

Сервер бухгалтерии



Содержание

Постановка задачи.....	3
Предварительные замечания.....	5
Аппаратное обеспечение.....	5
Установка системы и разметка диска.....	8
Настройка BIOS.....	9
Базовая настройка сервера.....	10
Защита загрузчика GRUB2.....	10
Установка доменного имени системы.....	11
Отключение IPv6.....	11
Отключение Magic SysRq.....	12
Поддержка туннелей GRE.....	12
Создание пользователей.....	13
Настройка списков доступа.....	14
Отключение ненужных служб.....	16
Настройка списка репозиторийев.....	17
Установка вспомогательного ПО.....	18
Настройка подсистемы UDEV.....	19
Настройка.....	19
Защита от падений (автоматическое возобновление).....	22
Проверка работоспособности и устойчивости.....	23
Настройка сети.....	24
Настройка сетевых интерфейсов.....	24
Маршрутизация.....	28
Управление пропускной способностью сети.....	29
Проверка работоспособности резервных каналов.....	31
Служба DNS/DHCP.....	31
Настройка.....	31
Защита от падений (автоматическое возобновление).....	37
Проверка работоспособности и устойчивости.....	38
Служба точного времени.....	41
Настройка.....	41
Защита от падений (автоматическое возобновление).....	45
Проверка работоспособности и устойчивости.....	46

Служба удалённого управления SSH.....	46
Настройка.....	47
Защита от падений (автоматическое возобновление).....	48
Проверка работоспособности и устойчивости.....	48
Настройка NUT.....	50
Настройка.....	50
Защита от падений (автоматическое возобновление).....	55
Проверка работоспособности и устойчивости.....	57
Настройка RAID1.....	61
Настройка.....	62
Защита от падений (автоматическое возобновление).....	64
Проверка работоспособности и устойчивости.....	66
Настройка Samba.....	66
Настройка.....	67
Защита от падений (автоматическое возобновление).....	70
Проверка работоспособности и устойчивости.....	71
Настройка резервного копирования.....	74
О дополнительных службах.....	77
AppArmor/SELinux.....	77
Защита от подбора паролей (fail2ban).....	78
Ротация журналов (logrotate).....	78
Служба точного времени (ntp).....	78
Почтовая служба.....	79
Автоматическое выключение системы.....	80
Установка и настройка сетевого фильтра (nftables).....	81
Установка сетевого фильтра.....	81
Сетевое окружение и потенциальные угрозы.....	81
Структура сетевого фильтра.....	83
Транзит пакетов и поддержка туннелей.....	85
Запуск и проверка сетевого фильтра.....	85
Завершение установки.....	100
Использование и обслуживание сервера.....	100
Порядок входа в систему.....	101
При обновлении системы.....	101
Мониторинг состояния аппаратных средств.....	101
Информация о сетевых адаптерах.....	102
Подключение клиентов.....	104
Обслуживание дискового массива.....	107
Замена устаревшего ключа локального репозитория.....	109

Постановка задачи

Требуется создать сервер бухгалтерии школы согласно следующей схеме сети:

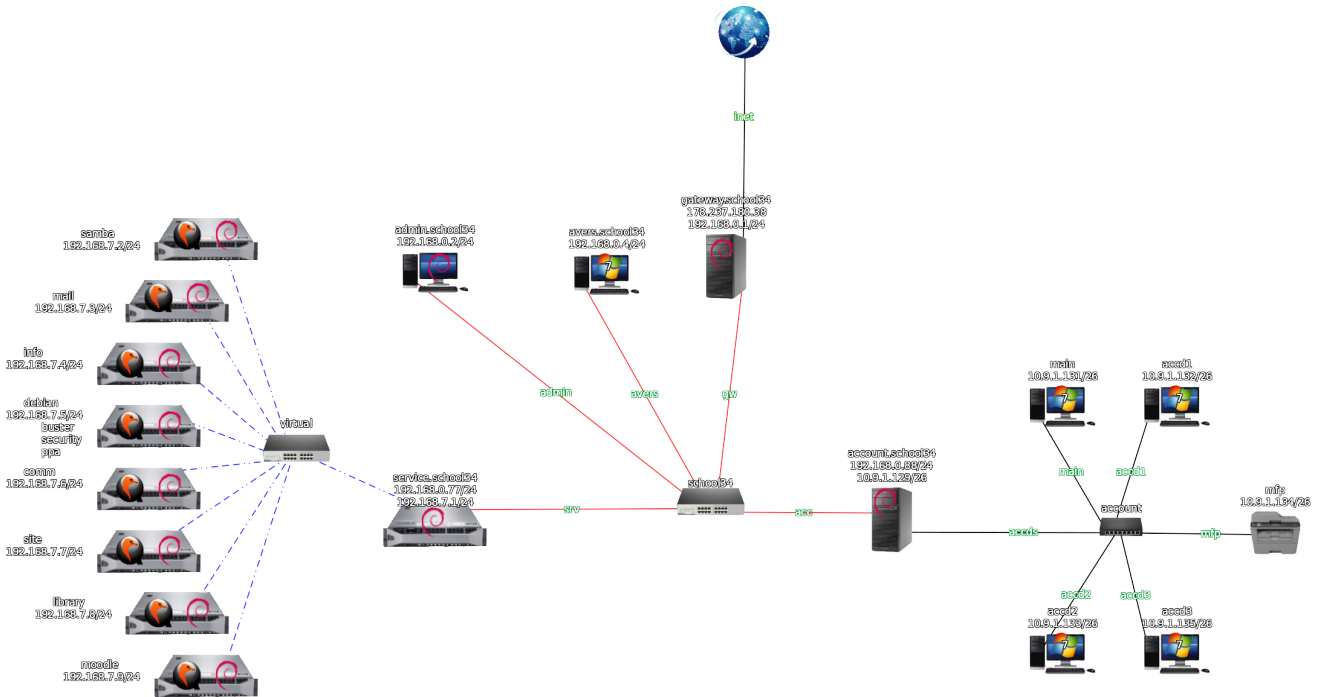


Рисунок 1: Фрагмент схемы сети

На данной схеме отображена видимая серверу бухгалтерии (server.account.school34) часть сети школы с указанием доменных имён и IPv4-адресов некоторых узлов. Большая часть сети скрыта, т. к. не должна быть видима ни рабочим станциям бухгалтерии, ни её серверу. На рисунке представлен также сервер виртуализации, обслуживающий домен service.school34. На базе этого сервера развернут ряд виртуальных машин, которые предоставляют разные сетевые сервисы клиентам сети школы. В ядре сети школы следует отметить сервер АИС «Аверс» и компьютер системного администратора.

Сервер должен предоставлять услуги DHCP только указанным клиентским устройствам на своём внутреннем интерфейсе, выдавая клиентам постоянные IPv4 адреса с идентификацией клиента по адресу канального уровня (lladdr, MAC-адрес). При этом адреса должны выдаваться в подсети 10.9.1.128/26 с учётом того, что адрес 10.9.1.129 занят самим сервером.

Сервер должен предоставлять клиентам услуги службы доменных имён, пересылая запросы о домене `service.school34` (192.168.7.0/24) к `server.service.school34`, о домене `school34` – к `gateway.school34`, а все остальные запросы – ко внешнему DNS-серверу. При этом внешний DNS-сервер должен быть постоянным с минимальной фильтрацией (YandexDNS), использование фильтрующего DNS-сервера (например, SkyDNS) не требуется.

Сервер должен предоставлять клиентам услуги службы точного времени, используя для собственной синхронизации gateway.school34, который является также сервером точного времени уровня школы и имеет псевдоним ntp.school34.

Сервер должен быть доступен для удалённого управления только с компьютера системного администратора (admin.school34).

Сервер должен предоставлять возможность быстрого переключения на резервное мобильное соединение с Интернет и обратно силами сотрудников бухгалтерии без участия системного администратора, при этом набор мобильных устройств должен быть строго ограничен и заранее известен серверу. При таком переключении также должны автоматически перестраиваться правила маршрутизации и сетевой фильтрации.

Для организации файлового обмена сервер должен предоставлять сетевой каталог, обеспечивая доступ к нему по протоколу smb (Samba, CIFS). Этот каталог должен быть доступен для чтения и записи всем рабочим станциям бухгалтерии.

Для организации системы резервного копирования сервер должен обладать программным RAID1-массивом, аппаратно обеспеченным двумя жёсткими дисками одной модели. На этом массиве должна быть сформирована файловая система для поддержки каталога резервного копирования.

Система резервного копирования должна состоять из двух сетевых каталогов и работать следующим образом. В сетевой каталог, расположенный на системном жёстком диске, помещаются файлы, которые следует занести в архив. В конце рабочего дня эти файлы объединяются в архив, который переносится (именно переносится, а не копируется) на RAID-массив. При этом список файлов, помещённых в архив дописывается в конец файла, играющего роль каталога-справочника по резервному хранилищу. Этот файл также хранится на RAID-массиве, имеет простой формат (html) и может быть получен клиентом для поиска нужного файла и архива, его содержащего. Содержимое RAID-массива также доступно по протоколу smb, но только для чтения. Такая схема работы позволяет защититься от вредоносного ПО, которое может попасть на рабочие станции бухгалтерии, например от шифровальщиков. Разумеется, рабочие станции, управляемые ОС Windows, защищены антивирусами и файерволами, но дополнительный рубеж обороны лишним не будет, учитывая что при появлении нового шифровальщика есть риск его проникновения в защищаемую сеть раньше, чем его сигнатура будет внесена в антивирусные базы. Что же касается управления резервным копированием, то оно, в силу специфики работы бухгалтерии в конкретной организации, выполняется вручную, т. е. сами бухгалтера решают, что и когда им необходимо сохранить на долговременной основе.

Межсетевой экран в своей работе должен отталкиваться от запрещающей политики по умолчанию. Должны быть разрешены практически любые исходящие соединения от рабочих станций бухгалтерии в любые внешние сети, соединения, устанавливаемые в ответ на них, а также соединения в рамках GRE-туннеля, необходимого для работы ПО на одной строго определённой заранее рабочей станции бухгалтерии. При этом сетевое МФУ не должно иметь возможности отправлять пакеты за пределы сети бухгалтерии. Также сеть бухгалтерии не должна иметь связь с не отображёнными на вышеприведённой схеме подсетями организации, а сервер должен иметь статический набор ARP-записей для всех клиентских устройств и доступных серверов школы.

Как уже было отмечено выше, при переключении на резервное Интернет-соединение и обратно изменения в таблицах маршрутизации и правилах фильтрации должны выполняться автоматически и своевременно. При необходимости также должны вноситься изменения и в ARP-таблицы.

Наконец, подходящим образом должна быть организована система энергообеспечения сервера и сетевого оборудования. Сервер должен получать энергию от источника

бесперебойного питания и иметь с ним информационный канал связи с тем, чтобы при отключении питания в энергосети организации ИБП мог своевременно уведомить сервер о низком уровне заряда батарей и инициировать корректное выключение сервера. При этом к этому же ИБП должен быть подключён и коммутатор бухгалтерии. Тем самым в совокупности с ИБП рабочих станций обеспечивается работоспособность практически всей подсети бухгалтерии, за исключением МФУ. Это позволяет штатно остановить работу сетевого программного обеспечения.

Также следует принять адекватные задаче низкоуровневые меры защиты сервера. При этом от шифрования файловых систем следует отказаться в силу физического расположения сервера в пределах помещений бухгалтерии и во избежание неоправданного снижения производительности. Также не предусматривается использование дисковых квот по той простой причине, что сетевые каталоги вынесены на отдельные файловые системы и управляются от имени единственного пользователя.

Предварительные замечания

Во фрагментах терминальных сеансов и при указании выполняемых команд используются следующие соглашения:

Выполняемые при настройке системы команды выделены жирным шрифтом, отклик системы или выдержки из конфигурационных файлов и сценариев такого выделения не имеют. Все команды приводятся с полным отображением приглашения командной строки, в котором отражается пользователь, от имени которого выполняется команда, и текущий рабочий каталог. Оба этих фактора значимы в большинстве приведённых команд и листингов.

Аппаратное обеспечение

В качестве платформы для реализации сервера бухгалтерии выбран стационарный компьютер из имеющегося машинного парка с некоторыми доработками. Далее в сокращённом виде приведены сведения об аппаратуре этого компьютера:

```
root@server:/# lshw
server
  description: Desktop Computer
  product: G41MT-S2P
  vendor: Gigabyte Technology Co., Ltd.
  width: 64 bits
  capabilities: smbios-2.4 dmi-2.4 smp vsyscall32
  configuration: boot=normal chassis=desktop uuid=00000000-0000-0000-0000-
1C6F65C2C56F
*-core
  description: Motherboard
  product: G41MT-S2P
  vendor: Gigabyte Technology Co., Ltd.
...
*-cpu
  description: CPU
  product: Pentium(R) Dual-Core CPU E5500 @ 2.80GHz
  vendor: Intel Corp.
  physical id: 4
  bus info: cpu@0
  version: Pentium(R) Dual-Core CPU E5500 @
  slot: Socket 775
```

```

size: 1286MHz
capacity: 4GHz
width: 64 bits
clock: 200MHz
capabilities: fpu fpu_exception wp vme de pse tsc msr pae mce cx8 apic sep
mtrr pge mca cmov pat pse36 clflush dts acpi mmx fxsr sse sse2 s
s ht tm pbe syscall nx x86-64 constant_tsc arch_perfmon pebs bts rep_good nopl cpuid
aperfmpperf pni dtes64 monitor ds_cpl vmx est tm2 ssse3 cx16 xtpr
pdc_m xsave lahf_lm pti tpr_shadow vnmi flexpriority dtherm cpufreq

```

*-cache:0

```

description: L1 cache
physical id: a
slot: Internal Cache
size: 64KiB
capacity: 64KiB
capabilities: synchronous internal write-back
configuration: level=1

```

*-cache:1

```

description: L2 cache
physical id: b
slot: External Cache
size: 2MiB
capacity: 2MiB
capabilities: synchronous internal write-back
configuration: level=2

```

*-memory

```

description: System Memory
physical id: 18
slot: System board or motherboard
size: 2GiB

```

*-bank:0

```

description: DIMM 400 MHz (2,5 ns)
physical id: 0
slot: A0
size: 2GiB
width: 196 bits
clock: 400MHz (2.5ns)

```

*-bank:1

```

description: DIMM [empty]
physical id: 1
slot: A1

```

*-bank:2

```

description: DIMM [empty]
physical id: 2
slot: A2

```

*-bank:3

```

description: DIMM [empty]
physical id: 3
slot: A3

```

*-pci

...

*-network

```

description: Ethernet interface
product: RTL8111/8168/8411 PCI Express Gigabit Ethernet Controller
vendor: Realtek Semiconductor Co., Ltd.
physical id: 0
bus info: pci@0000:02:00.0
logical name: enp2s0
version: 06
serial: 18:d6:c7:00:ea:6c
size: 1Gbit/s
capacity: 1Gbit/s

```

```

        width: 64 bits
        clock: 33MHz
        capabilities: pm msi pciexpress msix vpd bus_master cap_list ethernet
physical tp mii 10bt 10bt-fd 100bt 100bt-fd 1000bt 1000bt-fd autonegotiation
        configuration: autonegotiation=on broadcast=yes driver=r8169
duplex=full ip=192.168.0.88 latency=0 link=yes multicast=yes port=MII speed=1Gbit/s
        resources: irq:17 ioport:ce00(size=256) memory:fdefff000-fdefffff
memory:fddfc000-fddfffff
*-network DISABLED
        description: Ethernet interface
        product: AR8151 v1.0 Gigabit Ethernet
        vendor: Qualcomm Atheros
        physical id: 0
        bus info: pci@0000:03:00.0
        logical name: enp3s0
        version: c0
        serial: 1c:6f:65:c2:c5:6f
        capacity: 1Gbit/s
        width: 64 bits
        clock: 33MHz
        capabilities: pm msi pciexpress vpd bus_master cap_list ethernet
physical tp 10bt 10bt-fd 100bt 100bt-fd 1000bt-fd autonegotiation
        configuration: autonegotiation=on broadcast=yes driver=atl1c
driver version=1.0.1.1-NAPI latency=0 link=no multicast=yes port=twisted pair
        resources: irq:18 memory:fdcc0000-fdcfffff ioport:df00(size=128)
...
*-ide
        description: IDE interface
        product: NM10/ICH7 Family SATA Controller [IDE mode]
        vendor: Intel Corporation
        physical id: 1f.2
        bus info: pci@0000:00:1f.2
        logical name: scsi0
        version: 01
        width: 32 bits
        clock: 66MHz
        capabilities: ide pm isa_compat_mode bus_master cap_list emulated
        configuration: driver=ata_piix latency=0
        resources: irq:19 ioport:1f0(size=8) ioport:3f6 ioport:170(size=8)
ioport:376 ioport:f800(size=16)
*-disk
        description: ATA Disk
        product: WDC WD3200AAKS-0
        vendor: Western Digital
        physical id: 0.0.0
        bus info: scsi@0:0.0.0
        logical name: /dev/sda
        version: 3A01
        serial: WD-WMAT10439126
        size: 298GiB (320GB)
        capabilities: partitioned partitioned:dos
        configuration: ansiversion=5 logicalsectorsize=512 sectorsize=512
signature=6b449e09
...

```

В качестве комментариев к приведённым сведениям необходимо отметить следующее:

Сервер обладает вполне достаточными для выполнения поставленных задач центральным процессором, объёмом и характеристиками оперативной памяти, жёсткими дисками, сетевыми адаптерами и прочими системами.

В данной выдержке не отражены жёсткие диски, предназначенные для формирования RAID-массива, они будут описаны отдельно в соответствующем разделе.

Установка системы и разметка диска

Установка системы выполнена без какой либо графической оболочки, но со стандартными системными утилитами и службой ssh.

Системный жёсткий диск разбит на разделы следующим образом (в выводе ниже опущены строки, не касающиеся жёсткого диска, а сами команды выполнены на уже настроенной системе с установленными утилитами):

```
root@server:/# /usr/sbin/fdisk -l /dev/sda
```

```
Disk /dev/sda: 298,1 GiB, 320071851520 bytes, 625140335 sectors
```

```
Disk model: WDC WD3200AAKS-0
```

```
Units: sectors of 1 * 512 = 512 bytes
```

```
Sector size (logical/physical): 512 bytes / 512 bytes
```

```
I/O size (minimum/optimal): 512 bytes / 512 bytes
```

```
Disklabel type: dos
```

```
Disk identifier: 0x6b449e09
```

Device	Boot	Start	End	Sectors	Size	Id	Type
/dev/sda1	*	2048	39063551	39061504	18,6G	83	Linux
/dev/sda2		39063552	46876671	7813120	3,7G	82	Linux swap / Solaris
/dev/sda3		46876672	85938175	39061504	18,6G	83	Linux
/dev/sda4		85938176	625139711	539201536	257,1G	83	Linux

```
root@server:/# /sbin/parted -l /dev/sda
```

```
Model: ATA WDC WD3200AAKS-0 (scsi)
```

```
Disk /dev/sda: 320GB
```

```
Sector size (logical/physical): 512B/512B
```

```
Partition Table: msdos
```

```
Disk Flags:
```

Number	Start	End	Size	Type	File system	Flags
1	1049kB	20,0GB	20,0GB	primary	ext4	boot
2	20,0GB	24,0GB	4000MB	primary	linux-swap(v1)	
3	24,0GB	44,0GB	20,0GB	primary	ext4	
4	44,0GB	320GB	276GB	primary	ext4	

```
root@server:/# mount -l
```

```
/dev/sda1 on / type ext4 (rw,relatime,errors=remount-ro) [rootfs]
```

```
/dev/sda3 on /var type ext4 (rw,nosuid,nodev,relatime) [var]
```

```
/dev/sda4 on /srv/samba type ext4 (rw,nosuid,nodev,noexec,relatime) [samba]
```

```
root@server:/# df -h
```

Файловая система	Размер	Использовано	Дост	Использовано%	Смонтировано в
udev	965M	0	965M	0%	/dev
tmpfs	196M	5,5M	191M	3%	/run
/dev/sda1	19G	885M	17G	6%	/
tmpfs	980M	0	980M	0%	/dev/shm
tmpfs	5,0M	0	5,0M	0%	/run/lock
tmpfs	980M	0	980M	0%	/sys/fs/cgroup
/dev/sda3	19G	256M	18G	2%	/var
/dev/sda4	253G	61M	240G	1%	/srv/samba
tmpfs	196M	0	196M	0%	/run/user/1000

Такая схема разметки дисков выбрана по следующим соображениям:

DOS-таблица является наиболее простой и достаточной для рассматриваемого сервера, особенно с учётом года выпуска аппаратной части. На раздел подкачки выделено 4 Гб, остальное дисковое пространство разбито на три раздела, благодаря чему каталог изменяемых данных /var и сетевые каталоги размещены в отдельных файловых системах. Это сделано в рамках базовых мер по защите сервера. Большая часть диска отведена именно под сетевые каталоги, тогда как для остальных компонентов системы выделено минимально достаточное (с некоторым запасом) дисковое пространство.

При загрузке системы были выявлены попытки обращения ядра к дисководу гибких дисков:

```
root@server:/# dmesg | grep fd0
[ 24.880285] print_req_error: I/O error, dev fd0, sector 0
[ 47.704340] print_req_error: I/O error, dev fd0, sector 0
[ 74.660564] print_req_error: I/O error, dev fd0, sector 0
root@server:/# lsmod | grep floppy
floppy                86016  0
```

Поскольку такого дисковода в системе нет, а возможность отключения присутствующего на материнской плате контроллера не предусмотрена в BIOS системы, то наиболее правильным решением является отключение соответствующего модуля ядра. Для этого достаточно внести этот модуль в «чёрный» список и обновить образ initramfs, после чего потребуется (и для проверки в том числе) перезагрузить систему. При этом из-за сценариев, вызываемых при выполнении обновления загрузочного образа, следует дополнить значение переменной окружения PATH. Все эти действия продемонстрированы в следующем фрагменте терминального сеанса:

```
root@server:/# echo "blacklist floppy" > /etc/modprobe.d/blacklist.conf
root@server:/# PATH=$PATH:/sbin
root@server:/# update-initramfs -u
update-initramfs: Generating /boot/initrd.img-4.19.0-16-amd64
root@server:/# reboot
```

После перезагрузки убедиться в отсутствии сообщений о проблемах, а также о том, что модуль ядра не загружен, можно командами:

```
root@server:/# dmesg | grep fd0
root@server:/# lsmod | grep floppy
```

Настройка BIOS

BIOS установленной в системе материнской платы позволяет выполнить следующие существенные для работы сервера настройки:

- защита паролем входа в настройки BIOS;
- включение PME Event Wake Up (включение системы по внутреннему таймеру в начале рабочего дня);
- AC Back Function Full-On (автоматическое включение системы после возврата потерянного энергопитания);
- отключение USB Keyboard/Mouse/Storage Function, что при этом не мешает работе источника бесперебойного питания.

Базовая настройка сервера

В этом разделе описаны меры, применимые ко многим подобным серверам. В частности рассмотрены способы минимизации числа служб, настройки сетевой подсистемы и некоторых систем безопасности.

Защита загрузчика GRUB2

Защита загрузчика не представляется избыточной, даже не смотря на то, что сервер физически расположен в помещении с ограниченным доступом. Эта защита легко реализуется и не ведет к возрастанию нагрузки на сервер в нормальном режиме его работы.

В первую очередь требуется создать отдельный конфигурационный файл, в котором будут аккумулярованы параметры, реализующие эту защиту. Его следует расположить в каталоге **/etc/grub.d**, а его имя должно начинаться с такой числовой константы, чтобы он был обработан последним. Содержимое этого файла должно соответствовать шаблону, представленному в файле **/etc/grub.d/40_custom**. Таким образом, в рассматриваемой системе файл имеет имя **/etc/grub.d/40_custom** и заполнен следующим образом:

```
#!/bin/sh
exec tail -n +3 $0
set superusers="root"
password_pbkdf2 root PBKDF2
```

При этом литерал PBKDF2 – это лишь заглушка для контрольной суммы пароля, дающего доступ к редактированию параметров загрузки от имени пользователя root.

По готовности этого файла следует ограничить доступ к этому файлу и заменить заглушку на реальную контрольную сумму командами (после первой команды потребуется дважды ввести пароль для GRUB):

```
root@server:/etc/grub.d# PBKDF2=$(grub-mkpasswd-pbkdf2 | grep -o grub.\*)
root@server:/etc/grub.d# sed s/PBKDF2/$PBKDF2/ -i /etc/grub.d/42_custom
root@server:/etc/grub.d# chmod u+x,go-rw /etc/grub.d/42_custom
```

Затем в конфигурационном файле **/etc/default/grub** следует привести некоторые строки к следующему виду:

```
GRUB_TIMEOUT=1
GRUB_DISABLE_RECOVERY="true"
```

Тем самым из основного меню GRUB удаляются все вспомогательные пункты, а также устанавливается интервал в одну секунду для инициации перехода к редактированию параметров загрузки.

Предпоследним шагом следует обновить меню загрузчика при помощи команд:

```
root@server:/etc/grub.d# PATH=$PATH:/sbin
root@server:/etc/grub.d# update-grub
```

Первая из них изменяет в текущей сессии значение системной переменной PATH для обеспечения работы второй, которая, как раз, и формирует конечный конфигурационный файл **/boot/grub/grub.cfg**, который и используется непосредственно загрузчиком. Несмотря на то, что этот файл запрещено редактировать вручную, т. к. все изменения в

нём теряются при очередном вызове команды `update-grub`, иного выхода не остаётся, поскольку имеющиеся на данный момент механизмы настройки GRUB2 не предоставляют иного механизма для загрузки системы в штатном режиме без ввода установленного пароля.

Чтобы сделать такую загрузку возможной, следует лишь дописать в строку меню GRUB2, используемую по умолчанию, ключ `--unrestricted`, что приведёт эту строку к подобному виду:

```
menuentry 'Debian GNU/Linux' --unrestricted --class debian --class gnu-linux ...
```

Подобную правку следует выполнять при каждом обновлении меню GRUB2, в том числе если оно случается в процессе установки обновлений.

Установка доменного имени системы

С учётом того, что в системе не предполагается использование протокола IPv6, для настройки полного имени домена следует сначала заполнить файл `/etc/hosts`, затем внести имя системы без доменной части в файл `/etc/hostname` и выполнить команду для смены имени хоста в текущей сессии. Все необходимые команды, а также содержимое конфигурационных файлов представлены в нижеследующем фрагменте терминального сеанса. Последняя команда в нём демонстрирует настроенное полное доменное имя системы (FQDN):

```
root@server:/# cat /etc/hosts
127.0.0.1      localhost.localdomain localhost
127.0.1.1      server.account.school34 server
root@server:/# cat /etc/hostname
server
root@server:/# hostname -f
server.account.school34
```

Отключение IPv6

Отключение поддержки IPv6 ядром системы может быть выполнено как для текущей сессии, так и на постоянной основе. Для отключения на постоянной основе следует привести содержимое файла `/etc/sysctl.conf` к виду, не противоречащему следующим строкам в нём:

```
#Disabling IPv6
net.ipv6.conf.all.disable_ipv6=1
net.ipv6.conf.default.disable_ipv6=1
net.ipv6.conf.lo.disable_ipv6=1
```

Для отключения только в текущей сессии достаточно выполнить команды:

```
root@server:/# echo '1' > /proc/sys/net/ipv6/conf/all/disable_ipv6
root@server:/# echo '1' > /proc/sys/net/ipv6/conf/default/disable_ipv6
root@server:/# echo '1' > /proc/sys/net/ipv6/conf/lo/disable_ipv6
```

Для проверки успешности отключения следует перезапустить сетевую подсистему командой

```
root@server:/# systemctl restart networking
```

и в выводе команды

```
root@server:/# ip address show
```

убедиться в отсутствии назначенных адресов IPv6:

```
1: lo: <LOOPBACK,UP,LOWER_UP> mtu 65536 qdisc noqueue state UNKNOWN group default
qlen 1000
    link/loopback 00:00:00:00:00:00 brd 00:00:00:00:00:00
    inet 127.0.0.1/8 scope host lo
        valid_lft forever preferred_lft forever
2: enp2s0: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc pfifo_fast state UP group
default qlen 1000
    link/ether 18:d6:c7:00:ea:6c brd ff:ff:ff:ff:ff:ff
    inet 192.168.0.88/24 brd 192.168.0.255 scope global enp2s0
        valid_lft forever preferred_lft forever
3: enp3s0: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc pfifo_fast state UP group
default qlen 1000
    link/ether 1c:6f:65:c2:c5:6f brd ff:ff:ff:ff:ff:ff
    inet 10.9.1.129/26 brd 10.9.1.191 scope global enp3s0
        valid_lft forever preferred_lft forever
```

Замечание: здесь представлен вывод при полностью настроенной сетевой подсистеме. Описание таких настроек приведено ниже, здесь достаточно отметить лишь отсутствие у интерфейсов IPv6-адресов.

Отключение Magic SysRq

Отключение Magic SysRq может показаться избыточным или даже вредным: так, например, в случае сбоя системы не будет возможности её аварийно остановить или сбросить содержимое буферов на диск. Однако в силу физической доступности сервера пользователям, есть, пусть и призрачная, вероятность того, что некий злоумышленник выполнит нажатие подобной комбинации клавиш. Это может привести к остановке сервера и отказам в обслуживании клиентов. При этом защита организовывается настолько просто, что пренебрегать ею не рационально.

Для отключения «волшебных» сочетаний в текущей сессии достаточно команды:

```
root@server:/# echo '0' > /proc/sys/kernel/sysrq
```

Для отключения на постоянной основе достаточно привести файл `/etc/sysctl.conf` к виду, не противоречащему следующей строке из него

```
kernel.sysrq=0
```

Поддержка туннелей GRE

Для работы одного из клиентских приложений на одной из рабочих станций используется туннелирование при помощи протокола GRE. Для обеспечения функционирования такого туннеля требуется обеспечить загрузку необходимых модулей ядра и поддержку со стороны межсетевого экрана. Здесь описаны только меры, относящиеся к модулям ядра, настройка сетевого фильтра приведены далее в соответствующем разделе.

Существует два подхода к загрузке модулей: при включении системы и в процессе её работы по мере необходимости. Исходя из характера функционирования

рассматриваемой сети наиболее рационально применить первый подход. Для этого достаточно внести дополнения в файл **/etc/modules** :

```
#Loading modules needed for GRE tunnel
ip-gre
ip-nat-pptp
```

Создание пользователей

Далее при описании службы защищённого удалённого входа `ssh` указано, что вход в систему удалённо от имени суперпользователя заблокирован. Чтобы всё же иметь возможность удалённого управления, следует создать обычного пользователя, от имени которого и будет осуществляться вход. После открытия сессии в системе под такой учётной записью появляется, при необходимости, возможность повысить свои полномочия при помощи `su`. Ограничения на системные ресурсы для этого пользователя на данный момент не вводятся, однако по опыту эксплуатации системы это может быть сделано в дальнейшем.

Создание такого пользователя (пусть его логин – `administrator`) можно выполнить как на этапе установки системы, так и после, например, при помощи команды

```
root@server:/# /usr/sbin/adduser administrator
```

Кроме создания пользователя, необходимо обратить внимание на несколько аспектов его работы в системе.

Во-первых, по умолчанию вновь созданный домашний каталог этого пользователя доступен для чтения всем в системе. В случае, если это неприемлемо, исправить ситуацию можно при помощи команды

```
root@server:/# chmod 0750 /home/administrator
```

оставив, таким образом, полный доступ к каталогу для самого пользователя и доступ на чтение для членов его группы (коих кроме него самого в системе всё равно нет). Для всех остальных доступ окажется закрыт.

Вторым аспектом является использование `sudo`. Эта команда позволяет выполнять команды с административными привилегиями, не входя в систему от имени суперпользователя. Такой подход таит в себе некоторую опасность: злоумышленник, зная логин и пароль такого пользователя может получить весьма обширные права в системе (если политики `sudo` настроены) или практически неограниченные, если настройки `sudo` приняты по умолчанию. Для решения этой проблемы проще всего (и наиболее оптимально на подобном сервере) воспользоваться следующим подходом: не использовать `sudo` вообще, вынуждая пользователя для повышения своих прав в системе вводить (и, соответственно, знать) пароль суперпользователя.

Таким образом, чтобы лишить пользователя возможности использования `sudo` достаточно исключить его из группы `wheel`, если он в ней состоит. Сделать это можно командой `usermod`, оставив пользователя `administrator` только в своей (одноимённой) группе. Убедиться в успешности изменений можно выполнив от имени этого пользователя строго после повторного входа в систему команду `groups`. Далее приведён фрагмент терминального сеанса, демонстрирующий все эти операции:

```
root@server:/# /usr/sbin/usermod -G administrator administrator
root@server:/# su guest
administrator@server:/$ groups
administrator
administrator@server:/$ exit
exit
root@server:/#
```

Также для организации работы файлового сервера полезно создать отдельного пользователя (а в общем случае – пользователей), от имени которых будут выполняться все операции с сетевыми каталогами и их содержимым. Этот пользователь не должен иметь возможности входа в систему, ему не нужны командный интерпретатор и домашний каталог. Создание такого пользователя выполняется командой:

```
root@server:/# /sbin/adduser buh --shell /dev/null --no-create-home --disabled-login
```

Для тех же целей следует увеличить предельно допустимое количество одновременно открытых файловых дескрипторов для пользователя. Это связано с тем, что по умолчанию в Linux это значение равно 1024, тогда как в Windows оно равно 16384. Поскольку и клиенты рассматриваемого сервера работают под управлением ОС Windows, и системная служба smbd при своём запуске также устанавливает в системе это значение, то рационально установить его вручную заранее и задокументировать это изменение. Для этого следует внести в файл **/etc/security/limits.conf** следующие строки:

```
* - nofile 16384
root - nofile 16384
```

После чего систему следует перезагрузить. Конечно, можно изменить этот параметр и в текущей сессии системы, а перезагрузку выполнить позже. Для изменения предела только в текущей сессии достаточно выполнить команду:

```
root@server:/# ulimit -n 16384
```

Проверить успешность внесения изменений как в текущей сессии, так и после перезагрузки системы можно следующим образом:

```
root@server:/# ulimit -Hn -Sn
open files          (-n) 16384
open files          (-n) 16384
```

Настройка списков доступа

Многие демоны используют файлы **/etc/hosts.allow** и **/etc/hosts.deny** как источники информации о том, кому разрешено пользоваться услугами этих демонов. Эти конфигурационные файлы являются частью механизма tcp wrappers. Кроме того, в системе может быть установлен демон tcpd, который сам выполняет такую проверку, прежде чем передать сетевой пакет, инициирующий соединение, целевому демону.

Чтобы выяснить, использует ли демон некоторой службы этот механизм, как правило достаточно убедиться, что исполняемый файл демона собран с использованием библиотеки libwrap. Следующий фрагмент терминального сеанса демонстрирует, что демон sshd использует эту библиотеку, а dnsmasq, ntpd, компоненты NUT и smbd – нет:

```
root@server:/# ldd /sbin/upsd | grep libwrap
root@server:/# ldd /sbin/upsdrvctl | grep libwrap
```

```
root@server:/# ldd /sbin/upsmon | grep libwrap
root@server:/# ldd /sbin/upssched | grep libwrap
root@server:/# ldd /sbin/smbd | grep libwrap
root@server:/# ldd /sbin/ntpd | grep libwrap
root@server:/# ldd /sbin/dnsmasq | grep libwrap
root@server:/# ldd /sbin/sshd | grep libwrap
libwrap.so.0 => /lib/x86_64-linux-gnu/libwrap.so.0 (0x00007fa52ea65000)
```

Однако, несмотря на то, что программы подсистемы NUT не используют библиотеку libwrap, они, тем не менее, используют эти конфигурационные файлы.

Проверить, установлен ли в системе демон tcpd, можно командой

```
root@server:/# dpkg -l | grep tcpd
```

Как видно из пустого отклика команды, этот демон не установлен.

В общем случае такая проверка выполняется файерволом, а механизм tcp wrappers считается устаревшим и уступившим место как раз сетевым экранам. Тем не менее, учитывая простоту настройки и нетребовательность к ресурсам, имеет смысл внести в указанные конфигурационные файлы записи для всех используемых системой демонов, оставив, тем самым, решение по использованию этого механизма разработчикам демонов и сборщикам дистрибутива. Иными словами, если ими будет принято решение включить поддержку в новой версии ПО, то настраиваемая система будет автоматически готова к таким изменениям. Кроме того, никакое ПО не застраховано от ошибок разработчиков и уязвимостей. Использование этого механизма может оказаться дополнительным рубежом защиты при компрометации сетевого фильтра.

В данной системе не имеет смысла установка tcpd, т. к. ssh использует этот механизм самостоятельно, а остальные демоны просто не принимают подключений на внешнем интерфейсе.

