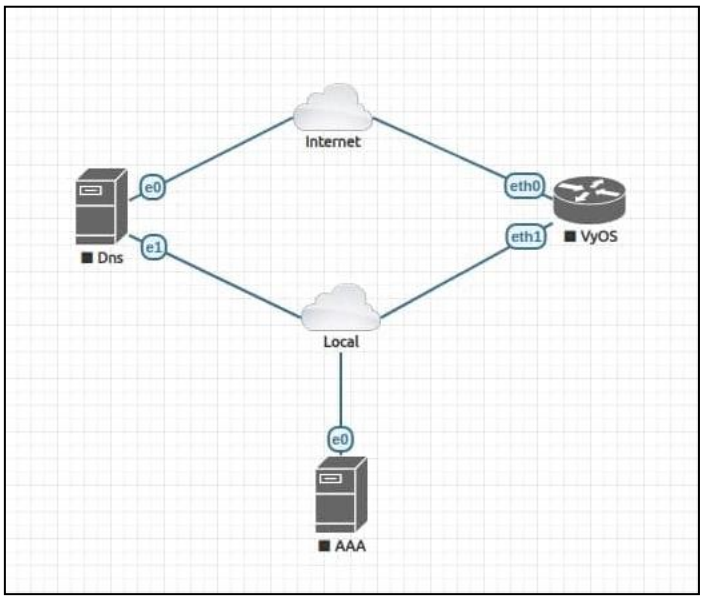
Курс «Основные сервисы на Linux для предприятия».

Методические указания по выполнению лабораторной работы № 2.

Автор курса: Павел Семенец
Автор методического пособия: Антон Трифонцов

1. Условия лабораторной работы.

Необходимо реализовать следующую схему на гипервизоре KVM:



Для этого понадобится три виртуальные машины (ВМ):
1) VyOS: ОС VyOS (Debian 10), Процессор 2, Память 512 MB, Диск SCSI 2 ГБ; Сеть NIC 2; Контроллер Virtio SCSI;
2) DNS: ОС Ubuntu 20.04 (Debian 10), Процессор 2, Память 1 ГВ, Диск SCSI 8 ГБ; Сеть NIC 2; Контроллер Virtio SCSI;
3) AAA: ОС Ubuntu 20.04 (Debian 10), Процессор 2, Память 1 ГБ, Диск SCSI 8 ГБ; Сеть NIC 1; Контроллер Virtio SCSI.

У ВМ VyOS и DNS есть выход в интернет, у ВМ AAA выход только в локальную сеть.

2. Создание и установка новых ВМ.

Установка гипервизора KVM описана по *ссылке*.
Предполагается, что сети vm-net и vm-int уже были созданы в предыдущей работе.
XML-файл сети vm-net:

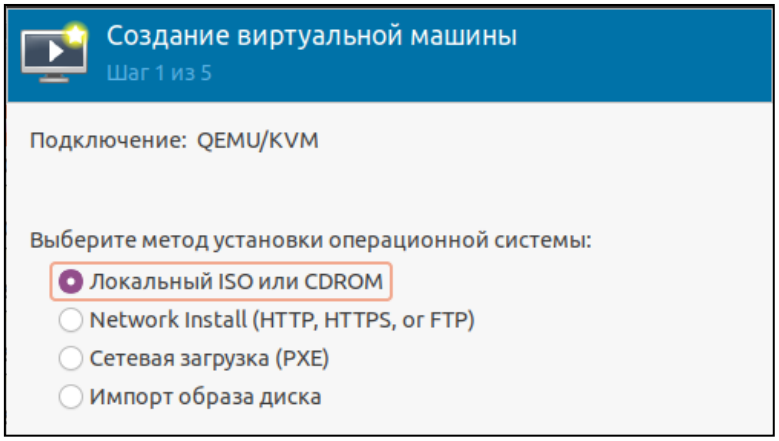
```
<network>
  <name>vm-net</name>
  <forward mode="nat">
    <nat>
      <port start="1024" end="65535"/>
    </nat>
  </forward>
  <bridge name="vm-net0" stp="on" delay="0"/>
  <ip address="10.100.10.1" netmask="255.255.255.192">
    <tftp root="/srv/tftp"/>
    <dhcp>
      <range start="10.100.10.12" end="10.100.10.62"/>
      <bootp file="pxelinux.0" server="10.100.10.1"/>
    </dhcp>
  </ip>
</network>
```

XML-файл сети vm-int:

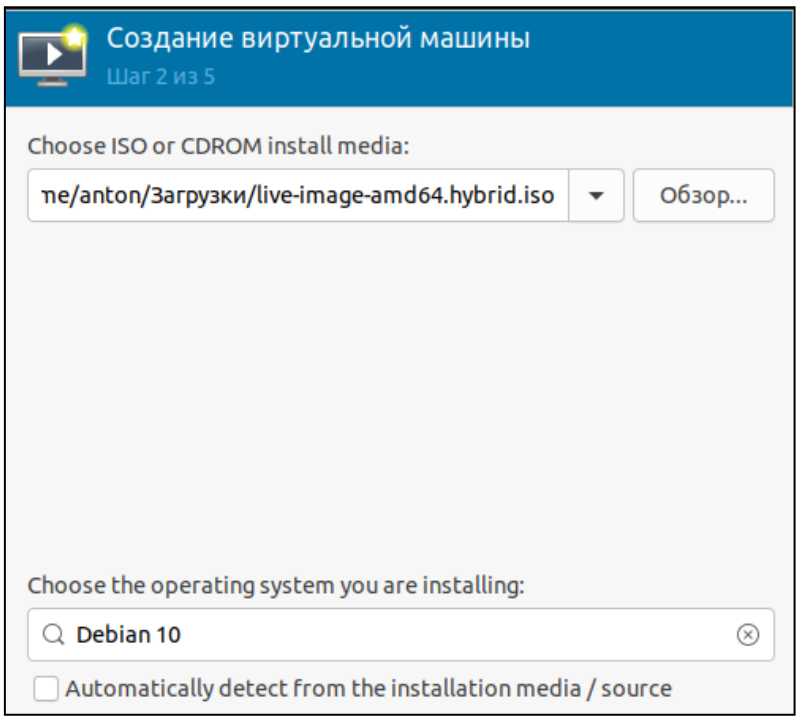
```
<network>
  <name>vm-int</name>
  <bridge name="virbr0" stp="on" delay="0"/>
  <domain name="vm-int"/>
</network>
```

2.1. Создание ВМ VyOS.

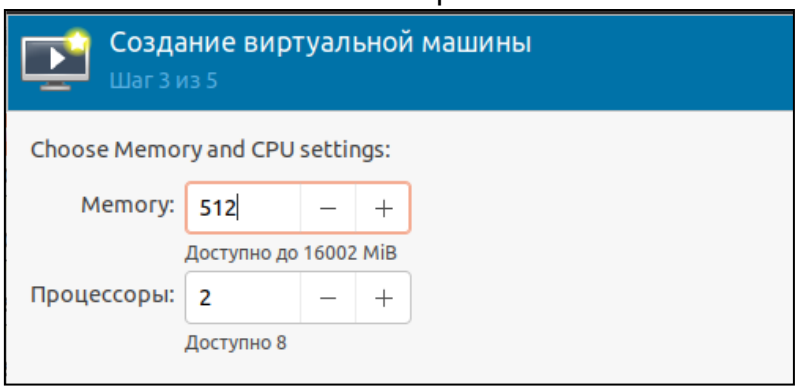
Шаг 1. Выбираем метод установки ОС через Локальный ISO.



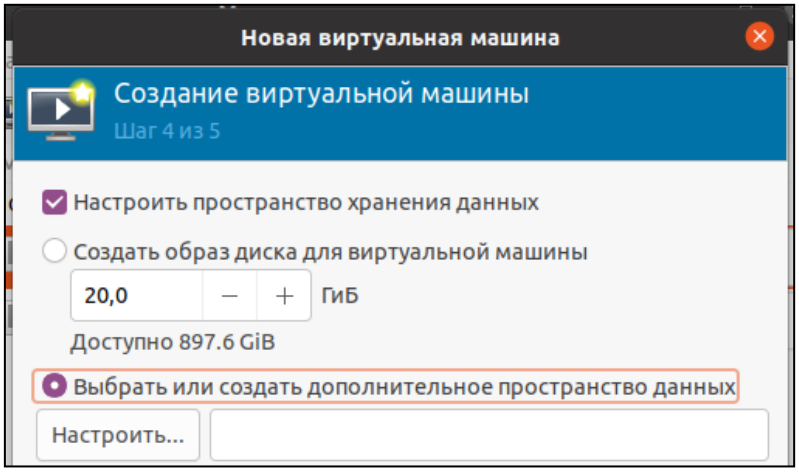
Шаг 2. Указываем путь до скаченного *образа* и ОС – Debian 10.



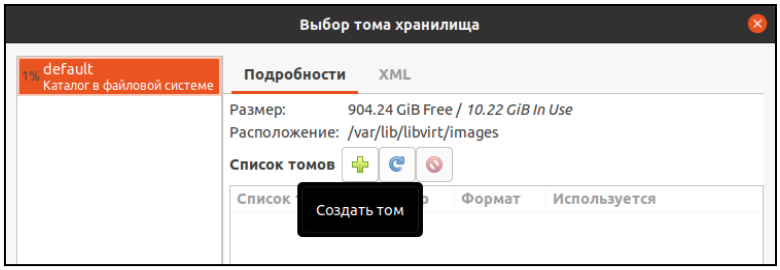
Шаг 3. Указываем объем оперативной памяти и количество процессоров.



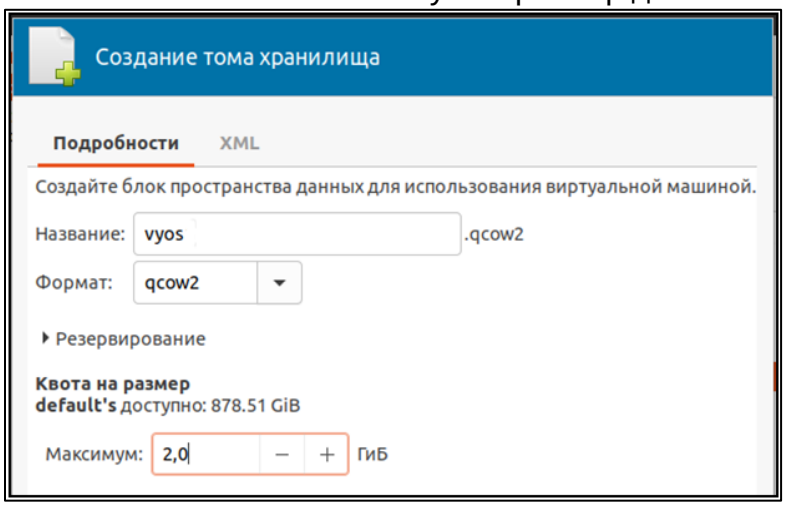
Шаг 4. Выбираем создать дополнительное пространство данных.



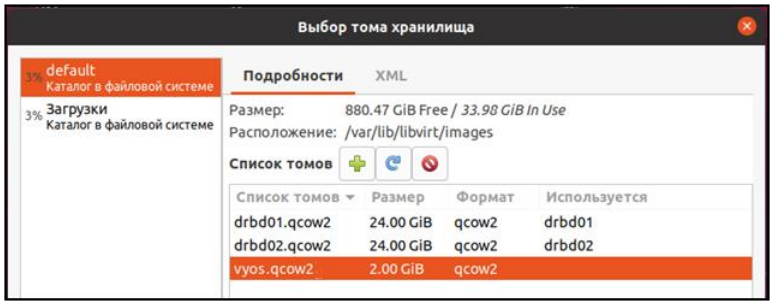
Создаем новый том.



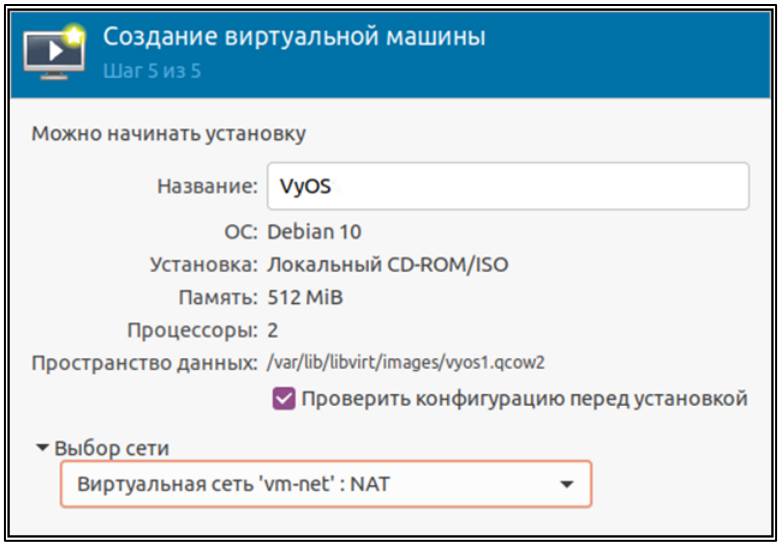
Указываем название тома vyos и размер диска 2 ГБ.



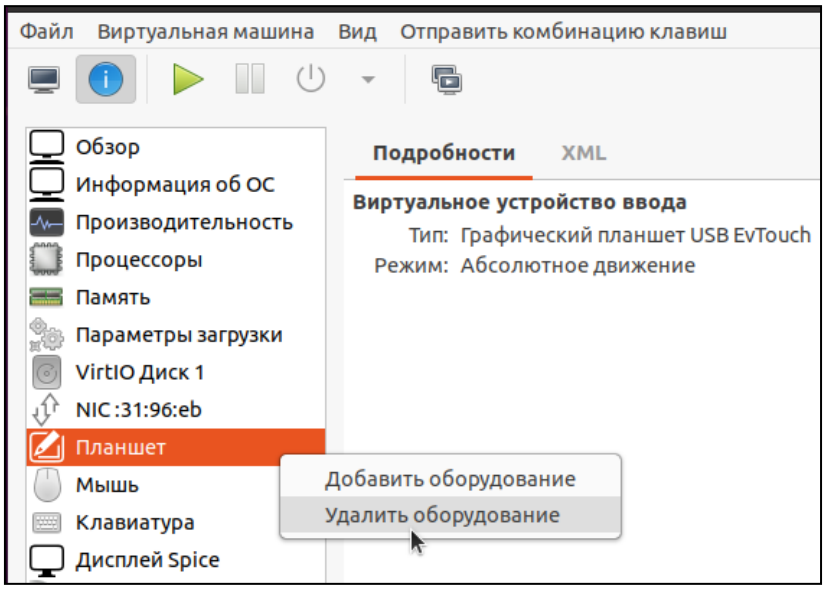
Выбираем созданный том vyos.qcow2.



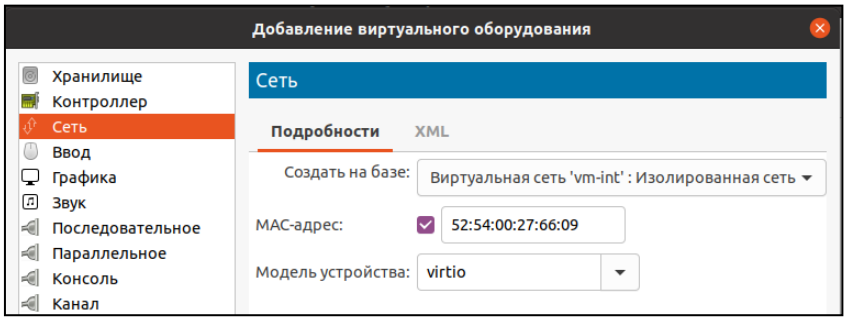
Шаг 5. Указываем название ВМ – VyOS. Ставим галочку напротив Проверить конфигурацию перед установкой.



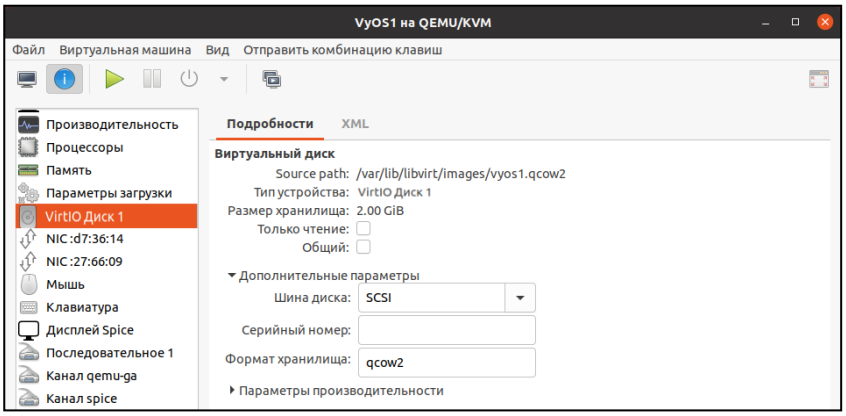
В настройках ВМ удаляем Планшет и Sound ich9.



Добавляем еще один сетевой контроллер на базе виртуальной сети ‘vm-int’: Изолированная сеть.



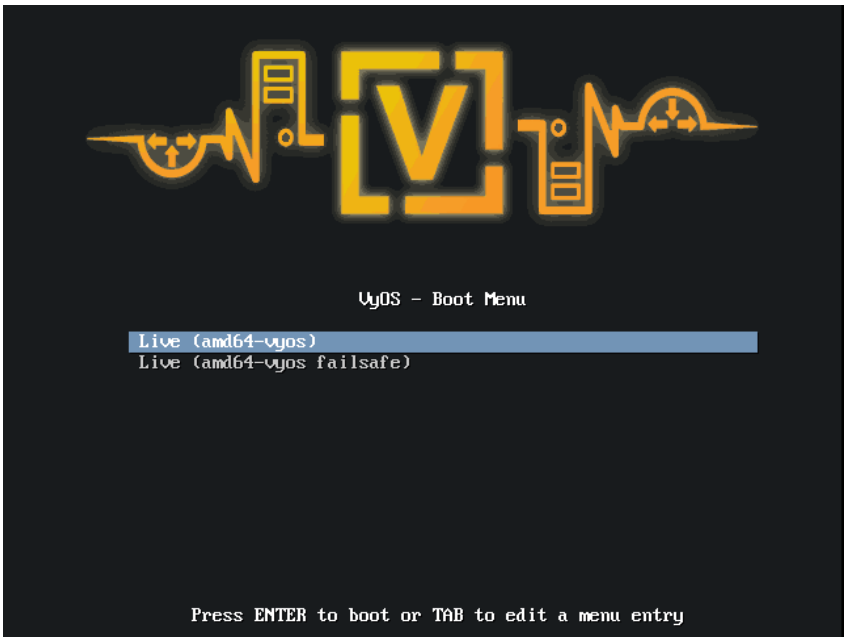
Для оборудования VirtIO Диск 1 устанавливаем шину SCSI. Нажимаем Применить.



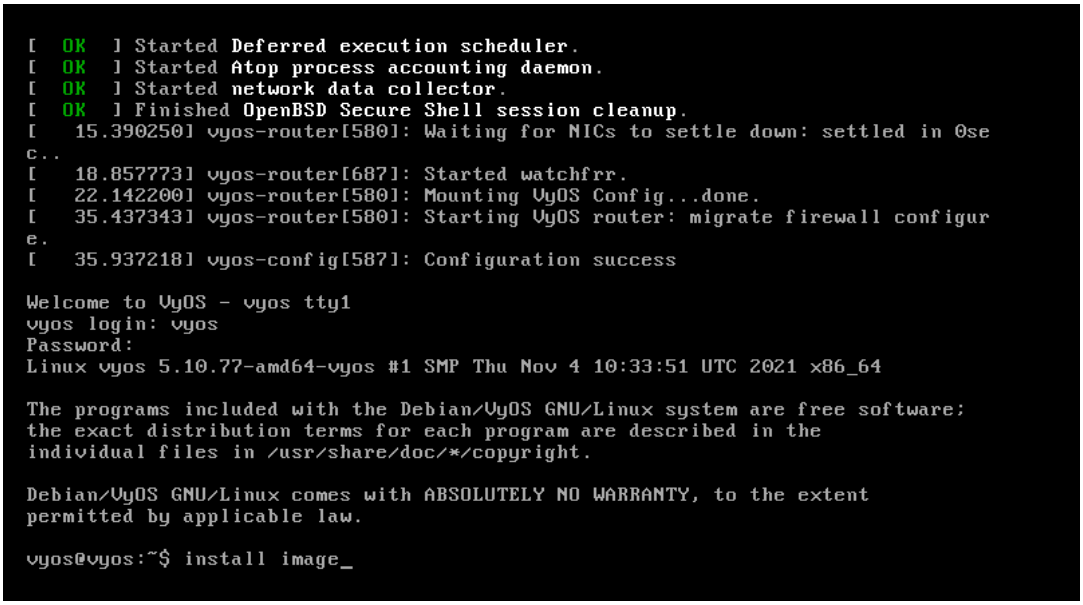
Настройка закончена, запускаем ВМ.

