

vps

Info

Na serveru běží veřejné služby (web pro java aplikace) a privátní služby přes vpn (ssh, redmine, git, maven repositář). Zabezpečení je postaveno na firewalu, který blokuje všechno kromě veřejných služeb a vpn.

Nainstalovaný systém je **debian 7 (wheezy)**. Původně jsem zkoušel debian 6, ale nefungoval v něm shorewall. Pak to běželo na arch linuxu, ale ten není od vpsfree moc podporovaný a navíc má rolling-updates, takže obsahují i hodně velké změny (upgrade glibc, init systému apod.), což může lehce všechno rozjet do stavu, kdy se to musí komplet přeinstalovat.

Základ

Aktualizace systému

```
apt-get update          # nahraje info o aktualnich verzich
apt-get upgrade         # upgraduje baliky na nejnovější verze
```

Základní balíky a nastavení

```
apt-get install rsyslog man bzip2 wget sudo htop cron-apt

# Oracle Java:
# je potřeba java-package 0.50+ kuli podpore server-jre, tohle je lepší než
# povolovat backports repositář
wget
http://ftp.cz.debian.org/debian/pool/contrib/j/java-package/java-package_0.5
3~bpo70+1_all.deb
dpkg -i java-package_0.53~bpo70+1_all.deb
wget --no-check-certificate --no-cookies --header "Cookie:
oraclelicense=accept-securebackup-cookie" \
http://download.oracle.com/otn-pub/java/jdk/7u55-b13/server-jre-7u55-linux-x
64.tar.gz
make-jpkg server-jre-7u55-linux-x64.tar.gz
dpkg -i oracle-java7-jre_7u55_amd64.deb
```

/etc/ssh/sshd_config

Zkopírovat klic na přihlášení např. ssh-copy-id root@example.com, zkontrolovat, že to funguje, pak zakázat login s heslem:

```
PasswordAuthentication no
```

/etc/vim/vimrc

```
set mouse-=a
colorscheme elflord
syntax on
```

/etc/cron-apt/config

```
MAILON="upgrade"
MAILTO="user@example.com"
```

Firewall

Nastavení firewallu se dělá pomocí balíku *shorewall*, detaily viz. <http://shorewall.net/standalone.htm>, <https://wiki.debian.org/HowTo/shorewall>.

```
apt-get install shorewall
cd /etc/shorewall
# adresar by mel byt prazdny, krome shorewall.conf
```

/etc/shorewall/zones

Nastavení zón (\$FW v ostatních souborech se automaticky nahrazuje „fw“).

#ZONE	TYPE	OPTIONS	IN	OUT
#			OPTIONS	OPTIONS
fw	firewall			
net	ipv4			
vpn	ipv4			

/etc/shorewall/policy

Tohle je nastavení implicitních akcí (vyhodnocuje se v zadaném pořadí!).

#SOURCE	DEST	POLICY	LOG	LIMIT:	CONNLIMIT:
#			LEVEL	BURST	MASK
# povol spojeni "ze serveru na internet"					
\$FW	net	ACCEPT			
# zahod vsechno "z internetu na server"					
net	all	DROP	info		
# odmitni vsechno "z vpn na internet" (aby si vpn klienti nebrouzdali pres server)					
vpn	net	REJECT	info		
# povol vsechno ostatni "z vpn"					
vpn	all	ACCEPT			

```
# The FOLLOWING POLICY MUST BE LAST
all          all          REJECT          info
```

/etc/shorewall/interfaces

```
FORMAT 2
#####
###
#ZONE          INTERFACE          OPTIONS
net            venet0              tcpflags,logmartians,nosmurfs
vpn            tun0
```

/etc/shorewall/rules

```
#ACTION SOURCE          DEST          PROTO DEST          SOURCE          ORIGINAL
RATE    USER/   MARK    CONNLIMIT    TIME    HEADERS          PORT          SWITCH          PORT(S)          DEST
#
LIMIT   GROUP
#SECTION ALL
#SECTION ESTABLISHED
#SECTION RELATED
SECTION NEW

# povoleni SSH sluzby pro klienty z internetu (NEDELAT, v pripade nouze se
# lze pripojit k terminalu pres administraci VPS)
# - pro vsechny
#ACCEPT net          $FW          tcp          ssh
# - pro urcitou IP adresu
#ACCEPT net:78.80.8.27 $FW          tcp          ssh
# - pro skupinu IP adres (subnet)
#ACCEPT net:81.25.21.0/24 $FW          tcp          ssh

# OpenVPN
ACCEPT net          $FW          udp          1194
ACCEPT $FW          net          udp          -          1194

# WEB
ACCEPT all          all          tcp          80
ACCEPT all          all          tcp          443
```