Зная, что подключаться по протоколу ssh разрешено только с компьютера системного администратора, имеющего адрес 192.168.0.2, по протоколу службы бесперебойного питания только с петлевого интерфейса, а по протоколам служб ntpd, smbд/nmbд и dns/dhср – только с внутреннего и петлевого интерфейсов, следует привести файл **/etc/hosts.allow** к следующему виду (комментарии для краткости опущены):

```
ntpd : 10.9.1.128/26 127.0.0.1
nmbд : 10.9.1.128/26 127.0.0.1
smbд : 10.9.1.128/26 127.0.0.1
dnsmasq : 10.9.1.128/26 127.0.0.1
sshd : 192.168.0.2
upsd : upsmon@localhost
```

В свою очередь файл **/etc/hosts.deny** запрещает любые другие подключения к серверу (комментарии также не приведены):

```
ALL: ALL
```

Убедиться в том, что ssh не принимает соединений с посторонних хостов, можно перезапустив эту службу командой

```
root@server:/# systemctl restart sshd
```

и попытавшись подключиться по ssh с любого отличного от сервера подсети хоста. Разумеется, на момент эксперимента фаервол не должен блокировать такое подключение.

Безусловно настраивать подобные параметры безопасности следует до подключения сервера к сети, а окончательно проверять – сразу после, по возможности защищая проверяемый сегмент сети иными средствами (например фаерволом вышестоящего сервера организации).

Отключение ненужных служб

Соблюдая правило необходимой достаточности, следует отключить и/или удалить все неиспользуемые службы, например: rsh, telnet, rpcbind. Проверить, установлены ли эти службы, можно одной из или сочетанием следующих (и, возможно, некоторых дополнительных) команд:

```
root@server:/# dpkg -l | grep rpcbind
root@server:/# which rsh
root@server:/# ss -l | grep telnet
```

Ни один из этих способов не универсален в том смысле, что однозначно показывает наличие или отсутствие той или иной службы. Как пример можно привести следующий фрагмент терминального сеанса:

```
root@server:/# which rsh
/usr/bin/rsh
root@server:/# ss -Htlp
LISTEN 0 16 127.0.0.1:nut 0.0.0.0:* users:(("upsd",pid=477,fd=4))
LISTEN 0 50 10.9.1.129:netbios-ssn 0.0.0.0:* users:(("smbd",pid=1411,fd=34))
LISTEN 0 32 127.0.0.1:domain 0.0.0.0:* users:(("dnsmasq",pid=465,fd=9))
LISTEN 0 32 10.9.1.129:domain 0.0.0.0:* users:(("dnsmasq",pid=465,fd=7))
LISTEN 0 128 192.168.0.88:ssh 0.0.0.0:* users:(("sshd",pid=481,fd=3))
LISTEN 0 50 10.9.1.129:microsoft-ds 0.0.0.0:* users:(("smbd",pid=1411,fd=33))
root@server:/# dpkg -l | grep rsh
root@server:/# ls -l /usr/bin | grep rsh
lrwxrwxrwx 1 root root 21 мая 4 18:27 rsh -> /etc/alternatives/rsh
root@server:/# ls -l /etc/alternatives | grep rsh
lrwxrwxrwx 1 root root 12 мая 4 18:27 rsh -> /usr/bin/ssh
lrwxrwxrwx 1 root root 28 мая 4 18:27 rsh.1.gz -> /usr/share/man/man1/ssh.1.gz
root@server:/#
```

Разбирая приведённый фрагмент, можно отметить следующее:

Первая команда показывает, что в системе присутствует исполняемый файл rsh, однако вторая команда показывает, что демон rsh не прослушивает никаких tcp-портов. Следующая команда сообщает о том, что в системе не установлено ни одного пакета, в названии которого упоминается rsh. При просмотре в подробном режиме свойств исполняемого файла выясняется, что он является символической ссылкой на другой файл, который также является символической ссылкой на исполняемый файл ssh. Таким образом, в данной системе rsh не установлен, вместо этого создан псевдоним для ssh.

Среди прочего в системе был установлен пакет telnet, содержащий только клиент, но поскольку он не требуется для работы системы, то пакет следует удалить:

```
root@server:/# apt-get purge telnet
```


После перезапуска системы можно убедиться, что система теперь прослушивает только порты, необходимые ей для выполнения поставленных задач (в том числе здесь отражены службы, установка и настройка которых описаны далее в этом документе):

```
root@server:/# ss -tulp
Netid State  Local Address:Port      Peer Address:Port
udp  UNCONN  127.0.0.1:domain    0.0.0.0:*  users:(("dnsmasq",pid=465,fd=8))
udp  UNCONN  10.9.1.129:domain   0.0.0.0:*  users:(("dnsmasq",pid=465,fd=6))
udp  UNCONN  0.0.0.0:bootps      0.0.0.0:*  users:(("dnsmasq",pid=465,fd=4))
udp  UNCONN  192.168.0.88:ntp     0.0.0.0:*  users:(("ntpd",pid=451,fd=22))
udp  UNCONN  10.9.1.129:ntp      0.0.0.0:*  users:(("ntpd",pid=451,fd=18))
udp  UNCONN  127.0.0.1:ntp       0.0.0.0:*  users:(("ntpd",pid=451,fd=17))
udp  UNCONN  0.0.0.0:ntp         0.0.0.0:*  users:(("ntpd",pid=451,fd=16))
udp  UNCONN  10.9.1.191:netbios-ns 0.0.0.0:*  users:(("nmbd",pid=1421,fd=17))
udp  UNCONN  10.9.1.129:netbios-ns 0.0.0.0:*  users:(("nmbd",pid=1421,fd=16))
udp  UNCONN  0.0.0.0:netbios-ns   0.0.0.0:*  users:(("nmbd",pid=1421,fd=14))
udp  UNCONN  10.9.1.191:netbios-dgm 0.0.0.0:*  users:(("nmbd",pid=1421,fd=19))
udp  UNCONN  10.9.1.129:netbios-dgm 0.0.0.0:*  users:(("nmbd",pid=1421,fd=18))
udp  UNCONN  0.0.0.0:netbios-dgm  0.0.0.0:*  users:(("nmbd",pid=1421,fd=15))
udp  UNCONN  0.0.0.0:54728        0.0.0.0:*  users:(("dnsmasq",pid=465,fd=10))
tcp  LISTEN  127.0.0.1:nut        0.0.0.0:*  users:(("upsd",pid=477,fd=4))
tcp  LISTEN  10.9.1.129:netbios-ssn 0.0.0.0:*  users:(("smbd",pid=1411,fd=34))
tcp  LISTEN  127.0.0.1:domain     0.0.0.0:*  users:(("dnsmasq",pid=465,fd=9))
tcp  LISTEN  10.9.1.129:domain    0.0.0.0:*  users:(("dnsmasq",pid=465,fd=7))
tcp  LISTEN  192.168.0.88:ssh     0.0.0.0:*  users:(("ssh",pid=481,fd=3))
tcp  LISTEN  10.9.1.129:microsoft-ds 0.0.0.0:*  users:(("smbd",pid=1411,fd=33))
```

В представленном фрагменте колонки Recv-Q и Send-Q удалены исключительно из соображений форматирования ввиду их ничтожной в контексте изложения информативности.

Настройка списка репозиториев

Сразу после установки системы для возможности установки дополнительного программного обеспечения следует заполнить список репозиториев, используемых системой, зарегистрировать публичные ключи локальных репозиториев и настроить сетевые адаптеры системы. Описание настройки сетевых интерфейсов приведено ниже для сохранения стройности и целостности изложения. Нет препятствий тому, чтобы настроить список репозиториев и установить дополнительное программное обеспечение после настройки проводной сети.

Конфигурационный файл `/etc/apt/sources.list`, содержащий список репозиториев, следует привести к виду:

```
deb http://debian.service.school34/buster/ buster main contrib non-free
deb http://debian.service.school34/security/ buster/updates main contrib
deb http://debian.service.school34/ppa/ buster main contrib
```

Затем необходимо получить публичную часть ключа локального репозитория (ppa), например, загрузив из него же файл `repository_key.asc` командой:

```
root@server:/# cd /tmp && wget http://debian.service.school34/ppa/repository_key.asc
```

Для использования этого ключа в системе должен присутствовать пакет `gnupg`. Если его нет, т. е. команда

```
root@server:/tmp# dpkg -l | grep gnupg
```

выдаёт пустой отклик, то его следует установить. Для этого следует выполнить следующие команды:

```
root@server:/tmp# apt-get update
root@server:/tmp# apt-get install gnupg
```

При этом первая из этих команд ожидаемо выдаст предупреждение о том, что ключ локального репозитория `ppa` не может быть проверен и поэтому этот репозиторий не будет использован.

Далее файл ключа локального репозитория следует зарегистрировать в системе управления пакетами командой:

```
root@server:/tmp# apt-key add repository_key.asc
```

После этого уже возможно обновить сведения о пакетах, хранящихся в указанных репозиториях:

```
root@server:/tmp# apt-get update
```

Импортировать ключи остальных репозиторий не нужно, т. к. они являются зеркалами официальных репозиторий и установлены вместе с системой.

Установка вспомогательного ПО

Весьма полезными при управлении сервером могут оказаться файловый менеджер Midnight Commander (пакет `mc`) и утилиты `tree`, `lshw` и `parted`. Для их установки достаточно выполнить команду:

```
root@server:/# apt-get install mc tree lshw parted
```

Для контроля состояния сервера весьма полезны средства просмотра данных от аппаратных датчиков. Эти средства входят в состав пакета `lm-sensors`, который необходимо установить, а затем провести процесс определения имеющихся в системе датчиков. Это можно сделать, выполнив следующие команды и последовав появившимся при этом инструкциям:

```
root@server:/# apt-get install lm-sensors
root@server:/# sensors-detect
```

После чего данные этих датчиков можно получить при помощи команды

```
root@server:/# sensors
```

Причём эта команда может быть выполнена даже с правами непривилегированного пользователя.

Ещё одним средством контроля состояния системы является технология S.M.A.R.T – оценка состояния жёсткого диска (или иного носителя информации) встроенной аппаратурой самодиагностики.

Для использования этой технологии необходимо установить пакет `smartmontools`, содержащий утилиты, позволяющие получать данные от жёсткого диска, интерпретировать и отображать их, а также выполнять тестирование диска. Также в состав этого пакета входит демон `smartd` (и соответствующая служба `systemd`), который в

автоматическом режиме следит за состоянием диска и информирует системного администратора. Об этом подробнее сказано в разделе «Использование сервера».

Установить эти программные средства можно при помощи команды:

```
root@server:/# apt-get install smartmontools
```

Их применение показано в разделе «Использование сервера».

На системах с несколькими сетевыми интерфейсами (как в рассматриваемом случае) или при необходимости полного контроля над аппаратурой интерфейса полезна утилита ethtool. Для её установки достаточно выполнить команду:

```
root@server:/# apt-get install ethtool
```

Настройка подсистемы UDEV

В рассматриваемой системе для обеспечения надёжности и безопасности функционирования потребуется провести тонкую настройку подсистемы udev, которая оповещает системное программное обеспечение о событиях, происходящих с устройствами, управляет правами доступа к ним, а также может создавать дополнительные ссылки для устройств в системном каталоге **/dev** и переименовывать сетевые интерфейсы. Согласно постановке задачи средствами udev следует обеспечить защиту от подключения посторонних usb-устройств (иных разъёмов для подключения съёмных носителей информации в системе не предусмотрено аппаратно) для защиты от несанкционированного копирования данных, а также подключение ограниченного числа заранее известных смартфонов в режиме usb-модемов для организации резервного Интернет-канала с автоматическими переключениями на этот канал при подключении смартфона или модема и обратно на проводной интерфейс при отключении usb-устройства. Также по шине usb осуществляется взаимодействие с источником бесперебойного питания, что требует не только авторизации этого устройства, но некоторых дополнительных действий для правильного выбора драйвера ИБП.

Настройка

Далее описание настройки udev будет выполнено на примере одного тестового смартфона, при реальной настройке количество и модели смартфонов будут меняться.

В первую очередь следует собрать всю необходимую информацию о разрешённых устройствах. Для этого каждое устройство по очереди следует подключить к системе и обнаружить его командой:

```
root@server:/# lsusb
Bus 002 Device 004: ID 22b8:2e76 Motorola PCS
Bus 002 Device 001: ID 1d6b:0002 Linux Foundation 2.0 root hub
Bus 005 Device 001: ID 1d6b:0001 Linux Foundation 1.1 root hub
Bus 004 Device 001: ID 1d6b:0001 Linux Foundation 1.1 root hub
Bus 003 Device 001: ID 1d6b:0001 Linux Foundation 1.1 root hub
Bus 003 Device 005: ID 0764:0501 Cyber Power System, Inc. CP1500 AVR UPS
Bus 001 Device 001: ID 1d6b:0001 Linux Foundation 1.1 root hub
```

Затем командой вида

```
root@server:/# udevadm info -a --name=/dev/bus/usb/002/004 >> \
/etc/udev/rules.d/35-usb-filter.rules
```

занести подробную сводку об устройстве в файл **/etc/udev/rules.d/usb-filter.rules**. Этот файл имеет расширение **.rules** согласно требованиям демона **systemd-udev**, но на данный момент содержит не правила работы для этого демона, а информацию, необходимую для их построения.

Важно заметить, что смартфон следует переключить в режим **usb-модема** до сбора информации о нём, т.к. большинство смартфонов меняют свой идентификатор устройства в зависимости от режима работы.

Дальнейший набор правил построен по принципу запрета всего кроме явно разрешённого, поэтому для корректного функционирования ИБП и **usb-модемов** (или смартфонов, выступающих в их роли) разрешающие правила должны быть составлены не только для самих устройств но и для **usb-концентраторов**, к которым их допустимо подключать. В рассматриваемом случае разрешены для использования любые **usb-порты** системы, поэтому правила должны быть составлены для каждого **usb-концентратора**. Для сбора сведений о концентраторах можно воспользоваться аналогичными приведённой выше командами:

```
root@server:/# udevadm info -a --name=/dev/bus/usb/001/001 >> \
/etc/udev/rules.d/35-usb-filter.rules
root@server:/# udevadm info -a --name=/dev/bus/usb/002/001 >> \
/etc/udev/rules.d/35-usb-filter.rules
root@server:/# udevadm info -a --name=/dev/bus/usb/003/001 >> \
/etc/udev/rules.d/35-usb-filter.rules
root@server:/# udevadm info -a --name=/dev/bus/usb/004/001 >> \
/etc/udev/rules.d/35-usb-filter.rules
root@server:/# udevadm info -a --name=/dev/bus/usb/005/001 >> \
/etc/udev/rules.d/35-usb-filter.rules
```

Следует при этом отметить, что в результате выполнения этих команд полученные данные будут избыточны как по составу атрибутов устройств, так и по отображению сведений об устройстве №1 на шине №2 (Linux Foundation 2.0 root hub). Дело в том, что при формировании сводки о смартфоне в её состав попала и сводка о всей цепочке устройств, через которую этот смартфон подключён.

Далее следует преобразовать информацию для правил в сами правила. Для этого файл **/etc/udev/rules.d/35-usb-filter.rules** следует привести к виду, схожему с точностью до конкретных устройств со следующим:

```
#Enable only known devices
#Skip not USB
SUBSYSTEM!="usb", GOTO="usb_filter_end"
#Skip all action except add or change
ACTION!="add|change", GOTO="usb_filter_end"

#System usb 2.0 hubs
ATTR{product}=="EHCI Host Controller", ATTR{idVendor}=="1d6b",
ATTR{idProduct}=="0002", ATTR{authorized}="1", GOTO="usb_filter_end"

#System usb 1.1 hubs
ATTR{product}=="UHCI Host Controller", ATTR{idVendor}=="1d6b",
ATTR{idProduct}=="0001", ATTR{authorized}="1", GOTO="usb_filter_end"

#UPS Cyber Power
```

```
ATTR{product}=="UPS VALUE", ATTR{idVendor}=="0764", ATTR{idProduct}=="0501",  
ATTR{authorized}="1", GOTO="usb_filter_end"
```

```
#Smartphone No 1
```

```
ATTR{product}=="SAMSUNG_Android", ATTR{serial}=="HG786KJ5KK6",  
ATTR{idVendor}=="04e8", ATTR{idProduct}=="6863", ATTR{authorized}="1",  
GOTO="usb_filter_end"
```

```
#Smartphone No 2
```

```
ATTR{product}=="SAMSUNG_Android", ATTR{serial}=="jfhwkv83vbi3hg9343",  
ATTR{idVendor}=="04e8", ATTR{idProduct}=="6863", ATTR{authorized}="1",  
GOTO="usb_filter_end"
```

```
#Smartphone No 3
```

```
ATTR{product}=="HRY-LX1", ATTR{serial}=="UNHM56GVF564GR3", ATTR{idVendor}=="12d1",  
ATTR{idProduct}=="108a", ATTR{authorized}="1", GOTO="usb_filter_end"
```

```
#Smartphone No 4
```

```
ATTR{product}=="AT0LL-AB-IDP_SN:2AD1B6A8", ATTR{serial}=="ihv3h343",  
ATTR{idVendor}=="2717", ATTR{idProduct}=="ff80", ATTR{authorized}="1",  
GOTO="usb_filter_end"
```

```
#Disable all other usb devices
```

```
SUBSYSTEMS=="usb", ACTION=="add|change", ENV{DEVTYPE}=="usb_device",  
ATTR{authorized}="0"
```

```
LABEL="usb_filter_end"
```

Первыми строками этого файла указывается, что он обрабатывает только события add (подключение) или change (изменение) для устройств подсистемы usb, остальные события и подсистемы игнорируются этим набором правил. Затем для каждого устройства определяется ряд атрибутов, играющих роль уникального идентификатора устройства. В рассматриваемом контексте таким идентификатором принят кортеж, состоящий из названия устройства (ATTR{product}), его серийного номера (ATTR{serial}) и идентификаторов производителя (ATTR{idVendor}) и устройства (ATTR{idProduct}). Связыванием воедино этих данных с разрешением на использование устройства (ATTR{authorized}="1") формируется правило для udevd. Для источника бесперебойного питания сделано небольшое исключение: серийный номер устройства не учитывается. В конце файла выставляется запрет (ATTR{authorized}="0") на подключение любых других устройств.

Опираясь на данные из этого файла можно сформировать и второй файл **/etc/udev/rules.d/77-net-alias.rules** :

```
#Making common alias for usb interfaces
```

```
#Skip not net
```

```
SUBSYSTEM!="net", GOTO="net_alias_end"
```

```
#Skip all action except add or change
```

```
ACTION!="add|change", GOTO="net_alias_end"
```

```
#Smartphone No 1
```

```
ENV{ID_VENDOR_ID}=="04e8", ENV{ID_MODEL_ID}=="6863", NAME="ethusb",  
GOTO="net_alias_end"
```

```
#Smartphone No 2
```

```
ENV{ID_VENDOR_ID}=="04e8", ENV{ID_MODEL_ID}=="6863", NAME="ethusb",  
GOTO="net_alias_end"
```

```
#Smartphone No 3
```

```
ENV{ID_VENDOR_ID}=="12d1", ENV{ID_MODEL_ID}=="108a", NAME="ethusb",  
GOTO="net_alias_end"
```

```
#Smartphone No 4
ENV{ID_VENDOR_ID}=="2717", ENV{ID_MODEL_ID}=="ff80", NAME="ethusb",
GOTO="net_alias_end"

LABEL="net_alias_end"
```

Этот файл структурно аналогичен предыдущему и построен опираясь на рекомендации, приведённые в [/usr/share/doc/udev/README.Debian.gz](#). Также на основании этого документа выбрана и схема именования конфигурационных файлов udev. Так файл 77-net-alias.rules имеет префикс 77 с тем, чтобы быть обработанным после системного файла 75-net-descriptor.rules, но до системного 80-net-setup-link.rules. Файл 35-usb-filter.rules также имеет префикс 35, чтобы быть обработанным раньше других файлов подсистемы usb. Наконец третий файл **/etc/udev/rules.d/99-ups.rules** должен обрабатываться одним из последних и предотвращать захват ИБП, работающего по протоколу USB HID, соответствующим HID-драйвером, оставляя тем самым возможность для NUT взять устройство под свой контроль. Подробнее это описано ниже в разделе «Настройка NUT». Этот файл содержит лишь одно правило:

```
SUBSYSTEMS=="usb", DRIVERS=="usbhid", ACTION=="add", ATTR{idVendor}=="0764", \
ATTR{idProduct}=="0501", ATTR{authorized}="0"
```

После завершения редактирования конфигурационных файлов следует перезапустить демон udevd и убедиться в том, что он распознал новую конфигурацию:

```
root@server:/# systemctl restart systemd-udev
root@server:/# systemctl status systemd-udev
• systemd-udev.service - udev Kernel Device Manager
  Loaded: loaded (/lib/systemd/system/systemd-udev.service; static; vendor preset:
  enabled)
  Active: active (running) since Thu 2021-05-06 10:34:47 MSK; 8s ago
    Docs: man:systemd-udev.service(8)
          man:udev(7)
Main PID: 791 (systemd-udev)
  Status: "Processing with 15 children at max"
    Tasks: 1
   Memory: 1.1M
    CGroup: /system.slice/systemd-udev.service
            └─791 /lib/systemd/systemd-udev
```

```
мая 06 10:34:47 server systemd[1]: Starting udev Kernel Device Manager...
мая 06 10:34:47 server systemd[1]: Started udev Kernel Device Manager.
```

Далее обеспечения работы правил не только при «горячем» подключении устройств, но и при загрузке системы необходимо обновить образ initramfs:

```
root@server:/# PATH=$PATH:/sbin
root@server:/# update-initramfs -u
```

При этом изменение переменной окружения PATH потребовалось для обеспечения работоспособности сценариев, вызываемых в ходе работы update-initramfs.

Защита от падений (автоматическое возобновление)

Управлением службами в системе занимается менеджер загрузки и служб systemd, именно он и обеспечивает (при должной настройке) автоматическое возобновление служб в случае их падения. В рассматриваемой системе такая настройка для службы

systemd-udevд выполнена изначально сборщиками дистрибутива операционной системы.

Проверка работоспособности и устойчивости

Проверку правильности работы службы разумно провести по следующему сценарию:

- выключить систему;
- подключить к ней смартфон в режиме usb-модема, usb-клавиатуру и usb-накопитель;
- включить систему и убедиться, что в системе обнаружен сетевой интерфейс ethusb, но не обнаружены ни клавиатура, ни накопитель;
- отключить все устройства от системы и подключить их снова, убедившись в неизменности результата.

После включения системы и удалённого к ней подключения по протоколу ssh (описано далее) наблюдается желаемый результат: клавиатура получает питание по шине usb, но нажатия её кнопок системой игнорируются, флеш-накопитель хотя и подключён к системе, но как блочное устройство не распознаётся, смартфон опознан как сетевой интерфейс ethusb, что можно видеть из следующего фрагмента терминального сеанса:

```
root@server:/# lsusb
Bus 005 Device 004: ID 8564:1000 Transcend Information, Inc. JetFlash
Bus 005 Device 003: ID 22b8:2e25 Motorola PCS
Bus 005 Device 001: ID 1d6b:0002 Linux Foundation 2.0 root hub
Bus 004 Device 001: ID 1d6b:0001 Linux Foundation 1.1 root hub
Bus 003 Device 001: ID 1d6b:0001 Linux Foundation 1.1 root hub
Bus 002 Device 001: ID 1d6b:0001 Linux Foundation 1.1 root hub
Bus 001 Device 002: ID 1a2c:4c5e China Resource Semico Co., Ltd
Bus 001 Device 001: ID 1d6b:0001 Linux Foundation 1.1 root hub
root@server:/# lsblk
NAME MAJ:MIN RM SIZE RO TYPE MOUNTPOINT
sda 8:0 0 298,1G 0 disk
├─sda1 8:1 0 18,6G 0 part /
├─sda2 8:2 0 3,7G 0 part [SWAP]
├─sda3 8:3 0 18,6G 0 part /var
└─sda4 8:4 0 257,1G 0 part /srv/samba
root@server:/# ip link show
1: lo: <LOOPBACK,UP,LOWER_UP> mtu 65536 qdisc noqueue state UNKNOWN mode DEFAULT
group default qlen 1000
    link/loopback 00:00:00:00:00:00 brd 00:00:00:00:00:00
2: enp2s0: <BROADCAST,MULTICAST> mtu 1500 qdisc htb state DOWN mode DEFAULT group
default qlen 1000
    link/ether 18:d6:c7:00:ea:6c brd ff:ff:ff:ff:ff:ff
3: enp3s0: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc pfifo_fast state UP mode
DEFAULT group default qlen 1000
    link/ether 1c:6f:65:c2:c5:6f brd ff:ff:ff:ff:ff:ff
4: gre0@NONE: <NOARP> mtu 1476 qdisc noop state DOWN mode DEFAULT group default qlen
1000
    link/gre 0.0.0.0 brd 0.0.0.0
5: gretap0@NONE: <BROADCAST,MULTICAST> mtu 1476 qdisc noop state DOWN mode DEFAULT
group default qlen 1000
    link/ether 00:00:00:00:00:00 brd ff:ff:ff:ff:ff:ff
6: erspan0@NONE: <BROADCAST,MULTICAST> mtu 1464 qdisc noop state DOWN mode DEFAULT
group default qlen 1000
    link/ether 00:00:00:00:00:00 brd ff:ff:ff:ff:ff:ff
7: ethusb: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc pfifo_fast state UNKNOWN
mode DEFAULT group default qlen 1000
    link/ether 36:e1:49:c8:cb:1c brd ff:ff:ff:ff:ff:ff
```

После отключения устройств и их повторного подключения к работающей системе результат идентичен.

Настройка сети

В этом разделе описывается только настройка сетевых интерфейсов самой системы, описание системы межсетевого экранирования приведено далее. Настройку сети можно разделить на следующие этапы:

- установка необходимого программного обеспечения;
- настройка сетевых интерфейсов;
- настройка маршрутизации и транзита пакетов;
- настройка системы управления пропускной способностью;

Для полноценной работы одной из сетевых карт в рассматриваемой системе требуется установить пакет `firmware-realtek`. Это можно сделать сразу после настройки проводного сетевого интерфейса или перенесением пакета вручную со сменного носителя. При настроенном сетевом интерфейсе и списке репозиторийев выполнить установку можно командой:

```
root@server:/# apt-get install firmware-realtek
```

Остальные необходимые для настройки сети средства (`iproute2`, `tc`) установлены вместе с системой.

Настройка сетевых интерфейсов

В первую очередь требуется выяснить точный состав доступных системе сетевых интерфейсов, а также убедиться в том, что установлено всё программное обеспечение, необходимое для их работы, а также что сами интерфейсы переключены в наиболее производительные рабочие режимы. В следующем фрагменте терминального сеанса показано, что в системе присутствует одновременно семь сетевых интерфейсов: петлевой, два проводных, три виртуальных, связанных с туннелем GRE, и смартфон в режиме usb-модема:

```
root@server:/# ip link show
1: lo: <LOOPBACK,UP,LOWER_UP> mtu 65536 qdisc noqueue state UNKNOWN mode DEFAULT
group default qlen 1000
    link/loopback 00:00:00:00:00:00 brd 00:00:00:00:00:00
2: enp2s0: <BROADCAST,MULTICAST> mtu 1500 qdisc htb state DOWN mode DEFAULT group
default qlen 1000
    link/ether 18:d6:c7:00:ea:6c brd ff:ff:ff:ff:ff:ff
3: enp3s0: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc pfifo_fast state UP mode
DEFAULT group default qlen 1000
    link/ether 1c:6f:65:c2:c5:6f brd ff:ff:ff:ff:ff:ff
4: gre0@NONE: <NOARP> mtu 1476 qdisc noop state DOWN mode DEFAULT group default qlen
1000
    link/gre 0.0.0.0 brd 0.0.0.0
5: gretap0@NONE: <BROADCAST,MULTICAST> mtu 1476 qdisc noop state DOWN mode DEFAULT
group default qlen 1000
    link/ether 00:00:00:00:00:00 brd ff:ff:ff:ff:ff:ff
6: erspan0@NONE: <BROADCAST,MULTICAST> mtu 1464 qdisc noop state DOWN mode DEFAULT
group default qlen 1000
    link/ether 00:00:00:00:00:00 brd ff:ff:ff:ff:ff:ff
7: ethusb: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc pfifo_fast state UNKNOWN
```



```
mode DEFAULT group default qlen 1000
  link/ether 36:e1:49:c8:cb:1c brd ff:ff:ff:ff:ff:ff
```

Перед настройкой сетевых интерфейсов для повышения устойчивости функционирования сервера в случае ошибок коммутации на сетевом оборудовании требуется внести некоторые дополнения в файл **/etc/sysctl.conf** :

```
#Securing arp
net.ipv4.conf.all.arp_ignore=2
net.ipv4.conf.all.arp_announce=2
```

Строка `arp_ignore` означает, что `arp`-запрос на разрешение `ip`-адреса обслуживается только если он пришёл на интерфейс, работающий под этим адресом, причём `ip`-адрес отправителя запроса находится в той же подсети.

Строка `arp_announce` означает, что посылать исходящий `arp`-пакет можно только с того интерфейса, который содержит анонсируемый `ip`-адрес.

Тем самым, если, например, во внешней сети появится хост с адресом, соответствующим подсети бухгалтерии, он не создаст для сервера конфликта адресов.

Эти изменения будут применены после перезагрузки сервера. Для их принятия в текущей сессии следует выполнить команды вида:

```
root@server:/# echo 2 > /proc/sys/net/ipv4/conf/all/arp_ignore
root@server:/# echo 2 > /proc/sys/net/ipv4/conf/all/arp_announce
```

Настройка сетевых интерфейсов выполняется внесением необходимых сведений в файл **/etc/network/interfaces**. В данном случае он имеет вид (с некоторыми несущественными сокращениями):

```
source /etc/network/interfaces.d/*

# The loopback network interface
auto lo
iface lo inet loopback

# Main external interface
auto enp2s0
iface enp2s0 inet static
    address 192.168.0.88/24
    gateway 192.168.0.1
    pre-up /etc/network/neighbors kernel
    post-up /etc/network/routes up
    post-down /etc/network/routes down

# Main internal interface
auto enp3s0
iface enp3s0 inet static
    address 10.9.1.129/26
    pre-up /etc/network/neighbors account

# Reserve external interface
allow-hotplug ethusb
iface ethusb inet dhcp
    pre-up ip link set enp2s0 down
    pre-up /etc/network/dns usb-pre-up
    post-up /etc/network/dns usb-post-up
    pre-down /etc/network/dns usb-pre-down
```

```
post-down ip link set enp2s0 up
post-down ip route add default via 192.168.0.1
post-down /etc/network/dns usb-post-down
```

Тем самым сетевой интерфейс, выступающий в роли внешнего (относительно клиентов) получает адрес в ядре сети, а внутренний – адрес в подсети бухгалтерии.

Для применения нового конфигурационного файла следует перезапустить сетевую подсистему командой:

```
root@server:/# systemctl restart networking
```

Затем следует убедиться в успешности перезапуска по выводу команды:

```
root@server:/# systemctl status networking
• networking.service - Raise network interfaces
  Loaded: loaded (/lib/systemd/system/networking.service; enabled; vendor preset:
  enabled)
  Active: active (exited) since Tue 2021-05-11 11:43:22 MSK; 2h 3min ago
  Docs: man:interfaces(5)
  Main PID: 704 (code=exited, status=0/SUCCESS)
  Tasks: 0 (limit: 2315)
  Memory: 0B
  CGroup: /system.slice/networking.service
```

```
мая 11 11:43:21 server systemd[1]: Starting Raise network interfaces...
мая 11 11:43:22 server systemd[1]: Started Raise network interfaces.
```

Наконец, увидеть новые сетевые настройки системы можно в выводе команды:

```
root@server:/# ip address show
1: lo: <LOOPBACK,UP,LOWER_UP> mtu 65536 qdisc noqueue state UNKNOWN group default
qlen 1000
  link/loopback 00:00:00:00:00:00 brd 00:00:00:00:00:00
  inet 127.0.0.1/8 scope host lo
    valid_lft forever preferred_lft forever
2: enp3s0: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc pfifo_fast state UP
group default qlen 1000
  link/ether 1c:6f:65:c2:c5:6f brd ff:ff:ff:ff:ff:ff
  inet 10.9.1.129/26 brd 10.9.1.191 scope global enp3s0
    valid_lft forever preferred_lft forever
3: enp2s0: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc htb state UP group
default qlen 1000
  link/ether 18:d6:c7:00:ea:6c brd ff:ff:ff:ff:ff:ff
  inet 192.168.0.88/24 brd 192.168.0.255 scope global enp2s0
    valid_lft forever preferred_lft forever
4: gre0@NONE: <NOARP> mtu 1476 qdisc noop state DOWN group default qlen 1000
  link/gre 0.0.0.0 brd 0.0.0.0
5: gretap0@NONE: <BROADCAST,MULTICAST> mtu 1476 qdisc noop state DOWN group default
qlen 1000
  link/ether 00:00:00:00:00:00 brd ff:ff:ff:ff:ff:ff
6: erspan0@NONE: <BROADCAST,MULTICAST> mtu 1464 qdisc noop state DOWN group default
qlen 1000
  link/ether 00:00:00:00:00:00 brd ff:ff:ff:ff:ff:ff
```

Как видно из этого вывода, в системе присутствуют три сетевых интерфейса, считая петлевой, они используют именно те адреса, которые и были им заданы, при этом адреса IPv6 не назначены. Таким образом настройки сетевых интерфейсов полностью соответствуют заданным выше требованиям. Также в выводе обозначен неактивный в данный момент GRE-туннель, о котором было сказано ранее.

Кроме того, перед и в процессе запуска сетевых интерфейсов выполняются четыре сценария: внесение статических записей в низкоуровневые адресные таблицы, настройка маршрутизации, настройка полосы пропускания для клиентских соединений и настройка службы доменных имён (только при запуске или остановке резервного Интернет-подключения).

Все сценарии расположены в том же каталоге **/etc/network**, что и основной конфигурационный файл сетевых интерфейсов, и доступны для чтения, записи и исполнения только суперпользователю. Создание этих файлов и установка их атрибутов может быть выполнена командами (в выводе последней из них отображены лишь описываемые файлы):

```
root@server:/# cd /etc/network
root@server:/etc/network# touch shaping routes neighbours dns
root@server:/etc/network# chmod 0700 shaping routes neighbours dns
root@server:/etc/network# ls -l
итого 28
-rwx----- 1 root root    0 мая 11 14:25 dns
...
-rw-r--r-- 1 root root 1167 мая 11 14:03 interfaces
...
-rwx----- 1 root root    0 мая 11 14:25 neighbours
-rwx----- 1 root root    0 мая 11 14:25 routes
-rwx----- 1 root root    0 мая 11 14:25 shaping
```

На данный момент сценарий внесения статических записей `arp` имеет вид:

```
#!/bin/sh

if [ $1 = "kernel" ]
then
    ip neighbour add 192.168.0.1 dev enp2s0 lladdr 00:0c:26:a7:b9:39 nud permanent
    ip neighbour add 192.168.0.2 dev enp2s0 lladdr 90:2b:34:48:08:b5 nud permanent
    ip neighbour add 192.168.0.77 dev enp2s0 lladdr 54:be:f7:28:4b:40 nud permanent
elif [ $1 = "account" ]
then
    ip neighbour add 10.9.1.131 dev enp3s0 lladdr 90:2b:34:a3:83:44 nud permanent
    ip neighbour add 10.9.1.132 dev enp3s0 lladdr 50:e5:49:33:de:d9 nud permanent
    ip neighbour add 10.9.1.133 dev enp3s0 lladdr 90:2b:34:a0:24:7b nud permanent
    ip neighbour add 10.9.1.135 dev enp3s0 lladdr 90:2b:34:96:16:53 nud permanent
    ip neighbour add 10.9.1.134 dev enp3s0 lladdr f8:0d:ac:78:8a:f1 nud permanent
fi
exit 0
```

По мере замены оборудования на сервере виртуализации, основном шлюзе или любом другом указанном в этом сценарии узле их аппаратные адреса должны быть заменены.

Вывод следующей команды, выполненной после перезапуска сетевой подсистемы, демонстрирует факт успешного внесения статических записей об аппаратных адресах:

```
root@server:/# ip neighbour show
192.168.0.1 dev enp2s0 lladdr 00:0c:26:a7:b9:39 PERMANENT
192.168.0.2 dev enp2s0 lladdr 90:2b:34:48:08:b5 PERMANENT
192.168.0.77 dev enp2s0 lladdr 54:be:f7:28:4b:40 PERMANENT
10.9.1.132 dev enp3s0 lladdr 50:e5:49:33:de:d9 PERMANENT
10.9.1.134 dev enp3s0 lladdr f8:0d:ac:78:8a:f1 PERMANENT
10.9.1.131 dev enp3s0 lladdr 90:2b:34:a3:83:44 PERMANENT
10.9.1.133 dev enp3s0 lladdr 90:2b:34:a0:24:7b PERMANENT
10.9.1.135 dev enp3s0 lladdr 90:2b:34:96:16:53 PERMANENT
```

Сценарий настройки службы доменных имён описан ниже в разделе «Служба DNS/DHCP».

