

Installation

Na pierwszych 2 slajdach są podane informacje ogólne i należy je traktować jako informacje wstępne

The SSH service should already be installed, but if it is not, install it from a Yum repository using the following command.

```
# yum install openssh-server
```

Turn on the SSH service and make sure it starts automatically on reboot.

```
# service sshd start
# chkconfig sshd on
```

The SSH service is configured using the "/etc/ssh/sshd_config" file. Configuration changes have to be followed by a restart of the service.

```
# service sshd restart
# # or
# service sshd reload
```

Firewall

Konfiguracja firewala nie jest dla nas istotna, ponieważ firewall powinien być wyłączony.

The server must have the TCP port 22 open. This can be achieved by adding the following entry to the type of firewall script described [here](#).

```
# Open port for NTP server.
iptables -A INPUT -p tcp --dport 22 -j ACCEPT
```

SELinux

If you are using SELinux, you will need to consider the following points.

The SELinux booleans associated with the SSH service are displayed using the `getsebool` command.

```
# getsebool -a | grep ssh
allow_ssh_keysign --> off
fenced_can_ssh --> off
ssh_chroot_rw_homedirs --> off
ssh_sysadm_login --> off
#
```

The `setsebool` command is used to set a specific boolean value.

```
# setsebool ssh_sysadm_login on
# setsebool ssh_sysadm_login off
```

More information on SELinux can be found [here](#).

Manual User Equivalence (Key-Based Authentication) Configuration

Assuming we have a two node cluster (rac1.localdomain, rac2.localdomain), log in as the "oracle" user and perform the following tasks on each node.

```
su - oracle
mkdir ~/.ssh
chmod 700 ~/.ssh
/usr/bin/ssh-keygen -t rsa # Accept the default settings.
```

The RSA public key is written to the ~/.ssh/id_rsa.pub file and the private key to the ~/.ssh/id_rsa file.

Log in as the "oracle" user on rac1.localdomain, generate an "authorized_keys" file and copy it to rac2.localdomain using the following commands.

```
su - oracle
cd ~/.ssh
cat id_rsa.pub >> authorized_keys
scp authorized_keys rac2.localdomain:/home/oracle/.ssh/
```

Next, log in as the "oracle" user on rac2.localdomain and perform the following commands.

```
su - oracle
cd ~/.ssh
cat id_rsa.pub >> authorized_keys
scp authorized_keys rac1.localdomain:/home/oracle/.ssh/
```

The "authorized_keys" file on both servers now contains the public keys generated on all nodes.

To enable SSH user equivalency on the cluster member nodes issue the following commands on each node.

```
ssh rac1 date
ssh rac2 date
ssh rac1.localdomain date
ssh rac2.localdomain date
exec /usr/bin/ssh-agent $SHELL
/usr/bin/ssh-add
```

Te komendy można wykonać w przypadku,
gdy procedura przedstawiona na następnych slajdach
nie przyniesie oczekiwanego rezultatu.