2.1.1. Установка операционной системы ВМ VyOS.

После запуска ВМ попадаем в меню выбора режима загрузки, выбираем первый.

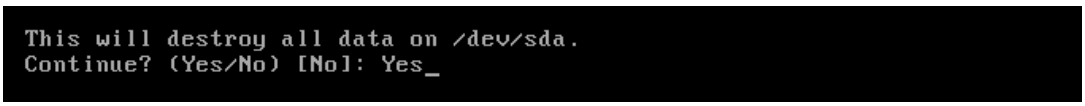


Логин и пароль: vyos.



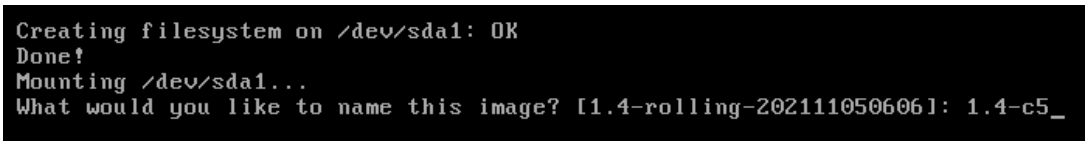
Устанавливаем образ на диск:
`vyos@vyos:~$ install image`

Далее везде нажимаем Enter.

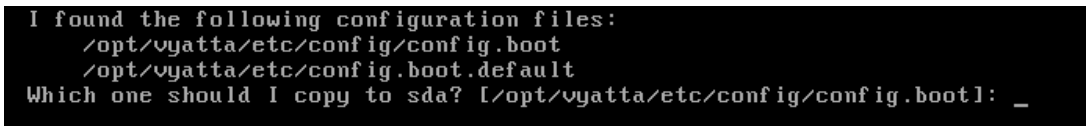


Пишем Yes, нажимаем Enter.

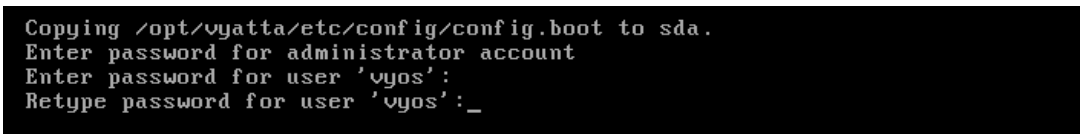
Нажимаем Enter.



Вводим название образа: **`1.4-c5`**. Нажимаем Enter.



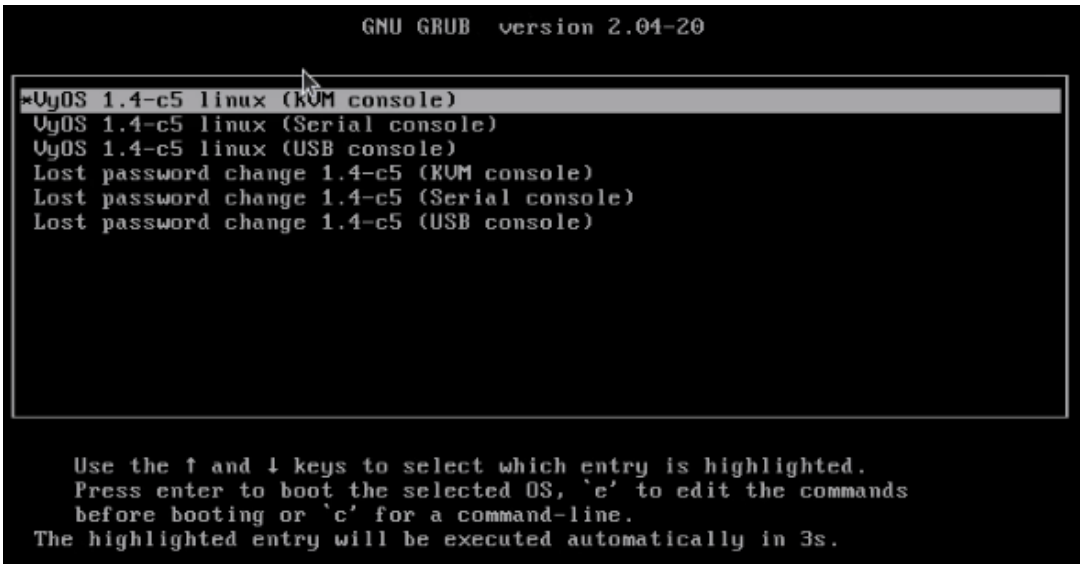
Нажимаем Enter.



Вводим пароль vyos и подтверждаем его.

Нажимаем Enter.

Установка завершена. Перезагружаем ВМ.

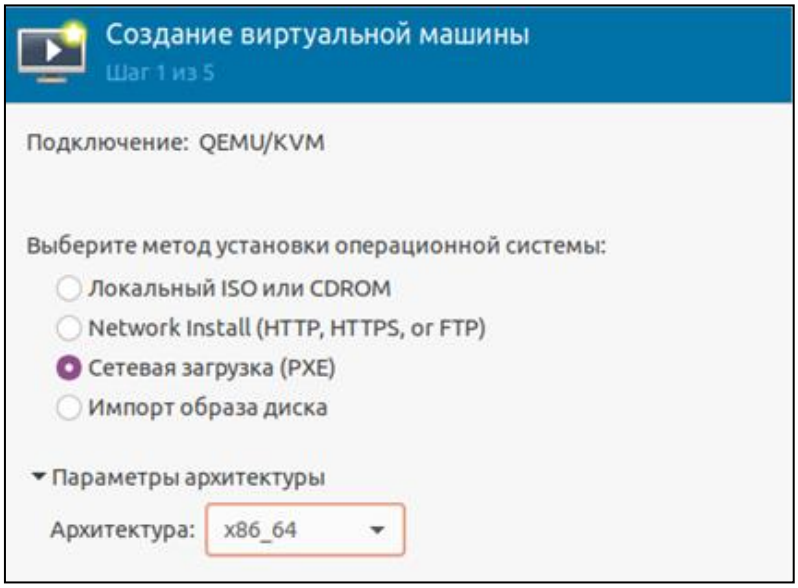


Выбираем первый образ и загружаемся.

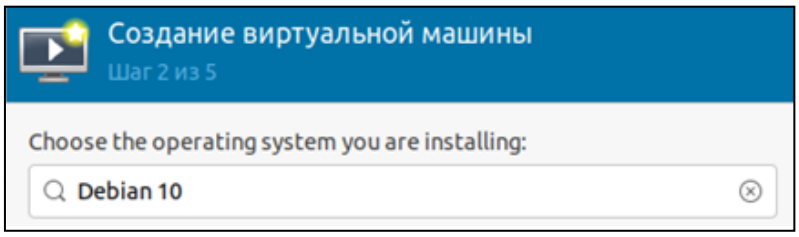
2.2. Создание ВМ DNS.

Создадим ВМ DNS со следующими параметрами:
- ОС Ubuntu 20.04 (Debian 10);
- оперативная память 1 ГБ, количество процессоров 2;
- диск 8 ГБ (диск разбить на следующие разделы: part1 - 1M, ef02, grub; part2 - 512M, 8200, swap; part3 - все остальное пространство, 8300, root);
- сетевой адаптер NIC1 (vm-net, внешняя), NIC2 (vm-int, изолированная).

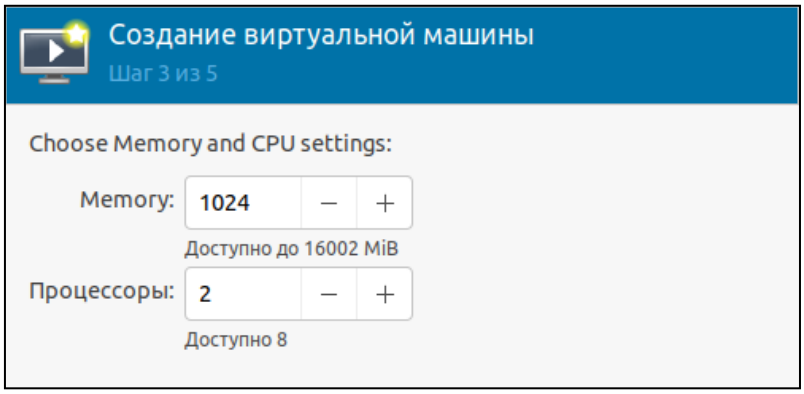
Шаг 1. Выбираем загрузку по сети.



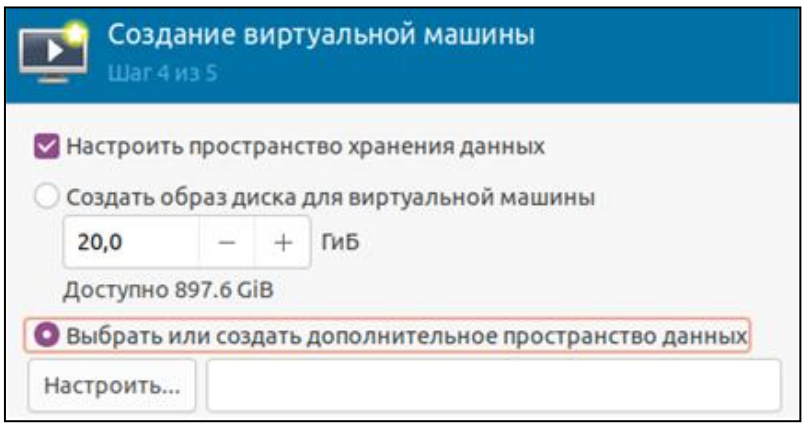
Шаг 2. Выбираем операционную систему Debian 10.



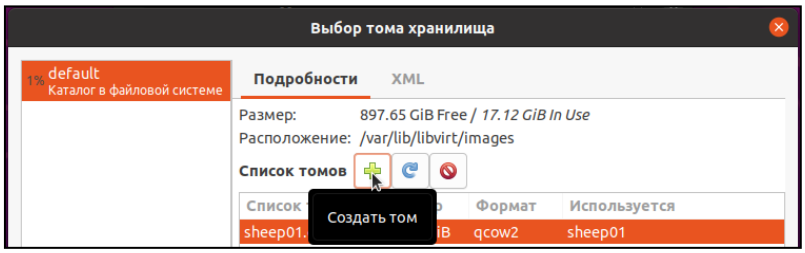
Шаг 3. Устанавливаем размер оперативной памяти – 1 ГБ и количество процессоров – 2.



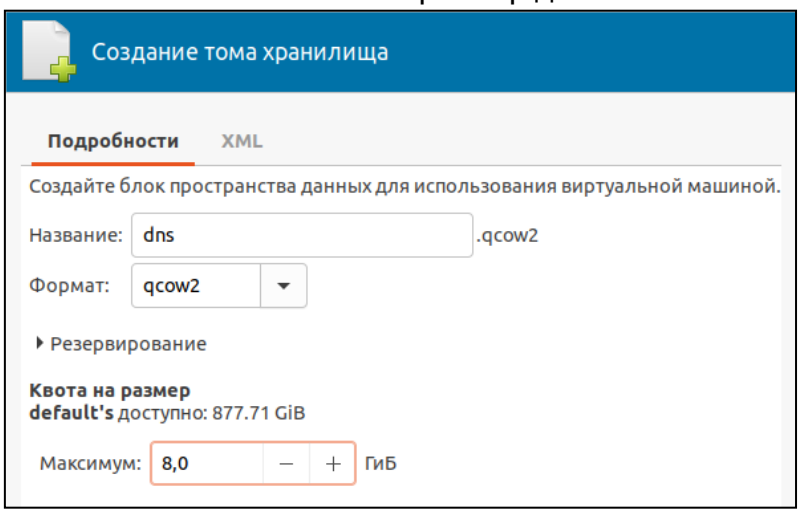
Шаг 4. Выбираем дополнительное пространство данных



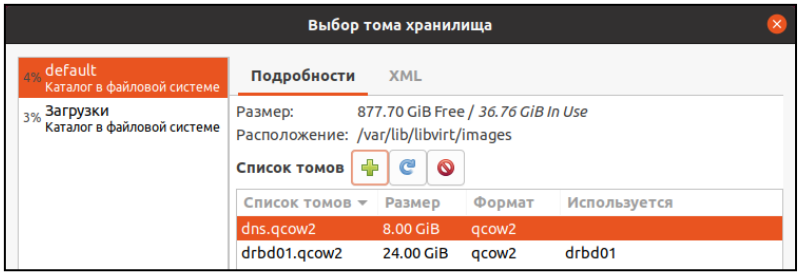
Создаем новый том



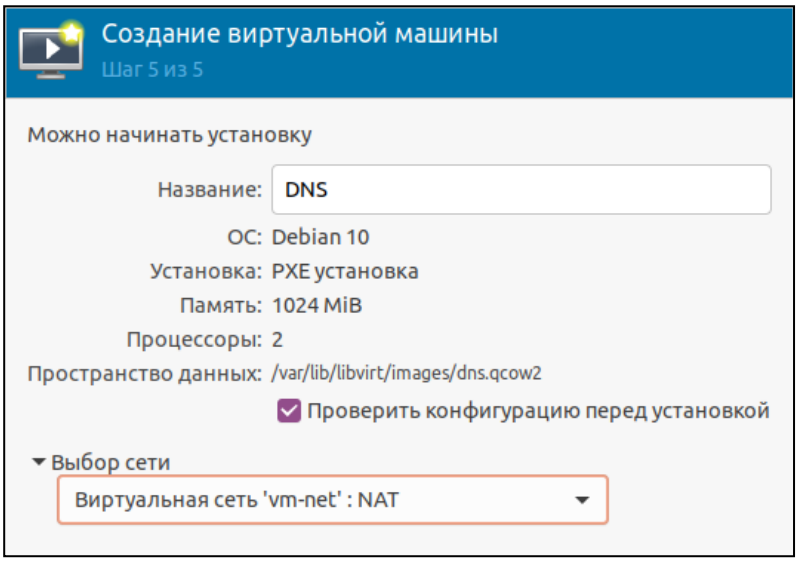
Указываем название: dns и размер диска 8 ГБ.



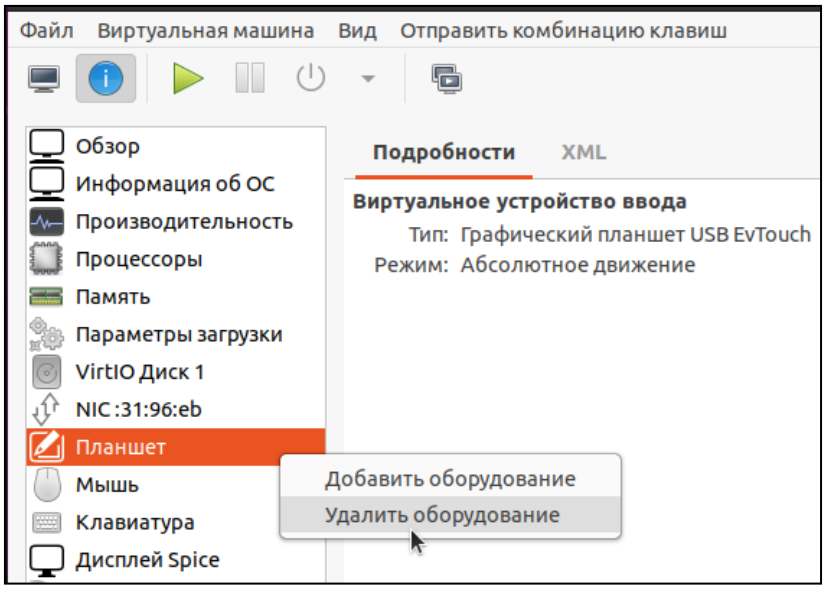
Выбираем созданный том dns.qcow2.



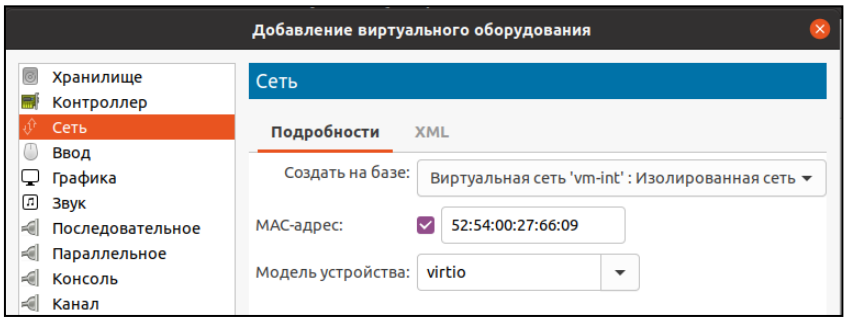
Шаг 5. Указываем название ВМ – DNS. Ставим галочку напротив Проверить конфигурацию перед установкой.



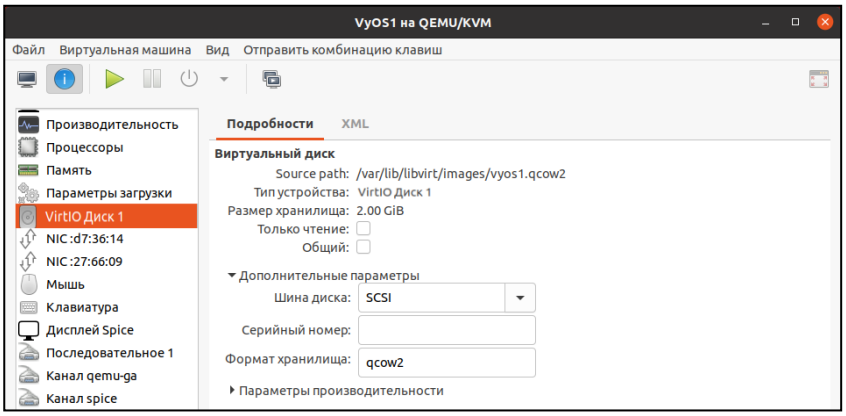
В настройках ВМ удаляем Планшет и Sound ich9.



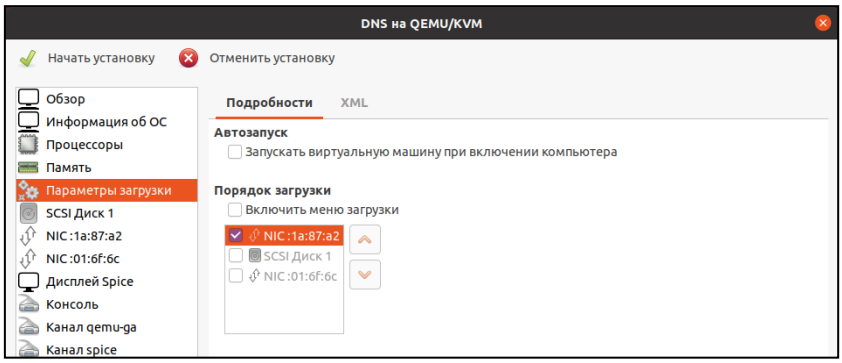
Добавляем еще один сетевой контроллер на базе виртуальной сети ‘vm-int’: Изолированная сеть.



Для оборудования VirtIO Диск 1 устанавливаем шину SCSI. Нажимаем Применить.



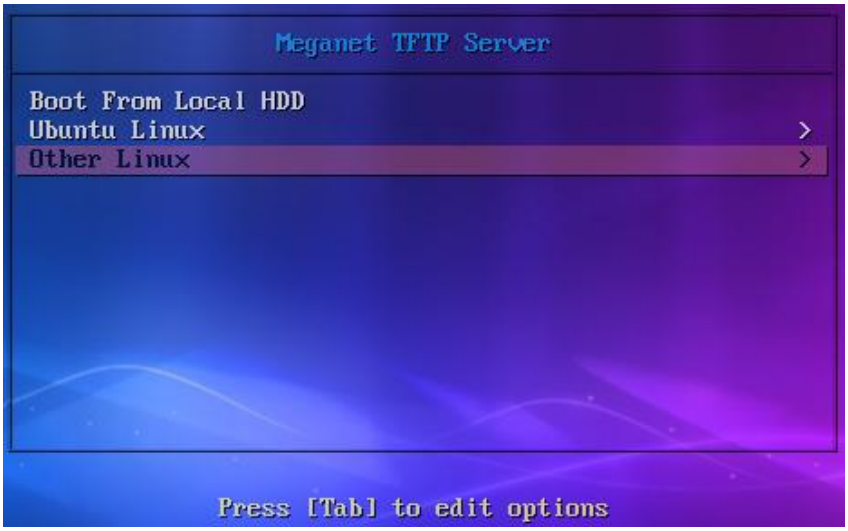
В Параметрах загрузки выставляем первым сетевой адаптер NIC1 (vm-net, внешняя).



