

# Usenet Download Server

## Basic System Setup

```
# yum update && yum install nfs-utils
# systemctl start rpcbind nfslock && systemctl enable rpcbind nfslock
# curl -sSL
https://www.blackgate.org/deployment/base_configuration_blackGATE.sh | bash
# vim /etc/motd
# mkdir /mnt/blackserv

# vim /etc/fstab
```

```
#
# /etc/fstab
# Created by anaconda on Wed Dec 6 13:07:04 2017
#
# Accessible filesystems, by reference, are maintained under '/dev/disk'
# See man pages fstab(5), findfs(8), mount(8) and/or blkid(8) for more info
#
/dev/mapper/centos_rhdl03-root / xfs defaults
0 0
UUID=a34686fa-4b05-43ed-b3bb-c1d754783bc2 /boot xfs
defaults 0 0
UUID=D653-2F5A /boot/efi vfat
umask=0077,shortname=winnt 0 0
/dev/mapper/centos_rhdl03-opt /opt xfs defaults
0 0
/dev/mapper/centos_rhdl03-swap swap swap defaults
0 0

10.0.0.21:/volume1/media /mnt/blackserv nfs rw,hard,intr 0 0
```

```
# mount -a
```

## Webserver / Application Proxy Setup

```
# yum -y install https://mirror.webtatic.com/yum/el7/webtatic-release.rpm
# yum update
# yum install -y wget git tar vim curl policycoreutils-python httpd php71w
php71w-dom php71w-mbstring php71w-gd php71w-json \
php71w-xml php71w-zip php71w-curl php71w-mcrypt php71w-pear php71w-opcache
```

```
# sed -i 's/^\#&/g' /etc/httpd/conf.d/welcome.conf
# sed -i 's/DocumentRoot "\/var\www\html"/DocumentRoot
"\/var\www\html\muximum"/' /etc/httpd/conf/httpd.conf

# mkdir /var/www/html/muximum
# git clone https://github.com/mescon/Muximum.git /var/www/html/muximum/
# chown -R apache:apache /var/www/html/muximum/
# chmod -R 775 /var/www/html/muximum/
# ls -l /var/www/html/muximum/

# vim /etc/httpd/conf.d/usenet_proxy.conf
```

```
ServerName www.blackgate.org
ServerTokens Prod
ServerSignature Off
```

```
<Directory /var/www/html/muximum>
    Options +FollowSymlinks
    AllowOverride All
    # Allow open access:
    Require all granted
</Directory>
```

```
<Location /nzbget/>
    ProxyPass http://127.0.0.1:6789/
    ProxyPassReverse http://127.0.0.1:6789/
</Location>
```

```
<Location /nzbhydra/>
    ProxyPass http://127.0.0.1:5075/
    ProxyPassReverse http://127.0.0.1:5075/
</Location>
```

```
<Location /sonarr/>
    RewriteEngine On
    RewriteRule /sonarr/(.*)
http://127.0.0.1:8989/usenetdl/sonarr/$1 [P]
    ProxyPassReverse http://127.0.0.1:8989/
</Location>
```

```
<Location /radarr/>
    ProxyPassReverse http://127.0.0.1:7878/
    RewriteEngine On
    RewriteRule /radarr/(.*)
http://127.0.0.1:7878/usenetdl/radarr/$1 [P]
</Location>
```

```
<Location /lidarr/>
    ProxyPassReverse http://127.0.0.1:8686/
    RewriteEngine On
    RewriteRule /lidarr/(.*)
```

```
http://127.0.0.1:8686/usenetdl/lidarr/$1 [P]
</Location>
```

```
Header set X-Content-Type-Options: "nosniff"
Header set X-Frame-Options: "sameorigin"
```

```
# setsebool httpd_can_network_connect 1 -P
# setsebool -P httpd_execmem 1
# semanage fcontext -a -t httpd_sys_rw_content_t
'/var/www/html/muximux(/.*)?'
# semanage fcontext -a -t httpd_sys_rw_content_t '/var/www/html/muximux'
# restorecon -Rv /var/www/html/
# ls -lZ /var/www/html/

# systemctl start httpd && systemctl status httpd
# systemctl enable httpd

# firewall-cmd --permanent --zone=public --add-service=http
# firewall-cmd --permanent --zone=public --add-service=https
# firewall-cmd --reload
```

**Achtung:** Für dieses Setup werden zwei Reverse Proxies gebraucht. Dieser Proxy, den wir gerade eingerichtet haben, proxyt ausschliesslich die Anfragen von zweiter Stelle (den ServiceNamen) auf das Backend. Zum proxen aller Services unter dem Sub-Verzeichniss “/usenetdl” wird der haupt Reverse Proxy (mit TLS-Termination) benötigt

