Санкт-Петербургское государственное бюджетное
профессиональное образовательное учреждение
«Академия управления городской средой, градостроительства и печати»

УТВЕРЖДАЮ
Заместитель директора
по учебно-производственной работе
О.В. Фомичева
«__» _____ 202_3 г.

МЕТОДИЧЕСКИЕ РЕКОМЕНДАЦИИ
по выполнению практических работ
по МДК.02.03 Организация администрирования компьютерных систем
ПМ.02 ОРГАНИЗАЦИЯ СЕТЕВОГО АДМИНИСТРИРОВАНИЯ ОПЕРАЦИОННЫХ СИСТЕМ

для специальности

09.02.06 Сетевое и системное администрирование

Санкт-Петербург
202_3 г.

Методические рекомендации рассмотрены на заседании методического совета СПб ГБПОУ «АУГСГиП»
Протокол № 2 от «19» 11 2023г.

Методические рекомендации одобрены на заседании цикловой комиссии информационных технологий
Протокол № 4 от «21» 11 2023 г.

Председатель цикловой комиссии: Караченцева М.С. _____

Разработчики: преподаватели СПб ГБПОУ «АУГСГиП»

С О Д Е Р Ж А Н И Е

ПОЯСНИТЕЛЬНАЯ ЗАПИСКА

Рабочая тетрадь по выполнению практических работ предназначены для организации работы на практических занятиях по МДК.02.03 «Организация администрирования компьютерных систем», которая является важной составной частью в системе подготовки специалистов среднего профессионального образования по специальности 09.02.06 «Сетевое и системное администрирование».

Практические занятия являются неотъемлемым этапом изучения учебной дисциплины и проводятся с целью:

- формирования практических умений в соответствии с требованиями к уровню подготовки обучающихся, установленными рабочей программой учебной дисциплины;
- обобщения, систематизации, углубления, закрепления полученных теоретических знаний;
- готовности использовать теоретические знания на практике.

Практические занятия по МДК.02.03 «Организация администрирования компьютерных систем» способствуют формированию в дальнейшем при изучении профессиональных модулей, следующих общих и профессиональных компетенций:

ОК 01 Выбирать способы решения задач профессиональной деятельности, применительно к различным контекстам

ОК 02 Осуществлять поиск, анализ и интерпретацию информации, необходимой для выполнения задач профессиональной деятельности

ОК 03 Планировать и реализовывать собственное профессиональное и личностное развитие.

ОК 04 Работать в коллективе и команде, эффективно взаимодействовать с коллегами, руководством, клиентами.

ОК 05 Осуществлять устную и письменную коммуникацию на государственном языке с учетом особенностей социального и культурного контекста.

ОК 06 Проявлять гражданско-патриотическую позицию, демонстрировать осознанное поведение на основе традиционных общечеловеческих ценностей.

ОК 07 Содействовать сохранению окружающей среды, ресурсосбережению, эффективно действовать в чрезвычайных ситуациях.

ОК 08 Использовать средства физической культуры для сохранения и укрепления здоровья в процессе профессиональной деятельности и поддержания необходимого уровня физической подготовленности.

ОК 09 Использовать информационные технологии в профессиональной деятельности

ОК 10 Пользоваться профессиональной документацией на государственном и иностранном языках.

ОК 11 Планировать предпринимательскую деятельность в профессиональной сфере

ПК 2.1 Администрировать локальные вычислительные сети и принимать меры по устранению возможных сбоев.

ПК 2.2 Администрировать сетевые ресурсы в информационных системах.

ПК 2.3. Обеспечивать сбор данных для анализа использования и функционирования программно-технических средств компьютерных сетей.

ПК 2.4. Взаимодействовать со специалистами смежного профиля при разработке методов, средств и технологий применения объектов профессиональной деятельности

В методических рекомендациях предлагаются к выполнению практические работы, предусмотренные учебной рабочей программой по МДК.02.03 «Организация администрирования компьютерных систем».

При разработке содержания практических работ учитывался уровень сложности освоения студентами соответствующей темы, общих и профессиональных компетенций, на формирование которых направлена дисциплина.

Выполнение практических работ в рамках МДК.02.03 «Организация администрирования компьютерных систем» позволяет освоить комплекс работ по выполнению практических заданий по всем темам МДК.02.03 «Организация администрирования компьютерных систем».

Рабочая тетрадь по МДК.02.03 «Организация администрирования компьютерных систем» имеют практическую направленность и значимость. Формируемые в процессе практических занятий умения могут быть использованы студентами в будущей профессиональной деятельности.

Рабочая тетрадь предназначена для студентов колледжа, изучающих МДК.02.03 «Организация администрирования компьютерных систем».

Оценки за выполнение практических работ выставляются по пятибалльной системе. Оценки за практические работы являются обязательными текущими оценками по учебной дисциплине и выставляются в журнале теоретического обучения.

1. Перечень практических работ по МДК 02.03 «Организация администрирования компьютерных систем»

№ раздела, темы	Освоение умений в процессе занятия	Формируемые ОК и ПК	Тема практического занятия	Кол-во часов
Тема 1. Проектирование и реализация серверной инфраструктуры	– администрировать локальные вычислительные сети; – принимать меры по устранению возможных сбоев; – обеспечивать защиту при подключении к информационно-телекоммуникационной сети "Интернет".	ОК 01-11 ПК 2.1-2.4	Практическая работа№ 1 Конфигурация программного обеспечения на серверах и рабочих станциях	2
			Практическая работа№ 2 Установка прав доступа и контроль использования сетевых ресурсов	2
			Практическая работа№ 3 Администрирование серверов	2
			Практическая работа№ 4 Расчёт стоимости сетевого оборудования и программного обеспечения	2
			Практическая работа№ 5 Регистрация пользователей локальной сети	2
			Практическая работа№ 6 Осуществление антивирусной защиты	2
			Практическая работа№ 7 Администрирование рабочих станций	2
			Практическая работа№ 8 Организация доступа к локальным сетям и Интернету	2
			Практическая работа№ 9 Установка и сопровождение сетевых сервисов	2
Тема 2. Реализация продвинутой серверной инфраструктуры	– администрировать локальные вычислительные сети; – принимать меры по устранению возможных сбоев; – обеспечивать защиту при под-	ОК 01-11 ПК 2.1-2.5	Практическая работа№ 10 Сбор данных для анализа использования программно-технических средств компьютерных сетей	2
			Практическая работа№ 11 Обеспечение сетевой безопасности	2
			Практическая работа№ 12 Проведение мониторинга сети	2

№ раздела, темы	Освоение умений в процессе занятия	Формируемые ОК и ПК	Тема практического занятия	Кол-во часов
	ключении к информационно-телекоммуникационной сети "Интернет".		Практическая работа№ 13 Принятие мер по восстановлению работоспособности локальной сети при сбоях или выходе из строя сетевого оборудования.	2
			Практическая работа№ 14 Выявление ошибок пользователей и программного обеспечения и принятие мер по их исправлению.	2
			Практическая работа№ 15 Обеспечение своевременного копирования, архивирования и резервирования данных.	2
			Практическая работа№ 16 Планирование и реализация стратегии виртуализации серверов	2
			Практическая работа№ 17 Планирование и реализация сетевой инфраструктуры и систем хранения данных для виртуализации	2
			Практическая работа№ 18 Планирование и развертывание виртуальных машин	2
			Практическая работа№ 19 Планирование и реализация решения по администрированию виртуализации	2
Тема 3. Администрирование серверов с ОС Linux	– устанавливать и настраивать серверы на базе ОС Debian – устанавливать и настраивать серверы на базе CentOS	ОК 01-11 ПК 2.1-2.5	Практическая работа№ 20 Установка сервера Debian	2
			Практическая работа№ 21 Настройка web-сервера в ОС Debian	2
			Практическая работа№ 22 Настройка сервера DNS в ОС Debian	2
			Практическая работа№ 23 Настройка сервера DHCP в ОС Debian	2

№ раздела, темы	Освоение умений в процессе занятия	Формируемые ОК и ПК	Тема практического занятия	Кол-во часов
			Практическая работа№ 24 Настройка файловых серверов в ОС Debian	2
			Практическая работа№ 25 Настройка контейнеров Docker	2
			Практическая работа№ 26 Установка сервера CentOS	2
			Практическая работа№ 27 Настройка web-сервера в CentOS	2
			Практическая работа№ 28 Настройка сервера DNS в CentOS	2
			Практическая работа№ 29 Настройка сервера DHCP в CentOS	2
			Практическая работа№ 30 Установка и настройка OpenVPN	2
			Практическая работа№ 31 Применение протокола IPsec и SSH	2
			Практическая работа№ 32 Настройка регистрации действий	2
			Практическая работа№ 33 Установка и настройка OpenLDAP	2
			Практическая работа№ 34 Установка и настройка IPtables	2
			Практическая работа№ 35 Поиск уязвимостей информационных систем	2

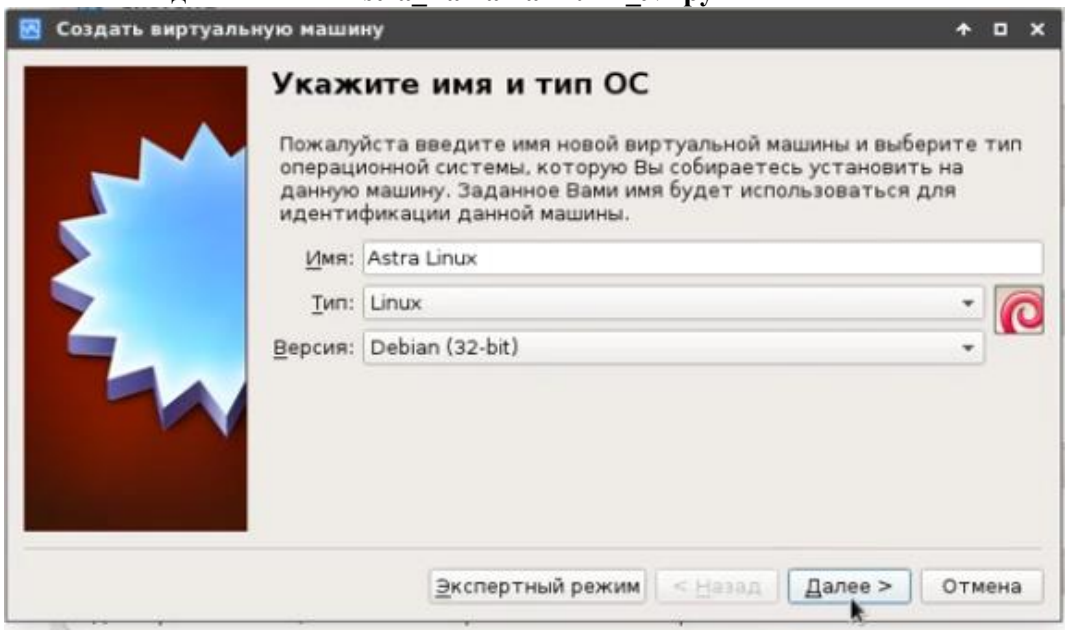
2. Описание порядка выполнения практических работ

2.1 Практическая работа№ 1
Конфигурация программного обеспечения на серверах и рабочих станциях. Установка прав доступа и контроль использования сетевых ресурсов

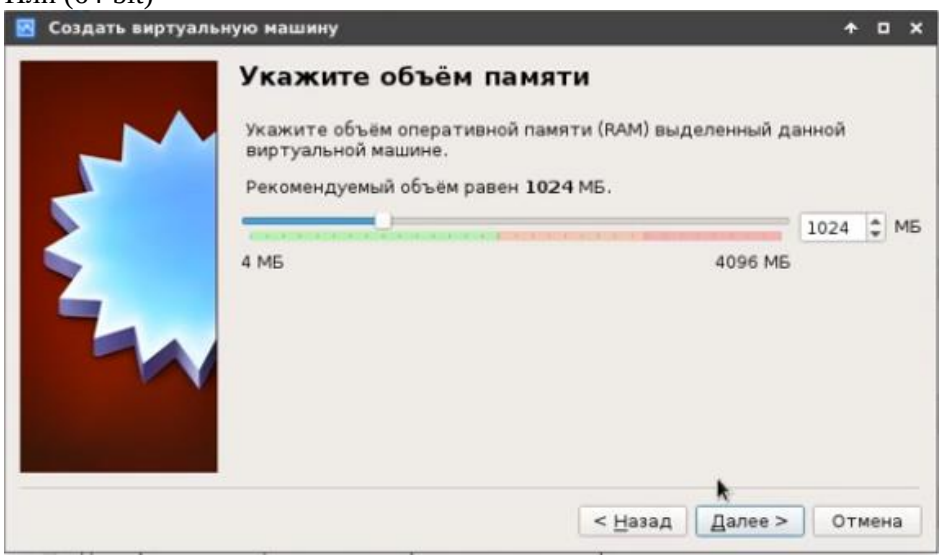
Задание 1:
Установка Astra Linux.

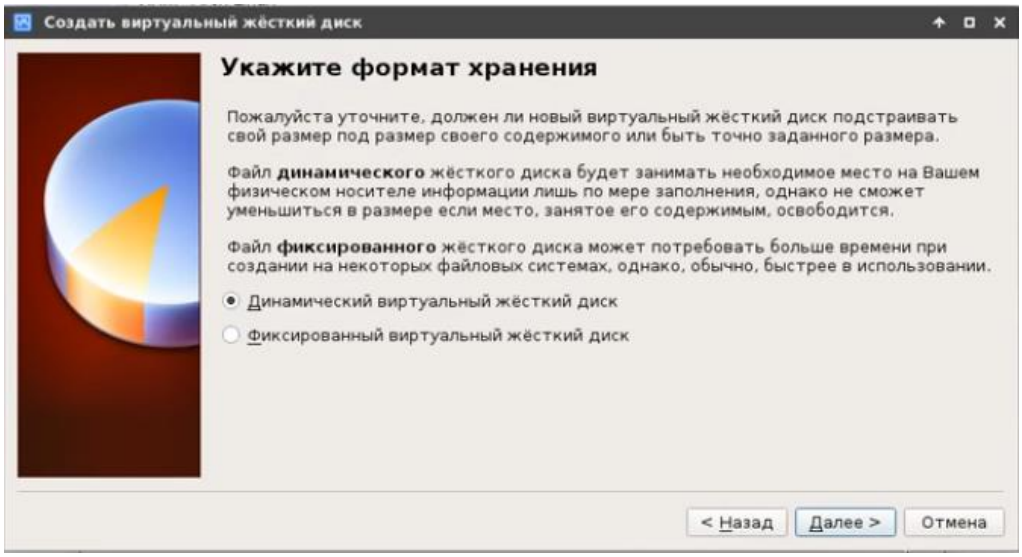
1. Запустить VirtualBox. Выбрать Создать. Далее указать параметры, показанные ниже на рисунках.

Имя машине должно быть **Astra_ВашаФамилия_№группы**

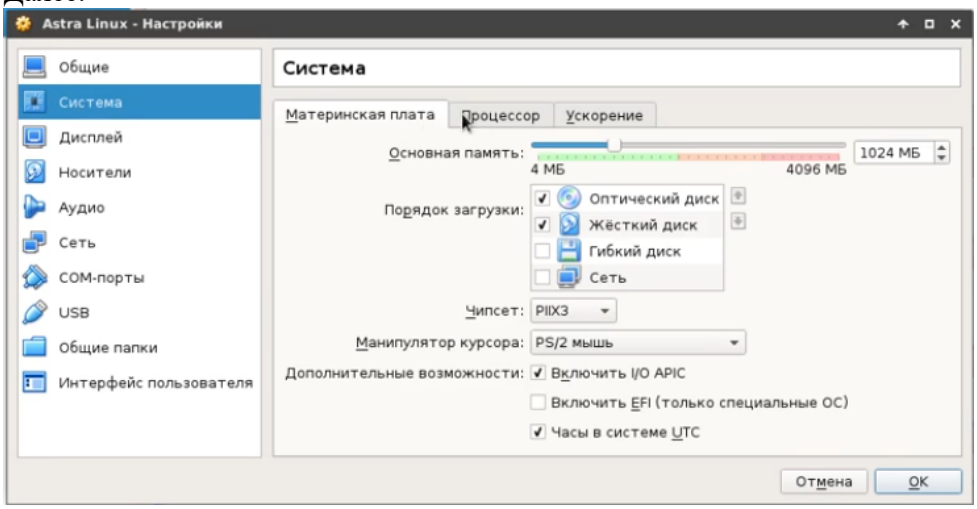


Или (64-bit)

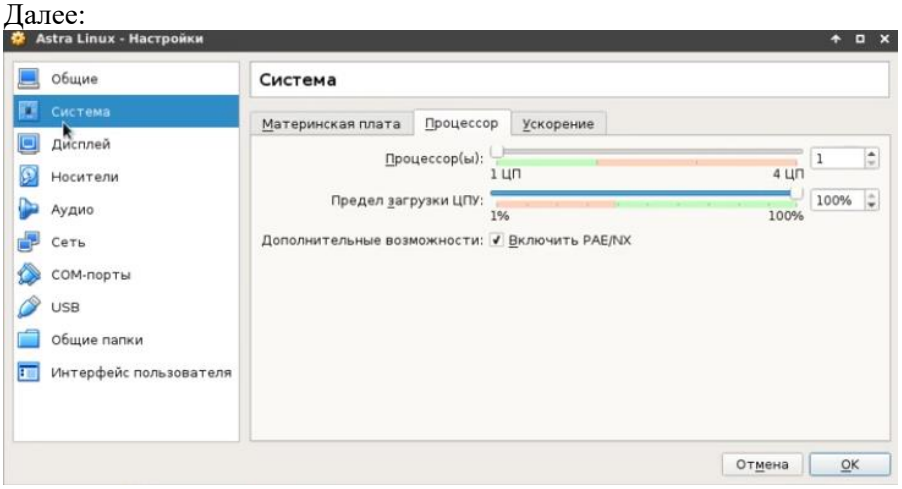




Далее:



Далее:



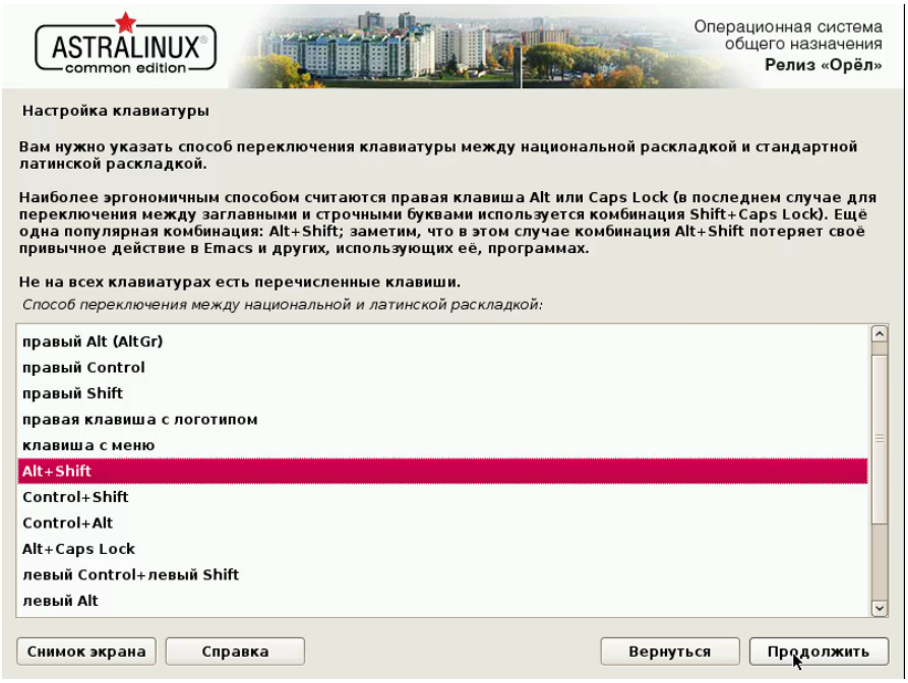
Далее необходимо указать, где находится файл *.iso

Запуск установщика

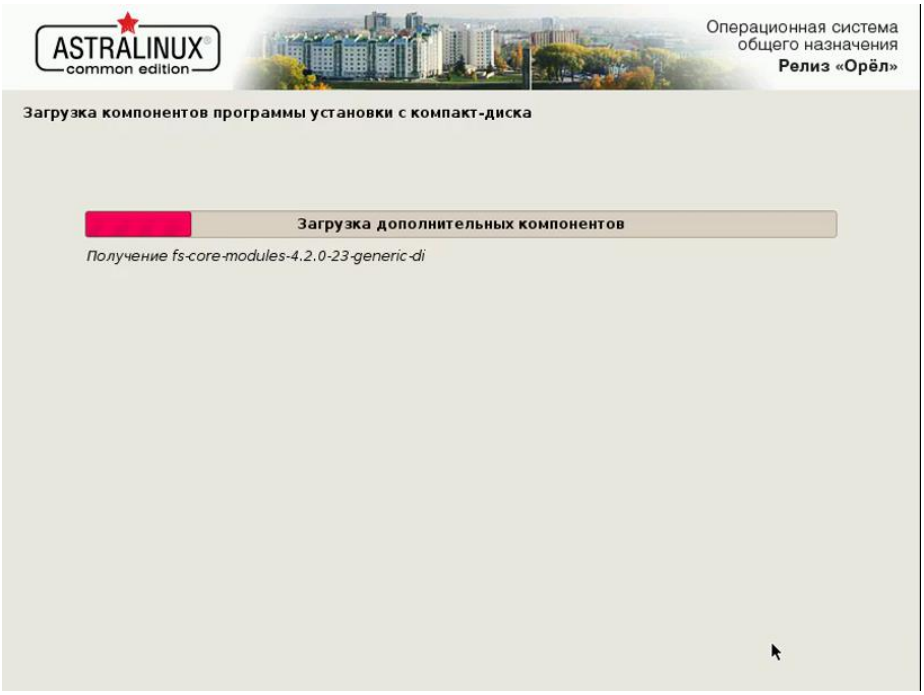
Сразу же после перезагрузки появится меню выбора способа установки. Выберите **"Графическая установка"**:

На первом шаге установщика вам нужно принять лицензионное соглашение разработчиков:

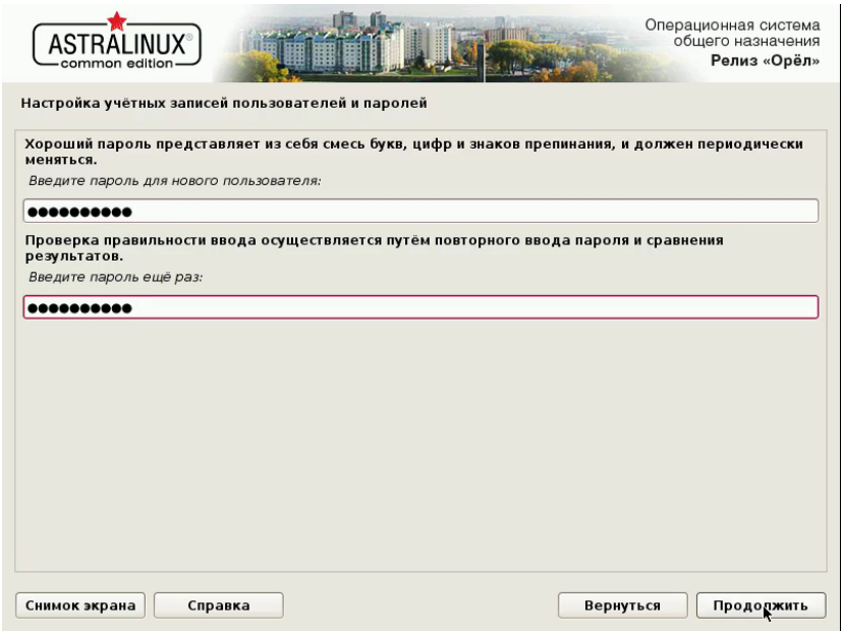
Выберите клавишу, с помощью которой будет переключаться раскладка:



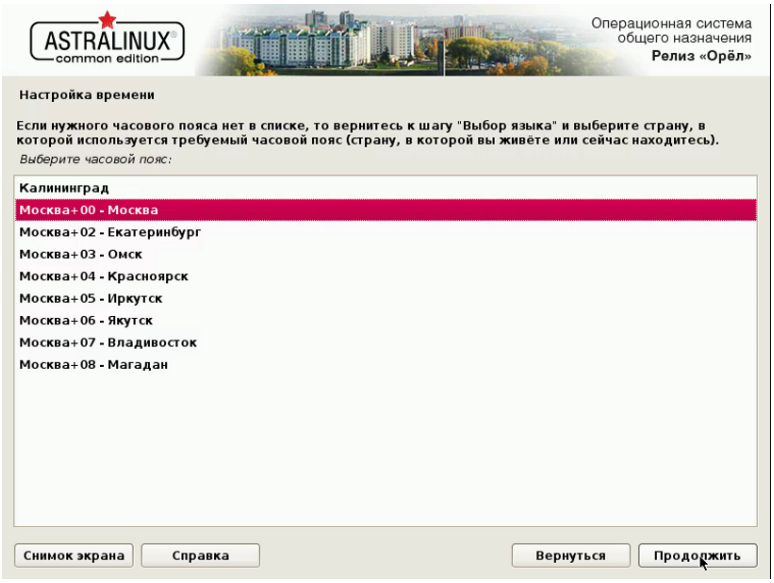
Дождитесь загрузки всех необходимых компонентов:



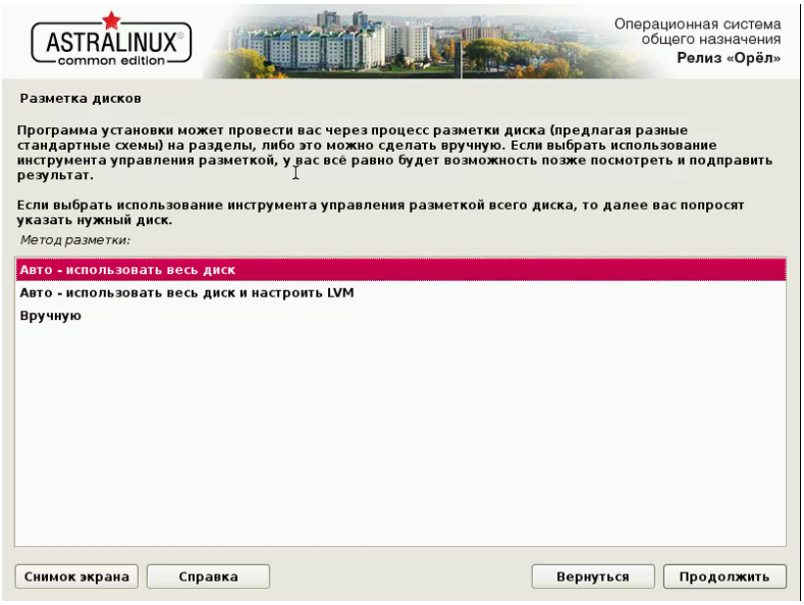
Введите имя компьютера, оно будет использоваться для обнаружения компьютера в локальной сети:



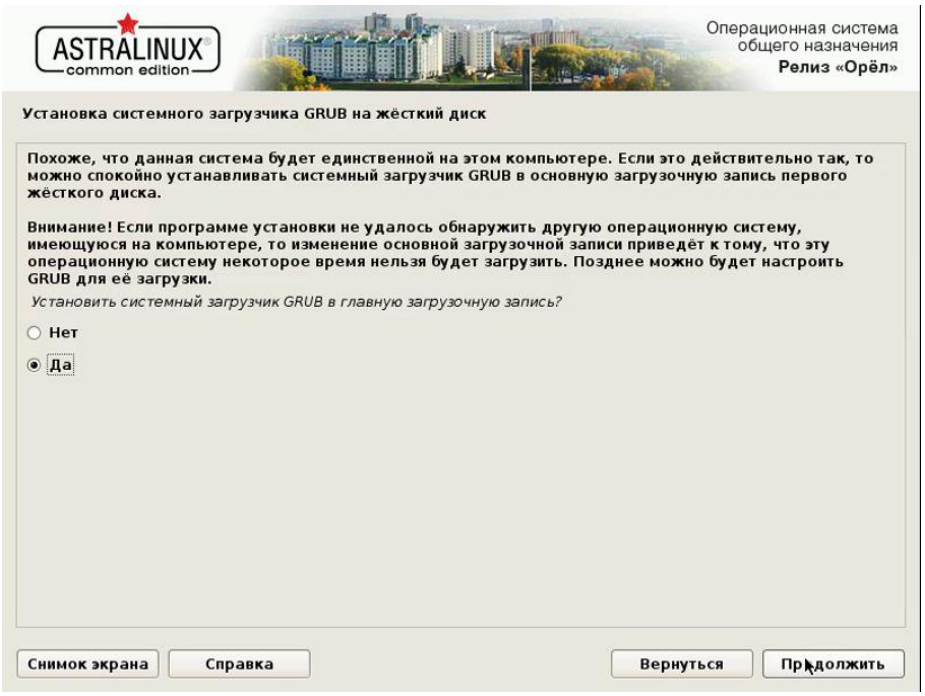
Выберите ваш часовой пояс:

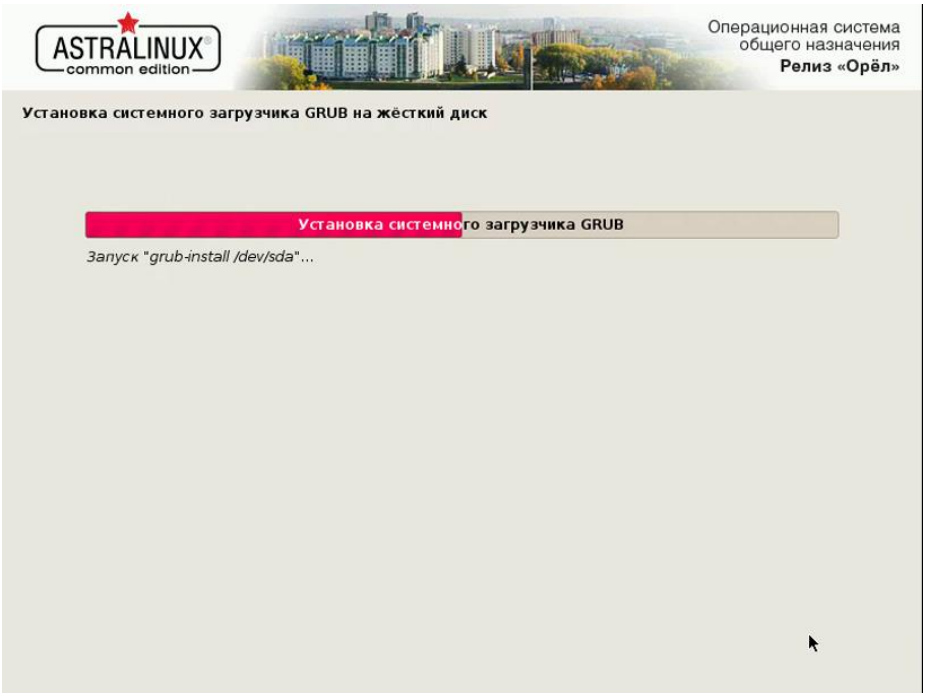


Дальше нужно выполнить разметку диска, вы можете выбрать автоматический вариант.