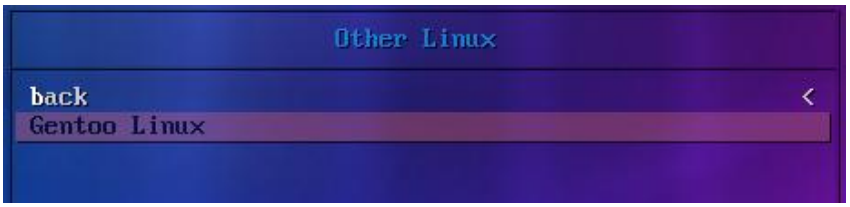
Настройка закончена, запускаем ВМ.

2.2.1. Установка операционной системы ВМ DNS.

После запуска ВМ, происходит загрузка по сети из образа oVirt-toolsSetup.iso (необходимо скачать заранее). Выбираем Other Linux.



Затем Gentoo Linux.



После загрузки Gentoo Linux переходим к созданию разделов на диске.
Набираем в консоле: **# gdisk /dev/sda**

```
livecd ~ # gdisk /dev/sda
GPT fdisk (gdisk) version 1.0.3

Partition table scan:
  MBR: not present
  BSD: not present
  APM: not present
  GPT: not present

Creating new GPT entries.
```

Далее команда o – удаляем все разделы (если таковые были).
Команда n – создаем новый раздел:

- первый раздел grub (Enter, Enter, +1M, ef02);
- второй раздел swap (Enter, Enter, +512M, 8200);
- третий раздел root (Enter, Enter, Enter).

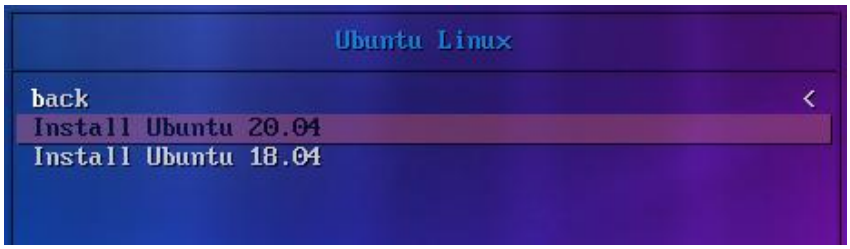
```
Command (? for help): n
Partition number (1-128, default 1):
First sector (34-16777182, default = 2048) or {+-}size{KMGTP}:
Last sector (2048-16777182, default = 16777182) or {+-}size{KMGTP}: +1M
Current type is 'Linux filesystem'
Hex code or GUID (L to show codes, Enter = 8300): ef02
Changed type of partition to 'BIOS boot partition'

Command (? for help): n
Partition number (2-128, default 2):
First sector (34-16777182, default = 4096) or {+-}size{KMGTP}:
Last sector (4096-16777182, default = 16777182) or {+-}size{KMGTP}: +512M
Current type is 'Linux filesystem'
Hex code or GUID (L to show codes, Enter = 8300): 8200
Changed type of partition to 'Linux swap'

Command (? for help): n
Partition number (3-128, default 3):
First sector (34-16777182, default = 1052672) or {+-}size{KMGTP}:
Last sector (1052672-16777182, default = 16777182) or {+-}size{KMGTP}:
Current type is 'Linux filesystem'
Hex code or GUID (L to show codes, Enter = 8300):
Changed type of partition to 'Linux filesystem'

Command (? for help):
```

Далее записываем произведенные действия - w, подтверждаем - Y и перезагружаемся.

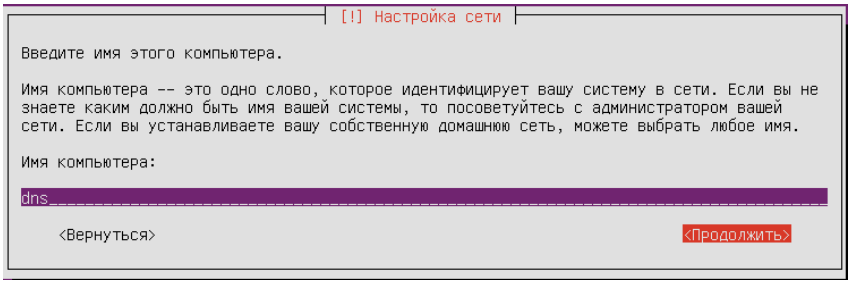
```
Command (? for help): w

Final checks complete. About to write GPT data. THIS WILL OVERWRITE EXISTING
PARTITIONS!!

Do you want to proceed? (Y/N): Y
OK; writing new GUID partition table (GPT) to /dev/sda.
The operation has completed successfully.
livecd ~ # reboot_
```

В меню загрузки выбираем Ubuntu Linux.

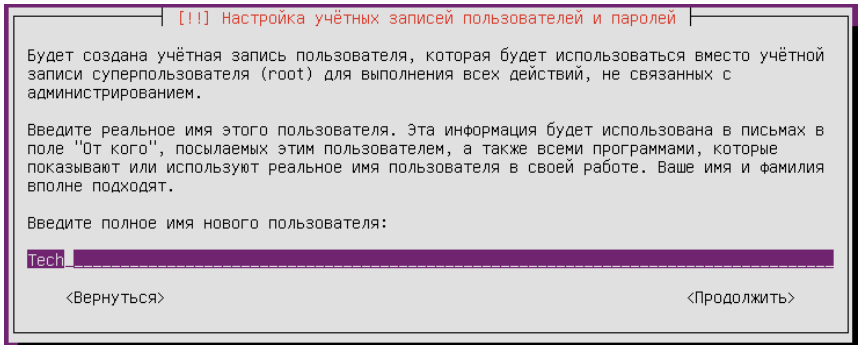
Затем Install Ubuntu 20.04



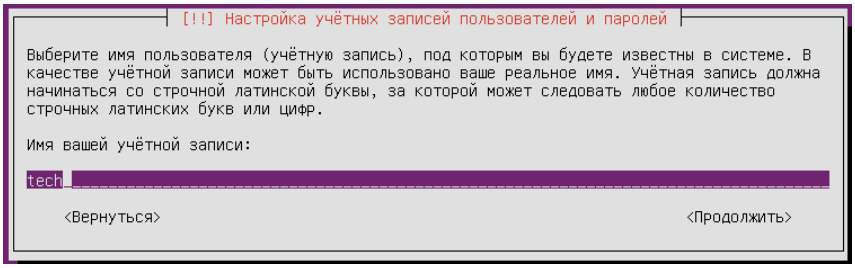
Устанавливаем Ubuntu 20.04.
Указываем имя компьютера – dns.



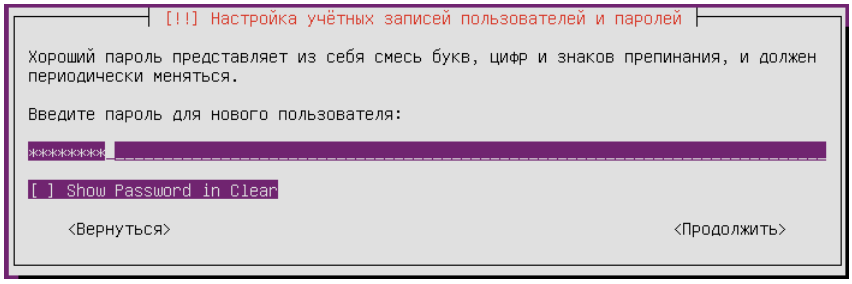
Вводим полное имя пользователя - Tech:



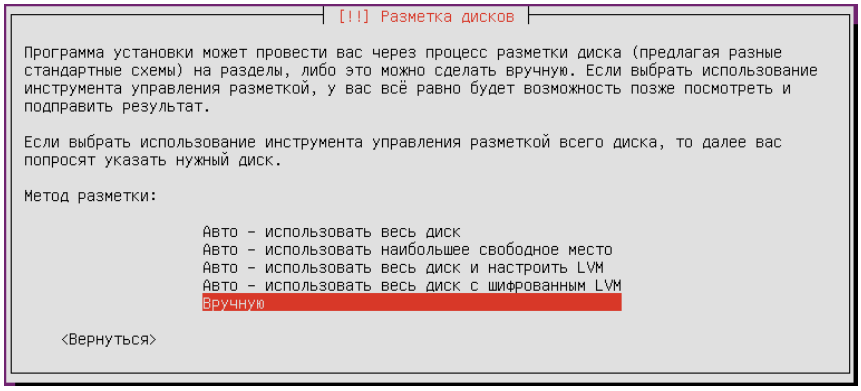
И имя учетной записи - tech:



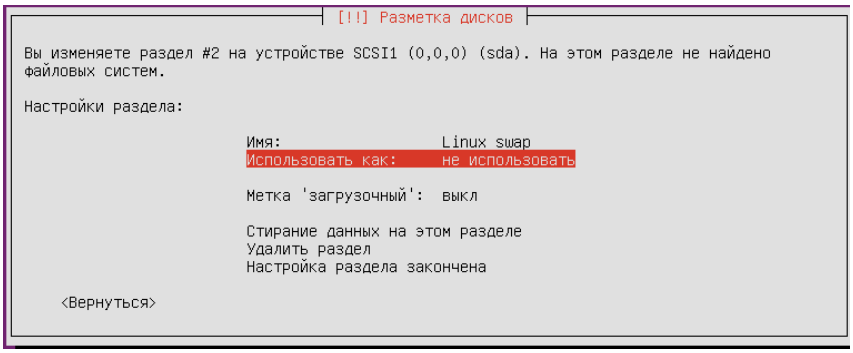
Вводим пароль для учетной записи:

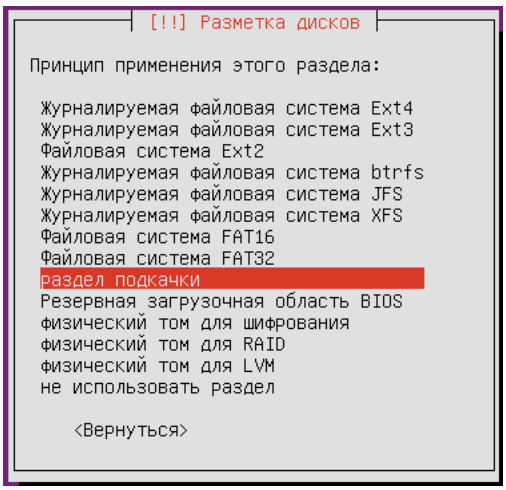


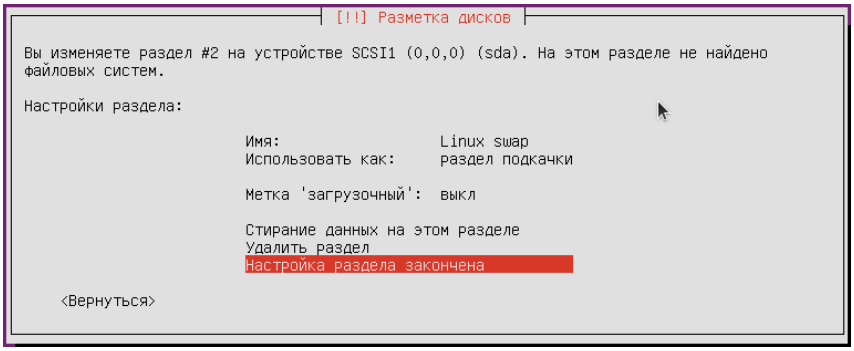
Разметку дисков делаем вручную:



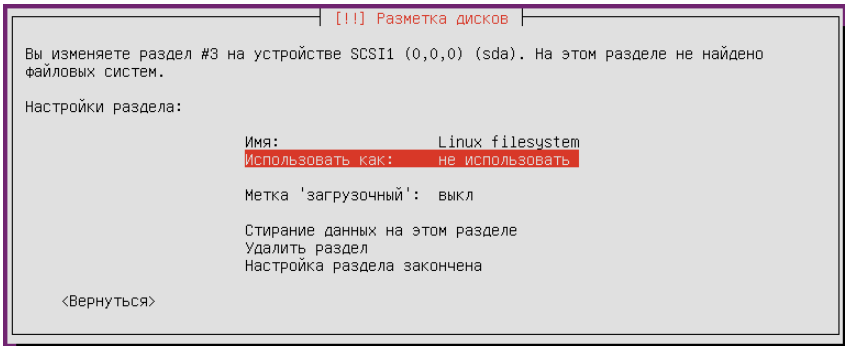
Первый раздел не трогаем, начинаем настройку со второго раздела:

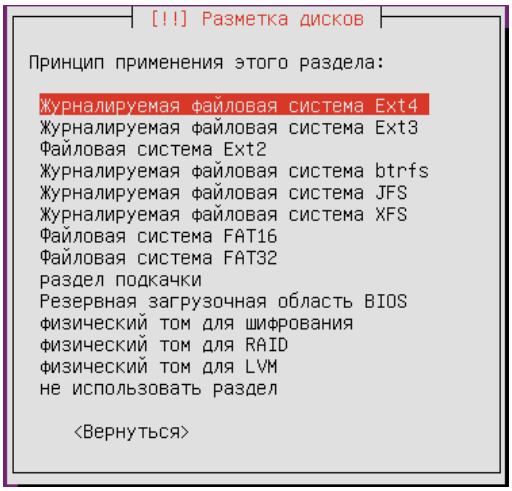


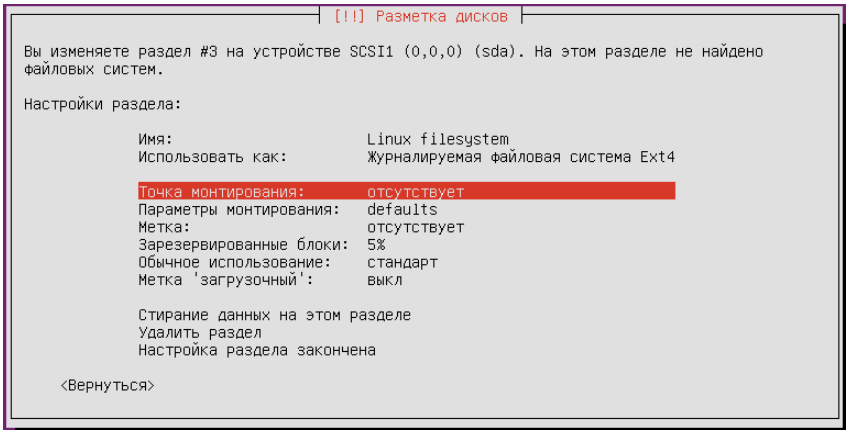


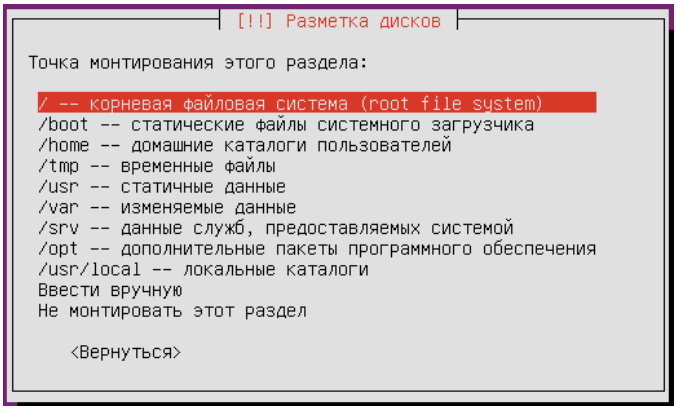


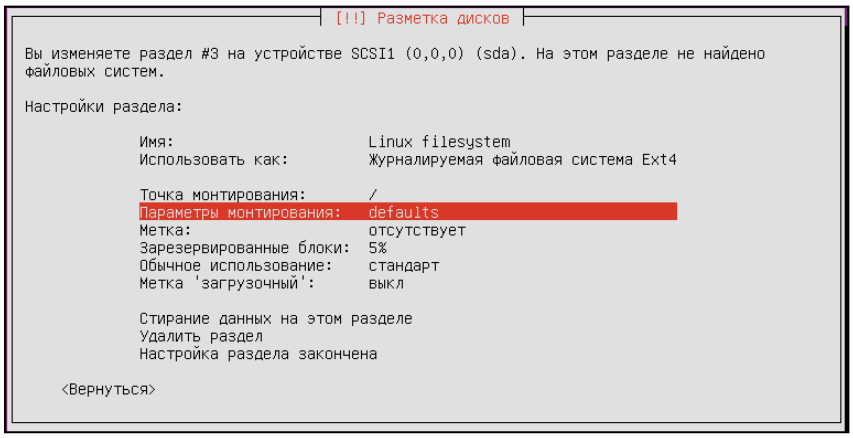
Настраиваем третий раздел:



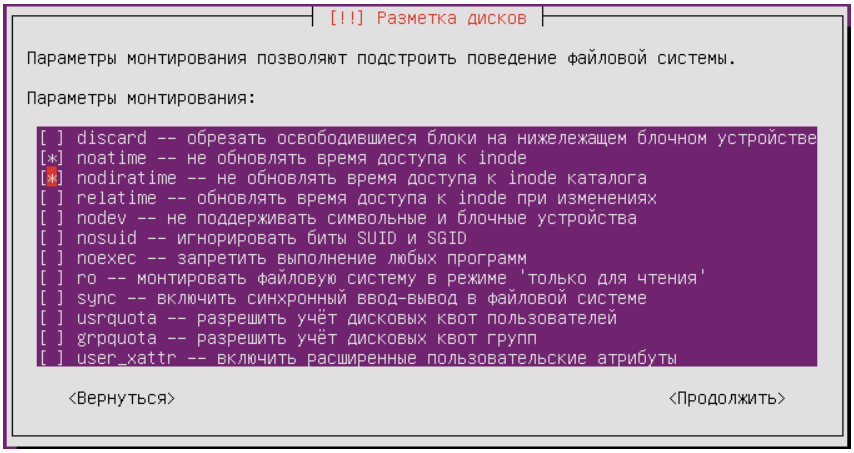


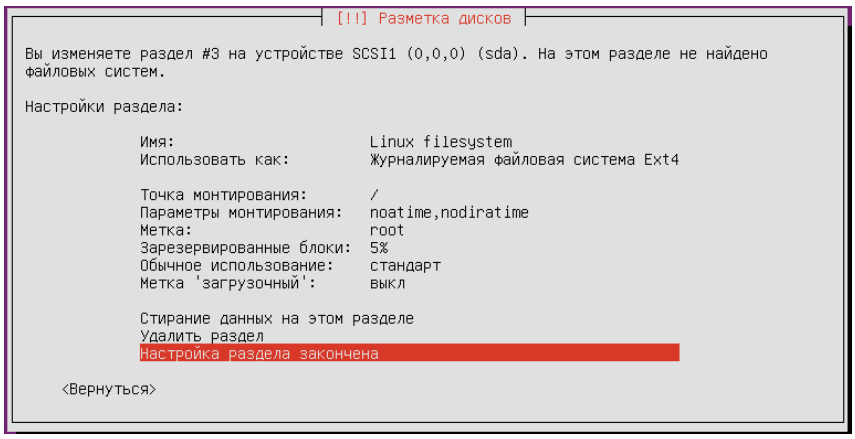


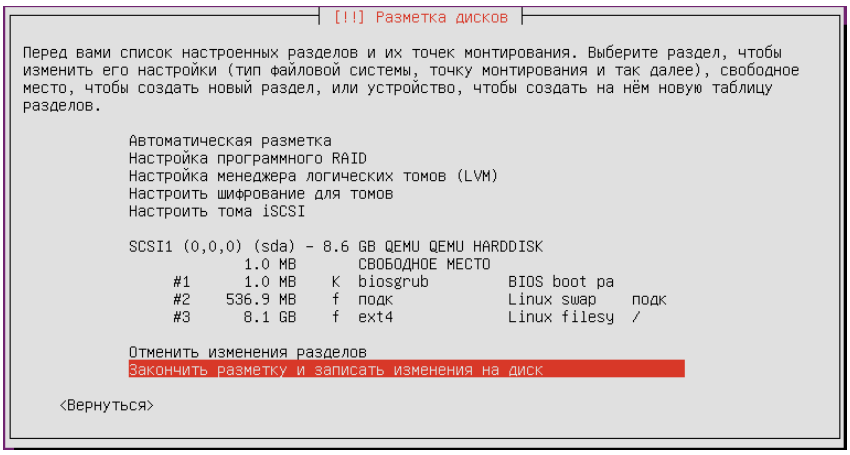




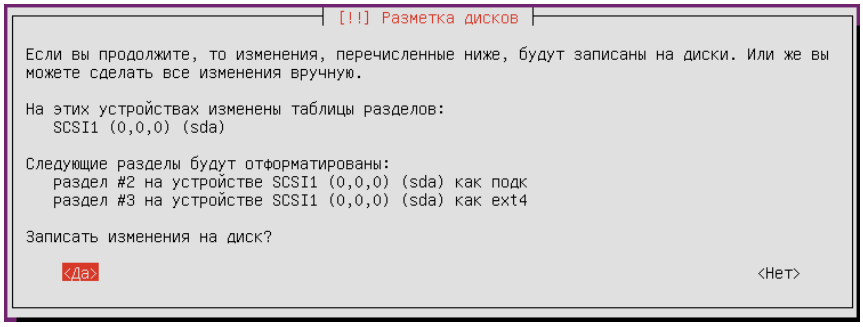
В параметрах монтирования отмечаем noatime и nodiratime

