

Содержание

4 OS installation requirements for SORM-3 information system 3

4 OS installation requirements for SORM-3 information system

CentOS 7 x86_64 OS should be installed

(http://mirror.yandex.ru/centos/7/isos/x86_64/CentOS-7-x86_64-Minimal-1804.iso). You should not need to disable SELinux. After installation you should run the following command:

```
yum update -y
```

as a root and restart after the updates installation completes:

```
shutdown -r now
```

After the reboot, perform the installation of the auxiliary software using the following command: `yum install -y vim-enhanced sudo mc git gcc openssl-devel wget screen sysstat setroubleshoot ntp lvm2 epel-release vsftpd`.

1. I/O subsystem specifications

- a. Disk drives available for the operating system have to be organized in RAID (> RAID0), which implies that the system will operate when a single physical disk fails;
- b. The volumes used to create the file system for PostgreSQL should provide latency at most 10-12 ms with 3000 iops to write (sequential write) and 7000 iops to read (sequential read).

2. OS disk space allocation:

- a. The installation should use LVM for all file systems;
- b. At least 100GB is required when combining the root file system and the /var file system;
- c. Otherwise, at least 50GB is required when the root file system is not combined with the /var one.
- d. The /opt file system should be located on a separate logical volume in a separate volume group; its size should be at least 100GB;
- e. The /var/lib/pgsql file system should be located on a separate logical volume in a separate volume group; the size is calculated by the operator based on the channel bandwidth [Typical SORM-3 equipment](#). When using multiple disks for this VG, LV should be created using stripe (the -i and -l options); the number of stripes == the number of disks.

3. OS configuration requirements

- a. Users of the wheel group are given permission to use all commands in the context of all users: you should add to the /etc/sudoers file the following line: `%wheel ALL=(ALL) NOPASSWD: ALL`
- b. and create the following users:

```
useradd -m -g wheel -u 3000 AAlekseenko  
useradd -m -g wheel -u 3001 Ilya.Volzhev
```

```

useradd -m -g wheel -u 3002 denis.alexandrov
useradd -m -g wheel -u 3003 stanislav.polevik
useradd -m -g wheel -u 3004 andrey.voloshin
useradd -m -g wheel -u 3005 alexander.suleymanov
useradd -m -g wheel -u 3006 kirill.ivanov
useradd -m -g wheel -u 3007 konstantin.mikhaylov

openssl rand -base64 32 | passwd --stdin AAlekseenko
openssl rand -base64 32 | passwd --stdin Ilya.Volzhev
openssl rand -base64 32 | passwd --stdin denis.alexandrov
openssl rand -base64 32 | passwd --stdin stanislav.polevik
openssl rand -base64 32 | passwd --stdin andrey.voloshin
openssl rand -base64 32 | passwd --stdin alexander.suleymanov
openssl rand -base64 32 | passwd --stdin kirill.ivanov
openssl rand -base64 32 | passwd --stdin konstantin.mikhaylov

```

using corresponding SSH keys, available to authentication:

```

mkdir ~AAlekseenko/.ssh && echo 'ssh-ed25519
AAAAC3NzaC1lZDI1NTE5AAAAIPWzgE2at7UudgJZLAzwKlF/5Rmctqmju2qEbR8yboEi
AAlekseenko@ssh.vasexperts.ru' > ~AAlekseenko/.ssh/authorized_keys && chown
-R AAlekseenko:wheel ~AAlekseenko/.ssh
mkdir ~Ilya.Volzhev/.ssh && echo 'ssh-ed25519
AAAAC3NzaC1lZDI1NTE5AAAAIGbwz32CWcgHfEFn68uGojHXEAzuEA8kSvPLrZQ0z7/B
Ilya.Volzhev@ssh.vasexperts.ru' > ~Ilya.Volzhev/.ssh/authorized_keys &&
chown -R Ilya.Volzhev:wheel ~Ilya.Volzhev/.ssh
mkdir ~denis.alexandrov/.ssh && echo 'ssh-ed25519
AAAAC3NzaC1lZDI1NTE5AAAAIMULzwCAXyXDACbJ57ZtrbKstD0QJhWsfm+F6yPb5KJd
denis.alexandrov@ssh.vasexperts.ru' > ~denis.alexandrov/.ssh/authorized_keys
&& chown -R denis.alexandrov:wheel ~denis.alexandrov/.ssh
mkdir ~stanislav.polevik/.ssh && echo 'ssh-ed25519
AAAAC3NzaC1lZDI1NTE5AAAAIBNlqVYWkvUn4pJuX70PkUucgp4cihiZ6fFIzTUnKtEk
stanislav.polevik@ssh.vasexperts.ru' >
~stanislav.polevik/.ssh/authorized_keys && chown -R stanislav.polevik:wheel
~stanislav.polevik/.ssh
mkdir ~andrey.voloshin/.ssh && echo 'ssh-ed25519
AAAAC3NzaC1lZDI1NTE5AAAAIPsdFrdfNudtBBWr3iIn/xyJeCL5/yLSQZT9A5LKG2GS
andrey.voloshin@ssh.vasexperts.ru' > ~andrey.voloshin/.ssh/authorized_keys
&& chown -R andrey.voloshin:wheel ~andrey.voloshin/.ssh
mkdir ~alexander.suleymanov/.ssh && echo 'ssh-ed25519
AAAAC3NzaC1lZDI1NTE5AAAAIDJyZH9r+Pbfsklh1hjtMQwyqCVn57x8cj7y20HqfP2t
alexander.suleymanov@ssh.vasexperts.ru' >
~alexander.suleymanov/.ssh/authorized_keys && chown -R
alexander.suleymanov:wheel ~alexander.suleymanov/.ssh
mkdir ~kirill.ivanov/.ssh && echo 'ssh-ed25519
AAAAC3NzaC1lZDI1NTE5AAAAIKSbxFhBHiPxRYvReknP0Rez5YK76p2LFkc0n7mj03co
kirill.ivanov@ssh.vasexperts.ru' > ~kirill.ivanov/.ssh/authorized_keys &&
chown -R kirill.ivanov:wheel ~kirill.ivanov/.ssh
mkdir ~konstantin.mikhaylov/.ssh && echo 'ssh-ed25519
AAAAC3NzaC1lZDI1NTE5AAAAII/0LHqdxq6Fo4v+w55rbYoe3ElJWk4Vf+/dY3GCWYY/