Укажите, нужно ли устанавливать загрузчик Grub. Если на вашем компьютере установлен только один Linux дистрибутив, то установка загрузчика обязательна.





Установка Astra Linux VirtualBox или на жесткий диск завершена и теперь вы можете перезагрузить компьютер, чтобы пользоваться новой системой.

Выберите пункт, подчеркнутый по умолчанию, версия ядра generic, в меню загрузчика Grub:

Вставьте скриншот с установленной операционной системой Astra.

Установка ОС Windows

1. Запустить VirtualBox. Выбрать Создать.
   Имя машине должно быть Win10_ВашаФамилия_№группы
2. Установить из *.iso файла Windows 10
3. Вставьте скриншот с установленной операционной системой Windows.

Установка ОС Ubuntu

1. Запустить VirtualBox. Выбрать Создать. Далее указать параметры, показанные ниже на рисунках.
   Имя машине должно быть Ubuntu_ВашаФамилия_№группы
2. Установить из *.iso файла Ubuntu

3. Вставьте скриншот с установленной операционной системой Ubuntu.

Задание 2:

1. Открыть Центр управления сетями → Изменение параметров адаптера. Просмотреть Сетевые подключения. В отчёт вставить скриншот окна с Состоянием.
2. В окне Состояние открыть Свойства. Перейти на IP версии 4.
   Измените настройки с Автоматических.
   IP-адрес выберите из диапазона 192.168.6.50 – 192.168.6.70.
   Маска 255.255.255.0.
   Шлюз — адрес вашей host-машины.
   DNS-сервер — такой же как у host-машины.
   В отчёт скриншот с вашими настройками.
3. Откройте свойства IP версии 6. Просмотрите его настройки.
   В отчёте напишите примерный вид первых трёх полей с комментариями.
4. Выйдите в Центр управления сетями и общим доступом. Перейдите в Изменить дополнительные параметры общего доступа → Включите общий доступ к файлам и принтерам. В отчёт вставьте скриншот с выполненным заданием.
5. Выключите виртуальную машину с Windows.
   Откройте Настройки вашей виртуальной машины с Windows → Сеть → Адаптер 2. Включите сетевой адаптер с типом подключения Виртуальный адаптер хоста. Далее откройте Дополнительно → Обновите Mac-адрес.
   В отчёт скриншот с выполненным заданием.
6. Включите виртуальную машину с Windows.
   Откройте Центр управления сетями и общим доступом → Изменение параметров адаптера.
   В отчёт поместите скриншот и ответ на вопросы:
   — какой адаптер был подключен в предыдущем пункте?
   — для чего иногда нужно включать несколько сетевых адаптеров?
7. Выключите виртуальную машину с Windows. Включите виртуальную машину с Linux.
8. Откройте терминал в Linux.
   Начнём с информации об имени сетевого адаптера с помощью команды:
   sudo lshw -C network
   Вставьте скриншот в отчёт и ответы на следующие вопросы по скриншоту:
   — фирма сетевой карты?
   — логическое имя?
   — поддерживаемая скорость сетевой карты?
   — разрядность сетевой карты и частота?
9. Для настройки вручную сетевых параметров необходимо отредактировать файл конфигурации /etc/network/interfaces. Это можно сделать с помощью команды
   sudo nano /etc/network/interfaces
   Далее пишите следующий текст, используя ip-адреса те же, что использовали на виртуальной машине Windows. Вместо eth0 логическое имя вашего адаптера из предыдущего пункта.
   iface eth0 inet static
   address ….
   netmask …
   gateway …
   dns-nameservers …
   auto eth0
   В отчёт вставьте скриншот с вашими настройками.
   Выполните команду **ifconfig** в терминале (если не установлено, установить).
   В отчёт скриншот с результатом.
10. Сохраните. Закройте.
    Выключите вашу виртуальную машину с Linux.
    Добавьте ей в настройках виртуальный сетевой адаптер.
    Включите снова.

Выполните команду **ifconfig** в терминале.
В отчёте ответьте на вопрос: что изменилось в информации ifconfig?

11. Верните обратно динамические настройки для вашего первого адаптера. Для этого нужно снова открыть файл конфигурации и заменить текст на:
iface eth0 inet dhcp
auto eth0
Сохраните. Выйдите. Выполните **ifconfig**.
Скриншот с результатом в отчёт.

2.2 Практическая работа№ 2
Администрирование серверов.
Расчёт стоимости сетевого оборудования и программного обеспечения

Задание 1:

1. Установка и настройка ADRMS на Windows Server 2012 R2

1. Служба **Active Directory Right Management Services** – одна из стандартных ролей Windows Server, позволяющая организовать защиту пользовательских данных от несанкционированного использования. Защита информации реализуется за счет шифрования и подписывания документов, причем владелец документа или файла может сам определить, каким пользователям можно открывать, редактировать, распечатывать, пересылать и выполнять другие операции с защищенной информацией. Нужно понимать, что защита документов с помощью ADRMS возможно только в приложениях, разработанных с учетом этой службы (AD RMS-enabled applications). Благодаря AD RMS можно обеспечить защиту конфиденциальных данных как внутри, так и за пределами корпоративной сети.

Несколько важных требования, которые нужно учесть при планировании и развертывании решения AD RMS:

- Желательно использовать выделенный сервер AD RMS. Не рекомендуется совмещать роль AD RMS с ролью контроллера домена, сервера Exchange, SharePoint Server или центра сертификации (CA)
- У пользователей AD должен быть заполнен атрибут email
- На компьютерах пользователей RMS сервер должен быть добавлен в зону доверенных сайтов IE (Trusted Sites). Проще всего это сделать с помощью групповой политики.

2. Прежде чем приступить непосредственно к развертыванию ADRMS, нужно выполнить ряд подготовительных шагов. В первую очередь необходимо создать в Active Directory отдельную сервисную запись для ADRMS с бессрочным паролем, например с именем svcadrms (для службы ADRMS можно создать и особую управляемую учетную запись AD — типа gMSA).

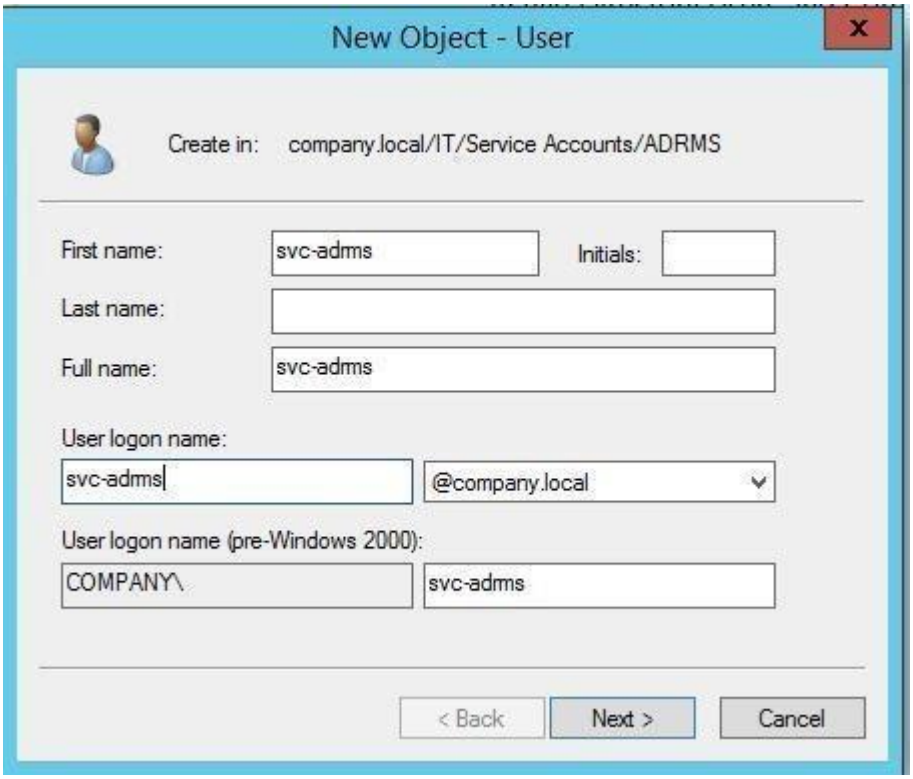


Рис. 59

3. В DNS-зоне создадим отдельную ресурсную запись, указывающую на AD RMS сервер. Допустим его имя будет – **adrms**.

Рис. 60

4. Приступим к установке роли ADRMS на сервере с Windows Server 2012 R2. Откройте консоль Serve Manager и установите роль **Active Directory Rights Management Service** (здесь все просто – просто соглашайтесь с настройками и зависимостями поумолчанию).

Рис. 61

5. После того, как установка роли ADRMS и сопутствующих ей ролей, и функций закончится, чтобы перейти в режим настройки роли ADRMS, щелкните по ссылке **Perform additional configuration**.

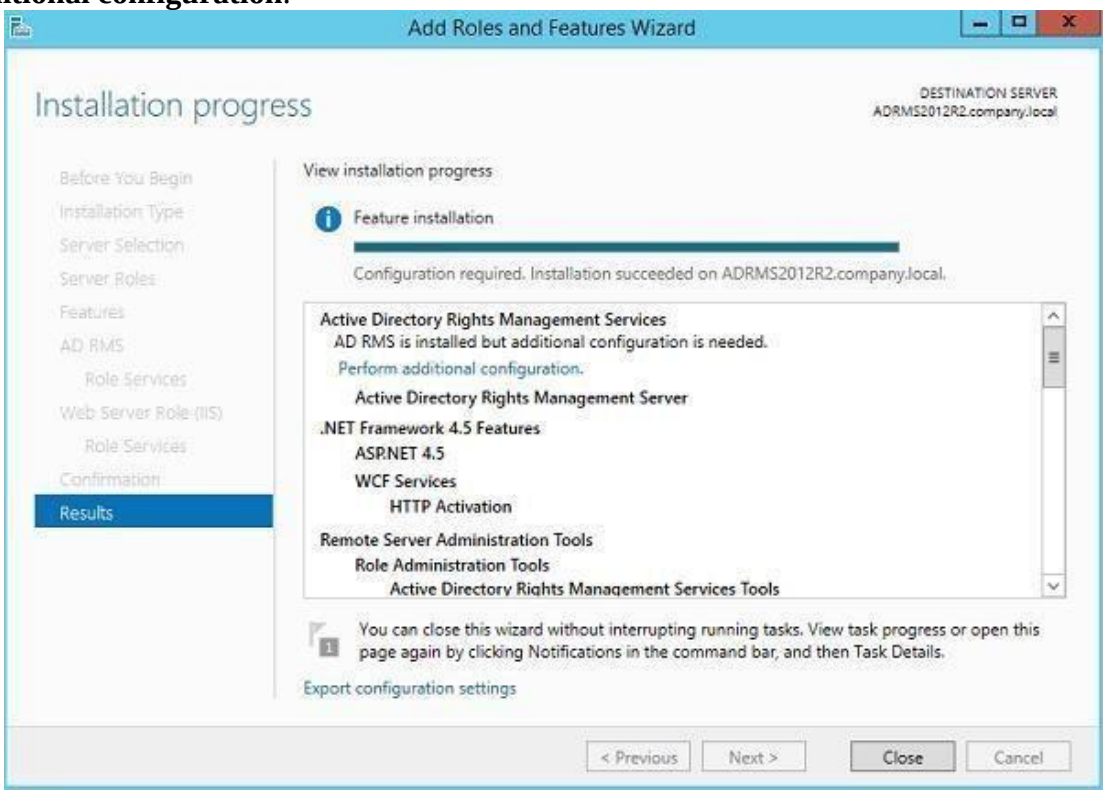


Рис. 62

6. В мастере настройки выберем, что мы создаем новый корневой кластер AD RMS (**Create a new AD RMS root cluster**).

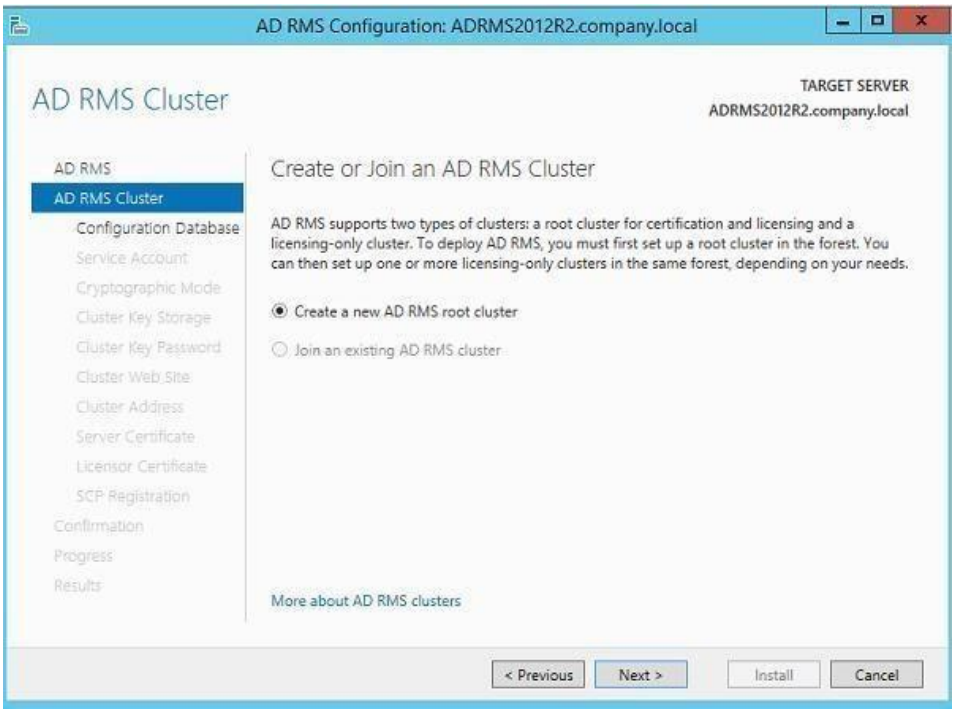
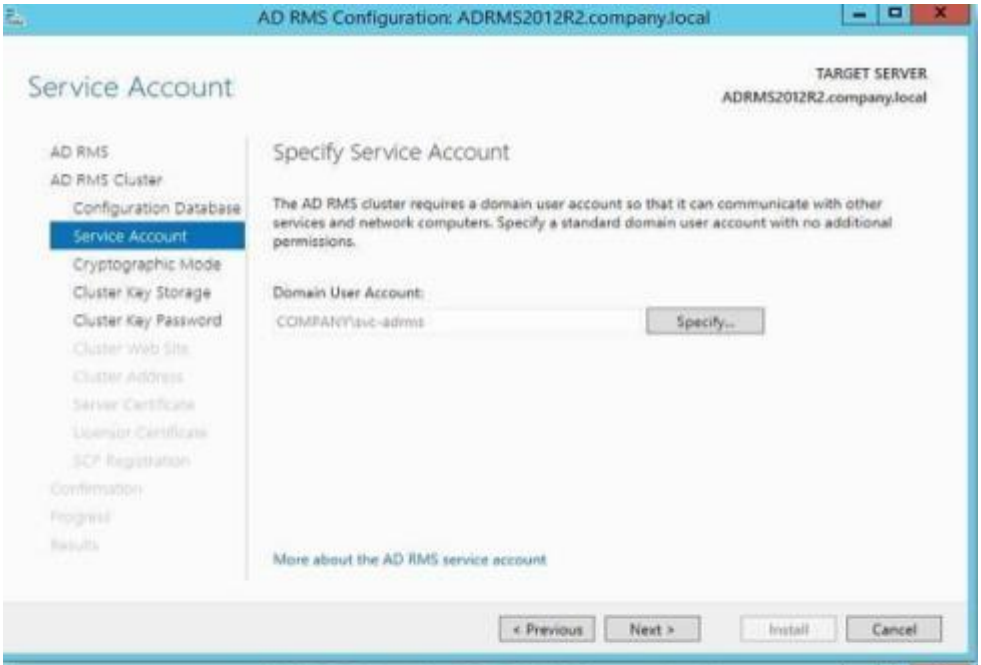


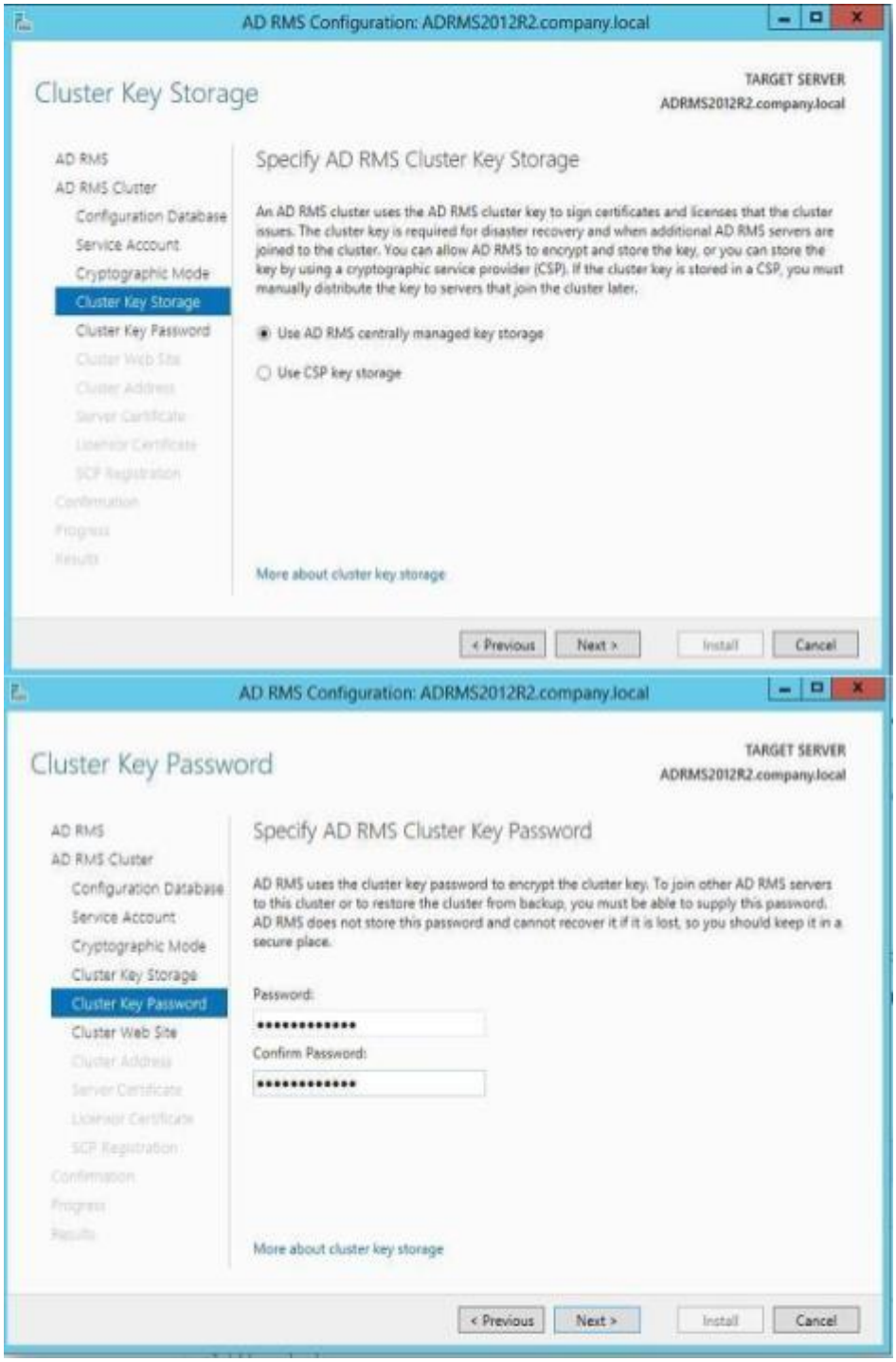
Рис. 63

7. В качестве базы данных RMS будем использовать внутреннюю базу данных Windows (**Use Windows Internal Database on this server).**

8. Затем укажем созданную ранее сервисную учетную запись (svc-adrms), используемый криптографический алгоритм, метод хранения ключа кластера RMS и его пароль.


AD RMS Configuration: ADRMS2012R2.company.local
Service Account
TARGET SERVER
ADRMS2012R2.company.local
AD RMS
AD RMS Cluster
Configuration Database
Service Account
Cryptographic Mode
Cluster Key Storage
Cluster Key Password
Cluster Web Site
Cluster Address
Server Certificate
Licensor Certificate
SCP Registration
Confirmation
Progress
Results
Specify Service Account
The AD RMS cluster requires a domain user account so that it can communicate with other services and network computers. Specify a standard domain user account with no additional permissions.
Domain User Account:
COMPANY\svc-adms
Specify...
More about the AD RMS service account
< Previous
Next >
Install
Cancel


AD RMS Configuration: ADRMS2012R2.company.local
Cryptographic Mode
TARGET SERVER
ADRMS2012R2.company.local
AD RMS
AD RMS Cluster
Configuration Database
Service Account
Cryptographic Mode
Cluster Key Storage
Cluster Key Password
Cluster Web Site
Cluster Address
Server Certificate
Licensor Certificate
SCP Registration
Confirmation
Progress
Results
Specify Cryptographic Mode
AD RMS can operate under two modes which differ on the basis of the cryptographic key length and the strength of signature hashes. Cryptographic mode 2 is recommended for new cluster deployments where you have ensured that all AD RMS client computers have been updated to support it. As cryptographic mode 2 cannot be undone, if you are unsure of full support within this cluster or any other clusters that it will share a trusted user domain (TUD) relationship with, select cryptographic mode 1 instead.
Cryptographic Mode 2 (RSA 2048-bit keys/SHA-256 hashes)
Cryptographic Mode 1 (RSA 1024-bit keys/SHA-1 hashes)
More about the AD RMS cryptographic mode
< Previous
Next >
Install
Cancel



9. Задайте веб-адрес кластера AD RMS, к которому будут обращаться RMS-клиенты (рекомендуется использовать защищенное SSL соединение).

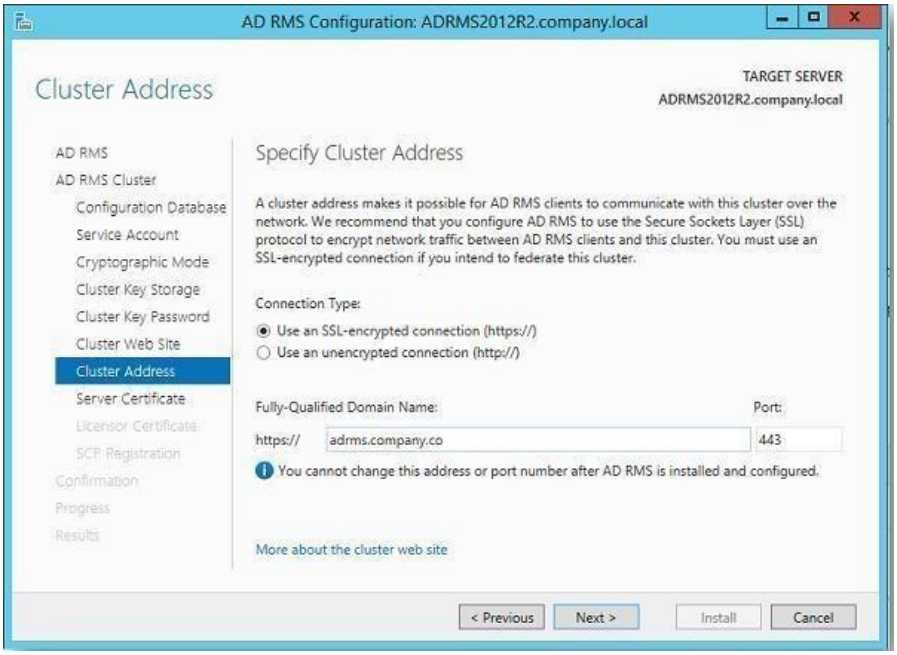
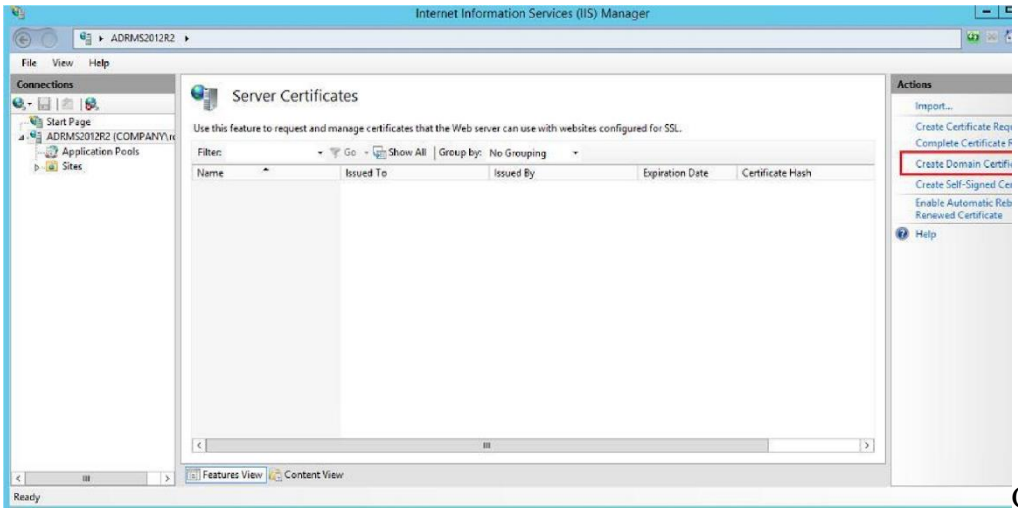


Рис. 64

Не закрывайте мастер настройки AD RMS!