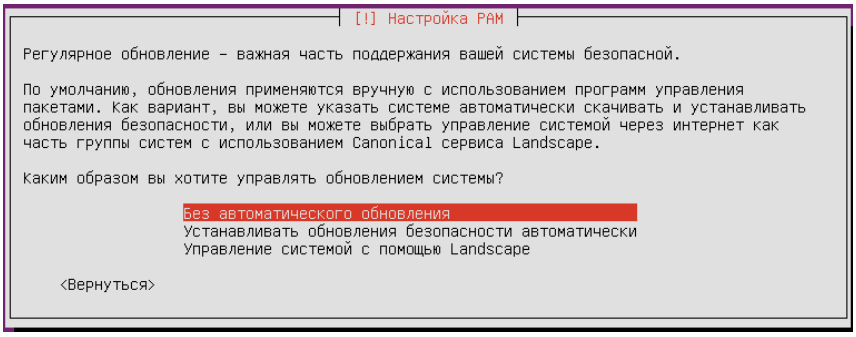




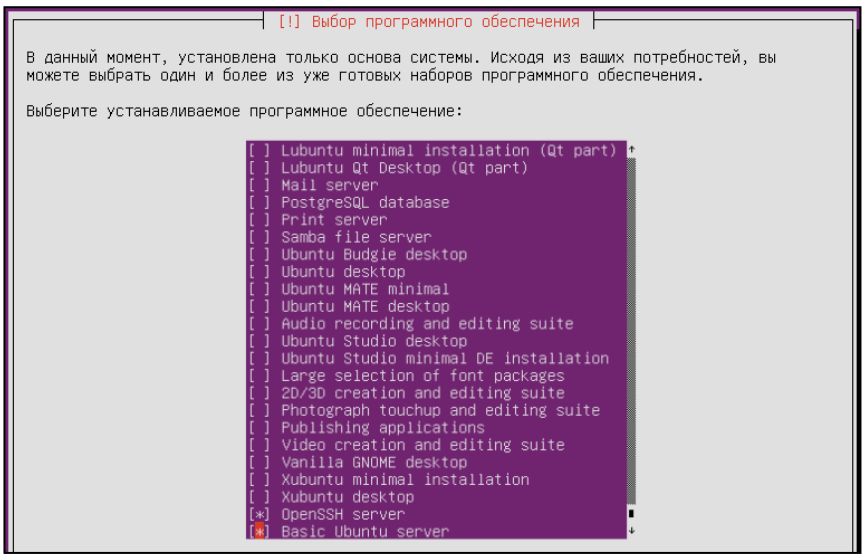


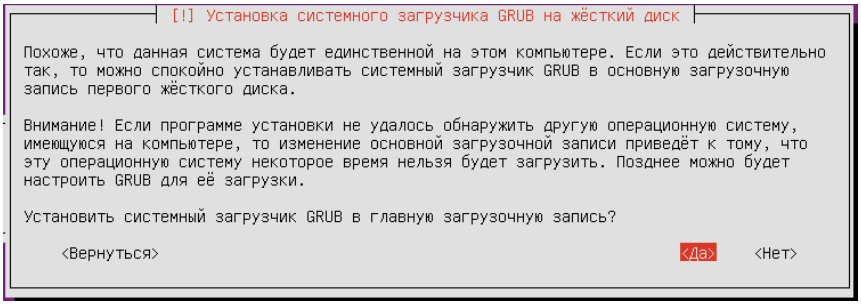
Подтверждаем изменения. Разметка дисков закончена.

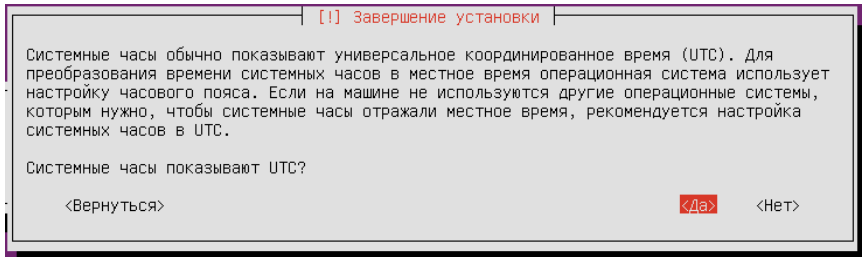


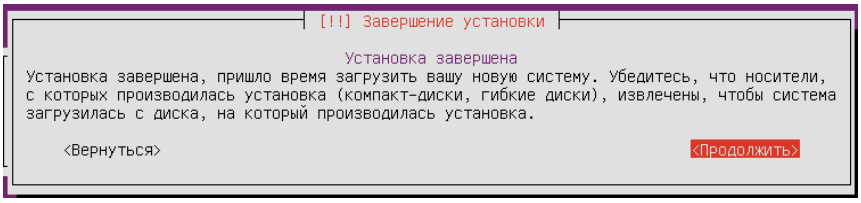


Отмечаем для установки OpenSSH server и Basic Ubuntu server:





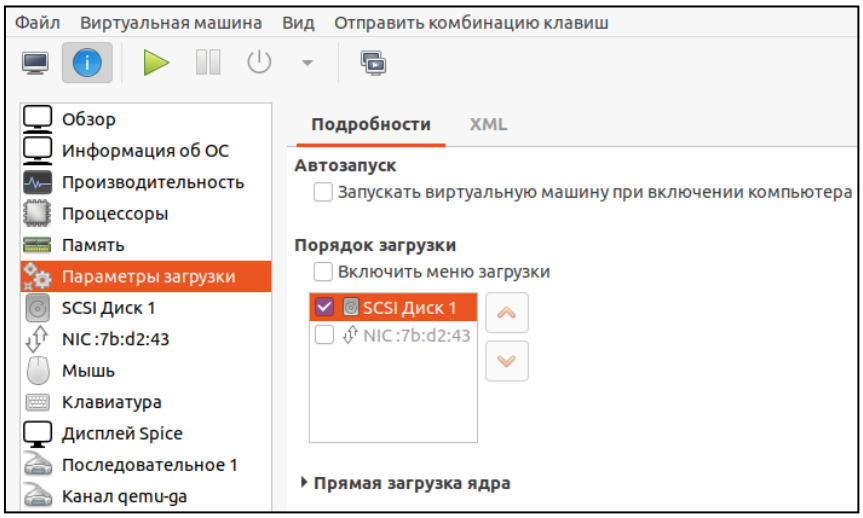




Установка ОС завершена. После нажатия кнопки Продолжить, система перезагрузится.

Принудительно выключаем ВМ. Заходим в настройки ВМ.

В Параметрах загрузки устанавливаем первым SCSI Диск 1. Нажимаем кнопку Применить.



Запускаем ВМ. После загрузки должна появиться консоль терминала. Если экран остается черным, нажимаем Alt+F2. Установка ОС завершена.

2.3. Создание ВМ ААА.

Создадим ВМ ААА со следующими параметрами:
- ОС Ubuntu 20.04 (Debian 10);
- оперативная память 1 ГБ, количество процессоров 2;
- диск 8 ГБ (диск разбить на следующие разделы: part1 - 1M, ef02, grub; part2 - 512M, 8200, swap; part3 - все остальное пространство, 8300, root);
- сетевой адаптер NIC1 (vm-int, изолированная).

Последовательность действий и параметров при создании и установке операционной системы ВМ ААА такая же, за исключением:
- название тома хранилища – aaa.qcow2;
- название ВМ – ААА;
- при загрузке по сети, используем сетевой адаптер vm-net (внешняя), после установки ОС меняем на vm-int (изолированная);
- имя компьютера – aaa.

3.1. Настройка ВМ VyOS.

Введем следующие команды:

```
vyos@vyos:$ configure
[edit]
vyos@vyos# set service ssh
set interfaces ethernet eth0 address 10.100.10.4/26
set interfaces ethernet eth1 address 172.16.0.1/24
commit
save
set nat source rule 5 destination address !172.16.0.0/24
set nat source rule 5 source address 172.16.0.0/24
set nat source rule 5 outbound-interface any
set nat source rule 5 translation address 10.100.10.4
commit
save
show nat
```

Должны будем увидеть следующий вывод:

```
vyos@vyos# show nat
 source {
     rule 5 {
         destination {
             address !172.16.0.0/24
         }
         outbound-interface any
         source {
             address 172.16.0.0/24
         }
         translation {
             address 10.100.10.4
         }
     }
 }
[edit]
vyos@vyos# _
```

Настройка ВМ VyOS закончена.

3.2. Настройка ВМ DNS.

Настроим сеть через утилиту netplan.

root@dns:~# ip a

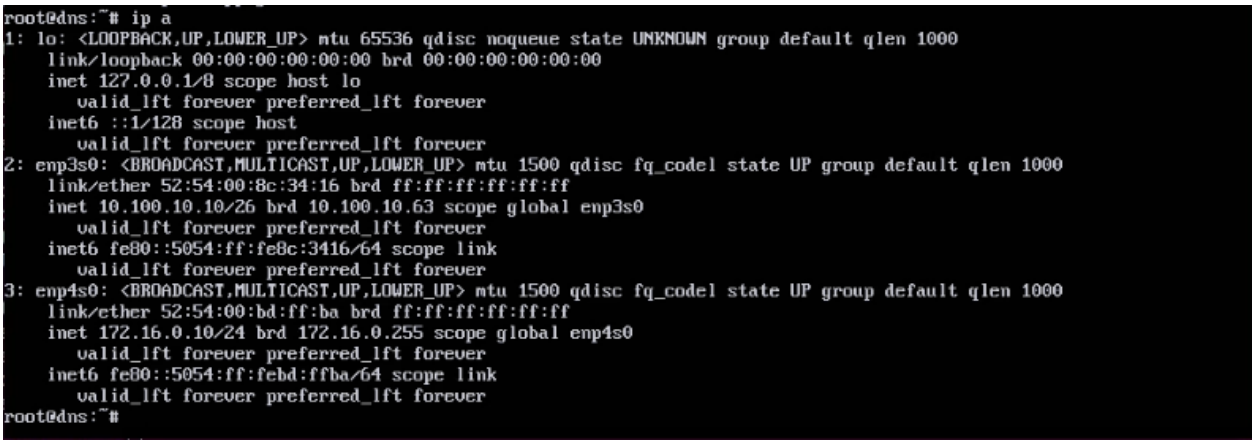
root@dns:~# vim /etc/netplan/01-netcfg.yaml

```
network:
  version: 2
  renderer: networkd
  ethernets:
    enp3s0:
      dhcp4: no
       addresses: [10.100.10.10/26]
       gateway4: 10.100.10.1
       nameservers:
         addresses: [8.8.8.8]
    enp4s0:
      dhcp4: no
       addresses: [172.16.0.10/24]
```

root@dns:~# netplan generate

root@dns:~# netplan apply

root@dns:~# ip a



Проверим ping до VyOS и DNS-сервера Google:

root@dns:~# ping 172.16.0.1

root@dns:~# ping 8.8.8.8

```
root@dns:~# ping 172.16.0.1
PING 172.16.0.1 (172.16.0.1) 56(84) bytes of data.
64 bytes from 172.16.0.1: icmp_seq=1 ttl=64 time=0.705 ms
64 bytes from 172.16.0.1: icmp_seq=2 ttl=64 time=0.586 ms
64 bytes from 172.16.0.1: icmp_seq=3 ttl=64 time=0.594 ms
^C
--- 172.16.0.1 ping statistics ---
3 packets transmitted, 3 received, 0% packet loss, time 2004ms
rtt min/avg/max/mdev = 0.586/0.628/0.705/0.054 ms
root@dns:~# ping 8.8.8.8
PING 8.8.8.8 (8.8.8.8) 56(84) bytes of data.
64 bytes from 8.8.8.8: icmp_seq=1 ttl=108 time=23.5 ms
64 bytes from 8.8.8.8: icmp_seq=2 ttl=108 time=23.1 ms
64 bytes from 8.8.8.8: icmp_seq=3 ttl=108 time=23.4 ms
^C
--- 8.8.8.8 ping statistics ---
3 packets transmitted, 3 received, 0% packet loss, time 2003ms
rtt min/avg/max/mdev = 23.102/23.335/23.528/0.176 ms
root@dns:~#
```

Заданные адреса доступны.

Подключимся к ВМ DNS с host-машины через ssh:

```
root@host:~# ssh -l tech 10.100.10.10
tech@dns:~$ sudo su -
root@dns:~# apt update
root@dns:~# apt dist-upgrade
```

Установим DNS-сервер Bind9:

```
root@dns:~# apt install bind9
root@dns:~# systemctl stop bind9
```

Отредактируем файл конфигурации загрузчика grub:

```
root@dns:~# vim /etc/default/grub
```

Отключим из загрузки модуль AppArmor, отредактировав строку:

```
GRUB_CMDLINE_LINUX_DEFAULT="apparmor=0"
```

```
# If you change this file, run 'update-grub' afterwards to update
# /boot/grub/grub.cfg.
# For full documentation of the options in this file, see:
#   info -f grub -n 'Simple configuration'

GRUB_DEFAULT=0
GRUB_TIMEOUT_STYLE=hidden
GRUB_TIMEOUT=0
GRUB_DISTRIBUTOR=`lsb_release -i -s 2> /dev/null || echo Debian`
GRUB_CMDLINE_LINUX_DEFAULT="apparmor=0"
GRUB_CMDLINE_LINUX=""
```

```
root@dns:~# update-grub
root@dns:~# reboot
```

Установим для удобства Midnight Commander:

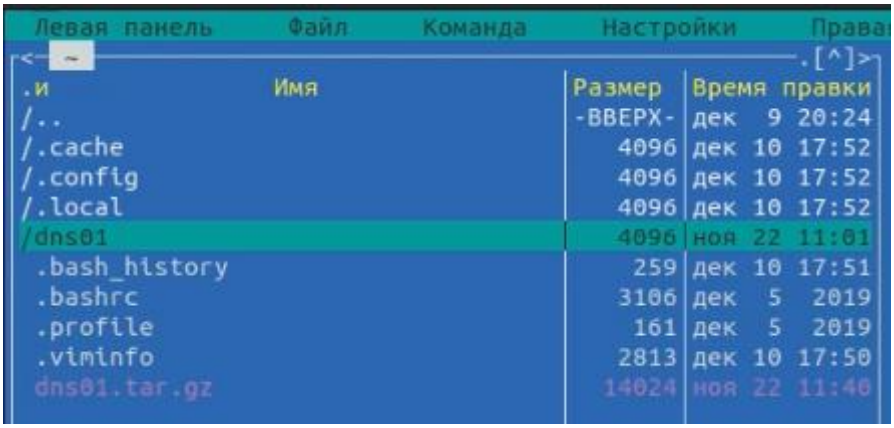
```
root@dns:~# apt install mc
root@dns:~# mc
```

Скачаем файл dns01.tar.gz:

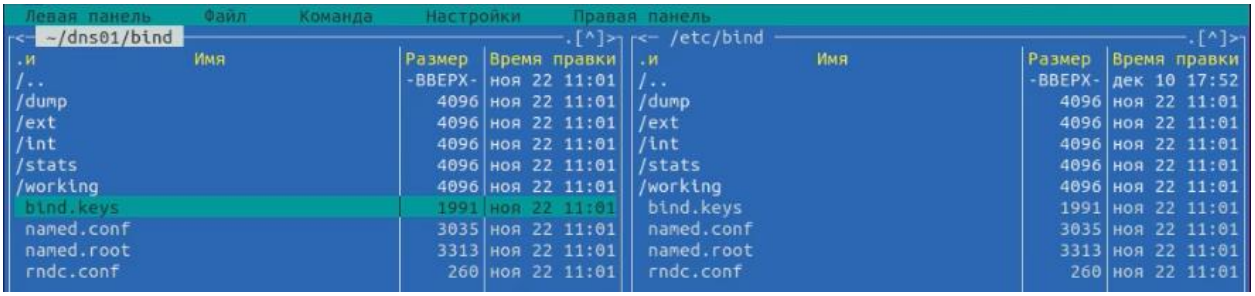
```
root@dns:~# wget
http://vcp.meganet.ru/dists/gb/linux/coruscant/dns01.tar.gz
```

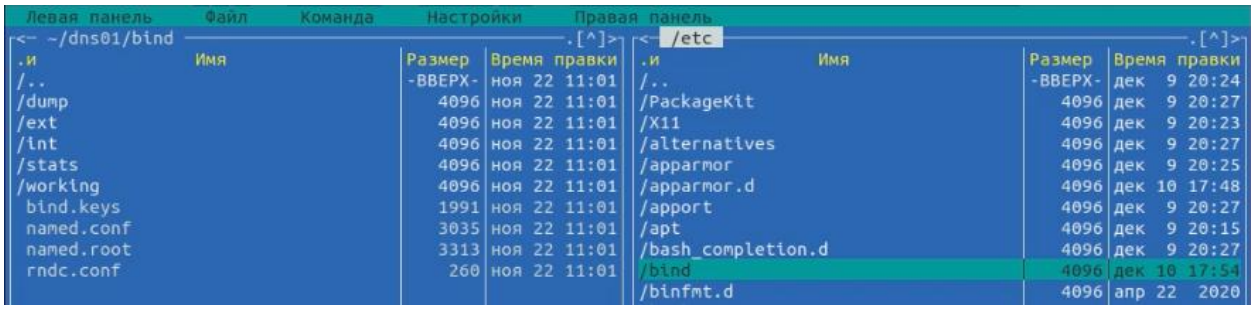
И распакуем его:

```
root@dns:~# tar -xvpf dns01.tar.gz
```



Появилась директория dns01, заходим в нее и далее в директорию bind.
В правой панели MC переходим в директорию /etc/bind/ и удаляем из нее все файлы.
Копируем все содержимое директории /root/dns01/ в /etc/bind/.





Изменим владельца и группу для директории /etc/bind/:

```
root@dns:/etc# chown -R bind:bind bind
```

Отредактируем файл /etc/bind/named.conf:

```
root@dns:/etc/bind# vim named.conf
```

```
// Acl Data

acl "ext" { 127.0.0.0/8; 10.100.10.0/26; };
acl "int" { 172.16.0.0/24; };
acl "mgmt" { 127.0.0.0/8; 172.16.0.0/24; };

// DNS Key
key "rndc-key" {
        algorithm hmac-sha256;
        secret "AzR8VAiTYoBoG6C2j2oEliWWnML1iSd+RJ2fpyOPN1I=";
};

// Control
controls {
       inet 127.0.0.1 port 953 allow { mgmt; } keys { "rndc-key"; };
       inet 172.16.0.10 port 953 allow { mgmt; } keys { "rndc-key"; };
};
```

```
statistics-channels {
    inet 172.16.0.10 port 80 allow { mgmt; };
};

// Logging Section
logging {
```

```
// Internal view section
view "int-in" {
    match-clients { int; };
    recursion yes;
    allow-recursion { ext; int; };

    zone "." {
        type hint;
        file "/etc/bind/named.root";
    };

    zone "example.int" {
        type master;
        file "/etc/bind/int/db.example.int";
        allow-update { mgmt; };
    };
};
```

```
// External View Section
view "ext-in" {
    match-clients { ext; int; any; };
    recursion yes;
    allow-recursion { ext; int; };

    zone "." {
        type hint;
        file "/etc/bind/named.root";
    };

    zone "example.int" {
        type master;
        file "/etc/bind/ext/db.example.int";
        allow-update { mgmt; };
    };
};
```

Перейдем в директорию /etc/bind/ext/ и переименуем файл db.galaxy-net.ml:
root@dns:/etc/bind/ext# mv db.galaxy-net.ml db.example.int

Отредактируем файл db.example.int:
root@dns:/etc/bind/ext# vim db.example.int

```
$TTL 600
$ORIGIN example.int.
@                               IN  SOA dns01.example.int. tech.example.int. (
                                        2021121001  ; Serial number
                                        600         ; refresh
                                        60          ; retry
                                        600         ; Expiry
                                        600         ; Minimum
                                    )

@                               IN  NS      dns01.example.int.

dns01                   600     IN  A       10.100.10.10
```

Скопируем файл /etc/bind/ext/db.example.int в директорию /etc/bind/int/.
Из директории /etc/bind/int/ удалим файл db.galaxy-net.ml.

