Инструкция по настройке прокси-сервера на RED OS 7.3

Оглавление

Разработчик инструкции: П.В. Сметанин
тел. (34922) 2-23-42

Разработчик инструкции: П.В. Сметанин
тел. (34922) 2-23-42

СИСТЕМНЫЕ ТРЕБОВАНИЯ

Системные требования сервера, взятые из расчёта 50 одновременных подключений:
- Обязательно наличие **не менее 2 сетевых карт** (пропускной способностью 1Гб/с.), одна из которых смотрит во внешнюю сеть (**Интернет**), другая во внутреннюю (**локальная сеть предприятия**).
- Обязательно наличие **не менее 2 накопителей одинакового объёма** (от 80 Гб), для создания программного RAID-массива уровня 1 (зеркало).
- Объём оперативной памяти **не менее 2048 Мб**.
- Процессор **не ниже Intel Celeron 2.50GHz** и **не менее 2 физических ядер** (количество виртуальных ядер и потоков по желанию, здесь рассматриваются минимальные характеристики необходимые для функционирования прокси-сервера).

ОСНОВНЫЕ ОБОЗНАЧЕНИЯ

текст в терминале или командной строке
#комментарии или действия, выполняемые в граф. интерфейсе. Жирным выделена **важная информация**
текст в файле или вывод команды
дополнительная информация

I. УСТАНОВКА И НАСТРОЙКА ОС

1. Установка ОС

1.1. Записываем ISO-образ на флешку

#загружаем UltraISO и устанавливаем себе на компьютер.
#вставляем флешку, запускаем программу: Самозагрузка -> Записать образ жесткого диска -> указываем флешку -> указываем ISO-образ -> Форматировать -> Записать

1.2. Загружаемся со съёмного носителя

#Выбираем **Установить RED OS.**

1.3. Язык установки

#Русский (Россия)

1.4. Дата/время

#Екатеринбург, 24-часа

Разработчик инструкции: П.В. Сметанин
тел. (34922) 2-23-42

1.5. Клавиатура

#Ставим приоритет английскому языку

1.6. Выбор программ

#Сервер с графическим интерфейсом -> Дополнительное ПО: выбираем всё (лишние программы никак не повлияют на работоспособность сервера).

1.7. Место установки

#ВАЖНО! Если у вас на сервере уже собран аппаратный RAID, то в конфигурации устройств хранения оставляете **Автоматически** и ставим галочку **Зашифровать данные**.
#выбираем оба диска -> Конфигурация устройств хранения: По-своему -> Готово
#в открывшемся окне жмём «+» -> Точка монтирования: **/boot** -> Требуемый объём: **1G** -> Добавить точку монтирования -> Тип устройства: RAID -> Уровень RAID: **RAID 1** -> галочку Зашифровать -> версия LUKS: luks2 -> остальное без изменений
#снова жмём «+» -> Точка монтирования: **swap** -> Требуемый объём: **4G** (объём должен минимум в **2 раза** превышать объём RAM сервера, т.е. если у вас установлено 2Гб RAM, то здесь указываете 4G и т.д.) -> Добавить точку монтирования -> Тип устройства: RAID -> Уровень RAID: **RAID 1** -> галочку Зашифровать -> версия LUKS: luks2 -> остальное без изменений
#снова жмём «+» -> Точка монтирования: / -> Требуемый объём: **(оставляем поле пустым)** -> Добавить точку монтирования -> Тип устройства: RAID -> Уровень RAID: **RAID 1** -> галочку Зашифровать -> версия LUKS: luks2 -> остальное без изменений -> Готово
#Парольная фраза: указываем **сложный пароль ->** Подтверждаем повторным вводом -> Сохранить парольную фразу -> Принять изменения

1.8. Имя сети и узла

#в данной инструкции в качестве примера будем использовать следующие сетевые настройки: **10.10.73.0/24** - сеть, смотрящая во внешку; **192.168.0.0/24** - внутренняя локальная сеть.
#ВАЖНО! У вас должны быть свои сетевые настройки.
#Выбираем сетевую карту, смотрящую во **внешку (Интернет)** -> Настроить -> Параметры IPv4 -> Вручную -> Добавить -> Адрес: **10.10.73.100** -> Маска сети: **24** -> Шлюз: **10.10.73.1** -> Серверы DNS: **10.10.73.5, 10.10.73.9** -> Требовать адресацию IPv4 для этого соединения -> Сохранить.
#Выбираем сетевую карту, смотрящую в **локалку** -> Настроить -> Параметры IPv4 -> Вручную -> Добавить -> Адрес: **192.168.0.10** -> Маска сети: **24** -> Шлюз: **(оставляем поле пустым)**-> Серверы DNS: **(оставляем поле пустым)** -> Требовать адресацию IPv4 для этого соединения -> Сохранить -> Готово

1.9. Пароль ROOT

#Пароль root: **суперсложный пароль** -> Подтверждаем повторным вводом -> снимаем все галочки, если стоят -> Готово

1.10. Создание пользователя

#Полное имя: **Администратор** -> Имя пользователя: **LocalAdmin** -> галочка Сделать этого пользователя администратором -> галочка Требовать пароль: указываем **сложный пароль ->** Подтверждаем повторным вводом -> Готово
#Начать установку
#по окончании установки Перезагрузка системы

2. Настройка ОС

2.1. Первая настройка

#вводим пароль для расшифровки дисков (он будет запрашиваться всякий раз при включении)
#принимаем лицензионное соглашение -> Завершить

Разработчик инструкции: П.В. Сметанин
тел. (34922) 2-23-42

2.2. Проверка RAID-массива

#вводим команду:

cat /proc/mdstat

#**если создавали** программный RAID, то должен получиться примерно такой вывод, где **2/2** означает зеркало **RAID 1**:

```
Personalities : [raid1]
md125 : active raid1 sdb3[1] sda3[0]
      15711232 blocks super 1.2 [2/2] [UU]
      bitmap: 1/1 pages [4KB], 65536KB chunk
md126 : active raid1 sda2[0] sdb2[1]
      1047552 blocks super 1.2 [2/2] [UU]
      bitmap: 0/1 pages [0KB], 65536KB chunk
md127 : active raid1 sdb1[1] sda1[0]
      4195328 blocks super 1.2 [2/2] [UU]
      bitmap: 0/1 pages [0KB], 65536KB chunk
unused devices: <none>
```

#как восстановить RAID 1 после замены диска см. **здесь**

2.3. Проверяем права на директорию и обновляем систему.