```

```
konstantin.mikhaylov@ssh.vasexperts.ru' >  
~konstantin.mikhaylov/.ssh/authorized_keys && chown -R  
konstantin.mikhaylov:wheel ~konstantin.mikhaylov/.ssh
```

c. Add the following keys used for authentication to the root user:

```
mkdir /root/.ssh  
echo 'ssh-ed25519  
AAAAC3NzaC1lZDI1NTE5AAAAILBz8xQUuLBZzVqSph0RGVLIuqyYM0TLyp/y1e3jmV7F  
evgueni.gavrilov@it-grad.ru' >> /root/.ssh/authorized_keys  
echo 'ssh-ed25519  
AAAAC3NzaC1lZDI1NTE5AAAAILLzeWIDUXmUqFIriBvLUkv/PFdcM8rgThYyG8ZnwdLq  
dmitry.kozlov@it-grad.ru' >> /root/.ssh/authorized_keys
```

Furthermore, you should make sure that root user authentication in sshd is possible using SSH keys: the PermitRootLogin parameter should not be set in no value in the /etc/ssh/sshd_config file; valid values are without-password (ssh public key only login-authentication) or yes (authentication is available both using password and ssh key).

d. FirewallD is used as a firewall.

e. Switch sshd to port 22022 / tcp.

In SELinux add a non-standard port to the list of allowed ones using the following command:

```
semanage port -a -t ssh_port_t -p tcp 22022
```

Add corresponding changes to the /etc/ssh/sshd_config configuration file by default using the following command:

```
sed -i.BAK -e 's,^#Port 22,Port 22022,' /etc/ssh/sshd_config
```

or you can do it manually by modifying the Port parameter and setting it to 22022.

Allow ssh access from ssh.vasexperts.ru (5.101.76.50):

```
firewall-cmd --permanent --add-rich-rule='rule family="ipv4" source  
address="5.101.76.50/32" port port=22022 protocol="tcp" accept'  
firewall-cmd --reload
```

in order you changes (usage of non-standard port) take effect you should issue the following command:

```
systemctl reload sshd
```

g. the same way, if necessary, you should add "your" IP addresses

h. create cdr user (the operator data will be loaded from the billing or NAT translations on behalf of this user):

```
useradd -m -s /sbin/nologin cdr
openssl rand -base64 32 | passwd --stdin cdr
```

i. ensure that the vsftpd configurations (the /etc/vsftpd/vsftpd.conf file) reads as follows:

```
listen=YES
background=YES
pam_service_name=vsftpd
tcp_wrappers=YES
anonymous_enable=NO
local_enable=YES
write_enable=YES
connect_from_port_20=NO
xferlog_enable=NO
xferlog_file=/var/log/vsftpd.log
async_abor_enable=YES
chroot_local_user=YES
chroot_list_enable=NO
chroot_list_file=/etc/vsftpd/chroot_list
allow_writeable_chroot=YES
userlist_enable=NO
userlist_deny=NO
user_config_dir=/etc/vsftpd/users
force_dot_files=YES
local_umask=022
dirmessage_enable=YES
pasv_enable=YES
pasv_max_port=10100
pasv_min_port=10090
hide_file=NO
tcp_wrappers=YES
ascii_upload_enable=YES
ascii_download_enable=YES
local_umask=022
```