Маршрутизация

Следует отметить, что связь устройств в локальной сети организации реализована при помощи разбиения локальной сети на изолированные подсети, защищённые своими межсетевыми экранами, одновременно выполняющими роль серверов таких подсетей. Одним из таких серверов (для подсети бухгалтерии) и является рассматриваемая система. При этом эти серверы отвечают за маршрутизацию пакетов и выполнение преобразования адресов (NAT). С точки зрения маршрутизации каждый такой сервер должен, с одной стороны, предоставлять клиентам кратчайший маршрут до целевого узла сети, а с другой – не предоставлять маршрута к тем подсетям, взаимодействие с которыми не предусмотрено логикой работы сети и организации. В данном случае, сервер бухгалтерии должен предоставлять клиентам только маршруты до ядра сети (в том числе главного шлюза и сервера АИС «Аверс») и до сети виртуальных серверов. Также должен быть предоставлен маршрут по умолчанию, направленный через главный шлюз организации. Здесь не рассматриваются вопросы защиты от попыток попасть в другие подсети, используя, таблицы маршрутизации других серверов, например, главного шлюза, т. к. защита от таких атак – удел межсетевого экрана самого этого шлюза. Далее в разделе «Установка и настройка сетевого фильтра (nftables)» описаны меры, противодействующие использованию этого сервера для той же цели.

Касаясь вопроса преобразования адресов, следует отметить, что никакого влияния на маршрутизацию внутри локальной сети NAT не имеет. При обращении ко внешним ресурсам симметричный NAT выполняется на главном шлюзе организации. Внутри локальной сети такое преобразование может иметь в своём применении как положительные, так и отрицательные черты. К отрицательным аспектам следует отнести повышенную нагрузку на машину, выполняющую это преобразование, и невозможность в журналах внешних (относительно подсети) серверов выяснить исходный IP-адрес клиента. Эту же невозможность в некоторых случаях следует считать достоинством. Например, применяя NAT на сервере подсети бухгалтерии, можно добиться достаточно надёжного сокрытия информации о количестве хостов в этой подсети, их внутренних адресах, используемых каждым из них портов и т. д. Как следствие, станет невозможно определить, с какого из компьютеров бухгалтерии выполнялось подключение к локальному информационному серверу. При применении того же механизма к беспроводной подсети результатом будет сокрытие следов клиента-злоумышленника в журналах того же инфосервера. Именно поэтому в беспроводной сети NAT не используется, а в подсети бухгалтерии используется.

К маршрутам, создаваемым по умолчанию на основе настроек сетевых интерфейсов следует добавить дополнительные сетевые маршруты с тем, чтобы слать пакеты по кратчайшему пути. Для этого в упомянутый ранее сценарий настройки маршрутизации достаточно привести к виду:

```
root@server:/# cat /etc/network/routes
#!/bin/sh

if [ $1 = "up" ]
then
    ip route add 192.168.7.0/24 via 192.168.0.77 dev enp2s0
elif [ $1 = "down" ]
then
```

```
ip route del 192.168.7.0/24 via 192.168.0.77 dev enp2s0
fi
exit 0
```

После перезапуска сетевой подсистемы при помощи следующих команд можно убедиться как в наличии, так и в работоспособности добавленного маршрута:

```
root@server:/# ip route show
default via 192.168.0.1 dev enp2s0 onlink
10.9.1.128/26 dev enp3s0 proto kernel scope link src 10.9.1.129
192.168.0.0/24 dev enp2s0 proto kernel scope link src 192.168.0.88
192.168.7.0/24 via 192.168.0.77 dev enp2s0
root@server:/# traceroute 192.168.7.1
traceroute to 192.168.7.1 (192.168.7.1), 30 hops max, 60 byte packets
1  server.service.school34 (192.168.7.1)  0.625 ms  0.581 ms  0.559 ms
root@server:/# ping -c 3 debian.service.school34
PING debian.service.school34 (192.168.7.5) 56(84) bytes of data.
64 bytes from debian.service.school34 (192.168.7.5): icmp_seq=1 ttl=63 time=4.51 ms
64 bytes from debian.service.school34 (192.168.7.5): icmp_seq=2 ttl=63 time=1.56 ms
64 bytes from debian.service.school34 (192.168.7.5): icmp_seq=3 ttl=63 time=1.12 ms

--- debian.service.school34 ping statistics ---
3 packets transmitted, 3 received, 0% packet loss, time 5ms
rtt min/avg/max/mdev = 1.116/2.394/4.513/1.509 ms
```

В представленной первой командой таблице маршрутизации присутствуют также маршруты, созданные на основе настроек сетевых интерфейсов. В совокупности они составляют полный набор маршрутов, необходимых серверу для работы.

Управление пропускной способностью сети

Настройка полосы пропускания для клиентских соединений выполнена средствами механизма traffic control и утилиты tc исходя из следующих соображений:

- трафик делится на три класса: соединения с сервером системы «Аверс», соединения с внутренним сервером виртуализации и соединения с ресурсами Интернет
- для трафика к серверу системы «Аверс» гарантируется полоса пропускания 20 Мбит/с с возможностью её расширения до 50 Мбит/с
- для трафика к внешним Интернет-ресурсам гарантируется полоса пропускания 2 Мбит/с с возможностью её расширения до 10 Мбит/с
- для трафика к внутренним серверам гарантируется полоса пропускания 100 Мбит/с с возможностью её расширения до 1000 Мбит/с
- полоса пропускания для трафика к серверу системы «Аверс» ограничена сверху 50 Мбит/с с целью ограничения максимальной нагрузки на этот сервер со стороны рабочих станций бухгалтерии, т.к. предоставляемые этим сервером данные востребованы бухгалтерией изредка, а другие подсети создают существенную нагрузку для этого сервера
- полоса пропускания к внешним Интернет-ресурсам ограничена сверху шириной внешнего канала школы
- полоса пропускания для трафика к внутренним серверам школы ограничена сверху пропускной способностью проводного сетевого интерфейса описываемой системы
- оверкоммитинг, т.е. установка суммарной гарантированной пропускной способности, превышающей аппаратные возможности интерфейса, не допускается

Исходя из приведённых соображений сценарий настройки полосы пропускания для транзитных сетевых соединений принимает вид:

```
#!/bin/sh
```

```
#Сбрасываем текущее состояние
tc qdisc delete dev enp2s0 root
#Ставим корневую дисциплину
tc qdisc add dev enp2s0 root handle 1: htb default 13
#Создаём корневой класс
tc class add dev enp2s0 parent 1: classid 1:1 htb rate 1000mbit ceil 1000mbit
#Создаём подклассы: avers, service, inet
tc class add dev enp2s0 parent 1:1 classid 1:11 htb rate 20mbit      ceil 50mbit
tc class add dev enp2s0 parent 1:1 classid 1:12 htb rate 100mbit     ceil 1000mbit
tc class add dev enp2s0 parent 1:1 classid 1:13 htb rate 2mbit       ceil 10mbit
#Настраиваем дисциплины для подклассов
tc qdisc add dev enp2s0 parent 1:11 handle 10:0 sfq perturb 10
tc qdisc add dev enp2s0 parent 1:12 handle 20:0 sfq perturb 10
tc qdisc add dev enp2s0 parent 1:13 handle 30:0 sfq perturb 10
#Настраиваем фильтры для классификации трафика (по умолчанию - inet)
tc filter add dev enp2s0 protocol ip parent 1:0 prio 1 u32 match ip dst 192.168.0.4
flowid 1:11
tc filter add dev enp2s0 protocol ip parent 1:0 prio 1 u32 match ip dst
192.168.7.0/24 flowid 1:12
```

В первую очередь сценарий удаляет текущую корневую дисциплину вместе со всеми её компонентами и подключает в качестве корневой дисциплину htb (Hierarchical Token Bucket), при этом указывается, что весь не классифицированный (default) трафик должен быть обработан с помощью дисциплин класса 1:13 (Интернет-трафик). Затем создаётся корневой класс, куда будет попадать весь трафик (это нужно для реализации заимствования). В этом классе пропускная способность ограничивается согласно аппаратным возможностям. Далее создаются три подкласса, обеспечивающие заданное разделение ширины канала. После этого создаются три дисциплины sfq (Stochastic Fairness Queueing), по одной для каждого класса. В завершение создаются два фильтра: первый классифицирует трафик к серверу АИС «Аверс», второй – к серверу виртуализации. Неклассифицированный трафик считается трафиком, направленным в Интернет.

После запуска сценария проверить состояние подсистемы контроля трафика можно следующими командами:

```
root@server:/# /sbin/tc qdisc show
qdisc noqueue 0: dev lo root refcnt 2
qdisc pfifo_fast 0: dev enp3s0 root refcnt 2 bands 3 priomap 1 2 2 2 1 2 0 0 1 1 1 1
1 1 1 1
qdisc htb 1: dev enp2s0 root refcnt 2 r2q 10 default 0x13 direct_packets_stat 0
direct_qlen 1000
qdisc sfq 30: dev enp2s0 parent 1:13 limit 127p quantum 1514b depth 127 divisor 1024
perturb 10sec
qdisc sfq 10: dev enp2s0 parent 1:11 limit 127p quantum 1514b depth 127 divisor 1024
perturb 10sec
qdisc sfq 20: dev enp2s0 parent 1:12 limit 127p quantum 1514b depth 127 divisor 1024
perturb 10sec
root@server:/# /sbin/tc class show dev enp2s0
class htb 1:11 parent 1:1 leaf 10: prio 0 rate 20Mbit ceil 50Mbit burst 1600b cburst
1600b
class htb 1:1 root rate 1Gbit ceil 1Gbit burst 1375b cburst 1375b
class htb 1:13 parent 1:1 leaf 30: prio 0 rate 2Mbit ceil 10Mbit burst 1600b cburst
```

```

1600b
class htb 1:12 parent 1:1 leaf 20: prio 0 rate 100Mbit ceil 1Gbit burst 1600b cburst
1375b
root@server:/# /sbin/tc filter show dev enp2s0
filter parent 1: protocol ip pref 1 u32 chain 0
filter parent 1: protocol ip pref 1 u32 chain 0 fh 800: ht divisor 1
filter parent 1: protocol ip pref 1 u32 chain 0 fh 800::800 order 2048 key ht 800
bkt 0 flowid 1:11 not_in_hw
  match c0a80004/ffffffff at 16
filter parent 1: protocol ip pref 1 u32 chain 0 fh 800::801 order 2049 key ht 800
bkt 0 flowid 1:12 not_in_hw
  match c0a80700/ffffff00 at 16

```

Проверка работоспособности резервных каналов

Разумеется, эту проверку следует выполнять после настройки межсетевого экрана, т. к. его правила могут при неполной настройке препятствовать использованию резервного канала. Сама проверка сводится к возможности выхода в Интернет с клиентского компьютера как при подключённом к серверу смартфоне в режиме usb-модема, так и при работе на основном Интернет-канале.

Служба DNS/DHCP

Настройка

Для реализации этой службы следует установить пакет `dnsmasq` со всеми зависимостями:

```

root@server:/# apt-get update
root@server:/# apt-get install dnsmasq

```

Затем следует привести конфигурационный файл `/etc/dnsmasq.conf` к виду:

```

### General configuration ###

listen-address=127.0.0.1
listen-address=10.9.1.129
cache-size=500
domain-needed
bind-dynamic
filterwin2k
no-resolv
no-poll
no-hosts
bogus-priv

server=77.88.8.7@enp2s0
server=/school34/192.168.0.1@enp2s0
server=/0.168.192.in-addr.arpa/192.168.0.1@enp2s0
server=/service.school34/192.168.0.77@enp2s0
server=/7.168.192.in-addr.arpa/192.168.0.77@enp2s0

local=/account.school34/
domain=school34,192.168.0.0/24
domain=service.school34,192.168.7.0/24
domain=account.school34,10.9.1.128/26

mx-target=mail.service.school34
localmx

```

DHCP configuration

```
no-dhcp-interface=lo
no-dhcp-interface=enp2s0
dhcp-range=10.9.1.128,static,infinite
dhcp-option=6,10.9.1.129
dhcp-option=42,10.9.1.129
```

```
#log-queries
#log-facility=/var/log/dnsmasq.log
```

```
dhcp-host=90:2b:34:a3:83:44,main,10.9.1.131
dhcp-host=50:e5:49:33:de:d9,accd1,10.9.1.132
dhcp-host=90:2b:34:a0:24:7b,accd2,10.9.1.133
dhcp-host=90:2b:34:96:16:53,accd3,10.9.1.135
dhcp-host=f8:0d:ac:78:8a:f1,mfp,10.9.1.134
```

DNS configuration

```
address=/server.account.school34/10.9.1.129
ptr-record=129.1.9.10.in-addr.arpa,server.account.school34
mx-host=server.account.school34,mail.service.school34,50
txt-record=server.account.school34,"account department subnet server"
```

```
address=/samba.account.school34/10.9.1.129
ptr-record=129.1.9.10.in-addr.arpa,samba.account.school34
mx-host=samba.account.school34,mail.service.school34,50
txt-record=samba.account.school34,"account department file server"
```

```
address=/ntp.account.school34/10.9.1.129
ptr-record=129.1.9.10.in-addr.arpa,ntp.account.school34
mx-host=ntp.account.school34,mail.service.school34,50
txt-record=ntp.account.school34,"account department time server"
```

```
address=/main.account.school34/10.9.1.131
ptr-record=131.1.9.10.in-addr.arpa,main.account.school34
mx-host=main.account.school34,mail.service.school34,50
txt-record=main.account.school34,"senior accountant workstation"
```

```
address=/accd1.account.school34/10.9.1.132
ptr-record=132.1.9.10.in-addr.arpa,accd1.account.school34
mx-host=accd1.account.school34,mail.service.school34,50
txt-record=accd1.account.school34,"accountant workstation"
```

```
address=/accd2.account.school34/10.9.1.133
ptr-record=133.1.9.10.in-addr.arpa,accd2.account.school34
mx-host=accd2.account.school34,mail.service.school34,50
txt-record=accd2.account.school34,"accountant workstation"
```

```
address=/accd3.account.school34/10.9.1.135
ptr-record=135.1.9.10.in-addr.arpa,main.account.school34
mx-host=accd3.account.school34,mail.service.school34,50
txt-record=accd3.account.school34,"accountant workstation"
```

```
address=/mfp.account.school34/10.9.1.134
ptr-record=134.1.9.10.in-addr.arpa,mfp.account.school34
mx-host=mfp.account.school34,mail.service.school34,50
txt-record=mfp.account.school34,"account department network MFP"
```

Тем самым, службе dnsmasq предписывается следующий порядок работы:

Служба принимает запросы только на локальном петлевом и внутреннем клиентском интерфейсах. Кэш DNS-записей увеличен со 150 (по умолчанию) до 500 записей, т. к. объем оперативной памяти в системе этому не препятствует, а на быстродействии службы такое увеличение сказывается положительно. Объем в 500 записей выбран как наиболее оптимальный по опыту эксплуатации сети, частью которой является сеть, обслуживаемая описываемой системой.

Параметр `bind-dynamic` предписывает службе реагировать на изменение состояния сетевых интерфейсов. Таким образом, возможен запуск службы при отключённом сетевом интерфейсе устройства с автоматическим «подхватом» этого интерфейса при его запуске.

В своей работе служба не использует системные конфигурационные файлы встроенного DNS-клиента. Все необходимые для ее работы параметры указаны в основном конфигурационном файле службы, приведенном выше.

Службе известны три локальных домена (`school34`, `service.school34` и `account.school34`). Каждому из них (кроме обслуживаемого самой системой домена `account.school34`) сопоставлен свой DNS-сервер, обращение к которому выполняется через внешний интерфейс системы. Разрешение DNS-запросов о прочих немаршрутизируемых в Интернете адресах блокируется параметром `bogus-priv`. DNS-запросы о внешних хостах разрешаются при помощи YandexDNS.

Параметры обслуживания электронной почты заданы таким образом, чтобы почтовым адресом по умолчанию для всех клиентов подсети являлся адрес локального почтового сервера организации (`mail.service.school34`).

Протокол DHCP в беспроводной подсети раздает клиентам адреса в диапазоне 10.9.1.130 – 10.9.1.190 (62 адреса, соответственно не более 62 клиентов одновременно). Запросы на получение адреса, очевидно, принимаются только с внутреннего интерфейса системы. Кроме предоставления адреса сервер предоставляет клиенту услуги DNS-службы (`dhcp-option 6`) и службы точного времени (`dhcp-option 42`). Следует отметить, что адрес выдается только зарегистрированным в конфигурационном файле клиентам, причём адрес клиенту выдается всегда одинаковый, т. е. настройка адреса динамическая, а сам адрес статический.

Служба ведет журнал в файле `/var/log/daemon.log`, где кратко отражаются сведения о запуске службы, ее настройках и т. д. Ведение подробного журнала службы в данной конфигурации отключено, однако раскомментировав соответствующие две строки в приведенном файле, можно включить ведение журнала. При этом следует помнить, что ведение журнала, даже с учетом ротации журнальных файлов (см. «О дополнительных службах»), может служить направлением для атаки на сервер: отправка огромного количества бессмысленных запросов способно переполнить дисковое пространство сервера. В рассматриваемой системе нет необходимости в подробном журналировании происходящего, поэтому оно и отключено.

По завершении редактирования конфигурационного файла для начала работы службы остается её лишь перезапустить командой

```
root@server:/# systemctl restart dnsmasq
```

Убедиться в успешности перезапуска можно по выводу команды

```
root@server:/# systemctl status dnsmasq
```

```
• dnsmasq.service - dnsmasq - A lightweight DHCP and caching DNS server
  Loaded: loaded (/lib/systemd/system/dnsmasq.service; enabled; vendor preset:
  enabled)
  Active: active (running) since Tue 2021-05-11 08:00:27 MSK; 1h 35min ago
  Process: 370 ExecStartPre=/usr/sbin/dnsmasq --test (code=exited, status=0/SUCCESS)
  Process: 381 ExecStart=/etc/init.d/dnsmasq systemd-exec (code=exited,
  status=0/SUCCESS)
  Process: 389 ExecStartPost=/etc/init.d/dnsmasq systemd-start-resolvconf
  (code=exited, status=0/SUCCESS)
  Main PID: 388 (dnsmasq)
    Tasks: 1 (limit: 2315)
    Memory: 3.0M
    CGroup: /system.slice/dnsmasq.service
            └─388 /usr/sbin/dnsmasq -x /run/dnsmasq/dnsmasq.pid -u dnsmasq -7
  /etc/dnsmasq.d,.dpkg-dist,.dpkg-old,.dpkg-new --local-service --trust-
  anchor=.,20326,8,2,e06d44b80b8f1d39a95c0b0d7c65d08458e8
```

```
мая 11 08:00:26 server systemd[1]: Starting dnsmasq - A lightweight DHCP and caching
DNS server...
мая 11 08:00:27 server dnsmasq[370]: dnsmasq: syntax check OK.
мая 11 08:00:27 server systemd[1]: Started dnsmasq - A lightweight DHCP and caching
DNS server.
```

Служба работает от имени системного пользователя dnsmasq. Идентификатор процесса (PID) по умолчанию вносится в файл **/run/dnsmasq/dnsmasq.pid**. В ходе работы процесс dnsmasq реагирует на сигнал SIGUSR1 отправкой в системный журнал сообщений о текущих клиентах, статистике их запросов и т.д. Следующая команда позволяет отправить этот сигнал нужному процессу и открыть выборку из системного журнала, содержащую все упоминания о dnsmasq за последний час:

```
root@server:/# kill -s SIGUSR1 $(cat /run/dnsmasq/dnsmasq.pid) && journalctl -u
dnsmasq --since -1h
```

На этом настройка основного режима работы служб DNS/DHCP завершена, однако следует предусмотреть изменение конфигурации службы при переключении на резервное Интернет-соединение через usb-модем (смартфон). В таком режиме происходит отключение подсети бухгалтерии и её сервера от локальной сети организации, а значит ресурсы этой локальной сети перестают быть доступными. Следовательно, dnsmasq в таком режиме работы не должен ничего знать о локальной сети. Наиболее простым способом реализации такого поведения службы является её перезапуск при смене внешнего интерфейса с использованием альтернативного конфигурационного файла **/etc/dnsmasq.usb.conf**:

```
### General configuration ###
```

```
listen-address=127.0.0.1
listen-address=10.9.1.129
cache-size=500
domain-needed
bind-dynamic
filterwin2k
no-resolv
no-poll
no-hosts
bogus-priv
```

```
server=77.88.8.7@ethusb
```

```

local=/school34/
local=/account.school34/
domain=account.school34,10.9.1.128/26
selfmx

### DHCP configuration ###

no-dhcp-interface=lo
no-dhcp-interface=ethusb
dhcp-range=10.9.1.128,static,infinite
dhcp-option=6,10.9.1.129
dhcp-option=42,10.9.1.129

#log-queries
#log-facility=/var/log/dnsmasq.log

dhcp-host=90:2b:34:a3:83:44,main,10.9.1.131
dhcp-host=50:e5:49:33:de:d9,accd1,10.9.1.132
dhcp-host=90:2b:34:a0:24:7b,accd2,10.9.1.133
dhcp-host=90:2b:34:96:16:53,accd3,10.9.1.135
dhcp-host=f8:0d:ac:78:8a:f1,mfp,10.9.1.134

### DNS configuration ###

address=/server.account.school34/10.9.1.129
ptr-record=129.1.9.10.in-addr.arpa,server.account.school34
txt-record=server.account.school34,"account department subnet server"

address=/samba.account.school34/10.9.1.129
ptr-record=129.1.9.10.in-addr.arpa,samba.account.school34
mx-host=samba.account.school34,mail.service.school34,50
txt-record=samba.account.school34,"account department file server"

address=/ntp.account.school34/10.9.1.129
ptr-record=129.1.9.10.in-addr.arpa,ntp.account.school34
mx-host=ntp.account.school34,mail.service.school34,50
txt-record=ntp.account.school34,"account department time server"

address=/main.account.school34/10.9.1.131
ptr-record=131.1.9.10.in-addr.arpa,main.account.school34
txt-record=main.account.school34,"senior accountant workstation"

address=/accd1.account.school34/10.9.1.132
ptr-record=132.1.9.10.in-addr.arpa,accd1.account.school34
txt-record=accd1.account.school34,"accountant workstation"

address=/accd2.account.school34/10.9.1.133
ptr-record=133.1.9.10.in-addr.arpa,accd2.account.school34
txt-record=accd2.account.school34,"accountant workstation"

address=/accd3.account.school34/10.9.1.135
ptr-record=135.1.9.10.in-addr.arpa,main.account.school34
txt-record=accd3.account.school34,"accountant workstation"

address=/mfp.account.school34/10.9.1.134
ptr-record=134.1.9.10.in-addr.arpa,mfp.account.school34
txt-record=mfp.account.school34,"account department network MFP"

```

Следует отметить отличия альтернативной конфигурации от основной:

Во-первых, изменено имя внешнего интерфейса, и на нём также отключен протокол DHCP. Во-вторых, доступным остался лишь один внешний сервер имён (YandexDNS). Наконец, изменена логика работы с почтовыми (MX) записями: при отсутствии единого почтового сервера конечным адресатом для каждого клиента сети является он сам.

Обеспечение перезапуска службы ложится на файлы **/etc/network/interfaces** и **/etc/default/dnsmasq**. Последний файл содержит настройки dnsmasq по умолчанию и позволяет легко изменить путь к основному конфигурационному файлу. В файле настройки сетевых интерфейсов следует при запуске резервного соединения выполнить такую подмену, а при его остановке вернуть исходное состояние. Такая подмена выполняется при помощи сценария **/etc/network/dns** :

```
#!/bin/sh

if [ $1 = "usb-pre-up" ]
then
    systemctl stop dnsmasq
elif [ $1 = "usb-post-up" ]
then
    sed s/"#DNSMASQ_OPTS="/"/DNSMASQ_OPTS="/ -i /etc/default/dnsmasq
    echo "nameserver 127.0.0.1" > /etc/resolv.conf
    systemctl start dnsmasq
elif [ $1 = "usb-pre-down" ]
then
    systemctl stop dnsmasq
    sed s/"DNSMASQ_OPTS="/"/#DNSMASQ_OPTS="/ -i /etc/default/dnsmasq
elif [ $1 = "usb-post-down" ]
then
    echo "nameserver 127.0.0.1" > /etc/resolv.conf
    systemctl start dnsmasq
fi
exit 0
```

Так перед запуском резервного интерфейса следует остановить службу dnsmasq, а сразу после необходимо выполнить замену конфигурационного файла этой службы, указать в файле **/etc/resolv.conf**, что сам сервер должен по-прежнему пользоваться услугами собственной службы, а не обращаться к dns-серверу предоставленному мобильным провайдером, и вновь запустить службу dnsmasq. Перед отключением интерфейса следует остановить службу доменных имён и вернуться к основному конфигурационному файлу. Наконец, после отключения резервного соединения остаётся вновь запустить службу.

Замечание 1: В файле **/etc/network/interfaces** порядок команд, вызываемых при запуске или остановке резервного интерфейса значим. И в основной, и в резервной конфигурациях dnsmasq используется параметр **bind-dynamic**, позволяющий службе начать работу даже если необходимые ей сетевые интерфейсы ещё не запущены. В этой ситуации dnsmasq автоматически «подхватывает» вновь запущенные сетевые интерфейсы по мере их готовности. Однако при таком поведении никак не предотвращается возможность захвата резервного соединения из основной конфигурации и наоборот. Соблюдение же порядка команд в файле **/etc/network/interfaces** позволяет этого избежать.

Замечание 2: В Debian предусмотрены подкаталоги **/etc/network/*.d** для размещения подобных сценариев. Однако исключительно из стилистических соображений на рассматриваемой системе применён не предлагаемый Debian подход с группировкой

действия по вызывающим их событиям, а собственный с группировкой действий по сетевым интерфейсам.

При этом в файл **/etc/default/dnsmasq** следует внести небольшую поправку: т.к. альтернативная конфигурация dnsmasq описана в файле, **/etc/dnsmasq.usb.conf**, то строку

```
#DNMASQ_OPTS="--conf-file=/etc/dnsmasq.alt"
```

в файле **/etc/default/dnsmasq** следует привести к виду:

```
#DNMASQ_OPTS="--conf-file=/etc/dnsmasq.usb.conf"
```

По завершении настройки службы следует указать системе использовать собственную службу для собственных же нужд. Для этого следует привести конфигурационный файл **/etc/resolv.conf** к следующему виду:

```
nameserver 127.0.0.1
```

Затем следует перезапустить сетевую подсистему командой:

```
root@server:/# systemctl restart networking
```

Это требуется для того, чтобы порождённые самой системой dns-запросы о локальных серверах отправлялись сразу к серверу виртуализации, а не перенаправлялись туда главным плюсом.

Защита от падений (автоматическое возобновление)

Для организации автоматического восстановления работоспособности службы после сбоя можно воспользоваться средствами системы инициализации systemd.

В первую очередь следует проверить текущие параметры службы командой:

```
root@server:/# systemctl show dnsmasq
Type=forking
Restart=no
PIDFile=/run/dnsmasq/dnsmasq.pid
NotifyAccess=none
RestartUSec=100ms
TimeoutStartUSec=1min 30s
TimeoutStopUSec=1min 30s
```

Как видно из отклика команды (он приведён здесь в сильно сокращённом виде, содержащем важные в текущем контексте строки), перезапуск службы отключён.

Затем следует выполнить редактирование службы при помощи команды:

```
root@server:/# systemctl edit dnsmasq
```

При этом будет открыт текстовый редактор, в котором необходимо внести определённые правки. В случае подтверждения сохранения изменений при выходе из текстового редактора в системе создаётся файл **/etc/systemd/system/dnsmasq.service.d/override.conf**, в котором эти поправки и хранятся. При очередном запуске службы они переопределяют исходные значения параметров. Для решения задачи перезапуска службы при сбоях достаточно следующего содержимого этого файла:

```
[Service]
Restart=on-failure
```

Стоит отметить, что такие изменения в службе сохраняются и при установке обновлений системы.

После внесения изменений следует перезапустить службу и убедиться, что изменения параметров учтены:

```
root@server:/# systemctl restart dnsmasq
root@server:/# systemctl show dnsmasq
Type=forking
Restart=on-failure
PIDFile=/run/dnsmasq/dnsmasq.pid
NotifyAccess=none
RestartUSec=100ms
TimeoutStartUSec=1min 30s
TimeoutStopUSec=1min 30s
```

Проверка работоспособности и устойчивости

Проверка работоспособности службы тривиальна: выполняется подключение клиента (разумеется, внесённого временно в конфигурационный файл) по протоколу DHCP и на нём выполняется ряд DNS-запросов. В приведённом далее фрагменте терминального сеанса, выполненного на клиентском устройстве под управлением live-диска Xubuntu 20.04, демонстрируется факт подключения к сети, получения IP-адреса, сетевой маски, сетевых маршрутов и адресов шлюза. Тестирование выполняется при помощи небольшого вспомогательного сценария, расположенного в каталоге /tmp:

```
xubuntu@xubuntu:/# cat /tmp/dnstest.sh
#!/bin/bash -x
ip address show
ip route show
host samba.account.school34
host avers.school34
host mail.service.school34
host yandex.ru
host mcst.ru
```

Параметр -x интерпретатора позволяет в ходе работы сценария отображать выполняемые команды, в приводимом далее фрагменте терминального сеанса эти команды выделены жирным шрифтом. Разумеется, сценарий сделан исполняемым.

```
xubuntu@xubuntu:/tmp$ ./dnstest.sh
+ ip address show
...
2: enp2s0: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc mq state UP group
default qlen 1000
    link/ether 60:eb:69:b0:45:e2 brd ff:ff:ff:ff:ff:ff
    inet 10.9.1.136/26 brd 10.9.1.191 scope global noprefixroute enp2s0
        valid_lft forever preferred_lft forever
    inet6 fe80::ce14:bdee:a6db:a8bc/64 scope link noprefixroute
        valid_lft forever preferred_lft forever
3: wlp3s0: <NO-CARRIER,BROADCAST,MULTICAST,UP> mtu 1500 qdisc noqueue state DOWN
group default qlen 1000
...
+ ip route show
default via 10.9.1.129 dev enp2s0 proto dhcp metric 100
```

```

10.9.1.128/26 dev enp2s0 proto kernel scope link src 10.9.1.136 metric 100
169.254.0.0/16 dev enp2s0 scope link metric 1000
+ host samba.account.school34
samba.account.school34 has address 10.9.1.129
samba.account.school34 mail is handled by 50 mail.service.school34.
+ host avers.school34
avers.school34 has address 192.168.5.51
+ host mail.service.school34
mail.service.school34 has address 192.168.7.3
mail.service.school34 mail is handled by 50 mail.service.school34.
+ host yandex.ru
yandex.ru has address 213.180.193.56
yandex.ru has IPv6 address 2a02:6b8:a::a
yandex.ru mail is handled by 10 mx.yandex.ru.
+ host mcst.ru
mcst.ru has address 84.201.189.147
mcst.ru mail is handled by 50 tretyak2.mcst.ru.
xubuntu@xubuntu:/tmp$ ./dnstest.sh
+ ip address show
...
2: enp2s0: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc mq state UP group
default qlen 1000
    link/ether 60:eb:69:b0:45:e2 brd ff:ff:ff:ff:ff:ff
    inet 10.9.1.136/26 brd 10.9.1.191 scope global noprefixroute enp2s0
        valid_lft forever preferred_lft forever
    inet6 fe80::ce14:bcee:a6db:a8bc/64 scope link noprefixroute
        valid_lft forever preferred_lft forever
3: wlp3s0: <NO-CARRIER,BROADCAST,MULTICAST,UP> mtu 1500 qdisc noqueue state DOWN
group default qlen 1000
...
+ ip route show
default via 10.9.1.129 dev enp2s0 proto dhcp metric 100
10.9.1.128/26 dev enp2s0 proto kernel scope link src 10.9.1.136 metric 100
169.254.0.0/16 dev enp2s0 scope link metric 1000
+ host samba.account.school34
samba.account.school34 has address 10.9.1.129
samba.account.school34 mail is handled by 50 mail.service.school34.
+ host avers.school34
Host avers.school34 not found: 3(NXDOMAIN)
+ host mail.service.school34
Host mail.service.school34 not found: 3(NXDOMAIN)
+ host yandex.ru
yandex.ru has address 213.180.193.56
yandex.ru has IPv6 address 2a02:6b8:a::a
yandex.ru mail is handled by 10 mx.yandex.ru.
+ host mcst.ru
mcst.ru has address 84.201.189.147
mcst.ru mail is handled by 50 tretyak2.mcst.ru.
xubuntu@xubuntu:/tmp$

```

В представленном фрагменте часть откликов команд сокращены в неинформативной части. В то же время со стороны сервера в журнальном файле **/var/log/dnsmasq.log** можно видеть следующие строки (также с сокращениями):

```

Jun 21 14:41:49 dnsmasq[1273]: query[A] samba.account.school34 from 10.9.1.136
Jun 21 14:41:49 dnsmasq[1273]: config samba.account.school34 is 10.9.1.129
Jun 21 14:41:49 dnsmasq[1273]: query[AAAA] samba.account.school34 from 10.9.1.136
Jun 21 14:41:49 dnsmasq[1273]: config samba.account.school34 is NODATA-IPv6
Jun 21 14:41:49 dnsmasq[1273]: query[MX] samba.account.school34 from 10.9.1.136
Jun 21 14:41:49 dnsmasq[1273]: config samba.account.school34 is <MX>
Jun 21 14:41:49 dnsmasq[1273]: query[A] avers.school34 from 10.9.1.136

```

```

Jun 21 14:41:49 dnsmasq[1273]: forwarded avers.school34 to 192.168.0.1
Jun 21 14:41:49 dnsmasq[1273]: reply avers.school34 is 192.168.5.51
Jun 21 14:41:49 dnsmasq[1273]: query[AAAA] avers.school34 from 10.9.1.136
Jun 21 14:41:49 dnsmasq[1273]: forwarded avers.school34 to 192.168.0.1
Jun 21 14:41:49 dnsmasq[1273]: query[MX] avers.school34 from 10.9.1.136
Jun 21 14:41:49 dnsmasq[1273]: forwarded avers.school34 to 192.168.0.1
Jun 21 14:41:49 dnsmasq[1273]: query[A] mail.service.school34 from 10.9.1.136
Jun 21 14:41:49 dnsmasq[1273]: forwarded mail.service.school34 to 192.168.0.77
Jun 21 14:41:49 dnsmasq[1273]: reply mail.service.school34 is 192.168.7.3
Jun 21 14:41:49 dnsmasq[1273]: query[AAAA] mail.service.school34 from 10.9.1.136
Jun 21 14:41:49 dnsmasq[1273]: forwarded mail.service.school34 to 192.168.0.77
Jun 21 14:41:49 dnsmasq[1273]: query[MX] mail.service.school34 from 10.9.1.136
Jun 21 14:41:49 dnsmasq[1273]: forwarded mail.service.school34 to 192.168.0.77
Jun 21 14:41:49 dnsmasq[1273]: query[A] yandex.ru from 10.9.1.136
Jun 21 14:41:49 dnsmasq[1273]: forwarded yandex.ru to 77.88.8.7
Jun 21 14:41:49 dnsmasq[1273]: reply yandex.ru is 213.180.193.56
Jun 21 14:41:49 dnsmasq[1273]: query[AAAA] yandex.ru from 10.9.1.136
Jun 21 14:41:49 dnsmasq[1273]: forwarded yandex.ru to 77.88.8.7
Jun 21 14:41:49 dnsmasq[1273]: reply yandex.ru is 2a02:6b8:a::a
Jun 21 14:41:49 dnsmasq[1273]: query[MX] yandex.ru from 10.9.1.136
Jun 21 14:41:49 dnsmasq[1273]: forwarded yandex.ru to 77.88.8.7
Jun 21 14:41:49 dnsmasq[1273]: query[A] mcst.ru from 10.9.1.136
Jun 21 14:41:49 dnsmasq[1273]: forwarded mcst.ru to 77.88.8.7
Jun 21 14:41:54 dnsmasq[1273]: reply mcst.ru is 84.201.189.147
Jun 21 14:41:54 dnsmasq[1273]: query[AAAA] mcst.ru from 10.9.1.136
Jun 21 14:41:54 dnsmasq[1273]: forwarded mcst.ru to 77.88.8.7
Jun 21 14:41:54 dnsmasq[1273]: reply mcst.ru is NODATA-IPv6
Jun 21 14:41:54 dnsmasq[1273]: query[MX] mcst.ru from 10.9.1.136
Jun 21 14:41:54 dnsmasq[1273]: forwarded mcst.ru to 77.88.8.7
...
Jun 21 14:46:30 dnsmasq[1598]: started, version 2.80 cachesize 500
Jun 21 14:46:30 dnsmasq[1598]: compile time options: IPv6 GNU-getopt DBus i18n IDN
DHCP DHCPv6 no-Lua TFTP conntrack ipset auth DNSSEC loop-detect inotify dumpfile
Jun 21 14:46:30 dnsmasq-dhcp[1598]: DHCP, static leases only on 10.9.1.128, lease
time infinite
Jun 21 14:46:30 dnsmasq[1598]: using local addresses only for domain
account.school34
Jun 21 14:46:30 dnsmasq[1598]: using local addresses only for domain school34
Jun 21 14:46:30 dnsmasq[1598]: using nameserver 77.88.8.7#53(via ethusb)
Jun 21 14:46:30 dnsmasq[1598]: cleared cache
Jun 21 14:46:33 dnsmasq[1598]: query[A] samba.account.school34 from 10.9.1.136
Jun 21 14:46:33 dnsmasq[1598]: config samba.account.school34 is 10.9.1.129
Jun 21 14:46:33 dnsmasq[1598]: query[AAAA] samba.account.school34 from 10.9.1.136
Jun 21 14:46:33 dnsmasq[1598]: config samba.account.school34 is NODATA-IPv6
Jun 21 14:46:33 dnsmasq[1598]: query[MX] samba.account.school34 from 10.9.1.136
Jun 21 14:46:33 dnsmasq[1598]: config samba.account.school34 is <MX>
Jun 21 14:46:33 dnsmasq[1598]: query[A] avers.school34 from 10.9.1.136
Jun 21 14:46:33 dnsmasq[1598]: config avers.school34 is NXDOMAIN
Jun 21 14:46:33 dnsmasq[1598]: query[A] avers.school34 from 10.9.1.136
Jun 21 14:46:33 dnsmasq[1598]: config avers.school34 is NXDOMAIN
Jun 21 14:46:33 dnsmasq[1598]: query[A] mail.service.school34 from 10.9.1.136
Jun 21 14:46:33 dnsmasq[1598]: config mail.service.school34 is NXDOMAIN
Jun 21 14:46:33 dnsmasq[1598]: query[A] mail.service.school34 from 10.9.1.136
Jun 21 14:46:33 dnsmasq[1598]: config mail.service.school34 is NXDOMAIN
Jun 21 14:46:33 dnsmasq[1598]: query[MX] yandex.ru from 10.9.1.136
Jun 21 14:46:33 dnsmasq[1598]: forwarded yandex.ru to 77.88.8.7
Jun 21 14:46:33 dnsmasq[1598]: query[AAAA] mcst.ru from 10.9.1.136
Jun 21 14:46:33 dnsmasq[1598]: forwarded mcst.ru to 77.88.8.7
Jun 21 14:46:36 dnsmasq[1598]: reply mcst.ru is NODATA-IPv6
Jun 21 14:46:40 dnsmasq[1598]: exiting on receipt of SIGTERM

```


Из приведённого фрагмента видно, что в 14:41 был выполнен сценарий тестирования при подключении к Интернету через основное соединение, а в 14:46 произошёл перезапуск демона в связи с переходом на мобильное соединение с Интернет с последующим повторным выполнением сценария. В первом случае получены положительные ответы на все запросы кроме разве IPv6-запросов, при том что отказ от IPv6 в сети был оговорен ранее. Во втором случае запросы о других подсетях организации кроме рассматриваемой были не только отклонены, но не было даже попыток перенаправить эти запросы вышестоящим dns-серверам. Полученные результаты полностью соответствуют требуемому режиму работы службы.