#логинимся в систему

#открываем терминал (Ctrl+Alt+T) и вводим команду:

cd /home/LocalAdmin/.config/dconf/

ls -la

#должно быть примерно так:

```
drwx------.  2 LocalAdmin LocalAdmin 4096 дек 17 10:48 .
drwx------. 11 LocalAdmin LocalAdmin 4096 дек 17 10:48 ..
-rw-r--r--.  1 LocalAdmin LocalAdmin 7075 дек 17 10:48 user
```

#если нет, вводим:

sudo chown LocalAdmin user

#скачиваем с сервера более свежие версии пакетов

sudo dnf -y update && sudo dnf -y upgrade

2.4. Установка дополнительных пакетов

#вводим команду

sudo dnf -y install gnome-disk-utility gparted lshw htop lsscsi gnome-multi-writer nautilus

2.5. Настраиваем DNS

#удаляем сетевой интерфейс моста virbr0:

sudo virsh net-destroy default

sudo virsh net-undefine default

Разработчик инструкции: П.В. Сметанин
тел. (34922) 2-23-42

sudo service libvirtd restart

reboot

#проверяем, должно остаться 3 интерфейса (внешний, внутренний и петлевой):

ifconfig

enp0s3: flags=4163<UP,BROADCAST,RUNNING,MULTICAST> mtu 1500
        inet **10.10.73.100** netmask **255.255.255.0** broadcast 10.10.73.255
        inet6 fe80::a00:27ff:fe99:f39e prefixlen 64 scopeid 0x20<link>
        ether 08:00:27:99:f3:9e txqueuelen 1000 (Ethernet)
        RX packets 546592 bytes 792978499 (756.2 MiB)
        RX errors 0 dropped 0 overruns 0 frame 0
        TX packets 242319 bytes 16529526 (15.7 MiB)
        TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0
enp0s8: flags=4163<UP,BROADCAST,RUNNING,MULTICAST> mtu 1500
        inet **192.168.0.10** netmask **255.255.255.0** broadcast 192.168.0.255
        inet6 fe80::a00:27ff:fe77:3cb0 prefixlen 64 scopeid 0x20<link>
        ether 08:00:27:77:3c:b0 txqueuelen 1000 (Ethernet)
        RX packets 186 bytes 16699 (16.3 KiB)
        RX errors 0 dropped 0 overruns 0 frame 0
        TX packets 182 bytes 23099 (22.5 KiB)
        TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0
lo: flags=73<UP,LOOPBACK,RUNNING> mtu 65536
        inet **127.0.0.1** netmask 255.0.0.0
        inet6 ::1 prefixlen 128 scopeid 0x10<host>
        loop txqueuelen 1000 (Local Loopback)
        RX packets 27 bytes 4138 (4.0 KiB)
        RX errors 0 dropped 0 overruns 0 frame 0
        TX packets 27 bytes 4138 (4.0 KiB)
        TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0

…

#указываем внешние DNS сервера

sudo vim /etc/systemd/resolved.conf

[Resolve]

…

DNS=**10.10.73.5 10.10.73.9**

…

#проверяем пинг наружу

ping ya.ru

PING ya.ru (87.250.250.242) 56(84) bytes of data.
64 bytes from ya.ru (87.250.250.242): icmp_seq=1 ttl=249 time=4.12 ms
64 bytes from ya.ru (87.250.250.242): icmp_seq=2 ttl=249 time=4.04 ms
64 bytes from ya.ru (87.250.250.242): icmp_seq=3 ttl=249 time=4.24 ms

Разработчик инструкции: П.В. Сметанин
тел. (34922) 2-23-42

64 bytes from ya.ru (87.250.250.242): icmp_seq=4 ttl=249 time=4.08 ms

nslookup ya.ru

Server: 10.10.73.5
Address: 10.10.73.5#53
Non-authoritative answer:
Name: ya.ru
Address: 87.250.250.242
Name: ya.ru
Address: 2a02:6b8::2:242
#если всё работает, продолжаем установку

2.6. Вводим сервер в домен

#открываем терминал и вводим команду:
#установка из репозитория программы

sudo dnf -y install join-to-domain

#ввод компьютера в домен
#Заходим: Системные -> Ввод ПК в домен -> Домен Windows/Samba
#Заполняем строки: **Имя домена**, **Имя компьютера**, Имя **администратора** домена и **Пароль** администратора домена. Жмём Да.
#Вылезет окошко «*Компьютер успешно введён в домен*»
#проверяем доступность домена

realm list

#проверяем новое имя компьютера

hostname

#Если имя осталось прежним, меняем имя компьютера принудительно:
sudo hostnamectl set-hostname ИМЯ_КОМПЬЮТЕРА

2.7. Наделяем группу *Администраторы домена* полномочиями **sudo.**

#логинимся под **root**

su -

#заходим в редактор настроек **root.** Сам файл находится тут: **/etc/sudoers** *i - редактировать, Esc - закончить редактировать, :wq! - сохранить и выйти*

visudo

#ищем строчку:
Allows people in group wheel to run all commands
%wheel ALL=(ALL) ALL
#и прописываем под ней дополнительную строку:
%Администраторы\ домена ALL=(ALL) ALL
#теперь все члены группы *Администраторы домена* могут выполнять команды от привилегии **sudo.**
#если производите настройку пользователю домена, то под строкой **%Администраторы\ домена** укажите следующее:
%Пользователи\ домена ALL=(ALL) ALL

Разработчик инструкции: П.В. Сметанин
тел. (34922) 2-23-42

exit

#Чтобы вводить в терминале команды от локального администратора, находясь под обычным пользователем, нужно набрать:

su LocalAdmin

#или ваша учётка, если вы администратор домена

#Перезагружаеся

reboot

#заходим -> выбираем «Нет в списке?» и вводим поочерёдно ***Имя пользователя домена*** и ***Пароль*** учётной записи, под которой вы собираетесь залогиниться.

#указывать домен не нужно

username

****

2.8. Настраиваем протокол **VNC**.

