima:

sshd	0:off	1:off	2:on	3:on	4:on	5:off	6:off
vncserver	0:off	1:off	2:off	3:off	4:off	5:off	6:off
yppasswdd	0:off	1:off	2:off	3:off	4:off	5:off	6:off
ypserv	0:off	1:off	2:off	3:off	4:off	5:off	6:off
ypxfrd	0:off	1:off	2:off	3:off	4:off	5:off	6:off
arpwatch	0:off	1:off	2:off	3:off	4:off	5:off	6:off
smb	0:off	1:off	2:off	3:off	4:off	5:off	6:off

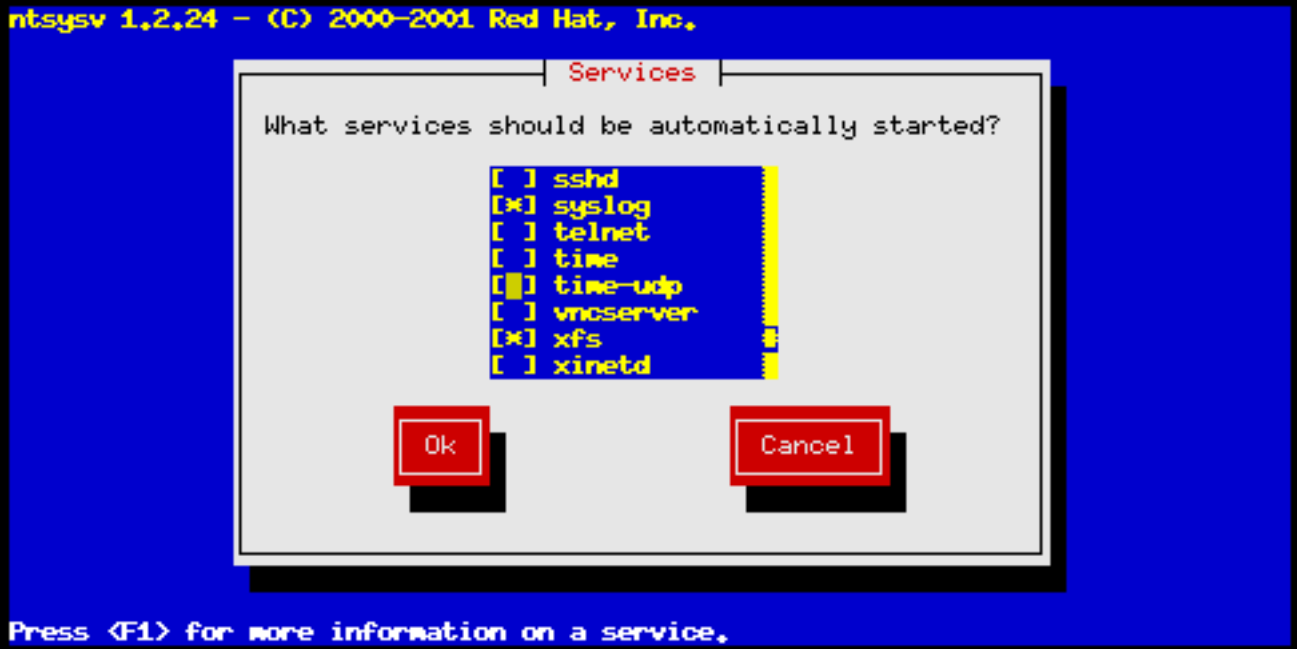
xinetd based services:

daytime: off

echo: off

ftp: off

Izgled alternativnog programa iz kojeg se jos jednostavnije može upravljati sa različitim servisima, uključujući i one koje pripadaju xinetd-u. Radi se o programu ntsysv koji dolazi sa Red Hatom.