10. Установка SSL-сертификата на сайт IIS. Сертификат может быть самоподписаным (в дальнейшем его нужно будет добавить в доверенные на всех клиентах), или выданным корпоративным/внешним центром сертификации (CA). Сформируем сертификат с помощью уже имеющегося корпоративного CA. Для этого откройте консоль IIS Manager (**inetmgr**) и перейдите в раздел Server Certificates. В правом столбце щелкните по ссылке **Create Domain Certificate** (создать сертификат домена).



Сгенерируйте новый сертификат с помощью мастера и привяжите его к серверу IIS.

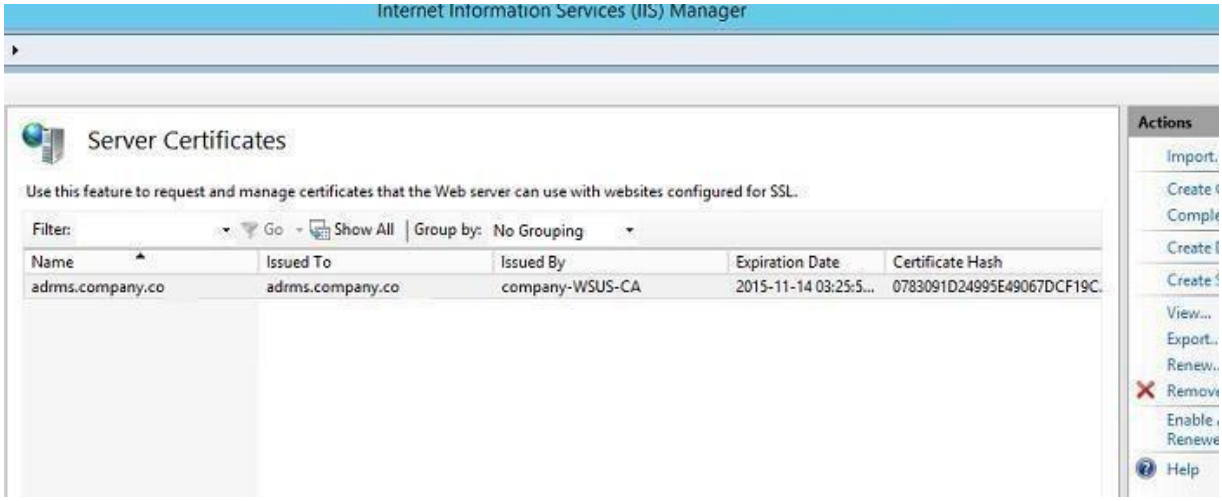


Рис. 65

11. Вернитесь в окно настройки роли AD RMS и выберите сертификат, который планируется использовать для шифрования трафика AD RMS.

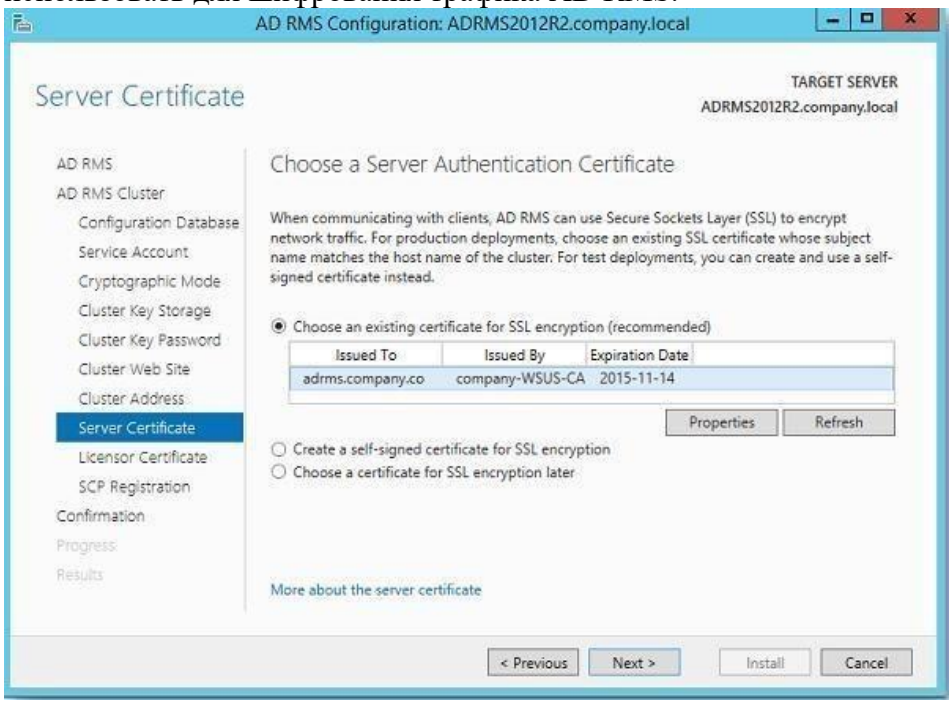


Рис. 66

12. Отметьте, что точку SCP нужно зарегистрировать в AD немедленно (**Register the SCP now**).

Примечание. Для регистрации точки SCP в Active Directory нужно обладать правами Enterprise Admins.

13. Запустите консоль ADRMS.

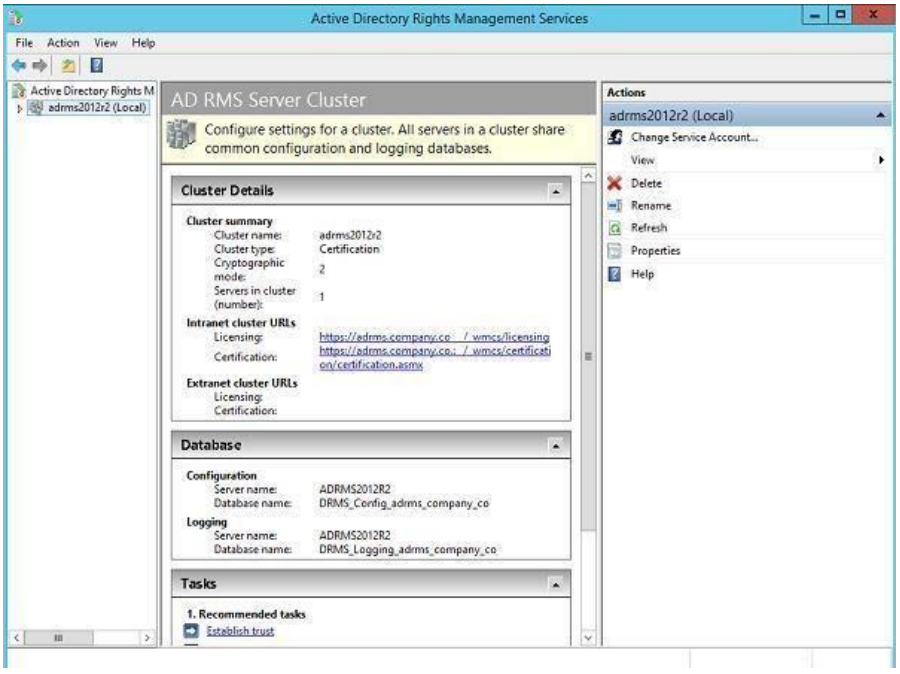


Рис. 67

Для примера создадим новый шаблон политики RMS. Предположим мы хотим создать шаблон RMS, позволяющий владельцу документа разрешить всем просмотр защищенных этим шаблоном писем без прав редактирования/пересылки. Для этого перейдем в раздел **Rights Policy Templates** и щелкнем по кнопке **Create Distributed Rights Policy Template**.

Нажав кнопку **Add**, добавим языки, поддерживаемые этим шаблоном и имя политики для каждого из языков.

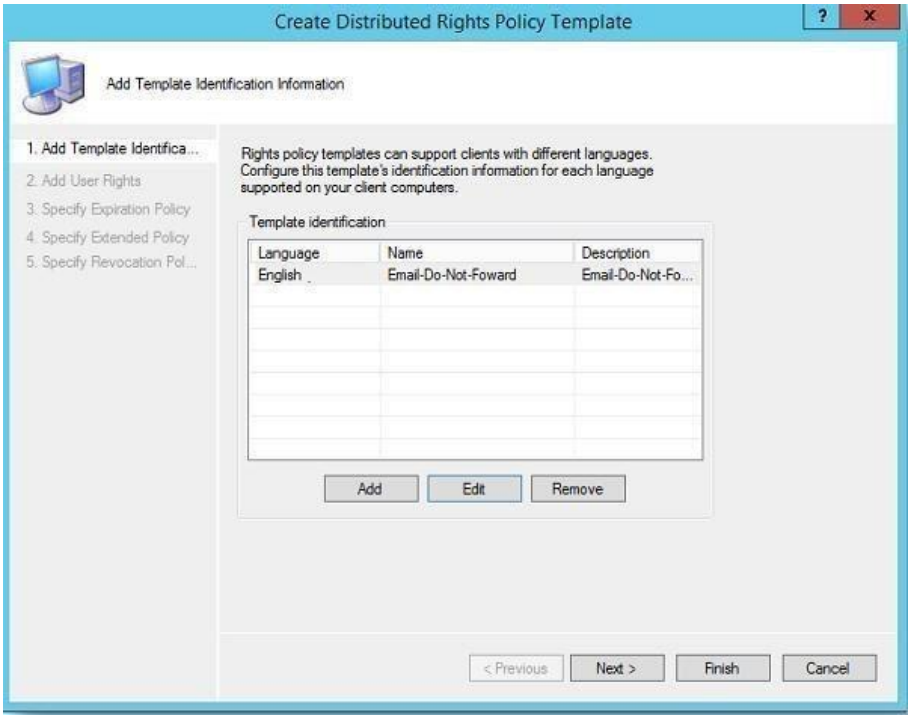


Рис. 68

14. Укажем, что все (**Anyone**) могут просматривать (**View**) содержимое защищенного автором документа.

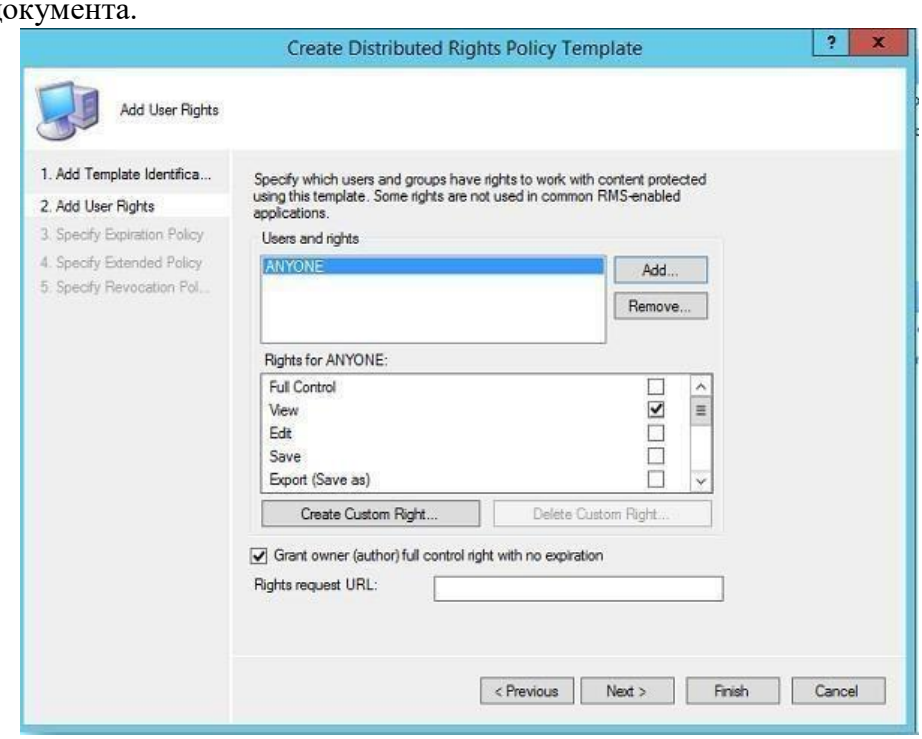


Рис. 69

15. Укажем, что срок окончания действия политики защиты не ограничен (**Never expires**).

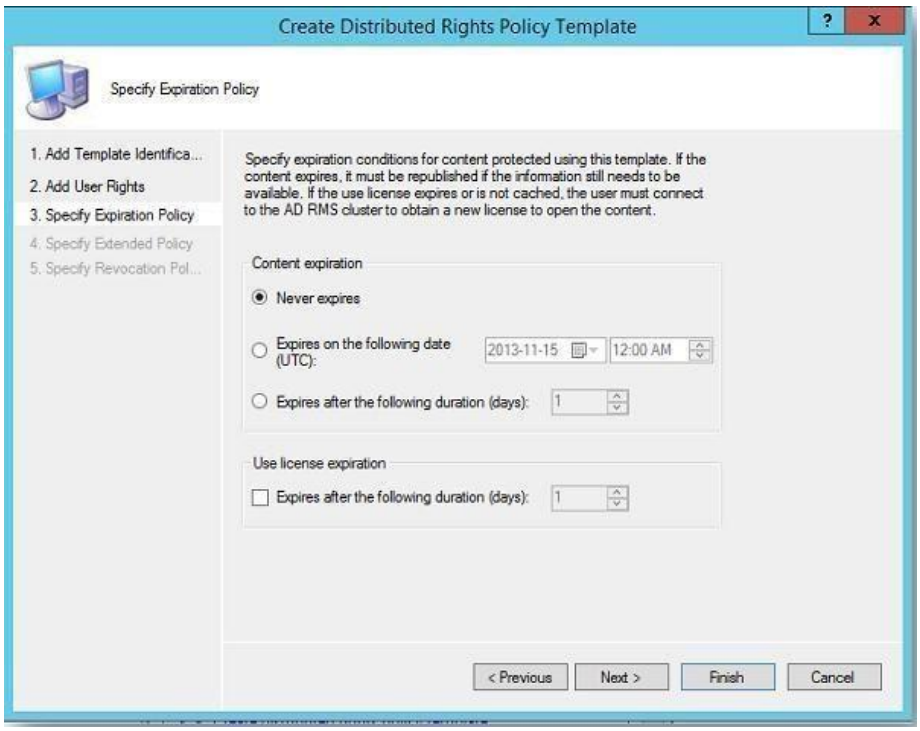


Рис. 70

16. На следующем шаге укажем, что защищенное содержимое можно просматривать в браузере с помощью расширений IE (**Enable users to view protected content using a browser add-on**).

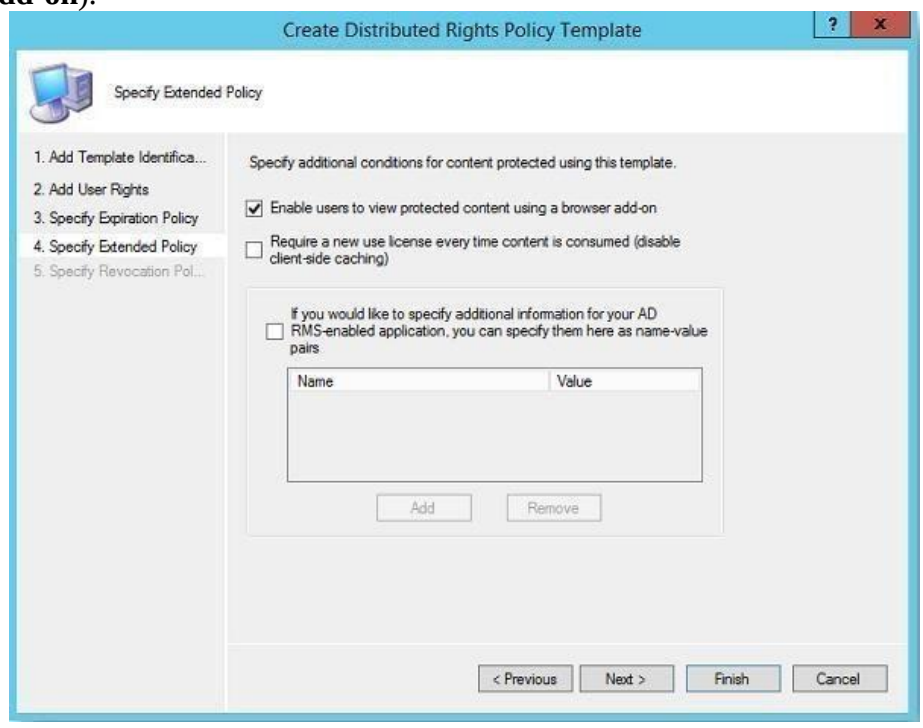
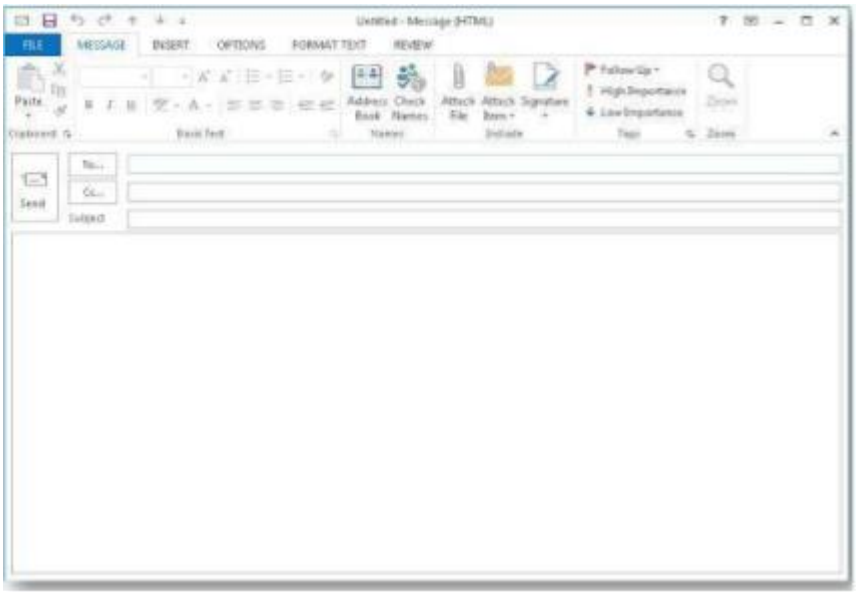
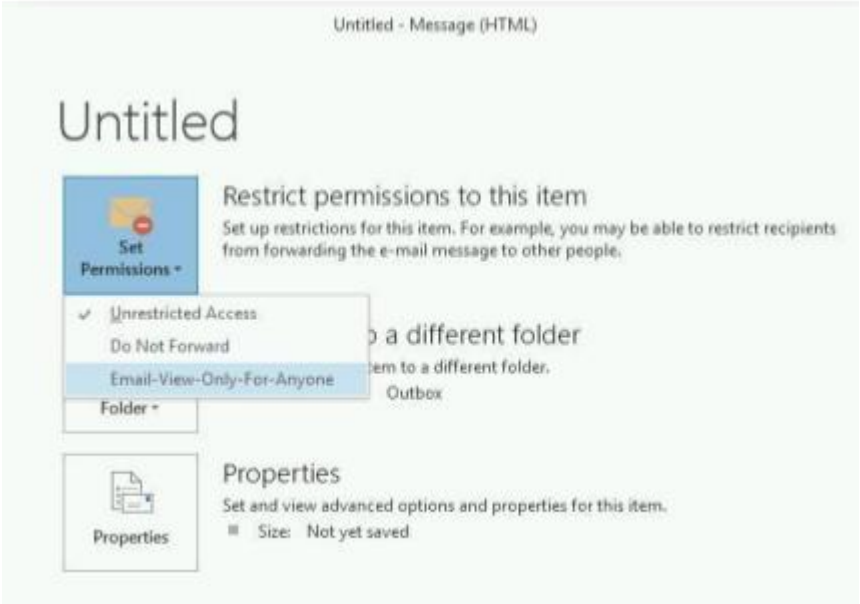


Рис. 71

Протестируем созданный шаблон RMS в **Outlook Web App**, для чего создадим новое пустое письмо, в свойствах которого нужно щелкнуть по кнопке **Set Permissions**. В выпадающем меню выберите имя шаблона (**Email-View-Onl-For-Anyone**).





Примечание. Если список шаблонов RMS открывается с ошибкой, или созданные шаблоны отсутствуют, проверьте что адрес сайта AD RMS относится к зоне Local Intranet /Trusted zone , а текущий пользователь может авторизоваться на IIS сервера RMS.

Отправим письмо, защищенное RMS, другому пользователю.

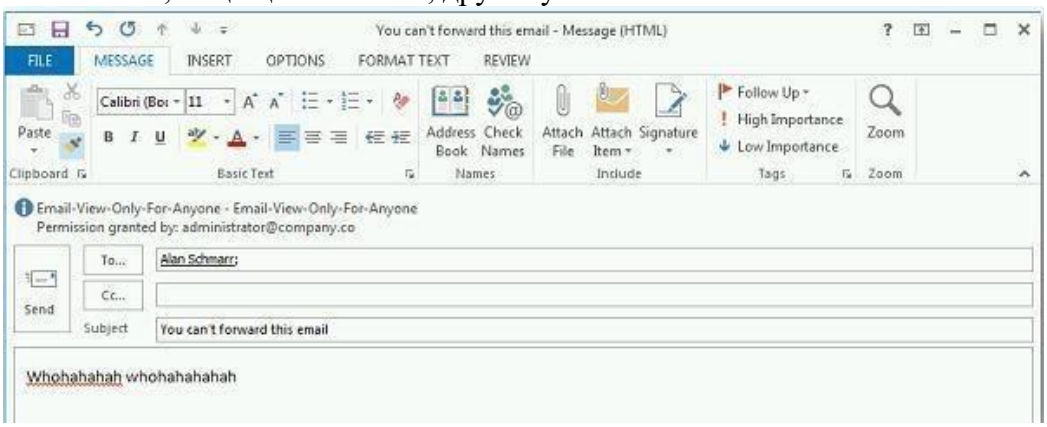


Рис. 72

17. Посмотрим, как выглядит защищенное письмо в ящике получателя.

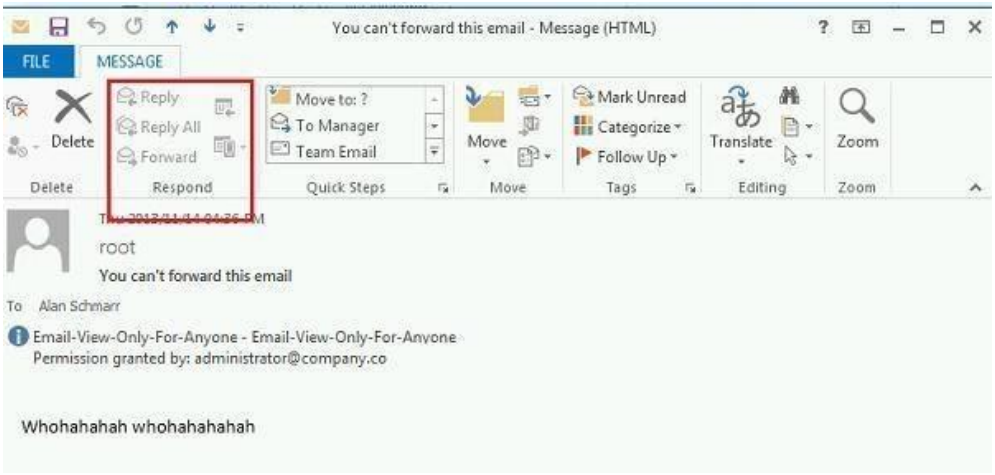


Рис. 73

Как мы видим, кнопки Ответить и Переслать недоступны, а в информационной панели указан используемый шаблон защиты документа и его владелец.

Сделайте скриншоты (фотографии) процесса настройки ADRMS и вставьте в отчёт

Задание 2:

1. Составьте список сетевого и компьютерного оборудования необходимого для комплектования одной учебной аудитории (компьютерного класса).
2. Составьте список программного обеспечения необходимого для комплектования одной учебной аудитории (компьютерного класса)
3. Используя открытые источники в сети интернет, составьте смету стоимости оборудования и ПО из пп. 1 и2

В качестве примера можно использовать вашу учебную аудиторию, установленное в нем оборудование, и ПО.

Сделайте скриншоты (фотографии) цен в интернет магазине и вставьте в отчёт

2.3 Практическая работа№ 3
Регистрация пользователей локальной сети. Осуществление антивирусной защиты

Задание 1:

1. Настройка имени сервера и статического IP-адреса
   1. Откройте **Пуск > Компьютер (пр. кнопкой мыши) > Свойства** (Рис.1).

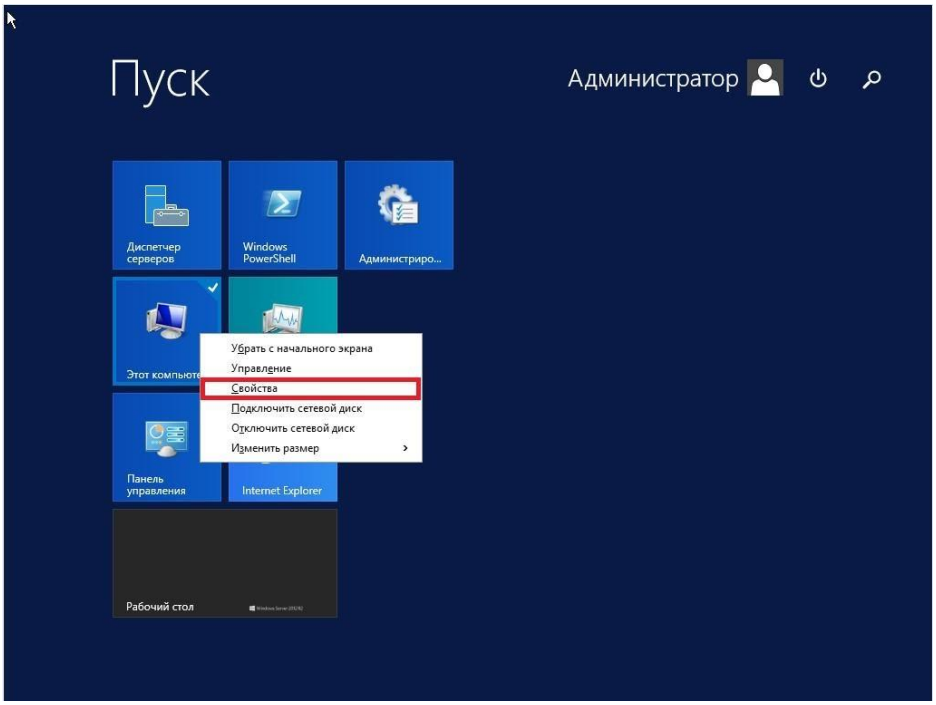


Рис. 12

2. В открывшемся окне выберите **Изменить параметры**.

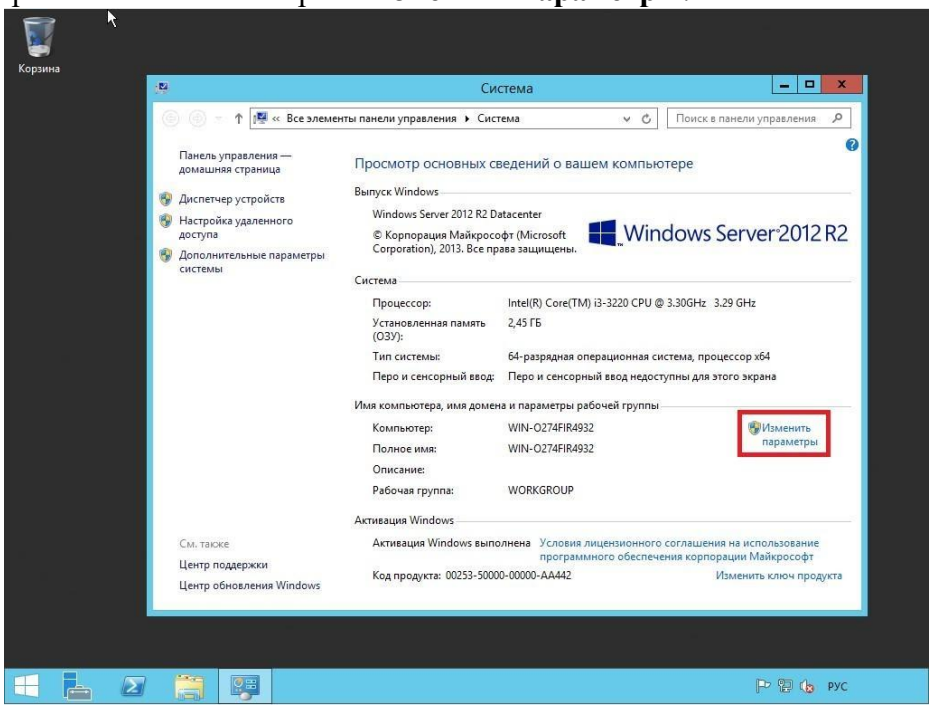


Рис. 13

3. В **Свойствах системы** выберите вкладку **Имя компьютера** и нажмите **Изменить…** . В появившемся окне укажите новое имя сервера в поле Имя компьютера (*прим. в данном руководстве это SERVER2012R2*), затем нажмите **ОК**.