#все манипуляции проделываем от **root**

su -

#устанавливаем пакет программы удалённого доступа x11vnc

dnf -y install x11vnc

#Задаём **пароль**

x11vnc -storepasswd "пароль" /etc/vncpasswd

#закрываем права на редактирование файла с паролем

chmod 544 /etc/vncpasswd

#заходим в сервисную директорию и редактируем файл для автозапуска сервиса x11vnc.service

vim /lib/systemd/system/x11vnc.service

#Заполняем файл следующим содержимым:

#Через запятую, без пробелов введите **ip-адреса**, которые будут подключаться к настраиваемой машине по протоколу VNC (пример: **192.168.1.15,192.168.1.26**)

```
[Unit]
Description=x11vnc server for GDM
After=display-manager.service
[Service]
ExecStart=/usr/bin/x11vnc -allow ip_адреса -many -shared -forever -nomodtweak -capslock -display :0 -auth guess -noxdamage -rfbauth /etc/vncpasswd
Restart=on-failure
RestartSec=3
[Install]
WantedBy=graphical.target
```

#даём файлу **x11vnc.service** права на выполнение

chmod ugo+x /lib/systemd/system/x11vnc.service

#Перезагружаем демона

systemctl daemon-reload

#Добавляем службу в автозагрузку и запускаем её

systemctl enable x11vnc.service

systemctl start x11vnc.service

Разработчик инструкции: П.В. Сметанин
тел. (34922) 2-23-42

#Запускаем службу и проверяем статус, должен быть *«active (running)»*

systemctl status x11vnc.service

#выходим из-под root

exit

#Далее устанавливаем на вашей *Windows*-машине клиент VNC (примеры клиентов: TightVNC, UltraVNC, TigerVNC, VNC Viewer, TurboVNC) и прописываем в строке поиска **ip-адрес_машины_к_которой_подключаетесь:порт_подключения (в основном это 5900).**

#Вводим пароль, ставим галочку сохранить и проверяем подключение.

2.9. Настраиваем протокол **SSH.**

#редактируем файл настроек, меняем стандартный порт с 22 на **2002**, меняем интернет протокол на **IPv4**, отключаем аутентификацию под **root** (вход должен производиться от имени пользователя, в дальнейшем можно перелогиниться под **root.** Это нужно для ведения журнала: кто и когда заходил.)

#открываем терминал и вводим:

sudo vim /etc/ssh/sshd_config

#ищем строчки:

#Port 22

#AddressFamily any

…

#PermitRootLogin prohibit-password

#редактируем их следующим образом, не забываем убрать решётки (раскомментировать):

Port **2002**

AddressFamily **inet**

…

PermitRootLogin **no**

#сохраняемся, оповещаем SELinux об изменении порта и перезапускаем службу

sudo semanage port -a -t ssh_port_t -p tcp 2002

sudo systemctl restart sshd

#На Windows-машине скачиваем и запускаем клиента **Putty.**

#В поле **Host Name (or IP address)** вводим ip-адрес машины к которой подключаемся. В поле **port** вводим **2002**. Больше ничего не меняем -> *Open*.

#Вводим логин **вашей учётной записи** или локального администратора (не забываем, что под root система не впустит), вводим пароль.

#Чтобы подключиться из Linux-машины, откройте терминал и введите:

ssh имя_пользователя@ip-адрес

II. ПОДГОТОВКА И НАСТРОЙКА ПРОКСИ-СЕРВЕРА

3. Настраиваем маршрутизацию

#разрешаем проброс между сетевыми интерфейсами в файле */etc/sysctl.conf* и отключаем проброс по протоколу ipv6:

sudo vim /etc/sysctl.conf

sysctl settings are defined through files in

/usr/lib/sysctl.d/, /run/sysctl.d/, and /etc/sysctl.d/.

Vendors settings live in /usr/lib/sysctl.d/.

Разработчик инструкции: П.В. Сметанин
тел. (34922) 2-23-42

To override a whole file, create a new file with the same in
/etc/sysctl.d/ and put new settings there. To override
only specific settings, add a file with a lexically later
name in /etc/sysctl.d/ and put new settings there.
For more information, see sysctl.conf(5) and sysctl.d(5).
net.ipv4.ip_forward=1
net.ipv6.conf.all.disable_ipv6 = 1
net.ipv6.conf.default.disable_ipv6 = 1
net.ipv6.conf.lo.disable_ipv6 = 1
#проверяем
sudo sysctl -p
net.ipv4.ip_forward = 1
net.ipv6.conf.all.disable_ipv6 = 1
net.ipv6.conf.default.disable_ipv6 = 1
net.ipv6.conf.lo.disable_ipv6 = 1
#проверяем что по умолчанию в iptables все разрешено и нет никаких правил:
sudo iptables -F
sudo iptables -t nat -F
sudo iptables -S
-P INPUT ACCEPT
-P FORWARD ACCEPT
-P OUTPUT ACCEPT
-N LIBVIRT_FWI
-N LIBVIRT_FWO
-N LIBVIRT_FWX
-N LIBVIRT_INP
-N LIBVIRT_OUT
#чтобы добавить новый маршрут (новую подсеть) к локальной IP подсети вашего прокси сервера в таблицу маршрутизации, введите команду:
sudo ip route add 192.168.3.0/24 via 192.168.0.1
sudo ip route add 192.168.4.0/24 via 192.168.0.1
sudo ip route add 192.168.5.0/24 via 192.168.0.1
#и т.д.
#смотрим маршруты
ip route
#должно быть примерно следующее:
default via **10.10.73.1** dev eno1 proto static metric 101
192.168.0.0/24 dev eno2 proto kernel scope link src **192.168.0.10** metric 100
192.168.3.0/24 via 192.168.0.1dev eno2
192.168.4.0/24 via 192.168.0.1dev eno2
192.168.5.0/24 via 192.168.0.1dev eno2

Разработчик инструкции: П.В. Сметанин
тел. (34922) 2-23-42

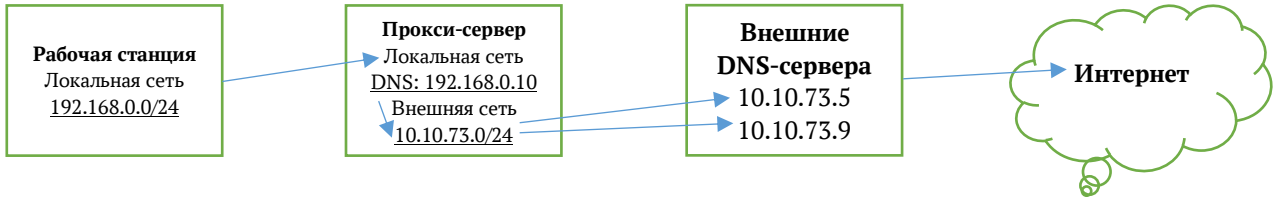
10.10.73.0/24 dev eno1 proto kernel scope link src **10.10.73.100** metric 101
#чтобы удалить созданный вручную маршрут, выполните:
sudo ip route del 192.168.3.0/24
#смотрим прослушивается ли порт 53:
netstat -ntulp
Active Internet connections (only servers)
Proto Recv-Q Send-Q Local Address Foreign Address State PID/Program name
…
tcp 0 0 **127.0.0.1:53** 0.0.0.0:* LISTEN -
…
udp 0 0 **127.0.0.1:53** 0.0.0.0:* -
#выполним сохранение правил
sudo service iptables save
iptables: Saving firewall rules to /etc/sysconfig/iptables:[OK]
#лишним не будет сохранить правила в файл:
sudo iptables-save > iptables.backup