Vrijeme za upgrade

RPM je alat za manipulaciju .rpm paketima.

instalacija: rpm –i nešto-1.2.3.i386.rpm

deinstalacija: rpm –e nešto-1.2.3

upgrade: rpm –Fvh nešto-1.2.3.i386.rpm

informacije o paketu: rpm –qi nešto-1.2.3

popis fajlova u paketu: rpm –ql nešto-1.2.3.i386.rpm

popis svih instaliranih paketa: rpm –qa

kompajliranje .i386.src.rpm paketa: rpm –bb nešto-1.2.3.i386.src.rpm

Ako znate što radite ponekad će vam biti korisni i parametri

--nodeps, --force, i --allmatches

Restrikcije.

Što je više restrikcija na serveru to će server biti stabilniji i sigurniji. Po defaultu u gotovo svim distribucijama Linux se ponaša previše liberalno.

gcc, su, sudo, inetd, wall, samo su neke od naredbi koje je dobro odmah zabraniti korisnicima.

Korisnici nemaju razloga gledati vaše logove i konfiguracijske fajlove. Zaštite ih.

Ako u neki direktorij korisnik mora imati pravo ulaziti, ne mora imati i pravo čitanja:

chmod 711 / /home /etc

Primjer pronalaženja suidanih programa:

```
find / -type f \( -perm -04000 -o -perm -02000 \) -ls
```

Nakon što smo pronašli sve suidane programe maknut ćemo ili ćemo zabraniti one koji nam nisu potrebni.
Budite naročito oprezni ako niste sigurni što točno radite i kakvi će biti rezultati.

Iptables

Sa Red Hatom već dolazi firewall program pod imenom iptables.

Odlučite na kojim portovima ćete očekivati konekcije, sve ostale portove jednostavno zabranite.

Ako se nitko ne može spajati na portove koje za to niste namijenili zbunit ćete gotovo sve crackere, i vjerojatno sve postojeće crve koji su uspjeli postaviti shell na nekom portu.

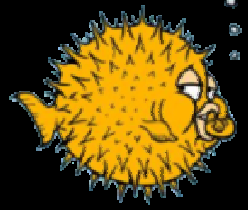
Za zabranjivanje portova i IP adresa uglavnom se koristi skripta u kojoj se iptables poziva sa svojim parametrima.

Najjednostavniji oblik takve skripte izgledao bi ovako:

```
#!/bin/bash
IPTABLES=/usr/local/bin/iptables
PORT_ALLOW=“21 22 25 80 443”
$IPTABLES –F

if [ "$PORT_ALLOW" != "" ] ; then
   echo -n “Dopušteni portovi: "
   for port in ${PORT_ALLOW} ; do
   ${IPTABLES} -A INPUT –p tcp --dport ${port} –j ACCEPT
   echo -n "${port} "
   done
   echo
fi
$IPTABLES –A INPUT –p tcp –m state –state new,invalid –j DROP
```

SSH server



SSH server je jedan od najvažnijih servisa za samog administratora.

Ukoliko vam se ne sviđa verzija koja dolazi sa Red Hatom iz nekog razloga, vaše slijedeće odredište bit će www.openssh.org.

Preporučujem skidanje sourcea s kojim dolazi i openssh.spec fajl koji će nam poslužiti za izradu openssh rpm paketa.

SSH je jedan od kritičnih servisa i zato je uvijek dobro imati najnoviju stabilnu verziju.

Konfiguracija SSH servera

Konfiguracija ssh servera obavlja se u fajlu pod imenom

/etc/ssh/sshd_config

Pozornost u tom fajlu treba obratiti na slijedeće linije:

Default:	Preporučljivo:
Protocol 2,1	Protocol 2
PermitRootLogin no	PermitRootLogin yes
RhostsAuthentication yes	RhostAuthentication no
HostbasedAuthentication no	HostbasedAuthentication no
X11Forwarding yes	X11Forwarding no
UseLogin no	UseLogin no

ProFTPD server

Po defaultu sa RedHatom dolazi WU-FTPD ftp server.

Kako smatram da komfor nikada ne smije ići na štetu sigurnosti umjesto wu-ftpd-a postavit ćemo proFTPD.