Проверка устойчивости системы к сбоям демонстрируется в следующем фрагменте терминального сеанса (отклики некоторых команд сокращены в неинформативной в применении к контексту части):

```
root@server:/# ps -ef | grep dnsmasq
dnsmasq  1100      1  0 13:34 ?        00:00:00 /usr/sbin/dnsmasq -x ...
root     1111    652  0 13:37 pts/0    00:00:00 grep dnsmasq
root@server:/# kill -9 1100
root@server:/# ps -ef | grep dnsmasq
dnsmasq  1121      1  0 13:37 ?        00:00:00 /usr/sbin/dnsmasq -x ...
root     1130    652  0 13:37 pts/0    00:00:00 grep dnsmasq
root@server:/# systemctl status dnsmasq
• dnsmasq.service - dnsmasq - A lightweight DHCP and caching DNS server
  Loaded: loaded (/lib/systemd/system/dnsmasq.service; enabled; vendor preset:
  enabled)
  Drop-In: /etc/systemd/system/dnsmasq.service.d
           └─override.conf
  Active: active (running) since Tue 2021-05-11 13:37:41 MSK; 18s ago
  Process: 1113 ExecStartPre=/usr/sbin/dnsmasq --test (code=exited, status=0/SUCCESS)
  Process: 1114 ExecStart=/etc/init.d/dnsmasq systemd-exec (code=exited,
  status=0/SUCCESS)
  Process: 1122 ExecStartPost=/etc/init.d/dnsmasq systemd-start-resolvconf
  (code=exited, status=0/SUCCESS)
  Main PID: 1121 (dnsmasq)
    Tasks: 1 (limit: 2315)
   Memory: 1.3M
    CGroup: /system.slice/dnsmasq.service
            └─1121 /usr/sbin/dnsmasq -x /run/dnsmasq/dnsmasq.pid -u dnsmasq -7 ...
```

В этом фрагменте выясняется PID текущего процесса dnsmasq (1100), затем этот процесс подвергается умышленному завершению, после чего повторно выясняется PID процесса dnsmasq. Из того, что PID = 1121 существует и отличается от предыдущего, следует однозначный вывод, что служба была перезапущена после падения. Это подтверждается также и выводом последней команды.

Служба точного времени

Настройка

В первую очередь требуется установить пакеты ntp и ntpdate со всеми зависимостями:

```
root@server:/# apt-get update
root@server:/# apt-get install ntp ntpdate
```

В состав этих пакетов входит ряд программ:

- ntpd – демон,
- ntpq – стандартная программа для запросов,
- ntpdc – расширенная программа для запросов,
- ntpdate – программа-клиент для установки времени в системе по ntp,
- sntp – простой сетевой клиент
- и другие (генераторы ключей, служебные, отладочные, симуляторы и др.)

Перед настройкой ntp.conf следует сначала настроить часовой пояс (файл `/etc/localtime`), что правильнее всего выполнить при помощи команды:

```
root@server:/# dpkg-reconfigure tzdata
```

Для настройки службы точного времени следует привести конфигурационный файл `/etc/ntp.conf` к виду:

```
# /etc/ntp.conf, configuration for ntpd; see ntp.conf(5) for help

driftfile /var/lib/ntp/ntp.drift

server ntp.school34      iburst prefer
server ntp1.vniiftri.ru iburst
server ntp2.vniiftri.ru iburst
server ntp3.vniiftri.ru iburst

interface ignore all
interface listen 192.168.0.88
interface listen 10.9.1.128/26

restrict default ignore
restrict ntp.school34      noquery notrap
restrict ntp1.vniiftri.ru noquery notrap
restrict ntp2.vniiftri.ru noquery notrap
restrict ntp3.vniiftri.ru noquery notrap
restrict 127.0.0.1
restrict 10.9.1.128 mask 255.255.255.192 kod notrap nomodify nopeer noquery limited
```

Пояснения к конфигурационному файлу:

`driftfile` – файл для записи частотной коррекции аппаратных часов, обновляется демоном один раз в час;

`interface` – параметр, описывающий взаимодействие с сетевыми интерфейсами системы; из нескольких таких директив к сетевому пакету применяется последняя подходящая.

`server` – используемый для синхронизации внешний сервер, таких серверов может быть задано несколько на случай недоступности одного или нескольких из них; дополнительный параметр `prefer` означает, что в случае доступности приоритет использования должен быть отдан именно этому серверу.

`iburst` – отсылать по 8 пакетов за 2 секунды вместо одного, это позволяет синхронизироваться быстрее (за несколько секунд вместо нескольких минут), однако не подходит для сетей с низкой пропускной способностью;

`kod` – kiss of death – отправлять в ответ на пакет, нарушающий ограничения по нагрузке на сервис ответный пакет с уведомлением;

`notrap` – не реализовывать функционал определения положения хоста по IPv6;

`nomodify` – отклонять запросы, которые пытаются изменить состояние сервера, разрешены только запросы, которые лишь получают ответ;

`nopeer` – отклонять не авторизованные запросы на установление связи, не касается пакетов, которые не устанавливают связь, т.е. клиентов обслуживать, но не синхронизироваться с ними;

`restrict default` – эта строка задает ограничения по умолчанию, здесь по умолчанию все пакеты кроме далее явно обозначенных игнорируются;

`restrict` – ввод ограничения на хост или сеть (хост может быть задан как именем, так и адресом, бессмысленно задавать именем хост, имеющий несколько ip адресов, как, например, `debian.pool.ntp.org`, т.к. в этом случае будет срабатывать правило по умолчанию), сеть задается своим адресом и маской, а далее следуют ключи, ограничивающие эту сеть или хост;

`limited` – отклонять запросы на синхронизацию, если превышены ограничения на трафик, заданные командой `discard` (там по умолчанию минимальный интервал между пакетами 1 секунда, а средний – 3 секунды), если при этом установлен еще и флаг `kod`, то отправляется ответный пакет;

`noquery` – отклонять запросы от `ntpq` и `ntpd`, сервис точного времени не затрагивается;

Таким образом, конфигурационный файл вынуждает `ntpd` работать по следующим правилам:

- 1. синхронизироваться разрешено только с заранее определенными ntp-серверами
- 2. по умолчанию игнорируются все пакеты, отправляемые к серверу, кроме явно разрешенных
- 3. внешним ntp-серверам не разрешено обращаться к локальному со служебными запросами или пользоваться IPv6
- 4. соединениям с самого сервера разрешено все, т.е. локально можно выполнять любые запросы к ntp-серверу, управлять им, отслеживать его состояние
- 5. локальные клиенты допускаются только из указанной подсети, они ограничены по пропускной способности, не могут влиять на сервер времени и посылать к нему служебные запросы, им разрешено лишь получать от сервера точное время

Остается лишь перезапустить подсистему точного времени командой

```
root@server:/# systemctl restart ntp
```

и убедиться, что служба находится в работоспособном состоянии, при помощи команды

```
root@server:/# systemctl status ntp
```

При отсутствии ошибок вывод этой команды будет похож на:

```
• ntp.service - Network Time Service
  Loaded: loaded (/lib/systemd/system/ntp.service; enabled; vendor preset: enabled)
  Active: active (running) since Wed 2021-05-12 11:54:46 +03; 6s ago
    Docs: man:ntpd(8)
  Process: 2487 ExecStart=/usr/lib/ntp/ntp-systemd-wrapper (code=exited,
status=0/SUCCESS)
 Main PID: 2493 (ntpd)
   Tasks: 2 (limit: 2315)
  Memory: 1.3M
   CGroup: /system.slice/ntp.service
```

```
└─2493 /usr/sbin/ntpd -p /var/run/ntpd.pid -g -u 108:113
```

```
мая 12 11:54:46 server systemd[1]: Started Network Time Service.
мая 12 11:54:46 server ntpd[2493]: proto: precision = 0.070 usec (-24)
мая 12 11:54:46 server ntpd[2493]: Listen and drop on 0 v6wildcard [::]:123
мая 12 11:54:46 server ntpd[2493]: Listen and drop on 1 v4wildcard 0.0.0.0:123
мая 12 11:54:46 server ntpd[2493]: Listen normally on 2 lo 127.0.0.1:123
мая 12 11:54:46 server ntpd[2493]: Listen normally on 3 enp3s0 10.9.1.129:123
мая 12 11:54:46 server ntpd[2493]: Listen normally on 4 enp2s0 192.168.0.88:123
мая 12 11:54:46 server ntpd[2493]: Listening on routing socket on fd #21 for
interface updates
мая 12 11:54:46 server ntpd[2493]: kernel reports TIME_ERROR: 0x2041: Clock
Unsynchronized
мая 12 11:54:46 server ntpd[2493]: kernel reports TIME_ERROR: 0x2041: Clock
Unsynchronized
```

В этом отклике следует отметить сообщения о том, что часы не синхронизированы (Clock Unsynchronized), что ожидаемо, т. к. служба запустилась только что, а для синхронизации требуется некоторое время. Следующие команды, выполненные через две минуты после перезапуска службы, демонстрируют успешность синхронизации:

```
root@server:/# /sbin/ntpdate -q localhost
server 127.0.0.1, stratum 2, offset 0.000005, delay 0.02567
12 May 11:58:07 ntpdate[2518]: adjust time server 127.0.0.1 offset 0.000005 sec
root@server:/# ntpq -p
      remote           refid      st t when poll reach  delay  offset  jitter
=====
*ntp3.vniiftri.r .MRS.          1 u   56  256  377   30.848 -0.296   2.041
+ntp1.vniiftri.r .MRS.          1 u  190  256  375   30.600 -0.207   0.298
+ntp2.vniiftri.r .MRS.          1 u  256  256  377   30.672 -0.338   3.232
```

В частности, здесь указано, что системные часы синхронизированы с локальным сервером с точностью до 0.000005 секунды, при этом локальный сервер имеет уровень stratum 2, отклонение его часов от часов вышестоящего сервера ntp.school34 0,296 миллисекунды, а дисперсия отклонений по результатам нескольких последних запросов составила 2,041 миллисекунды.

При заданных настройках демон ntpd тем не менее прослушивает все интерфейсы системы, но игнорирует пакеты согласно своему конфигурационному файлу. В том числе демон прослушивает и IPv6-адреса. В этом можно убедиться в выводе команды:

```
root@server:/# ss -ulp | grep ntp
UNCONN    0      0      192.168.0.88:ntp      0.0.0.0:*      users: (("ntpd",pid=385,fd=23))
UNCONN    0      0       10.9.1.129:ntp      0.0.0.0:*      users: (("ntpd",pid=385,fd=19))
UNCONN    0      0       127.0.0.1:ntp      0.0.0.0:*      users: (("ntpd",pid=385,fd=18))
UNCONN    0      0        0.0.0.0:ntp      0.0.0.0:*      users: (("ntpd",pid=385,fd=17))
UNCONN    0      0          [::]:ntp          [::]:*      users: (("ntpd",pid=385,fd=16))
```

Для принудительного отключения поддержки протокола IPv6 демоном следует привести конфигурационный файл **/etc/default/ntp** к виду:

```
NTPD_OPTS='-4 -g'
```

После перезапуска службы можно убедиться, что протокол IPv6 больше не используется:

```
root@server:/# systemctl restart ntp
root@server:/# ss -ul | grep ntp
UNCONN    0      0      192.168.0.88:ntp      0.0.0.0:*      users: (("ntpd",pid=815,fd=19))
UNCONN    0      0       10.9.1.129:ntp      0.0.0.0:*      users: (("ntpd",pid=815,fd=18))
```

```
UNCONN    0    0    127.0.0.1:ntp    0.0.0.0:*    users:(("ntpd",pid=815,fd=17))
UNCONN    0    0    0.0.0.0:ntp    0.0.0.0:*    users:(("ntpd",pid=815,fd=16))
```

Свой системный журнал служба ведет в файле `/var/log/daemon.log`, в который записывают свои сообщения и другие службы. Для получения выборки сообщений, связанных со службой ntp, за некоторый интервал времени (например 08:00-10:30 текущего дня) следует воспользоваться командой

```
root@server:/#journalctl -u ntp --since 08:00 --until 10:30
```

Защита от падений (автоматическое возобновление)

Для организации автоматического восстановления работоспособности службы после сбоя можно воспользоваться средствами системы инициализации systemd.

В первую очередь следует проверить текущие параметры службы командой:

```
root@server:/# systemctl show ntp
Type=forking
Restart=no
NotifyAccess=none
RestartUSec=100ms
TimeoutStartUSec=1min 30s
TimeoutStopUSec=1min 30s
...
```

Как видно из отклика команды (он приведён здесь в сильно сокращённом виде, содержащем важные в текущем контексте строки), перезапуск службы отключён.

Затем следует выполнить редактирование службы при помощи команды:

```
root@server:/# systemctl edit ntp
```

При этом будет открыт текстовый редактор, в котором необходимо внести определённые правки. В случае подтверждения сохранения изменений при выходе из текстового редактора в системе создаётся файл `/etc/systemd/system/ntp.service.d/override.conf`, в котором эти поправки и хранятся. При очередном запуске службы они переопределяют исходные значения параметров. Для решения задачи перезапуска службы при сбоях достаточно следующего содержимого этого файла:

```
[Service]
Restart=on-failure
```

Стоит отметить, что такие изменения в службе сохраняются и при установке обновлений системы.

После внесения изменений следует перезапустить службу и убедиться, что изменения параметров учтены:

```
root@server:/# systemctl restart ntp
root@server:/# systemctl show ntp
Type=forking
Restart=on-failure
NotifyAccess=none
RestartUSec=100ms
TimeoutStartUSec=1min 30s
```

TimeoutStopUSec=1min 30s

...

Проверка работоспособности и устойчивости

Проверка работоспособности службы тривиальна: с подключённого к внутренней сети Windows-клиента выполняется запрос на корректировку времени.

Проверка устойчивости системы к сбоям демонстрируется в следующем фрагменте терминального сеанса (отклики некоторых команд сокращены в неинформативной в применении к контексту части):

```
root@server:/# systemctl status ntp
• ntp.service - Network Time Service
  Loaded: loaded (/lib/systemd/system/ntp.service; enabled; vendor preset: enabled)
  Drop-In: /etc/systemd/system/ntp.service.d
           └─override.conf
  Active: active (running) since Thu 2021-05-13 09:56:43 +03; 5min ago
  Docs: man:ntpd(8)
  Process: 861 ExecStart=/usr/lib/ntp/ntp-systemd-wrapper (code=exited,
status=0/SUCCESS)
  Main PID: 867 (ntpd)
    Tasks: 2 (limit: 2315)
   Memory: 1.3M
   CGroup: /system.slice/ntp.service
           └─867 /usr/sbin/ntpd -p /var/run/ntpd.pid -4 -g -u 108:113
root@server:/# kill -9 867
root@server:/# systemctl status ntp
• ntp.service - Network Time Service
  Loaded: loaded (/lib/systemd/system/ntp.service; enabled; vendor preset: enabled)
  Drop-In: /etc/systemd/system/ntp.service.d
           └─override.conf
  Active: active (running) since Thu 2021-05-13 10:02:13 +03; 5s ago
  Docs: man:ntpd(8)
  Process: 878 ExecStart=/usr/lib/ntp/ntp-systemd-wrapper (code=exited,
status=0/SUCCESS)
  Main PID: 884 (ntpd)
    Tasks: 2 (limit: 2315)
   Memory: 1.2M
   CGroup: /system.slice/ntp.service
           └─884 /usr/sbin/ntpd -p /var/run/ntpd.pid -4 -g -u 108:113
```

В этом фрагменте проверяется статус службы точного времени, из ответного сообщения выясняется PID текущего процесса ntpd (867), затем этот процесс подвергается умышленному завершению, после чего повторно проверяется статус службы и выясняется PID вновь запущенного процесса ntpd (884). Работоспособность службы также подтверждается повторной успешной синхронизацией с сервером того же клиента.

Служба удалённого управления SSH

Установка ssh может быть выполнена как на этапе установки системы, так и после при помощи команд

```
root@server:/# apt-get update
root@server:/# apt-get install ssh
```

На данном сервере ssh используется как единственное средство удалённого входа и администрирования. Учитывая положение сервера в сети, ssh принимает соединения только на внешнем (недоступном для клиентов) интерфейсе. Тем не менее, должны быть предприняты существенные меры по защите системы удалённого администрирования. Эти меры распределены на три эшелона: сетевой фильтр, базовую настройку сервера и, собственно, настройку самой службы ssh.

О настройке сетевого экранирования сказано ниже, о базовой настройке сервера – выше при описании списков доступа и статических арг-записей.

Касательно настройки самой службы следует выделить два направления: защита от несанкционированного доступа и обеспечение безотказного функционирования.

Настройка

Основной конфигурационный файл службы `/etc/ssh/sshd_config` следует привести к виду:

```
Port 22
ListenAddress 192.168.0.88
AddressFamily inet
Protocol 2
PermitRootLogin no
AllowUsers administrator
PasswordAuthentication yes
PubkeyAuthentication no
KerberosAuthentication no
HostbasedAuthentication no
IgnoreRhosts yes
PermitEmptyPasswords no
X11Forwarding no
```

Тем самым обеспечивается следующий порядок работы сервера ssh:

Port 22 – сервер принимает сообщения на 22 порту по протоколу tcp.

Protocol 2 – используется протокол ssh версии 2.

AddressFamily inet – разрешено использование протокола IPv4, использование IPv6 отключено.

PermitRootLogin no – удалённый вход в систему от имени суперпользователя запрещён.

AllowUsers administrator – удалённый вход разрешён только для пользователя administrator.

PasswordAuthentication yes – разрешена аутентификация только по паролю, остальные методы запрещены, т. к. не используются.

PermitEmptyPasswords no – запрещено использование пустых паролей.

X11Forwarding no – не разрешена передача трафика по протоколу X11, т. к. он не используется сервером.

После приведения этого файла к заданному виду следует перезапустить службу ssh, затем убедиться, что она успешно запущена и принимает соединения только по протоколу IPv4. Это можно сделать следующими командами:

```
root@debian:/# systemctl restart sshd
root@debian:/# systemctl status sshd
```

- ssh.service - OpenBSD Secure Shell server
 Loaded: loaded (/lib/systemd/system/ssh.service; enabled; vendor preset: enabled)
 Active: active (running) since Wed 2021-05-05 11:58:06 MSK; 10s ago
 Docs: man:sshd(8)
 man:sshd_config(5)
 Process: 3838 ExecStartPre=/usr/sbin/sshd -t (code=exited, status=0/SUCCESS)
 Main PID: 3839 (sshd)
 Tasks: 1 (limit: 2315)
 Memory: 1.1M
 CGroup: /system.slice/ssh.service
 └─3839 /usr/sbin/sshd -D

```

мая 05 11:58:06 server.account.school34 systemd[1]: Starting OpenBSD Secure Shell
server...
мая 05 11:58:06 server.account.school34 sshd[3839]: Server listening on 192.168.0.88
port 22.
мая 05 11:58:06 server.account.school34 systemd[1]: Started OpenBSD Secure Shell
server.
root@debian:/# ss -Htulp | grep ssh
tcp LISTEN 0 128 192.168.0.88:ssh 0.0.0.0:* users:(("sshd",pid=3839,fd=3))

```

Свой системный журнал служба ведёт в файле **/var/log/daemon.log**, в который записывают свои сообщения и другие службы. Для получения выборки сообщений, связанных со службой ssh, за некоторый интервал времени (например за текущий день) следует воспользоваться командой

```
root@server:/#journalctl -u ssh --since today
```

Защита от падений (автоматическое возобновление)

Управлением службами в системе занимается менеджер загрузки и служб systemd, именно он и обеспечивает (при должной настройке) автоматическое возобновление служб в случае их падения. В рассматриваемой системе такая настройка для службы ssh выполнена изначально сборщиками дистрибутива операционной системы.

Проверка работоспособности и устойчивости

Для проверки состояния службы можно воспользоваться командой

```
root@debian:/# systemctl status ssh
```

Чтобы проверить, какие порты, адреса и интерфейсы прослушивает ssh, можно воспользоваться командой

```
root@debian:/# ss -Htulp | grep ssh
```

Следующий фрагмент терминального сеанса демонстрирует попытки входа в систему удалённо с разрешённого хоста под разными учётными записями:

```

administrator@admin:~$ ssh administrator@server.account.school34
administrator@server.account.school34's password:
Last login: Thu May 13 09:43:44 2021 from 192.168.0.2

```

The programs included with the Debian GNU/Linux system are free software; the exact distribution terms for each program are described in the individual files in /usr/share/doc/*/copyright.

Debian GNU/Linux comes with ABSOLUTELY NO WARRANTY, to the extent


```

permitted by applicable law.
administrator@server:~$ logout
Connection to server.account.school34 closed.
administrator@admin:~$ ssh root@server.account.school34
root@server.account.school34's password:
Permission denied, please try again.
root@server.account.school34's password:
Permission denied, please try again.
root@server.account.school34's password:
root@server.account.school34: Permission denied (password,keyboard-interactive).
administrator@admin:~$

```

Как видно из этого фрагмента, вход от имени administrator выполнен успешно, а вход от имени root не выполнен даже несмотря на ввод правильного пароля.

При попытке входа с хоста, отличного от разрешённого (как из внутренней сети, так и из внешней), клиент получает сообщение (при отключённом файрволе рассматриваемого сервера):

```

xubuntu@xubuntu:~$ ssh administrator@server.account.school34
ssh_exchange_identification: read: Connection reset by peer

```

Это является результатом работы списков контроля доступа. При включённом файрволе клиент получает иное сообщение:

```

xubuntu@xubuntu:~$ ssh administrator@server.account.school34
ssh: connect to host administrator@server.account.school34 port 22: Connection timed out

```

Однако в обоих случаях в подключении к серверу отказано.

Один из способов проверки автоматического восстановления работоспособности приведён в следующем терминальном сеансе:

```

root@server:/# date
Чт мая 13 11:21:07 +03 2021
root@server:/# ps -ef | grep ssh
root      420      1  0 08:00 ?        00:00:00 /usr/sbin/sshd -D
root      774     420  0 09:43 ?        00:00:00 sshd: administrator [priv]
adminis+  776     774  0 09:43 ?        00:00:00 sshd: administrator@pts/0
root      970     794  0 11:21 pts/0    00:00:00 grep ssh
root@server:/# kill -9 420
root@server:/# ps -ef | grep ssh
root      774      1  0 09:43 ?        00:00:00 sshd: administrator [priv]
adminis+  776     774  0 09:43 ?        00:00:00 sshd: administrator@pts/0
root      972      1  0 11:22 ?        00:00:00 /usr/sbin/sshd -D
root      975     794  0 11:22 pts/0    00:00:00 grep ssh
root@server:/# systemctl status sshd
• ssh.service - OpenBSD Secure Shell server
Loaded: loaded (/lib/systemd/system/ssh.service; enabled; vendor preset: enabled)
Active: active (running) since Thu 2021-05-13 11:22:08 +03; 14s ago
Docs: man:sshd(8)
     man:sshd_config(5)
Process: 971 ExecStartPre=/usr/sbin/sshd -t (code=exited, status=0/SUCCESS)
Main PID: 972 (sshd)
Tasks: 4 (limit: 2315)
Memory: 10.1M
CGroup: /system.slice/ssh.service
└─774 sshd: administrator [priv]
└─776 sshd: administrator@pts/0

```

```
└─777 -bash
└─972 /usr/sbin/sshd -D
```

```
мая 13 11:22:08 server systemd[1]: Starting OpenBSD Secure Shell server...
```

```
...
мая 13 11:22:08 server systemd[1]: Started OpenBSD Secure Shell server.
root@server:/#
```

Здесь сначала отображены системные дата и время, затем получен список выполняющихся процессов в системе, отфильтрованный по упоминанию ssh. Как можно видеть из этого списка, в системе работает процесс sshd с pid 420. Следующая команда убивает этот процесс. Далее демонстрируется, что процесс действительно убит, но уже работает новый процесс с PID 972 и служба успешно перезапущена. При этом следует отметить, что эксперимент выполнялся через ssh-подключение с компьютера системного администратора, но разрыва соединения не произошло.

Настройка NUT

Энергоснабжение рассматриваемого сервера осуществляется с помощью линейно-интерактивного источника бесперебойного питания Cyber Power BR650ELCD-RU. Для обеспечения правильной обработки перебоев с электропитанием на сервере следует установить и настроить систему NUT (network UPS tools).

Настройка

Начать следует с установки NUT. Система состоит из трёх компонентов: драйвера ИБП, демона и клиента. Команда

```
root@server:/# apt-get install nut
```

устанавливает все необходимые пакеты. По завершении её выполнения следует определиться с драйвером ИБП. Для этого в первую очередь необходимо найти в составе пакета nut-server файл-справочник по драйверам источников бесперебойного питания:

```
root@server:/# dpkg -L nut-server | grep driver
/lib/systemd/system/nut-driver.service
/usr/share/nut/driver.list
```

Далее следует собрать сведения об используемом ИБП при помощи команд (вывод частично сокращён в неинформативной части):

```
root@server:/# lsusb
```

```
...
Bus 003 Device 007: ID 0764:0501 Cyber Power System, Inc. CP1500 AVR UPS
```

```
...
root@server:/# lsusb -d 0764:0501 -v
```

```
Bus 003 Device 079: ID 0764:0501 Cyber Power System, Inc. CP1500 AVR UPS
```

```
Device Descriptor:
```

bLength	18
bDescriptorType	1
bcdUSB	1.10
bDeviceClass	0
bDeviceSubClass	0
bDeviceProtocol	0
bMaxPacketSize0	8

```

idVendor      0x0764 Cyber Power System, Inc.
idProduct     0x0501 CP1500 AVR UPS
bcdDevice     0.01
iManufacturer 3 CPS
iProduct      1 UPS VALUE
iSerial       0
bNumConfigurations 1
Configuration Descriptor:
  bLength      9
  bDescriptorType 2
  wTotalLength 0x0022
  bNumInterfaces 1
  bConfigurationValue 1
  iConfiguration 0
  bmAttributes 0xc0
    Self Powered
  MaxPower     50mA
Interface Descriptor:
  bLength      9
  bDescriptorType 4
  bInterfaceNumber 0
  bAlternateSetting 0
  bNumEndpoints 1
  bInterfaceClass 3 Human Interface Device
  bInterfaceSubClass 0
  bInterfaceProtocol 0

```

...

Уже из этого вывода можно на основании класса интерфейса предположить, что ИБП управляется драйвером `usbhid-ups`. Отклик команды

```

root@server:/# cat /usr/share/nut/driver.list | grep CP1500
"Cyber Power Systems" "ups" "2" "CP1500AVRLCD" "USB" "usbhid-ups"

```

подтверждает это предположение. Также в нём указано, что уровень поддержки равен 2, что означает фрагментарную поддержку открытого протокола.

Далее следует приступить к настройке компонентов NUT с учётом сведений, приведённых в руководстве драйвера

```

root@server:/# man 8 usbhid-ups

```

Во-первых потребуется обеспечить правильную реакцию системы на обнаружение ИБП. Так как это устройство принадлежит классу HID (Human Interface Device), то при обнаружении оно захватывается соответствующим драйвером, делая невозможным применение драйвера NUT. Чтобы этого избежать, надо поставить подсистему `udev` в известность о том, что драйвер HID не должен применяться. Именно это и было сделано в разделе «Настройка подсистемы UDEV».

Следующим шагом является заполнение конфигурационного файла `/etc/nut/ups.conf`. В этом файле должны быть созданы секции с описанием каждого устройства, подключённого к рассматриваемой системе (единственный ИБП в данном случае). Внутри этой секции должны быть указаны драйвер и необходимые для его работы параметры. На этом сервере файл имеет вид:

```

maxretry=3

```

```

[BR650ELCD]

```

```
driver=usbhid-ups
port=auto
vendorid=0764
productid=0501
default.battery.voltage.high=13.6
default.battery.voltage.low=10.5
desc="Cyber Power BR650ELCD-RU"
```

В начале файла стоит параметр `maxretry`. Он указывает количество попыток запуска драйвера. Этот параметр полезен для управления медленными устройствами. Далее следует заголовок секции, содержащий имя ИБП для NUT. Имя может быть выбрано достаточно свободно, но должно быть единым для нескольких конфигурационных файлов. Внутри секции указан драйвер, порт подключения ИБП, идентификаторы производителя и устройства, текстовое описание ИБП, а также напряжения, создаваемые батареей ИБП при высоком и низком уровнях заряда.

Параметр `port` обязателен, но при подключении по шине USB его значение может оказаться непостоянным даже если ИБП всегда подключён к одному и тому же разъёму. В таком случае и указывается значение `auto`. При этом поиск соответствующего устройства выполняется по иным критериям, например, по идентификаторам модели устройства и производителя. В применении к имеющемуся источнику питания следует отметить, что в памяти устройства не указан его серийный номер (значение 0 в приведённом выше дескрипторе устройства для поля `iSerial`). Поэтому в общем случае подключение двух таких ИБП к одному серверу (схема «Big Box» в терминологии NUT, применяемая, например, если сервер имеет два блока питания) может оказаться весьма проблематичным. Однако в рассматриваемой системе этой проблемы нет. Касательно значений напряжений, то они были выяснены заранее по опыту предыдущей эксплуатации ИБП, а их получение в настроенной системе будет продемонстрировано ниже. При первоначальной настройке нового ИБП эти данные могут быть «угаданы» на основании внутренней конструкции ИБП, документации к нему, подсказках драйвера при первых запусках и характеристиках используемых батарей. Так имеющееся устройство содержит одну батарею номинальным напряжением 12 В, откуда первоначальной оценкой значений параметров можно принять 12 В и 10 В для высокого и низкого заряда соответственно.

После настройки драйвера следует настроить демон `upsd`. Его конфигурационный файл `/etc/nut/upsd.conf` согласно имеющейся постановке задачи должен содержать лишь одну строку:

```
LISTEN 127.0.0.1 3493
```

Таким образом демон будет прослушивать стандартный порт NUT только на локальном петлевом интерфейсе. В случае если системному администратору потребуется удалённо управлять демоном, то подключение к системе следует выполнять по протоколу `ssh`. Конечно это препятствует созданию централизованной системы мониторинга ИБП уровня организации, но задача создания такой системы на данный момент не ставится, к тому же сервер бухгалтерии должен быть максимально автономен и защищён.

Кроме того, демону необходимо предоставить список разрешённых пользователей, их паролей и полномочий. Здесь в файл `/etc/nut/upsd.users` следует привести с точностью до комментариев к виду:

```
[upsmmon]
password = upspasswd
upsmmon master
```

Это означает, что демону известен пользователь `upsmmon` с паролем `upspasswd`, имеющий право не только получать данные от ИБП, но и управлять им.

Далее требуется настроить клиентскую программу, при помощи которой и будут выполняться обращения к демону. Простейшей такой программой является `upsmmon`, которая настраивается через файл `/etc/nut/upsmmon.conf`. Для рассматриваемой системы будет достаточно следующего конфигурационного файла:

```
RUN_AS_USER nut
MONITOR BR650ELCD@localhost 1 upsmmon upspasswd master
MINSUPPLIES 1
SHUTDOWNCMD "/sbin/shutdown -h +0"
POLLFREQ 5
POLLFREQALERT 5
HOSTSYNC 15
DEADTIME 15
POWERDOWNFLAG /etc/killpower
RBWARNTIME 43200
NOCOMMWARNTIME 300
FINALDELAY 5
```

В этом конфигурационном файле большинство параметров имеют значения по умолчанию. Пояснения к ним могут быть легко найдены в

```
root@server:/# man 5 upsmmon.conf
```

Следует отметить лишь то, что процесс `upsmmon` запускается от имени системного пользователя `nut` и отслеживает состояние ИБП `BRL650ELCD`, подключённого к серверу и управляемому также с него.