4. Настраиваем проброс DNS соединений через сервер Bind

#ВАЖНО! настраивается только в том случае, если ваша локальная сеть не имеет выхода в интернет. Тогда в качестве DNS придётся указать ip-адрес прокси сервера.
#если у вашей локальной сети есть свои DNS смотрящие во внешку, то сервер Bind настраивать не нужно. Сразу переходите к **п.10**.
#устанавливаем утилиту для синхронизации времени, отключаем cronyd и запускаем ntpd
sudo dnf -y install ntp
sudo systemctl disable chronyd
sudo systemctl enable ntpd
sudo systemctl stop chronyd
sudo systemctl start ntpd
#настраиваем временную зону (время +2 от московского):
sudo cp /usr/share/zoneinfo/Asia/Yekaterinburg /etc/localtime
#синхронизируем время с внешним сервером:
sudo ntpdate -s time.yandex.ru
#устанавливаем программу
sudo dnf -y install bind
#разрешаем автозапуск:
sudo systemctl enable named
#запускаем сервис имен:
sudo systemctl start named
#проверяем, что он работает корректно:
sudo systemctl status named

Разработчик инструкции: П.В. Сметанин
тел. (34922) 2-23-42

#сервер bind в данной конфигурации будет перенаправлять все DNS-запросы через себя на указанный ему DNS-сервер(а) (в нашем случае это **10.10.73.5** и **10.10.73.9** смотрящие во внешку).



#для этого в его конфиге нужно указать куда передавать трафик, по какому порту и сетевой карте слушать запросы, а также указать перенаправление запроса.
#все это осуществляется редактированием файла */etc/named.conf*, где: **forwaders** — вышестоящий DNS, используемый в случаях, когда в базе не удаётся найти URL-запрос. **listen-on** — адреса, через которые будет обслуживаться наш DNS-сервер.

sudo vim /etc/named.conf

```
options {
        listen-on port 53 { 127.0.0.1; 192.168.0.10; };
        directory       "/var/named";
        dump-file       "/var/named/data/cache_dump.db";
        statistics-file "/var/named/data/named_stats.txt";
        memstatistics-file "/var/named/data/named_mem_stats.txt";
        allow-query     { any; };
        forward first;
        forwarders { 10.10.73.5; 10.10.73.9; };
        recursion yes;
        dnssec-validation no;
        managed-keys-directory "/var/named/dynamic";
        pid-file "/run/named/named.pid";
        session-keyfile "/run/named/session.key";
        include "/etc/crypto-policies/back-ends/bind.config";
};
logging {
        channel default_debug {
                file "data/named.run";
                severity dynamic;
        };
};
zone "." IN {
        type hint;
        file "named.ca";
};
include "/etc/named.rfc1912.zones";
include "/etc/named.root.key";
```

#в текущей версии RED OS 7.3 управлять демоном bind пытается служба **rndc**, а ей для этого нужны ключи.

Разработчик инструкции: П.В. Сметанин
тел. (34922) 2-23-42

#их можно создать командой:

sudo rndc-confgen -c /dev/urandom

#после чего отредактируем файл */etc/bind/named.conf* указав путь к сформированным ключам

sudo vim /etc/named.conf

```
…
include "/etc/named.rfc1912.zones";
include "/etc/named.root.key";
include "/etc/rndc.key";
controls {
inet 127.0.0.1 port 953
allow { 127.0.0.1; } keys { "rndc-key"; };
};
```

#в итоге должно получиться:

```
options {
        listen-on port 53 { 127.0.0.1; 192.168.0.10; };
        directory       "/var/named";
        dump-file       "/var/named/data/cache_dump.db";
        statistics-file "/var/named/data/named_stats.txt";
        memstatistics-file "/var/named/data/named_mem_stats.txt";
        allow-query     { any; };
        forward first;
        forwarders { 10.10.73.5; 10.10.73.9; };
        recursion yes;
        dnssec-validation no;
        managed-keys-directory "/var/named/dynamic";
        pid-file "/run/named/named.pid";
        session-keyfile "/run/named/session.key";
        include "/etc/crypto-policies/back-ends/bind.config";
};
logging {
        channel default_debug {
                file "data/named.run";
                severity dynamic;
        };
};
zone "." IN {
        type hint;
        file "named.ca";
};
include "/etc/named.rfc1912.zones";
include "/etc/named.root.key";
include "/etc/rndc.key";
```

Разработчик инструкции: П.В. Сметанин
тел. (34922) 2-23-42

controls {
inet 127.0.0.1 port 953
allow { 127.0.0.1; } keys { "rndc-key"; };
};
#сохраните и закройте файл, после чего дайте ему права на чтение.

sudo chmod 644 /etc/rndc.key

#затем проверьте синтаксис файла конфигурации.

sudo named-checkconf

#если ничего не выдал, то все в порядке. Перезапустим службу bind9 и смотрим статус:

sudo systemctl restart named

sudo systemctl status named

● named.service - Berkeley Internet Name Domain (DNS)
Loaded: **loaded** (/usr/lib/systemd/system/named.service; enabled; vendor preset: disabled)
Active: **active (running)** since Fri 2021-12-17 17:45:42 +05; 59s ago
Process: 8384 ExecStartPre=/bin/bash -c if [! "$DISABLE_ZONE_CHECKING" == "yes"]; then /usr/sbin/named-checkconf -z "$NAMEDCONF"; else
Process: 8386 ExecStart=/usr/sbin/named -u named -c ${NAMEDCONF} $OPTIONS (code=exited, status=0/SUCCESS)
Main PID: 8387 (named)
Tasks: 20 (limit: 2330)
Memory: 49.0M
CPU: 166ms
CGroup: /system.slice/named.service
└─8387 /usr/sbin/named -u named -c /etc/named.conf
дек 17 17:46:20 PROXY001.PTO.LOCAL named[8387]: validating login.live.com/CNAME: bad cache hit (com/DS)
дек 17 17:46:20 PROXY001.PTO.LOCAL named[8387]: broken trust chain resolving 'login.live.com/A/IN': 10.10.73.5#53
дек 17 17:46:24 PROXY001.PTO.LOCAL named[8387]: validating login.live.com/CNAME: bad cache hit (com/DS)
дек 17 17:46:24 PROXY001.PTO.LOCAL named[8387]: broken trust chain resolving 'login.live.com/A/IN': 10.10.73.9#53
дек 17 17:46:24 PROXY001.PTO.LOCAL named[8387]: validating detectportal.firefox.com/CNAME: bad cache hit (com/DS)
…