Kao i u slučaju ssh, možemo opet otići na site www.proftpd.org

I tamo skinuti source trenutne stabilne verzije proftpd-a.

Kompajiranje i instaliranje izvodi se isto kao i openssh, iz priloženog .spec fajla.

Konfiguracija ProFTPD servera

```
ServerName                      "FTP server"
ServerIdent                     on "FTP server“
ServerType                      standalone
DefaultServer                   on
DefaultRoot                     ~
ShowDotFiles                    on
ShowSymlinks                    on
DeferWelcome                    on
UseReverseDNS                   off
IdentLookups                    off
RequireValidShell               yes
MaxInstances                    64
RootLogin                       off
UseFtpUsers                     on
Umask                           022
PidFile                         /var/run/proftpd.pid
User                            proftpd
Group                           proftpd
AllowRetrieveRestart            on
AllowStoreRestart               on
AllowOverwrite                  on
LsDefaultOptions                "-a"
PathAllowFilter                 "^[a-zA-Z0-9 /._-]+$“
PathDenyFilter                  "%" ^-
TimeoutLogin                    120
TimeoutIdle                     900
TimeoutNoTransfer               900
TimeoutStalled                  3600
ExtendedLog                     /var/log/proftpd
<Limit ALLOW_CHMOD WRITE>
  AllowAll
</Limit>
```

MySQL

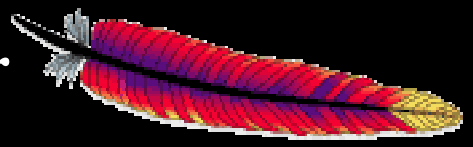
MySQL je relacijska baza pogodna za web.

Velik broj siteova danas je baziran upravo na MySQL-u zbog njegove brzine i velikog broja skripti koje ga podržavaju.

Na www.mysql.org mogu se pronaći rpm paketi sa svježim verzijama MySQL-a. Nakon instalacije MySQL-a potrebno je učiniti ono što sama instalacija preporučuje – postaviti root password.

Uvijek izbjegavajte neselektivno dozvoljavanje konekcija na vaš MySQL server sa drugih hostova.

Apache web server



Web server s pokrićem. Ovaj web server ima dugu povijest i pošteno je zaradio titulu najraširenijeg web servera u svijetu.
Solidan web server koji može sve što je administratoru potrebno, a što ne može, postoje moduli koji nadopunjavaju praznine.

Vjerojatno sve funkcije na koje ste navikli na nekim drugim web serverima mogu biti implementirane i kod Apachea.

S obzirom da je sam Apache toliko razrađen softvare, skinut ćemo sa adrese www.apache.org cijeli source i sami odlučiti koji dijelovi Apachea su nam potrebni.

Što je sve potrebno za kompajliranje Apache web servera?

Kao prvo nabavit ćemo slijedeće pakete:

- **php (source)**
- **mod_ssl (potrebno za ssl ali mi ga u primjerima nećemo koristiti)**
- **apache 1.3.x**
- **mod_throttle**
- **mod_gzip**

Nakon što sve te pakete raspakiramo u jednom zajedničkom direktoriju, možemo pristupiti kompajliranju Apachea. Prije svega postavit ćemo sve direktorije modula unutar apache-a u direktorij src/modules.

Kao prvo moramo pripremiti apache prije svega ostaloga.

Dakle potrebno je ući u apache direktorij i izdati naredbu:

```
./configure
```

Izlazimo iz apache direktorija i ulazimo u php direktorij.

U php direktoriju izdajemo slijedeću naredbu:

```
./configure --with-config-file-path=/usr/local/apache/config --with-zlib      \
--with-jpeg-dir=/usr --with-tiff-dir=/usr --with-png-dir=/usr --with-ttf       \
--with-mysql=/usr --enable-track-vars --enable-memory-limit                    \
--with-mhash=/usr --with-apache=../apache --with-pdflib=/usr                   \
--with-gd=/usr --enable-gd-native-ttf --with-ttf=/usr
make
make install
```

Sada ulazimo u apache direktorij i izdajemo redom slijedeće naredbe:

```
export LIBS="-lwrap"                                                    \
./configure --with-layout=Apache --prefix=/usr/local/apache             \
--enable-module=headers --enable-module=expires                         \
--with-perl=/usr/bin/perl --activate-module=src/modules/php4/libphp4.a  \
--enable-module=rewrite --add-module=src/modules/mod_gzip/mod_gzip.c    \
--add-module=src/modules/mod_throttle/mod_throttle.c
make
make install
```

I naš Apache web server je spreman za pokusnu vožnju.