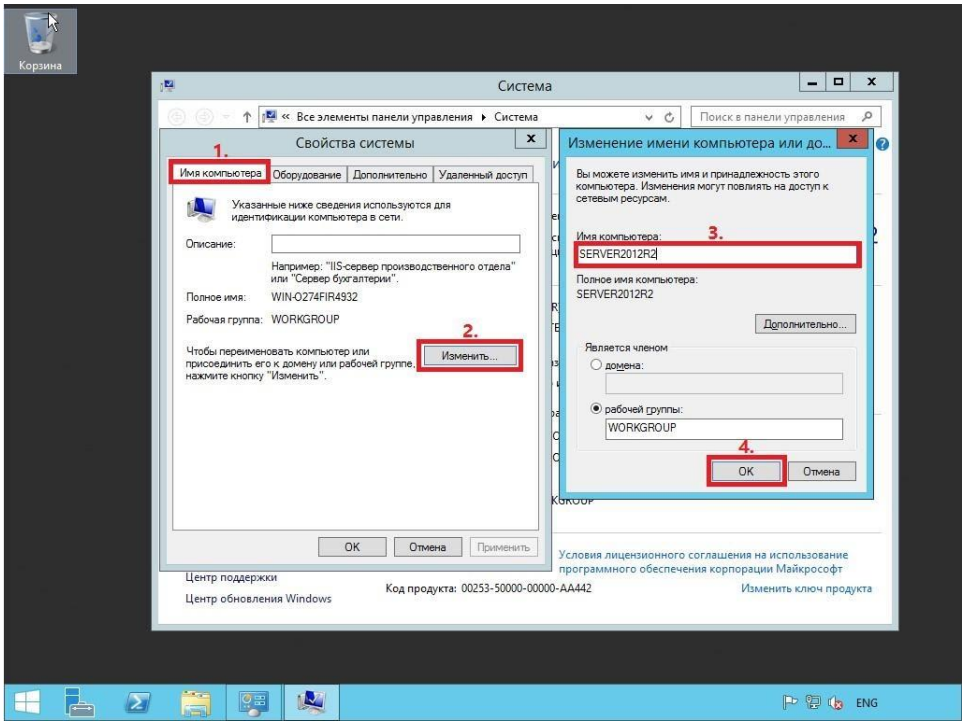


Рис. 14

4. Система предупредит о том, что для применения новых настроек необходимо перезагрузить сервер. Нажмите кнопку **ОК**

5.

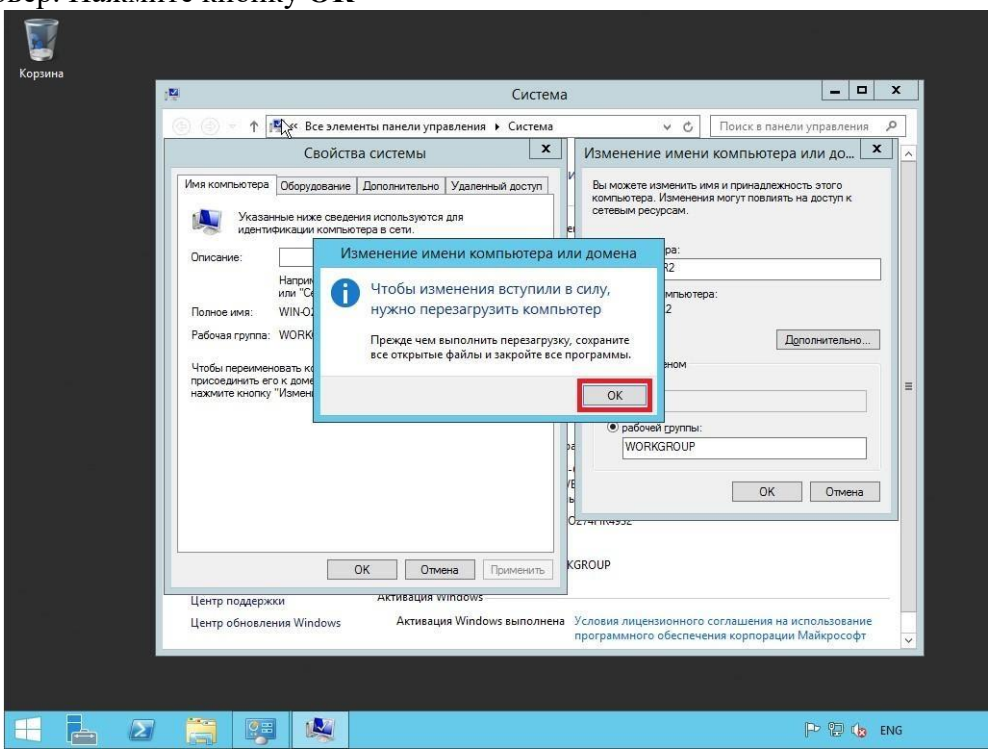


Рис. 15

6. После перезагрузки, в правом нижнем углу кликните (пр. кнопкой мыши) на иконке сетевого соединения. В открывшемся меню выберите **Центр управления сетями и общим доступом**

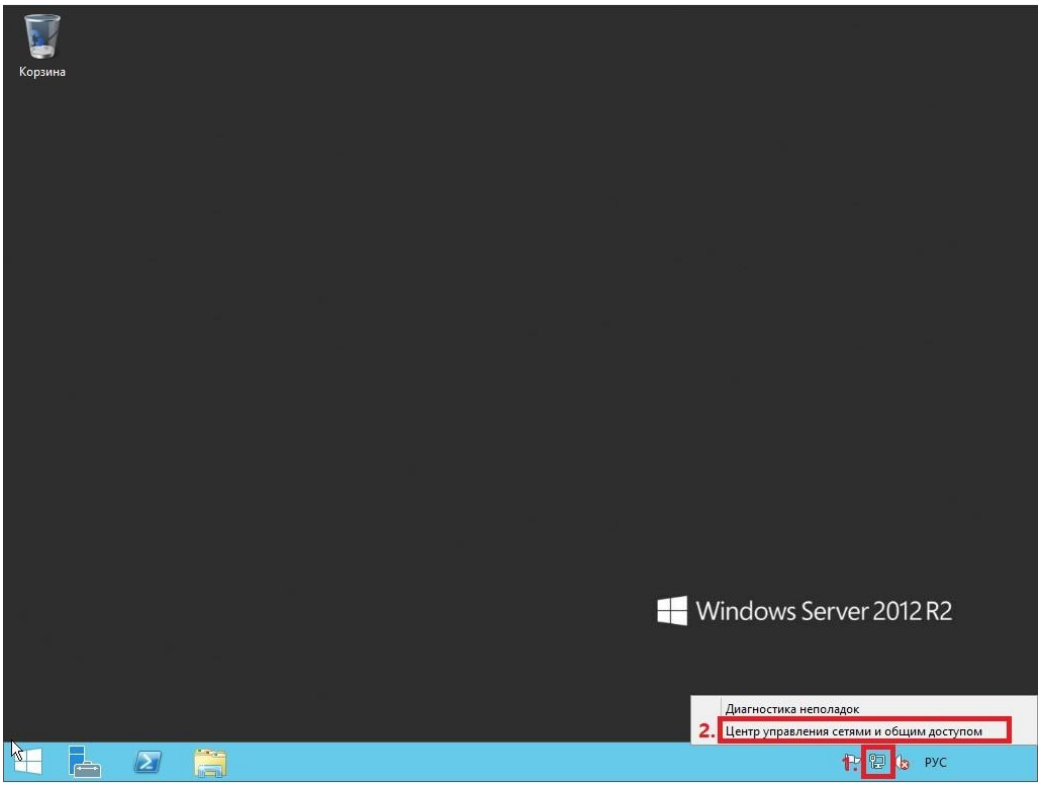


Рис. 16

7. В открывшемся окне выберите **Изменение параметров адаптера**

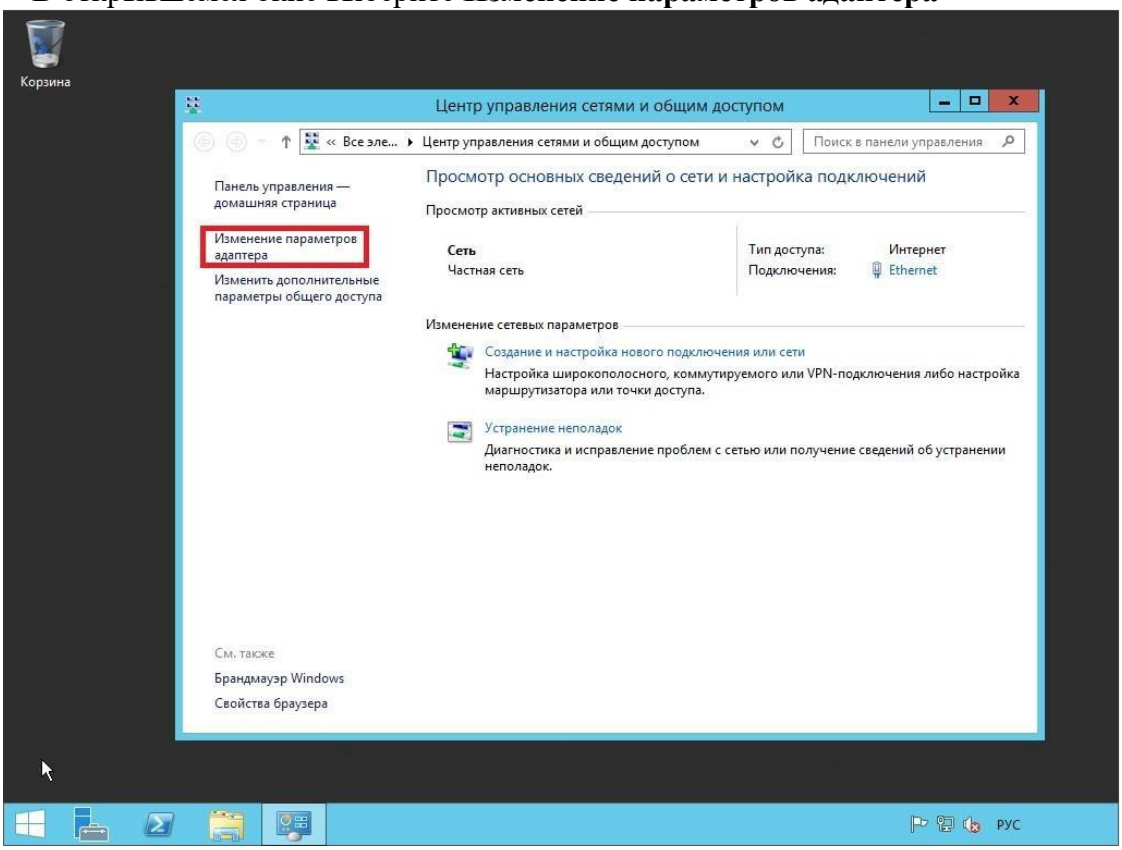


Рис. 17

8. В открывшемся окне Сетевые подключения нажмите правой кнопкой мыши на сетевом подключении и выберите пункт **Свойства**. В появившемся окне выделите **Протокол Интернета версии 4 (TCP/IPv4)** и нажмите **Свойства**

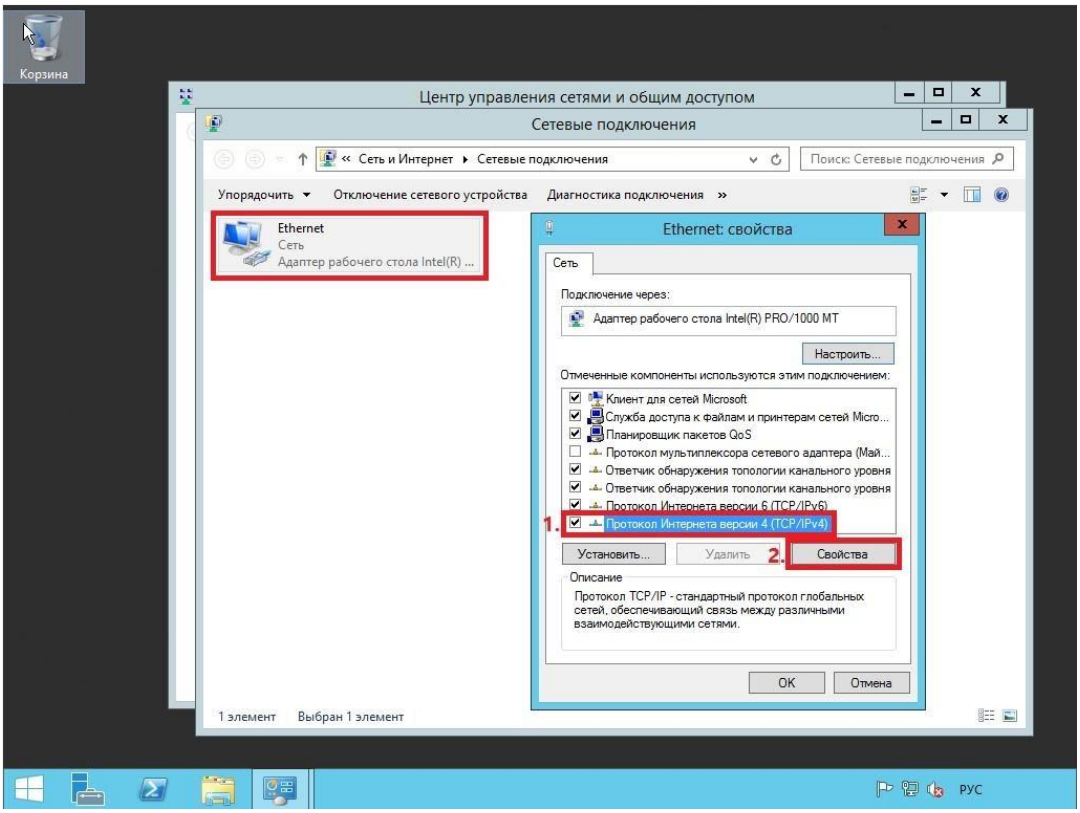


Рис. 18

9. В свойствах, на вкладке Общие выберите пункт **Использовать следующий IP-адрес**. В соответствующие поля введите **свободный IP-адрес**, **маску подсети** и **основной шлюз**. Затем выберите пункт **Использовать следующие адреса DNS-серверов**. В поле **предпочитаемый DNSсервер** введите **IP-адрес сервера**, после чего нажмите **ОК**.

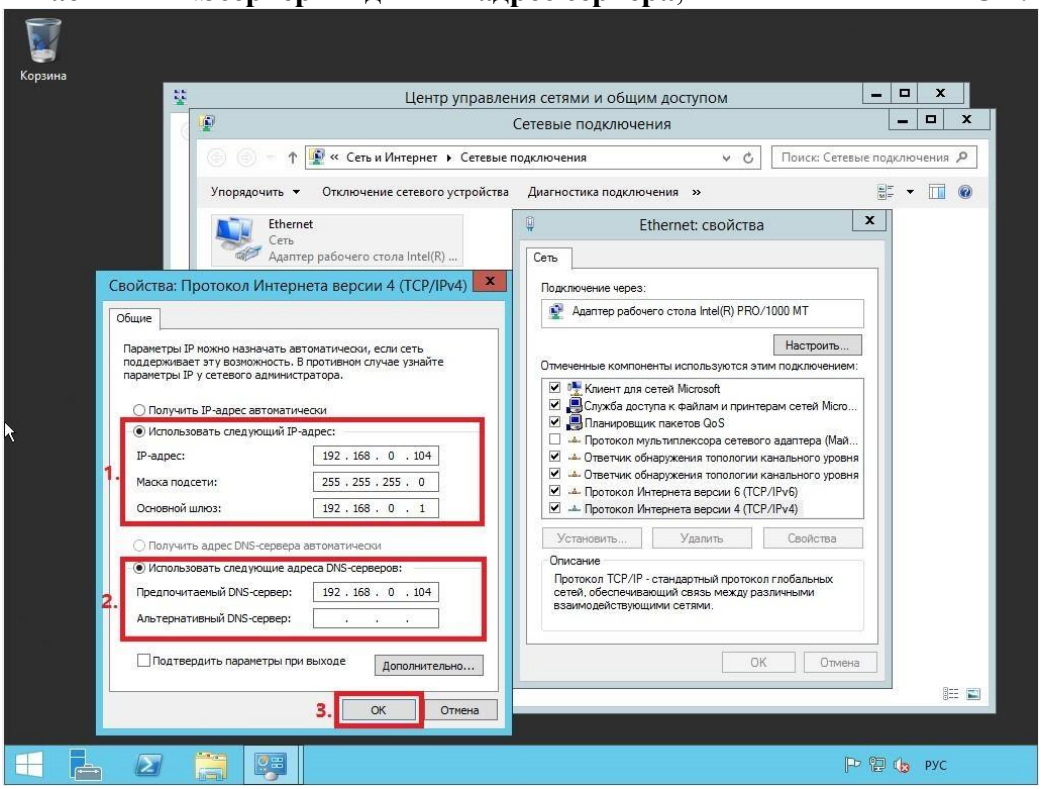


Рис. 19

Установка роли Active Directory Domain Services

1. Откройте окно диспетчера сервера и выберите пункт **Добавить роли и компоненты** (Рис.9).

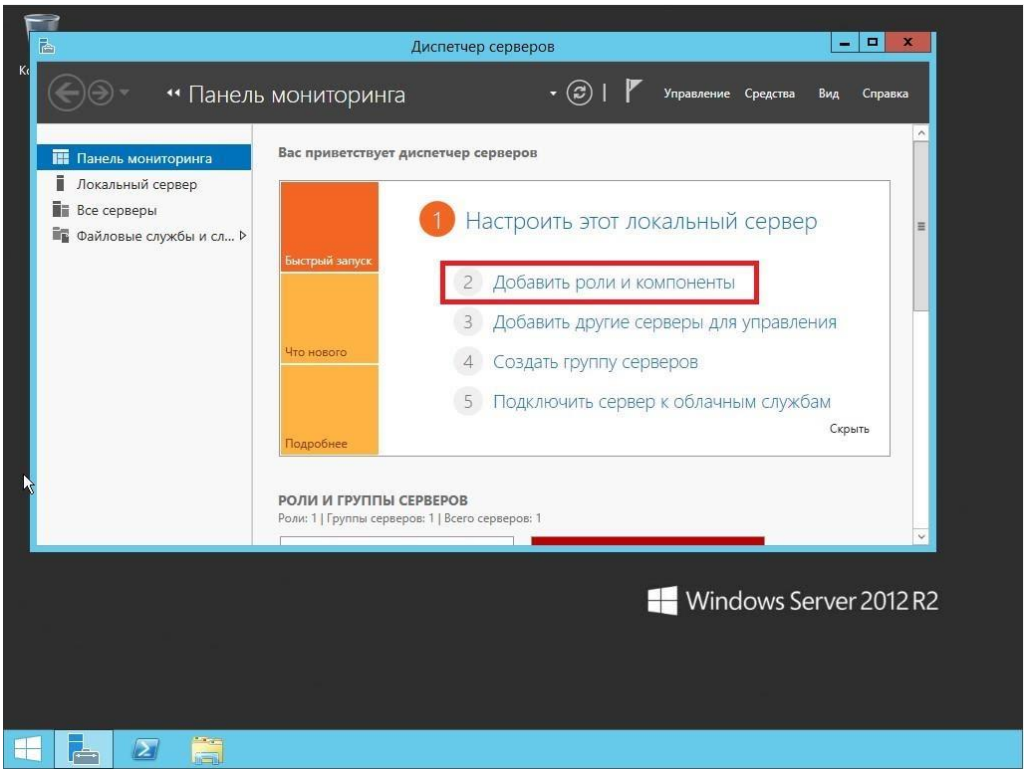


Рис. 20

2. В появившемся окне нажмите **Далее**

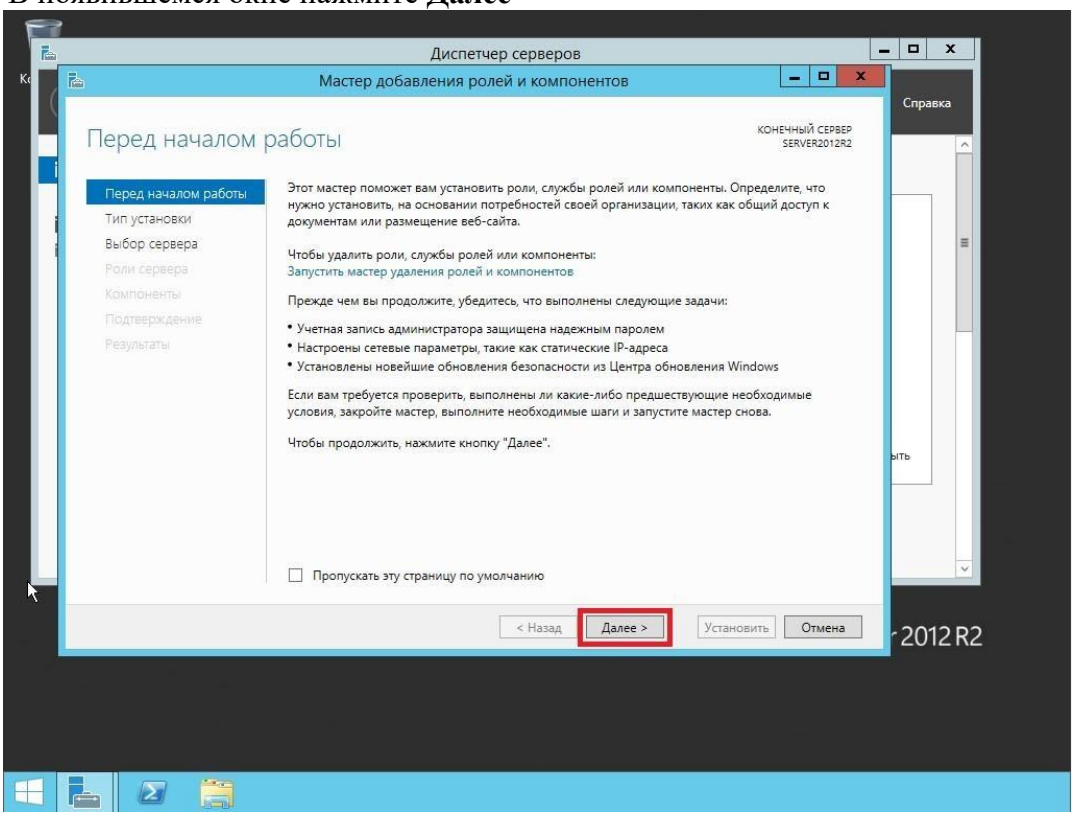


Рис. 21

3. Выберите пункт **Установка ролей и компонентов**, затем нажмите **Далее**

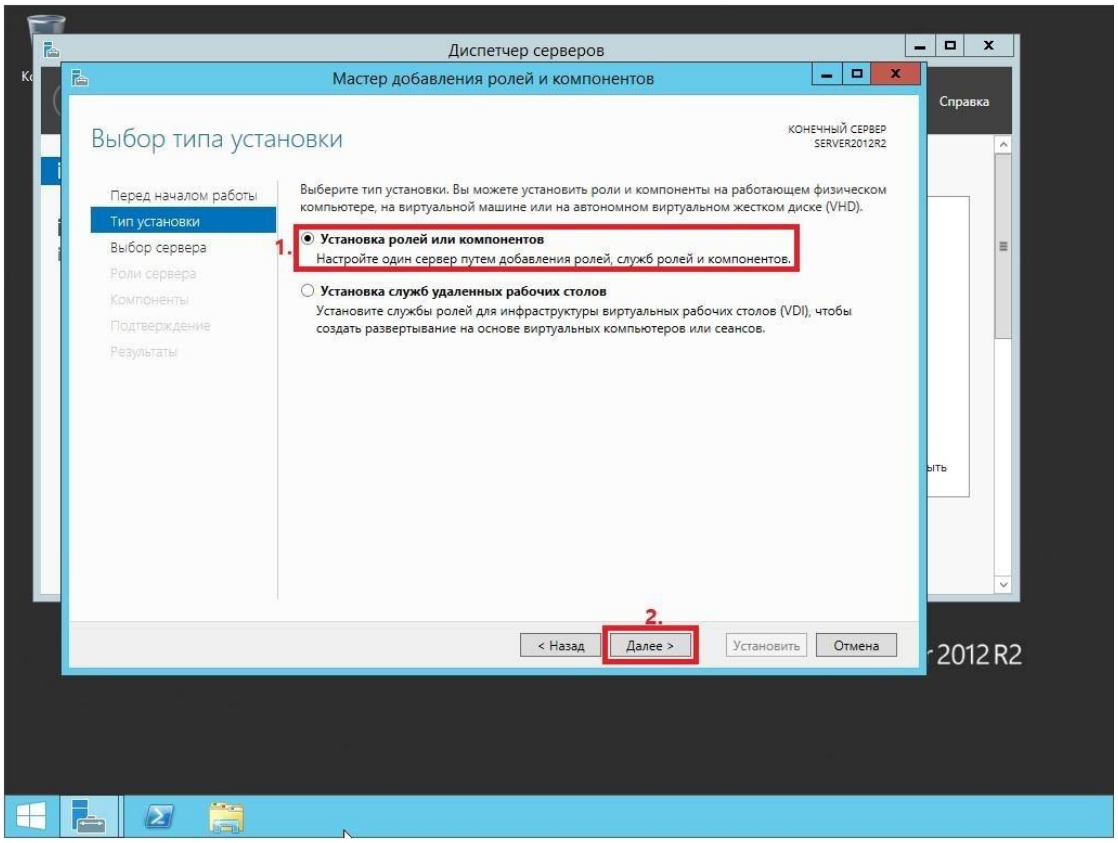


Рис. 22

4. Выберите сервер, на который будет производиться установка роли, затем нажмите **Далее**

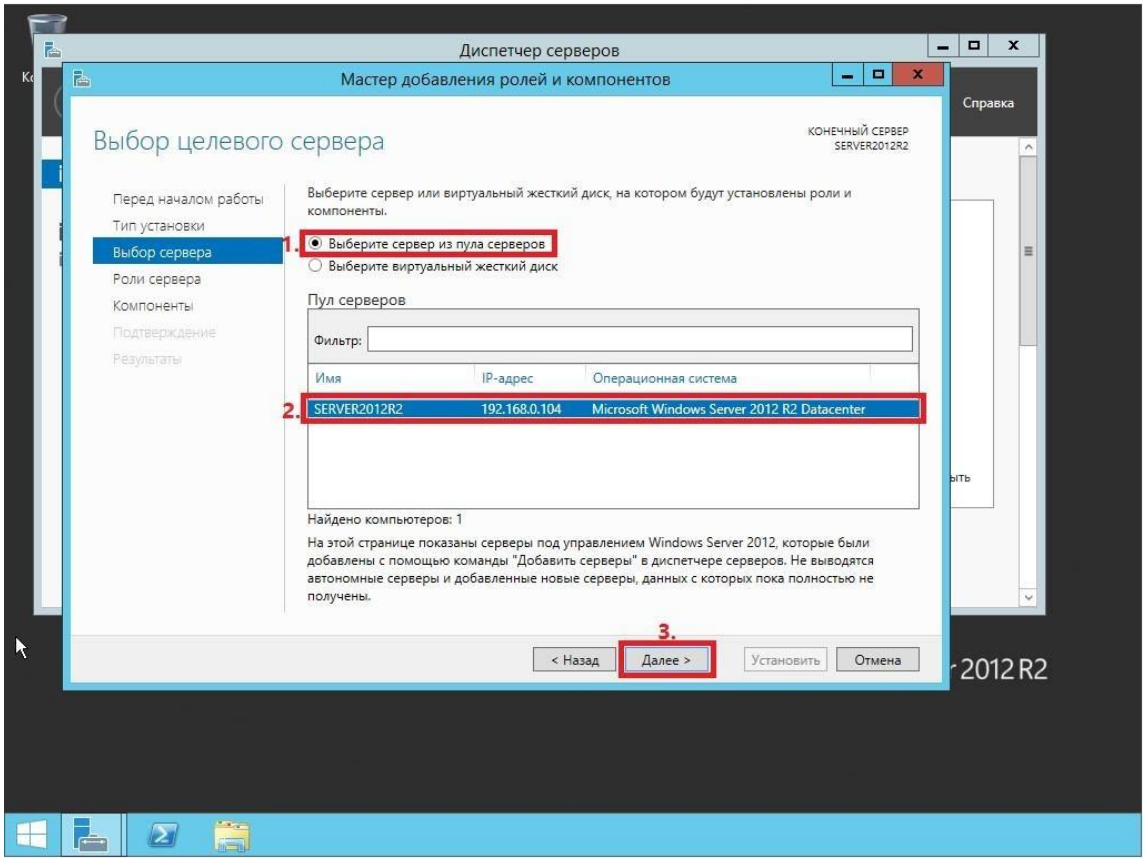


Рис. 23

-

5. Выберите роль **Доменные службы Active Directory**, на следующем этапе Мастер установки ролей предупредит, что для установки роли Доменные службы Active Directory нужно установить несколько компонентов. Нажмите **Добавить компоненты**