#проверяем статус службы rndc.
#если нет всяких Warning-ов, то всё в порядке.

sudo rndc status

version: **BIND 9.16.16-RH** (Stable Release) <id:0c314d8>
running on PROXY001.PTO.LOCAL: Linux x86_64 5.10.29-1.el7.x86_64 #1 SMP Mon Apr 12 13:55:18 MSK 2021
boot time: Fri, 17 Dec 2021 12:45:42 GMT
last configured: Fri, 17 Dec 2021 12:45:42 GMT
configuration file: /etc/named.conf
CPUs found: 6
worker threads: 6
UDP listeners per interface: 6
number of zones: 103 (97 automatic)
debug level: 0

Разработчик инструкции: П.В. Сметанин
тел. (34922) 2-23-42

xfers running: 0
xfers deferred: 0
soa queries in progress: 0
query logging is OFF
recursive clients: 0/900/1000
tcp clients: 0/150
TCP high-water: 0
server is up and running

#если всё в порядке, проверяем как сам сервер (и машины локальной сети, у которых он указан в качестве DNS) разрешают DNS запросы:

nslookup ya.ru 192.168.0.10

Server: **192.168.0.10**
Address: **192.168.0.10#53**
Non-authoritative answer:
Name: ya.ru
Address: 87.250.250.242
Name: ya.ru
Address: 2a02:6b8::2:242

#смотрим прослушивается ли порт 53 внутренним ip-адресом нашего сервера **192.168.0.10**:

netstat -ntulp

```
Proto Recv-Q Send-Q Local Address           Foreign Address         State       PID/Program name
tcp        0      0 192.168.0.10:53         0.0.0.0:*               LISTEN      -
tcp        0      0 127.0.0.53:53           0.0.0.0:*               LISTEN      -
tcp        0      0 0.0.0.0:22              0.0.0.0:*               LISTEN      -
…
udp        0      0 192.168.0.10:53         0.0.0.0:*                           -
…
```

5. Установка и настройка Squid с HTTPS и фильтрацией

5.1. Установка

#установите программу

sudo dnf install squid

5.2. Подготовка сертификатов

#Чтобы работал HTTPS, необходимо сгенерировать сертификаты. Для этого создаем папку для хранения сертификатов, например, в */etc/squid/sslcert*:

sudo mkdir /etc/squid/sslcert

#переходим эту папку

cd /etc/squid/sslcert

#генерируем ключ:

sudo openssl genrsa -out /etc/squid/sslcert/squid.key

#создаем csr-запрос, используя ключ

sudo openssl req -new -key /etc/squid/sslcert/squid.key -out /etc/squid/sslcert/squid.csr

нужно будет указать всю необходимую информацию для csr (естественно у вас будут свои данные):

Country Name (2 letter code) [AU]:**RU**
State or Province Name (full name) [Some-State]:**YANAO**
Locality Name (eg, city) []:**Salekhard**
Organization Name (eg, company) [Internet Widgits Pty Ltd]:**PTO**
Organizational Unit Name (eg, section) []:**IT**
Common Name (e.g. server FQDN or YOUR name) []:**PROXY001**
Email Address []:**it@pto.yanao.ru**
Password

#подписываем сертификат самим собой:

sudo openssl x509 -req -days 3650 -in /etc/squid/sslcert/squid.csr -signkey /etc/squid/sslcert/squid.key -out /etc/squid/sslcert/squid.pem

#генерируем корневой доверительный сертификат, который нужно будет добавить на компьютер клиента:

sudo openssl x509 -in /etc/squid/sslcert/squid.pem -outform DER -out squid.der

#далее нужно дать права на папку с сертификатами для службы squid:

sudo chown -R :squid /etc/squid/sslcert

5.3. Настройка файла Squid.conf

#открываем файл настроек:

sudo vim /etc/squid/squid.conf

#устанавливаем правила доступа для локальной сети, если у вас две или более сети, укажите их одна под другой.
acl localnet src **192.168.0.0/24**
#**http_access** устанавливает разрешения на доступ, чтобы разрешить весь трафик, добавляем следующую строчку:
#важно, чтобы она была выше запрещающей **http_access deny all** иначе магия не сработает.
http_access allow all
#находим строку с **http_port 3128** и под ней дописываем строки:
#создание прозрачного HTTP-порта 3129:
http_port 3129 intercept
#создание порта 3130 для https-трафика с генерацией сертификатов по ключу:
https_port 3130 intercept ssl-bump generate-host-certificates=on dynamic_cert_mem_cache_size=4MB cert=/etc/squid/sslcert/squid.pem key=/etc/squid/sslcert/squid.key
#отключаем проверку сертификатов:
sslproxy_cert_error allow all
#перенаправляем весь трафик на squid:
always_direct allow all
#устанавливаем безопасное соединение с сервером, затем с клиентом, используя имитированный сертификат сервера:
ssl_bump server-first all
#устанавливаем TCP-туннель, не расшифровывая прокси-трафик:
ssl_bump none all