Prije samog pokretanja apache web server treba konfigurirati i pripremiti za pokretanje.

Ako želimo da se apache pokreće kao neprivilegirani korisnik, na primjer httpd, moramo kreirati grupu i korisnika httpd.

To možemo učiniti na slijedeći način:

```
groupadd httpd
adduser -g httpd -d /dev/null -s /dev/null -c Apache -M httpd
```

Ako želimo koristiti mod_gzip koji smo ukompajlirali u apachea onda moramo napraviti i direktorij koji će biti korišten za odlaganje privremenih fajlova koji nastaju prilikom kompresije podataka. U taj direktorij apache mora imati pravo pristupa pa se možemo poslužiti slijedećim nizom naredbi:

```
mkdir /usr/local/apache/tmp
chgrp root.httpd /usr/local/apache/tmp
chmod 730 /usr/local/apache/tmp
```

Konfiguracija Apache web servera uglavnom se vrši iz fajla

/usr/local/apache/etc/httpd.conf.

Sada ćemo obraditi konfiguraciju liniju po liniju.

```
# /usr/local/apache/etc/httpd.conf
ServerType standalone
PidFile logs/httpd.pid
ResourceConfig /dev/null
AccessConfig /dev/null
ServerSignature off
ServerTokens ProductOnly
HostnameLookups off
ExtendedStatus on
UseCanonicalName on
Timeout 300
KeepAlive on
MaxKeepAliveRequests 512
KeepAliveTimeout 5
MinSpareServers 4
MaxSpareServers 12
StartServers 12
MaxClients 256
MaxRequestsPerChild 2048
```

```
DirectoryIndex index.html index.shtml index.htm index.php index.php3
AccessFileName .htaccess
 Port 80
 User httpd
 Group httpd
 ServerRoot /usr/local/apache
 DocumentRoot /usr/local/apache
 TypesConfig /usr/local/apache/conf/mime.types
 MIMEMagicFile /usr/local/apache/conf/magic

LogFormat "%v %h %l %u %t \"%r\" %>s %b \"%{Referer}i\" \"%     \
{User-Agent}i\"" common
LogFormat "%h %l %u %t \"%r\" %>s %b" common
LogFormat "%h %u %t \"%r\" %>s %b" common
LogFormat "%{Referer}i -> %U" referer
LogFormat "%{User-agent}i" agent
CustomLog logs/custom_log combined

 <Location /server-status>
       SetHandler server-status
       order deny,allow
       deny from all
       allow from localhost
 </Location>
```

```
<Directory />
        AllowOverride None
        Order Deny,Allow
        Deny from All
</Directory>

<Directory /home/sites>
        Options +IncludesNOEXEC -Indexes
        XBitHack off
        AllowOverride AuthConfig FileInfo Indexes Options Limit
        Order Deny,Allow
        Allow from all
</Directory>

IndexIgnore .??* *~ *# HEADER* README* RCS CVS *,v *,t
<Files "\.*">
        Order allow,deny
        deny from all
</Files>
```

```
<IfModule mod_gzip.c>
    mod_gzip_on Yes
    mod_gzip_min_http 1001
    mod_gzip_minimum_file_size 256
    mod_gzip_maximum_file_size 1000000
    mod_gzip_maximum_inmem_size 10000000
    mod_gzip_keep_workfiles No
    mod_gzip_temp_dir /usr/local/apache/tmp
    mod_gzip_can_negotiate Yes
    mod_gzip_item_include file \.cgi$
    mod_gzip_item_include file \.pl$
    mod_gzip_item_include file \.htm$
    mod_gzip_item_include file \.html$
    mod_gzip_item_include mime text/.*
    mod_gzip_item_include mime httpd/unix-directory
    mod_gzip_item_include handler ^perl-script$
    mod_gzip_item_include handler ^server-status$
    mod_gzip_item_include mime “application/x-httpd-php.*”
    mod_gzip_item_include file \.php$
    mod_gzip_item_include file “\.css$“
    mod_gzip_item_exclude file “\.js$“
    mod_gzip_item_exclude file “\.wml$”
    mod_gzip_item_exclude mime ^image/.*
</IfModule>
```

```
Alias /icons /usr/local/apache/icons

NameVirtualHost 192.168.1.1

<VirtualHost 192.168.1.1>
ServerName www.domena.tld
ServerAlias domena.tld
ServerAdmin webmaster@domena.tld
RewriteEngine on
RewriteCond %{HTTP_HOST}      !^192.168.1.1(:80)?$
RewriteCond %{HTTP_HOST}      !^www.domena.tld(:80)?$
RewriteRule ^/(.*)            http://www.domena.tld/$1 [L,R]
RewriteOptions inherit
Alias /icons/ /usr/local/apache/icons/
DocumentRoot /home/sites/site145/web
</VirtualHost>
```