Beispiel Haupt-Proxy Stanza:

```
...
RewriteEngine on
RewriteRule ^/usenetdl$ /usenetdl/ [R]

ProxyPass          /usenetdl/ http://172.168.0.11/
ProxyPassReverse   /usenetdl/ http://172.168.0.11/
```

## Install Usenet Services

### Setup NZBget - Binary Downloader

```
# cd /opt/
# wget https://nzbget.net/download/nzbget-latest-bin-linux.run
# sh nzbget-latest-bin-linux.run
# ls -l
# cd /opt/nzbget/
# ./nzbget -s
# useradd -r nzbget
```

```
# chown -R nzbget:nzbget /opt/nzbget  
  
# vim /etc/systemd/system/nzbget.service
```

```
[Unit]  
Description=NZBGet Daemon  
Documentation=http://nzbget.net/Documentation  
After=network.target  
  
[Service]  
User=nzbget  
Group=usenetserv  
Type=forking  
ExecStart=/opt/nzbget/nzbget -c /opt/nzbget/nzbget.conf -D  
ExecStop=/opt/nzbget/nzbget -Q  
ExecReload=/opt/nzbget/nzbget -0  
KillMode=process  
Restart=on-failure  
  
[Install]  
WantedBy=multi-user.target
```

```
# systemctl daemon-reload  
# systemctl start nzbget.service && systemctl status nzbget.service  
# systemctl enable nzbget.service  
  
# vim /opt/nzbget/nzbget.conf
```

```
...  
MainDir=${AppDir}/downloads  
...  
DestDir=/mnt/blackserv/downloads/_done  
...  
InterDir=/mnt/blackserv/downloads/intermediate  
...  
NzbDir=/mnt/blackserv/downloads/nzb  
...  
AuthorizedIP=127.0.0.1  
...
```

Der Rest von NZBget wird via Web-GUI konfiguriert

## Setup NzbHydra - Usenet Search Tool

```
# yum groupinstall -y "development tools"
# yum install -y zlib-devel openssl-devel sliteq-devel bzip2-devel python
ncurses-devel sqlite-devel readline-devel tk-devel gdbm-devel db4-devel
libpcap-devel xz-devel expat-devel

# cd /tmp/
# wget http://python.org/ftp/python/2.7.14/Python-2.7.14.tar.xz
# tar xf Python-2.7.14.tar.xz
# cd Python-2.7.14
# ./configure --prefix=/usr/local --enable-unicode=ucs4 --enable-shared
LDFLAGS="-Wl,-rpath /usr/local/lib"
# make && make altinstall
# /usr/local/bin/python2.7

# git clone https://github.com/theotherp/nzbhydra /opt/nzbhydra

# useradd -r nzbhydra
# chown -R nzbhydra:nzbhydra /opt/nzbhydra
# useradd -r nzbhydra
# ls -l /opt/nzbhydra/

# sed -e 's/"urlBase": null,/"urlBase": "\/usenetdl\/nzbhydra",/' -i
/opt/nzbhydra/settings.cfg
# /usr/local/bin/python2.7 /opt/nzbhydra/nzbhydra.py --daemon --nobrowser
# cat /opt/nzbhydra/settings.cfg | grep /nzbhydra
```

```
"externalUrl": "https://blackgate.org/usenetdl/nzbhydra",
"urlBase": "\/usenetdl\/nzbhydra",
```

```
# vim /etc/systemd/system/nzbhydra.service
```

```
[Unit]
Description=NZBHydra Daemon
Documentation=https://github.com/theotherp/nzbhydra
After=network.target

[Service]
User=nzbhydra
Group=usenetserver
Type=simple
ExecStart=/usr/local/bin/python2.7 /opt/nzbhydra/nzbhydra.py --nobrowser
```

```
KillMode=process
Restart=on-failure
```

```
[Install]
WantedBy=multi-user.target
```

```
# systemctl daemon-reload
# systemctl start nzbhydra.service && systemctl status nzbhydra.service
# systemctl enable nzbhydra.service

# netstat -tulpen
```