Разработчик инструкции: П.В. Сметанин
тел. (34922) 2-23-42

#указываем расположение программы генерации сертификатов и устанавливаем кэширование:
sslcrtd_program /usr/lib64/squid/security_file_certgen -s /var/lib/ssl_db -M 4MB
#сохраняем и закрываем файл
#дальше нужно пересоздать базу данных сертификатов:
rm -rf /var/lib/ssl_db
sudo /usr/lib64/squid/security_file_certgen -c -s /var/lib/ssl_db -M 4MB
#устанавливаем службу squid владельцем на папку базы данных сертификатов:
sudo chown -R squid:squid /var/lib/ssl_db
#добавляем политики SELinux:
sudo semanage fcontext -a -t squid_conf_t '/var/lib/ssl_db/index.txt'
sudo semanage fcontext -a -t squid_conf_t '/var/lib/ssl_db/size'
#применяем политики:
sudo /sbin/restorecon -v /var/lib/ssl_db/index.txt
sudo /sbin/restorecon -v /var/lib/ssl_db/size
#запускаем squid:
sudo systemctl start squid
#проверяем работоспособность squid:
sudo systemctl status squid
#Squid должен быть активен.
#добавляем squid в автозагрузку:
sudo systemctl enable squid
#вновь открываем файл настроек:
sudo vim /etc/squid/squid.conf
#настройте директорию для кэша, где: **ufs** - файловая система (ufs для SQUID является самой подходящей), **/var/spool/squid** — директория хранения кэша, **4096** — объем пространства в мегабайтах, которое будет выделено под кэш, **32** — количество каталогов первого уровня, которое будет создано для размещения кэша, **256** — количество каталогов второго уровня, которое будет создано для размещения кэша.
cache_dir ufs /var/spool/squid 4096 32 256
#теперь создайте структуру папок под кэш:
sudo squid -z
#перезапускаем squid:
sudo systemctl restart squid
#вновь открываем файл настроек:
sudo vim /etc/squid/squid.conf
#настраиваем контроль доступа к сайтам
#после строки **acl Safe_ports port 777** прописываем:
acl CONNECT method CONNECT
acl BLOCKED url_regex -i '/etc/squid/blocklist'
#тем самым создается список доступа к сайтам.
#далее перед строкой **http_access allow localnet** для блокировки списка доступа прописываем:

Разработчик инструкции: П.В. Сметанин
тел. (34922) 2-23-42

```
http_access deny BLOCKED
#в результате должно получиться следующее:
#locale
acl localnet src 192.168.0.0/24
acl localnet src 192.168.3.0/24
acl localnet src 192.168.4.0/24
acl localnet src 192.168.5.0/24
#local ports
acl SSL_ports port 443
acl Safe_ports port 80          # http
acl Safe_ports port 21          # ftp
acl Safe_ports port 443         # https
acl Safe_ports port 70          # gopher
acl Safe_ports port 210         # wais
acl Safe_ports port 1025-65535  # unregistered ports
acl Safe_ports port 280         # http-mgmt
acl Safe_ports port 488         # gss-http
acl Safe_ports port 591         # filemaker
acl Safe_ports port 777         # multiling http
#setting
acl CONNECT method CONNECT
acl BLOCKED url_regex -i '/etc/squid/blocklist'
http_access deny BLOCKED
http_access allow all
http_access deny !Safe_ports
http_access deny CONNECT !SSL_ports
http_access allow localhost manager
http_access deny manager
http_access allow localnet
http_access allow localhost
http_access deny all
#network
http_port 3128
http_port 3129 intercept
https_port 3130 intercept ssl-bump generate-host-certificates=on dynamic_cert_mem_cache_size=4MB cert=/etc/squid/sslcert/squid.pem key=/etc/squid/sslcert/squid.key
#certificate and cache
sslproxy_cert_error allow all
always_direct allow all
ssl_bump server-first all
ssl_bump none all
sslcrtd_program /usr/lib64/squid/security_file_certgen -s /var/lib/ssl_db -M 4MB
cache_dir ufs /var/spool/squid 100 16 256
```

Разработчик инструкции: П.В. Сметанин
тел. (34922) 2-23-42

coredump_dir /var/spool/squid
refresh_pattern ^ftp: 1440 20% 10080
refresh_pattern ^gopher: 1440 0% 1440
refresh_pattern -i (/cgi-bin/|\?) 0 0% 0
refresh_pattern . 0 20% 4320
#сохраняем и закрываем файл.
#создаем файл со списком сайтов для блокировки:
sudo vim /etc/squid/blocklist
#например:
facebook.com
vk.com
youtube.com
#если нужен более развёрнутый список, вот моя подборка https://disk.yandex.ru/d/Twr7ZS9ccXzcTg
#сохраняем файл.
#перечитываем конфигурацию squid:
sudo systemctl reload squid
#блокировка сайтов настроена, осталось перенаправить трафик на сервер squid.

5.4. Перенаправление трафика

#включаем ip_forwarding для разрешения проходящего трафика через сервер:
sudo su
echo 1 >> /proc/sys/net/ipv4/ip_forward
exit
#затем открываем файл */usr/lib/sysctl.d/50-default.conf* и прописываем в самом конце строку:
sudo vim /usr/lib/sysctl.d/50-default.conf
net.ipv4.ip_forward = 1
#добавляем правила для iptables.
#ВАЖНО! Если добавить эти правила, все запросы из локальной сети по портам **80** и **443** будут перенаправляться на порты squid **3129** и **3130** автоматически.
#если вы хотите включать и выключать прокси на клиентской машине вручную, указывая в настройках сети **192.168.0.10:3128** эти правила добавлять не нужно.
sudo iptables -t nat -A PREROUTING -p tcp --dport 80 -j REDIRECT --to-port 3129
sudo iptables -t nat -A PREROUTING -p tcp --dport 443 -j REDIRECT --to-port 3130
#выполним сохранение правил
sudo service iptables save
iptables: Saving firewall rules to /etc/sysconfig/iptables:[OK]
#лишним не будет сохранить правила в файл:
sudo iptables-save > iptables.backup
#Перезапускаем squid и проверяем статус:
sudo systemctl restart squid
sudo systemctl status squid

Разработчик инструкции: П.В. Сметанин
тел. (34922) 2-23-42

```
● squid.service - Squid caching proxy
     Loaded: loaded (/usr/lib/systemd/system/squid.service; enabled; vendor preset: disabled)
     Active: active (running) since Thu 2021-12-23 17:52:15 +05; 1h 34min ago
       Docs: man:squid(8)
    Process: 13101 ExecStartPre=/usr/libexec/squid/cache_swap.sh (code=exited, status=0/SUCCESS)
   Main PID: 13103 (squid)
      Tasks: 9 (limit: 2330)
     Memory: 35.8M
        CPU: 2.152s
     CGroup: /system.slice/squid.service
             ├─13103 /usr/sbin/squid --foreground -f /etc/squid/squid.conf
             ├─13106 (squid-1) --kid squid-1 --foreground -f /etc/squid/squid.conf
             ├─13107 (security_file_certgen) -s /var/lib/ssl_db -M 4MB
             ├─13108 (security_file_certgen) -s /var/lib/ssl_db -M 4MB
             ├─13109 (security_file_certgen) -s /var/lib/ssl_db -M 4MB
             ├─13110 (security_file_certgen) -s /var/lib/ssl_db -M 4MB
             ├─13111 (security_file_certgen) -s /var/lib/ssl_db -M 4MB
             ├─13112 (logfile-daemon) /var/log/squid/access.log
             └─13113 (unlinkd)
дек 23 17:52:15 PROXY001.PTO.LOCAL systemd[1]: Starting Squid caching proxy...
дек 23 17:52:15 PROXY001.PTO.LOCAL squid[13103]: 2021/12/23 17:52:15| WARNING: BCP 177 violation. Detected
дек 23 17:52:15 PROXY001.PTO.LOCAL squid[13103]: Squid Parent: will start 1 kids
дек 23 17:52:15 PROXY001.PTO.LOCAL squid[13103]: Squid Parent: (squid-1) process 13106 started
дек 23 17:52:15 PROXY001.PTO.LOCAL systemd[1]: Started Squid caching proxy.
```