Отредактируем файл db.example.int:

```
root@dns:/etc/bind/int# vim db.example.int
```

```
$TTL 600
$ORIGIN example.int.
@                               IN  SOA dns01.example.int. tech.example.int. (
                                        2021121001  ; Serial number
                                        600         ; refresh
                                        60          ; retry
                                        600         ; Expiry
                                        600         ; Minimum
                                    )

@                               IN  NS      dns01.example.int.

dns01                   600     IN  A       172.16.0.10
aaa                     600     IN  A       172.16.0.11
```

Перейдем в директорию /var/log/ и создадим в ней директорию named:

```
root@dns:/var/log# mkdir named
```

Изменим владельца и группу для директории /etc/bind/:

```
root@dns:/var/log# chown -R bind:bind named
```

Перезапустим сервис Bind9:

```
root@dns:/etc/bind/int# systemctl stop bind9
root@dns:/etc/bind/int# systemctl start bind9
```

Введем команду:

```
root@dns:/etc/bind/int# dig @172.16.0.10 soa example.int
```

```
root@dns:/etc/bind/int# dig @172.16.0.10 soa example.int

; <<>> DiG 9.16.1-Ubuntu <<>> @172.16.0.10 soa example.int
; (1 server found)
;; global options: +cmd
;; Got answer:
;; ->>HEADER<<- opcode: QUERY, status: NOERROR, id: 12456
;; flags: qr aa rd ra; QUERY: 1, ANSWER: 1, AUTHORITY: 0, ADDITIONAL: 1

;; OPT PSEUDOSECTION:
; EDNS: version: 0, flags:; udp: 4096
; COOKIE: 368944d2ee9b20d50100000061b36f669bcaf830824a202e (good)
;; QUESTION SECTION:
;example.int.                   IN      SOA

;; ANSWER SECTION:
example.int.            600     IN      SOA     dns01.example.int. tech.example.int. 2021121001 600 60 600 600

;; Query time: 0 msec
;; SERVER: 172.16.0.10#53(172.16.0.10)
;; WHEN: Пт дек 10 18:16:54 MSK 2021
;; MSG SIZE  rcvd: 115

root@dns:/etc/bind/int#
```

Получим следующий вывод. Обратите внимание на секцию ANSWER SECTION, если настройка верна, в ней будет отражена запись dns01.example.int.

Введем команду:
root@dns:/etc/bind/int# dig @172.16.0.10 a dns01.example.int

```
root@dns:/etc/bind/int# dig @172.16.0.10 a dns01.example.int

; <<>> DiG 9.16.1-Ubuntu <<>> @172.16.0.10 a dns01.example.int
; (1 server found)
;; global options: +cmd
;; Got answer:
;; ->>HEADER<<- opcode: QUERY, status: NOERROR, id: 38992
;; flags: qr aa rd ra; QUERY: 1, ANSWER: 1, AUTHORITY: 0, ADDITIONAL: 1

;; OPT PSEUDOSECTION:
; EDNS: version: 0, flags:; udp: 4096
; COOKIE: aa7a9e6dd31f6c8e0100000061b36f72581c2ae243c055af (good)
;; QUESTION SECTION:
;dns01.example.int.             IN      A

;; ANSWER SECTION:
dns01.example.int.      600     IN      A       172.16.0.10

;; Query time: 0 msec
;; SERVER: 172.16.0.10#53(172.16.0.10)
;; WHEN: Пт дек 10 18:17:06 MSK 2021
;; MSG SIZE  rcvd: 90

root@dns:/etc/bind/int#
```

3.3. Настройка ВМ AAA.

Настроим сеть через утилиту netplan.
Проверим доступные сетевые интерфейсы:
root@aaa:~# ip a
enp8s0

Отредактируем файл 01-netcfg.yaml:
root@aaa:~# vim /etc/netplan/01-netcfg.yaml

```
network:
  version: 2
  renderer: networkd
  ethernets:
    enp8s0:
      dhcp4: no
       addresses: [172.16.0.11/24]
       gateway4: 172.16.0.1
       nameservers:
         addresses: [172.16.0.10]
```

Применяем изменения:
root@aaa:~# netplan generate
root@aaa:~# netplan apply

Переходим на ВМ **VyOS**.
Введем следующие команды:

```
vyos@vyos:$ configure
[edit]
vyos@vyos# set protocols static route 0.0.0.0/0 next-hop
10.100.10.1 distance 1
```

```
commit
save
```

Проверяем доступность адресов с ВМ **AAA**:

```
root@aaa:~# ping 8.8.8.8
root@aaa:~# ping 172.16.0.1
```

Перейдем в консоль терминала host-машины и подключимся к ВМ **DNS**:

```
root@host:~# ssh -l tech 10.100.10.10
tech@dns:~$ sudo su -
```

Подключимся к ВМ **AAA**:

```
root@dns:~# ssh -l tech 172.16.0.11
tech@aaa:~$ sudo su -
root@aaa:~#
```

Скачаем архив aaa.tar.gz по *ссылке*:

```
root@aaa:~# wget
https://vcp.meganet.ru/dists/db/linux/coruscant/aaa.tar.gz
```

Для удобства навигации запустим Midnight Commander (apt install mc):

```
root@aaa:~# mc
```

Распакуем архив aaa.tar.gz:

```
root@aaa:~# tar -xvpf aaa.tar.gz
```

Заходим в новь созданную директорию aaa. Из файла ldap.txt нам понадобятся следующие команды (пока не вводим):

```
slapadd -n 0 -F /etc/ldap/slapd.d -l 0001_cn_config_init.ldif

ldapadd -Y EXTERNAL -H ldapi:/// -f
0002_cn_config_schema_init.ldif

ldapadd -Y EXTERNAL -H ldapi:/// -f
0003_cn_config_backend_init.ldif

ldapmodify -Y EXTERNAL -H ldapi:/// -D 'cn=config' -f
0004_cn_config_dbindex_modify.ldif

ldapadd -x -H ldap://127.0.0.1 -D cn=admin,dc=example,dc=int -W
-f 0005_cn_config_frontend_init.ldif
```

Перейдем в директорию ~/aaa/ldiff/

```
root@aaa:~# cd ~/aaa/ldiff/
```

Устанавливаем **OpenLDAP** сервер:
root@aaa:~aaa/ldiff# apt install slapd
В процессе установки попросят задать пароль – задаем.

root@aaa:~aaa/ldiff# apt install ldap-utils

Останавливаем процесс slapd:
root@aaa:~aaa/ldiff# systemctl stop slapd

Очищаем директорию:
root@aaa:~aaa/ldiff# rm –rf /etc/ldap/slapd.d/*

```
root@aaa:~/aaa/ldiff# ls -la
итого 28
drwxr-xr-x 2 tech tech 4096 окт 23 14:38 .
drwxr-xr-x 3 tech tech 4096 ноя 22 11:12 ..
-rw-r--r-- 1 tech tech  347 окт 23 14:38 0001_cn_config_init.ldif
-rw-r--r-- 1 tech tech  337 окт 23 14:38 0002_cn_config_schema_init.ldif
-rw-r--r-- 1 tech tech  915 окт 23 14:38 0003_cn_config_backend_init.ldif
-rw-r--r-- 1 tech tech  463 окт 23 14:38 0004_cn_config_dbindex_modify.ldif
-rw-r--r-- 1 tech tech  437 окт 23 14:38 0005_cn_config_frontend_init.ldif
```

Начинаем редактировать наши файлы конфигурации.
root@aaa:~aaa/ldiff# vim 0001_cn_config_init.ldif

```
dn: cn=config
objectClass: olcGlobal
cn: config
olcPidFile: /var/run/slapd/slapd.pid
olcArgsFile: /var/run/slapd/slapd.args
olcLogLevel: none
olcToolThreads: 1

dn: olcDatabase=config,cn=config
objectClass: olcDatabaseConfig
olcDatabase: config
olcAccess: to * by dn.exact=gidNumber=0+uidNumber=0,cn=peercred,cn=external,cn=auth manage by * break
~
```

Должен иметь следующий вид.

Вставляем первую команду из файла ldap.txt:
root@aaa:~aaa/ldiff# slapadd -n 0 -F /etc/ldap/slapd.d -l 0001_cn_config_init.ldif

```
root@aaa:~/aaa/ldiff# slapadd -n 0 -F /etc/ldap/slapd.d -l 0001_cn_config_init.ldif
_################## 100.00% eta   none elapsed            none fast!
Closing DB...
```

root@aaa:~aaa/ldiff# chown –R openldap:openldap /etc/ldap

Запускаем процесс slapd:
root@aaa:~aaa/ldiff# systemctl start slapd

Проверяем статус процесса:

root@aaa:~aaa/ldiff# systemctl status slapd

```
root@aaa:~/aaa/ldiff# systemctl status slapd
● slapd.service - LSB: OpenLDAP standalone server (Lightweight Directory Access Protocol)
     Loaded: loaded (/etc/init.d/slapd; generated)
    Drop-In: /usr/lib/systemd/system/slapd.service.d
             └─slapd-remain-after-exit.conf
     Active: active (running) since Tue 2021-12-14 16:36:34 MSK; 15s ago
       Docs: man:systemd-sysv-generator(8)
    Process: 5359 ExecStart=/etc/init.d/slapd start (code=exited, status=0/SUCCESS)
      Tasks: 2 (limit: 1071)
     Memory: 3.1M
     CGroup: /system.slice/slapd.service
             └─5376 /usr/sbin/slapd -h ldap:/// ldapi:/// -g openldap -u openldap -F /etc/ldap/slapd.d

дек 14 16:36:34 aaa systemd[1]: Starting LSB: OpenLDAP standalone server (Lightweight Directory Access Protocol)...
дек 14 16:36:34 aaa slapd[5359]:  * Starting OpenLDAP slapd
дек 14 16:36:34 aaa slapd[5375]: @(#) $OpenLDAP: slapd  (Ubuntu) (Apr  8 2021 04:22:01) $
                                        Debian OpenLDAP Maintainers <pkg-openldap-devel@lists.alioth.debian.org>
дек 14 16:36:34 aaa slapd[5376]: slapd starting
дек 14 16:36:34 aaa slapd[5359]:    ...done.
дек 14 16:36:34 aaa systemd[1]: Started LSB: OpenLDAP standalone server (Lightweight Directory Access Protocol).
root@aaa:~/aaa/ldiff#
```

Очищаем директорию:

root@aaa:~aaa/ldiff# rm –rf /var/lib/ldap/*

Создаем директорию:

root@aaa:~aaa/ldiff# mkdir –p /var/lib/ldap/dc=example,dc=int

root@aaa:~aaa/ldiff# chown –R openldap:openldap /var/lib/ldap

Отредактируем второй файл конфигурации:

root@aaa:~aaa/ldiff# vim 0002_cn_config_schema_init.ldif

```
dn: cn=schema,cn=config
objectClass: olcSchemaConfig
cn: schema

include: file:///etc/ldap/schema/core.ldif
include: file:///etc/ldap/schema/cosine.ldif
include: file:///etc/ldap/schema/inetorgperson.ldif
include: file:///etc/ldap/schema/misc.ldif
include: file:///etc/ldap/schema/nis.ldif
include: file:///etc/ldap/schema/openldap.ldif
~
```

Должен иметь следующий вид.

Вставляем вторую команду из файла ldap.txt:

root@aaa:~aaa/ldiff# ldapadd -Y EXTERNAL -H ldapi:/// -f 0002_cn_config_schema_init.ldif

```
root@aaa:~/aaa/ldiff# ldapadd -Y EXTERNAL -H ldapi:/// -f 0002_cn_config_schema_init.ldif
SASL/EXTERNAL authentication started
SASL username: gidNumber=0+uidNumber=0,cn=peercred,cn=external,cn=auth
SASL SSF: 0
adding new entry "cn=schema,cn=config"

adding new entry "cn=core,cn=schema,cn=config"

adding new entry "cn=cosine,cn=schema,cn=config"

adding new entry "cn=inetorgperson,cn=schema,cn=config"

adding new entry "cn=misc,cn=schema,cn=config"

adding new entry "cn=nis,cn=schema,cn=config"

adding new entry "cn=openldap,cn=schema,cn=config"

root@aaa:~/aaa/ldiff#
```

Генерируем пароль:

root@aaa:~aaa/ldiff# slappasswd

```
root@aaa:~/aaa/ldiff# slappasswd
New password:
Re-enter new password:
{SSHA}xrhEgmjt+WOU+0P1/DWnKUIpUs0fd2hS
root@aaa:~/aaa/ldiff#
```

Копируем пароль (выделенная строка) в буфер обмена или текстовый файл.

Отредактируем третий файл конфигурации:

root@aaa:~aaa/ldiff# vim 0003_cn_config_backend_init.ldif

```
dn: cn=module,cn=config
objectClass: olcModuleList
cn: module
olcModulepath: /usr/lib/ldap
olcModuleload: back_hdb

dn: olcDatabase=hdb,cn=config
objectClass: olcDatabaseConfig
objectClass: olcHdbConfig
olcDatabase: {1}hdb
olcSuffix: dc=example,dc=int
olcDbDirectory: /var/lib/ldap/dc=example,dc=int
olcRootDN: cn=admin,dc=example,dc=int
olcRootPW: {SSHA}xrhEgmjt+WOU+0P1/DWnKUIpUs0fd2hS
olcDbConfig: set_cachesize 0 2097152 0
olcDbConfig: set_lk_max_objects 1500
olcDbConfig: set_lk_max_locks 1500
olcDbConfig: set_lk_max_lockers 1500
olcDbIndex: objectClass eq
olcLastMod: TRUE
olcMonitoring: TRUE
olcDbCheckpoint: 512 30
olcAccess: to attrs=userPassword by
dn="cn=admin,dc=example,dc=int" write by anonymous auth by self
write by * none
olcAccess: to attrs=shadowLastChange by self write by * read
olcAccess: to dn.base="" by * read
olcAccess: to * by dn="cn=admin,dc=example,dc=int" write by *
read
```

Должен иметь следующий вид.

Вставляем третью команду из файла ldap.txt:

root@aaa:~aaa/ldiff# ldapadd -Y EXTERNAL -H ldapi:/// -f 0003_cn_config_backend_init.ldif

```
root@aaa:~/aaa/ldiff# ldapadd -Y EXTERNAL -H ldapi:/// -f 0003_cn_config_backend_init.ldif
SASL/EXTERNAL authentication started
SASL username: gidNumber=0+uidNumber=0,cn=peercred,cn=external,cn=auth
SASL SSF: 0
adding new entry "cn=module,cn=config"

adding new entry "olcDatabase=hdb,cn=config"

root@aaa:~/aaa/ldiff#
```

Отредактируем четвертый файл конфигурации:

```
root@aaa:~aaa/ldiff# vim 0004_cn_config_dbindex_modify.ldif
```

```
dn: olcDatabase={1}hdb,cn=config
changetype: modify
add: olcDbIndex
olcDbIndex: cn pres,sub,eq
-
add: olcDbIndex
olcDbIndex: sn pres,sub,eq
-
add: olcDbIndex
olcDbIndex: uid pres,sub,eq
-
add: olcDbIndex
olcDbIndex: displayName pres,sub,eq
-
add: olcDbIndex
olcDbIndex: default sub
-
add: olcDbIndex
olcDbIndex: uidNumber eq
-
add: olcDbIndex
olcDbIndex: gidNumber eq
-
add: olcDbIndex
olcDbIndex: mail,givenName eq,subinitial
-
add: olcDbIndex
olcDbIndex: dc eq
~
```

Должен иметь следующий вид.

Вставляем четвертую команду из файла ldap.txt:

```
root@aaa:~aaa/ldiff# ldapmodify -Y EXTERNAL -H ldapi:/// -D
'cn=config' -f 0004_cn_config_dbindex_modify.ldif
```

```
root@aaa:~/aaa/ldiff# ldapmodify -Y EXTERNAL -H ldapi:/// -D 'cn=config' -f 0004_cn_config_dbindex_modify.ldif
SASL/EXTERNAL authentication started
SASL username: gidNumber=0+uidNumber=0,cn=peercred,cn=external,cn=auth
SASL SSF: 0
modifying entry "olcDatabase={1}hdb,cn=config"
```

Отредактируем пятый файл конфигурации:

```
root@aaa:~aaa/ldiff# vim 0005_cn_config_frontend_init.ldif
```

```
dn: dc=example,dc=int
objectClass: top
objectClass: dcObject
objectclass: organization
o: example
dc: example

dn: cn=admin,dc=example,dc=int
objectClass: simpleSecurityObject
objectClass: organizationalRole
cn: admin
userPassword: {SSHA}xrhEgmjt+WOU+0P1/DWnKUIpUs0fd2hS

dn: ou=people,dc=example,dc=int
objectClass: organizationalUnit
ou: people

dn: ou=groups,dc=example,dc=int
objectClass: organizationalUnit
ou: groups
```

Вставляем пятую команду из файла ldap.txt:

```
root@aaa:~aaa/ldiff# ldapadd -x -H ldap://127.0.0.1 -D
cn=admin,dc=example,dc=int -W -f
0005_cn_config_frontend_init.ldif
```

Вводим пароль, который задавали при выполнении второй команды из файла ldap.txt.

Переходим на ВМ **VyOS**.