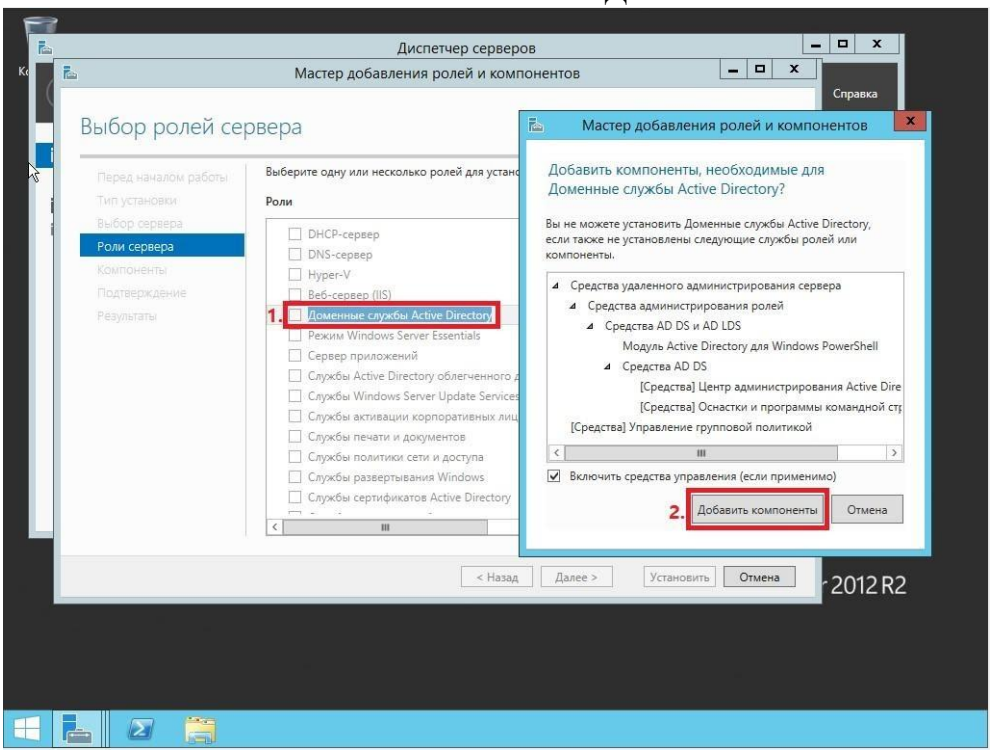


Рис. 24

6. Убедитесь, что после установки необходимых компонентов напротив **Доменные службы Active Directory** стоит галочка, затем нажмите **Далее**

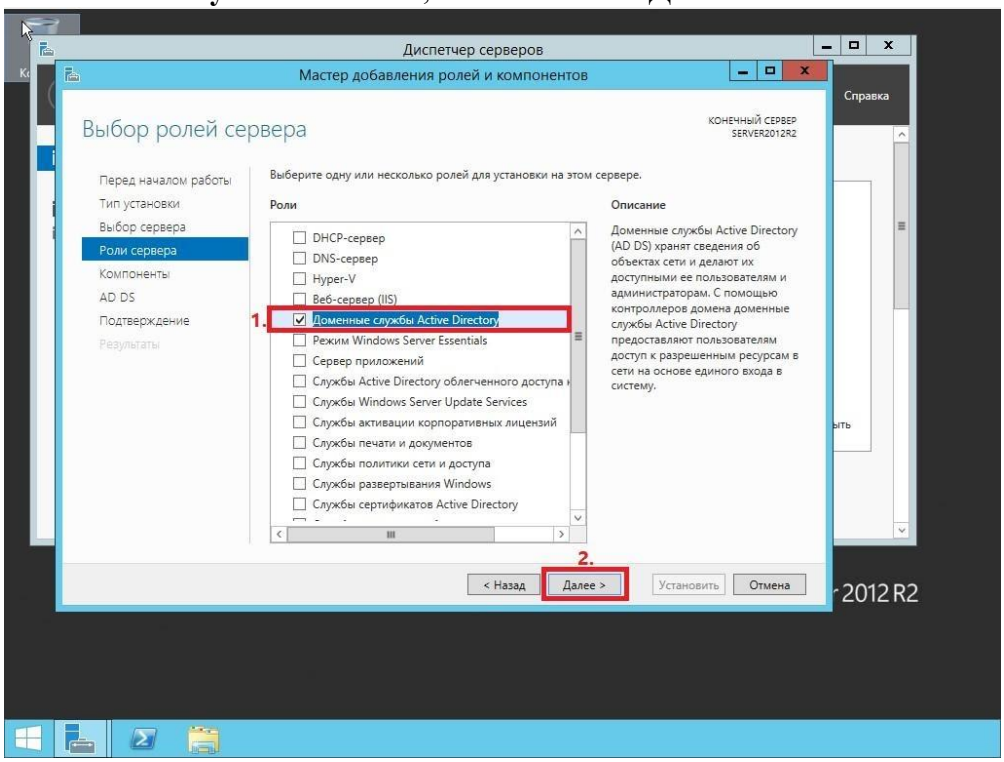


Рис. 25

7. На этапе добавления компонентов оставьте все значения по умолчанию и нажмите **Далее**

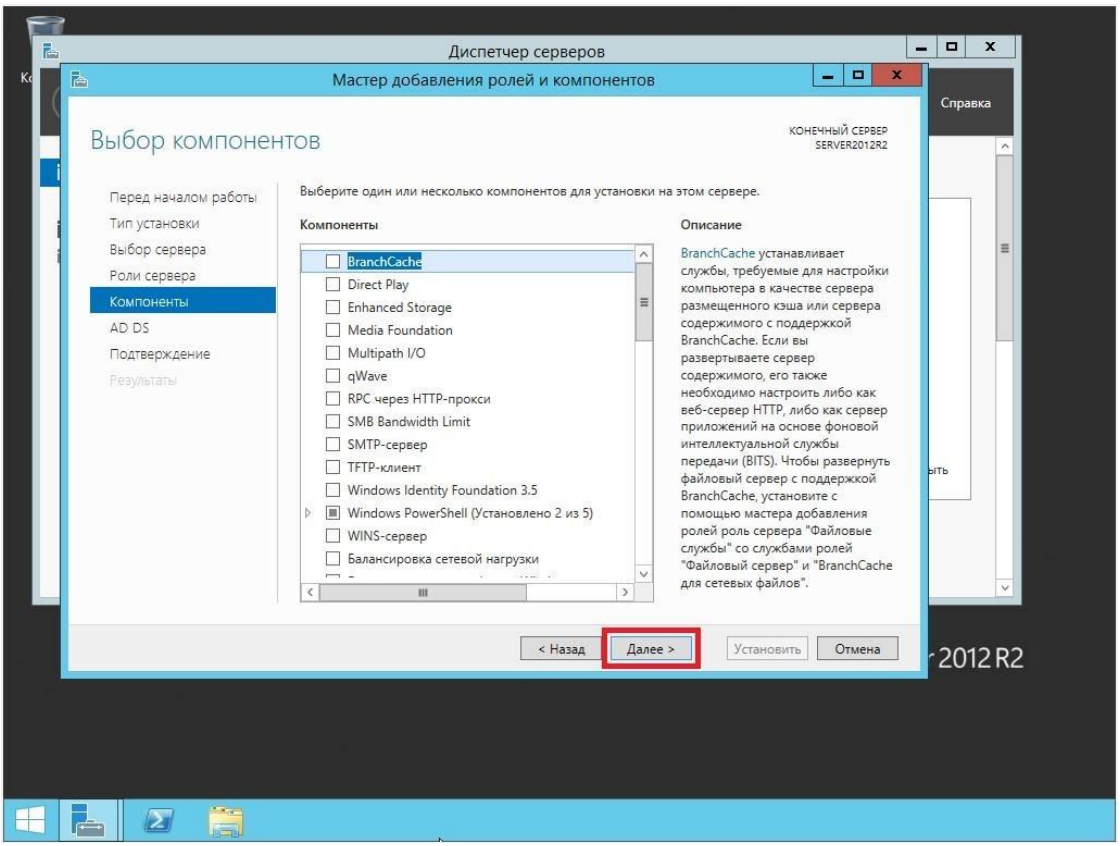


Рис. 26

8. Ознакомьтесь с дополнительной информацией касательно Доменных служб Active Directory, затем нажмите **Далее**

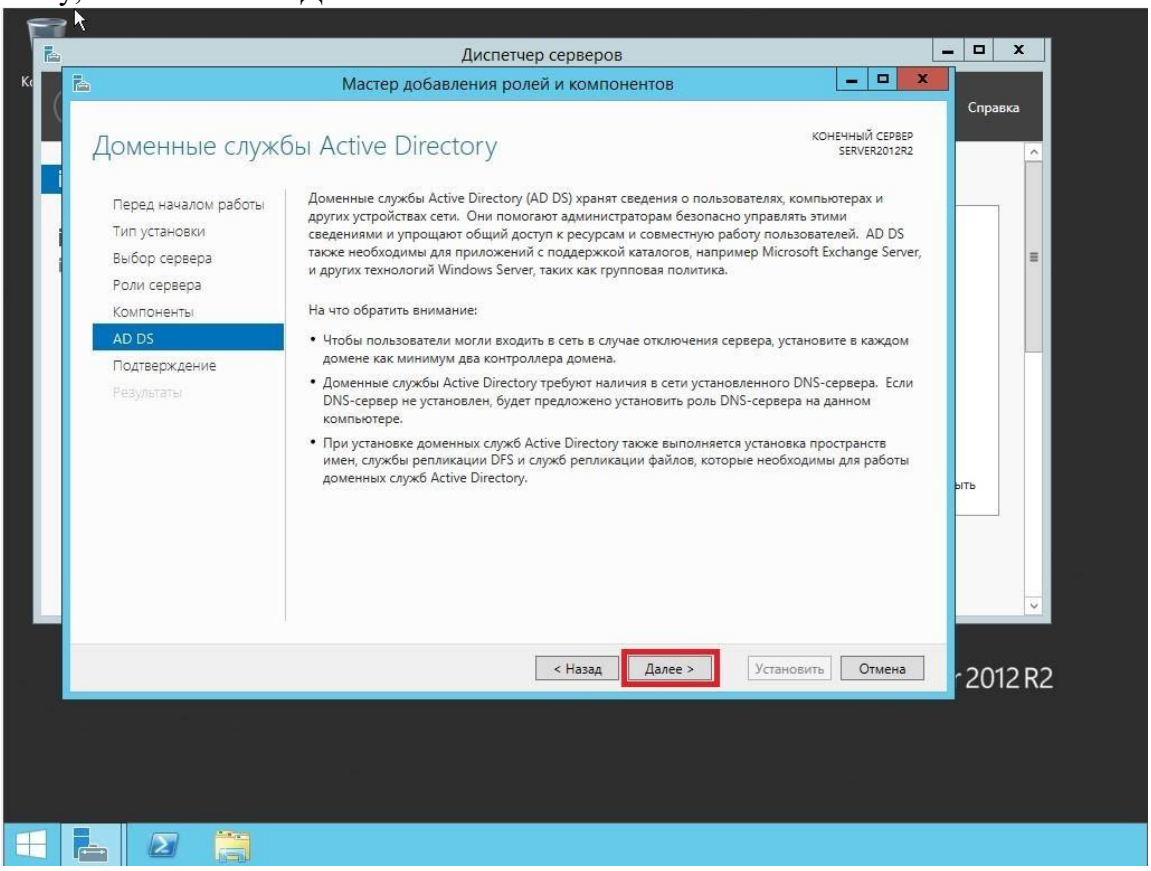


Рис. 27

9. Для начала установки роли нажмите **Установить**

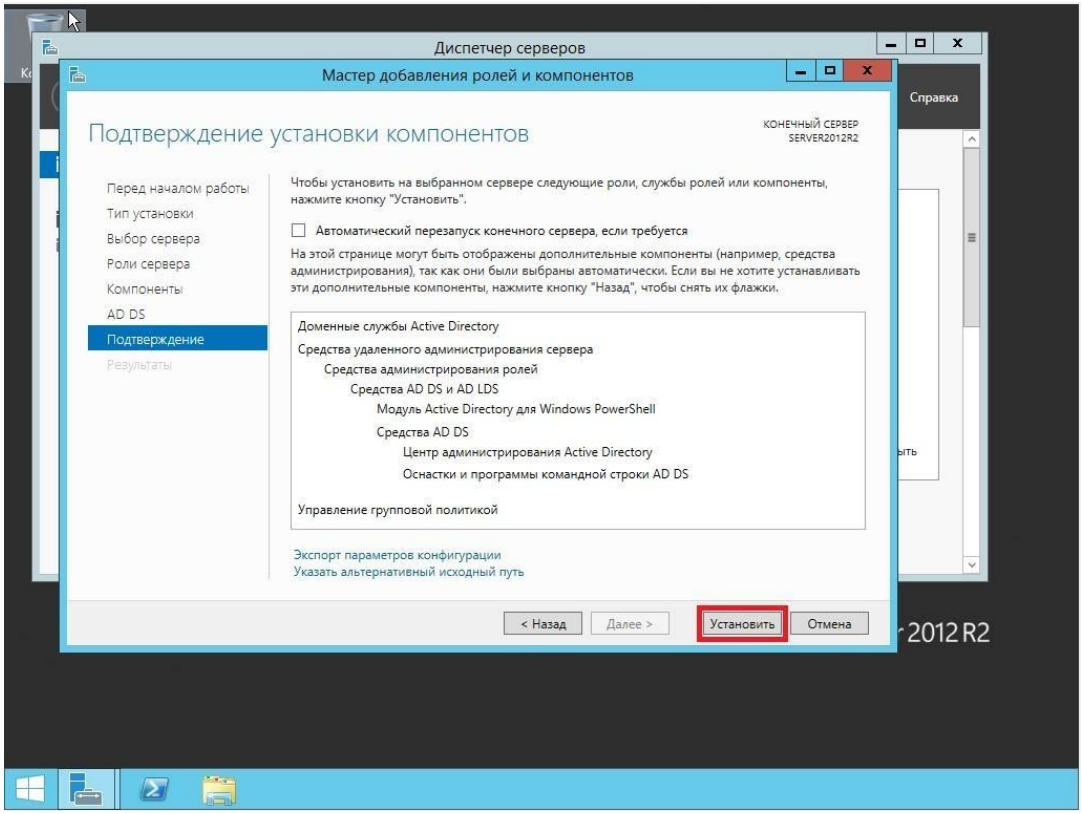


Рис. 28

10. После окончания установки нажмите **Повысить роль этого сервера до уровня контроллера домена**

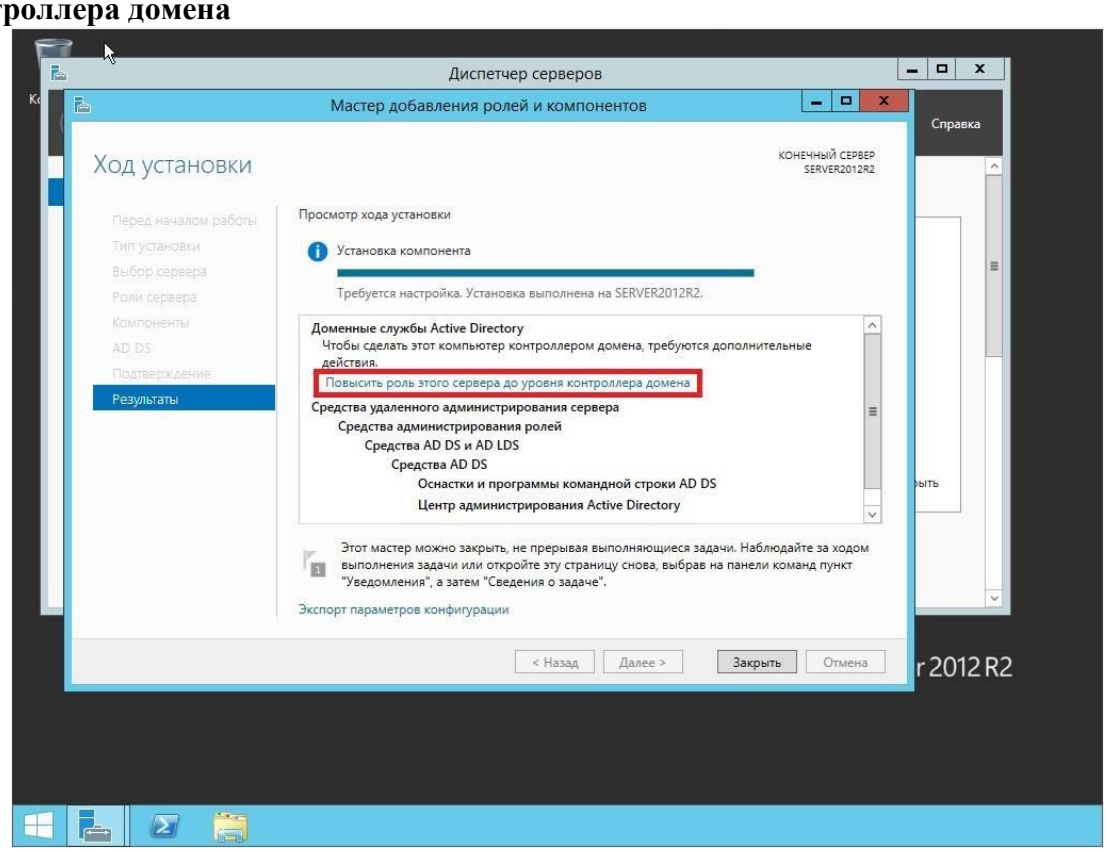


Рис. 29

11. Выберите пункт **Добавить новый лес**, затем в поле **Имя корневого домена** введите имя домена (*прим. в данном руководстве это example.local, Вы можете выбрать любое другое*), затем нажмите **Далее**

ВАЖНО! Домен вида .local или аналогичный можно использовать в качестве тестового, однако, он имеет ряд недостатков, а именно: 1) Вы никак не сможете подтвердить владение им для получения публичного SSL-сертификата; 2) Такое имя невозможно использовать из внешней сети; 3) Данный способ именования вступает в противоречие с глобальным DNS, так как не гарантирует его уникальность что приводит к потенциальным коллизиям.
Рекомендуется создавать согласованное пространство имен. Например имея домен wbsh.ru (который использует сайт), домен Active Directory делать суб-доменом, например: server. wbsh.ru. Либо использовать разные домены, например wbsh.ru — для сайта, а wbsh.net — для Active Directory.

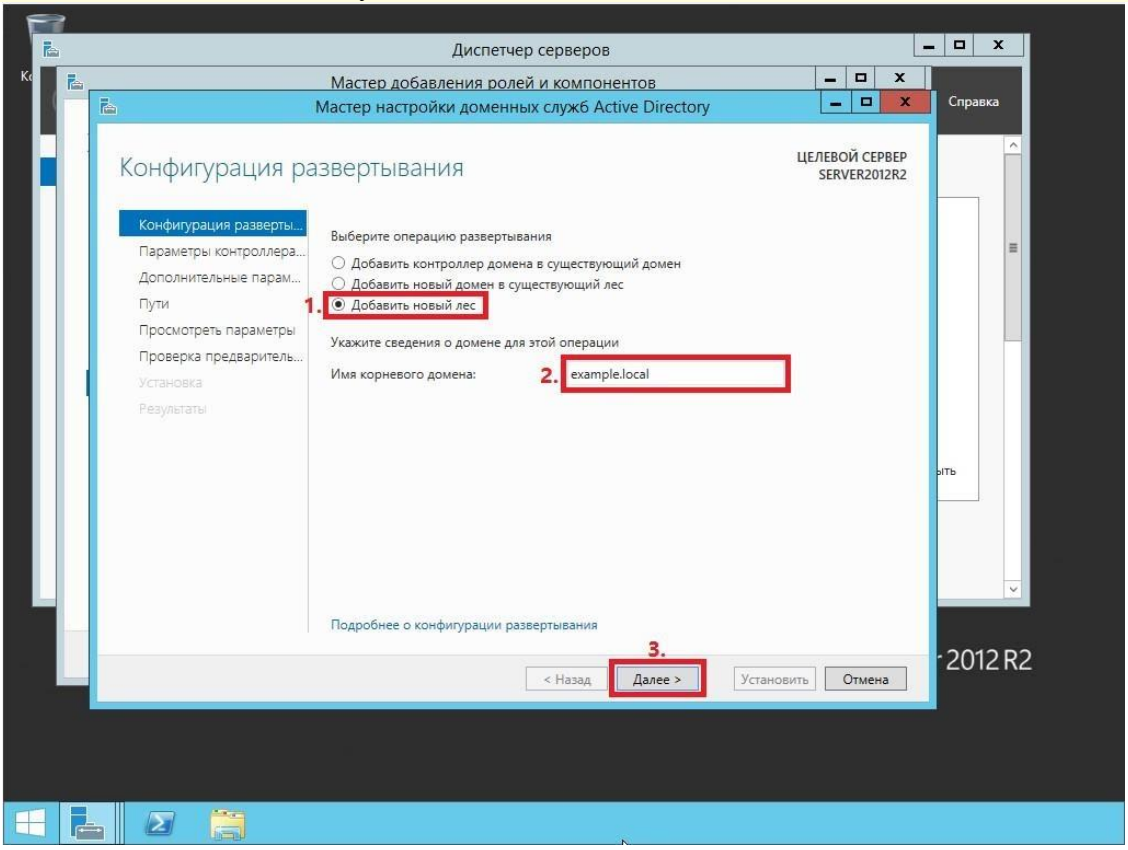


Рис. 30

12. На следующем шаге предлагается выбрать функциональный уровень нового леса и корневого домена. Если вы добавляете новый лес и планируете в дальнейшем использовать серверы на базе операционной системы Windows Server 2012 R2, то можете не менять функциональный уровень леса и корневого домена. Установите галочку напротив **DNS-сервер**, придумайте и введите пароль для режима восстановления служб каталогов в соответствующие поля, затем нажмите **Далее**