/etc/shorewall/shorewall.conf

```
STARTUP_ENABLED=Yes
```

/etc/default/shorewall

```
startup=1
```

Pár užitečných příkazů:

```
/etc/init.d/shorewall start|stop|restart|...
shorewall status
shorewall show
shorewall safe-start
shorewall safe-restart
```

OpenVPN

```
apt-get install openvpn
cp -a /usr/share/openvpn/easy-rsa /etc/openvpn
cd /etc/openvpn/easy-rsa
```

/etc/openvpn/easy-rsa/vars

```
export KEY_SIZE=2048
export KEY_COUNTRY="CZ"
export KEY_PROVINCE="Czech Republic"
export KEY_CITY="Prague"
export KEY_ORG="MOJE FIRMA s.r.o."
export KEY_EMAIL="support@example.com"
export KEY_OU=""
```

```
source vars
./clean-all
./build-ca # zadat např. openvpn-ca jako Common Name/Name
./build-key-server mujserver
./build-key tonda # nebo build-key-pass pro zaheslovani privatnich klicu
./build-key cenda
...
./build-dh
cd keys
openvpn --genkey --secret ta.key
cp {ca.crt,dh2048.pem,ta.key,inter.{crt,key}} /etc/openvpn
chmod 600 /etc/openvpn/{ta.key,inter.key}
```

/etc/openvpn/server.conf

```
dev tun
port 1194
;proto tcp
proto udp
# VPN subnet - vybrat neco nahodnyho z
http://en.wikipedia.org/wiki/Private\_network#Private\_IPv4\_address\_spaces
# urcite ne 10.0.0.0, 10.1.1.0, 192.168.0.0, 192.168.1.0 - to pouziva
vetsina "domacich" siti
server 10.134.75.0 255.255.255.0
ifconfig-pool-persist ipp.txt
ca ca.crt
crl-verify crl.pem # viz. revokace certifikatu
```

```
cert inter.crt
key inter.key
dh dh2048.pem
tls-auth ta.key 0
cipher AES-256-CBC
comp-lzo yes
```

client.conf

```
dev tun
port 1194
proto udp
client
remote mujserver.example.com
ca ca.crt
cert tonda.crt
key tonda.key
tls-auth ta.key 1
remote-cert-tls server
cipher AES-256-CBC
comp-lzo yes
```

Teď už je třeba jenom poslat každému klientovi `client.conf`, `ta.key` a odpovídající `crt` a `key` soubor. **Doporučuje se přesunout `ca.key` na offline úložiště a odstranit key soubory všech klientů.**

```
# predpoklada nastaveni sendmailu (dale v navodu)
cd keys
key="tonda" email="tonda@example.com"
zippwd=$(dd if=/dev/urandom bs=1 count=10 2>/dev/null | base64 | head -c 8)
rm -v $key.7z; 7z a -p $zippwd ca.crt $key.{crt,key} ta.key && mailx -s
"openvpn keys" -a $key.7z $email <<<"heslo k archivu dodam"; rm -v $key.7z
echo "heslo na rozbalení $key.7z: $zippwd"
```

Revokace certifikátů

```
cd /etc/openvpn/easy-rsa
source vars
./revoke-full jmeno_certifikátu
cp -v crl.pem /etc/openvpn
```

sendmail interface pro SMTP server

Některé komponenty (např. redmine) potřebují posílat emaily přes sendmail interface (např. jejich SMTP klient z nějakého důvodu nefunguje se SMTP serverem). Proto se dá nainstalovat lepší SMTP klient, který podporuje sendmail interface. Detaily viz. <http://msmtp.sourceforge.net/doc/msmtp.html>.

```
apt-get purge exim4-config exim4 exim4-base exim4-daemon-light
```

```
apt-get install msmtplib
ls -l /usr/sbin/sendmail
# musi ukazovat na /usr/msmtplib
```

/etc/msmtplib

```
# Accounts will inherit settings from this section
defaults
auth                on
tls                 on
tls_certcheck      off
#tls_trust_file     /usr/share/ca-
certificates/mozilla/Thawte_Premium_Server_CA.crt

account            blackhole
host               smtp.example.com
port               465
from               blackhole@example.com
user               blackhole@example.com
password           my_password
tls_starttls      off

account default : blackhole
```

web server

Nginx

Nginx krom jiného umožňuje provozovat více různých web serverů na stejném portu (např. tomcat pro java web aplikace + apache pro php + passenger pro ruby aplikace).