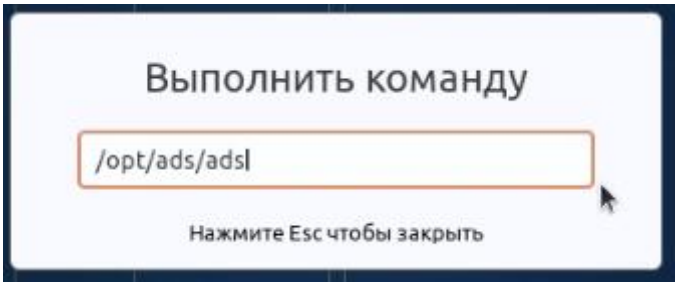
Введем следующие команды:

```
vyos@vyos:$ configure
[edit]
vyos@vyos# set nat destination rule 5 source address 0.0.0.0/0
set nat destination rule 5 destination address 10.100.10.4
set nat destination rule 5 destination port 389
set nat destination rule 5 inbound-interface eth0
set nat destination rule 5 translation address 172.16.0.11
set nat destination rule 5 translation port 389
set nat destination rule 5 protocol tcp_udp
commit
save
```

Переходим на host-машину, скачиваем Apache Directory Studio:

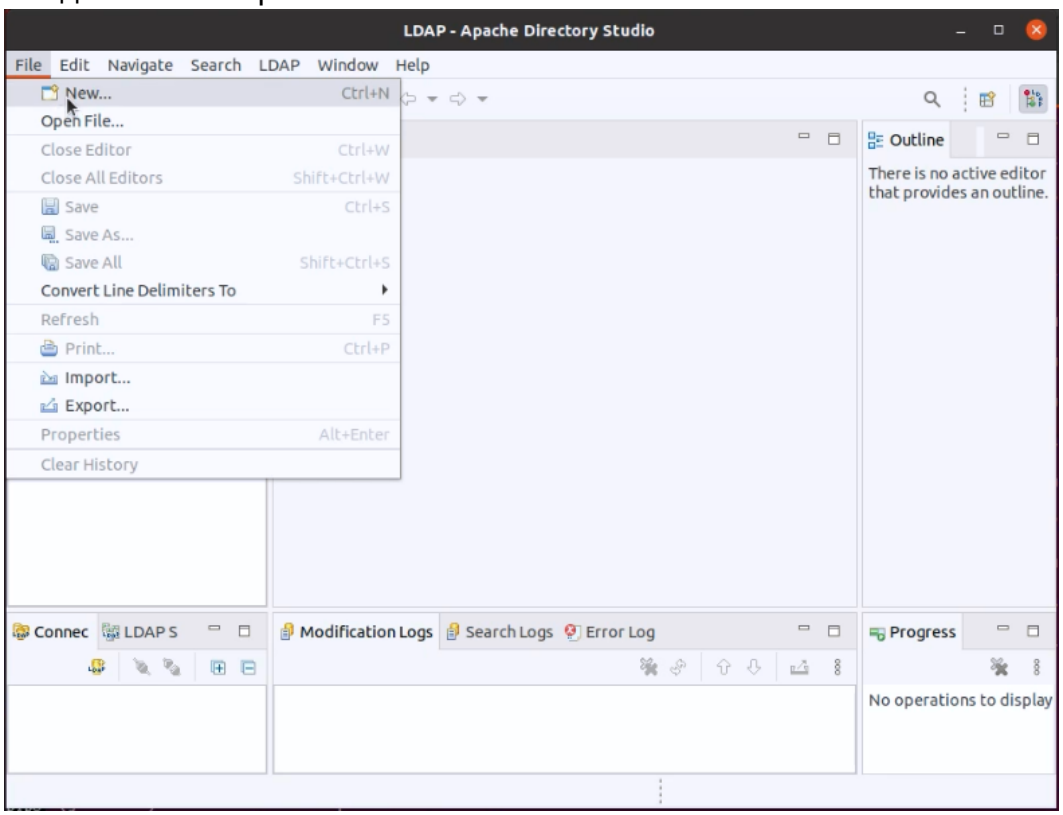
```
user@host:~$ sudo chown user:user /opt
user@host:~$ cd /opt
user@host:/opt$ wget
https://dlcdn.apache.org/directory/studio/2.0.0.v20210717-
M17/ApacheDirectoryStudio-2.0.0.v20210717-M17-
linux.gtk.x86_64.tar.gz

user@host:/opt$ tar –xvpf ApacheDirectoryStudio-2.0.0.v20210717-
M17-linux.gtk.x86_64.tar.gz

user@host:/opt$ mv ApacheDirectoryStudio ads
user@host:/opt$ cd /ads
user@host:/opt/ads$ ln –s ApacheDirectoryStudio ads
user@host:/opt$ sudo apt install default-jdk default-jre
```

Запускаем **Apache Directory Studio**:
нажимаем Alt+F2, вписываем: /opt/ads/ads

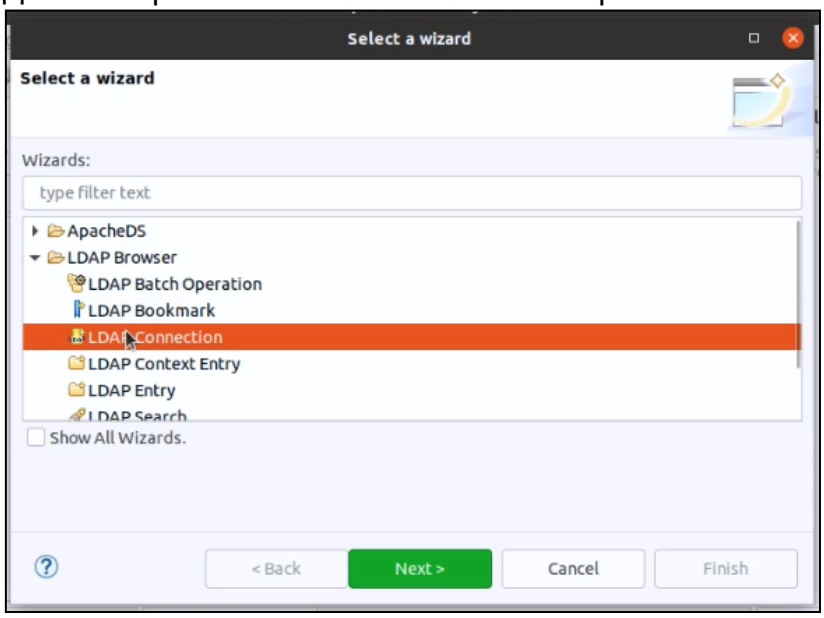


Начинается запуск Apache Directory Studio:

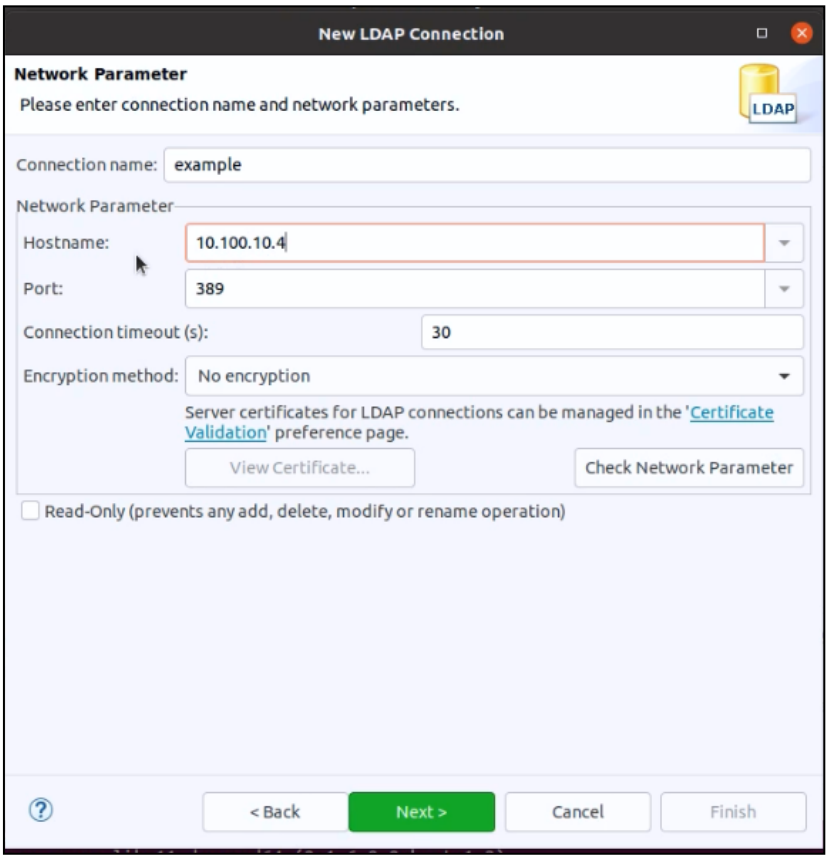
Начинаем настройку LDAP - Apache Directory Studio.
Создаем новый файл:



Далее открываем LDAP Browser и выбираем LDAP Connection, далее Next.

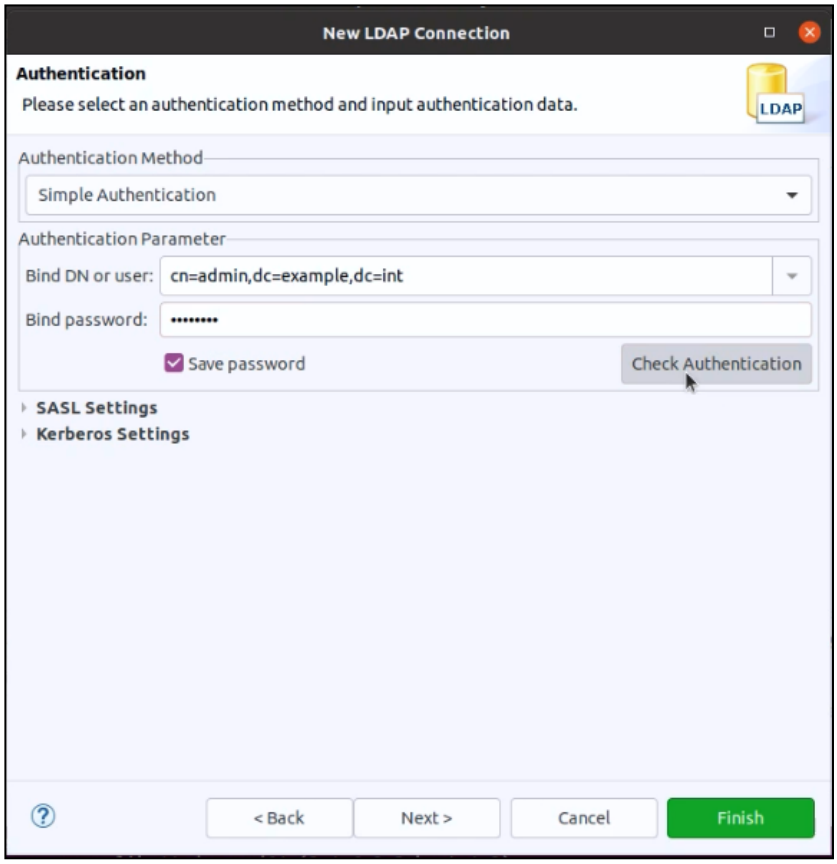


Вписываем значения в Connection name и Hostname:

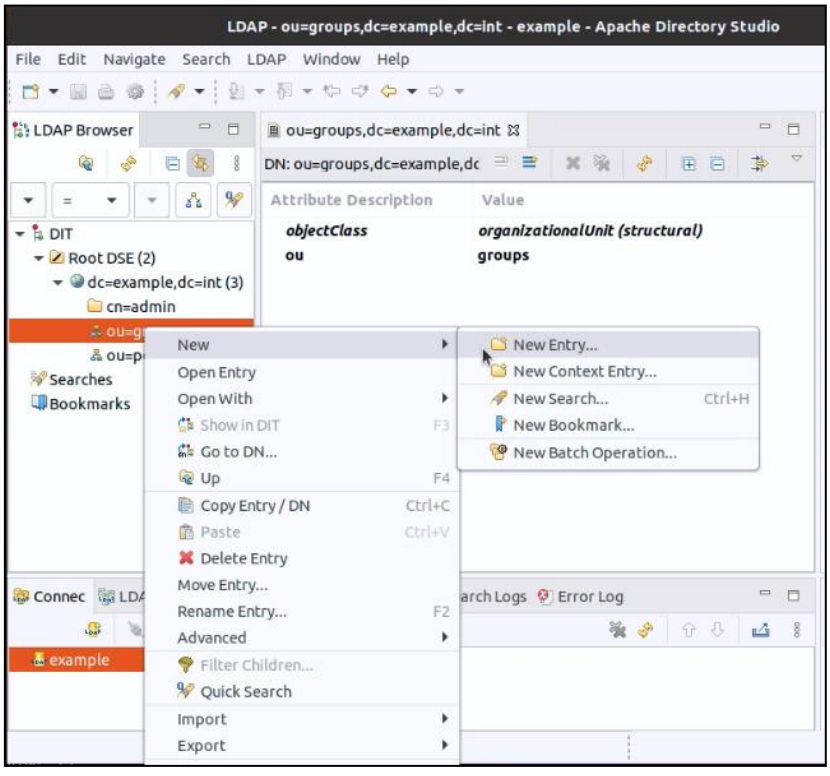


Проверяем параметры сети, нажимаем Check Network Parametr. Если все настроено правильно, увидим окно: «The connection was established successfully.»

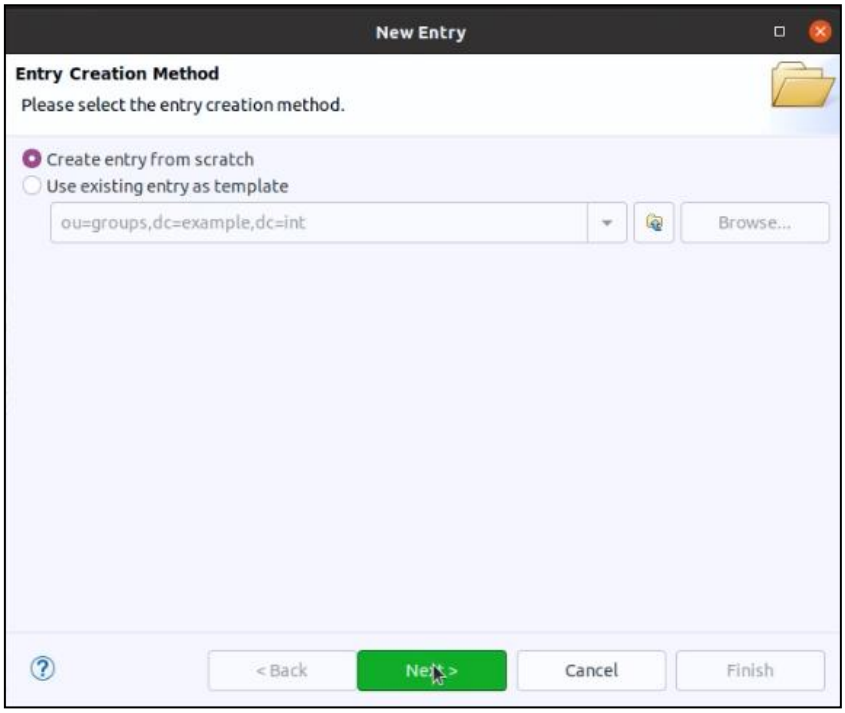
Вписываем значения в Bind DN or user и Bind password. Пароль указываем тот, который задавали командой slappasswd.



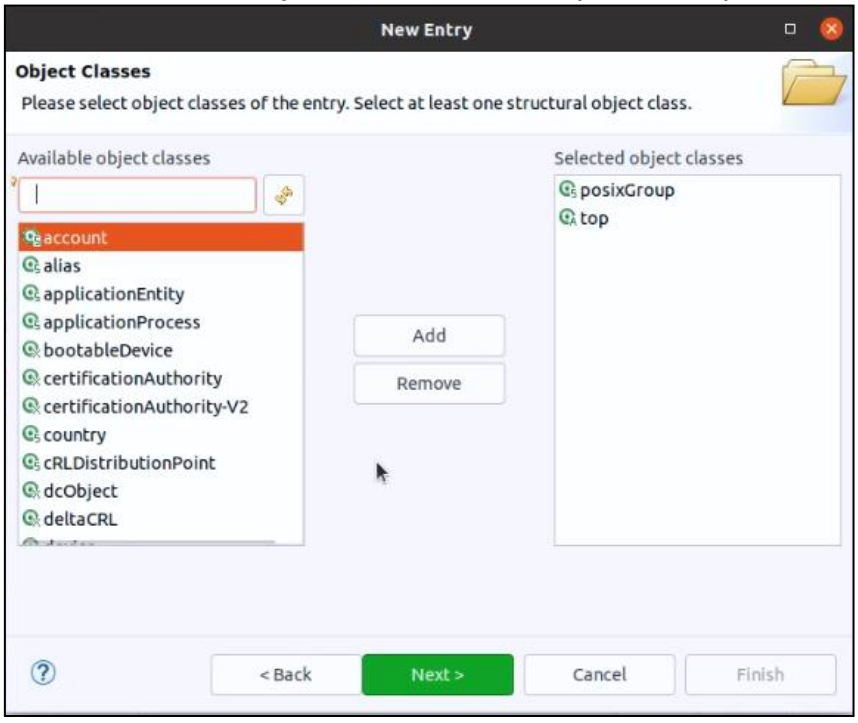
Проверяем аутентификацию, нажимаем Check Authentication. Если все настроено правильно, увидим окно: «The authentication was successful.»
Нажимаем Finish.
Открываем ou=groups, создаем новую запись - New Entry:



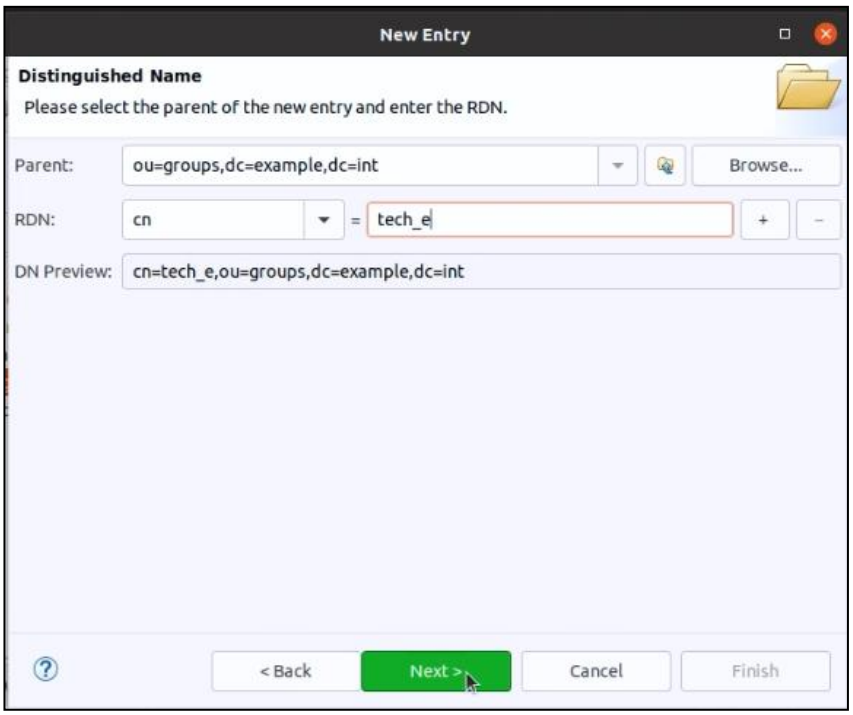
Нажимаем Next:



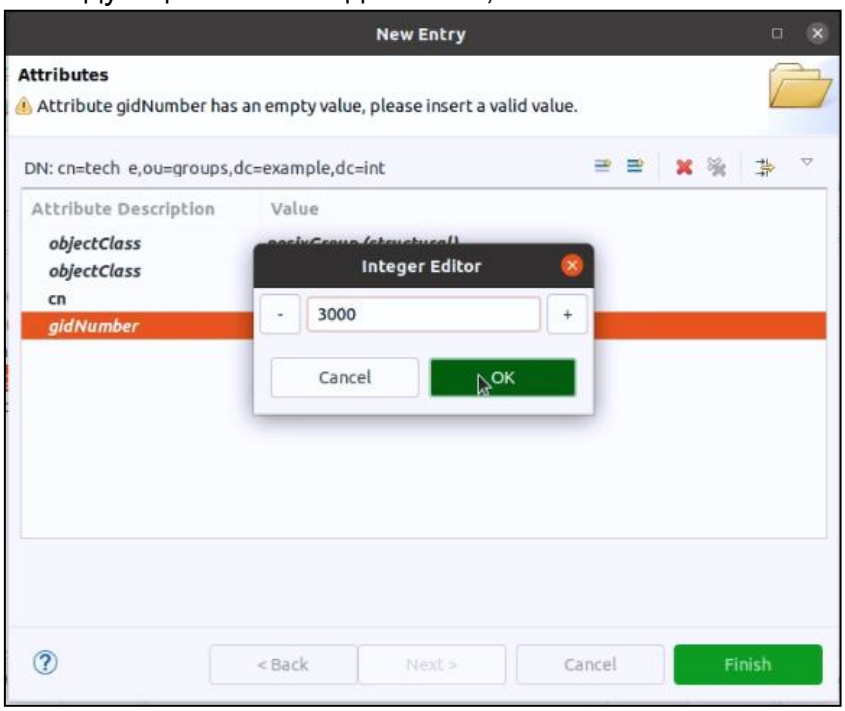
В окне Available object classes вводим posixGroup и нажимаем Add и Next.



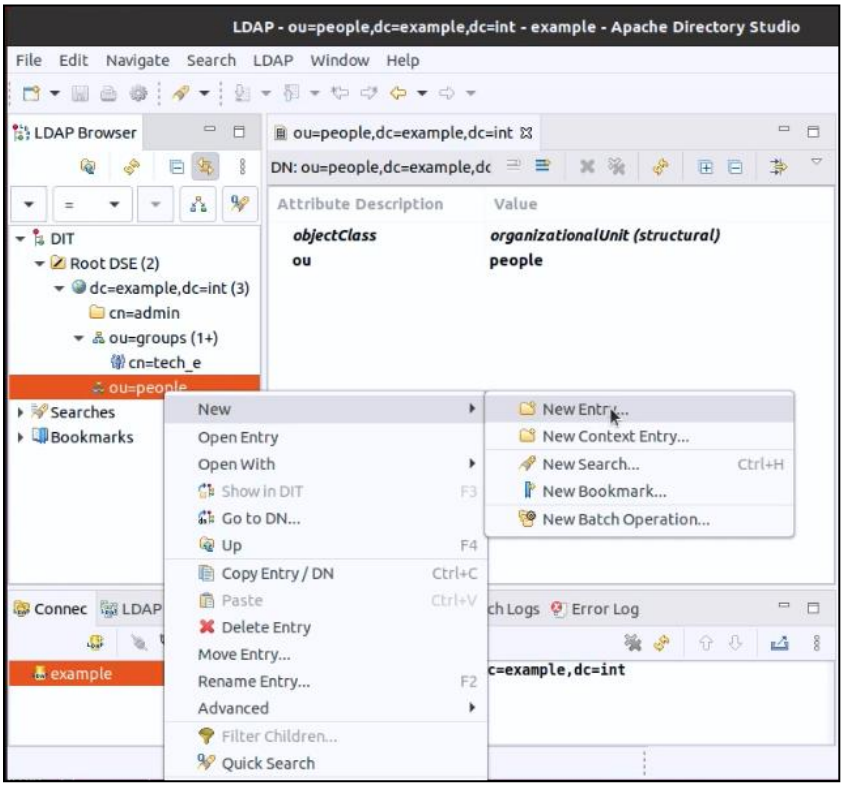
В окне RDN вводим cn, в окне после знака равно – tech_e, далее Next.