---

## Setup Sonarr - TV Download Tool

```
useradd -r sonarr
useradd -r radarr
vim /etc/httpd/conf/httpd.conf
systemctl reload httpd
yum install epel-release yum-utils -y
rpm --import
"http://keyserver.ubuntu.com/pks/lookup?op=get&search=0x3FA7E0328081BFF6A14D
A29AA6A19B38D3D831EF"
yum-config-manager --add-repo http://download.mono-project.com/repo/centos/
yum install wget mediainfo libzen libmediainfo curl gettext mono-core mono-
devel sqlite.x86_64
yum -y install wget git par2cmdline p7zip unrar unzip tar gcc python-
feedparser python-configobj python-cheetah python-dbus python-devel libxslt-
devel
cd /tmp/
wget http://update.sonarr.tv/v2/master/mono/NzbDrone.master.tar.gz
tar -xvf NzbDrone.master.tar.gz -C /opt/
mkdir /opt/sonarr
mkdir /opt/sonarr/bin
mv /opt/NzbDrone/* /opt/sonarr/bin
rm /opt/NzbDrone
rmdir /opt/NzbDrone
ls -l /opt/
ls -l /opt/sonarr/
ls -l /opt/sonarr/bin/
ls -l /opt/sonarr
chown -R sonarr:sonarr /opt/sonarr
vim /etc/systemd/system/sonarr.service
vim /etc/systemd/system/radarr.service
systemctl daemon-reload
systemctl start sonarr
systemctl status sonarr
```

```
vim /opt/sonarr/bin/NzbDrone.exe.config
systemctl status sonarr
curl http://127.0.0.1:8989
netstat -tulpen
systemctl stop sonarr
systemctl status sonarr
/usr/bin/mono /opt/sonarr/bin/NzbDrone.exe -nobrowser -data /opt/sonarr
systemctl status sonarr
ls -l /opt/sonarr/
ls -l /opt/sonarr/bin/
vim /etc/systemd/system/sonarr.service
systemctl daemon-reload
systemctl status sonarr
systemctl start sonarr
systemctl status sonarr
systemctl status sonarr -l
man useradd
sealert -a /var/log/audit/audit.log
echo > /var/log/audit/audit.log
sealert -a /var/log/audit/audit.log
systemctl start sonarr
systemctl status sonarr
ls -l /root/
ls -la /opt/sonarr/bin/
ls -la /opt/sonarr/
ls -lisa /opt/sonarr/
ls -lisa /opt/sonarr/bin/
ls -lisa /root/
ls -lisa /root/.config/
vim /etc/passwd
systemctl status sonarr
systemctl restart sonarr
systemctl status sonarr
systemctl enable sonarr
vim /opt/sonarr/.config/NzbDrone/config.xml
systemctl restart sonarr
systemctl status sonarr
netstat -tulpen
vim /opt/sonarr/.config/NzbDrone/config.xml
systemctl restart sonarr
vim /etc/httpd/conf/httpd.conf
systemctl reload httpd
vim /etc/httpd/conf/httpd.conf
systemctl reload httpd
vim /etc/httpd/conf/httpd.conf
systemctl reload httpd
vim /opt/sonarr/.config/NzbDrone/config.xml
systemctl restart sonarr
systemctl reload httpd
vim /opt/sonarr/.config/NzbDrone/config.xml
vim /etc/httpd/conf/httpd.conf
```

```
systemctl reload httpd
vim /etc/httpd/conf/httpd.conf
systemctl reload httpd
vim /etc/httpd/conf/httpd.conf
systemctl reload httpd
vim /etc/httpd/conf/httpd.conf
systemctl reload httpd
vim /etc/httpd/conf/httpd.conf
curl -v http://127.0.0.1:8989/
curl -v http://127.0.0.1:8989/sonarr/
vim /etc/httpd/conf/httpd.conf
systemctl reload httpd
vim /etc/httpd/conf/httpd.conf
systemctl reload httpd
history
```