Protože potřebuji **passenger** pro **ruby** aplikace (např. **redmine**), neda se to instalovat z debianich balicku.

```
apt-key adv --keyserver keyserver.ubuntu.com --recv-keys 561F9B9CAC40B2F7
apt-get install apt-transport-https ca-certificates
echo "deb https://oss-binaries.phusionpassenger.com/apt/passenger wheezy
main" > /etc/apt/sources.list.d/passenger.list
chmod 600 /etc/apt/sources.list.d/passenger.list
apt-get update
apt-get install nginx-extras passenger
```

Pokud se bude používat SSL, tak je potřeba vygenerovat certifikát:

```
openssl req -new -x509 -nodes -out /etc/nginx/server.crt -keyout
/etc/nginx/server.key
```

/etc/nginx/conf/nginx.conf

```
#user nobody;
worker_processes 1;

error_log /var/log/nginx/error.log;
pid /var/run/nginx.pid;

#error_log logs/error.log notice;
#error_log logs/error.log info;

#pid logs/nginx.pid;

events {
    worker_connections 128; # maximalni pocet spojeni -
    http://wiki.nginx.org/EventsModule#worker_connections
}

http {
    passenger_root
    /usr/lib/ruby/vendor_ruby/phusion_passenger/locations.ini;
    passenger_ruby /usr/bin/ruby;

    include mime.types;
    default_type application/octet-stream;

    #log_format main '$remote_addr - $remote_user [$time_local] "$request"
    ,
    # '$status $body_bytes_sent "$http_referer" '
    # '"$http_user_agent" "$http_x_forwarded_for"';

    #access_log logs/access.log main;

    sendfile on;
    #tcp_nopush on;

    #keepalive_timeout 0;
    keepalive_timeout 65;

    #gzip on;

    ssl_certificate server.crt;
    ssl_certificate_key server.key;

    proxy_set_header X-Real-IP $remote_addr;
    proxy_set_header X-Forwarded-For $proxy_add_x_forwarded_for;
    proxy_set_header Host $http_host;
}
```

Tomcat

Web server je tomcat 7, protožev něm chceme provozovat jednoduchý javovský web aplikace (tzn. potřebujeme něco v javě, ale nepotřebujeme super-druper aplikační server).

```
apt-get install tomcat7
```

conf/server.xml

```
<Server port="8005" shutdown="SHUTDOWN">
  <Service name="Catalina">
    <Connector port="8081"
protocol="org.apache.coyote.http11.Http11NioProtocol"
      connectionTimeout="20000"
      redirectPort="443"
      minSpareThreads="2" maxThreads="10" />
    <Engine name="Catalina" defaultHost="www.example.com">
      <Host name="www.example.com" appBase="webapps-moje"
        unpackWARs="true" autoDeploy="true">
        <Valve className="org.apache.catalina.valves.AccessLogValve"
directory="logs"
          prefix="access_log." suffix=".log"
          pattern="%h %l %u %t &quot;%r&quot; %s %b" />
        </Host>
      </Engine>
    </Service>
  </Server>
```

appBase je zmenena, protoze upgrade tomcatu by mohl prepsat aplikace ve /var/lib/tomcat7/webapps (minimalne nektery distribuce to delaly).