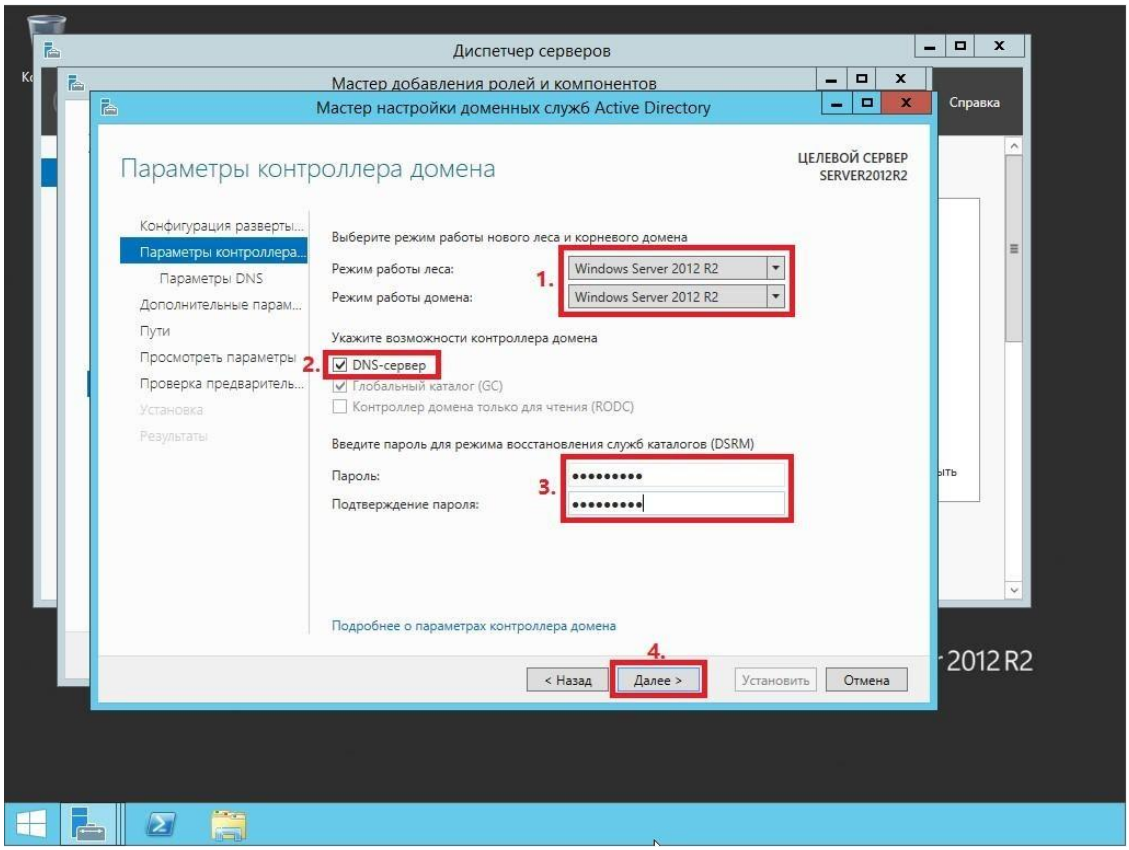


Рис. 31

13. Оставьте значение NetBIOS по умолчанию и нажмите **Далее**

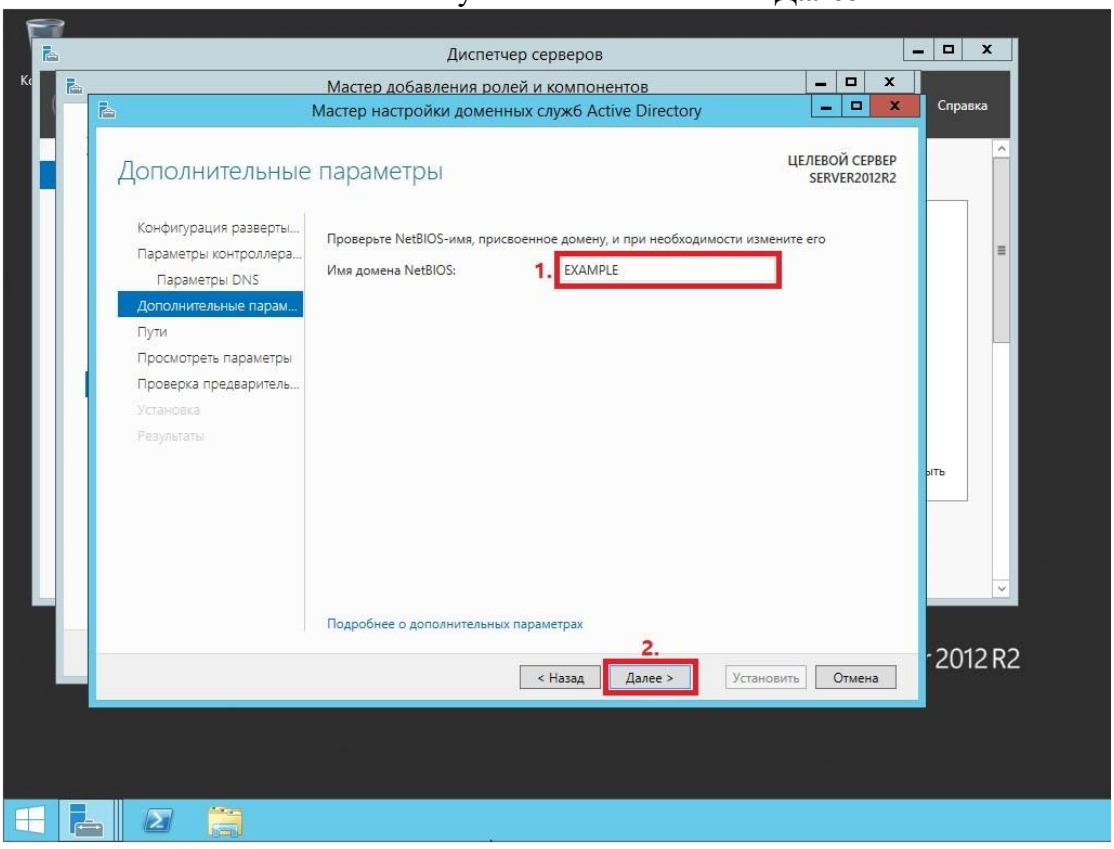


Рис. 32

14. Оставьте настройки по умолчанию и нажмите **Далее**

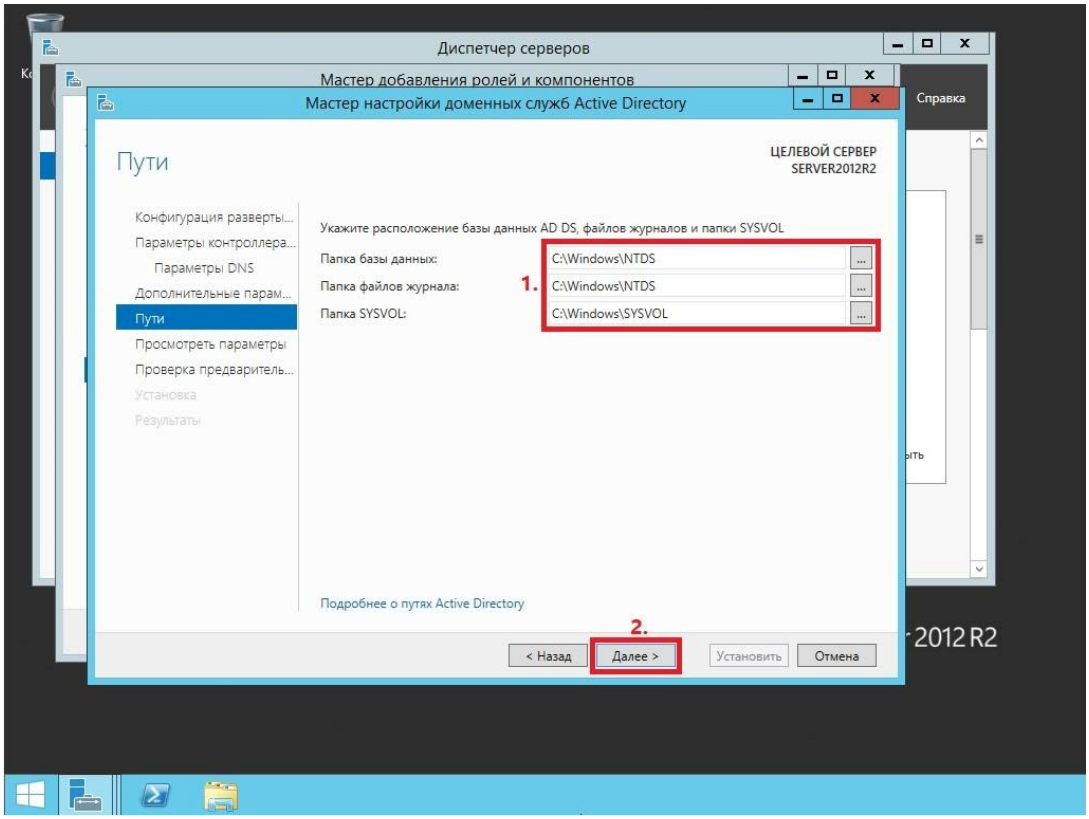


Рис. 33

15. В окне со сводной информацией по настройке сервера нажмите **Далее**

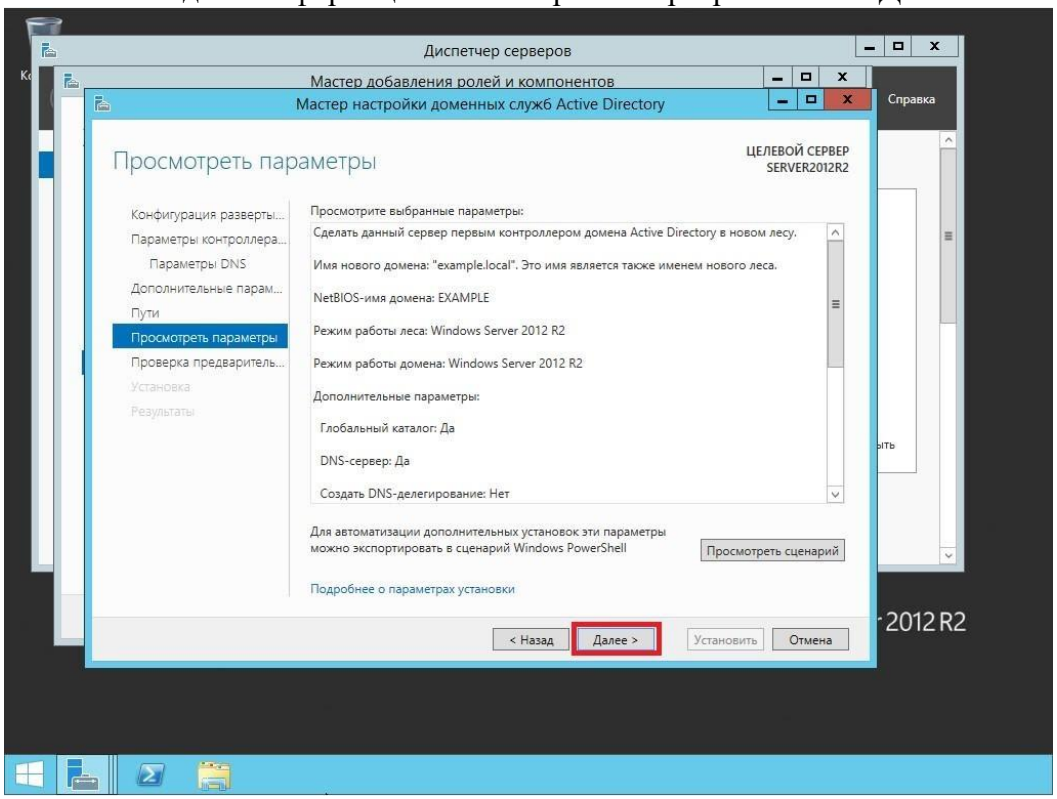


Рис. 34

16. Далее Мастер настройки доменных служб Active Directory проверит все ли предварительные требования соблюдены и выведет отчет. Нажмите **Установить** (Рис.24).

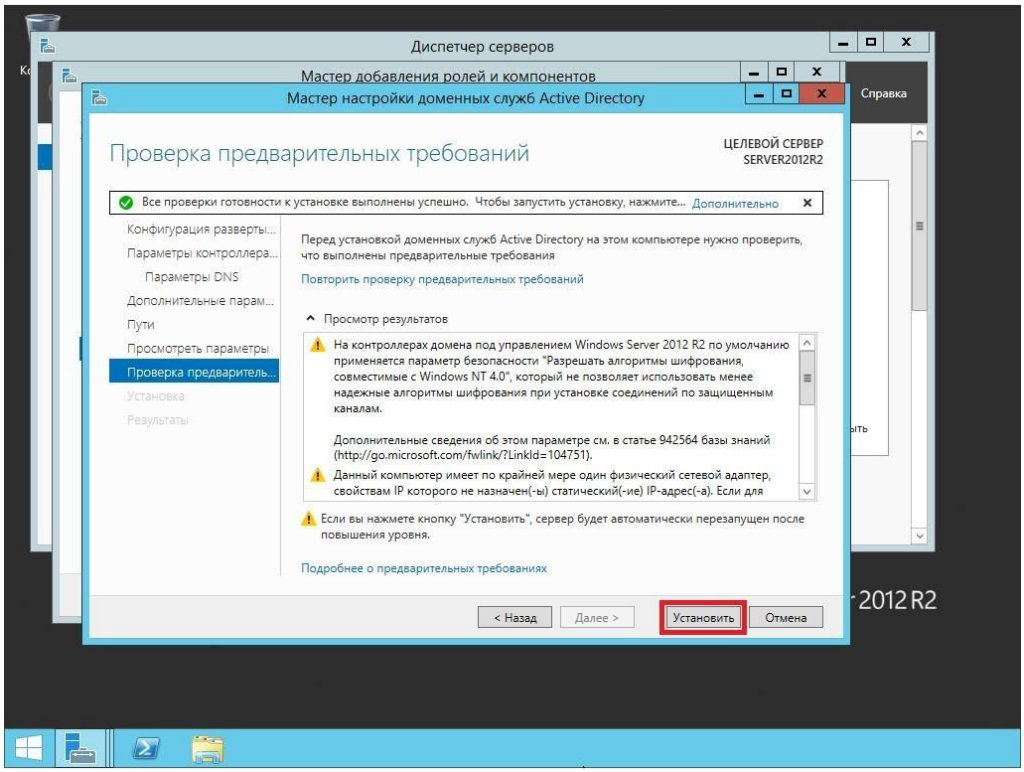


Рис. 35

17. После того как роль вашего сервера будет повышена до уровня контроллера домена, сервер автоматически перезагрузится. Перед тем как сервер начнет перезагружаться вы увидите предупреждение

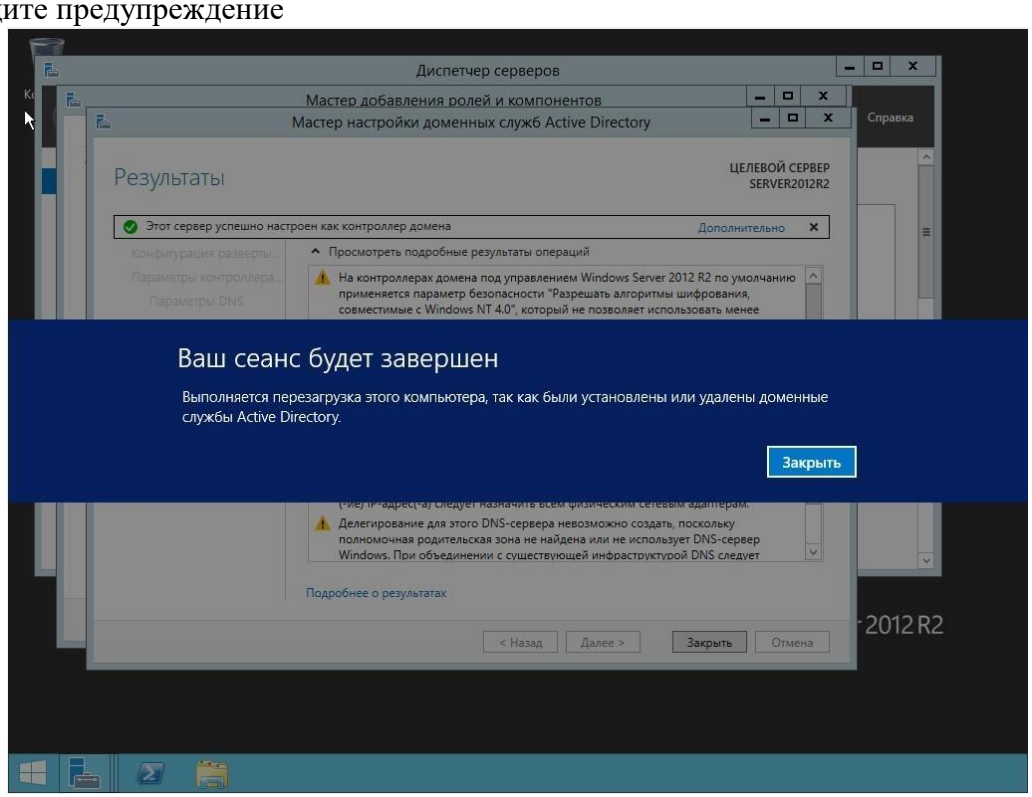


Рис. 36

18. После повышения роли сервера до уровня контроллера домена и перезагрузки — зайдите в систему под учетной записью с правами администратора домена

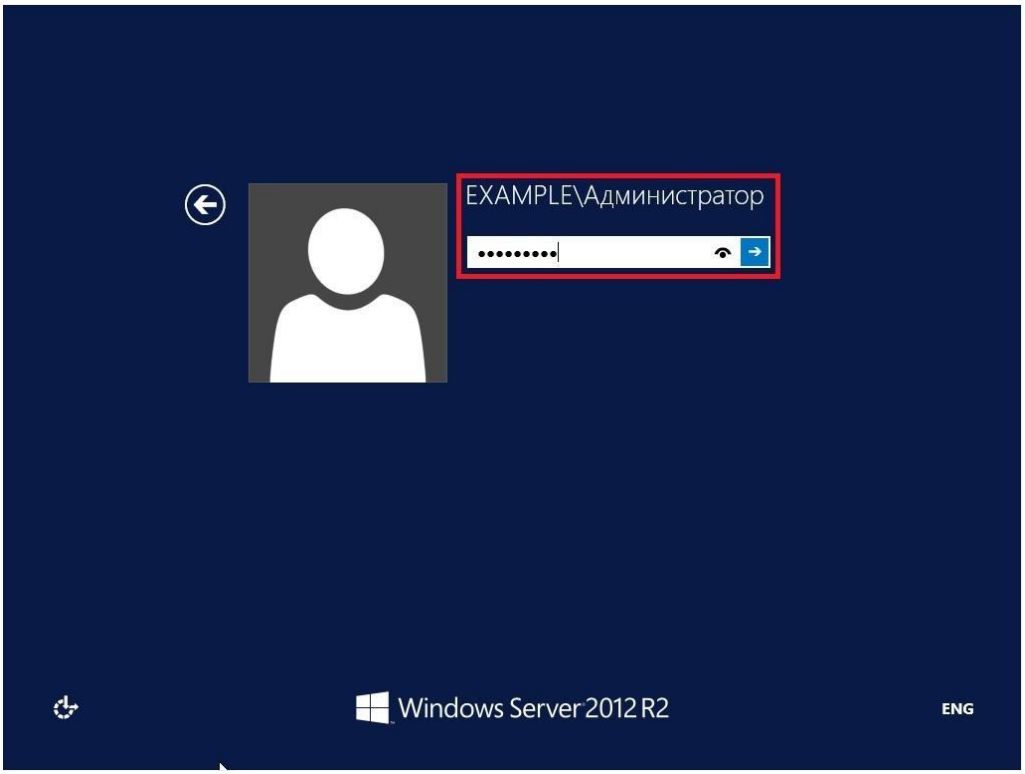


Рис. 37

Установка контроллера домена Active Directory в Windows Server 2012 R2 завершена!

Сделайте скриншоты (фотографии) процесса установки контроллера домена Active Directory и вставьте в отчёт.

Настройка политики паролей учетных записей в Active Directory

1. Что бы изменить политику паролей для пользователей, находящихся в домене, заходим в **«Диспетчер серверов»** далее в верхнем меню жмем **"Средства"** и переходим в раздел **"Управления групповой политикой"**

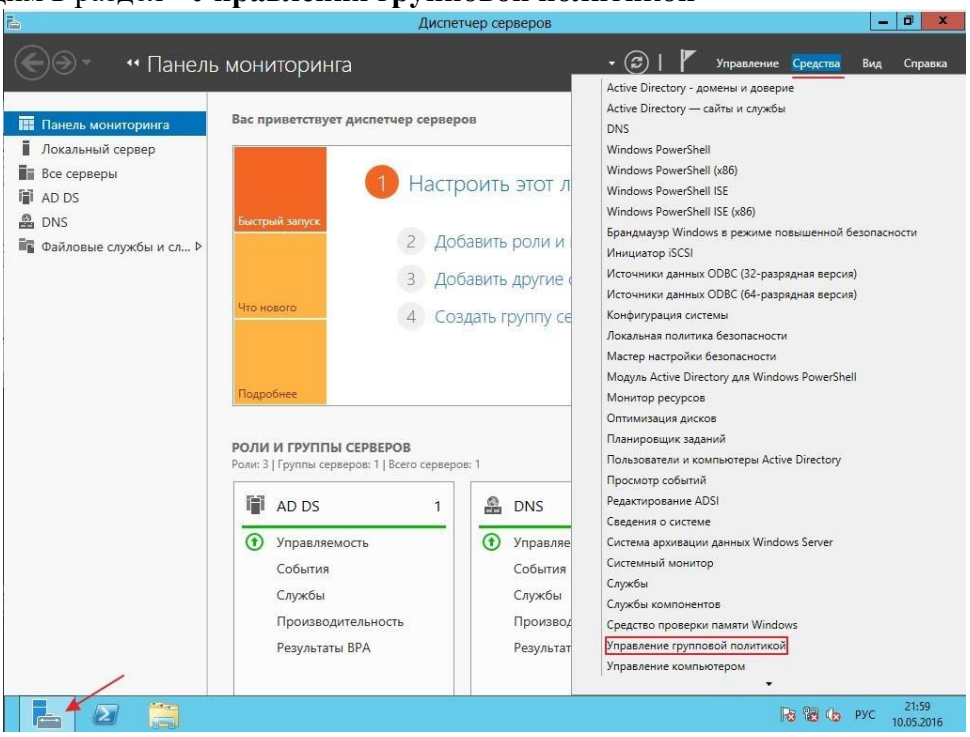


Рис. 38

2. Для того что бы изменить политику паролей необходимо изменить политику по умолчанию домена (Default Domain Police) для этого нажмите ПКМ по данной политике и нажмите на пункт **"Изменить"**

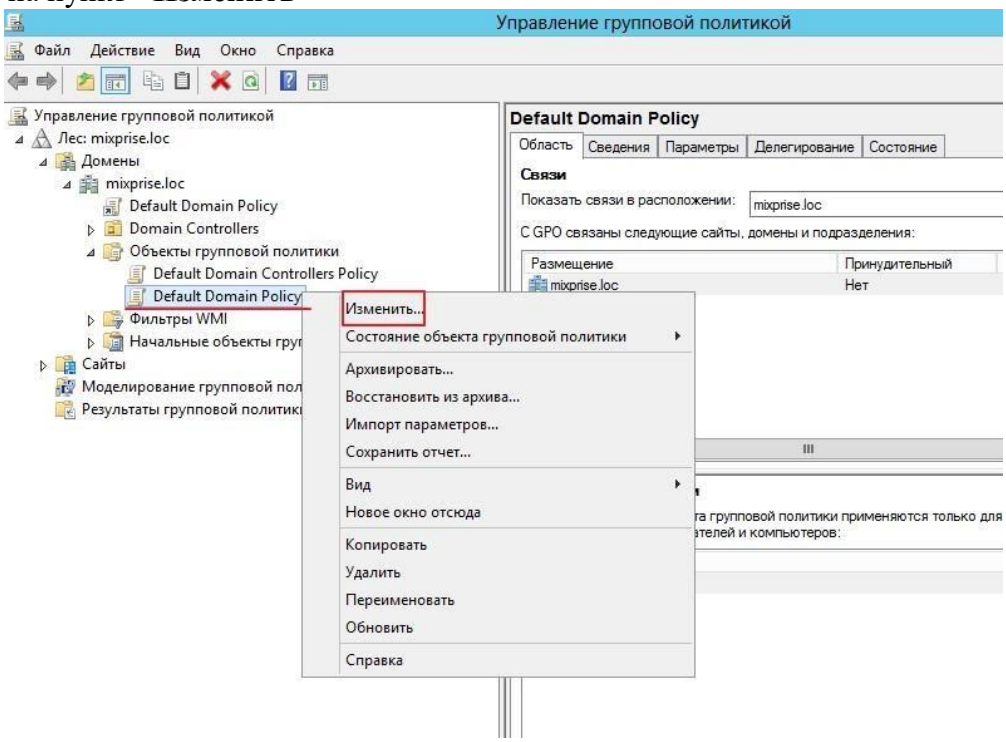


Рис. 39

3. В открывшимся окне открылся редактор, теперь необходимо найти где же изменять саму политику паролей, редактирование происходит в разделе **"Конфигурации компьютера"** далее разворачиваем папку **"Конфигурация Windows"** дальше открываем **"Параметры безопасности и политики учетных записей"**

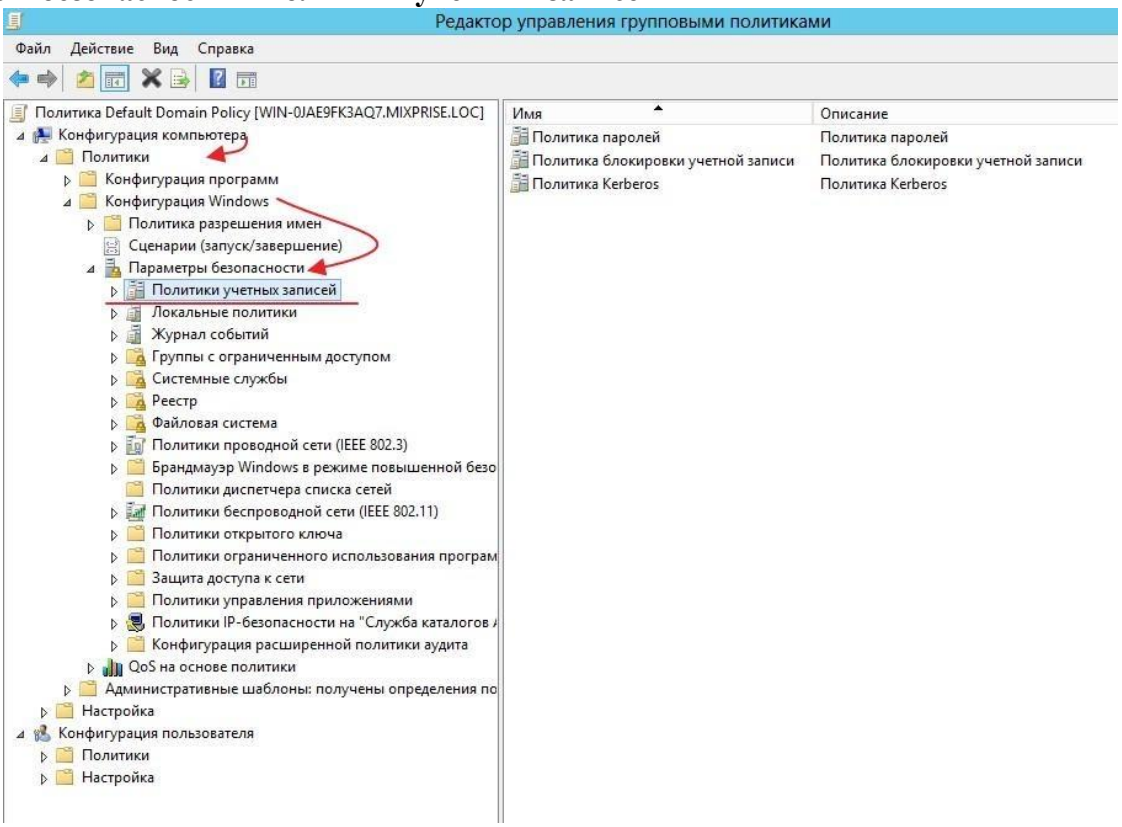


Рис. 40

Начнем настройку с первого раздела под названием **"Политика паролей"** открываем его

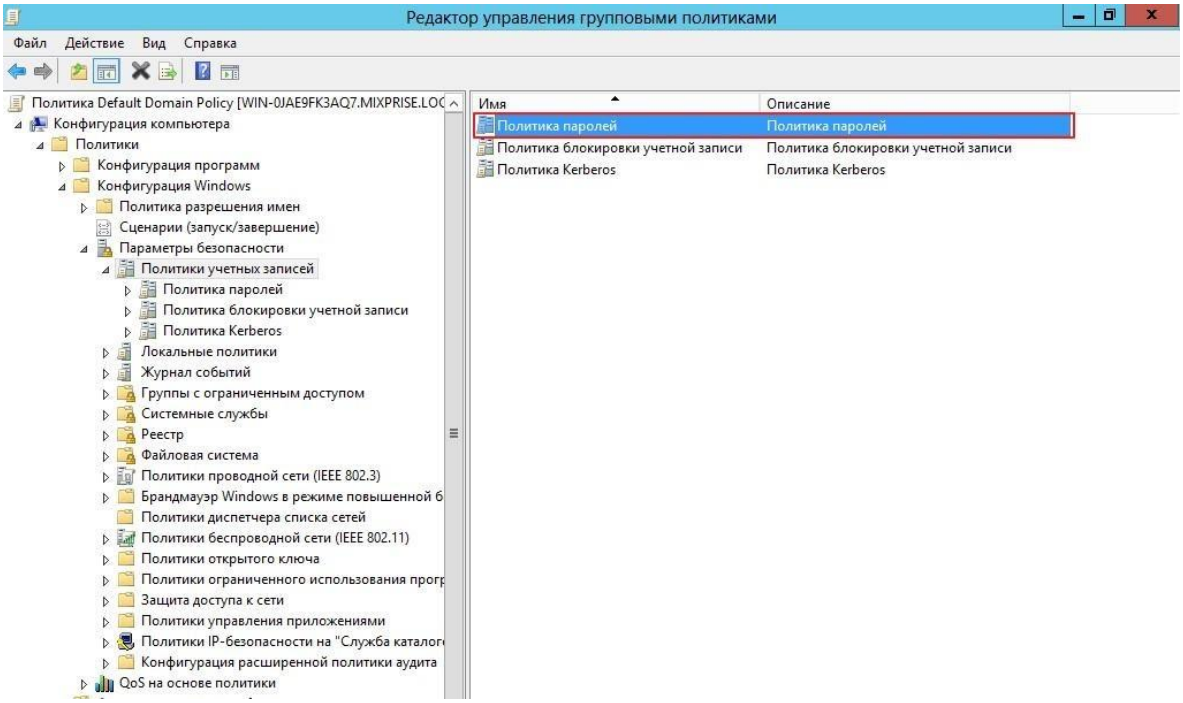


Рис. 41

В открывшимся окошке нам доступно для изменения 6 пунктов, каждый из них мы можем изменить, достаточно просто кликнуть на него.

4. Открываем вкладку **"Вести журнал паролей"** с помощью нее определятся числовое значение новых паролей, которые применяются к пользователю прежде чем он сможет снова использовать предыдущий пароль, здесь я оставляем все по умолчанию.