6. Настройка клиентской машины

#большинство сайтов используют технологию HSTS для предотвращения MiTM-атак, поэтому если вы хотите настроить Squid для фильтрации трафика в своей организации, вам следует добавить ранее сформированный сертификат.

#для разных операционных систем сертификаты различаются, для Windows - это сертификат **squid.der**, для Linux - **squid.pem.**

#оба сертификата расположены на прокси сервере, по пути: */etc/squid/sslcert/*

#ВАЖНО! Если в п.10.4. вы добавили правила для iptables, то после добавления сертификата прокси сервер сразу начнёт работать. Если нет, то вам нужно указать в настройках сети или браузера адрес и порт вашего прокси сервера, у меня это **192.168.0.10** порт **3128**.

6.1. В браузере Chrome

#в меню браузера переходим по пути: Настройки -> Конфиденциальность и безопасность -> Безопасность -> Настроить сертификаты -> Центры сертификации -> Импорт.

#устанавливаем все галочки и нажимаем «ОК».

6.2. В браузере Mozilla Firefox

#в меню браузера переходим по пути: Настройки -> Приватность и защита -> Сертификаты -> Просмотр сертификатов -> Центры сертификации -> Импортировать

Разработчик инструкции: П.В. Сметанин
тел. (34922) 2-23-42

6.3. На Windows машине

#установим на машину **Windows** программу по **SSH** копированию файлов **pscp.exe** по ссылке: *https://www.chiark.greenend.org.uk/~sgtatham/putty/latest.html*

#откроем командную строку и введём команды:

cd c:

pscp localadmin@ip_адрес_прокси_сервера:/etc/squid/sslcert/squid.der c:\конечная_директория

localaadmin@192.168.0.10's password:

squid.der | 0 kB | 1.0 kB/s | ETA: 00:00:00 | 100%

#сочетание клавиш Win+R -> mmc -> Файл -> Добавить или удалить оснастку… -> Сертификаты -> Добавить -> учётной записи компьютера -> Готово -> Ок

#слева в консоли находим Сертификаты -> Доверенные корневые центры сертификации -> Сертификаты -> ПКМ на пустое пространство -> Все задачи -> Импорт…

#находим скачанный сертификат и устанавливаем

6.4. Средствами GPO на Windows Server в Active Directory

#скопируйте сертификат с разрешением ***.der** на ваш контроллер домена. Можно воспользоваться инструкцией выше.

#сочетание клавиш Win+R -> gpmc.msc -> выбираем домен -> ПКМ по Объекты групповой политики -> Создать -> укажите Имя групповой политики -> ОК

#ПКМ по созданной GPO -> Изменить -> в редакторе GPO перейдите в раздел Конфигурация компьютера -> Политики -> Конфигурация Windows -> Параметры безопасности -> Политики открытого ключа -> Доверенные корневые центры сертификации -> в левой части окна ПКМ -> Импорт…

#укажите путь к сертификату, который вы скачали -> размещаем в Доверенные корневые центры сертификации -> ОК

#политика распространения сертификатов создана. Протестируйте её, выполнив на клиенте обновление GPO командой **gpupdate /force**.

#если вы хотите, чтобы политика распространения сертификата применялась только к определённым пользователям или группам, выберите в консоли Объекты групповой политики созданную вами политику. На вкладке **Область** в секции **Фильтры безопасности** удалите группу **Прошедшие проверку пользователи** и добавьте вашу группу или пользователя.

III. ДОПОЛНИТЕЛЬНЫЕ НАСТРОЙКИ

7. Просмотр статистики прокси сервера Squid

#чтобы посмотреть логи всех подключений, выполните

sudo cat /var/log/squid/access.log

#чтобы отфильтровать конкретный ip-адрес

sudo cat /var/log/squid/access.log | grep ip_адрес

#записать вывод в файл

sudo cat /var/log/squid/access.log | grep ip_адрес > squid.log

#также статистику можно просматривать различными анализаторами, например, LightSquid, Free-SA, SARG и др., у каждого есть свои преимущества и недостатки, поэтому я не стану расписывать здесь какой-либо из них, этой информации полно в Интернете.

8. Резервное копирование и восстановление системы

#воспользуемся системой резервного копирования TimeShift, т.к. это наиболее приемлемый вариант для резервирования системных директорий Linux

#установите программу

sudo dnf -y install timeshift

Разработчик инструкции: П.В. Сметанин
тел. (34922) 2-23-42

#Программы - Системные - Резервное копирование и восстановление (далее снимок)
в нашем случае используется файловая система ext4, поэтому выбираем тип снимков RSYNC - Далее - Выберите диск для снимка - Далее - укажите расписание и количество сохранённых снимков - Далее - по необходимости включите снимки домашних каталогов - Далее - Готово
#программа настроена, чтобы выполнить выполнить снимок нажмите Создать.
#все снимки хранятся в директории /run/timeshift/backup
#для восстановления запустите программу TimeShift - выберите нужный снимок - Восстановить - выберите устройство, куда будут восстановлены файлы - Далее
#по окончании восстановления, перезагрузите сервер

9. Журналирование

#содержит глобальные системные логи, в том числе те, которые регистрируются при запуске системы. В этот лог записываются несколько типов сообщений: почта, cron, сервисы, ядро, аутентификация и другие:

/var/log/messages

#содержит информацию, которая регистрируется при загрузке системы.

/var/log/boot.log

#отображает информацию о последней сессии всех пользователей. Это нетекстовый файл, для его просмотра необходимо использовать команду **lastlog**.