---

## Setup Radarr- Movie Download Tool

```
cd /tmp
wget
https://github.com/Radarr/Radarr/releases/download/v0.2.0.980/Radarr.develop
.0.2.0.980.linux.tar.gz
tar -xvf Radarr.develop.0.2.0.980.linux.tar.gz

mkdir /opt/radarr
mkdir /opt/radarr/bin

cp -R Radarr/* /opt/radarr/bin/
ls -l /opt/radarr/bin/

chown -R radarr:radarr /opt/radarr/

vim /etc/systemd/system/radarr.service
systemctl start radarr && systemctl status radarr
systemctl enable radarr

vim /opt/radarr/.config/Radarr/config.xml
systemctl restart radarr
vim /etc/httpd/conf/httpd.conf
ip a
yum install nfs-utils -y
systemctl start rpcbind && systemctl enable rpcbind
vim /etc/fstab
mount -a
df -h
vim /etc/httpd/conf/httpd.conf
yum update
```



```
groupadd usenetserver
man usermod
usermod -a -G usenetserver nzbget
usermod -a -G usenetserver nzbhydra
usermod -a -G usenetserver radarr
usermod -a -G usenetserver sonarr
usermod -a -G apache michael
vim /etc/group

umount /mnt/blackserv
df -h
chgrp -R usenetserver /mnt/blackserv/
ls -l /mnt/
chmod -R 775 /mnt/blackserv/
mount -a

chgrp -R usenetserver /opt/sonarr
chgrp -R usenetserver /opt/nzb*
chgrp -R usenetserver /opt/radarr/

vim /etc/systemd/system/sonarr.service
vim /etc/systemd/system/nzbhydra.service
vim /etc/systemd/system/nzbget.service

vim /opt/radarr/.config/Radarr/config.xml

vim /opt/sonarr/.config/NzbDrone/config.xml

cd /tmp
wget
https://github.com/Radarr/Radarr/releases/download/v0.2.0.995/Radarr.develop
.0.2.0.995.linux.tar.gz
tar -xvf Radarr.develop.0.2.0.995.linux.tar.gz
mkdir /opt/radarr
mkdir /opt/radarr/bin
cp -R Radarr/* /opt/radarr/bin/

ls -l /opt/radarr/bin/
cat /opt/radarr/.mono/
cat /opt/radarr/.config/

chown -R radarr:radarr /opt/radarr/
vim /opt/radarr/.config/Radarr/config.xml

vim /etc/systemd/system/nzbget.service
vim /etc/systemd/system/radarr.service
```

```
systemctl status radarr.service
systemctl status radarr.service -l
vim /etc/systemd/system/nzbget.service
netstat -tulpen
vim /etc/httpd/conf/httpd.conf
firewall-cmd --list-services
firewall-cmd --list-ports
firewall-cmd --zone=public --add-port=8989/tcp --permanent
firewall-cmd --reload
firewall-cmd --zone=public --add-port=7878/tcp --permanent
firewall-cmd --reload
```

```
yum search mono-locale-extras
yum install mono-locale-extras
yum update
reboot
yum update
```

```
restorecon -Rv /var/www/html/
cd /var/www/html/
ls -la
setenforce 0
cat README.md
cd ..
ls -la
chmod apache:apache html/
```

```
vim /etc/systemd/system/radarr.service
vim /opt/sonarr/.config/NzbDrone/config.xml
vim /opt/radarr/.config/NzbDrone/config.xml
vim /opt/radarr/.config/Radarr/config.xml
systemctl restart radarr
systemctl reload httpd
vim /etc/httpd/conf/httpd.conf
systemctl reload httpd
ls -la /var/www/html/
mkdir /var/www/html/_staticPage
```

```
vim server_backup_script.sh
systemctl status httpd.service
ls -la /opt/
chmod +x server_backup_script.sh
ifconfig
./server_backup_script.sh
vim /etc/crontab
```

## Setup Lidarr - Music Download Tool

```
Install / compile Chromaprint dependency:
# yum install git ffmpeg ffmpeg-devel fftw taglib boost cmake cmake3

# cd /tmp/
# git clone https://bitbucket.org/acoustid/chromaprint.git
# cd chromaprint/
# ls -la
# cmake3 -DCMAKE_BUILD_TYPE=Release -DBUILD_TOOLS=ON .
# make
# make install
# which fpcalc
# /bin/fpcalc

# cd /tmp/
# wget
https://github.com/lidarr/Lidarr/releases/download/v0.4.0.524/Lidarr.develop
.0.4.0.524.linux.tar.gz
# tar -xvf Lidarr.develop.0.4.0.524.linux.tar.gz
# ls -la Lidarr/
# mkdir -p /opt/lidarr/bin
# cp -R /tmp/Lidarr/* /opt/lidarr/bin/

# ls -la /opt/lidarr/bin/
# useradd -r lidarr
# usermod -a -G usenetserver lidarr
# chown -R lidarr:lidarr /opt/lidarr/

# vim /opt/lidarr/.config/Lidarr/config.xml

# /usr/bin/mono /opt/lidarr/bin/Lidarr.exe -nobrowser -data /opt/lidarr
CTRL + C

# vim /etc/systemd/system/lidarr.service

ls -la /root/.config/Lidarr/
ls -la /home/
ls -la /root/.config/
rm -Rf /root/.config/
vim /etc/group
vim /etc/passwd

# systemctl daemon-reload
# systemctl start lidarr.service && systemctl status lidarr.service
# systemctl enable lidarr.service

firewall-cmd --zone=public --add-port=8686/tcp --permanent
firewall-cmd --reload
```

Last update: **2019/07/10 10:44**