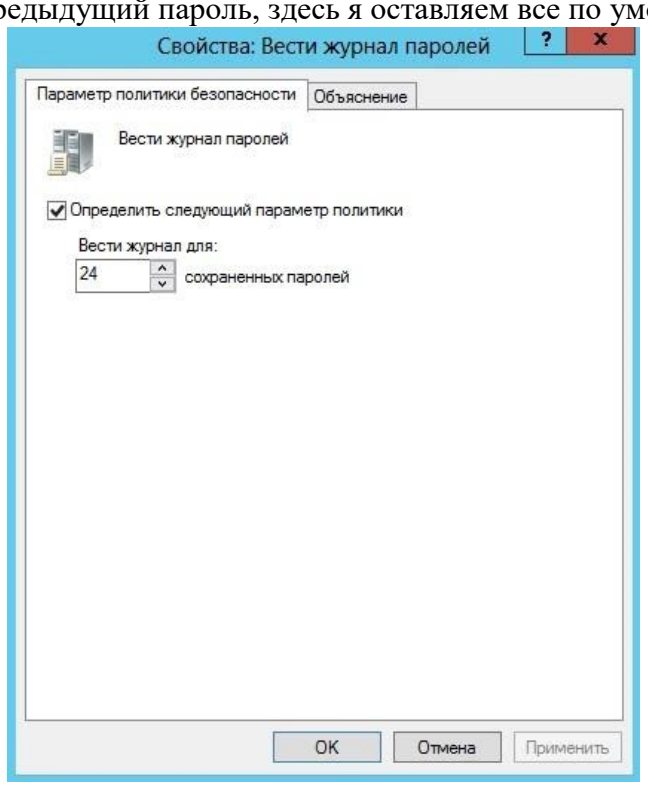


Рис. 42

5. Следующая вкладка **"Максимальный срок действия пароля"** по умолчанию это 42 дня, с помощью этой политики определяется временной интервал, в котором используется пароль прежде чем система вновь потребует от пользователя этот пароль поменять, убираем чекбокс и жмем **"Применить"**

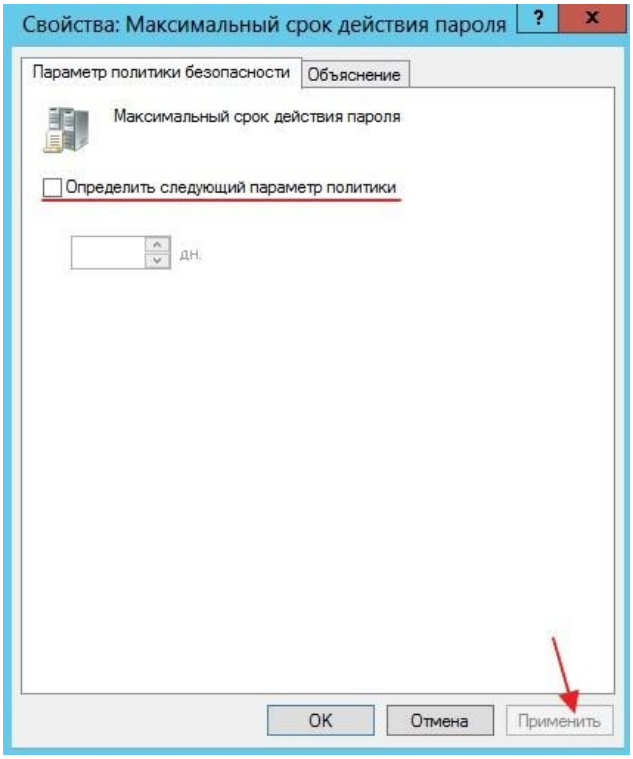


Рис. 43

Переходим на раздел **"Минимальная длина пароля"**. По умолчанию это 7-8 символов, выставляем как минимум 4 символа

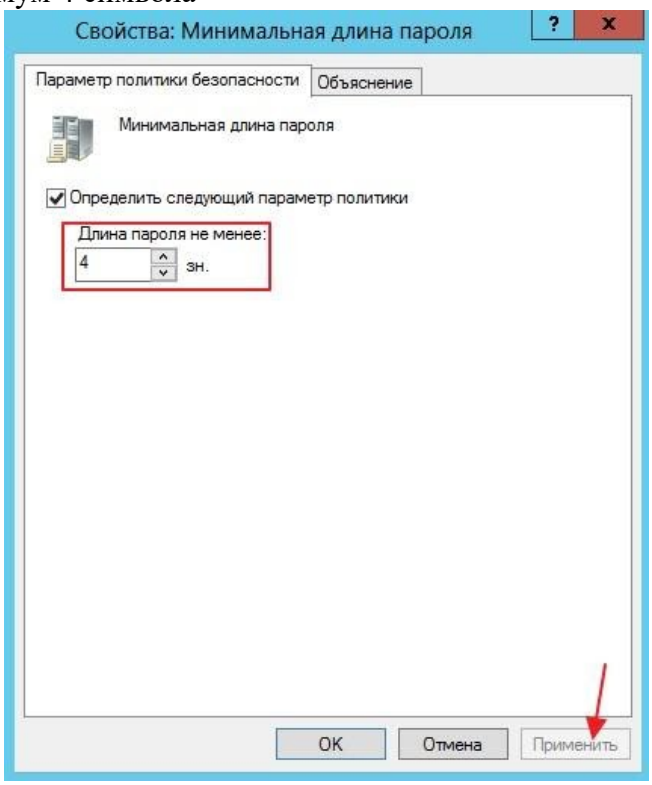


Рис. 44

6. Далее **"Минимальный срок действия пароля"** с помощью него определяется время, за которое пользователь не может изменить пароль по умолчанию значение ноль дней. Убираем галочку и применяем настройки

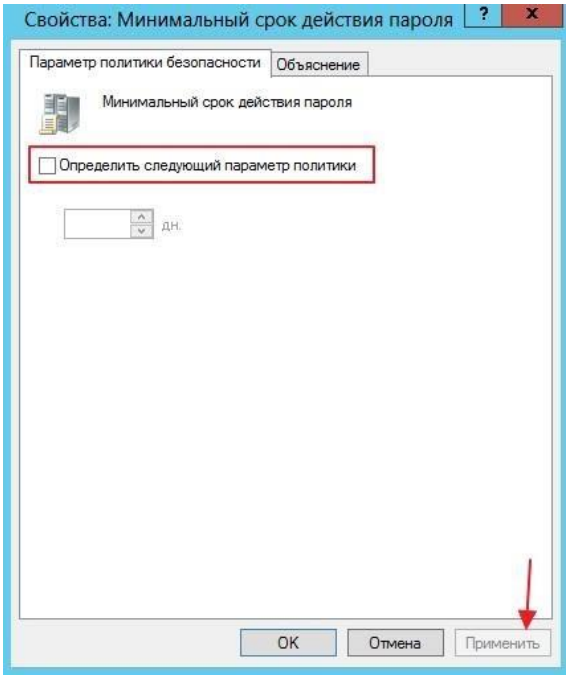


Рис. 45

7. Переходим к **"Пароль должен отвечать требованиям сложности"** это те требования, когда в пароле обязательно должны присутствовать английские буквы, верхнего и нижнего регистра, цифры, не алфавитные символы и т.д. Ставим **"Отключен"** и кликаем **"Применить".**

ВАЖНО! При настройке «живого» сервера эти параметры должны быть включены!

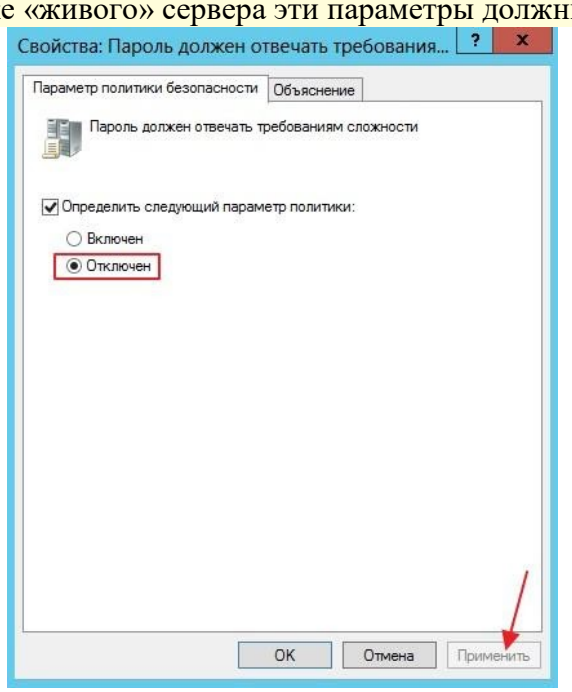


Рис. 46

8. Последняя политика **"Хранить пароли"**, скажем лишь то, что если в ее включить пароли ваших пользователей в системе будут храниться в открытом виде и если злоумышленник доберется до вашей сети, то он легко сможет получить доступ к файлам!

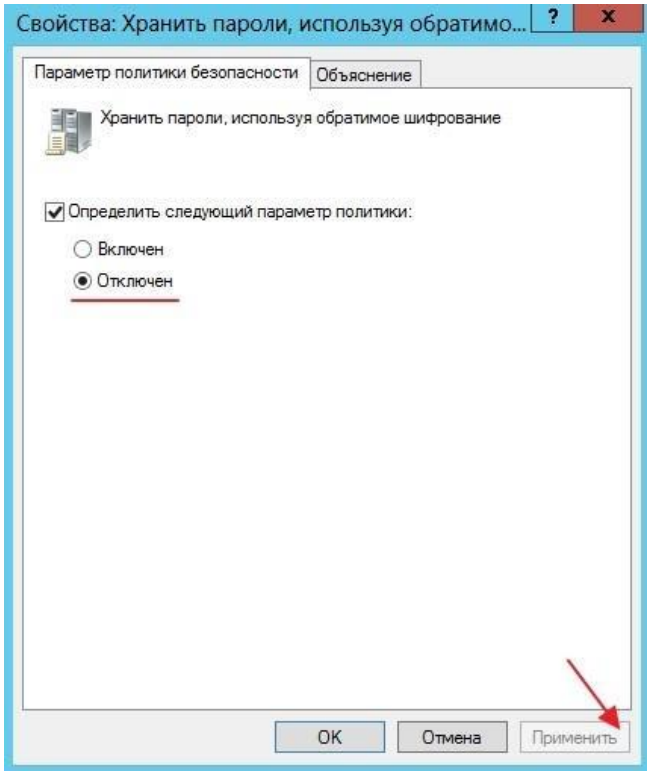


Рис. 47

9. Рассмотрим **"Политику блокировки учетной записи"**. В данной политике доступны 3 блока это: **"Время до сброса счетчика блокировки"** выставляете время блокировки аккаунта, в качестве примера поставим значение равное 30 мин

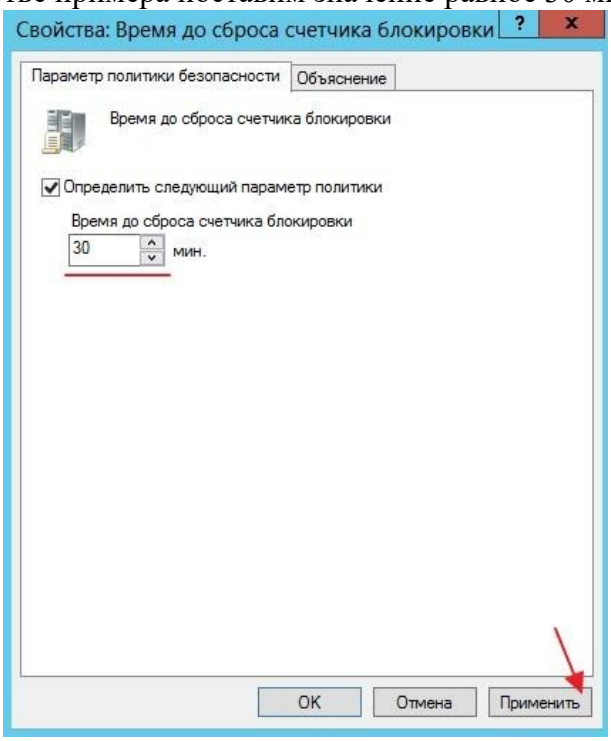


Рис. 48

"Пороговое значение блокировки» прежде, чем аккаунт будет заблокирован, грубо говоря выставляете попытки ввода неверного пароля, после чего наступит блокировка аккаунта пользователя, выставим в качестве примера 3 раза

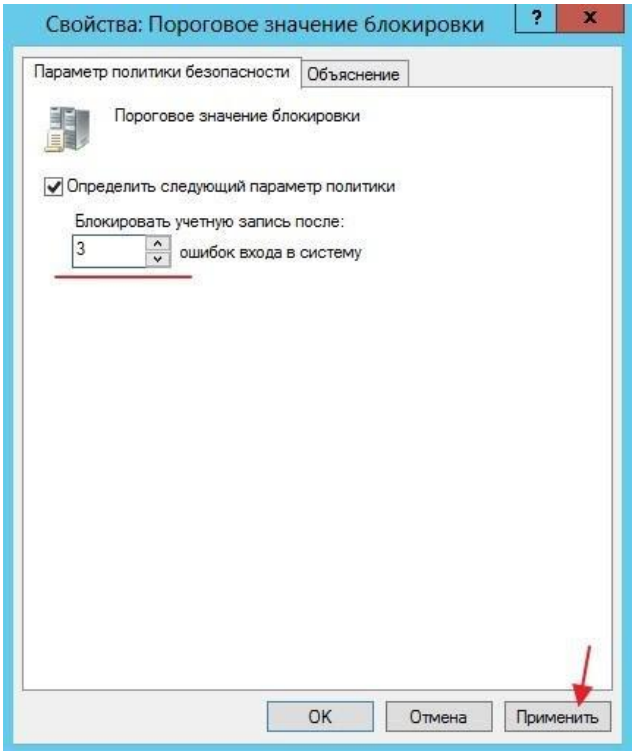


Рис. 49

"Продолжительность блокировки учетной записи" означает что если вы ввели скажем 4 раза неверный пароль, при такой настройке можете подождать 30 мин, и у вас снова будет доступно 3 попытки ввода пароля

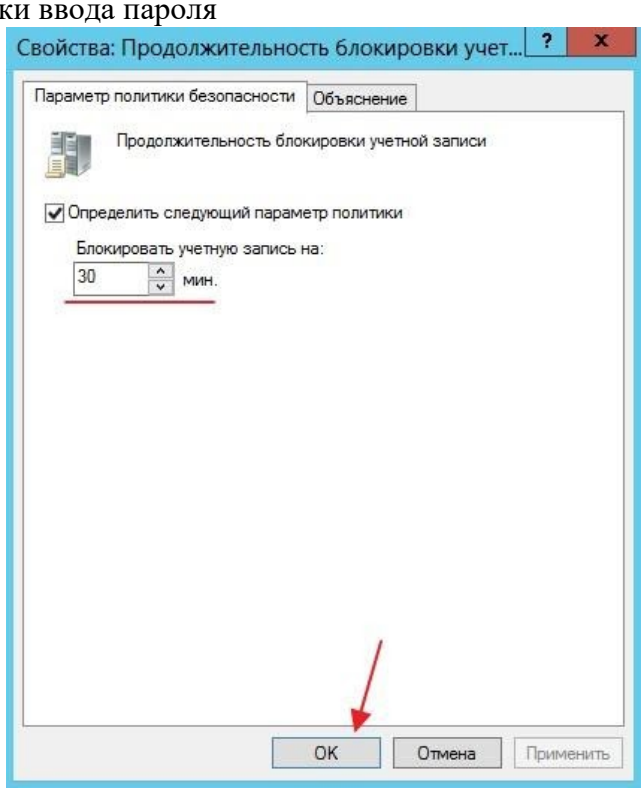


Рис. 50

10. Теперь что бы изменить пароль у пользователя заходим в **"Пользователи и компьютеры Active Directory"**

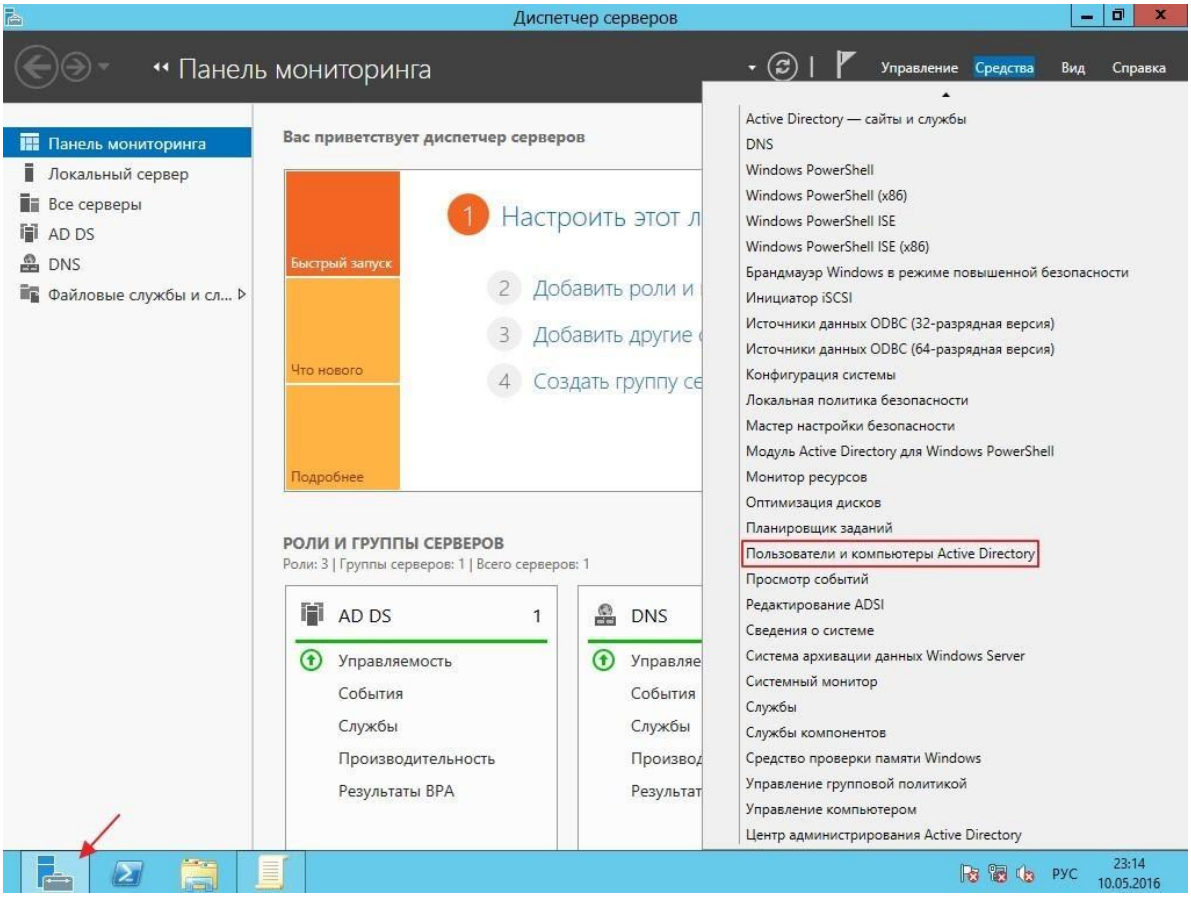


Рис. 51

Ищем учётную запись, для которой мы хотим изменить пароль, в нашем примере создайте учетную запись **«Admin».**

11. Теперь попробуем войти на сервер с новой учетной записью и новым паролем, для этого выполните **"Выход из системы"**

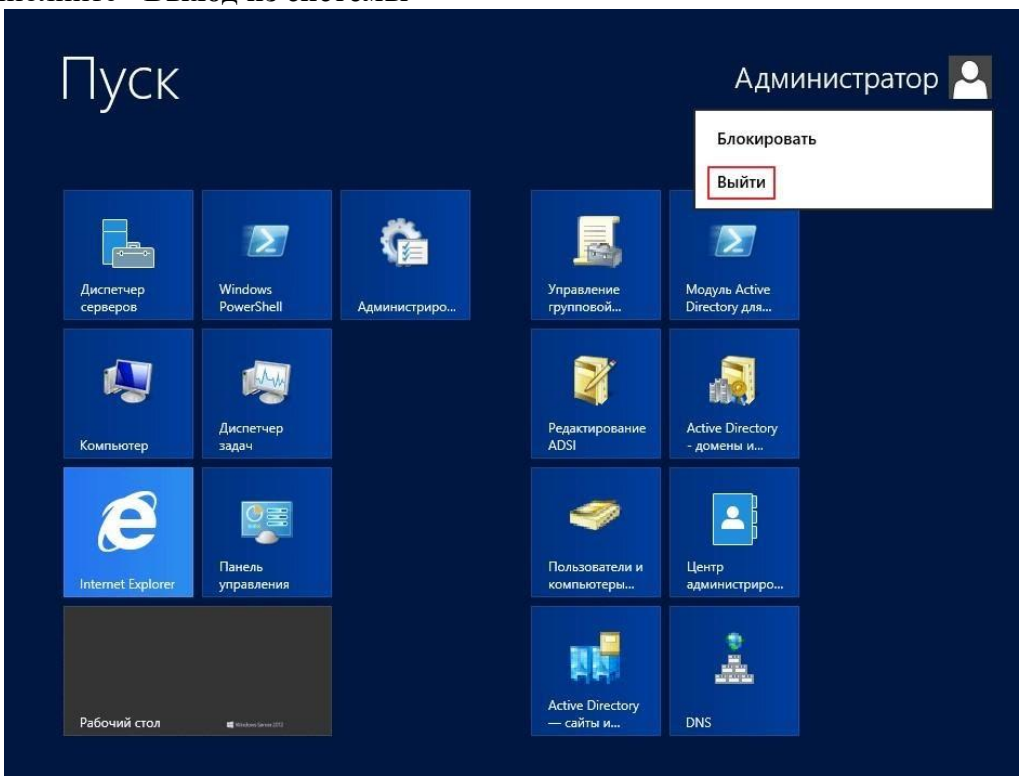


Рис. 52

Осуществляем вход с новой учетной записью

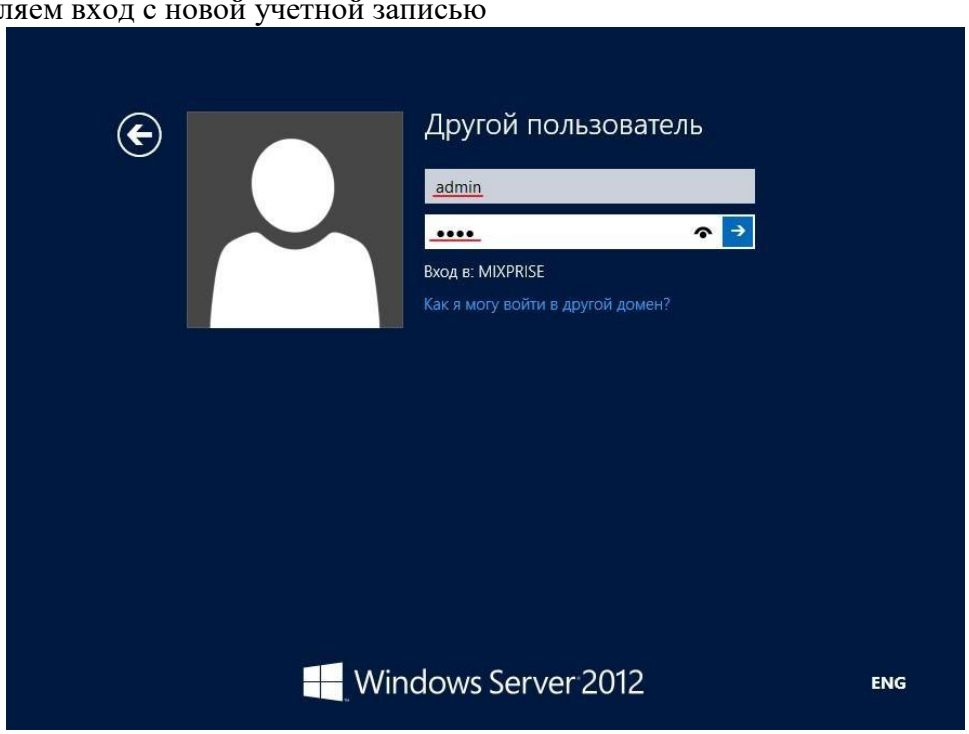


Рис. 53

Сделайте скриншоты (фотографии) процесса настройки контроллера домена Active Directory и вставьте в отчёт.

Создание пользователя в Powershell с параметрами New-ADUser

1. Итак, представим, что нам нужно срочно создать 50 однотипных учетных записей.

Пишем вот такой скрипт:

$org="OU=Students,DC=contoso,DC=com"
$username="student" $count=1..50 foreach ($i in $count)
{ New-AdUser -Name $username$i -Path $org -passThru }

Где:

- Name - логин
- GivenName - имя
- SurName - фамилия
- AccountPassword - пароль, который мы объявили в переменной
- Enbaled - делает пользователя активным

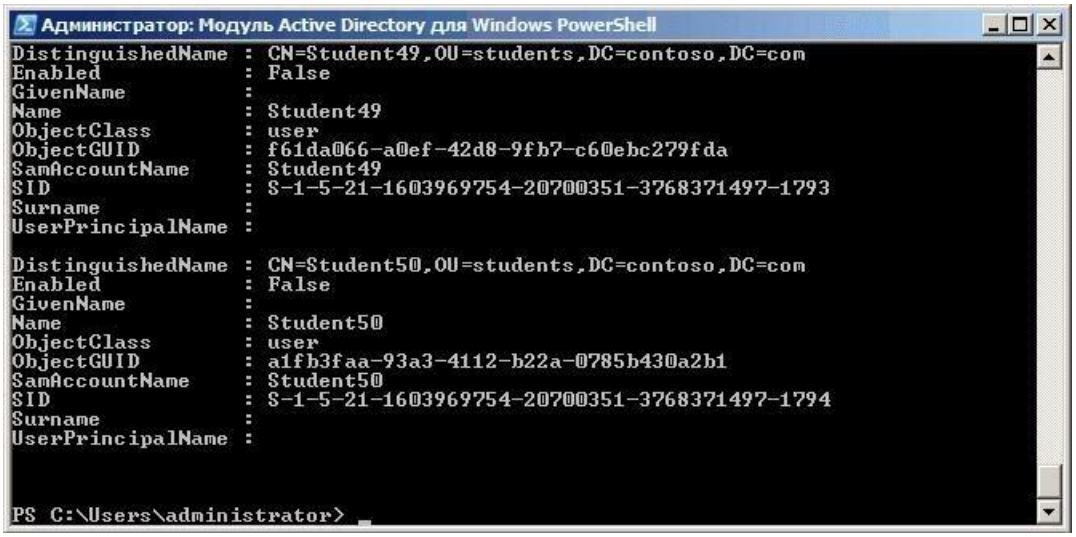


Рис. 54

Запускаем скрипт, и в подразделении Students создается 50 пользователей с именами student1-student50. По умолчанию учетки создаются отключенными, и пользователи все равно будут вынуждены к вам обращаться для их активации. Избежим этого:

```
$org="OU=Students,DC=contoso,DC=com"
$username="student" $count=1..50 foreach ($i in $count)
{ New-AdUser -Name $username$i -Path $org -Enabled $True -ChangePasswordAtLogon
$true `
-AccountPassword (ConvertTo-SecureString «p@$$w0rd» -AsPlainText -force) -passThru }
```

Здесь создаем учетные записи уже активными и задаем *p@$$w0rd* как пароль по умолчанию, а также указываем сменить его при первом входе в систему. Чтобы не передавать пароль в открытом виде, используем командлет *ConvertTo-SecureString*, который переводит текстовую строку в защищенный формат.