/var/log/lastlog

#лог файл Linux содержит информацию о неудачных попытках входа в систему. Для просмотра файла удобно использовать команду **sudo last -f /var/log/btmp**

/var/log/btmp

#регистрирует всю информацию об установке пакетов с помощью **Dnf**.

/var/log/dnf.log

#содержит информацию, относящуюся к аутентификации и авторизации. Например, **SSHd** регистрирует здесь все, в том числе неудачные попытки входа в систему.

/var/log/secure

#содержит журнал входов пользователей в систему. С помощью команды ***last -f /var/log/wtmp*** можно узнать кто и когда вошел в систему.

/var/log/wtmp

#содержит информацию и журналы файлового сервера **Samba**, который используется для подключения к общим папкам **Windows**.

/var/log/samba/

#содержит **.cap** файлы, собранные пакетом **Sysstat**.

/var/log/sa/

#используется системным демоном безопасности, который управляет удаленным доступом к каталогам и механизмами аутентификации.

/var/log/sssd/

#примеры просмотра логов:
#обычный просмотр.

sudo cat /var/log/dnf.log

#просмотр в режиме прокрутки.

sudo less /var/log/dnf.log

#просмотр первых 10 строк.

sudo head /var/log/dnf.log

#просмотр последних 10 строк.

sudo tail /var/log/dnf.log

Разработчик инструкции: П.В. Сметанин
тел. (34922) 2-23-42

#просмотр в режиме реального времени.

sudo tail -F /var/log/dnf.log

#просмотр в режиме редактирования.

sudo vim /var/log/dnf.log или ***nano /var/log/dnf.log***

#выводит только строки, содержащие слово **gui**. Вместо **gui** можно подставить любое другое слово.

sudo cat /var/log/dnf.log | grep gui

#в графическом виде все журналы можно просматривать так: *Системные -> Просмотр системных журналов*

10. Проверка состояния жёсткого диска

#открываем терминал и вводим:

#устанавливаем программу

sudo dnf –y install gnome-disk-utility

#Далее заходим: *Стандартные -> Диски -> Меню (три точки) -> Данные самодиагностики и SMART.*

11. Использование памяти диска

#покажет все разделы с указанием свободного места

df –h

#скачиваем программу графической работы с дисками

sudo apt-get install gparted

#Далее: *Приложения -> Системные -> GParted*

12. Мониторинг температуры процессора.

#устанавливаем программу

sudo dnf -y install lm_sensors

#определяем аппаратную часть системы.

#Везде соглашаемся «**y**»

sudo sensors-detect

#запускаем программу

sensors

13. Информация о процессоре.

#полная информация о процессоре

lscpu

#скачиваем программу

sudo dnf –y install hwloc

#блочная информация о процессоре

Разработчик инструкции: П.В. Сметанин
тел. (34922) 2-23-42

hwloc-ls

14. Информация об оперативной памяти.

#краткая информация об оперативной памяти

free –h

#полная информация об оперативной памяти

cat /proc/meminfo

#ещё один способ узнать информацию об оперативной памяти

vmstat –s

15. Информация о системе.

#скачиваем программу

sudo dnf -y install lshw

#запускаем

sudo lshw –short

#скачиваем программу

sudo dnf -y install inxi

#запускаем

inxi -F

#в графической оболочке это можно сделать через: *Параметры -> Информация о системе*

16. Мониторинг работы системы

#делает снимок всех процессов в системе

ps axu

#чтобы убить процесс

kill PID

#найти определённый процесс

ps axu | ***grep*** *название приложения*

#убить все процессы этого приложения

killall *название приложения*

#в графической оболочке это можно сделать через: *Системные -> Системный монитор*

IV. РЕШЕНИЕ ПРОБЛЕМ

Разработчик инструкции: П.В. Сметанин
тел. (34922) 2-23-42

17. Восстановление программного RAID

#если вы создали программный RAID уровня 1 (зеркало), то при выходе из строя одного из дисков, вся информация сохранится на втором носителе.
#для восстановления выключите сервер (если не поддерживается горячая замена) замените вышедший из строя диск на новый и запустите систему.
#убедитесь, что диск определился в системе

sudo fdisk -l

#если у вас 2 диска, то новый будет назван как **/dev/sdb** и будет не размечен
#выполняем копирование диска /dev/sda на диск /dev/sdb

sudo -d /dev/sda | sfdisk /dev/sdb

#проверяем блочность

sudo lsblk -f

#смотрим соответствие зеркала и раздела. Например, зеркало **md1** - раздел **sda3**, зеркало **md0** - раздел **sda2**, зеркало **md2** - раздел **sda4** (у вас может быть своё присвоение)

cat /proc/mdstat

#добавляем зеркало к разделам второго диска

sudo mdadm -add /dev/md1 /dev/sdb3
sudo mdadm -add /dev/md2 /dev/sdb4
sudo mdadm -add /dev/md0 /dev/sdb2

#ждём окончания копирования и смотрим результат

cat /proc/mdstat

#должен получиться примерно такой вывод, где **2/2** означает зеркало **RAID 1**:

```
Personalities : [raid1]
md125 : active raid1 sdb3[1] sda3[0]
      15711232 blocks super 1.2 [2/2] [UU]
      bitmap: 1/1 pages [4KB], 65536KB chunk
          md126 : active raid1 sda2[0] sdb2[1]
      1047552 blocks super 1.2 [2/2] [UU]
      bitmap: 0/1 pages [0KB], 65536KB chunk
md127 : active raid1 sdb1[1] sda1[0]
      4195328 blocks super 1.2 [2/2] [UU]
      bitmap: 0/1 pages [0KB], 65536KB chunk
unused devices: <none>
```

18. Убрать иконку пользователя при запуске системы

#открываем командную строку и вводим:
#логинимся под root

su –

#заходим в директорию логирования и находим нужного пользователя, редактируем файл

cd /var/lib/AccountsService/users/
ls –la
vim smetaninpv

Разработчик инструкции: П.В. Сметанин
тел. (34922) 2-23-42

#меняем значение false на **true**
SystemAccount=**true**
#перелогиниваемся. Лишней иконки быть не должно.
#Если пользователь, чью иконку отключаем уволился, то его запись можно удалить
su –
cd /var/lib/AccountsService/users/
rm -rf имя_пользователя
#если не требуется хранить его документы, то удаляем всю домашнюю директорию пользователя
cd /home/
rm –Rfv пользователь@домен

Разработчик инструкции: П.В. Сметанин
тел. (34922) 2-23-42