Наконец в файле `/etc/nut/nut.conf` следует выбрать режим работы всей системы NUT. В рассматриваемой ситуации этот файл должен содержать (кроме комментариев) лишь строку

```
MODE=standalone
```

Следует сделать ряд замечаний, касающихся безопасности системы. В файлах `upsmmon.conf` и `upsd.conf` открытым текстом указан пароль, позволяющий управлять питанием системы, пусть и только локально. Поэтому эти файлы не должны быть доступны для чтения никому, кроме системного пользователя, от имени которого они работают. При этом даже для этого пользователя они должны быть недоступны для записи. Эти меры уже реализованы сборщиками дистрибутива операционной системы, что хорошо видно из следующего фрагмента терминального сеанса:

```
root@server:/# ls -l /etc/nut
итого 24
-rw-r----- 1 root nut 1544 мая 31 10:12 nut.conf
-rw-r----- 1 root nut  208 мая 28 10:08 ups.conf
-rw-r----- 1 root nut   22 мая 31 10:02 upsd.conf
-rw-r----- 1 root nut 2184 мая 31 10:20 upsd.users
-rw-r----- 1 root nut  249 мая 31 10:32 upsmmon.conf
-rw-r----- 1 root nut 3887 июн  1 2018 upssched.conf
```

После завершения редактирования конфигурационных файлов остаётся лишь перезапустить в указанном порядке три службы компонентов NUT и убедиться в успешности этого перезапуска:

```
root@server:/# systemctl stop nut-monitor
root@server:/# systemctl stop nut-server
root@server:/# systemctl stop nut-driver
root@server:/# systemctl start nut-driver
root@server:/# systemctl start nut-server
root@server:/# systemctl start nut-monitor
root@server:/# systemctl status nut-driver
• nut-driver.service - Network UPS Tools - power device driver controller
  Loaded: loaded (/lib/systemd/system/nut-driver.service; static; vendor preset:
  enabled)
  Active: active (running) since Mon 2021-05-31 15:57:09 +03; 17s ago
  Process: 25208 ExecStart=/sbin/upsdrvctl start (code=exited, status=0/SUCCESS)
  Main PID: 25210 (usbhid-ups)
    Tasks: 1 (limit: 2314)
   Memory: 856.0K
    CGroup: /system.slice/nut-driver.service
            └─25210 /lib/nut/usbhid-ups -a BR650ELCD
```

```
мая 31 15:57:09 server systemd[1]: Starting Network UPS Tools - power device driver
controller...
мая 31 15:57:09 server upsdrvctl[25208]: Using subdriver: CyberPower HID 0.4
мая 31 15:57:09 server upsdrvctl[25208]: Network UPS Tools - Generic HID driver 0.41
(2.7.4)
мая 31 15:57:09 server upsdrvctl[25208]: USB communication driver 0.33
мая 31 15:57:09 server upsdrvctl[25208]: cps_adjust_battery_scale: battery readings
will be scaled by 2/3
мая 31 15:57:09 server upsdrvctl[25208]: Network UPS Tools - UPS driver controller
2.7.4
мая 31 15:57:09 server usbhid-ups[25210]: Startup successful
мая 31 15:57:09 server systemd[1]: Started Network UPS Tools - power device driver
controller.
```

```
root@server:/# systemctl status nut-server
• nut-server.service - Network UPS Tools - power devices information server
  Loaded: loaded (/lib/systemd/system/nut-server.service; enabled; vendor preset:
  enabled)
  Active: active (running) since Mon 2021-05-31 15:57:09 +03; 25s ago
  Process: 25211 ExecStart=/sbin/upsd (code=exited, status=0/SUCCESS)
  Main PID: 25212 (upsd)
    Tasks: 1 (limit: 2314)
   Memory: 716.0K
    CGroup: /system.slice/nut-server.service
            └─25212 /lib/nut/upsd
```

```
мая 31 15:57:09 server systemd[1]: Starting Network UPS Tools - power devices
information server...
мая 31 15:57:09 server upsd[25211]: fopen /var/run/nut/upsd.pid: No such file or
directory
мая 31 15:57:09 server upsd[25211]: listening on 127.0.0.1 port 3493
мая 31 15:57:09 server upsd[25211]: listening on 127.0.0.1 port 3493
мая 31 15:57:09 server upsd[25211]: Connected to UPS [BR650ELCD]: usbhid-ups-
BR650ELCD
мая 31 15:57:09 server upsd[25211]: Connected to UPS [BR650ELCD]: usbhid-ups-
BR650ELCD
мая 31 15:57:09 server upsd[25212]: Startup successful
мая 31 15:57:09 server systemd[1]: Started Network UPS Tools - power devices
information server.
мая 31 15:57:16 server upsd[25212]: User upsmon@127.0.0.1 logged into UPS [BR650ELCD]
```

```
root@server:/# systemctl status nut-monitor
```

```
• nut-monitor.service - Network UPS Tools - power device monitor and shutdown controller
```

```
Loaded: loaded (/lib/systemd/system/nut-monitor.service; enabled; vendor preset: enabled)
```

```
Active: active (running) since Mon 2021-05-31 15:57:16 +03; 28s ago
```

```
Process: 25215 ExecStart=/sbin/upsmon (code=exited, status=0/SUCCESS)
```

```
Main PID: 25217 (upsmon)
```

```
Tasks: 2 (limit: 2314)
```

```
Memory: 1.1M
```

```
CGroup: /system.slice/nut-monitor.service
```

```
└─25216 /lib/nut/upsmon
```

```
└─25217 /lib/nut/upsmon
```

```
мая 31 15:57:16 server systemd[1]: Starting Network UPS Tools - power device monitor and shutdown controller...
```

```
мая 31 15:57:16 server upsmon[25215]: fopen /var/run/nut/upsmon.pid: No such file or directory
```

```
мая 31 15:57:16 server upsmon[25215]: UPS: BR650ELCD@localhost (master) (power value 1)
```

```
мая 31 15:57:16 server upsmon[25215]: Using power down flag file /etc/killpower
```

```
мая 31 15:57:16 server upsmon[25216]: Startup successful
```

```
мая 31 15:57:16 server systemd[1]: nut-monitor.service: Can't open PID file /run/nut/upsmon.pid (yet?) after start: No such file or directory
```

```
мая 31 15:57:16 server systemd[1]: nut-monitor.service: Supervising process 25217 which is not our child. We'll most likely not notice when it exits.
```

```
мая 31 15:57:16 server systemd[1]: Started Network UPS Tools - power device monitor and shutdown controller.
```

```
мая 31 15:57:16 server upsmon[25217]: Init SSL without certificate database
```

Сохранение порядка перезапуска служб позволяет выполнить его без дополнительных сообщений о временных разрывах связи.

Защита от падений (автоматическое возобновление)

Для организации автоматического восстановления работоспособности служб после сбоя можно воспользоваться средствами системы инициализации systemd. При этом потеря связи с ИБП не является сбоем, таковым считается лишь аварийное завершение одного из процессов, которое хотя и крайне маловероятно, но может быть нивелировано следующими действиями.

В первую очередь следует проверить текущие параметры служб командами:

```
root@server:/# systemctl show nut-driver
```

```
Type=forking
```

```
Restart=no
```

```
NotifyAccess=none
```

```
RestartUSec=100ms
```

```
TimeoutStartUSec=1min 30s
```

```
TimeoutStopUSec=1min 30s
```

```
...
```

```
root@server:/# systemctl show nut-server
```

```
Type=forking
```

```
Restart=no
```

```
NotifyAccess=none
```

```
RestartUSec=100ms
```

```
TimeoutStartUSec=1min 30s
```

```
TimeoutStopUSec=1min 30s
```

```
...
```

```
root@server:/# systemctl show nut-monitor
Type=forking
Restart=no
PIDFile=/run/nut/upsmon.pid
NotifyAccess=none
RestartUSec=100ms
TimeoutStartUSec=1min 30s
TimeoutStopUSec=1min 30s
...
```

Как видно из отклика команд (он приведён здесь в сильно сокращённом виде, содержащем важные в текущем контексте строки), перезапуск служб отключён.

Затем следует выполнить редактирование каждой службы при помощи команд:

```
root@server:/# systemctl edit nut-driver
root@server:/# systemctl edit nut-server
root@server:/# systemctl edit nut-monitor
```

При этом будет открыт текстовый редактор, в котором необходимо внести определённые правки. При подтверждении сохранения изменений в системе создаются соответствующие файлы:

```
/etc/systemd/system/nut-driver.service.d/override.conf
/etc/systemd/system/nut-server.service.d/override.conf
/etc/systemd/system/nut-monitor.service.d/override.conf
```

При очередном запуске службы они переопределяют исходные значения параметров. Для решения задачи перезапуска службы при сбоях достаточно следующего содержимого, одинакового для всех файлов:

```
[Service]
Restart=on-failure
```

Стоит отметить, что такие изменения в службе сохраняются и при установке обновлений системы.

После внесения изменений следует перезапустить службы и убедиться, что изменения параметров учтены:

```
root@server:/# systemctl stop nut-monitor
root@server:/# systemctl stop nut-server
root@server:/# systemctl stop nut-driver
root@server:/# systemctl start nut-driver
root@server:/# systemctl start nut-server
root@server:/# systemctl start nut-monitor
root@server:/# systemctl show nut-driver
Type=forking
Restart=on-failure
NotifyAccess=none
RestartUSec=100ms
TimeoutStartUSec=1min 30s
TimeoutStopUSec=1min 30s
...
root@server:/# systemctl show nut-server
Type=forking
Restart=on-failure
NotifyAccess=none
RestartUSec=100ms
```



```
TimeoutStartUsec=1min 30s
TimeoutStopUsec=1min 30s
...
root@server:/# systemctl show nut-monitor
Type=forking
Restart=on-failure
PIDFile=/run/nut/upsmon.pid
NotifyAccess=none
RestartUsec=100ms
TimeoutStartUsec=1min 30s
TimeoutStopUsec=1min 30s
...
```

Проверка работоспособности и устойчивости

Для проверки текущего состояния ИБП следует использовать команду

```
root@server:/# upsc BR650ELCD
Init SSL without certificate database
battery.charge: 100
battery.charge.low: 10
battery.charge.warning: 20
battery.mfr.date: CPS
battery.runtime: 2100
battery.runtime.low: 300
battery.type: PbAcid
battery.voltage: 13.6
battery.voltage.high: 13.6
battery.voltage.low: 10.5
battery.voltage.nominal: 12
device.mfr: CPS
device.model: UPS VALUE
device.type: ups
driver.name: usbhid-ups
driver.parameter.pollfreq: 30
driver.parameter.pollinterval: 2
driver.parameter.port: auto
driver.parameter.productid: 0501
driver.parameter.synchronous: no
driver.parameter.vendorid: 0764
driver.version: 2.7.4
driver.version.data: CyberPower HID 0.4
driver.version.internal: 0.41
input.transfer.high: 280
input.transfer.low: 180
input.voltage: 234.0
input.voltage.nominal: 230
output.voltage: 234.0
ups.beeper.status: enabled
ups.delay.shutdown: 20
ups.delay.start: 30
ups.load: 20
ups.mfr: CPS
ups.model: UPS VALUE
ups.productid: 0501
ups.realpower.nominal: 390
ups.status: OL
ups.test.result: Done and passed
ups.timer.shutdown: -1
ups.timer.start: 0
ups.vendorid: 0764
```

В выводе этой команды наглядно показаны все параметры источника бесперебойного питания. Так, например, текущий уровень заряда батареи составляет 100% (battery.charge), низким зарядом, при котором инициируется отключение питания считается уровень в 10% (battery.charge.low), а предупредить систему о разряде батареи следует с уровня 20% (battery.charge.warning). При этом в текущий момент питание поступает от электросети (ups.status: OL), а напряжение на батарее при 100% заряда составляет 13.6 В.

Дальнейшую проверку работоспособности и устойчивости системы NUT можно разделить на несколько экспериментов:

Краткосрочная потеря связи с ИБП

Краткосрочная потеря связи с источником питания может быть имитирована физическим отключением usb-кабеля от ИБП. При этом согласно текущим настройкам upsmon краткосрочным считается отключение не более чем на 300 секунд.

Через 5 секунд после отключения кабеля в системе появилось широковещательное сообщение:

```
Broadcast message from nut@server (somewhere) (Wed Jun  2 15:14:11 2021):  
Communications with UPS BR650ELCD@localhost lost
```

После подключения кабеля появилось ещё одно такое сообщение:

```
Broadcast message from nut@server (somewhere) (Wed Jun  2 15:14:21 2021):  
Communications with UPS BR650ELCD@localhost established
```

Очевидно, что система отреагировала на оба события. Эти же сообщения также отражены в системном журнале:

```
root@server:/# cat /var/log/syslog | grep ups  
... Jun  2 15:14:08 server upsd[723]: Data for UPS [BR650ELCD] is stale - check driver  
Jun  2 15:14:11 server upsmon[726]: Poll UPS [BR650ELCD@localhost] failed - Data  
stale  
Jun  2 15:14:11 server upsmon[726]: Communications with UPS BR650ELCD@localhost lost  
Jun  2 15:14:16 server upsmon[726]: Poll UPS [BR650ELCD@localhost] failed - Data  
stale  
Jun  2 15:14:16 server upsd[723]: UPS [BR650ELCD] data is no longer stale  
Jun  2 15:14:21 server upsmon[726]: Communications with UPS BR650ELCD@localhost  
established
```

Долгосрочная потеря связи с ИБП

Эксперимент полностью аналогичен предыдущему, как и его результат: отсутствие связи с ИБП в течение 15 минут не спровоцировало выключение системы.

Аварийное завершение процессов NUT

Эксперимент следует проводить в следующем порядке: выяснить идентификаторы процессов, затем аварийно их завершить и наблюдать появляющиеся широковещательные сообщения:

```
root@server:/# ps -ef | grep ups  
nut      444      1  0 10:00 ?        00:00:01 /lib/nut/usbhid-ups -a BR650ELCD  
nut      462      1  0 10:00 ?        00:00:00 /lib/nut/upsd  
root     465      1  0 10:00 ?        00:00:00 /lib/nut/upsmon
```

```

nut      466    465    0 10:00 ?          00:00:00 /lib/nut/upsmon
root     634    496    0 13:53 pts/0      00:00:00 grep ups
root@server:/# kill -9 444

```

```

Broadcast message from nut@server (somewhere) (Thu Jun  3 13:54:15 2021):
Communications with UPS BR650ELCD@localhost lost

```

```

Broadcast message from nut@server (somewhere) (Thu Jun  3 13:54:20 2021):
Communications with UPS BR650ELCD@localhost established

```

```

root@server:/# kill -9 462

```

```

Broadcast message from nut@server (somewhere) (Thu Jun  3 13:54:35 2021):
Communications with UPS BR650ELCD@localhost lost

```

```

Broadcast message from nut@server (somewhere) (Thu Jun  3 13:54:40 2021):
Communications with UPS BR650ELCD@localhost established

```

```

root@server:/# kill -9 465

```

```

Broadcast message from nut@server (somewhere) (Thu Jun  3 13:54:55 2021):
upsmon parent process died - shutdown impossible

```

Из приведённого фрагмента терминального сеанса можно видеть, что процессы драйвера и демона перезапустились без замечаний, а процесс мониторинга был запущен в двух экземплярах: родительском и дочернем. При завершении родительского процесса, работающего от имени суперпользователя, система стала выдавать предупреждающие сообщения о том, что выключение системы при разряде батареи ИБП стало невозможным. При этом это сообщение повторяется в терминальном сеансе каждые две минуты. Повторный вызов команды

```

root@server:/# ps -ef | grep ups
nut      466      1    0 10:00 ?          00:00:00 /lib/nut/upsmon
nut      638      1    1 13:54 ?          00:00:07 /lib/nut/usbhid-ups -a BR650ELCD
nut      646      1    0 13:54 ?          00:00:00 /lib/nut/upsd
root     688    496    0 14:06 pts/0      00:00:00 grep ups

```

демонстрирует, что дочерний процесс upsmon продолжает выполняться от имени nut, а нового родительского процесса, работающего от имени root, нет. При этом служба systemd nut-monitor хотя остаётся активной.

Принудительный перезапуск службы полностью исправил ситуацию, что видно по новым идентификаторам процессов upsmon:

```

root@server:/# systemctl restart nut-monitor

```

```

root@server:/# systemctl status nut-monitor

```

```

• nut-monitor.service - Network UPS Tools - power device monitor and shutdown
controller
   Loaded: loaded (/lib/systemd/system/nut-monitor.service; enabled; vendor preset:
enabled)
   Drop-In: /etc/systemd/system/nut-monitor.service.d
            └─override.conf
   Active: active (running) since Thu 2021-06-03 14:12:21 +03; 2s ago

```

```

...
root@server:/# ps -ef | grep upsmon
root     705      1    0 14:12 ?          00:00:00 /lib/nut/upsmon
nut      706    705    0 14:12 ?          00:00:00 /lib/nut/upsmon
root     712    496    0 14:12 pts/0      00:00:00 grep upsmon

```

При повторном эксперименте аварийному завершению был подвергнут дочерний процесс, при этом служба была автоматически перезапущена без замечаний. При этом оба процесса, и дочерний, и родительский получили новые идентификаторы, т. е. также были запущены заново.

Таким образом можно сделать вывод, что работоспособность служб подтверждена, а меры по повышению устойчивости этих служб приняты не только при настройке этого сервера, но и при разработке самой системы NUT. Именно этим объясняются два процесса `upsmn`, т. к. родительский процесс не выполняет непосредственно полезную работу, но следит за состоянием запущенного дочернего процесса и при необходимости информирует систему, вызывая перезапуск всей службы. Такое разделение процессов позволяет вынести практически все риски аварийного завершения в пространство дочернего процесса.

Переход на батарейное питание и обратно без исчерпания заряда батареи

Имитация этой ситуации тривиальна: достаточно отключить ИБП от электросети и через некоторый короткий промежуток времени подключить его вновь. При этом в терминале системы были получены следующие широковещательные сообщения:

```
Broadcast message from nut@server (somewhere) (Thu Jun  3 14:31:18 2021):  
UPS BR650ELCD@localhost on battery
```

```
Broadcast message from nut@server (somewhere) (Thu Jun  3 14:31:38 2021):  
UPS BR650ELCD@localhost on line power
```

Они наглядно демонстрируют правильность функционирования системы.

Переход на батарейное питание и исчерпание заряда батареи

Для непрерывного мониторинга состояния ИБП в ходе этого эксперимента имеет смысл запустить команду

```
root@server:/# watch upsc BR650ELCD
```

При отключении ИБП от электросети через примерно две секунды появилось широковещательное сообщение

```
Broadcast message from nut@server (somewhere) (Fri Jun  4 13:56:18 2021):  
UPS BR650ELCD@localhost on battery
```

Затем наблюдалось равномерное снижение уровня заряда батареи, источник бесперебойного питания периодически выдавал звуковое оповещение, его индикатор показывал нагрузку 11%, что совпадало с данными, получаемыми от `upsc`. Наконец, через 50 минут работы от батареи её заряд был исчерпан на 90%, системой было принято решение об отключении, о чём свидетельствовали следующие широковещательные сообщения:

```
Broadcast message from nut@server (somewhere) (Fri Jun  4 14:46:49 2021):  
Executing automatic power-fail shutdown
```

```
Broadcast message from nut@server (somewhere) (Fri Jun  4 14:46:49 2021):  
UPS BR650ELCD@localhost battery is low
```

```
Broadcast message from nut@server (somewhere) (Fri Jun  4 14:46:49 2021):  
Auto logout and shutdown proceeding
```

```
exit
```

```
Session terminated, killing shell... ...killed.  
Terminated
```

Таким образом система сработала ожидаемым образом, штатно выключив сервер и сохранив некоторую часть заряда батареи с целью предотвращения её глубокого разряда и снижения ресурса.

Возврат сетевого питания после отключения системы

Через некоторое время (около минуты) после завершения предыдущего эксперимента ИБП был вновь подключён к электросети, после чего сервер автоматически включился. Этот результат ожидаем, т. к. отключение сервера произошло именно из-за исчезновения внешнего для ИБП напряжения.

Возврат сетевого питания в процессе отключения системы

При следующем отключении отключения системы при низком заряде батареи ИБП был вновь включён в электросеть именно в процессе выключения сервера. Результат также оказался желаемым: ИБП дождался выключения системы, выдержал некоторую паузу и инициировал загрузку сервера.

Возврат сетевого питания при штатно отключённой системе

Последний эксперимент в серии заключался в штатном выключении сервера без привязки к событиям питания с последующими отключением и подключением внешней электросети к ИБП. Включения сервера не произошло, что и следовало ожидать.

Таким образом, взаимодействие всех компонентов системы (сервер, ИБП, электросеть) отвечает постановке задачи.

Настройка RAID1

В рассматриваемой системе для развёртывания RAID-массива выделены два диска Hitachi объёмом по 500 Гб каждый, что видно из следующего сокращённого в не информативной части фрагмента терминального сеанса:

```
root@server:/# lshw  
server  
...  
    *-disk:0  
...  
    *-disk:1  
        description: ATA Disk  
        product: HGST HTS725050A7  
        physical id: 0.1.0  
        bus info: scsi@0:0.1.0  
        logical name: /dev/sdb  
        version: A530  
        serial: TF650AWE06VJ4V  
        size: 465GiB (500GB)  
...  
    *-disk:2  
        description: ATA Disk
```

```
product: HGST HTS725050A7
physical id: 1
bus info: scsi@1:0.0.0
logical name: /dev/sdc
version: A530
serial: TF650AWJ3UJ0TV
size: 465GiB (500GB)
```

...

При этом диск с серийным номером TF650AWE06VJ4V установлен в верхний слот в корпусе сервера, а диск с серийным номером TF650AWJ3UJ0TV, соответственно, в нижний. Знание физического расположения дисков может несколько ускорить процесс замены вышедшего из строя диска. Несмотря на то, что на момент настройки системы в наличии имеются запасные жёсткие диски той же модели, их установка в сервер не производится по следующим соображениям:

Во-первых, вероятность выхода дисков из строя невелика. Во-вторых, наличие запасного диска означает дополнительное энергопотребление и тепловыделение в системе на протяжении всего времени её работы. В-третьих, даже несмотря на то, что запасной диск не будет использоваться для хранения данных до выхода из строя одного из основных, при каждой загрузке системы он всё же будет запускаться, а при отключении системы останавливаться, что уже будет являться бесцельным расходом его механического ресурса. К тому же, материнская плата рассматриваемой системы не поддерживает «горячее» подключение/отключение дисков, а следовательно при выходе из строя одного из дисков для обслуживания системы всё равно потребуется её отключение.

Настройка

Настройку RAID-массива можно разделить на несколько этапов:

- установка необходимого ПО
- сборка и инициализация массива
- настройка автоматической сборки массива при загрузке системы
- создание раздела и файловой системы на RAID
- настройка автоматического монтирования раздела, размещённого на RAID
- обновление начального загрузочного диска системы

В первую очередь необходимо установить пакет mdadm:

```
root@server:/# apt-get install mdadm
```

Создание массива выполняется командой вида:

```
root@server:/# PATH=$PATH:/sbin
root@server:/# mdadm --create /dev/md0 /dev/sdb /dev/sdc --level=1 --raid-devices=2
mdadm: partition table exists on /dev/sdb
mdadm: partition table exists on /dev/sdb but will be lost or
      meaningless after creating array
mdadm: Note: this array has metadata at the start and
      may not be suitable as a boot device. If you plan to
      store '/boot' on this device please ensure that
      your boot-loader understands md/v1.x metadata, or use
      --metadata=0.90
mdadm: partition table exists on /dev/sdc
mdadm: partition table exists on /dev/sdc but will be lost or
      meaningless after creating array
```

```
Continue creating array? y
mdadm: Defaulting to version 1.2 metadata
mdadm: array /dev/md0 started.
```

Следует дать пояснения к сообщениям, полученным в процессе создания массива. Так mdadm предупреждает, что на каждом из дисков существует таблица разделов, которая будет потеряна при создании массива. Также система предупреждает, что метаданные массива будут храниться в начале дисков, а значит для использования массива как загрузочного раздела следует убедиться, что загрузчик системы способен с этим справиться. Очевидно, что сохранение таблиц разделов на дисках не требуется, также RAID-массив в рассматриваемой системе не является загрузочным устройством. Поэтому создание массива подтверждено, массив создан и успешно запущен, что видно из вывода команды lsblk:

```
root@server:/# lsblk
NAME MAJ:MIN RM SIZE RO TYPE MOUNTPOINT
sda   8:0    0 298,1G  0 disk
├─sda1 8:1    0  18,6G  0 part /
├─sda2 8:2    0   3,7G  0 part [SWAP]
├─sda3 8:3    0  18,6G  0 part /var
└─sda4 8:4    0 257,1G  0 part /srv/samba
sdb   8:16   0 465,8G  0 disk
└─md0  9:0    0 465,7G  0 raid1
sdc   8:32   0 465,8G  0 disk
└─md0  9:0    0 465,7G  0 raid1
```

Теперь на запущенном массиве требуется создать файловую систему:

```
root@server:/# mkfs.ext4 /dev/md0
mke2fs 1.44.5 (15-Dec-2018)
Creating filesystem with 122063616 4k blocks and 30523392 inodes
Filesystem UUID: e401ae19-45a0-4766-aed0-fd34ebd42f8f
Superblock backups stored on blocks:
    32768, 98304, 163840, 229376, 294912, 819200, 884736, 1605632, 2654208,
    4096000, 7962624, 11239424, 20480000, 23887872, 71663616, 78675968,
    102400000
Allocating group tables: done
Writing inode tables: done
Creating journal (262144 blocks): done
Writing superblocks and filesystem accounting information: done
```

По завершении весьма полезно, хотя и не строго обязательно, выполнить пробное монтирование системы:

```
root@server:/# mkdir /srv/samba/archive
root@server:/# mount /dev/md0 /srv/samba/archive/
root@server:/# mount | grep md0
/dev/md0 on /srv/samba/archive type ext4 (rw,relatime)
```

Далее запись о монтировании массива необходимо внести в файл **/etc/fstab** очевидно с тем, чтобы монтирование происходило автоматически при загрузке системы. Несмотря на то, что запись в этом файле может начинаться с имени устройства (/dev/md0), а не уникального идентификатора (UUID), для согласованности с предыдущими записями в нём и однозначной идентификации массива использован именно UUID). Получить этот идентификатор можно следующей командой:

```
root@server:/# blkid /dev/md0
/dev/md0: UUID="e401ae19-45a0-4766-aed0-fd34ebd42f8f" TYPE="ext4"
```

С учётом логики использования массива имеет смысл задать достаточно строгие параметры монтирования, что в итоге приводит к такой (указанный выше UUID сокращён чтобы вписать запись на странице одной строкой) записи в **/etc/fstab**:

```
UUID=e4...f8f    /srv/samba/archive ext4    nodev,nosuid,noexec    0    2
```

Для тонкой настройки массива в первую очередь необходимо заполнить файл **/etc/mdadm/mdadm.conf** согласно руководству

```
root@server:/# man 5 mdadm.conf
```

Большую часть необходимых для этого данных можно получить из вывода команд

```
root@server:/# /sbin/mdadm --examine --scan
root@server:/# lsblk
```

В рассматриваемой системе этот файл с точностью до комментариев следует привести к виду:

```
HOMEHOST <system>
DEVICE /dev/sdb /dev/sdc
ARRAY /dev/md0 devices=/dev/sdb,/dev/sdc
AUTO homehost -all
MAILADDR root@localhost
```

Терм **<system>** в первой строке означает, что имя хоста следует взять из системных настроек. Строка **AUTO** описывает, какие массивы подлежат автоматической сборке при загрузке системы. В данном случае собирать следует только массивы, в метаданные которых внесено то же самое имя хоста, что и используемое системой. В строке **ARRAY** следует указывать минимально необходимое для сборки массива множество параметров. На рассматриваемой системе достаточно лишь имён дисков.

После завершения редактирования файла следует обновить образ загрузочного диска, перезагрузить систему и после перезагрузки убедиться при помощи команды **mount** или нижеописанными способами в работоспособности массива и файловой системы на нём:

```
root@server:/# PATH=$PATH:/sbin
root@server:/# update-initramfs -u
update-initramfs: Generating /boot/initrd.img-4.19.0-16-amd64
root@server:/# reboot
```

При использовании RAID-массивов важно не только иметь возможность такой массив создать, но и иметь все необходимые средства для контроля его текущего состояния, доступа к данным при повреждении его компонентов, замены этих компонентов. Необходимые для этого действия описаны далее в разделе «Использование и обслуживание сервера».

Защита от падений (автоматическое возобновление)

Для контроля в реальном времени состояния массива в системе непрерывно работает процесс **mdadm**:


```
root@server:/# ps -ef | grep mdadm
root      301      1   0  08:00 ?                00:00:00 /sbin/mdadm --monitor --scan
```

Запуск этого процесса осуществляется службой mdmonitor, что можно увидеть в выводе следующей команды:

```
root@server:/# systemctl status mdmonitor
• mdmonitor.service - MD array monitor
   Loaded: loaded (/lib/systemd/system/mdmonitor.service; static; vendor preset:
   enabled)
   Active: active (running) since Tue 2021-05-25 08:00:36 +03; 6h ago
 Main PID: 301 (mdadm)
    Tasks: 1 (limit: 2314)
   Memory: 476.0K
   CGroup: /system.slice/mdmonitor.service
           └─301 /sbin/mdadm --monitor --scan
...
```

По умолчанию эта служба не настроена на автоматический перезапуск:

```
root@server:/# systemctl show mdmonitor
Type=simple
Restart=no
NotifyAccess=none
RestartUsec=100ms
TimeoutStartUsec=1min 30s
TimeoutStopUsec=1min 30s
...
```

Поэтому следует выполнить редактирование службы (подобно другим ранее описанным) при помощи команды:

```
root@server:/# systemctl edit mdmonitor
```

При этом будет открыт текстовый редактор, в котором необходимо внести некоторые правки. Они будут сохранены в `/etc/systemd/system/mdmonitor.service.d/override.conf`. При очередном запуске службы они переопределяют исходные значения параметров. Для решения задачи перезапуска службы при сбоях достаточно следующего содержимого этого файла:

```
[Service]
Restart=on-failure
```

Стоит отметить, что такие изменения в службе сохраняются и при установке обновлений системы.

После внесения изменений следует перезапустить службу и убедиться, что изменения параметров учтены:

```
root@server:/# systemctl restart mdmonitor
root@server:/# systemctl show mdmonitor
Type=forking
Restart=on-failure
NotifyAccess=none
RestartUsec=100ms
TimeoutStartUsec=1min 30s
TimeoutStopUsec=1min 30s
...
```

Проверка работоспособности и устойчивости

Проверка работоспособности службы тривиальна: успешное монтирование файловой системы, расположенной на массиве означает его работоспособность.

Проверка устойчивости системы непрерывного мониторинга состояния массива к сбоям демонстрируется в следующем фрагменте терминального сеанса (отклики некоторых команд сокращены в неинформативной в применении к контексту части):

```
root@server:/# systemctl status mdmonitor
• mdmonitor.service - MD array monitor
  Loaded: loaded (/lib/systemd/system/mdmonitor.service; static; vendor preset:
enabled)
  Drop-In: /etc/systemd/system/mdmonitor.service.d
           └─override.conf
  Active: active (running) since Tue 2021-05-25 14:51:16 +03; 4s ago
  Main PID: 12788 (mdadm)
    Tasks: 1 (limit: 2314)
   Memory: 404.0K
    CGroup: /system.slice/mdmonitor.service
            └─12788 /sbin/mdadm --monitor --scan
```

мая 25 14:51:16 server systemd[1]: Started MD array monitor.

```
root@server:/# kill -9 12788
```

```
root@server:/# systemctl status mdmonitor
• mdmonitor.service - MD array monitor
  Loaded: loaded (/lib/systemd/system/mdmonitor.service; static; vendor preset:
enabled)
  Drop-In: /etc/systemd/system/mdmonitor.service.d
           └─override.conf
  Active: active (running) since Tue 2021-05-25 14:51:36 +03; 1s ago
  Main PID: 12797 (mdadm)
    Tasks: 1 (limit: 2314)
   Memory: 412.0K
    CGroup: /system.slice/mdmonitor.service
            └─12797 /sbin/mdadm --monitor --scan
```

мая 25 14:51:36 server systemd[1]: Started MD array monitor.

В этом фрагменте проверяется статус службы, из ответного сообщения выясняется PID текущего процесса mdadm (12788), затем этот процесс подвергается умышленному завершению, после чего повторно проверяется статус службы и выясняется PID вновь запущенного процесса mdadm (12797).

Настройка Samba

Согласно постановке задачи, а также разметке дисков и структуре RAID-массива служба Samba должна обеспечивать работу трёх сетевых каталогов по протоколу smb. Первый из них играет роль сетевой папки для оперативного обмена файлами, второй предназначен для помещения в него файлов, которые следует заархивировать и поместить в третий каталог в конце рабочего дня. При этом на RAID-массиве располагается только третий каталог – архив. RAID-массив смонтирован в системе как **/srv/samba/archive**, а два других каталога расположены на разделе системного жёсткого диска размером 240 Гб, смонтированного как **/srv/samba**, и называются соответственно **/srv/samba/shared** и **/srv/samba/backup**. При этом архивный каталог не доступен напрямую для записи сотрудникам бухгалтерии. Он позволяет им лишь доступ на чтение. Остальные два

каталога доступны и для чтения, и для записи. Как было отмечено в постановке задачи, запись в архив выполняется сервером на основе содержимого каталога **/srv/samba/backup** в конце рабочего дня. Тем самым достигается невозможность повредить или уничтожить содержимое архива случайными или намеренными действиями персонала или программного обеспечения (например, вируса-шифровальщика, что весьма вероятно, учитывая тот факт, что рабочие места бухгалтеров работают под управлением MS Windows). Однако такой подход делает актуальной проблему переполнения дискового пространства архива. Для её решения в составе сценария, выполняющего ежедневное архивирование, предусмотрен ряд команд для проверки доступного дискового пространства и уведомления системного администратора об угрозе переполнения диска. На момент написания этого документа используется временный метод информирования путём создания предупреждающего текстового файла в каталоге **/srv/samba/backup**, а также внесением записи в системный журнал при помощи команды `logger`. В ближайшей перспективе по мере введения в строй почтового сервера организации будет организовано информирование системного администратора о событиях в системе (переполнение дисков, их сбои, прочие проблемы) по внутренней электронной почте. Использование в данный момент публичных почтовых серверов расценивается как угроза безопасности системы, а локальная доставка почты отключена, т. к. может быть причиной переполнения дискового пространства системы, о чём подробнее сказано далее.

Перед описанием настройки службы уместно коротко осветить её внутреннее устройство и смежные вопросы. Так служба состоит из двух демонов: `smbd` и `nmbd`. Первый отвечает за сетевой доступ к файлам и принтерам, второй – за обслуживание NetBIOS-запросов. Настройка этих демонов выполняется единым конфигурационным файлом. Подробные сведения о `samba`, `smbd`, `nmbd` и конфигурационном файле можно получить из следующих встроенных руководств:

```
root@server:/# man 7 samba
root@server:/# man 8 smbd
root@server:/# man 8 nmbd
root@server:/# man 5 smb.conf
```

Кроме того следует отметить, что с этой службой ассоциирован пользователь `buh`. Следует понимать, что на самом деле таких пользователей два. Системный пользователь `buh` лишён командной оболочки, не имеет пароля и возможности входа в систему. Он нужен лишь для управления правами доступа в сетевых каталогах. Во внутренней базе пользователей Samba присутствует одноимённый пользователь, пароль для которого как раз назначен. Именно этот пользователь и его пароль подразумеваются к использованию при подключении к сетевому каталогу клиентской системы.

Настройка

В первую очередь потребуется установить пакеты:

```
root@server:/# apt-get install samba zip
```

А затем привести конфигурационный файл **/etc/samba/smb.conf** к виду:

```
[global]
workgroup = ACCOUNT
netbios name = samba
interfaces = enp3s0 lo
```

```
bind interfaces only = yes
hosts allow = 10.9.1.128/26 127.0.0.1
hosts deny = ALL
security = user
passdb backend = tdbsam
domain logons = no
domain master = no
```

```
[shared]
comment = Текущий обмен файлами
path = /srv/samba/shared
valid users = buh
force user = buh
force group = buh
read only = no
guest ok = no
```

```
[backup]
comment = Каталог для архивирования
path = /srv/samba/backup
valid users = buh
force user = buh
force group = buh
read only = no
guest ok = no
```

```
[archive]
comment = Архив
path = /srv/samba/archive
valid users = buh
force user = buh
force group = buh
read only = yes
guest ok = no
```

Приведённый конфигурационный файл состоит из нескольких секций: глобальной и описывающих каждый сетевой каталог отдельно. В глобальной секции указаны имя системы для NetBIOS, имя рабочей группы для служб сетевого обнаружения клиентских операционных систем, допустимые для приёма запросов адреса и интерфейсы, используемые механизмы безопасности и их параметры. Для каждого сетевого каталога кроме очевидно необходимого пути в локальном дереве каталогов указан также ряд параметров, определяющих взаимодействие с ним. Так заданы текстовые комментарии, пользователи, которым разрешено получать к ним доступ, а также пользователь и группа, от имени которых будут выполняться все действия в этих каталогах. Подробное описание каждого такого параметра здесь лишено смысла т. к. их названия прозрачны, подробная, объёмная и хорошо структурированная документация представлена вышеуказанными встроенными руководствами.

Следующим шагом требуется создать эти каталоги (кроме каталога archive созданного при развёртывании RAID-массива) и установить к ним (в том числе и к archive) соответствующие права доступа:

```
root@server:/# mkdir /srv/samba/shared /srv/samba/backup
root@server:/# chown buh:buh -R /srv/samba/shared
root@server:/# chown buh:buh -R /srv/samba/backup
root@server:/# chown buh:buh -R /srv/samba/archive
root@server:/# chmod 0700 -R /srv/samba/shared
```

```
root@server:/# chmod 0700 -R /srv/samba/backup
root@server:/# chmod 0500 -R /srv/samba/archive
```

При этом следует отметить, что сценарий резервного копирования, запускаемый от имени суперпользователя, при таких правах доступа всё равно имеет возможность создания файлов в каталогах архива.

Далее потребуется добавить пользователя buh в базу паролей Samba:

```
root@server:/# smbpasswd -a buh
```

После чего рекомендуется проверить правильность заполнения конфигурационного файла командой

```
root@server:/# testparm
```

Отклик этой команды практически полностью дублирует конфигурационный файл и поэтому здесь не приведён.