/etc/default/tomcat7

```
JAVA_HOME=/usr/lib/jvm/jre-7-oracle-x64
CATALINA_OPTS=-Djava.awt.headless=true -Xmx80m -XX:+UseConcMarkSweepGC
# povolit pro remote management (napr. jconsole nebo jvisualvm)
#JAVA_OPTS="${JAVA_OPTS} -Djava.rmi.server.hostname=mujserver.example.com -
Djava.net.preferIPv4Stack=true -Dcom.sun.management.jmxremote.ssl=false -
Dcom.sun.management.jmxremote.port=5000 -
Dcom.sun.management.jmxremote.authenticate=false"
```

Nastavit nginx, aby pozadavky preposilal na tomcat:

/etc/nginx/conf/nginx.conf

```
server {
  # JAVA web server - treba Tomcat
  listen *:80 default_server;
  listen *:443 ssl;
```



```
proxy_set_header X-Real-IP $remote_addr;
proxy_set_header X-Forwarded-For $proxy_add_x_forwarded_for;
proxy_set_header Host $http_host;

location / {
    proxy_pass http://127.0.0.1:8081;
}
}
```

Apache + PHP

Pro PHP experimenty:

/etc/nginx/conf/nginx.conf

```
server {
    # PHP + phpmyadmin
    listen *:80;
    listen *:443 ssl;
    server_name php.example.com; # tohle je dalsi DNS jmeno pro
verrejnou adresu vps serveru

    proxy_set_header X-Real-IP $remote_addr;
    proxy_set_header X-Forwarded-For $proxy_add_x_forwarded_for;
    proxy_set_header Host $http_host;

    location / {
        proxy_pass http://127.0.0.1:8082;
    }

    # PHPmyadmin jenom pres SSL
    location /phpmyadmin {
        if ($scheme = "http") {
            rewrite ^ https://$http_host$request_uri permanent;
        }
        if ($scheme = "https") {
            proxy_pass http://127.0.0.1:8082;
        }
    }
}
```

Git

Přístup k repozitářům gitů řídí **gitolite** .

```
# zkopirovat id_rsa.pub spravce gitů do /root/spravcegitů.pub
apt-get install gitolite
dpkg-reconfigure gitolite
```

```
# zmenit user na git
```

/var/lib/gitolite/.gitolite.rc

```
$REPO_UMASK = 0027; # nastavi soubory g+rx, aby k tomu mel pristup napr. redmine
```

/etc/ssh/sshd_config

Zakáže se autentikace heslem (všechno běží pouze přes certifikáty):

```
Match User git
PasswordAuthentication no
```

Mysql

Mysql je potřeba např. pro redmine (viz. níže). Více na <https://wiki.archlinux.org/index.php/MySQL> .

```
apt-get install mysql-server
mysql_secure_installation
```

Redmine

Podrobnosti viz. <http://www.redmine.org/projects/redmine/wiki/RedmineInstall> .

```
apt-get install ruby ruby-dev make imagemagick libmagickcore-dev
libmagickwand-dev libmysqlclient-dev
cd
VER=2.5.1
wget http://www.redmine.org/releases/redmine-$VER.tar.gz
tar xzf redmine-$VER.tar.gz -C /opt
chown -R root:root /opt/redmine-$VER
```

```
mysql -p # zepta se na heslo (viz. instalace mysql)
create database redmine character set utf8;
create user 'redmine'@'localhost' identified by 'my_password';
grant all privileges on redmine.* to 'redmine'@'localhost';
```

config/database.yml

```
production:
  adapter: mysql2
  database: redmine
  host: localhost
  username: redmine
  password: my_password
```

```
encoding: utf8
```

config/configuration.yml

```
production:
  email_delivery:
    delivery_method: :sendmail
```

Tohle je potreba udelat az po config/database.yml , aby to nahralo vsechny potrebny doplnky (hlavne teda ty na pristup k databazi).

```
cd /opt/redmine-$VER
gem install --no-user-install bundler
bundle install --system --without development test postgresql sqlite
rake generate_secret_token
useradd -m --home-dir /var/lib/redmine-$VER --shell /bin/bash --system
redmine
usermod -a -G git redmine
mkdir -p /var/lib/redmine-$VER/{tmp,public/plugin_assets}
tar c files log tmp public/plugin_assets | tar xv -C /var/lib/redmine-$VER
for i in files log tmp public/plugin_assets; do rm -Rf $i; ln -nfs
/var/lib/redmine-$VER/$i $i; done
chown -R redmine:redmine /var/lib/redmine-$VER
chmod -R ugo+r /var/lib/redmine-$VER
```