Apache je moćan web server sa gomilom opcija. Evo nekoliko primjera:

Omogućavanje prikazivanja wap stranica

```
AddType text/vnd.wap.wml .wml
AddType image/vnd.wap.wbmp .wbmp
AddType application/vnd.wap.wmlc .wmlc
AddType text/vnd.wap.wmlscript .wmls
AddType application/vnd.wap.wmlscriptc .wmlsc
```

Podrška za php stranice:

```
AddType application/x-httpd-php .php .php3 .phtml
```

Isključivanje cache servera:

```
ExpiresActive On
ExpiresDefault "access 0 seconds"
```

SSI, CGI skripte:

```
AddHandler cgi-script .cgi .pl
AddType text/html .shtml
AddHandler server-parsed .shtml
```

Nakon što smo konfigurirali naš Apache web server možemo se upoznati s naredbom /usr/local/apache/bin/apachectl.
Ova nareba ima nekoliko korisnih parametara:

start - start httpd
stop - stop httpd
restart - restart httpd if running
fullstatus - dump a full status screen; requires lynx and mod_status enabled
status - dump a short status screen; requires lynx and mod_status enabled
graceful - do a graceful restart by sending a SIGUSR1 or start if not running
configtest - do a configuration syntax test

Prije samog startanja apachea možemo provjeriti ima li kakvih sintaksnih pogrešaka u našem konfiguracijskom fajlu:
/usr/local/apache/httpd/bin/apachectl configtest

Ako je sve u redu možemo startati naš apache naredbom:
/usr/local/apache/httpd/bin/apachectl start

Bude li potrebno restartati apache da bi povukao nove opcije iz konfiguracijskog fajla, to možemo učiniti naredbom "/usr/local/apache/bin/apachectl graceful"

Za kraj kad smo se uvjerili da nas naš apache dobro služi, moramo se pobrinuti da se apache automatski digne nakon eventualnog reboota.
To ćemo učiniti tako što ćemo uz pomoć programa chkconfig dodati apachectl u /etc/rc.d direktorije.

Kao prvo kopirat ćemo fajl /usr/local/apache/bin/apachectl u /etc/rc.d/init.d.
Zatim ćemo na vrh fajla ali ispod "#!/bin/sh" linije dodati slijedeće linije:

```
# chkconfig: 3 80 15
# description: Apache control script designed to allow an easy command line \
#                    interface to controlling Apache.
# processname: apachectl
```

Nakon što smo editirali naš /etc/rc.d/init.d/apachectl možemo pokrenuti chkconfig na slijedeći način:

```
chkconfig --add /etc/rc.d/init.d/apachectl
```