Наконец, остаётся перезапустить службы smbd и nmbd и убедиться в успешности этого перезапуска:

```
root@server:/# service smbd restart
root@server:/# service nmbd restart
root@server:/# systemctl status smbd
• smbd.service - Samba SMB Daemon
  Loaded: loaded (/lib/systemd/system/smbd.service; enabled; vendor preset: enabled)
  Active: active (running) since Tue 2021-06-08 15:27:52 +03; 2min 53s ago
    Docs: man:smbd(8)
           man:samba(7)
           man:smb.conf(5)
  Process: 517 ExecStartPre=/usr/share/samba/update-apparmor-samba-profile
(code=exited, status=0/SUCCESS)
 Main PID: 522 (smbd)
   Status: "smbd: ready to serve connections..."
    Tasks: 4 (limit: 2314)
  Memory: 7.2M
   CGroup: /system.slice/smbd.service
           └─522 /usr/sbin/smbd --foreground --no-process-group
             └─525 /usr/sbin/smbd --foreground --no-process-group
               └─526 /usr/sbin/smbd --foreground --no-process-group
                 └─527 /usr/sbin/smbd --foreground --no-process-group
```

```
июн 08 15:27:52 server systemd[1]: Starting Samba SMB Daemon...
июн 08 15:27:52 server smbd[522]: [2021/06/08 15:27:52.848478,  0]
../lib/util/become_daemon.c:138(daemon_ready)
июн 08 15:27:52 server systemd[1]: Started Samba SMB Daemon.
июн 08 15:27:52 server smbd[522]:  daemon_ready: STATUS=daemon 'smbd' finished
starting up and ready to serve connections
```

```
root@server:/# systemctl status nmbd
• nmbd.service - Samba NMB Daemon
  Loaded: loaded (/lib/systemd/system/nmbd.service; enabled; vendor preset: enabled)
  Active: active (running) since Tue 2021-06-08 15:05:56 +03; 24min ago
    Docs: man:nmbd(8)
           man:samba(7)
           man:smb.conf(5)
 Main PID: 435 (nmbd)
   Status: "nmbd: ready to serve connections..."
    Tasks: 1 (limit: 2314)
  Memory: 14.5M
```

```
CGroup: /system.slice/nmbd.service
└─435 /usr/sbin/nmbd --foreground --no-process-group
```

```
июн 08 15:27:22 server nmbd[435]:
июн 08 15:27:22 server nmbd[435]: Samba name server SAMBA has stopped being a local
master browser for workgroup ACCOUNT on subnet 10.9.1.129
июн 08 15:27:22 server nmbd[435]:
июн 08 15:27:22 server nmbd[435]: *****
июн 08 15:27:39 server nmbd[435]: [2021/06/08 15:27:39.479661, 0]
../source3/nmbd/nmbd_become_lmb.c:397(become_local_master_stage2)
июн 08 15:27:39 server nmbd[435]: *****
июн 08 15:27:39 server nmbd[435]:
июн 08 15:27:39 server nmbd[435]: Samba name server SAMBA is now a local master
browser for workgroup ACCOUNT on subnet 10.9.1.129
июн 08 15:27:39 server nmbd[435]:
июн 08 15:27:39 server nmbd[435]: *****
```

Защита от падений (автоматическое возобновление)

Для организации автоматического восстановления работоспособности демонов `smbd` и `nmbd` после сбоя можно подобно другим ранее описанным службам воспользоваться средствами системы инициализации `systemd`.

Из следующего фрагмента терминального сеанса, демонстрирующего начальное состояние служб Samba, можно видеть, что их автоматический перезапуск не предусмотрен:

```
root@server:/# systemctl show smbd
```

```
Type=notify
Restart=no
PIDFile=/run/samba/smbd.pid
NotifyAccess=all
RestartUSec=100ms
TimeoutStartUSec=1min 30s
TimeoutStopUSec=1min 30s
```

```
...
```

```
root@server:/# systemctl show nmbd
```

```
Type=notify
Restart=no
PIDFile=/run/samba/nmbd.pid
NotifyAccess=all
RestartUSec=100ms
TimeoutStartUSec=1min 30s
TimeoutStopUSec=1min 30s
```

```
...
```

Затем следует выполнить редактирование каждой службы при помощи команд:

```
root@server:/# systemctl edit smbd
```

```
root@server:/# systemctl edit nmbd
```

При этом будет открыт текстовый редактор, в котором необходимо внести определённые правки. При подтверждении сохранения изменений в системе создаются соответствующие файлы:

```
/etc/systemd/system/smbd.service.d/override.conf
```

```
/etc/systemd/system/nmbd.service.d/override.conf
```

При очередном запуске службы они переопределяют исходные значения параметров. Для решения задачи перезапуска службы при сбоях достаточно следующего содержимого, одинакового для всех файлов:

```
[Service]
Restart=on-failure
```

Эти изменения в службе сохраняются и при установке обновлений системы.

После внесения изменений следует перезапустить службы и убедиться, что изменения параметров учтены:

```
root@server:/# systemctl restart smbd
root@server:/# systemctl show smbd
Type=notify
Restart=on-failure
PIDFile=/run/samba/smbd.pid
NotifyAccess=all
RestartUsec=100ms
TimeoutStartUsec=1min 30s
TimeoutStopUsec=1min 30s
...
root@server:/# systemctl restart nmbd
root@server:/# systemctl show nmbd
Type=notify
Restart=on-failure
PIDFile=/run/samba/nmbd.pid
NotifyAccess=all
RestartUsec=100ms
TimeoutStartUsec=1min 30s
TimeoutStopUsec=1min 30s
...
```

Проверка работоспособности и устойчивости

Проверку работоспособности системы можно разделить на следующие составляющие:

- проверка доступности сервера по имени, ip-адресу и средствами сетевого обнаружения
- проверка доступности сетевого каталога для оперативного обмена файлами на чтение и запись
- проверка доступности сетевого каталога для архивирования на чтение и запись
- проверка доступности сетевого архива на чтение и недоступности на запись (удаление файлов)

Для проверки был использован клиент под управлением Microsoft Windows 7. Первичная проверка была выполнена средствами графического пользовательского интерфейса и прошла успешно. Следующий фрагмент терминального сеанса демонстрирует обнаружение сетевых каталогов и пошаговую проверку каждого из них средствами командной строки Windows. Следует заметить, что для проверки каталога архива в нём средствами самого сервера был создан тестовый файл, удалённый также средствами сервера по завершении тестирования.

```
Microsoft Windows [Version 6.1.7601]
(с) Корпорация Майкрософт (Microsoft Corp.), 2009. Все права защищены.
```

```
C:\Users\user>net view /all
```

Имя сервера	Заметки
\\SAMBA	Samba 4.9.5-Debian
Команда выполнена успешно.	
C:\Users\user>net use X: \\samba.account.school134\shared	
Команда выполнена успешно.	
C:\Users\user>dir X:\	
Том в устройстве X имеет метку shared	
Серийный номер тома: 9EDD-DA4F	
Содержимое папки X:\	
09.06.2021 10:12	<DIR> .
18.05.2021 15:11	<DIR> ..
	0 файлов 0 байт
	2 папок 256 781 373 440 байт свободно
C:\Users\user>echo sample file > X:\sample.txt	
C:\Users\user>type X:\sample.txt	
sample file	
C:\Users\user>del X:\sample.txt	
C:\Users\user>dir X:\	
Том в устройстве X имеет метку shared	
Серийный номер тома: 9EDD-DA4F	
Содержимое папки X:\	
09.06.2021 10:39	<DIR> .
18.05.2021 15:11	<DIR> ..
	0 файлов 0 байт
	2 папок 256 781 373 440 байт свободно
C:\Users\user>net use Y: \\samba.account.school134\backup	
Команда выполнена успешно.	
C:\Users\user>dir Y:\	
Том в устройстве Y имеет метку backup	
Серийный номер тома: 2A02-4E9E	
Содержимое папки Y:\	
18.05.2021 15:10	<DIR> .
18.05.2021 15:11	<DIR> ..
	0 файлов 0 байт
	2 папок 256 781 373 440 байт свободно
C:\Users\user>echo sample file > Y:\sample.txt	
C:\Users\user>type Y:\sample.txt	
sample file	
C:\Users\user>del Y:\sample.txt	
C:\Users\user>dir Y:\	
Том в устройстве Y имеет метку backup	
Серийный номер тома: 2A02-4E9E	
Содержимое папки Y:\	
09.06.2021 10:41	<DIR> .
18.05.2021 15:11	<DIR> ..
	0 файлов 0 байт
	2 папок 256 781 373 440 байт свободно
C:\Users\user>net use Z: \\samba.account.school134\archive	
Команда выполнена успешно.	
C:\Users\user>dir Z:\	
Том в устройстве Z имеет метку archive	
Серийный номер тома: D1E3-6E0D	
Содержимое папки Z:\	
09.06.2021 10:43	<DIR> .
18.05.2021 15:11	<DIR> ..
09.06.2021 10:43	24 sample.archive.txt
18.05.2021 15:06	<DIR> lost+found
	1 файлов 24 байт
	3 папок 465 958 961 152 байт свободно
C:\Users\user>type Z:\sample.archive.txt	
sample readonly archive	


```

C:\Users\user>del Z:\sample.archive.txt
Z:\sample.archive.txt
Отказано в доступе.
C:\Users\user>echo sample file > Z:\sample.txt
Отказано в доступе.

```

В ходе работы приведённого терминального сеанса каталоги были подключены как сетевые диски X:\, Y:\, и Z:\. После чего в первых двух из них были успешно созданы, прочитаны и удалены простейшие текстовые файлы. Для последнего сетевого каталога (архива) чтение уже имеющегося в нём файла оказалось успешным, а все остальные операции – нет. Тем самым работоспособность системы подтверждена.

Устойчивость системы к сбоям демонов smbd и nmbd проверяется аналогично другим службам. Для этого выясняются идентификаторы процессов, эти процессы принудительно завершаются, а затем выполняется проверка состояния служб. Все эти действия отражены в следующем фрагменте терминального сеанса (отклики некоторых команд сокращены в неинформативной в применении к контексту части):

```

root@server:/# systemctl status smbd
• smbd.service - Samba SMB Daemon
  Loaded: loaded (/lib/systemd/system/smbd.service; enabled; vendor preset: enabled)
  Drop-In: /etc/systemd/system/smbd.service.d
    └─override.conf
  Active: active (running) since Wed 2021-06-09 13:01:50 +03; 8min ago
  Docs: man:smbd(8)
        man:samba(7)
        man:smb.conf(5)
  Process: 880 ExecStartPre=/usr/share/samba/update-apparmor-samba-profile
(code=exited, status=0/SUCCESS)
  Main PID: 884 (smbd)
  Status: "smbd: ready to serve connections..."
  Tasks: 5 (limit: 2314)
  Memory: 8.1M
  CGroup: /system.slice/smbd.service
    └─884 /usr/sbin/smbd --foreground --no-process-group
    └─886 /usr/sbin/smbd --foreground --no-process-group
    └─887 /usr/sbin/smbd --foreground --no-process-group
    └─888 /usr/sbin/smbd --foreground --no-process-group
    └─890 /usr/sbin/smbd --foreground --no-process-group
root@server:/# kill -9 884
root@server:/# systemctl status smbd
• smbd.service - Samba SMB Daemon
  Loaded: loaded (/lib/systemd/system/smbd.service; enabled; vendor preset: enabled)
  Drop-In: /etc/systemd/system/smbd.service.d
    └─override.conf
  Active: active (running) since Wed 2021-06-09 13:11:14 +03; 3s ago
  Docs: man:smbd(8)
        man:samba(7)
        man:smb.conf(5)
  Process: 904 ExecStartPre=/usr/share/samba/update-apparmor-samba-profile
(code=exited, status=0/SUCCESS)
  Main PID: 908 (smbd)
  Status: "smbd: ready to serve connections..."
  Tasks: 4 (limit: 2314)
  Memory: 7.2M
  CGroup: /system.slice/smbd.service
    └─908 /usr/sbin/smbd --foreground --no-process-group
    └─911 /usr/sbin/smbd --foreground --no-process-group
    └─912 /usr/sbin/smbd --foreground --no-process-group
    └─913 /usr/sbin/smbd --foreground --no-process-group

```

```
root@server:/# systemctl status nmbd
```

```
• nmbd.service - Samba NMB Daemon
  Loaded: loaded (/lib/systemd/system/nmbd.service; enabled; vendor preset: enabled)
  Drop-In: /etc/systemd/system/nmbd.service.d
           └─override.conf
  Active: active (running) since Wed 2021-06-09 13:14:47 +03; 38s ago
  Docs: man:nmbd(8)
        man:samba(7)
        man:smb.conf(5)
  Main PID: 961 (nmbd)
  Status: "nmbd: ready to serve connections..."
  Tasks: 1 (limit: 2314)
  Memory: 2.3M
  CGroup: /system.slice/nmbd.service
          └─961 /usr/sbin/nmbd --foreground --no-process-group
```

```
root@server:/# kill -9 961
```

```
root@server:/# systemctl status nmbd
```

```
• nmbd.service - Samba NMB Daemon
  Loaded: loaded (/lib/systemd/system/nmbd.service; enabled; vendor preset: enabled)
  Drop-In: /etc/systemd/system/nmbd.service.d
           └─override.conf
  Active: active (running) since Wed 2021-06-09 13:15:41 +03; 3s ago
  Docs: man:nmbd(8)
        man:samba(7)
        man:smb.conf(5)
  Main PID: 967 (nmbd)
  Status: "nmbd: ready to serve connections..."
  Tasks: 1 (limit: 2314)
  Memory: 2.3M
  CGroup: /system.slice/nmbd.service
          └─967 /usr/sbin/nmbd --foreground --no-process-group
```

Настройка резервного копирования

Подробно описанная в постановке задачи система резервного копирования реализуется несколькими файлами. Ключевую роль играет сценарий командной оболочки, выполняющий архивирование данных, дополнение каталога-справочника и выключение компьютера в конце рабочего дня. Этот сценарий расположен в каталоге **/usr/local/sbin** и имеет вид:

```
root@server:/# cat /usr/local/sbin/server-shutdown
#!/bin/bash
```

```
OLDPATH=$PATH
PATH=$PATH:/sbin
```

```
SED=$(which sed)
ZIP=$(which zip)
UNZIP=$(which unzip)
```

```
BACKUP_DIR=/srv/samba/backup
ARCHIVE_DIR=/srv/samba/archive
ARCHIVE_INDEX=$ARCHIVE_DIR/index.html
ARCHIVE_INDEX_TEMPLATE=/usr/local/share/index.template
ARCHIVE_NAME=archive_$(date +%Y_%m_%d).zip
ARCHIVE=$ARCHIVE_DIR/$ARCHIVE_NAME
ARCHIVE_REF=file:///samba.account.school34/archive/$ARCHIVE_NAME
```

```
HTML_OPEN=$(cat <<EOF
```

```

        <p>
            <a href=$ARCHIVE_REF><h2>Архив от $(date "+%d %B %Y")</h2></a>
            <details> <summary>Содержимое</summary>
        </pre>
EOF
)
HTML_CLOSE=$(cat <<EOF
    </pre>
    </details>
</p>
</body>
</html>
EOF
)

if [[ -n $SED && -n $ZIP && -n $UNZIP ]];
then
    logger -s -t server_shutdown -p syslog.info "server shutdown begins"
    logger -s -t server_shutdown -p syslog.info "stopping smbd and networking"
    systemctl stop smbd
    systemctl stop networking
    NEEDED_BLOCKS=$(du --total $BACKUP_DIR/* | tail -1 | sed s/"[:space:]].*"//g)
    NEEDED_INODES=1
    AVAIL_BLOCKS=$(df --output=avail $ARCHIVE_DIR | tail -1 | sed s/" " //g)
    AVAIL_INODES=$(df --output=iavail $ARCHIVE_DIR | tail -1 | sed s/" " //g)
    logger -s -t server_shutdown -p syslog.info "it is needed up to $NEEDED_BLOCKS
        blocks and $NEEDED_INODES inodes, whereas $AVAIL_BLOCKS
        S blocks and $AVAIL_INODES inodes available"
    if [[ $AVAIL_BLOCKS -gt $NEEDED_BLOCKS && $AVAIL_INODES -gt $NEEDED_INODES ]];
    then
        if [[ -n $(ls $BACKUP_DIR) ]];
        then
            cd $BACKUP_DIR
            zip -mqT $ARCHIVE -r ./
            if [ -z $(ls $BACKUP_DIR) ];
            then
                if [ ! -f /srv/samba/archive/index.html ];
                then
                    logger -s -t server_shutdown -p syslog.info "archive index does
                        not exist, new one created"
                    cat $ARCHIVE_INDEX_TEMPLATE > $ARCHIVE_INDEX
                    chown buh:buh $ARCHIVE_INDEX
                    chmod 0744 $ARCHIVE_INDEX
                fi
                sed s/"<\body.*>"//g -i $ARCHIVE_INDEX
                sed s/"<\html.*>"//g -i $ARCHIVE_INDEX
                echo $HTML_OPEN >> $ARCHIVE_INDEX
                unzip -l $ARCHIVE >> $ARCHIVE_INDEX
                echo $HTML_CLOSE >> $ARCHIVE_INDEX
                logger -s -t server_shutdown -p syslog.info "archive created, archive
                    index updated"
            else
                logger -s -t server_shutdown -p syslog.warning "failed to create
                    archive, all data remain in backup directory"
            fi
        else
            logger -s -t server_shutdown -p syslog.info "backup directory is empty,
                nothing to move to archive"
        fi
    else
        logger -s -t server_shutdown -p syslog.warning "available blocks or inodes
            are not enough, can't move data to archive"
    fi
fi

```

```

    fi
else
    logger -s -t server_shutdown -p syslog.warning "sed, zip or unzip missing, can't
        move data to archive"
fi

PATH=$OLDPATH
/sbin/shutdown -P now

```

Для обеспечения работоспособности этого сценария также потребуется файл-шаблон html-каталога архива. Он расположен в каталоге **/usr/local/share** и имеет вид:

```

root@server:/# cat /usr/local/share/index.template
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.0 Transitional//EN">
<html>
<head>
    <meta http-equiv="content-type" content="text/html; charset=utf-8"/>
    <title>Архив бухгалтерии МБОУ СОШ №34</title>
</head>
<style>
body {
    margin-left: 0px;
    margin-top: 0px;
    margin-right: 0px;
    margin-bottom: 0px;
}
a {
    text-decoration: none;
    color: #1C61B5;
}
a:link {
    text-decoration: none;
}
a:hover {
    text-decoration: none;
}
a:active {
    color: #000000;
    text-decoration: underline;
}
h1 {
    color: #0a0afa;
    font-size: 28px;
    margin: 0px;
}
h2 {
    font-size: 20px;
    margin-left: 30px;
    margin-bottom: 10px;
}
p {
    font-size: 14px;
    margin-left: 30px;
    margin-top: 10px;
    margin-right: 30px;
    margin-bottom: 10px;
}
details {
    margin-left: 30px;
    margin-top: 10px;
    margin-right: 30px;
    margin-bottom: 10px;
}

```

```

}
pre {
    font-size: 12px;
    margin-left: 30px;
    margin-top: 10px;
    margin-right: 30px;
    margin-bottom: 10px;
}
</style>
</head>
<body leftmargin="0" topmargin="0" marginwidth="0" marginheight="0">
    <h1 align="center">Архив бухгалтерии МБОУ СОШ №34</h1>
</body>
</html>

```

В этом файле целиком содержится вся html-страница, на основе которой будет формироваться каталог архива, вместе с таблицами стилей, таким образом, никаких внешних зависимостей у этой страницы нет, а значит сделать её доступной по сети без потери функциональности можно средствами файл-сервера без необходимости разворачивать в системе веб-сервер.

Сам сценарий при своём запуске проверяет наличие необходимых утилит, выясняет доступное и необходимое дисковое пространство, необходимое для выполнения архивирования, при отсутствии файла-каталога создаёт его на основе указанного шаблона и останавливает сетевую службу и службу сетевых каталогов. Затем сценарий выполняет создание архива (если каталог backup не пуст) и при успешном завершении этой операции очищает каталог архивирования и дополняет файл-каталог. Если же каталог архивирования пуст, то создание архива не происходит, индексный файл не меняется. В завершение происходит выключение всей системы. Действия сценария отражаются в системном журнале, что продемонстрировано в следующей выдержке:

```
root@server:/# cat /var/log/messages | grep server_shutdown
```

```

...
Jun 11 15:07:46 server server_shutdown: server shutdown begins
Jun 11 15:07:46 server server_shutdown: stopping smbd and networking
Jun 11 15:07:46 server server_shutdown: it is needed up to 760 blocks and 1 inodes,
whereas 455038044 blocks and 30523379 inodes available
Jun 11 15:07:46 server server_shutdown: archive created, archive index updated
...

```

Что касается создания индексного файла архива, то это может понадобиться в двух случаях: первый запуск сервера и полная очистка архива, например при его заполнении и переносе содержимого на съёмные носители, например, оптические диски.

Запуск сценария следует возложить на службу cron, реализация этого описана в разделе «Автоматическое выключение системы».

О дополнительных службах

AppArmor/SELinux

Использование систем мандатного разграничения прав доступа в рассматриваемом сервере представляется избыточным, поскольку основным сервисом, предоставляемым системой, является файловое хранилище, а связанные с ним процессы выполняются от

имени root. При этом они взаимодействуют только с внутренней сетью, клиенты которой равноправны, а доступ из внешней сети к ним закрыт. Вспомогательные процессы работают от имени собственных системных пользователей и их доступ к каталогам, содержащим пользовательские данные, ограничен дискреционно.

Защита от подбора паролей (fail2ban)

Поскольку в силу характера функционирования сервера единственным механизмом удалённого входа в систему является ssh, то от использования fail2ban вполне безопасно отказаться. Система хорошо защищена файерволом (описано далее) с указанием (как по ip, так и по mac-адресу) для него единственного хоста, с которого разрешен вход по ssh. Сам ssh-сервер системы имеет такие же ограничения, кроме того предприняты меры по защите от арг-атак (статическая запись), сам удалённый вход разрешен только для непривилегированного пользователя, учётная запись которого защищена паролем, а повышение привилегий требует знания пароля суперпользователя.

Что же касается службы обмена файлами, то доступ к ней возможен только с внутреннего сетевого интерфейса. С учётом физического размещения и сервера, и клиентских устройств, и сетевых коммутаторов и кабелей подсети бухгалтерии внутри одного помещения можно утверждать, что защита этого сервиса средствами fail2ban избыточна.

Ротация журналов (logrotate)

Системные журналы могут служить средством атаки на сервер с целью заполнить его дисковое пространство и парализовать, тем самым, его работу. Этому противостоит система ротации журнальных файлов.

При входе в систему по протоколу ssh соответствующая запись вносится в файл `/var/log/auth.log`. Учитывая предпринятые меры защиты ssh, можно утверждать, что атака возможна только с единственного не менее защищённого хоста из ядра сети (компьютер системного администратора).

Остальные доступные клиентам службы (dnsmasq, samba и ntp) не ведут собственных журналов клиентских обращений при указанной для них конфигурации и, тем самым, не могут быть направлением для такой атаки.

Как итог следует отметить, что настройки системы по умолчанию (ротация еженедельно) здесь вполне применимы.

Служба точного времени (ntp)

Сервер службы точного времени не только предоставляет услуги клиентам, но и следит за точностью системных часов машины, на которой развёрнут. Поэтому невозможна и бессмысленна работа других средств синхронизации часов в описываемой системе. В частности, простейший ntp-клиент, который входит в состав systemd, оказывается после установки и настройки ntpd в неработоспособном состоянии и должен быть отключён:

```
root@server:/# systemctl status systemd-timesyncd
• systemd-timesyncd.service - Network Time Synchronization
  Loaded: loaded (/lib/systemd/system/systemd-timesyncd.service; enabled; vendor
  preset: enabled)
  Drop-In: /usr/lib/systemd/system/systemd-timesyncd.service.d
           └─disable-with-time-daemon.conf
```

```

Active: inactive (dead)
Docs: man:systemd-timesyncd.service(8)
root@server:/# systemctl list-unit-files | grep timesyncd
systemd-timesyncd.service                                enabled
root@server:/# systemctl disable systemd-timesyncd
Removed /etc/systemd/system/dbus-org.freedesktop.timesync1.service.
Removed /etc/systemd/system/sysinit.target.wants/systemd-timesyncd.service.
root@server:/# systemctl list-unit-files | grep timesync
systemd-timesyncd.service                                disabled

```

Почтовая служба

Почтовая служба, даже локальная, может стать направлением для атаки: так, например, злоумышленник может генерировать огромный вал писем локально в системе, чем провоцировать переполнение файловой системы и отказ обслуживания. Поэтому даже локальную почтовую службу следует тщательно настраивать или отключать вовсе, если она не используется.

В рассматриваемой системе вместе с другими компонентами установлен агент пересылки почты (MTA) `exim4`. В перспективе планируется введение в эксплуатацию внутреннего почтового сервера организации, что позволит многим системным службам (например, системе мониторинга состояния жёсткого диска) уведомлять системного администратора о проблемах.

С точки зрения безопасности MTA может быть использован для переполнения почтовых ящиков пользователей, т. е. заполнения дискового пространства машины.

На данном этапе имеет смысл полностью удалить MTA из системы, тем более что это не ведет к удалению других компонентов системы. Итак, следующий сокращённый в незначительном выводе фрагмент терминального сеанса демонстрирует процедуру проверки возможности удаления и непосредственно удаления MTA:

```

root@server:/# dpkg -l | grep exim
ii  exim4-base      4.92-8+deb10u5  amd64  support files for all Exim MTA (v4) packages
ii  exim4-config    4.92-8+deb10u5  all     configuration for the Exim MTA (v4)
ii  exim4-daemon-light 4.92-8+deb10u5  amd64  lightweight Exim MTA (v4) daemon
root@server:/# apt-get -s purge exim4-daemon-light
Чтение списков пакетов... Готово
Построение дерева зависимостей
Чтение информации о состоянии... Готово
Следующие пакеты устанавливались автоматически и больше не требуются:
  exim4-base exim4-config libevent-2.1-6 libgnutls-dane0 libunbound8
Для их удаления используйте «apt autoremove».
Следующие пакеты будут УДАЛЕНЫ:
  exim4-daemon-light*
Обновлено 0 пакетов, установлено 0 новых пакетов, для удаления отмечено 1 пакетов, и
12 пакетов не обновлено.
Purge exim4-daemon-light [4.92-8+deb10u5]
root@server:/# apt-get purge exim4-daemon-light
Чтение списков пакетов... Готово
Построение дерева зависимостей
Чтение информации о состоянии... Готово
Следующие пакеты устанавливались автоматически и больше не требуются:
  exim4-base exim4-config libevent-2.1-6 libgnutls-dane0 libunbound8
Для их удаления используйте «apt autoremove».
Следующие пакеты будут УДАЛЕНЫ:
  exim4-daemon-light*
...

```

Вычищаются файлы настройки пакета `exim4-daemon-light` (4.92-8+deb10u5) ...

```
root@server:/# apt autoremove
```

Чтение списков пакетов... Готово

Построение дерева зависимостей

Чтение информации о состоянии... Готово

Следующие пакеты будут УДАЛЕНЫ:

`exim4-base` `exim4-config` `libevent-2.1-6` `libgnutls-dane0` `libunbound8`

...

Так как некоторые системные службы используют в своей работе почтовую систему, следует убедиться, что их работоспособность не нарушена. Так после успешной перезагрузки системы (в смысле отсутствия сообщений об ошибках запуска служб) следует проверить их состояние. Ниже приведён сокращённый отклик только одной такой команды проверки для краткости изложения, поскольку остальные команды дали аналогичный результат:

```
root@server:/home/administrator# systemctl status smartd
```

```
• smartd.service - Self Monitoring and Reporting Technology (SMART) Daemon
   Loaded: loaded (/lib/systemd/system/smartd.service; enabled; vendor preset:
   enabled)
```

```
   Active: active (running) since Thu 2021-06-03 10:00:15 +03; 2min 57s ago
```

...

```
root@server:/home/administrator# systemctl status networking
```

...

```
root@server:/home/administrator# systemctl status mdmonitor
```

...

```
root@server:/home/administrator# systemctl status nut-monitor
```

...

```
root@server:/home/administrator# systemctl status nut-server
```

...

```
root@server:/home/administrator# systemctl status nut-driver
```

...

```
root@server:/home/administrator# systemctl status ssh
```

...

```
root@server:/home/administrator# systemctl status ntp
```

...

```
root@server:/home/administrator# systemctl status cron
```

...

```
root@server:/home/administrator# systemctl status dnsmasq
```

...

```
root@server:/home/administrator# systemctl status smbd
```

...

В перспективе после развёртывания внутреннего почтового сервера организации МТА вновь может быть установлен в системе и настроен соответствующим образом. Тем временем при его отсутствии демон `smartd` при необходимости будет создавать файлы электронных писем в каталоге `/tmp`.

Автоматическое выключение системы

Автоматическое выключение системы в нерабочее время также выполняется средствами демона `cron`, для чего в его конфигурационный файл `/etc/crontab` внесены строки:

```
#Shutting down at night with backup archive creation
```

```
00 21 * * * root /usr/local/sbin/server-shutdown
```

Они обеспечивают выключение системы в 21:00 с уведомлением пользователей за 10 минут до этого. Очевидно, что выключение системы на ночь преследует три цели:

- энергосбережение;
- экономия ресурса аппаратной платформы;
- противодействие длительным атакам на подбор пароля.

Установка и настройка сетевого фильтра (nftables)

Установка сетевого фильтра

Для управления правилами файервола в современных версиях Debian рекомендовано использовать пакет nftables, который потребуется установить командой

```
root@server:/# apt-get install nftables
```

Сетевое окружение и потенциальные угрозы

Составление свода правил сетевого фильтра следует начинать с выяснения всех протоколов и портов, используемых системой и клиентами. Список сетевых портов и протоколов, на которых система готова принимать соединения, можно получить следующим образом:

```
root@server:/tmp# ss -tuln
```

Netid	State	Recv-Q	Send-Q	Local Address:Port	Peer Address:Port
udp	UNCONN	0	0	0.0.0.0:67	0.0.0.0:*
udp	UNCONN	0	0	10.9.1.129:123	0.0.0.0:*
udp	UNCONN	0	0	192.168.0.88:123	0.0.0.0:*
udp	UNCONN	0	0	127.0.0.1:123	0.0.0.0:*
udp	UNCONN	0	0	0.0.0.0:123	0.0.0.0:*
udp	UNCONN	0	0	10.9.1.191:137	0.0.0.0:*
udp	UNCONN	0	0	10.9.1.129:137	0.0.0.0:*
udp	UNCONN	0	0	0.0.0.0:137	0.0.0.0:*
udp	UNCONN	0	0	10.9.1.191:138	0.0.0.0:*
udp	UNCONN	0	0	10.9.1.129:138	0.0.0.0:*
udp	UNCONN	0	0	0.0.0.0:138	0.0.0.0:*
udp	UNCONN	0	0	0.0.0.0:46417	0.0.0.0:*
udp	UNCONN	0	0	127.0.0.1:53	0.0.0.0:*
udp	UNCONN	0	0	10.9.1.129:53	0.0.0.0:*
tcp	LISTEN	0	50	10.9.1.129:445	0.0.0.0:*
tcp	LISTEN	0	16	127.0.0.1:3493	0.0.0.0:*
tcp	LISTEN	0	50	10.9.1.129:139	0.0.0.0:*
tcp	LISTEN	0	32	127.0.0.1:53	0.0.0.0:*
tcp	LISTEN	0	32	10.9.1.129:53	0.0.0.0:*
tcp	LISTEN	0	128	192.168.0.88:22	0.0.0.0:*

Следует сделать ряд замечаний:

Во-первых, следует закрыть сетевым фильтром порт 67, работающий по протоколу UDP, для всех входящих через внешний интерфейс пакетов. Этот порт используется протоколом DHCP, который, согласно постановке задачи, должен обслуживать лишь клиентов внутренней сети. Этот порт открыт демоном dnsmasq, который также выполняет такую проверку, однако дополнительная защита файерволом лишней явно не будет.

Во-вторых, требуется для внешних интерфейсов закрыть порты 137 (netbios-ns) и 138 (netbios-dgm) для протокола udp. Доступным этот порт должен остаться лишь для локального петлевого и внутреннего интерфейсов. Эти порты используются для

обеспечения работы механизмов сетевого обнаружения Windows-клиентами, которым не следует знать структуру сети организации кроме собственной подсети.

В-третьих, не лишним будет закрыть для новых соединений по протоколу точного времени (порт 123) внешние интерфейсы, не полагаясь только на устойчивость самого демона `ntpd`. При этом способность самого демона инициировать соединения с вышестоящими серверами точного времени не пострадает.

В-четвертых, для снижения вероятности атак на отказ в обслуживании, следует ограничить максимальное количество пакетов в единицу времени для каждой службы.

Наконец, следует обеспечить максимальную защиту критической для безопасности системы службы `ssh`, а именно, допускать установление соединений с ней только с проверкой `ip`-адреса, аппаратного адреса, интерфейса, с которого пришёл пакет и с соблюдением ограничения на количество таких пакетов в единицу времени.

Отдельно следует рассмотреть вопрос о транзите пакетов. Во избежание попыток сканирования структуры всей сети организации с потенциально уязвимых клиентских компьютеров следует оставить возможность транзита сетевых пакетов только к серверу АИС «Аверс» (по `tcp`-порту 8082), к виртуальной подсети серверов организации (без ограничений по портам на данном этапе) и в Интернет.

При этом следует понимать, что под Интернетом понимается вообще любой сетевой адрес, кроме выше обозначенных, в том числе и адреса из других локальных подсетей, о которых серверу подсети бухгалтерии вообще ничего не известно. Казалось бы, что злоумышленник, зная структуру сети организации, сможет отправить пакет на один из открытых портов к некоторому компьютеру, например, административной подсети. Однако, в силу того, что сервер подсети бухгалтерии не имеет маршрута в эту подсеть в своей таблице маршрутизации, а также в силу того, что прямое направление пакета к серверу административной подсети через его адрес в ядре сети невозможно, единственным маршрутом останется главный шлюз, `файервол` которого пресечёт отправку опасного пакета. Кроме того, межсетевое экранирование организовано и на сервере атакуемой подсети.

Также следует обратить внимание на порт 46417, открытый службой доменных имён для общения с вышестоящими серверами. Подобное соединение не имеет фиксированного порта и при каждом запуске службы может и, скорее всего, будет меняться.

Остальные перечисленные порты открыты только на допустимых адресах и настройки `файервола` должны их лишь подтверждать.

Кроме того, в сетевом фильтре должны быть предусмотрены также и правила, обеспечивающие работу `GRE`-туннелей.

Наконец, на данном этапе по организационным причинам нет возможности составить точный список внешних ресурсов за пределами сети организации, к которым разрешено обращаться с клиентских компьютеров, точно так же как и соответствующие списки портов и протоколов, поэтому для подобных соединений принята доверительная политика по умолчанию.

Структура сетевого фильтра

Опираясь на вышесказанное и руководствуясь принципом запрета по умолчанию, можно так охарактеризовать структуру сетевого фильтра:

- связь с внешними сетями осуществляется с помощью механизма NAT,
- определены таблицы фильтрации для протоколов IPv4 и IPv6,
- в таблице фильтрации определены цепочки правил для входящих, исходящих и транзитных пакетов,
- для всех цепочек таблицы IPv6 задана политика сброса пакетов по умолчанию,
- для входящей и транзитной цепочек таблицы IPv4 определена политика сброса пакетов по умолчанию, для исходящей цепочки по умолчанию установлена доверительная политика,
- разрешены приём и транзит пакетов, относящихся к уже установленным соединениям,
- наконец внесены разрешающие правила для описанного выше трафика.