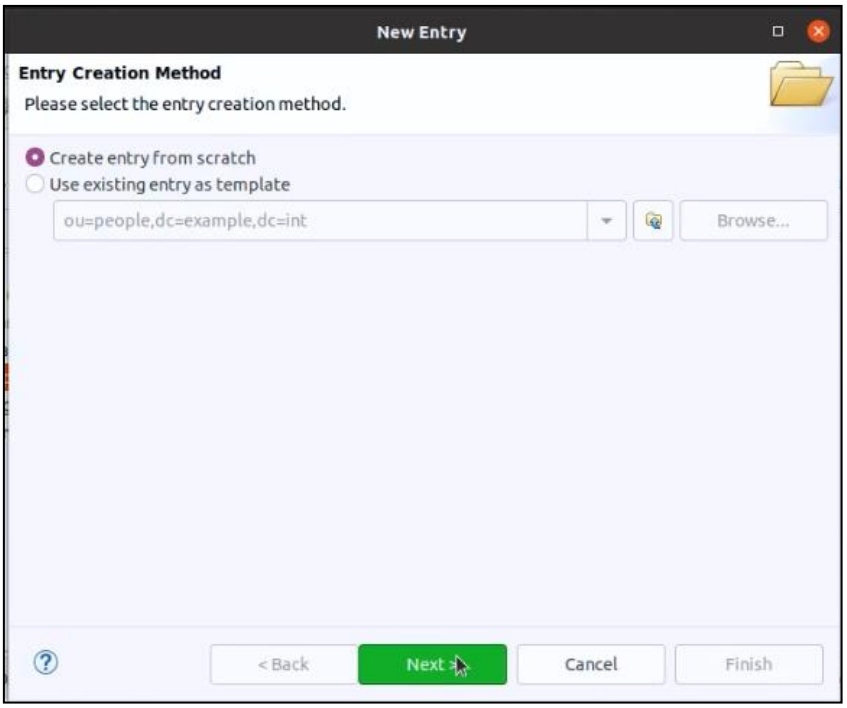
В следующем окне вводим 3000, нажимаем OK и Finish.



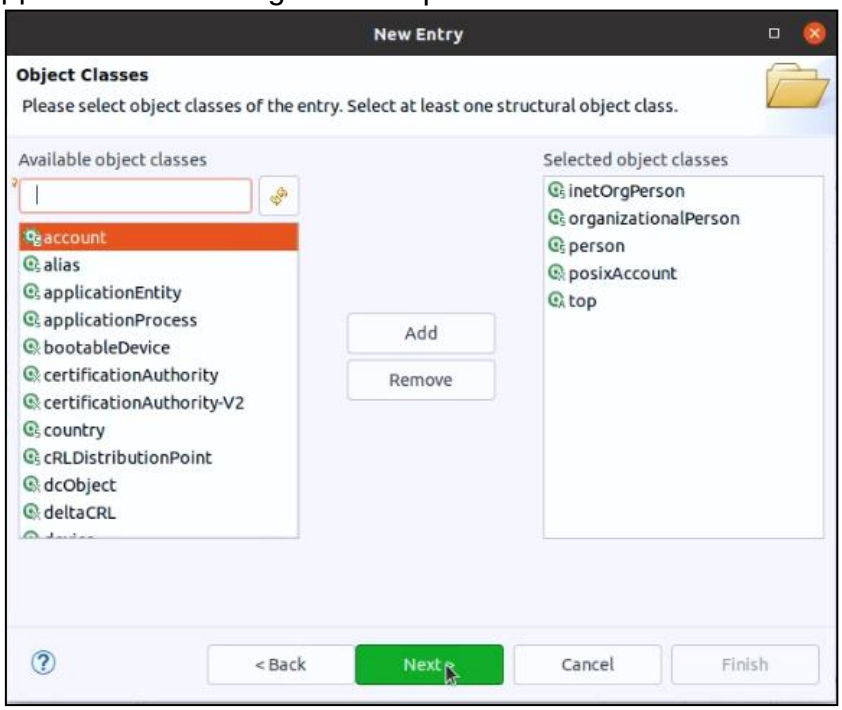
Открываем ou=people, создаем новую запись - New Entry:



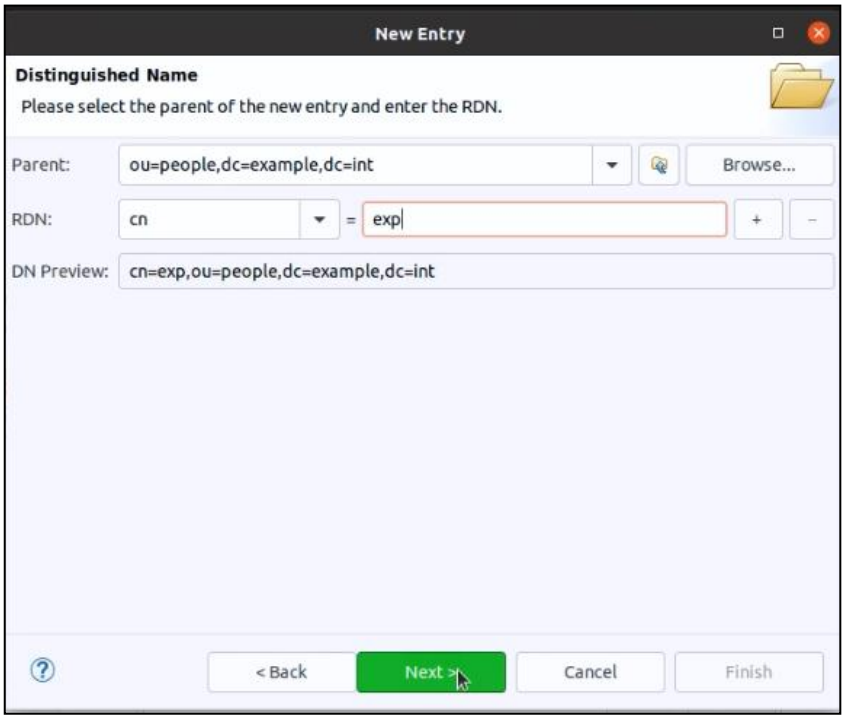
Нажимаем Next:



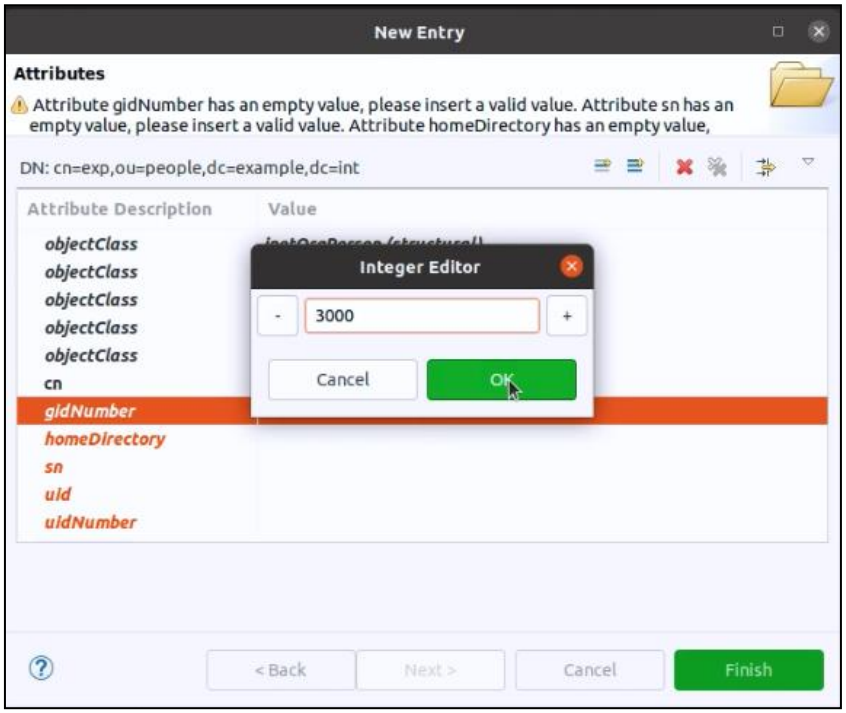
Добавляем inetOrgPerson и posixAccount. Нажимаем Next.



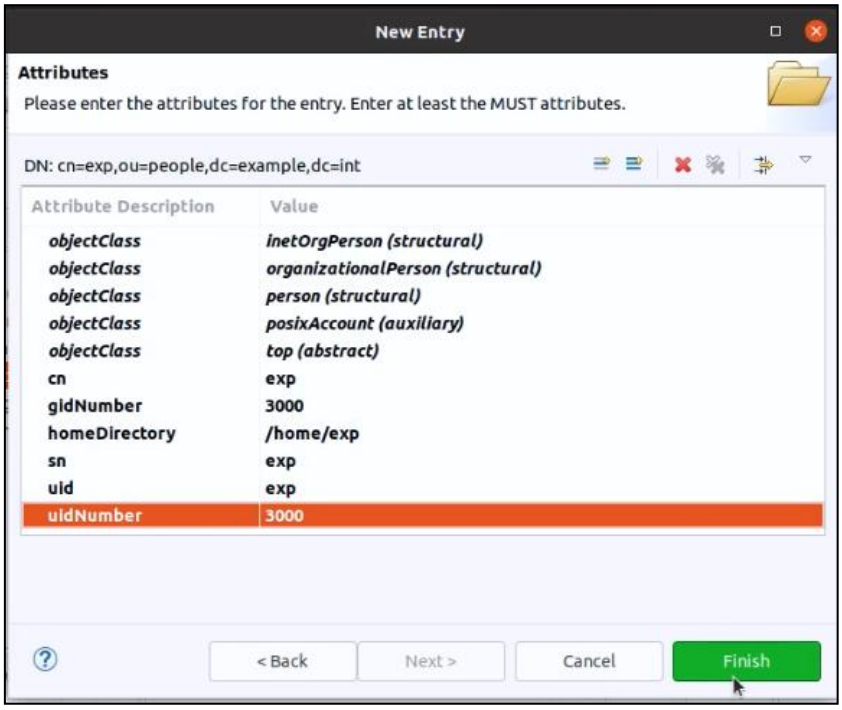
В окне RDN вводим cn, в окне после знака равно – exp, далее Next.



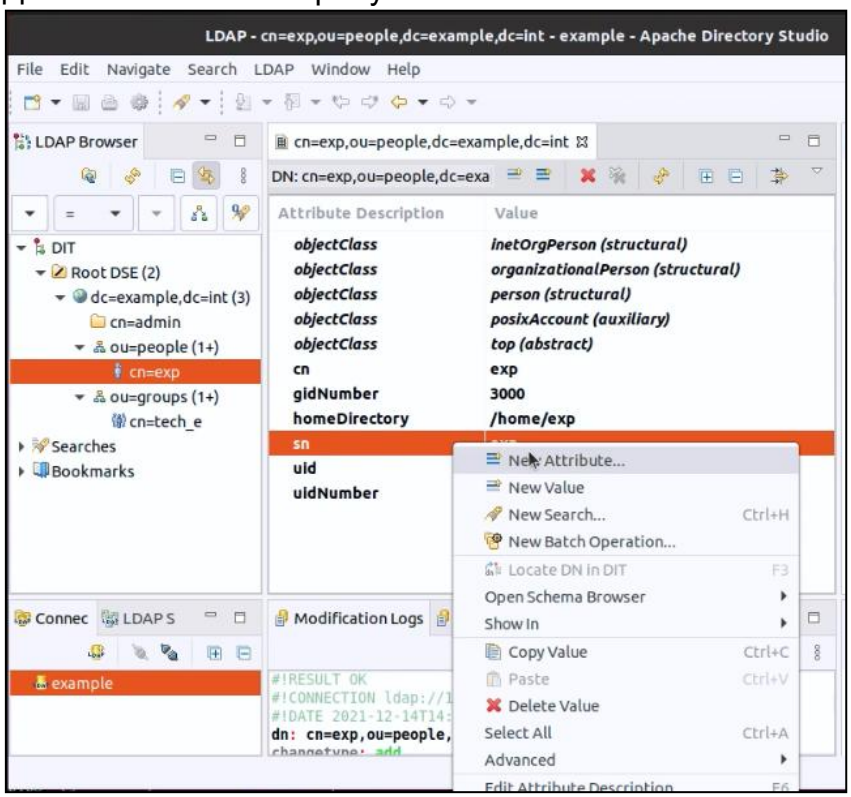
В следующем окне вводим 3000, нажимаем OK.



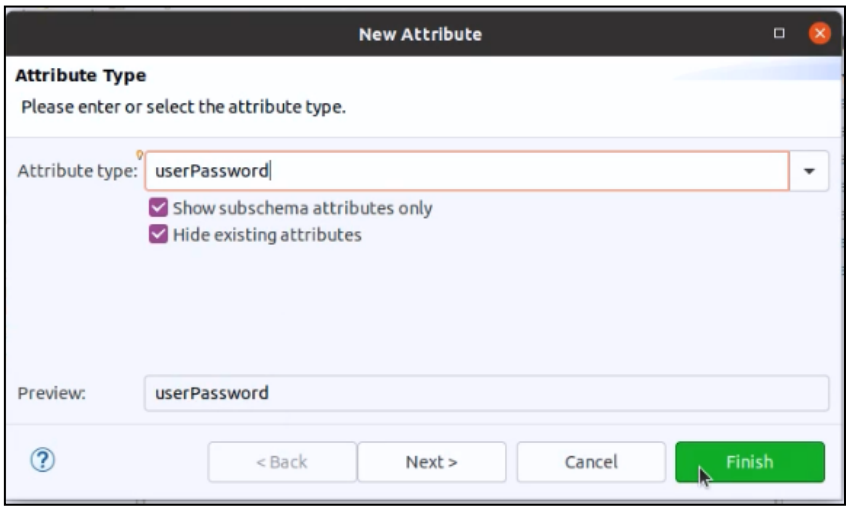
В поле homeDirectory вводим: /home/exp; в поле sn – exp; uid – exp; uidNumber – 3000. Нажимаешь Finish.



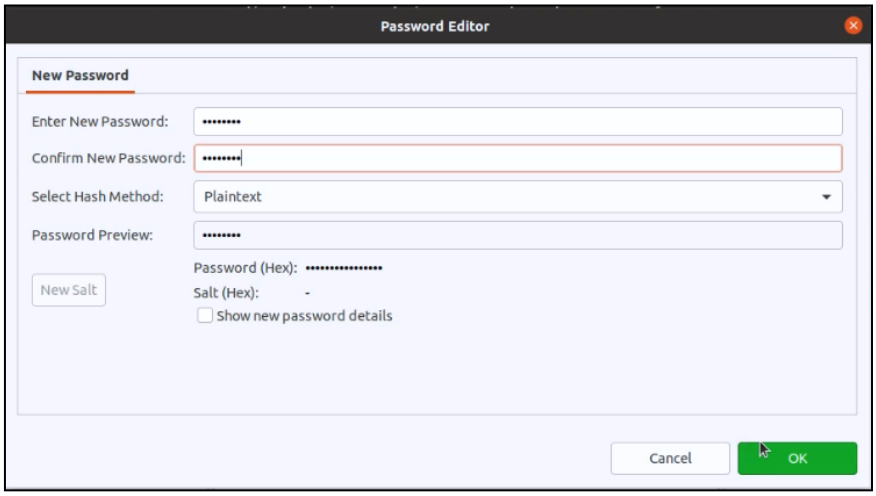
Добавляем новый атрибут:



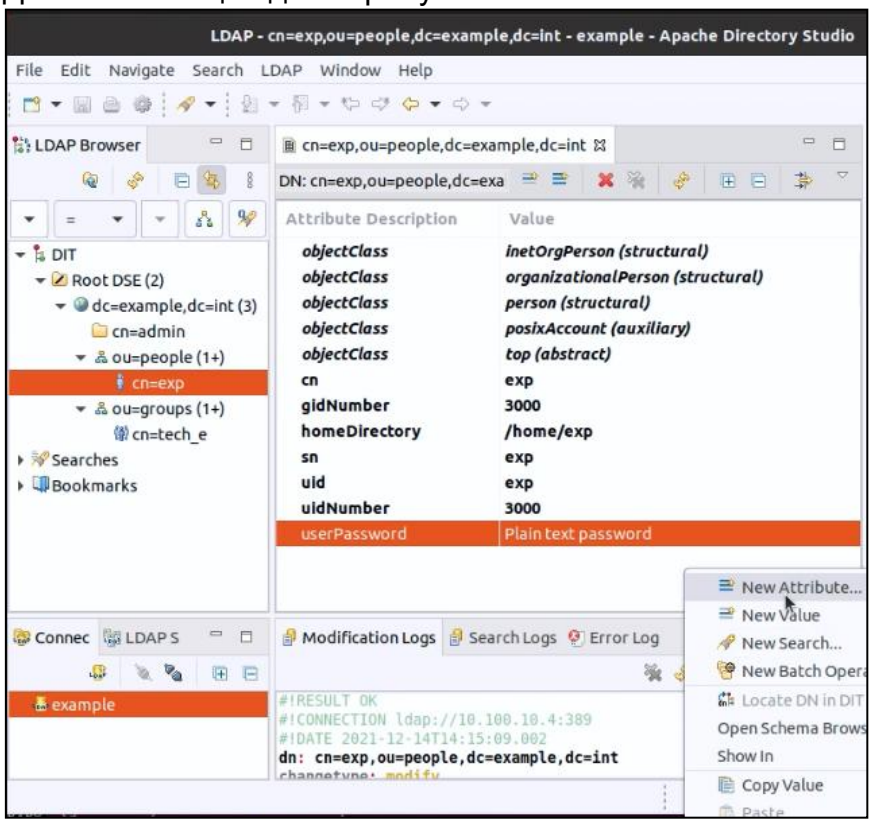
Вводим userPassword. Нажимаем Finish.



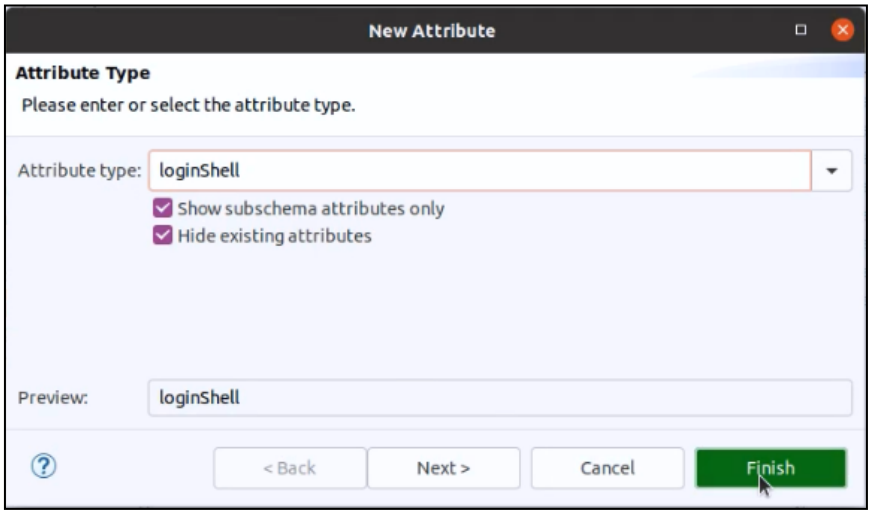
Вводим новый пароль и подтверждаем его. Нажимаем ОК.