You should now be able to SSH and SCP between servers without entering passwords.

```
[root@ol6-agrac-rac1 ~]# service sshd status
openssh-daemon (pid 2192) is running...
[root@ol6-agrac-rac1 ~]# service iptables status
iptables: Firewall is not running.
[root@ol6-agrac-rac1 ~]# getsebool -a | grep ssh
allow_ssh_keysign --> off
fenced_can_ssh --> off
ssh_chroot_full_access --> off
ssh_chroot_manage_apache_content --> off
ssh_chroot_rw_homedirs --> off
ssh_sysadm_login --> off
[root@ol6-agrac-rac1 ~]# █
```

```
[root@ol6-agraac-rac2 ~]# service sshd status
openssh-daemon (pid 2128) is running...
[root@ol6-agraac-rac2 ~]# service iptables status
iptables: Firewall is not running.
[root@ol6-agraac-rac2 ~]# getsebool -a | grep ssh
allow_ssh_keysign --> off
fenced_can_ssh --> off
ssh_chroot_full_access --> off
ssh_chroot_manage_apache_content --> off
ssh_chroot_rw_homedirs --> off
ssh_sysadm_login --> off
[root@ol6-agraac-rac2 ~]# █
```

```
[root@ol6-agrac-rac1 ~]# su - oracle
[oracle@ol6-agrac-rac1 ~]$ mkdir ~/.ssh
mkdir: cannot create directory `/home/oracle/.ssh': File exists
[oracle@ol6-agrac-rac1 ~]$ chmod 700 ~/.ssh
[oracle@ol6-agrac-rac1 ~]$ /usr/bin/ssh-keygen -t rsa
Generating public/private rsa key pair.
Enter file in which to save the key (/home/oracle/.ssh/id_rsa):
Enter passphrase (empty for no passphrase):
Enter same passphrase again:
Your identification has been saved in /home/oracle/.ssh/id_rsa.
Your public key has been saved in /home/oracle/.ssh/id_rsa.pub.
The key fingerprint is:
```

```
[root@ol6-agraac-rac2 ~]# su - oracle
[oracle@ol6-agraac-rac2 ~]$ mkdir ~/.ssh
mkdir: cannot create directory `/home/oracle/.ssh': File exists
[oracle@ol6-agraac-rac2 ~]$ chmod 700 ~/.ssh
[oracle@ol6-agraac-rac2 ~]$ /usr/bin/ssh-keygen -t rsa
Generating public/private rsa key pair.
Enter file in which to save the key (/home/oracle/.ssh/id_rsa):
Enter passphrase (empty for no passphrase):
Enter same passphrase again:
Your identification has been saved in /home/oracle/.ssh/id_rsa.
Your public key has been saved in /home/oracle/.ssh/id_rsa.pub.
The key fingerprint is:
```

```
[oracle@ol6-agrac-rac1 .ssh]$ ls -la
```

```
total 20
```

```
drwx-----. 2 oracle oinstall 4096 Jan 25 23:45 .
drwx-----. 26 oracle oinstall 4096 Jan 25 21:26 ..
-rw-----. 1 oracle oinstall 1675 Jan 25 23:30 id_rsa
-rw-r--r--. 1 oracle oinstall  415 Jan 25 23:30 id_rsa.pub
-rw-r--r--. 1 oracle oinstall 1214 Jan 25 23:38 known_hosts
```

```
[oracle@ol6-agrac-rac1 .ssh]$ cat id_rsa.pub >> authorized_keys
```

```
[oracle@ol6-agrac-rac1 .ssh]$ ls -la
```

```
total 24
```

```
drwx-----. 2 oracle oinstall 4096 Jan 25 23:46 .
drwx-----. 26 oracle oinstall 4096 Jan 25 21:26 ..
-rw-r--r--. 1 oracle oinstall  415 Jan 25 23:46 authorized_keys
-rw-----. 1 oracle oinstall 1675 Jan 25 23:30 id_rsa
-rw-r--r--. 1 oracle oinstall  415 Jan 25 23:30 id_rsa.pub
-rw-r--r--. 1 oracle oinstall 1214 Jan 25 23:38 known_hosts
```

```
[oracle@ol6-agrac-rac1 .ssh]$ scp authorized_keys ol6-agrac-rac2.localdomain:/home/oracle/.ssh
```

```
[oracle@ol6-agrac-rac2 ~]$ cd ~/.ssh
[oracle@ol6-agrac-rac2 .ssh]$ ls -la
total 24
drwx-----. 2 oracle oinstall 4096 Jan 25 23:48 .
drwx-----. 26 oracle oinstall 4096 Jan 25 23:12 ..
-rw-r--r--. 1 oracle oinstall  415 Jan 25 23:48 authorized_keys
-rw-----. 1 oracle oinstall 1675 Jan 25 23:35 id_rsa
-rw-r--r--. 1 oracle oinstall  415 Jan 25 23:35 id_rsa.pub
-rw-r--r--. 1 oracle oinstall  792 Dec 13 18:41 known_hosts
[oracle@ol6-agrac-rac2 .ssh]$ cat id_rsa.pub >> authorized_keys
[oracle@ol6-agrac-rac2 .ssh]$ scp authorized_keys ol6-agrac-rac1.localdomain:/home/oracle/.ssh/
```



```
[Oracle@ol6-agrac-rac1 .ssh]$ ssh ol6-agrac-rac2
```

```
[oracle@o16-agrac-rac2 ~]$ ssh o16-agrac-rac1
```

```
[oracle@ol6-agraac-rac1 .ssh]$ ssh ol6-agraac-rac2
```

```
Last login: Thu Jan 26 00:06:57 2017 from ol6-agraac-rac1.localdomain
```

```
[oracle@ol6-agraac-rac2 ~]$ ssh ol6-agraac-rac1
```

```
Last login: Thu Jan 26 00:07:52 2017 from ol6-agraac-rac2.localdomain
```

```
[oracle@ol6-agraac-rac1 ~]$ █
```

```
[oracle@ol6-agrac-rac2 .ssh]$ ssh ol6-agrac-rac1
Last login: Thu Jan 26 00:09:31 2017 from ol6-agrac-rac2.localdomain
[oracle@ol6-agrac-rac1 ~]$ ssh ol6-agrac-rac2
Last login: Thu Jan 26 00:09:08 2017 from ol6-agrac-rac1.localdomain
[oracle@ol6-agrac-rac2 ~]$ █
```

```
[oracle@ol6-agraac-rac1 grid]$ ssh ol6-agraac-rac1
Last login: Fri Jan 27 01:05:40 2017 from ol6-agraac-rac1.localdomain
[oracle@ol6-agraac-rac1 ~]$ exit
logout
Connection to ol6-agraac-rac1 closed.
[oracle@ol6-agraac-rac1 grid]$ ssh ol6-agraac-rac1
Last login: Fri Jan 27 01:05:47 2017 from ol6-agraac-rac1.localdomain
[oracle@ol6-agraac-rac1 ~]$ █
```

```
[oracle@ol6-agrac-rac2 ~]$ ssh ol6-agrac-rac2
Last login: Fri Jan 27 01:01:26 2017 from ol6-agrac-rac2.localdomain
[oracle@ol6-agrac-rac2 ~]$ exit
logout
Connection to ol6-agrac-rac2 closed.
[oracle@ol6-agrac-rac2 ~]$ ssh ol6-agrac-rac2
Last login: Fri Jan 27 01:02:22 2017 from ol6-agrac-rac2.localdomain
[oracle@ol6-agrac-rac2 ~]$ █
```