В итоге сценарий атомарной загрузки правил приобрёл вид:

```
root@server:/# cat /etc/nftables.conf
#!/usr/sbin/nft -f
```

```
flush ruleset
```

```
define IFACE_KERNEL = enp2s0
define IFACE_ACCD = enp3s0
define IFACE_USB = ethusb
define IFACE_EXT = { $IFACE_KERNEL, $IFACE_USB }
```

```
#FIXME: remove entries from CLIENTS and CLIENT_LLADDRS for
#dhcp-host=54:be:f7:2a:69:05,test,10.9.1.136
```

```
define CLIENT_LLADDRS = { 90:2b:34:a3:83:44, 50:e5:49:33:de:d9, 90:2b:34:a0:24:7b,
90:2b:34:96:16:53 }
define MFP_LLADDR = f8:0d:ac:78:8a:f1
define ACCD_LLADDRS = { $CLIENT_LLADDRS, $MFP_LLADDR }
define ADMIN_LLADDR = 90:2b:34:48:08:b5
```

```
define CLIENTS = { 10.9.1.131, 10.9.1.132, 10.9.1.133, 10.9.1.135, }
define MFP = 10.9.1.134
define ACCD_HOSTS = { $CLIENTS, $MFP }
define KRISTA_CLIENT = 10.9.1.133
define KRISTA_SERVER = 213.222.245.118
define ADMIN_HOST = 192.168.0.2
define AVERS_HOST = 192.168.0.4
define SERVICE_NET = 192.168.7.0/24
define GATEWAY = 192.168.0.1
```

```
define SSH_PORT = 22
define NTP_PORT = 123
define DHCP_PORT = 67
define DNS_PORT = 53
define SAMBA_PORTS = { 137, 138, 139, 445 }
define KRISTA_PORT = 1723
define AVERS_PORT = 8082
```

```
table ip nat {
    chain prerouting {
        type nat hook prerouting priority 0;
```

```

    policy accept;
    iifname $IFACE_EXT tcp dport $KRISTA_PORT dnat $KRISTA_CLIENT:$KRISTA_PORT
}
chain postrouting {
    type nat hook postrouting priority 0;
    policy accept;
    oifname $IFACE_EXT masquerade
}
}

table ip filter {
    chain input {
        type filter hook input priority 0;
        policy drop;
        iif lo accept
        ct state established,related accept
        iifname $IFACE_EXT icmp type echo-request limit rate 10/second accept
        iif $IFACE_ACCD ether saddr $CLIENT_LLADDRS ip saddr $CLIENTS icmp
type echo-request limit rate 50/second accept
        iif $IFACE_ACCD ether saddr $ACCD_LLADDRS ip saddr $ACCD_HOSTS udp
dport $NTP_PORT limit rate 300/second accept
        iif $IFACE_ACCD ether saddr $ACCD_LLADDRS udp
dport $DHCP_PORT limit rate 300/second accept
        iif $IFACE_ACCD ether saddr $CLIENT_LLADDRS ip saddr $CLIENTS udp
dport $DNS_PORT limit rate 300/second accept
        iif $IFACE_ACCD ether saddr $CLIENT_LLADDRS ip saddr $CLIENTS tcp
dport $DNS_PORT limit rate 300/second accept
        iif $IFACE_ACCD ether saddr $CLIENT_LLADDRS ip saddr $CLIENTS udp
dport $SAMBA_PORTS limit rate 200 mbytes/second accept
        iif $IFACE_ACCD ether saddr $CLIENT_LLADDRS ip saddr $CLIENTS tcp
dport $SAMBA_PORTS limit rate 200 mbytes/second accept
        iif $IFACE_KERNEL ether saddr $ADMIN_LLADDR ip saddr $ADMIN_HOST tcp
dport $SSH_PORT ct state new limit rate 1/second accept
    }
    chain forward {
        type filter hook forward priority 0;
        policy drop;
        ct state established,related accept
        ip saddr $CLIENTS oif $IFACE_KERNEL ip daddr $SERVICE_NET accept
        ip saddr $CLIENTS oif $IFACE_KERNEL ip daddr $AVERS_HOST tcp dport
$AVERS_PORT accept
        ip saddr $CLIENTS oif $IFACE_KERNEL rt nexthop $GATEWAY accept
        ip saddr $CLIENTS oifname $IFACE_USB accept
        ip saddr $KRISTA_CLIENT tcp sport $KRISTA_PORT ip daddr $KRISTA_SERVER tcp
dport $KRISTA_PORT accept
        ip saddr $KRISTA_SERVER tcp sport $KRISTA_PORT ip daddr $KRISTA_CLIENT tcp
dport $KRISTA_PORT accept
    }
    chain output {
        type filter hook output priority 0;
        policy accept;
    }
}

table ip6 filter {
    chain input {
        type filter hook input priority 0;
        policy drop;
    }
    chain forward {
        type filter hook forward priority 0;
        policy drop;
    }
}

```

```

}
chain output {
    type filter hook output priority 0;
    policy drop;
}
}

```

Транзит пакетов и поддержка туннелей

Прежде всего следует обеспечить транзит сетевых пакетов между интерфейсами рассматриваемого сервера. Для этого следует привести конфигурационный файл **/etc/sysctl.conf** к такому виду, чтобы в нём содержалась строка

```
net.ipv4.ip_forward=1
```

Обычно для этого достаточно её просто раскомментировать. Для включения транзита пакетов в текущей сессии без перезагрузки системы достаточно выполнить команду:

```
root@server:/# echo 1 > /proc/sys/net/ipv4/ip_forward
```

Также потребуется дописать в файл **/etc/sysctl.conf** следующие строки:

```
#Enabling netfilter conntrack helper
net.netfilter.nf_conntrack_helper=1
```

Тем самым подключается механизм отслеживания соединений, необходимый для реализации GRE-туннеля. Для текущей сессии (без перезагрузки) этот механизм может быть включён командой вида:

```
root@server:/# echo 1 > /proc/sys/net/netfilter/nf_conntrack_helper
```

Запуск и проверка сетевого фильтра

Автоматическая загрузка правил файервола обеспечена установщиком пакета **nftables** и не требует ручного вмешательства. Проверить текущий набор правил можно командой

```
root@debian:/# /sbin/nft list ruleset
```

Вывод этой команды с точностью до комментариев, инструкций предварительной очистки и подстановки списка разрешённых портов должен совпадать с приведённым выше сценарием загрузки правил файервола.

Убедиться в том, что правила были загружены до запуска сетевых интерфейсов, можно просмотрев содержимое файла **/var/log/daemon.log**, выдержка из которого с некоторыми сокращениями, но с сохранением порядка следования представленных, имеет вид:

```

Dec 10 14:05:41 server systemd[1]: Started udev Kernel Device Manager.
Dec 10 14:05:41 server systemd[1]: Started nftables.
Dec 10 14:05:41 server systemd[1]: Reached target Network (Pre).
...
Dec 10 14:05:41 server systemd[1]: Starting Raise network interfaces...
...
Dec 10 14:05:42 server systemd[1]: Started Raise network interfaces.
Dec 10 14:05:42 server systemd[1]: Reached target Network.
...
Dec 10 14:05:42 server systemd[1]: Reached target Network is Online.

```

Также посмотреть состояние службы можно командой:

```
root@server:/# systemctl status nftables
```

```
• nftables.service - nftables
  Loaded: loaded (/lib/systemd/system/nftables.service; enabled; vendor preset:
enabled)
  Active: active (exited) since Fri 2021-12-10 14:07:20 +03; 10s ago
    Docs: man:nft(8)
          http://wiki.nftables.org
  Process: 561 ExecStart=/usr/sbin/nft -f /etc/nftables.conf (code=exited,
status=0/SUCCESS)
 Main PID: 561 (code=exited, status=0/SUCCESS)
```

```
дек 10 14:07:20 server systemd[1]: Starting nftables...
```

```
дек 10 14:07:20 server systemd[1]: Started nftables.
```

Проверка работоспособности межсетевого экрана может быть разделена на тривиальную проверку доступности клиенту разрешённых сервисов, в том числе и расположенных вне подсети бухгалтерии, и проверку сетевым сканером nmap доступности тех или иных портов сервера с четырёх разных направлений:

- с компьютера системного администратора
- с произвольного компьютера из ядра сети
- с клиентского компьютера из подсети бухгалтерии
- с компьютера-нарушителя, искусственно введённого в подсеть бухгалтерии

Тестирование автоматизировано при помощи сценария /tmp/nmap.sh, содержимое которого прозрачно угадывается из его вывода, т. к. выполняемые команды отображаются с префиксом user@host.

Протокол сканирования с компьютера системного администратора:

```
root@admin:/tmp# chmod a+x nmap.sh
```

```
root@admin:/tmp# ./nmap.sh
```

Пинг-сканирование

```
user@host:/# nmap -sP 192.168.0.88
```

```
Starting Nmap 7.70 ( https://nmap.org ) at 2021-12-13 14:31 MSK
Nmap scan report for server.account.school34 (192.168.0.88)
Host is up (0.00021s latency).
MAC Address: 18:D6:C7:00:EA:6C (Tp-link Technologies)
Nmap done: 1 IP address (1 host up) scanned in 0.34 seconds
```

TCP SYN пинг

```
user@host:/# nmap -PS 192.168.0.88
```

```
Starting Nmap 7.70 ( https://nmap.org ) at 2021-12-13 14:31 MSK
Nmap scan report for server.account.school34 (192.168.0.88)
Host is up (0.00020s latency).
Not shown: 999 filtered ports
PORT      STATE SERVICE
22/tcp    open  ssh
MAC Address: 18:D6:C7:00:EA:6C (Tp-link Technologies)
```

```
Nmap done: 1 IP address (1 host up) scanned in 7.75 seconds
```

TCP ACK пинг

```
user@host:/# nmap -PA 192.168.0.88
```

```
Starting Nmap 7.70 ( https://nmap.org ) at 2021-12-13 14:31 MSK
Nmap scan report for server.account.school34 (192.168.0.88)
Host is up (0.00021s latency).
Not shown: 999 filtered ports
PORT      STATE SERVICE
22/tcp    open  ssh
```

MAC Address: 18:D6:C7:00:EA:6C (Tp-link Technologies)

Nmap done: 1 IP address (1 host up) scanned in 12.51 seconds

UDP пинг

user@host:/# nmap -PU 192.168.0.88

Starting Nmap 7.70 (<https://nmap.org>) at 2021-12-13 14:32 MSK

Nmap scan report for server.account.school34 (192.168.0.88)

Host is up (0.00019s latency).

Not shown: 999 filtered ports

PORT STATE SERVICE

22/tcp open ssh

MAC Address: 18:D6:C7:00:EA:6C (Tp-link Technologies)

Nmap done: 1 IP address (1 host up) scanned in 8.34 seconds

Различные типы пинг-пакетов ICMP

user@host:/# nmap -PE 192.168.0.88

Starting Nmap 7.70 (<https://nmap.org>) at 2021-12-13 14:32 MSK

Nmap scan report for server.account.school34 (192.168.0.88)

Host is up (0.00020s latency).

Not shown: 999 filtered ports

PORT STATE SERVICE

22/tcp open ssh

MAC Address: 18:D6:C7:00:EA:6C (Tp-link Technologies)

Nmap done: 1 IP address (1 host up) scanned in 7.74 seconds

user@host:/# nmap -PP 192.168.0.88

Starting Nmap 7.70 (<https://nmap.org>) at 2021-12-13 14:32 MSK

Nmap scan report for server.account.school34 (192.168.0.88)

Host is up (0.0011s latency).

Not shown: 999 filtered ports

PORT STATE SERVICE

22/tcp open ssh

MAC Address: 18:D6:C7:00:EA:6C (Tp-link Technologies)

Nmap done: 1 IP address (1 host up) scanned in 12.51 seconds

user@host:/# nmap -PM 192.168.0.88

Starting Nmap 7.70 (<https://nmap.org>) at 2021-12-13 14:32 MSK

Nmap scan report for server.account.school34 (192.168.0.88)

Host is up (0.00020s latency).

Not shown: 999 filtered ports

PORT STATE SERVICE

22/tcp open ssh

MAC Address: 18:D6:C7:00:EA:6C (Tp-link Technologies)

Nmap done: 1 IP address (1 host up) scanned in 12.32 seconds

Пинг с использованием протокола IP

user@host:/# nmap -PO 192.168.0.88

Starting Nmap 7.70 (<https://nmap.org>) at 2021-12-13 14:32 MSK

Nmap scan report for server.account.school34 (192.168.0.88)

Host is up (0.00019s latency).

Not shown: 999 filtered ports

PORT STATE SERVICE

22/tcp open ssh

MAC Address: 18:D6:C7:00:EA:6C (Tp-link Technologies)

Nmap done: 1 IP address (1 host up) scanned in 10.37 seconds

ARP пинг

user@host:/# nmap -PR 192.168.0.88

Starting Nmap 7.70 (<https://nmap.org>) at 2021-12-13 14:32 MSK

Nmap scan report for server.account.school34 (192.168.0.88)

Host is up (0.00020s latency).

Not shown: 999 filtered ports

PORT STATE SERVICE
22/tcp open ssh
MAC Address: 18:D6:C7:00:EA:6C (Tp-link Technologies)

Nmap done: 1 IP address (1 host up) scanned in 12.72 seconds

TCP connect сканирование

user@host:/# nmap -sT 192.168.0.88

Starting Nmap 7.70 (<https://nmap.org>) at 2021-12-13 14:33 MSK

Nmap scan report for server.account.school34 (192.168.0.88)

Host is up (0.00023s latency).

Not shown: 999 filtered ports

PORT STATE SERVICE

22/tcp open ssh

MAC Address: 18:D6:C7:00:EA:6C (Tp-link Technologies)

Nmap done: 1 IP address (1 host up) scanned in 4.68 seconds

UDP сканирование

user@host:/# nmap -sU 192.168.0.88

Starting Nmap 7.70 (<https://nmap.org>) at 2021-12-13 14:33 MSK

Nmap scan report for server.account.school34 (192.168.0.88)

Host is up (0.00017s latency).

All 1000 scanned ports on server.account.school34 (192.168.0.88) are open|filtered

MAC Address: 18:D6:C7:00:EA:6C (Tp-link Technologies)

Nmap done: 1 IP address (1 host up) scanned in 21.46 seconds

TCP NULL сканирование

user@host:/# nmap -sN 192.168.0.88

Starting Nmap 7.70 (<https://nmap.org>) at 2021-12-13 14:33 MSK

Nmap scan report for server.account.school34 (192.168.0.88)

Host is up (0.00019s latency).

All 1000 scanned ports on server.account.school34 (192.168.0.88) are open|filtered

MAC Address: 18:D6:C7:00:EA:6C (Tp-link Technologies)

Nmap done: 1 IP address (1 host up) scanned in 21.45 seconds

TCP FIN сканирование

user@host:/# nmap -sF 192.168.0.88

Starting Nmap 7.70 (<https://nmap.org>) at 2021-12-13 14:33 MSK

Nmap scan report for server.account.school34 (192.168.0.88)

Host is up (0.00019s latency).

All 1000 scanned ports on server.account.school34 (192.168.0.88) are open|filtered

MAC Address: 18:D6:C7:00:EA:6C (Tp-link Technologies)

Nmap done: 1 IP address (1 host up) scanned in 21.44 seconds

TCP Xmas сканирование

user@host:/# nmap -sX 192.168.0.88

Starting Nmap 7.70 (<https://nmap.org>) at 2021-12-13 14:34 MSK

Nmap scan report for server.account.school34 (192.168.0.88)

Host is up (0.00017s latency).

All 1000 scanned ports on server.account.school34 (192.168.0.88) are open|filtered

MAC Address: 18:D6:C7:00:EA:6C (Tp-link Technologies)

Nmap done: 1 IP address (1 host up) scanned in 21.44 seconds

TCP ACK сканирование

user@host:/# nmap -sA 192.168.0.88

Starting Nmap 7.70 (<https://nmap.org>) at 2021-12-13 14:34 MSK

Nmap scan report for server.account.school34 (192.168.0.88)

Host is up (0.00019s latency).

Not shown: 999 filtered ports

PORT STATE SERVICE

22/tcp unfiltered ssh

MAC Address: 18:D6:C7:00:EA:6C (Tp-link Technologies)

Nmap done: 1 IP address (1 host up) scanned in 12.23 seconds

TCP Window сканирование

user@host:/# nmap -sW 192.168.0.88

Starting Nmap 7.70 (<https://nmap.org>) at 2021-12-13 14:34 MSK

Nmap scan report for server.account.school34 (192.168.0.88)

Host is up (0.00020s latency).

Not shown: 999 filtered ports

PORT STATE SERVICE

22/tcp closed ssh

MAC Address: 18:D6:C7:00:EA:6C (Tp-link Technologies)

Nmap done: 1 IP address (1 host up) scanned in 12.73 seconds

TCP сканирование Мэймона

user@host:/# nmap -sM 192.168.0.88

Starting Nmap 7.70 (<https://nmap.org>) at 2021-12-13 14:35 MSK

Nmap scan report for server.account.school34 (192.168.0.88)

Host is up (0.00018s latency).

All 1000 scanned ports on server.account.school34 (192.168.0.88) are open|filtered

MAC Address: 18:D6:C7:00:EA:6C (Tp-link Technologies)

Nmap done: 1 IP address (1 host up) scanned in 21.45 seconds

Сканирование протокола IP

user@host:/# nmap -sO 192.168.0.88

Starting Nmap 7.70 (<https://nmap.org>) at 2021-12-13 14:35 MSK

Nmap scan report for server.account.school34 (192.168.0.88)

Host is up (0.00015s latency).

Not shown: 255 open|filtered protocols

PROTOCOL STATE SERVICE

1 open icmp

MAC Address: 18:D6:C7:00:EA:6C (Tp-link Technologies)

Nmap done: 1 IP address (1 host up) scanned in 5.08 seconds

Определение версий ОС и служб в сочетании с TCP SYN сканирование

user@host:/# nmap -O -sV -sS 192.168.0.88

Starting Nmap 7.70 (<https://nmap.org>) at 2021-12-13 14:35 MSK

Nmap scan report for server.account.school34 (192.168.0.88)

Host is up (0.00021s latency).

Not shown: 999 filtered ports

PORT STATE SERVICE VERSION

22/tcp open ssh OpenSSH 7.9p1 Debian 10+deb10u2 (protocol 2.0)

MAC Address: 18:D6:C7:00:EA:6C (Tp-link Technologies)

Warning: OSScan results may be unreliable because we could not find at least 1 open and 1 closed port

Device type: general purpose

Running: Linux 2.6.X

OS CPE: cpe:/o:linux:linux_kernel:2.6

OS details: Linux 2.6.18 - 2.6.22

Network Distance: 1 hop

Service Info: OS: Linux; CPE: cpe:/o:linux:linux_kernel

OS and Service detection performed. Please report any incorrect results at <https://nmap.org/submit/> .

Nmap done: 1 IP address (1 host up) scanned in 13.43 seconds

Протокол сканирования с компьютера из ядра сети (а именно с сервера виртуализации):

Пинг-сканирование

user@host:/# nmap -sP 192.168.0.88

Starting Nmap 7.70 (<https://nmap.org>) at 2021-12-13 14:52 MSK

Nmap scan report for 192.168.0.88

Host is up (0.00016s latency).

MAC Address: 18:D6:C7:00:EA:6C (Tp-link Technologies)

Nmap done: 1 IP address (1 host up) scanned in 0.39 seconds

TCP SYN пинг

user@host:/# nmap -PS 192.168.0.88

Starting Nmap 7.70 (<https://nmap.org>) at 2021-12-13 14:52 MSK

Nmap scan report for 192.168.0.88

Host is up (0.00017s latency).

All 1000 scanned ports on 192.168.0.88 are filtered

MAC Address: 18:D6:C7:00:EA:6C (Tp-link Technologies)

Nmap done: 1 IP address (1 host up) scanned in 21.51 seconds

TCP ACK пинг

user@host:/# nmap -PA 192.168.0.88

Starting Nmap 7.70 (<https://nmap.org>) at 2021-12-13 14:52 MSK

Nmap scan report for 192.168.0.88

Host is up (0.00019s latency).

All 1000 scanned ports on 192.168.0.88 are filtered

MAC Address: 18:D6:C7:00:EA:6C (Tp-link Technologies)

Nmap done: 1 IP address (1 host up) scanned in 21.49 seconds

UDP пинг

user@host:/# nmap -PU 192.168.0.88

Starting Nmap 7.70 (<https://nmap.org>) at 2021-12-13 14:52 MSK

Nmap scan report for 192.168.0.88

Host is up (0.00014s latency).

All 1000 scanned ports on 192.168.0.88 are filtered

MAC Address: 18:D6:C7:00:EA:6C (Tp-link Technologies)

Nmap done: 1 IP address (1 host up) scanned in 21.48 seconds

Различные типы пинг-пакетов ICMP

user@host:/# nmap -PE 192.168.0.88

Starting Nmap 7.70 (<https://nmap.org>) at 2021-12-13 14:53 MSK

Nmap scan report for 192.168.0.88

Host is up (0.00016s latency).

All 1000 scanned ports on 192.168.0.88 are filtered

MAC Address: 18:D6:C7:00:EA:6C (Tp-link Technologies)

Nmap done: 1 IP address (1 host up) scanned in 21.49 seconds

user@host:/# nmap -PP 192.168.0.88

Starting Nmap 7.70 (<https://nmap.org>) at 2021-12-13 14:53 MSK

Nmap scan report for 192.168.0.88

Host is up (0.00013s latency).

All 1000 scanned ports on 192.168.0.88 are filtered

MAC Address: 18:D6:C7:00:EA:6C (Tp-link Technologies)

Nmap done: 1 IP address (1 host up) scanned in 21.49 seconds

user@host:/# nmap -PM 192.168.0.88

Starting Nmap 7.70 (<https://nmap.org>) at 2021-12-13 14:53 MSK

Nmap scan report for 192.168.0.88

Host is up (0.00018s latency).

All 1000 scanned ports on 192.168.0.88 are filtered

MAC Address: 18:D6:C7:00:EA:6C (Tp-link Technologies)

Nmap done: 1 IP address (1 host up) scanned in 21.49 seconds

Пинг с использование протокола IP

user@host:/# nmap -PO 192.168.0.88

Starting Nmap 7.70 (<https://nmap.org>) at 2021-12-13 14:54 MSK

Nmap scan report for 192.168.0.88

Host is up (0.00017s latency).

All 1000 scanned ports on 192.168.0.88 are filtered

MAC Address: 18:D6:C7:00:EA:6C (Tp-link Technologies)

Nmap done: 1 IP address (1 host up) scanned in 21.48 seconds

ARP пинг

user@host:/# nmap -PR 192.168.0.88

Starting Nmap 7.70 (<https://nmap.org>) at 2021-12-13 14:54 MSK

Nmap scan report for 192.168.0.88

Host is up (0.00016s latency).

All 1000 scanned ports on 192.168.0.88 are filtered

MAC Address: 18:D6:C7:00:EA:6C (Tp-link Technologies)

Nmap done: 1 IP address (1 host up) scanned in 21.48 seconds

TCP connect сканирование

user@host:/# nmap -sT 192.168.0.88

Starting Nmap 7.70 (<https://nmap.org>) at 2021-12-13 14:54 MSK

Nmap scan report for 192.168.0.88

Host is up (0.00017s latency).

All 1000 scanned ports on 192.168.0.88 are filtered

MAC Address: 18:D6:C7:00:EA:6C (Tp-link Technologies)

Nmap done: 1 IP address (1 host up) scanned in 21.39 seconds

UDP сканирование

user@host:/# nmap -sU 192.168.0.88

Starting Nmap 7.70 (<https://nmap.org>) at 2021-12-13 14:55 MSK

Nmap scan report for 192.168.0.88

Host is up (0.00016s latency).

All 1000 scanned ports on 192.168.0.88 are open|filtered

MAC Address: 18:D6:C7:00:EA:6C (Tp-link Technologies)

Nmap done: 1 IP address (1 host up) scanned in 21.48 seconds

TCP NULL сканирование

user@host:/# nmap -sN 192.168.0.88

Starting Nmap 7.70 (<https://nmap.org>) at 2021-12-13 14:55 MSK

Nmap scan report for 192.168.0.88

Host is up (0.00016s latency).

All 1000 scanned ports on 192.168.0.88 are open|filtered

MAC Address: 18:D6:C7:00:EA:6C (Tp-link Technologies)

Nmap done: 1 IP address (1 host up) scanned in 21.49 seconds

TCP FIN сканирование

user@host:/# nmap -sF 192.168.0.88

Starting Nmap 7.70 (<https://nmap.org>) at 2021-12-13 14:56 MSK

Nmap scan report for 192.168.0.88

Host is up (0.00017s latency).

All 1000 scanned ports on 192.168.0.88 are open|filtered

MAC Address: 18:D6:C7:00:EA:6C (Tp-link Technologies)

Nmap done: 1 IP address (1 host up) scanned in 21.48 seconds

TCP Xmas сканирование

user@host:/# nmap -sX 192.168.0.88

Starting Nmap 7.70 (<https://nmap.org>) at 2021-12-13 14:56 MSK

Nmap scan report for 192.168.0.88

Host is up (0.00017s latency).

All 1000 scanned ports on 192.168.0.88 are open|filtered

MAC Address: 18:D6:C7:00:EA:6C (Tp-link Technologies)

Nmap done: 1 IP address (1 host up) scanned in 21.49 seconds

TCP ACK сканирование

user@host:/# nmap -sA 192.168.0.88

Starting Nmap 7.70 (<https://nmap.org>) at 2021-12-13 14:56 MSK

Nmap scan report for 192.168.0.88

Host is up (0.00017s latency).

All 1000 scanned ports on 192.168.0.88 are filtered

MAC Address: 18:D6:C7:00:EA:6C (Tp-link Technologies)

Nmap done: 1 IP address (1 host up) scanned in 21.49 seconds
TCP Window сканирование
user@host:/# nmap -sW 192.168.0.88
Starting Nmap 7.70 (<https://nmap.org>) at 2021-12-13 14:57 MSK
Nmap scan report for 192.168.0.88
Host is up (0.00023s latency).
All 1000 scanned ports on 192.168.0.88 are filtered
MAC Address: 18:D6:C7:00:EA:6C (Tp-link Technologies)

Nmap done: 1 IP address (1 host up) scanned in 21.47 seconds
TCP сканирование Мэймона
user@host:/# nmap -sM 192.168.0.88
Starting Nmap 7.70 (<https://nmap.org>) at 2021-12-13 14:57 MSK
Nmap scan report for 192.168.0.88
Host is up (0.00017s latency).
All 1000 scanned ports on 192.168.0.88 are open|filtered
MAC Address: 18:D6:C7:00:EA:6C (Tp-link Technologies)

Nmap done: 1 IP address (1 host up) scanned in 21.51 seconds
Сканирование протокола IP
user@host:/# nmap -sO 192.168.0.88
Starting Nmap 7.70 (<https://nmap.org>) at 2021-12-13 14:57 MSK
Nmap scan report for 192.168.0.88
Host is up (0.00017s latency).
Not shown: 255 open|filtered protocols
PROTOCOL STATE SERVICE
1 open icmp
MAC Address: 18:D6:C7:00:EA:6C (Tp-link Technologies)

Nmap done: 1 IP address (1 host up) scanned in 3.73 seconds
Определение версий ОС и служб в сочетании с TCP SYN сканирование
user@host:/# nmap -O -sV -sS 192.168.0.88
Starting Nmap 7.70 (<https://nmap.org>) at 2021-12-13 14:57 MSK
Nmap scan report for 192.168.0.88
Host is up (0.00017s latency).
All 1000 scanned ports on 192.168.0.88 are filtered
MAC Address: 18:D6:C7:00:EA:6C (Tp-link Technologies)
Too many fingerprints match this host to give specific OS details
Network Distance: 1 hop

OS and Service detection performed. Please report any incorrect results at
<https://nmap.org/submit/> .
Nmap done: 1 IP address (1 host up) scanned in 25.12 seconds

Протокол сканирования с компьютера-клиента:

Пинг-сканирование
user@host:/# nmap -sP 10.9.1.129
Starting Nmap 7.92 (<https://nmap.org>) at 2021-12-13 16:04 UTC
Nmap scan report for ntp.account.school34 (10.9.1.129)
Host is up (0.00027s latency).
MAC Address: 1C:6F:65:C2:C5:6F (Giga-byte Technology)
Nmap done: 1 IP address (1 host up) scanned in 0.63 seconds
TCP SYN пинг
user@host:/# nmap -PS 10.9.1.129
Starting Nmap 7.92 (<https://nmap.org>) at 2021-12-13 16:04 UTC
Nmap scan report for ntp.account.school34 (10.9.1.129)
Host is up (0.00022s latency).
Not shown: 997 filtered tcp ports (no-response)
PORT STATE SERVICE
53/tcp open domain
139/tcp open netbios-ssn

445/tcp open microsoft-ds
MAC Address: 1C:6F:65:C2:C5:6F (Giga-byte Technology)

Nmap done: 1 IP address (1 host up) scanned in 5.25 seconds

TCP ACK пинг

user@host:/# nmap -PA 10.9.1.129

Starting Nmap 7.92 (<https://nmap.org>) at 2021-12-13 16:04 UTC

Nmap scan report for ntp.account.school34 (10.9.1.129)

Host is up (0.00024s latency).

Not shown: 997 filtered tcp ports (no-response)

PORT STATE SERVICE

53/tcp open domain

139/tcp open netbios-ssn

445/tcp open microsoft-ds

MAC Address: 1C:6F:65:C2:C5:6F (Giga-byte Technology)

Nmap done: 1 IP address (1 host up) scanned in 5.16 seconds

UDP пинг

user@host:/# nmap -PU 10.9.1.129

Starting Nmap 7.92 (<https://nmap.org>) at 2021-12-13 16:05 UTC

Nmap scan report for ntp.account.school34 (10.9.1.129)

Host is up (0.00028s latency).

Not shown: 997 filtered tcp ports (no-response)

PORT STATE SERVICE

53/tcp open domain

139/tcp open netbios-ssn

445/tcp open microsoft-ds

MAC Address: 1C:6F:65:C2:C5:6F (Giga-byte Technology)

Nmap done: 1 IP address (1 host up) scanned in 4.66 seconds

Различные типы пинг-пакетов ICMP

user@host:/# nmap -PE 10.9.1.129

Starting Nmap 7.92 (<https://nmap.org>) at 2021-12-13 16:05 UTC

Nmap scan report for ntp.account.school34 (10.9.1.129)

Host is up (0.00023s latency).

Not shown: 997 filtered tcp ports (no-response)

PORT STATE SERVICE

53/tcp open domain

139/tcp open netbios-ssn

445/tcp open microsoft-ds

MAC Address: 1C:6F:65:C2:C5:6F (Giga-byte Technology)

Nmap done: 1 IP address (1 host up) scanned in 5.15 seconds

user@host:/# nmap -PP 10.9.1.129

Starting Nmap 7.92 (<https://nmap.org>) at 2021-12-13 16:05 UTC

Nmap scan report for ntp.account.school34 (10.9.1.129)

Host is up (0.00025s latency).

Not shown: 997 filtered tcp ports (no-response)

PORT STATE SERVICE

53/tcp open domain

139/tcp open netbios-ssn

445/tcp open microsoft-ds

MAC Address: 1C:6F:65:C2:C5:6F (Giga-byte Technology)

Nmap done: 1 IP address (1 host up) scanned in 5.16 seconds

user@host:/# nmap -PM 10.9.1.129

Starting Nmap 7.92 (<https://nmap.org>) at 2021-12-13 16:05 UTC

Nmap scan report for ntp.account.school34 (10.9.1.129)

Host is up (0.00029s latency).

Not shown: 997 filtered tcp ports (no-response)

PORT STATE SERVICE

53/tcp open domain

139/tcp open netbios-ssn
445/tcp open microsoft-ds
MAC Address: 1C:6F:65:C2:C5:6F (Giga-byte Technology)

Nmap done: 1 IP address (1 host up) scanned in 5.18 seconds

Пинг с использованием протокола IP

user@host:/# nmap -PO 10.9.1.129

Starting Nmap 7.92 (<https://nmap.org>) at 2021-12-13 16:05 UTC

Nmap scan report for ntp.account.school34 (10.9.1.129)

Host is up (0.00020s latency).

Not shown: 997 filtered tcp ports (no-response)

PORT	STATE	SERVICE
------	-------	---------

53/tcp	open	domain
--------	------	--------

139/tcp	open	netbios-ssn
---------	------	-------------

445/tcp	open	microsoft-ds
---------	------	--------------

MAC Address: 1C:6F:65:C2:C5:6F (Giga-byte Technology)

Nmap done: 1 IP address (1 host up) scanned in 4.66 seconds

ARP пинг

user@host:/# nmap -PR 10.9.1.129

Starting Nmap 7.92 (<https://nmap.org>) at 2021-12-13 16:05 UTC

Nmap scan report for ntp.account.school34 (10.9.1.129)

Host is up (0.00019s latency).

Not shown: 997 filtered tcp ports (no-response)

PORT	STATE	SERVICE
------	-------	---------

53/tcp	open	domain
--------	------	--------

139/tcp	open	netbios-ssn
---------	------	-------------

445/tcp	open	microsoft-ds
---------	------	--------------

MAC Address: 1C:6F:65:C2:C5:6F (Giga-byte Technology)

Nmap done: 1 IP address (1 host up) scanned in 4.58 seconds

TCP connect сканирование

user@host:/# nmap -sT 10.9.1.129

Starting Nmap 7.92 (<https://nmap.org>) at 2021-12-13 16:05 UTC

Nmap scan report for ntp.account.school34 (10.9.1.129)

Host is up (0.00035s latency).

Not shown: 997 filtered tcp ports (no-response)

PORT	STATE	SERVICE
------	-------	---------

53/tcp	open	domain
--------	------	--------

139/tcp	open	netbios-ssn
---------	------	-------------

445/tcp	open	microsoft-ds
---------	------	--------------

MAC Address: 1C:6F:65:C2:C5:6F (Giga-byte Technology)

Nmap done: 1 IP address (1 host up) scanned in 4.57 seconds

UDP сканирование

user@host:/# nmap -sU 10.9.1.129

Starting Nmap 7.92 (<https://nmap.org>) at 2021-12-13 16:05 UTC

Nmap scan report for ntp.account.school34 (10.9.1.129)

Host is up (0.00030s latency).

Not shown: 994 open|filtered udp ports (no-response)

PORT	STATE	SERVICE
------	-------	---------

53/udp	open	domain
--------	------	--------

67/udp	open	dhcps
--------	------	-------

123/udp	open	ntp
---------	------	-----

137/udp	open	netbios-ns
---------	------	------------

139/udp	closed	netbios-ssn
---------	--------	-------------

445/udp	closed	microsoft-ds
---------	--------	--------------

MAC Address: 1C:6F:65:C2:C5:6F (Giga-byte Technology)

Nmap done: 1 IP address (1 host up) scanned in 6.08 seconds

TCP NULL сканирование

user@host:/# nmap -sN 10.9.1.129

Starting Nmap 7.92 (<https://nmap.org>) at 2021-12-13 16:05 UTC
Nmap scan report for ntp.account.school34 (10.9.1.129)
Host is up (0.00019s latency).
All 1000 scanned ports on ntp.account.school34 (10.9.1.129) are in ignored states.
Not shown: 1000 open|filtered tcp ports (no-response)
MAC Address: 1C:6F:65:C2:C5:6F (Giga-byte Technology)

Nmap done: 1 IP address (1 host up) scanned in 21.29 seconds

TCP FIN сканирование

user@host:/# nmap -sF 10.9.1.129

Starting Nmap 7.92 (<https://nmap.org>) at 2021-12-13 16:06 UTC
Nmap scan report for ntp.account.school34 (10.9.1.129)
Host is up (0.00019s latency).
All 1000 scanned ports on ntp.account.school34 (10.9.1.129) are in ignored states.
Not shown: 1000 open|filtered tcp ports (no-response)
MAC Address: 1C:6F:65:C2:C5:6F (Giga-byte Technology)

Nmap done: 1 IP address (1 host up) scanned in 21.29 seconds

TCP Xmas сканирование

user@host:/# nmap -sX 10.9.1.129

Starting Nmap 7.92 (<https://nmap.org>) at 2021-12-13 16:06 UTC
Nmap scan report for ntp.account.school34 (10.9.1.129)
Host is up (0.00020s latency).
All 1000 scanned ports on ntp.account.school34 (10.9.1.129) are in ignored states.
Not shown: 1000 open|filtered tcp ports (no-response)
MAC Address: 1C:6F:65:C2:C5:6F (Giga-byte Technology)

Nmap done: 1 IP address (1 host up) scanned in 21.31 seconds

TCP ACK сканирование

user@host:/# nmap -sA 10.9.1.129

Starting Nmap 7.92 (<https://nmap.org>) at 2021-12-13 16:06 UTC
Nmap scan report for ntp.account.school34 (10.9.1.129)
Host is up (0.00028s latency).
Not shown: 997 filtered tcp ports (no-response)
PORT STATE SERVICE
53/tcp unfiltered domain
139/tcp unfiltered netbios-ssn
445/tcp unfiltered microsoft-ds
MAC Address: 1C:6F:65:C2:C5:6F (Giga-byte Technology)

Nmap done: 1 IP address (1 host up) scanned in 4.95 seconds

TCP Window сканирование

user@host:/# nmap -sW 10.9.1.129

Starting Nmap 7.92 (<https://nmap.org>) at 2021-12-13 16:06 UTC
Nmap scan report for ntp.account.school34 (10.9.1.129)
Host is up (0.00023s latency).
Not shown: 997 filtered tcp ports (no-response)
PORT STATE SERVICE
53/tcp closed domain
139/tcp closed netbios-ssn
445/tcp closed microsoft-ds
MAC Address: 1C:6F:65:C2:C5:6F (Giga-byte Technology)

Nmap done: 1 IP address (1 host up) scanned in 4.29 seconds

TCP сканирование Мэймона

user@host:/# nmap -sM 10.9.1.129

Starting Nmap 7.92 (<https://nmap.org>) at 2021-12-13 16:06 UTC
Nmap scan report for ntp.account.school34 (10.9.1.129)
Host is up (0.00019s latency).
Not shown: 997 open|filtered tcp ports (no-response)
PORT STATE SERVICE
53/tcp closed domain

139/tcp closed netbios-ssn
445/tcp closed microsoft-ds
MAC Address: 1C:6F:65:C2:C5:6F (Giga-byte Technology)

Nmap done: 1 IP address (1 host up) scanned in 4.77 seconds

Сканирование протокола IP

user@host:/# nmap -sO 10.9.1.129

Starting Nmap 7.92 (<https://nmap.org>) at 2021-12-13 16:07 UTC

Nmap scan report for ntp.account.school34 (10.9.1.129)

Host is up (0.00024s latency).

Not shown: 255 open|filtered n/a protocols (no-response)

PROTOCOL STATE SERVICE

1 open icmp

MAC Address: 1C:6F:65:C2:C5:6F (Giga-byte Technology)

Nmap done: 1 IP address (1 host up) scanned in 5.21 seconds

Определение версий ОС и служб в сочетании с TCP SYN сканирование

user@host:/# nmap -O -sV -sS 10.9.1.129

Starting Nmap 7.92 (<https://nmap.org>) at 2021-12-13 16:07 UTC

Nmap scan report for ntp.account.school34 (10.9.1.129)

Host is up (0.00051s latency).