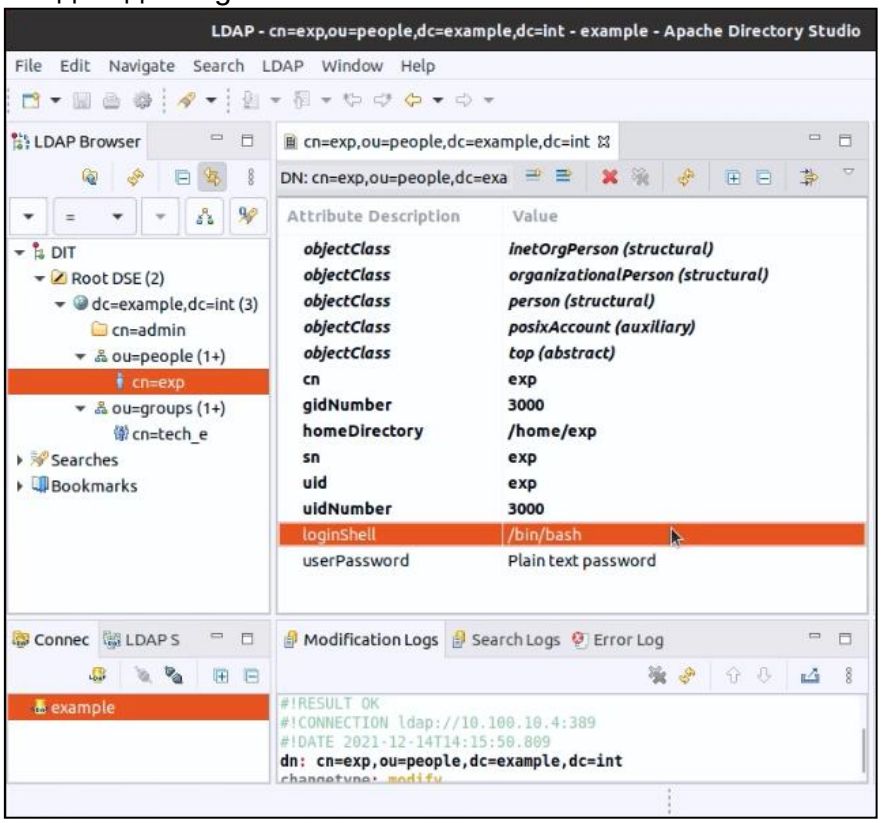
Добавляем еще один атрибут:



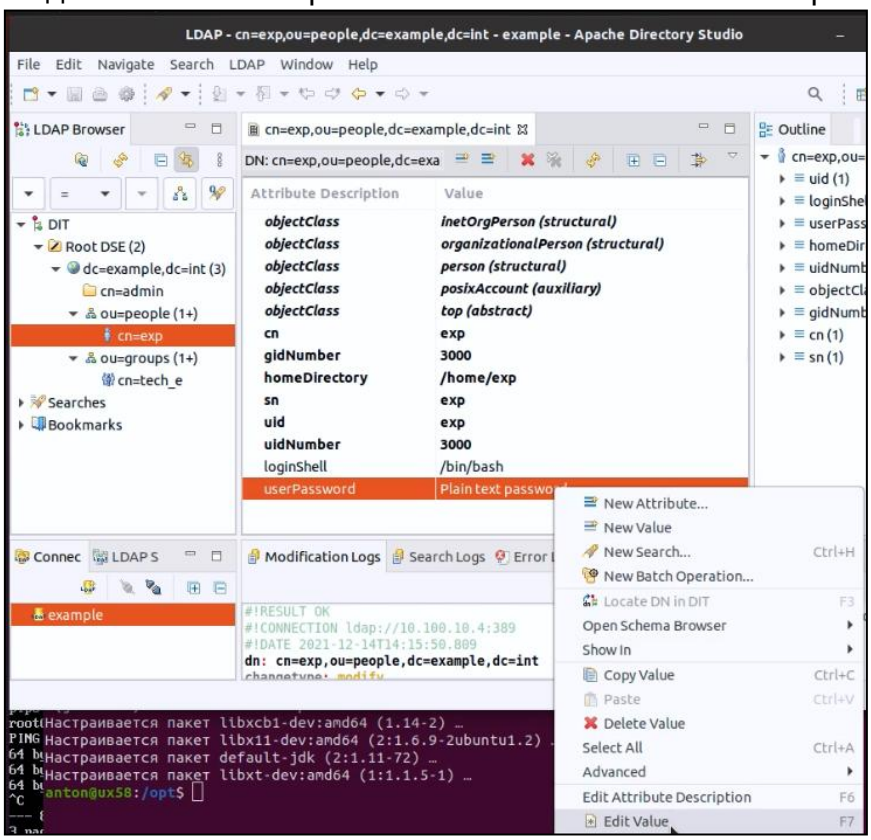
Вводим loginShell. Нажимаем Finish.



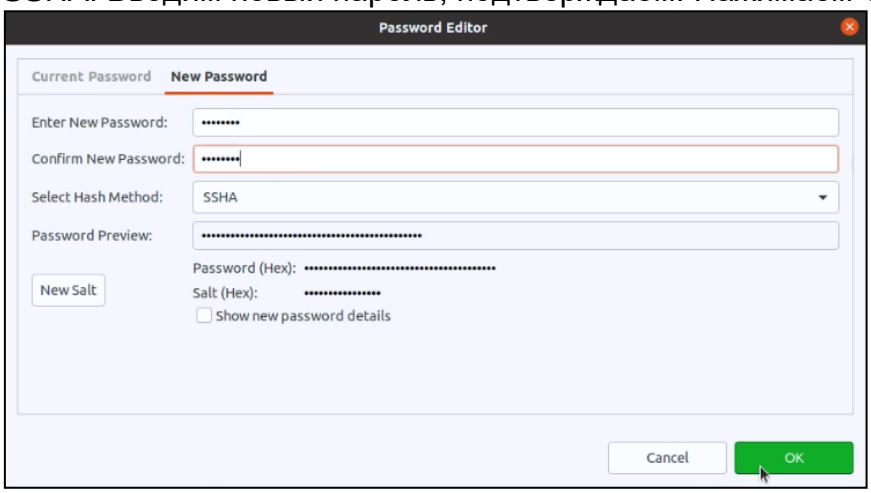
Вводим для loginShell значение: /bin/bash



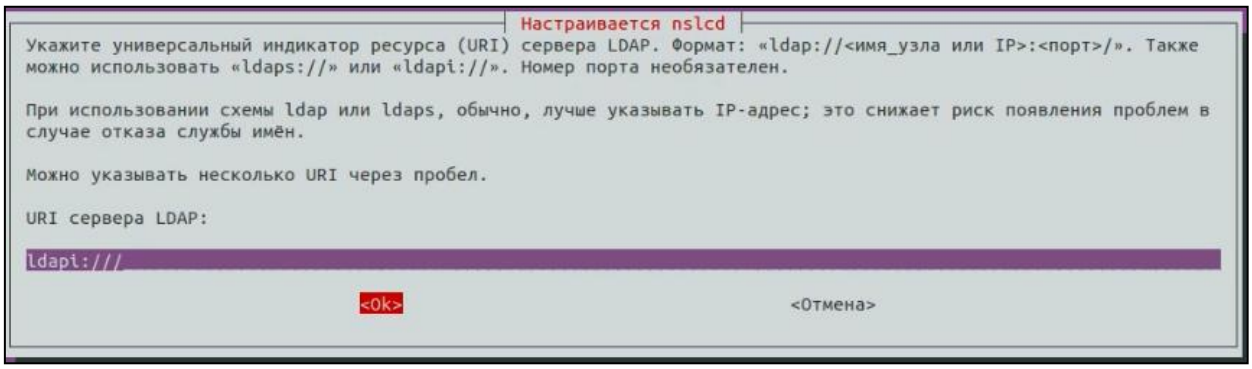
Выделяем Plain text password и нажимаем ПКМ. Выбираем Edit Value.

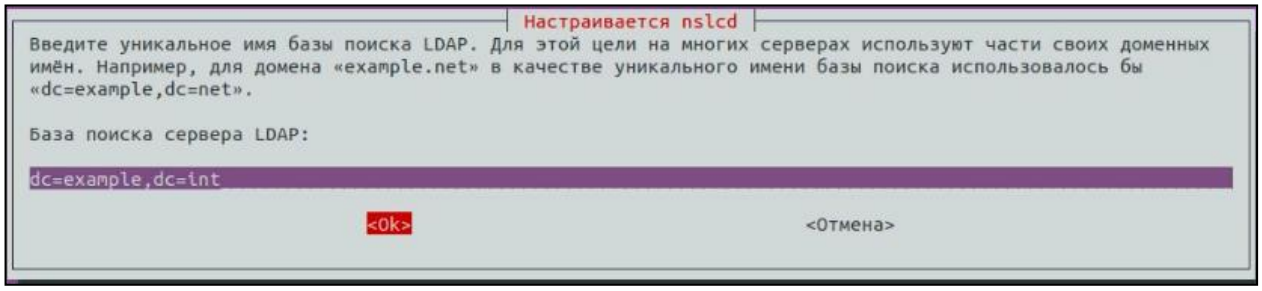


Переходим во вкладку New Password. В Select Hash Method меняем Plaintext на SSHA. Вводим новый пароль, подтверждаем. Нажимаем ОК.

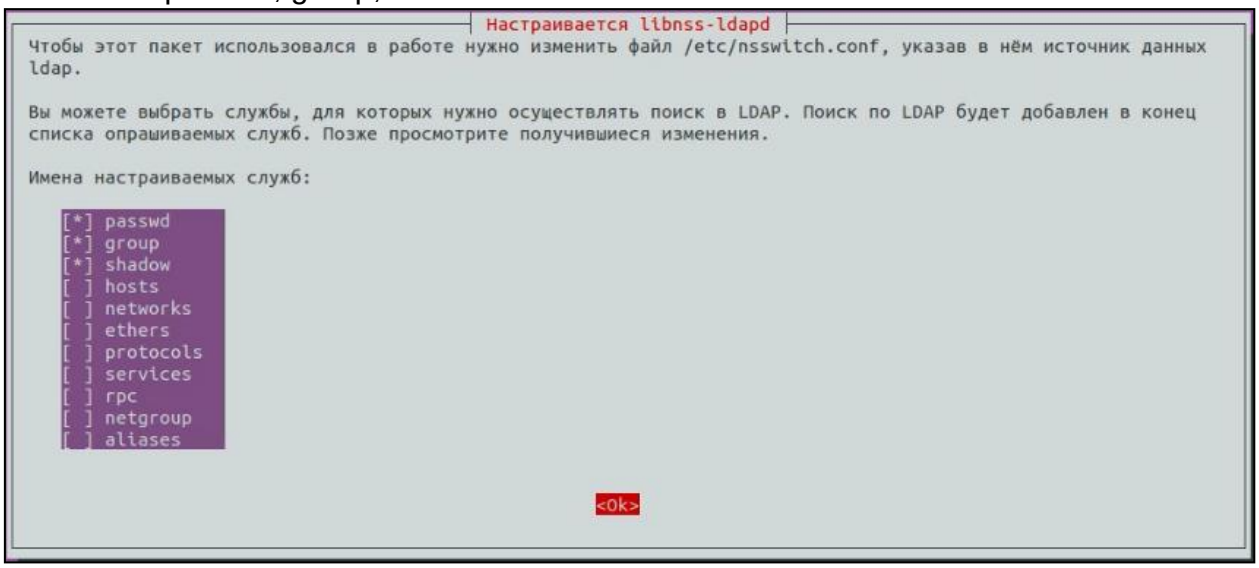


Переходим на ВМ **AAA**.
Устанавливаем следующие пакеты:
root@aaa:~/aaa/ldiff# apt install libnss-ldapd libpam-ldapd





Отмечаем passwd, group, shadow:

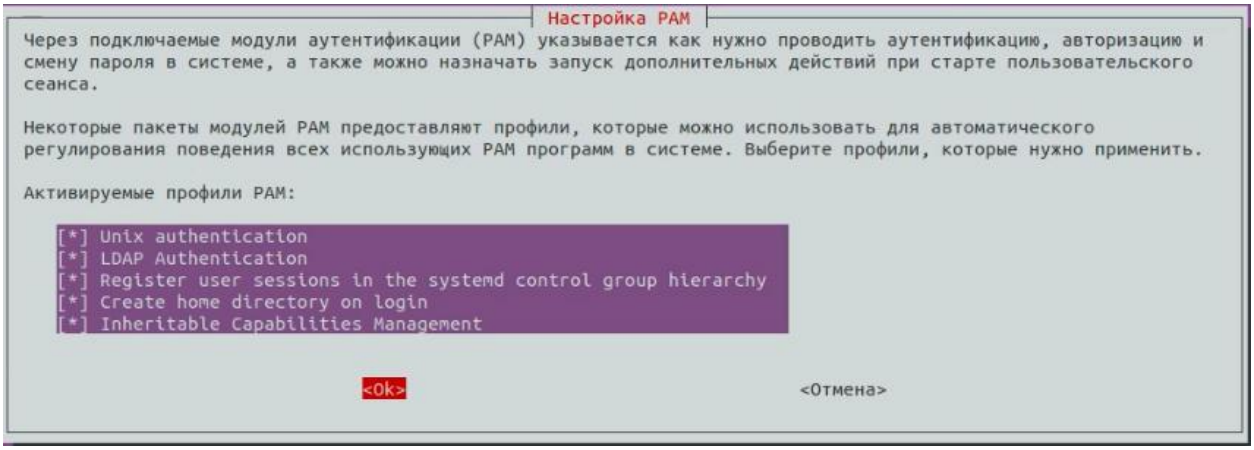


Получаем список учетных записей пользователей:
root@aaa:~/aaa/ldiff# getent passwd

В списке, среди прочих учетных записей, должны присутствовать: openldap, nslcd, exp.

Запускаем утилиту pam-auth-update:
root@aaa:~/aaa/ldiff# pam-auth-update

Все пункты должны быть отмечены:



Перезагружаем ВМ:

```
root@aaa:~/aaa/ldiff# reboot
```

После перезагрузки ВМ AAA пробуем зайти под пользователем exp:

```
root@dns:~# ssh -l exp 172.16.0.11
```

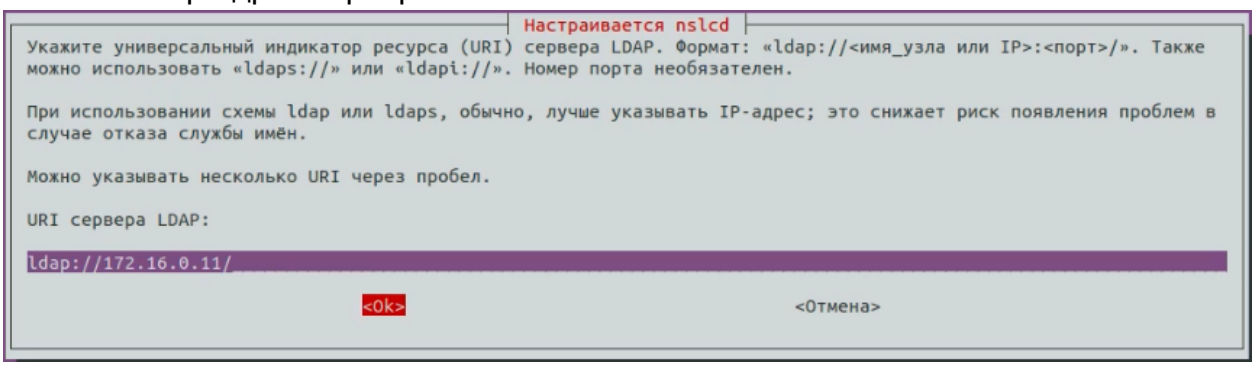
После успешной авторизации отключаемся:

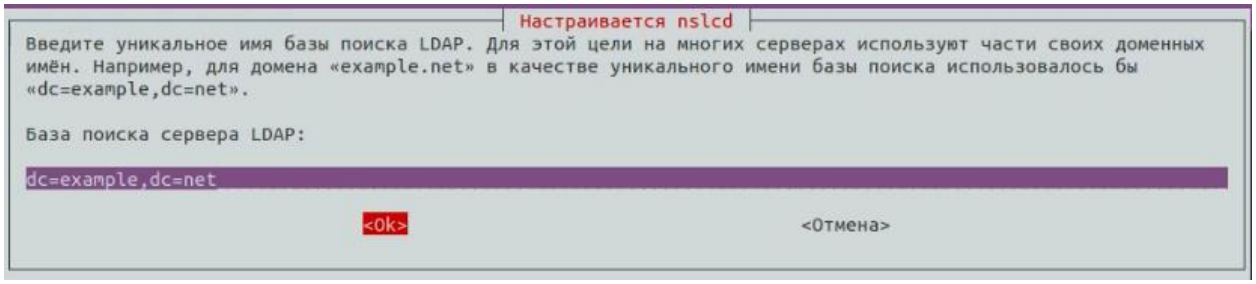
```
exp@aaa:~$ exit
```

Устанавливаем следующие пакеты:

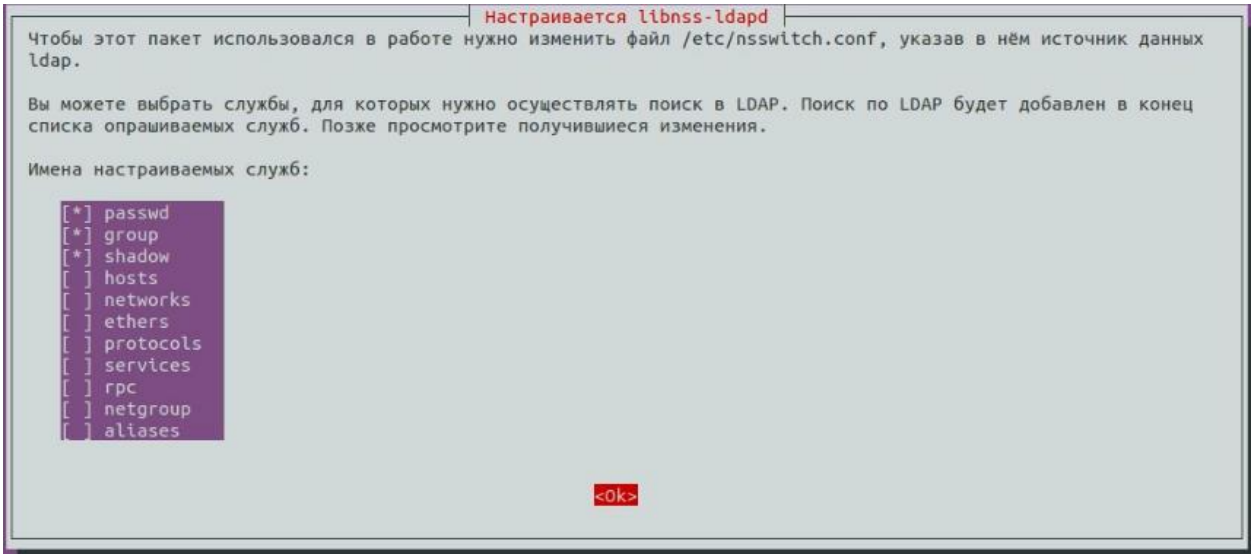
```
root@dns:~# apt install libpam-ldapd libnss-ldapd
```

Вписываем ip-адрес сервера AAA:





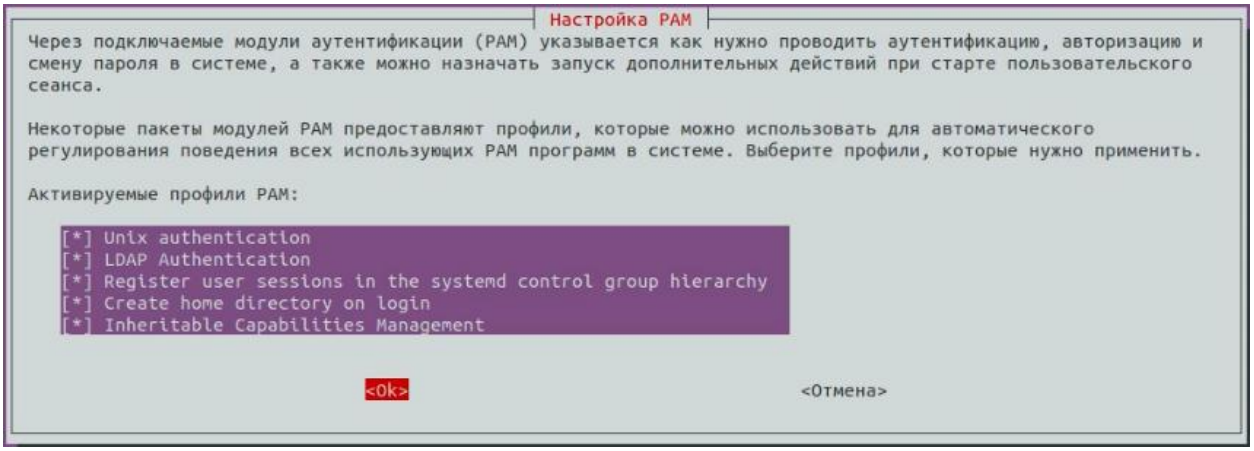
Отмечаем passwd, group, shadow:



Запускаем утилиту pam-auth-update:

```
root@dns:~/aaa/ldiff# pam-auth-update
```

Все пункты должны быть отмечены:



Получаем список учетных записей пользователей:

```
root@dns:~# getent passwd
```

В списке, среди прочих учетных записей, должны присутствовать: exp.

Перезагружаем ВМ:

```
root@dns:~# reboot
```

После перезагрузки ВМ DNS пробуем зайти под пользователем exp:

```
root@host:~# ssh -l exp 10.100.10.10
```

Если успешно авторизовались, значит настройка закончена.