Zkopírují se data ze starého serveru:

```
# nejak dostat data z files do /var/lib/redmine-1.4/files
mysql -u redmine -p redmine < dump_redmine_default_2012-05-28.sql | tee
restore.log
RAILS_ENV=production rake db:migrate
```



NOTE: Novou databázi lze vytvořit pomocí:

```
RAILS_ENV=production rake db:migrate
RAILS_ENV=production rake redmine:load_default_data
```



Instalaci lze otestovat spuštěním jednoduchého web serveru (podívat se na projekty a jestli funguje integrace s gitem a posílání emailů):

```
su - -s /bin/bash redmine
ruby script/rails server webrick -e production
```

Passenger v nginx

Detaily viz.

http://www.modrails.com/documentation/Users%20guide%20Nginx.html#install_on_debian_ubuntu .

```
apt-get install ruby-passenger
```

/etc/nginx/conf/nginx.conf

```
http {
    # P0Z0R: musi byt zapnuty passenger (viz. instalace nginx)

    server {
        listen 8080 default_server;
        root /opt/redmine-2.5.1/public;
        passenger_enabled on;
        # implicitne se pouzije aktualni owner/group souboru
        config/environment.rb
        passenger_user redmine;
        passenger_group redmine;
        client_max_body_size 100M; # nektere uploady do redmine budou vetsi nez
        default limit
    }
}
```

Thin v nginx (primitivni alternativa k passengeru)

```
gem install --no-user-install thin
thin install
```

Pridat nasledujici:

/opt/redmine-1.4/Gemfile

```
gem 'thin'
```

/etc/thin/redmine.yml

```
# comment
---
chdir: /opt/redmine-1.4
environment: production
timeout: 30
log: /var/log/thin/redmine.log
pid: /var/lib/redmine-1.4/thin.pid # musi byt zapisovatelny userem redmine
max_conns: 1024
max_persistent_conns: 100
require: []
wait: 30
socket: /var/lib/redmine-1.4/thin.sock # musi byt zapisovatelny userem
redmine
daemonize: true
```

```
user: redmine
group: redmine
servers: 1
```

A nakonec v `/etc/rc.conf` přidat `thin` do `DAEMONS`.

/etc/nginx/conf/nginx.conf

```
upstream redmine {
    server unix:/var/lib/redmine-1.4/thin.0.sock;
}

server {
    listen *:8080 default_server;
    client_max_body_size 100M;

    location / {
        proxy_pass http://redmine;
    }
}
```

nexus (maven repository)



NOTE: Možná by stalo za uvahu jenom hodit war do tomcatu, at tam zbytečne nejede 2x JVM. Ale bacha, tomcat je videt z internetu, my chceme nexus jenom na vpn.



```
useradd --system --shell /bin/bash --home-dir /var/lib/nexus -m nexus
wget http://www.sonatype.org/downloads/nexus-latest-bundle.tar.gz
tar xzf nexus-latest-bundle.tar.gz -C /opt
ln -nfsv /opt/nexus-2.7.0-05 /opt/nexus
mkdir /var/run/nexus
chown nexus:nexus /var/run/nexus
mkdir /var/lib/nexus/{logs,tmp}
chown nexus:nexus /var/lib/nexus/{logs,tmp}
rm -rfv /opt/nexus/{logs,tmp}
ln -fsv /var/lib/nexus/logs /opt/nexus
ln -fsv /var/lib/nexus/tmp /opt/nexus
cp /opt/nexus/bin/nexus /etc/init.d
chmod ugo+x /etc/init.d/nexus
update-rc.d nexus defaults
```

/etc/init.d/nexus

```
NEXUS_HOME="/opt/nexus"
#JAVA_HOME="/opt/jdk-7"
RUN_AS_USER="nexus"
```

```
PIDDIR="/var/lib/nexus" # musi byt writeable uzivatelem nexus
```

/opt/nexus/conf/nexus.properties

```
application-port=8083  
nexus-work=/var/lib/nexus
```

/opt/nexus/bin/jsw/conf/wrapper.conf

```
wrapper.java.maxmemory=80
```

Zbytek viz. <http://books.sonatype.com/nexus-book/reference/install-sect-repoman-post-install.html>

From:

<https://kb.vpsfree.cz/> - **Znalostní Báze**

Permanent link:

https://kb.vpsfree.cz/navody/uzivatele/stepan_schejbal

Last update: **2015/04/08 08:52**