Not shown: 997 filtered tcp ports (no-response)

PORT	STATE	SERVICE	VERSION
------	-------	---------	---------

53/tcp	open	domain	dnsmasq 2.80
--------	------	--------	--------------

139/tcp	open	netbios-ssn	Samba smbd 3.X - 4.X (workgroup: ACCOUNT)
---------	------	-------------	---

445/tcp	open	netbios-ssn	Samba smbd 3.X - 4.X (workgroup: ACCOUNT)
---------	------	-------------	---

MAC Address: 1C:6F:65:C2:C5:6F (Giga-byte Technology)

Warning: OSScan results may be unreliable because we could not find at least 1 open and 1 closed port

Device type: general purpose

Running: Linux 4.X|5.X

OS CPE: cpe:/o:linux:linux_kernel:4 cpe:/o:linux:linux_kernel:5

OS details: Linux 4.15 - 5.6, Linux 5.0 - 5.4

Network Distance: 1 hop

Service Info: Host: SAMBA

OS and Service detection performed. Please report any incorrect results at <https://nmap.org/submit/> .

Nmap done: 1 IP address (1 host up) scanned in 24.63 seconds

Протокол сканирования с компьютера-нарушителя из внутренней сети:

Пинг-сканирование

user@host:/# nmap -sP 10.9.1.129

Starting Nmap 7.92 (<https://nmap.org>) at 2021-12-13 16:11 UTC

Nmap scan report for 10.9.1.129

Host is up (0.00018s latency).

MAC Address: 1C:6F:65:C2:C5:6F (Giga-byte Technology)

Nmap done: 1 IP address (1 host up) scanned in 13.14 seconds

TCP SYN пинг

user@host:/# nmap -PS 10.9.1.129

Starting Nmap 7.92 (<https://nmap.org>) at 2021-12-13 16:11 UTC

Nmap scan report for 10.9.1.129

Host is up (0.00015s latency).

All 1000 scanned ports on 10.9.1.129 are in ignored states.

Not shown: 1000 filtered tcp ports (no-response)

MAC Address: 1C:6F:65:C2:C5:6F (Giga-byte Technology)

Nmap done: 1 IP address (1 host up) scanned in 34.31 seconds

TCP ACK пинг

user@host:/# nmap -PA 10.9.1.129

Starting Nmap 7.92 (<https://nmap.org>) at 2021-12-13 16:12 UTC

Nmap scan report for 10.9.1.129
Host is up (0.00016s latency).
All 1000 scanned ports on 10.9.1.129 are in ignored states.
Not shown: 1000 filtered tcp ports (no-response)
MAC Address: 1C:6F:65:C2:C5:6F (Giga-byte Technology)

Nmap done: 1 IP address (1 host up) scanned in 34.30 seconds

UDP пинг

user@host:/# nmap -PU 10.9.1.129

Starting Nmap 7.92 (<https://nmap.org>) at 2021-12-13 16:13 UTC

Nmap scan report for 10.9.1.129

Host is up (0.00025s latency).

All 1000 scanned ports on 10.9.1.129 are in ignored states.

Not shown: 1000 filtered tcp ports (no-response)

MAC Address: 1C:6F:65:C2:C5:6F (Giga-byte Technology)

Nmap done: 1 IP address (1 host up) scanned in 34.30 seconds

Различные типы пинг-пакетов ICMP

user@host:/# nmap -PE 10.9.1.129

Starting Nmap 7.92 (<https://nmap.org>) at 2021-12-13 16:13 UTC

Nmap scan report for 10.9.1.129

Host is up (0.00016s latency).

All 1000 scanned ports on 10.9.1.129 are in ignored states.

Not shown: 1000 filtered tcp ports (no-response)

MAC Address: 1C:6F:65:C2:C5:6F (Giga-byte Technology)

Nmap done: 1 IP address (1 host up) scanned in 34.29 seconds

user@host:/# nmap -PP 10.9.1.129

Starting Nmap 7.92 (<https://nmap.org>) at 2021-12-13 16:14 UTC

Nmap scan report for 10.9.1.129

Host is up (0.00018s latency).

All 1000 scanned ports on 10.9.1.129 are in ignored states.

Not shown: 1000 filtered tcp ports (no-response)

MAC Address: 1C:6F:65:C2:C5:6F (Giga-byte Technology)

Nmap done: 1 IP address (1 host up) scanned in 34.30 seconds

user@host:/# nmap -PM 10.9.1.129

Starting Nmap 7.92 (<https://nmap.org>) at 2021-12-13 16:14 UTC

Nmap scan report for 10.9.1.129

Host is up (0.00028s latency).

All 1000 scanned ports on 10.9.1.129 are in ignored states.

Not shown: 1000 filtered tcp ports (no-response)

MAC Address: 1C:6F:65:C2:C5:6F (Giga-byte Technology)

Nmap done: 1 IP address (1 host up) scanned in 34.30 seconds

пинг с использование протокола IP

user@host:/# nmap -PO 10.9.1.129

Starting Nmap 7.92 (<https://nmap.org>) at 2021-12-13 16:15 UTC

Nmap scan report for 10.9.1.129

Host is up (0.00017s latency).

All 1000 scanned ports on 10.9.1.129 are in ignored states.

Not shown: 1000 filtered tcp ports (no-response)

MAC Address: 1C:6F:65:C2:C5:6F (Giga-byte Technology)

Nmap done: 1 IP address (1 host up) scanned in 34.31 seconds

ARP пинг

user@host:/# nmap -PR 10.9.1.129

Starting Nmap 7.92 (<https://nmap.org>) at 2021-12-13 16:15 UTC

Nmap scan report for 10.9.1.129

Host is up (0.00017s latency).

All 1000 scanned ports on 10.9.1.129 are in ignored states.

Not shown: 1000 filtered tcp ports (no-response)

MAC Address: 1C:6F:65:C2:C5:6F (Giga-byte Technology)

Nmap done: 1 IP address (1 host up) scanned in 34.31 seconds

TCP connect сканирование

user@host:/# nmap -sT 10.9.1.129

Starting Nmap 7.92 (<https://nmap.org>) at 2021-12-13 16:16 UTC

Nmap scan report for 10.9.1.129

Host is up (0.00018s latency).

All 1000 scanned ports on 10.9.1.129 are in ignored states.

Not shown: 1000 filtered tcp ports (no-response)

MAC Address: 1C:6F:65:C2:C5:6F (Giga-byte Technology)

Nmap done: 1 IP address (1 host up) scanned in 34.21 seconds

UDP сканирование

user@host:/# nmap -sU 10.9.1.129

Starting Nmap 7.92 (<https://nmap.org>) at 2021-12-13 16:17 UTC

Nmap scan report for 10.9.1.129

Host is up (0.00028s latency).

All 1000 scanned ports on 10.9.1.129 are in ignored states.

Not shown: 1000 open|filtered udp ports (no-response)

MAC Address: 1C:6F:65:C2:C5:6F (Giga-byte Technology)

Nmap done: 1 IP address (1 host up) scanned in 34.34 seconds

TCP NULL сканирование

user@host:/# nmap -sN 10.9.1.129

Starting Nmap 7.92 (<https://nmap.org>) at 2021-12-13 16:17 UTC

Nmap scan report for 10.9.1.129

Host is up (0.00019s latency).

All 1000 scanned ports on 10.9.1.129 are in ignored states.

Not shown: 1000 open|filtered tcp ports (no-response)

MAC Address: 1C:6F:65:C2:C5:6F (Giga-byte Technology)

Nmap done: 1 IP address (1 host up) scanned in 34.28 seconds

TCP FIN сканирование

user@host:/# nmap -sF 10.9.1.129

Starting Nmap 7.92 (<https://nmap.org>) at 2021-12-13 16:18 UTC

Nmap scan report for 10.9.1.129

Host is up (0.00023s latency).

All 1000 scanned ports on 10.9.1.129 are in ignored states.

Not shown: 1000 open|filtered tcp ports (no-response)

MAC Address: 1C:6F:65:C2:C5:6F (Giga-byte Technology)

Nmap done: 1 IP address (1 host up) scanned in 34.30 seconds

TCP Xmas сканирование

user@host:/# nmap -sX 10.9.1.129

Starting Nmap 7.92 (<https://nmap.org>) at 2021-12-13 16:18 UTC

Nmap scan report for 10.9.1.129

Host is up (0.00018s latency).

All 1000 scanned ports on 10.9.1.129 are in ignored states.

Not shown: 1000 open|filtered tcp ports (no-response)

MAC Address: 1C:6F:65:C2:C5:6F (Giga-byte Technology)

Nmap done: 1 IP address (1 host up) scanned in 34.32 seconds

TCP ACK сканирование

user@host:/# nmap -sA 10.9.1.129

Starting Nmap 7.92 (<https://nmap.org>) at 2021-12-13 16:19 UTC

Nmap scan report for 10.9.1.129

Host is up (0.00028s latency).

All 1000 scanned ports on 10.9.1.129 are in ignored states.

Not shown: 1000 filtered tcp ports (no-response)

MAC Address: 1C:6F:65:C2:C5:6F (Giga-byte Technology)

Nmap done: 1 IP address (1 host up) scanned in 34.28 seconds
TCP Window сканирование
user@host:/# nmap -sW 10.9.1.129
Starting Nmap 7.92 (<https://nmap.org>) at 2021-12-13 16:19 UTC
Nmap scan report for 10.9.1.129
Host is up (0.00017s latency).
All 1000 scanned ports on 10.9.1.129 are in ignored states.
Not shown: 1000 filtered tcp ports (no-response)
MAC Address: 1C:6F:65:C2:C5:6F (Giga-byte Technology)

Nmap done: 1 IP address (1 host up) scanned in 34.30 seconds
TCP сканирование Мэймона
user@host:/# nmap -sM 10.9.1.129
Starting Nmap 7.92 (<https://nmap.org>) at 2021-12-13 16:20 UTC
Nmap scan report for 10.9.1.129
Host is up (0.00019s latency).
All 1000 scanned ports on 10.9.1.129 are in ignored states.
Not shown: 1000 open|filtered tcp ports (no-response)
MAC Address: 1C:6F:65:C2:C5:6F (Giga-byte Technology)

Nmap done: 1 IP address (1 host up) scanned in 34.31 seconds
Сканирование протокола IP
user@host:/# nmap -sO 10.9.1.129
Starting Nmap 7.92 (<https://nmap.org>) at 2021-12-13 16:21 UTC
Nmap scan report for 10.9.1.129
Host is up (0.00019s latency).
All 256 scanned ports on 10.9.1.129 are in ignored states.
Not shown: 256 open|filtered n/a protocols (no-response)
MAC Address: 1C:6F:65:C2:C5:6F (Giga-byte Technology)

Nmap done: 1 IP address (1 host up) scanned in 19.45 seconds
Определение версий ОС и служб в сочетании с TCP SYN сканирование
user@host:/# nmap -O -sV -sS 10.9.1.129
Starting Nmap 7.92 (<https://nmap.org>) at 2021-12-13 16:21 UTC
Nmap scan report for 10.9.1.129
Host is up (0.00016s latency).
All 1000 scanned ports on 10.9.1.129 are in ignored states.
Not shown: 1000 filtered tcp ports (no-response)
MAC Address: 1C:6F:65:C2:C5:6F (Giga-byte Technology)
Too many fingerprints match this host to give specific OS details
Network Distance: 1 hop

OS and Service detection performed. Please report any incorrect results at
<https://nmap.org/submit/> .
Nmap done: 1 IP address (1 host up) scanned in 37.53 seconds

Анализируя протоколы проверки сетевым сканером, можно сделать следующие выводы:

При сканировании системы с компьютера системного администратора доступным оказался только порт службы ssh, что и следовало ожидать. Кроме того, оказались возможны проверка доступности системы при помощи протокола ICMP и получение его аппаратного адреса, что также не противоречит постановке задачи и ожидаемым результатам. При определении версий программного обеспечения служба ssh не только позволила определить собственную версию, но и сообщила версию операционной системы. Однако считать это недостатком не следует, так как при проверке системы с постороннего компьютера из ядра сети получить этих сведений не удалось.

При сканировании системы клиентом подсети удалось обнаружить только порты допустимых к использованию клиентами служб, служба удалённого администрирования

(ssh) не была выявлена. При определении версий программного обеспечения удалось выяснить, что услуги dns-сервера предоставляет dnsmasq версии 2.80, работающий под управлением ОС Linux с версией ядра 4.15-5.6. Также были обнаружены сервис Samba (версии 3.X – 4.X) и рабочая группа ACCOUNT. Такие результаты также следует считать полностью соответствующими поставленным требованиям.

При сканировании с компьютера-нарушителя, искусственно введённого в сеть бухгалтерии с назначенным вручную IP-адресом, удалось выяснить только адрес канального уровня внутреннего интерфейса сервера и тот факт, что сервер работал на момент тестирования. Никаких сведений о работающих службах, версиях программного обеспечения или операционной системы получено не было.

Кроме защиты самого сервера подсети бухгалтерии, межсетевой экран должен повышать безопасность и клиентов этой сети. Так попытка найти маршрут с сервера виртуализации к одному из клиентских устройств, имеющему адрес 10.9.1.134, завершилась неудачей, как и попытка обнаружения этого клиента утилитой ping:

```
root@virtserver:/# traceroute 10.9.1.134
traceroute to 10.9.1.134 (10.9.1.134), 30 hops max, 60 byte packets
 1  192.168.0.1 (192.168.0.1)  0.547 ms  0.501 ms  0.468 ms
 2  server.account.school34 (192.168.0.88)  0.415 ms  0.384 ms  0.359 ms
 3  * * *
...
30 * * *
root@virtserver:/# ping 10.9.1.134
PING 10.9.1.134 (10.9.1.134) 56(84) bytes of data.
From 192.168.0.1: icmp_seq=2 Redirect Host(New nexthop: 192.168.0.88)
...
^C
--- 10.9.1.134 ping statistics ---
21 packets transmitted, 0 received, 100% packet loss, time 482ms
```

Замечание: здесь имя сервера виртуализации (server, server.service.school34) в приглашении командной строки изменено во избежание путаницы с именем сервера подсети бухгалтерии (server, server.account.school34) на virtserver.

При этом клиент имеет свободный доступ к виртуальным серверам, расположенным на этом сервере виртуализации. Одновременно с тем, и клиент не имеет доступа к устройствам в других подсетях.

Завершение установки

В завершение установки имеет смысл обновить систему, очистить локальный кэш пакетов и удалить пакеты, которые системе стали не нужны. Это можно сделать командами:

```
root@server:/# apt-get upgrade
root@server:/# apt-get autoremove
root@server:/# apt-get clean
```

Использование и обслуживание сервера

Описанные далее сценарии, команды и методики предназначены для использования в процессе функционирования системы для контроля её состояния или смены режимов

функционирования. Дальнейшее описание можно воспринимать как краткое руководство (howto).

Порядок входа в систему

В силу описанных выше настроек ssh и сетевого фильтра предусмотрен следующий порядок удалённого входа в систему: выполнить вход с компьютера системного администратора от имени administrator, а затем при необходимости выполнить повышение привилегий до root локально. Эти действия продемонстрированы в следующем фрагменте терминального сеанса:

```
administrator@admin:~$ ssh administrator@server.account.school34
administrator@server.account.school34's password:
Last login: Thu Oct 21 10:58:58 2021 from 192.168.0.2
```

```
The programs included with the Debian GNU/Linux system are free software;
the exact distribution terms for each program are described in the
individual files in /usr/share/doc/*/copyright.
```

```
Debian GNU/Linux comes with ABSOLUTELY NO WARRANTY, to the extent
permitted by applicable law.
```

```
administrator@server:~$ su
Password:
root@server:/home/administrator#
```

При обновлении системы

Поскольку в системе приняты меры по защите загрузчика, включающие в том числе и ручную правку итогового конфигурационного файла **/boot/grub/grub.cfg**, который и используется непосредственно загрузчиком, возникает необходимость повторно выполнять такие ручные правки в том числе и при обновлении системы, если в ходе этого обновления выполнялась пересборка и указанного конфигурационного файла. Обычно такая пересборка выполняется командой `update-grub`. Если ручные правки не будут выполнены, загрузка системы без ввода пароля для GRUB2 окажется невозможной.

Чтобы сделать такую загрузку возможной, следует лишь дописать в строку меню GRUB2, используемую по умолчанию, ключ `--unrestricted`, что приведёт эту строку к подобному виду:

```
menuentry 'Debian GNU/Linux' --unrestricted --class debian --class gnu-linux ...
```

Мониторинг состояния аппаратных средств

Для получения сведений от аппаратных датчиков достаточно выполнить от имени любого пользователя команду `sensors`, например:

```
administrator@server:/# sensors
coretemp-isa-0000
Adapter: ISA adapter
Core 0:      +35.0°C  (high = +76.0°C, crit = +100.0°C)
Core 1:      +36.0°C  (high = +76.0°C, crit = +100.0°C)
...
```

Здесь вывод команды значительно сокращён для краткости изложения.

Для получения информации от встроенной аппаратуры самодиагностики жёсткого диска можно воспользоваться следующими выполняемыми с правами суперпользователя командами:

Поиск всех S.M.A.R.T-совместимых устройств:

```
root@server:/# /sbin/smartctl --scan
/dev/sda -d scsi # /dev/sda, SCSI device
/dev/sdb -d scsi # /dev/sdb, SCSI device
/dev/sdc -d scsi # /dev/sdc, SCSI device
```

Получение информации об устройстве:

```
root@server:/# /sbin/smartctl /dev/sda -i
smartctl 6.6 2017-11-05 r4594 [x86_64-linux-4.19.0-16-amd64] (local build)
Copyright (C) 2002-17, Bruce Allen, Christian Franke, www.smartmontools.org
```

```
=== START OF INFORMATION SECTION ===
Model Family:      Western Digital Caviar Blue (SATA)
Device Model:      WDC WD3200AAKS-00B3A0
Serial Number:     WD-WMAT10439126
LU WWN Device Id:  5 0014ee 055cb8cb4
Firmware Version:  01.03A01
User Capacity:     320 071 851 520 bytes [320 GB]
Sector Size:       512 bytes logical/physical
Device is:         In smartctl database [for details use: -P show]
ATA Version is:    ATA8-ACS (minor revision not indicated)
SATA Version is:   SATA 2.5, 3.0 Gb/s
Local Time is:     Thu May 27 11:24:43 2021 +03
SMART support is:  Available - device has SMART capability.
SMART support is:  Enabled
```

Получение подробных сведений об устройстве, относящихся к S.M.A.R.T (объёмный вывод команды опущен для краткости):

```
root@server:/# /sbin/smartctl /dev/sda -a
```

Получение ещё более подробных сведений об устройстве, относящихся к S.M.A.R.T и не только (ещё более объёмный вывод команды также опущен для краткости):

```
root@server:/# /sbin/smartctl /dev/sda -x
```

Информация о сетевых адаптерах

В процессе функционирования сервера может появиться необходимость в получении сведений о работе его сетевых интерфейсов. Для этого могут быть полезны, например, следующие команды:

```
root@server:/# /sbin/ethtool enp2s0
root@server:/# /sbin/ethtool -i enp2s0
root@server:/# /sbin/ethtool -S enp2s0
root@server:/# /sbin/ethtool -p enp2s0 10
```

Первая команда позволяет получить подробные сведения о возможных и текущих режимах работы сетевого адаптера, вторая отображает информацию об используемом им драйвере, третья выводит подробную статистику интерфейса, а последняя позволяет в

течение 10 секунд (в этом примере) мигать светодиодом сетевой карты, что бывает полезно для сопоставления физического порта сервера с его именем в системе.

Сложность заключается в том, что не все сетевые адаптеры поддерживают перечисленные функции. Так оба интерфейса системы не поддерживают индикацию светодиодом, а внешняя карта не поддерживает и сбор статистики.

Тем не менее, следующий фрагмент терминального сеанса, приведённый с некоторыми незначущими сокращениями, демонстрирует, что обе карты активны и работают в полнодуплексном режиме с пропускной способностью 1 Гб/с:

```
root@server:/home/administrator# /sbin/ethtool enp2s0
```

```
Settings for enp2s0:
```

```
Supported ports: [ TP MII ]
```

```
Supported link modes:   10baseT/Half 10baseT/Full
                        100baseT/Half 100baseT/Full
                        1000baseT/Half 1000baseT/Full
```

```
Supported pause frame use: Symmetric Receive-only
```

```
Supports auto-negotiation: Yes
```

```
Supported FEC modes: Not reported
```

```
Advertised link modes:  10baseT/Half 10baseT/Full
                        100baseT/Half 100baseT/Full
                        1000baseT/Half 1000baseT/Full
```

```
Advertised pause frame use: Symmetric Receive-only
```

```
Advertised auto-negotiation: Yes
```

```
Advertised FEC modes: Not reported
```

```
Link partner advertised link modes:  10baseT/Half 10baseT/Full
                                      100baseT/Half 100baseT/Full
                                      1000baseT/Full
```

```
Link partner advertised pause frame use: Symmetric
```

```
Link partner advertised auto-negotiation: Yes
```

```
Link partner advertised FEC modes: Not reported
```

```
Speed: 1000Mb/s
```

```
Duplex: Full
```

```
Port: MII
```

```
PHYAD: 0
```

```
Transceiver: internal
```

```
Auto-negotiation: on
```

```
Supports Wake-on: pumbg
```

```
Wake-on: g
```

```
Current message level: 0x00000033 (51)
```

```
drv probe ifdown ifup
```

```
Link detected: yes
```

```
root@server:/home/administrator# /sbin/ethtool -i enp2s0
```

```
driver: r8169
```

```
version:
```

```
firmware-version: rtl_nic/rtl8168e-2.fw
```

```
expansion-rom-version:
```

```
bus-info: 0000:02:00.0
```

```
supports-statistics: yes
```

```
supports-test: no
```

```
supports-eeprom-access: no
```

```
supports-register-dump: yes
```

```
supports-priv-flags: no
```

```
root@server:/home/administrator# /sbin/ethtool -S enp2s0
```

```
NIC statistics:
```

```
tx_packets: 2219
```

```
rx_packets: 4690
```

```
tx_errors: 0
```

```
rx_errors: 0
```

```
rx_missed: 0
```

```

align_errors: 0
tx_single_collisions: 0
tx_multi_collisions: 0
unicast: 2373
broadcast: 2317
multicast: 0
tx_aborted: 0
tx_underrun: 0
root@server:/home/administrator# /sbin/ethtool enp3s0
Settings for enp3s0:
    Supported ports: [ TP ]
    Supported link modes:   10baseT/Half 10baseT/Full
                           100baseT/Half 100baseT/Full
                           1000baseT/Full
    Supported pause frame use: No
    Supports auto-negotiation: Yes
    Supported FEC modes: Not reported
    Advertised link modes:  Not reported
    Advertised pause frame use: No
    Advertised auto-negotiation: Yes
    Advertised FEC modes: Not reported
    Speed: 1000Mb/s
    Duplex: Full
    Port: Twisted Pair
    PHYAD: 0
    Transceiver: internal
    Auto-negotiation: on
    MDI-X: Unknown
    Supports Wake-on: pg
    Wake-on: d
    Current message level: 0x0000003f (63)
                           drv probe link timer ifdown ifup
    Link detected: yes
root@server:/home/administrator# /sbin/ethtool -i enp3s0
driver: atl1c
version: 1.0.1.1-NAPI
firmware-version:
expansion-rom-version:
bus-info: 0000:03:00.0
supports-statistics: no
supports-test: no
supports-eeprom-access: yes
supports-register-dump: yes
supports-priv-flags: no
root@server:/home/administrator# /sbin/ethtool -S enp3s0
no stats available
root@server:/home/administrator# /sbin/ethtool -p enp2s0 10
Cannot identify NIC: Operation not supported
root@server:/home/administrator# /sbin/ethtool -p enp3s0 10
Cannot identify NIC: Operation not supported

```

Подключение клиентов

Наконец, необходимо сделать замечание о подключении клиентских устройств: только само клиентское устройство определяет тот набор служб, которым оно готово пользоваться. Сервер лишь может эти службы предлагать. Так, например, DHCP-клиент Windows-устройств не использует полученное от сервера имя хоста. Другие устройства вполне могут игнорировать и доменное имя, и предлагаемые серверы точного времени и доменных имён. В большинстве случаев это не таит в себе каких-либо сложностей для

самой сети. Исключением являются лишь попытки обхода фильтрации по доменным именам или попытки использования сторонних прокси-серверов, чему, однако, противостоит межсетевой экран.

Также в функционале сервера не заложена возможность предоставлять клиентам информацию об услугах других серверов локальной сети. Таким образом ярлыки на рабочих столах, закладки в интернет-обозревателях и т. п. остаются на усмотрение самих клиентов.

Подключение клиентов к сетевому МФУ

Для подключения клиентов к сетевому МФУ следует в первую очередь подключить клиентскую рабочую станцию и сетевое МФУ к сети, а затем установить на клиентской рабочей станции загруженное с официального сайта производителя программное обеспечение для многофункционального устройства. При этом определение МФУ на клиентском компьютере происходит автоматически, во всяком случае для рассматриваемого в текущем контексте МФУ HP LaserJet Pro MFP M428fdn. Отдельно стоит отметить, что по умолчанию сканирование на компьютер на этом устройстве отключено. Для его включения требуется войти в веб-интерфейс устройства по следующему URL:

<https://mfp.account.school34>

и перейдя там в раздел «Параметры» → «Защита» → «Параметры администратора», установить флаг «Сканирование с компьютера или с мобильного устройства».

Этот URL также может быть использован для создания ярлыка на рабочих столах компьютеров бухгалтерии.

Отдельного внимания заслуживает возможность установки собственного SSL-сертификата для МФУ, заверенного локальным удостоверяющим центром организации. Для этого следует создать запрос на подпись и передать его в локальный УЦ. Затем, получив сертификат, его следует установить в МФУ. При этом закрытый ключ и соответствующий запрос на подпись могут быть созданы как на самом МФУ, так и вне его. Второй способ в рассматриваемой сети более предпочтителен, т. к. в организации действует собственный удостоверяющий центр с централизованными генерацией и хранением закрытых ключей, запросов на подпись и сертификатов. В таком случае все операции по созданию закрытого ключа, CSR и соответствующего им сертификата выполняются в локальном УЦ, а на МФУ устанавливаются готовый сертификат и закрытый ключ. Это выполняется через веб-интерфейс устройства. При этом сертификат и закрытый ключ должны быть представлены в форме единого зашифрованного файла в формате PKCS#12.

Разумеется при этом на клиентах (в данном случае, на компьютерах бухгалтерии) должен быть установлен корневой сертификат локального УЦ. Следует помнить о том, что некоторые программы используют системное хранилище сертификатов, а некоторые ведут собственное, например, браузер Mozilla Firefox.

Подключение клиентов к файловому серверу

Подключение клиентов к файловому серверу удобнее всего выполнять с помощью ярлыков на Рабочем столе клиентских машин. Как альтернативу можно рассмотреть подключение каталогов на постоянной основе как сетевых дисков. Однако в таком

случае, как показала практика предыдущей эксплуатации samba-серверов, сетевые каталоги `shared` и `backup` окажутся не только беззащитны перед вирусом-шифровальщиком, проникшим на любую из клиентских рабочих станций, но и будут легко им обнаруживаться. Использование ярлыков хотя и не может ничего гарантировать, но на практике неоднократно «защищало» сетевой каталог от скомпрометированной рабочей станции. В рассматриваемой системе ярлыки должны иметь следующие адреса объектов (в Windows-нотации):

```
\\samba.account.school34\shared
\\samba.account.school34\backup
\\samba.account.school34\archive
```

Также для быстрого доступа к индексному файлу уместно создать для него отдельный ярлык с адресом объекта:

```
file:///samba.account.school34/archive/index.html
```

Этот ярлык может быть создан не только в дополнение к ярлыку архива, но и вместо него, но это уже является вопросом комфорта пользователей и не более.

Что касается включения рабочих станций в группу `ACCOUNT`, то этот вопрос следует оставить на усмотрения системного администратора, обслуживающего рабочие станции бухгалтерии и установленное на них программное обеспечение. Сетевые каталоги останутся доступными в любом случае.

Замена узла сети бухгалтерии бухгалтерии

В случае замены рабочей станции или сетевого МФУ бухгалтерии на сервере следует выполнить такие правки:

Во-первых, в файле `/etc/network/neighbours` следует заменить канальный адрес клиента (MAC-адрес, `lladdr`).

Во-вторых, в файле `/etc/dnsmasq.conf` следует также заменить этот адрес в соответствующей строке `dhcp-host...`

В-третьих, следует актуализировать конфигурационный файл `/etc/nftables` межсетевого экрана. В частности, требуется заменить адреса канального и сетевого уровней выбывающего узла соответствующими адресами вновь устанавливаемого в переменных `CLIENT_LLADDRES`, `CLIENTS` и, возможно, `MFP_LLADDR` и `MFP` (если происходит замена МФУ) или `KRISTA_CLIENT` (если происходит замена именно этой рабочей станции).

Затем остаётся лишь перезапустить службы и удостовериться в их работоспособности:

```
root@server:/tmp# systemctl restart nftables
root@server:/tmp# systemctl restart networking
root@server:/tmp# systemctl restart dnsmasq
root@server:/tmp# systemctl status networking
...
root@server:/tmp# systemctl status dnsmasq
...
```

Разумеется, затем следует настроить изменённый узел: создать сетевые ярлыки, установить корневые сертификаты, настроить подключение к МФУ или установить на новую МФУ её сертификат. Всё это следует делать в соответствии с приведёнными выше инструкциями по первичной настройке аналогичных узлов сети.

Замена смартфона сотрудника бухгалтерии

В случае замены смартфона у сотрудника бухгалтерии следует выполнить следующие операции:

Во-первых, получить необходимые сведения о новом смартфоне серией команд вида (с точностью до адреса телефона на шине USB):

```
root@server:/# lsusb
root@server:/# udevadm info -a /dev/bus/usb/005/005
root@server:/# udevadm info -q all -n /dev/bus/usb/005/005
```

Затем на основании полученных данных следует внести правки в файлы **/etc/udev/rules.d/35-usb-filter.rules** и **/etc/udev/rules.d/77-net-alias.rules**.

Далее надо сообщить демону udev о необходимости перечитать правила работы при помощи команды:

```
root@server:/# udevadm control --reload-rules
```

Наконец, следует внести изменения в начальный загрузочный диск с тем, чтобы подключение смартфона корректно обрабатывалось даже на этапе загрузки системы. Это можно сделать командами:

```
root@server:/# PATH=$PATH:/sbin
root@server:/# update-initramfs -u
```

В иные подсистемы (например, в межсетевой экран) вносить изменения не требуется.

Замена узла ядра сети

В случае замены рабочей станции системного администратора, серверов виртуализации или системы «Аверс», а также главного шлюза на сервере бухгалтерии следует выполнить такие правки:

В файле **/etc/network/neighbours** следует заменить канальный адрес изменённого узла (MAC-адрес, lladdr).

Затем следует актуализировать конфигурационный файл **/etc/nftables** меж сетевого экрана. В частности, требуется заменить адреса канального (только для компьютера системного администратора) и сетевого уровней выбывающего узла соответствующими адресами вновь устанавливаемого в переменных ADMIN_LLADDR, ADMIN_HOST, AVERS_HOST и GATEWAY.

Затем остаётся лишь перезапустить службы и удостовериться в их работоспособности:

```
root@server:/tmp# systemctl restart nftables
root@server:/tmp# systemctl status nftables
root@server:/tmp# systemctl restart networking
root@server:/tmp# systemctl status networking
```

Обслуживание дискового массива

Условно можно выделить три задачи обслуживания дискового массива на данной системе:

- контроль текущего состояния массива

- доступ к данным при повреждении одного из дисков
- замена повреждённого диска

Для отслеживания состояния массива существует несколько способов.

Во-первых в системе непрерывно работает systemd-служба mdmonitor, описанная при настройке массива. Она уведомляет root@localhost о проблемах средствами электронной почты, работающей внутри системы.

Во-вторых, вывод команды

```
root@server:/# cat /proc/mdstat
Personalities: [raid1] [linear] [multipath] [raid0] [raid6] [raid5] [raid4] [raid10]
md0 : active raid1 sdb[0] sdc[1]
      488254464 blocks super 1.2 [2/2] [UU]
      bitmap: 0/4 pages [0KB], 65536KB chunk
```

```
unused devices: <none>
```

отображает текущее состояние всех массивов в системе. Для просмотра непрерывно меняющейся (например при синхронизации дисков) информации из этого файла можно использовать и следующую команду

```
root@server:/# watch cat /proc/mdstat
```

Подробные сведения о массиве можно получить так:

```
root@server:/# mdadm --detail /dev/md0
```

Аналогично, подробные данные о диске, входящем в массив, извлекаются так:

```
root@server:/# mdadm --examine /dev/sdb
```

Проверку целостности массива можно выполнить при помощи следующих трёх последовательных команд:

```
root@server:/# echo check > /sys/block/md0/md/sync_action
root@server:/# watch cat /proc/mdstat
root@server:/# cat /sys/block/md0/md/mismatch_cnt
```

Первая команда инициирует проверку. При этом на рассматриваемой системе проверка заняла более одного часа. Вторая команда позволяет следить за ходом этой проверки, третья – показывает результат.

В случае отказа одного из дисков вывод команды

```
root@server:/# mdadm --detail /dev/md0
```

покажет неактивный (inactive) массив с одним работающим устройством, на котором присутствует суперблок. При этом в случае отказа диска на этапе загрузки система будет в течение некоторого времени пытаться собрать массив, а затем перейдёт в аварийный режим.

Если требуется получить доступ к данным массива до восстановления его состава, то необходимо пересобрать и заново смонтировать массив. Эти операции могут быть выполнены следующей серией команд:

```
root@server:/# mdadm --stop /dev/md0
root@server:/# mdadm /dev/md0 --assemble --run --verbose
root@server:/# mount /dev/md0 /srv/samba/archive/
```

Для восстановления состава массива следует выключить систему, извлечь повреждённый и установить запасной носитель, включить систему. После чего по выводу команды

```
root@server:/# mdadm --detail /dev/md0
```

можно сделать выводы, что новый диск включён в состав массива, но не работает, т. к. на нём отсутствует суперблок. Диск оказался включённым в состав массива, т. к. получил в системе то же самое имя (например, /dev/sdb), что и вышедший из строя диск, поскольку подключён по той же самой шине к тому же самому разъёму на материнской плате, имеет тот же размер и модель. Поэтому следует размонтировать и остановить массив, а затем ввести новый диск в его состав.

```
root@server:/# umount /dev/md0
root@server:/# mdadm --stop /dev/md0
root@server:/# mdadm --add /dev/sdb
```

После этого при помощи команды

```
root@server:/# mdadm --detail /dev/md0 --verbose
```

убедиться, что в массиве уже два работающих диска, вновь добавленный синхронизируется с исправным. Дождавшись окончания синхронизации

```
root@server:/# watch mdadm --detail /dev/md0 --verbose
```

остаётся лишь смонтировать массив или перезагрузить систему.

Следует пояснить, почему имеющийся в наличии запасной диск не был установлен в систему и указан при настройке RAID как запасной. Такое решение было принято на том основании, что используемые в массиве диски находятся в хорошем состоянии, рабочая нагрузка на них сравнительно невелика, следовательно вероятность их отказа незначительна. Запасной диск может не понадобиться на протяжении всего срока службы сервера, а установка его изначально будет бесцельно расходовать его ресурс и электроэнергию. В то же время, регулярный мониторинг сервера системным администратором, возможность краткосрочного вывода сервера из эксплуатации и беспрепятственный физический доступ администратора к нему позволяют своевременно выявить необходимость замены диска и провести её.

Замена устаревшего ключа локального репозитория

По истечении срока действия ключа локального репозитория его необходимо заменить. Для этого достаточно добавить новый ключ ровно тем же способом, что при первоначальной настройке. Но для поддержания чистоты системы совсем не лишним будет удалить старый ключ. Следующий фрагмент терминального сеанса (с некоторыми сокращениями ответов системы) демонстрирует необходимые действия:

```
root@server:/# apt-key list
/etc/apt/trusted.gpg
-----
pub   rsa3072 2019-08-20 [SC] [просрочен с: 2021-08-19]
      FF63 9F36 C9A5 DE8B 16AF  F1E6 5647 BD17 421C B415
```

```

uid          [   просрочен   ] maintainer <maintainer@localhost>

...
root@server:/# apt-key del "FF63 9F36 C9A5 DE8B 16AF  F1E6 5647 BD17 421C B415"
OK
root@server:/# cd /tmp && wget http://debian.service.school34/ppa/repository_key.asc
...
root@server:/tmp# apt-key add repository_key.asc
OK
root@server:/tmp# apt-key list
/etc/apt/trusted.gpg
-----
pub   rsa4096 2021-10-08 [SC] [   годен до: 2031-10-06]
      C395 1533 5648 35F7 8A0F  C094 057B A98A E75D EEFA
uid          [ неизвестно ] maintainer <debmaintainer@mail.service.school34>
sub   rsa4096 2021-10-08 [E] [   годен до: 2031-10-06]
...
root@server:/tmp# apt-get update
Пол:1 http://debian.service.school34/buster buster InRelease [122 kB]
Пол:2 http://debian.service.school34/security buster/updates InRelease [65,4 kB]
Игн:3 http://debian.service.school34/ppa buster InRelease
Пол:4 http://debian.service.school34/ppa buster Release [1 655 B]
Пол:5 http://debian.service.school34/ppa buster Release.gpg [833 B]
Пол:6 http://debian.service.school34/ppa buster/main amd64 Packages [4 148 B]
Чтение списков пакетов... Готово

```