2. Теперь сделаем наш скрипт чуть более гибким. Используя командлет *Read-Host* заставим наш скрипт запрашивать имя и количество пользователей:

```
$org="OU=Students,DC=contoso,DC=com"
$username=Read-Host "Enter name"
$number=Read-Host "Enter number"
$count=1..$number foreach ($i in $count)
{ New-AdUser -Name $username$i -Path $org -Enabled $True -ChangePasswordAtLogon $true
`
-AccountPassword (ConvertTo-SecureString "p@$$w0rd" -AsPlainText -force) -passThru }
```

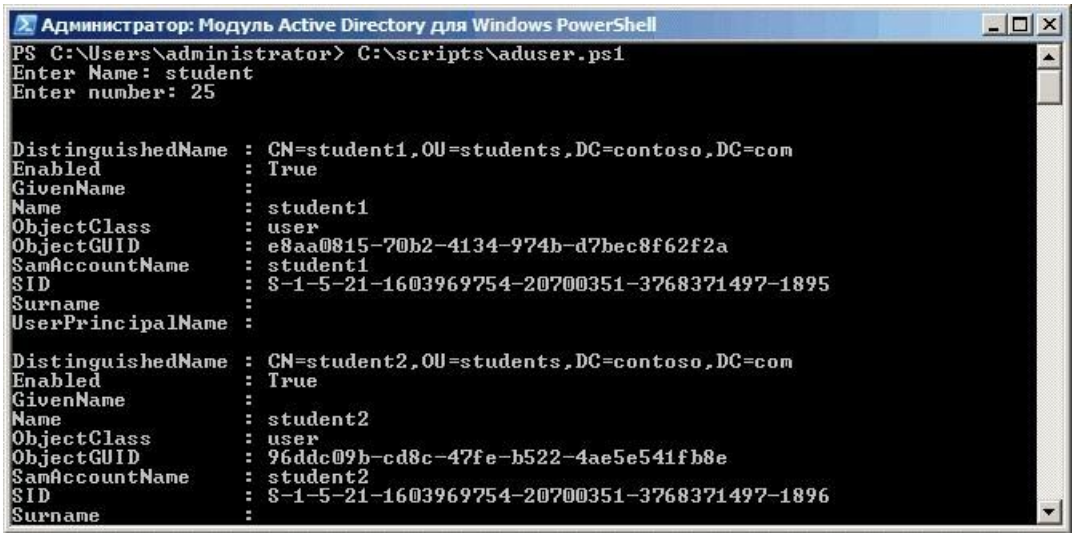


Рис. 55

Учетные записи созданы, пользователи могут заходить в систему и работать. Теперь их надо настроить — добавить в группы безопасности, прописать домашний каталог, сценарии входа и т.п. Сделать это можно с помощью шаблона. Проще говоря, создаем шаблонную учетную запись, полностью настраиваем ее, а затем делаем с нее нужное количество копий с помощью параметра *-Instance* :

```
$template = Get-AdUser -Identity "student"
$org="OU=Students,DC=contoso,DC=com"

$username=Read-Host "Enter name"
$number=Read-Host "Enter number"
$count=1..$number foreach ($i in $count)
{ New-AdUser -Name $username$i -UserPrincipalName $username$i -Path $org -Instance `
$template -Enabled $True -ChangePasswordAtLogon $true `
-AccountPassword (ConvertTo-SecureString "p@$$w0rd" -AsPlainText -force) -passThru }
```

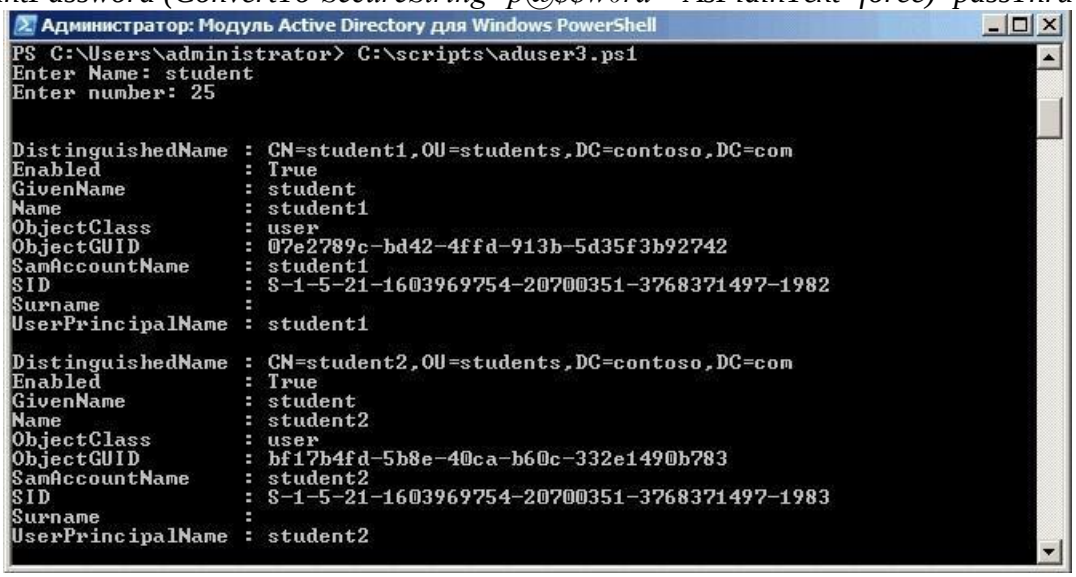
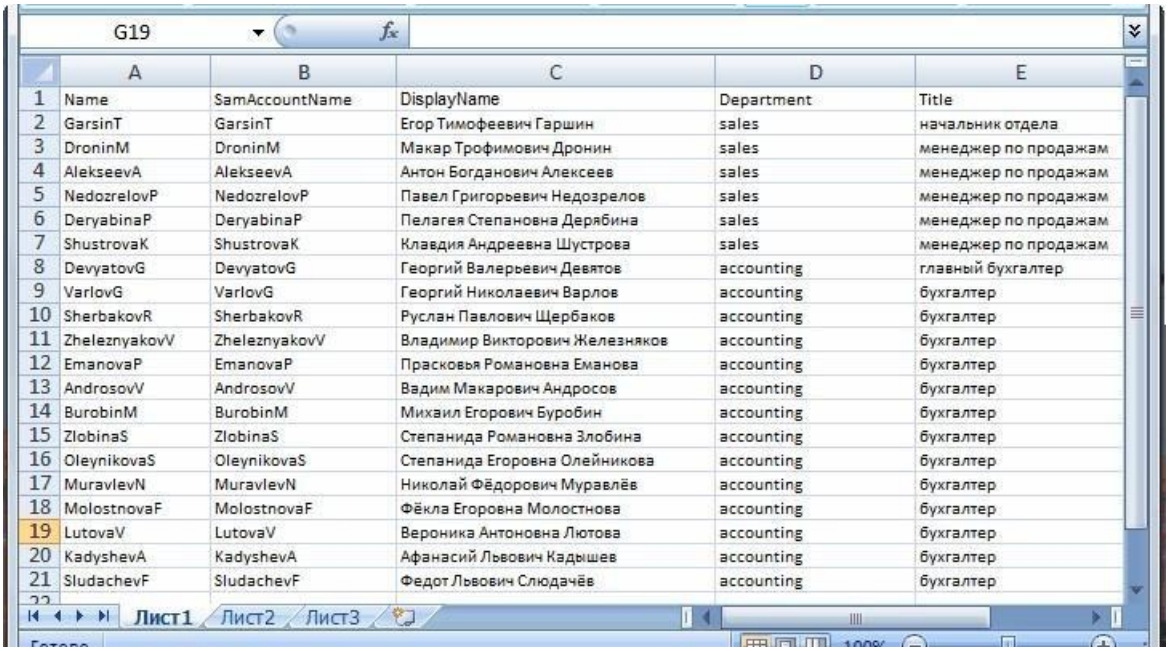


Рис. 56

3. Следующий способ автоматизировать создание учетных записей — импортировать их из CSV-файла. Этот способ подойдет в том случае, если вам предоставили список пользователей, и им надо завести учетные записи в соответствии с этим списком. Как пра-

вило, подобные списки создаются в Excel в виде таблицы со столбцами Имя, Должность, Отдел и т.п., примерно такого вида:



	A	B	C	D	E
1	Name	SamAccountName	DisplayName	Department	Title
2	GarsinT	GarsinT	Егор Тимофеевич Гаршин	sales	начальник отдела
3	DroninM	DroninM	Макар Трофимович Дронин	sales	менеджер по продажам
4	AlekseevA	AlekseevA	Антон Богданович Алексеев	sales	менеджер по продажам
5	NedozrelovP	NedozrelovP	Павел Григорьевич Недозрелов	sales	менеджер по продажам
6	DeryabinaP	DeryabinaP	Пелагея Степановна Дерябина	sales	менеджер по продажам
7	ShustrovaK	ShustrovaK	Клавдия Андреевна Шустрова	sales	менеджер по продажам
8	DevyatovG	DevyatovG	Георгий Валерьевич Девятов	accounting	главный бухгалтер
9	VarlovG	VarlovG	Георгий Николаевич Варлов	accounting	бухгалтер
10	SherbakovR	SherbakovR	Руслан Павлович Щербаков	accounting	бухгалтер
11	ZheleznyakovV	ZheleznyakovV	Владимир Викторович Железняков	accounting	бухгалтер
12	EmanovaP	EmanovaP	Прасковья Романовна Еманова	accounting	бухгалтер
13	AndrosovV	AndrosovV	Вадим Макарович Андросов	accounting	бухгалтер
14	BurobinM	BurobinM	Михаил Егорович Буробин	accounting	бухгалтер
15	ZlobinaS	ZlobinaS	Степанида Романовна Злобина	accounting	бухгалтер
16	OleynikovaS	OleynikovaS	Степанида Егоровна Олейникова	accounting	бухгалтер
17	MuravlevN	MuravlevN	Николай Фёдорович Муравлёв	accounting	бухгалтер
18	MolostnovaF	MolostnovaF	Фёкла Егоровна Молостнова	accounting	бухгалтер
19	LutovaV	LutovaV	Вероника Антоновна Лютова	accounting	бухгалтер
20	KadyshevA	KadyshevA	Афанасий Львович Кадышев	accounting	бухгалтер
21	SludachevF	SludachevF	Федот Львович Слюдачёв	accounting	бухгалтер

Рис. 57

Наша задача — сохранить его в формате CSV и затем указать в скрипте с помощью командлета *ImportCSV*. Если ваш CSV-файл содержит все необходимые столбцы, то *New-ADUser* автоматически свяжет их с правильными атрибутами пользователя :

```
$csv = Import-CSV -Path "C:\scripts\users.csv"
$csv | New-AdUser  -Path $org -Enabled $True -ChangePasswordAtLogon $true
`
```

```
-AccountPassword (ConvertTo-SecureString "p@$$w0rd" -AsPlainText -force) -passThru
```

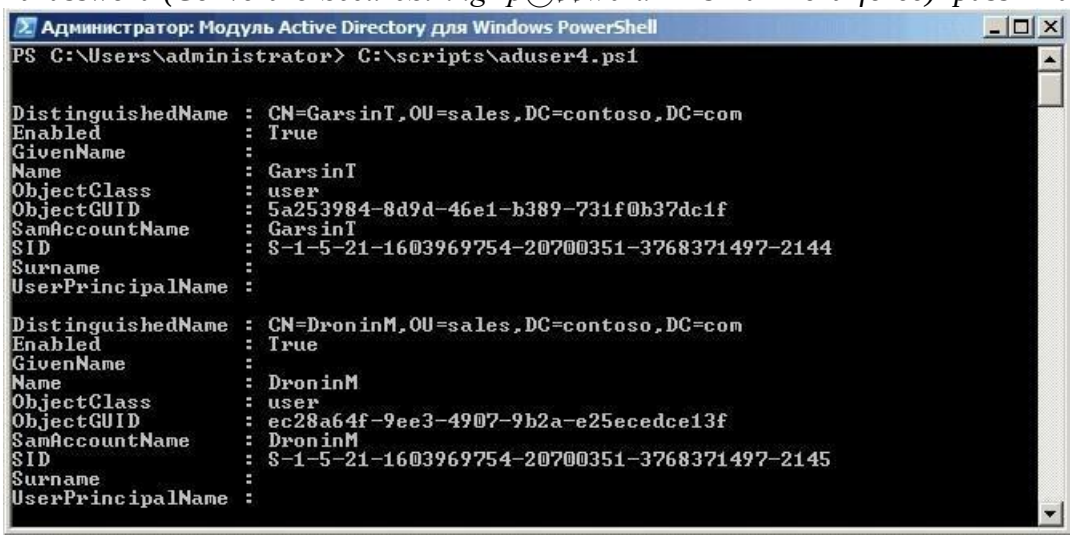


Рис. 58

Таким образом можно импортировать сотни новых пользователей за несколько секунд, но есть в этом методе и подводные камни:

- Названия столбцов должны **полностью** совпадать с названиями атрибутов пользователя, например Name (Имя), Organization (Организация), Title (должность), иначе ничего не получиться.

- В таблице **обязательно** нужно указать SamAccountName, в противном случае будет выдана ошибка о том, что учетная запись уже существует.
- Если атрибуты задаватьть в русской раскладке, как в нашем примере, то могут возникнуть проблемы с кодировкой. В решении этой проблемы мне помогло извлечение содержимго CSVфайла с помощью командлета *Get-Content* и сохранение его в другой CSV-файл: *Get-Content users.csv >> users1.csv*. После этого все русскоязычные атрибуты стали отображаться нормально.

Сделайте скриншоты (фотографии) процесса добавления пользователей домена Active Directory и вставьте в отчёт.

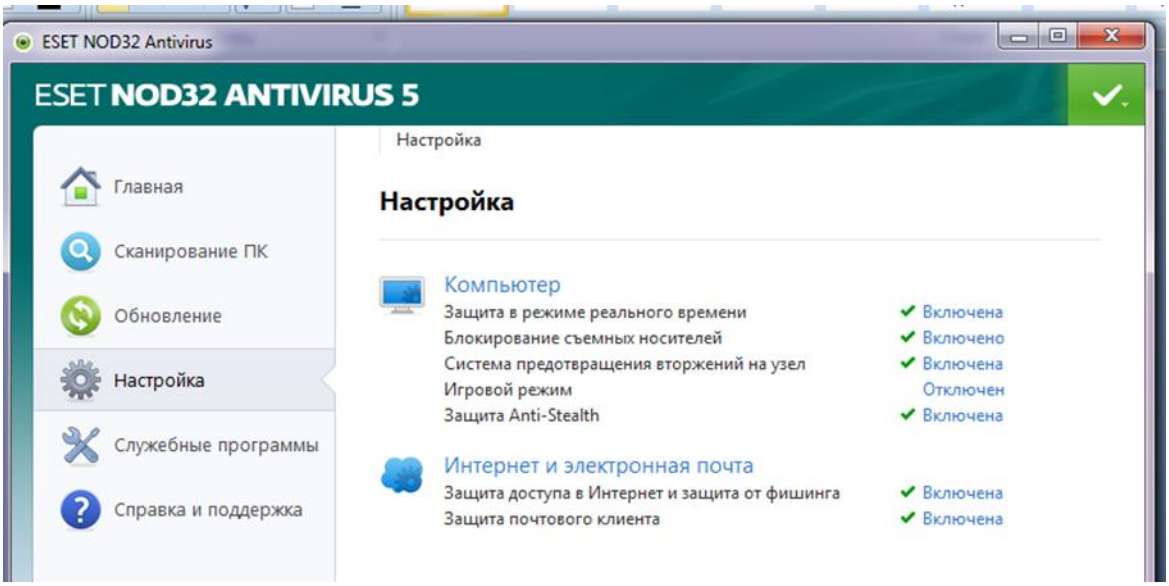
Задание 2:

1 Настройка антивирусной программы

Начнем с настройки Защиты компьютера. Откроем антивирусную про-грамму командой Пуск – Все программы – … или в правой части панели задач

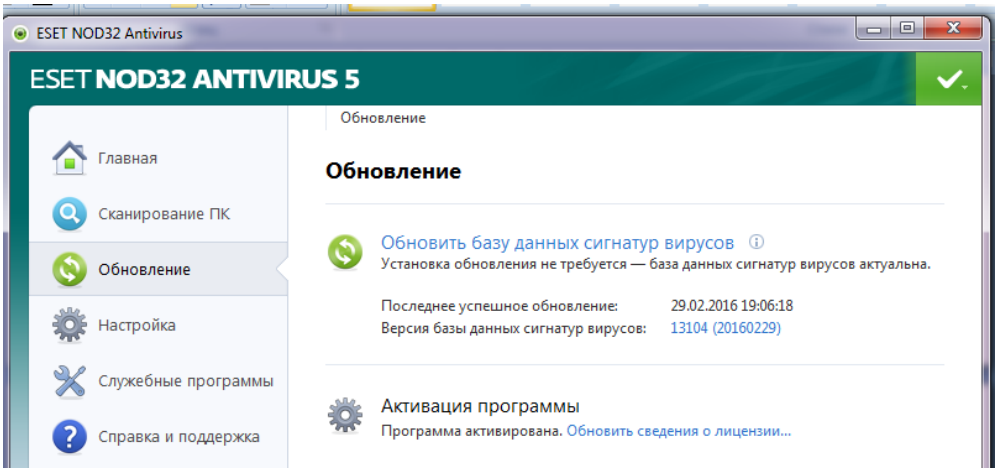
нажмем на значок антивирусной программы, например 

В открывшемся окне выберем раздел «Настройка», а в нем соответствующие параметры:



2 Обновление базы данных сигнатур вирусов

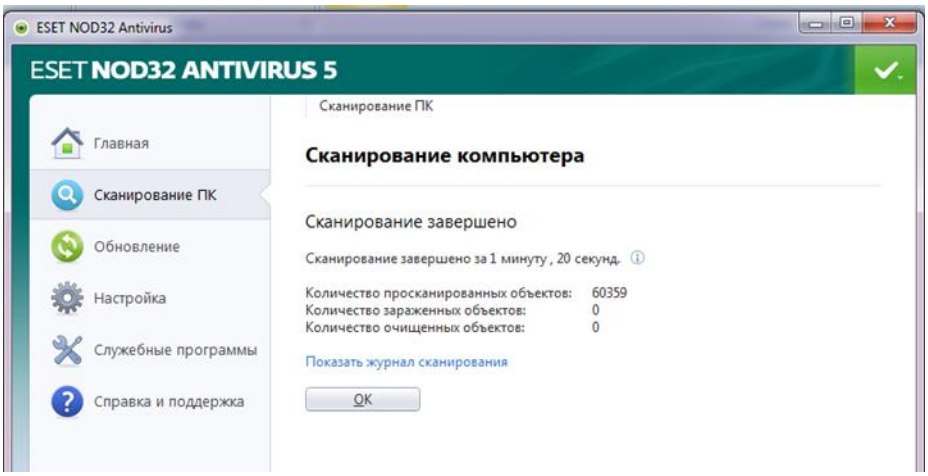
Просмотрим информацию о текущих базах, выбрав слева раздел ОБНОВЛЕНИЕ.



Обновим базу данных.

3 Сканирование дисков.

Для проверки дисков выберем раздел «Сканирование ПК», а в нем «Выборочное сканирование»:В открывшемся окне выберем диски для проверки и нажмем «Сканировать»:



Подождем окончания сканирования:

Познакомимся с отчетом:**Содержание отчета:** отчет по практической работе должен содержать: основные определения, рассуждения по выполнению заданий, необходимые изображения, вывод по работе.

2.4 Практическая работа№ 4
Администрирование рабочих станций.
Организация доступа к локальным сетям и Интернету

Задание 1:

1. Запускаем Управление компьютером → Планировщик заданий.
   Выберете справа Создание простой задачи → дайте ей любое имя → установите триггер однократно через 1 минуту → Действие Запуск программы → выберите любой exe-файл из System32 → Далее → Готово.
   Просмотрите Сводку планировщика заданий — есть ли там ваша задача?
   Если нет, попробуйте создать такую задачу, которую сможете увидеть в действии.
2. Создать в Планировщике заданий напоминание об «посещении habr» с помощью следующих действий: Создать простую задачу → при входе в Windows → запустить программу → C:\Windows\System32\msg.exe → Добавить аргументы — посетить habr → Далее → Поставьте флажок на открытие свойств после Готово →Готово.

В свойствах поставьте Выполнение для всех пользователей.
Флажок на Выполнять с наивысшими правами
На вкладке Параметры установите флажок на При сбое выполнения перезапускать через 1 минуту.
Для проверки работы данного задания, перезапустите вашу машину.
В отчёт вставить скриншот с напоминанием.
В отчёт вставить скриншот с информацией из Сводки в Планировщике заданий.

3. С помощью Создания задачи установите действие выключение (в system32 shutdown) через 2 минуты. Сохраните. Убедитесь, что работает. Должно появится окно, оповещающее о том, что вас сеанс будет завершён. Это окно в виде скриншота вставить в отчёт.
4. В Планировщике заданий → в Библиотеке планирования просмотрите все ваши Задачи. Вставьте информацию по этим задачам в отчёт в виде скриншота.

Из Библиотеки задачи можно изменять, удалять, запускать, завершать и т.п.

5. Одну задачу Удалите.

Одну задачу Выполните.
Одну задачу Отключите.
К одной задаче поменяйте Свойства.
Все изменения/запуски и т.п. зафиксировать скриншотами.

6. На Вкладке Журнал можно просмотреть все Сведения по задаче, на которой в данный момент находится курсор.

По любой задаче просмотрите Сведения и информацию о работе задачи вставьте в отчёт в виде скриншота.

Задание 2:

1. Установить роль удаленного доступа.

Для этого в оснастке Server Manager запускаем мастер добавления ролей и выбираем роль «Remote Access» со всеми дополнительными фичами.

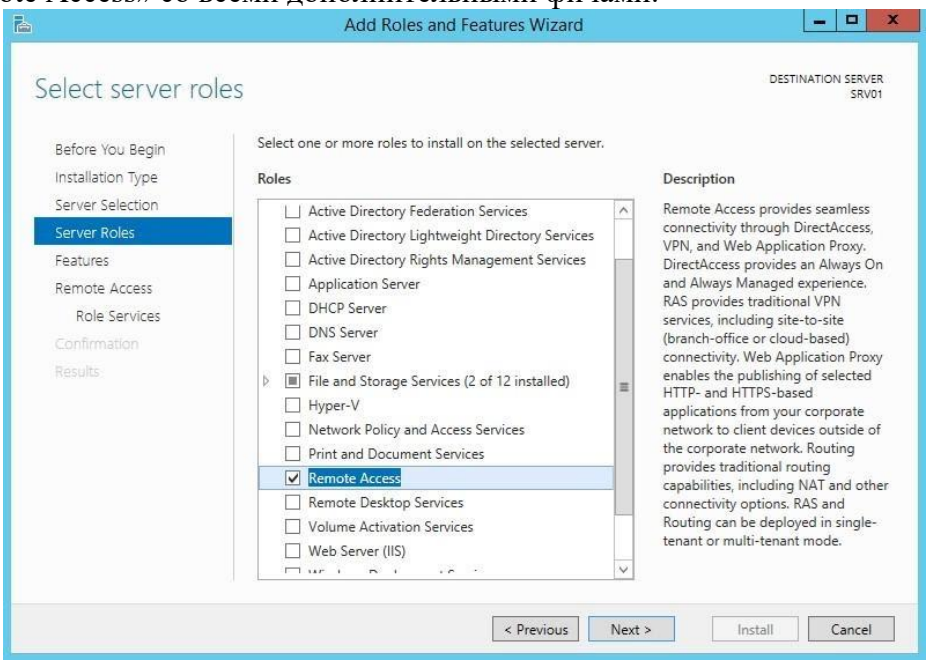


Рис. 133

И затем в списке сервисов для данной роли выбираем «DirectAccess and VPN (RAS)».

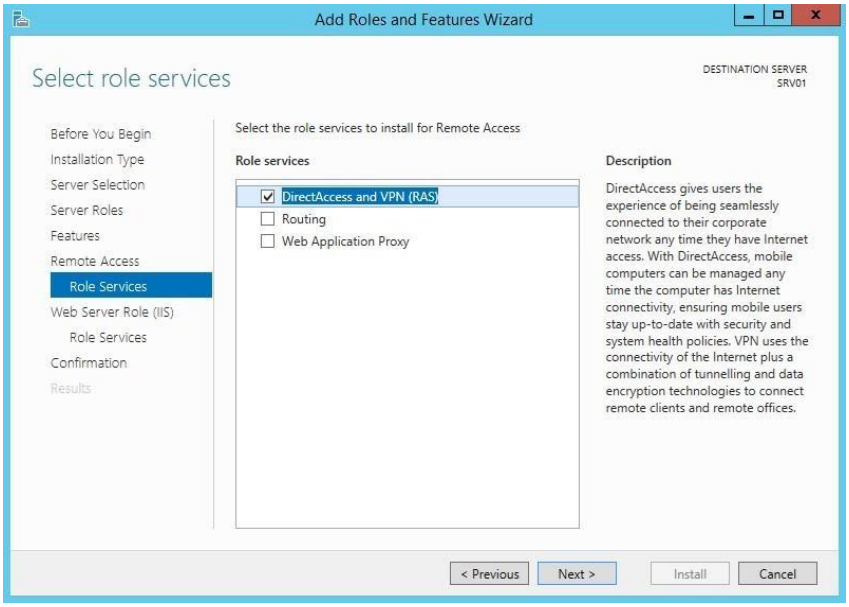


Рис. 134

Кроме роли удаленного доступа и инструментов управления будут дополнительно установлены web-сервер IIS и внутренняя база данных Windows. Полный список устанавливаемых компонентов можно просмотреть в финальном окне мастера, перед подтверждением запуска установки.

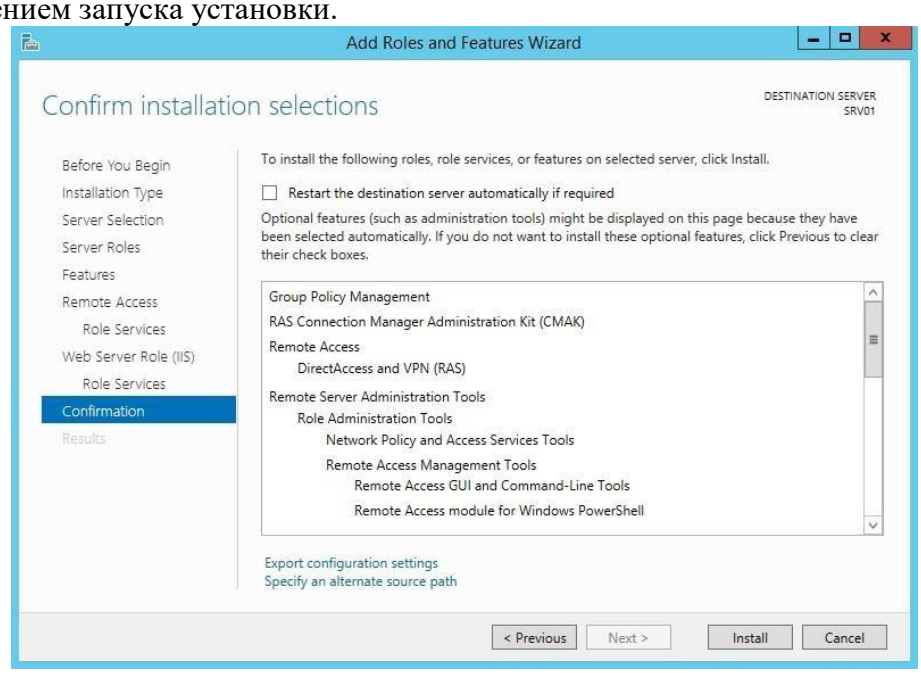


Рис. 135

Все то же самое, только гораздо быстрее, можно проделать с помощью PowerShell. Для этого надо открыть консоль и выполнить команду:

Install-WindowsFeature -Name Direct-Access-VPN -IncludeAllSubFeature In-cludeManagementTools

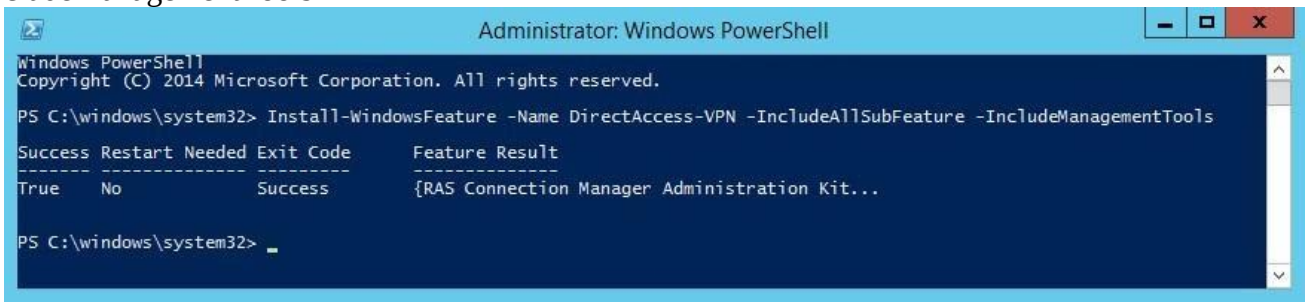


Рис. 136

После установки роли нам потребуется включить и настроить службу с помощью оснастки «Routing and Remote Access». Для ее открытия жмем **Win+R** и вводим команду **rrasmgmt.msc.**

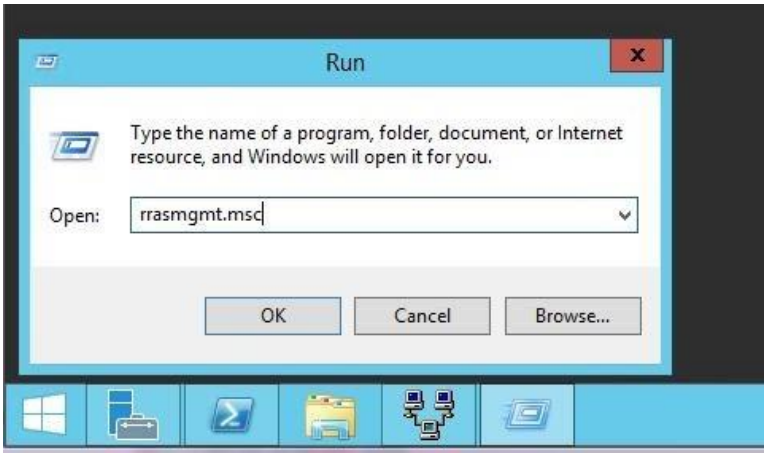


Рис. 137

В оснастке выбираем имя сервера, жмем правой клавишей мыши и в открывшемся меню выбираем пункт «Configure and Enable Routing and Remote Access».

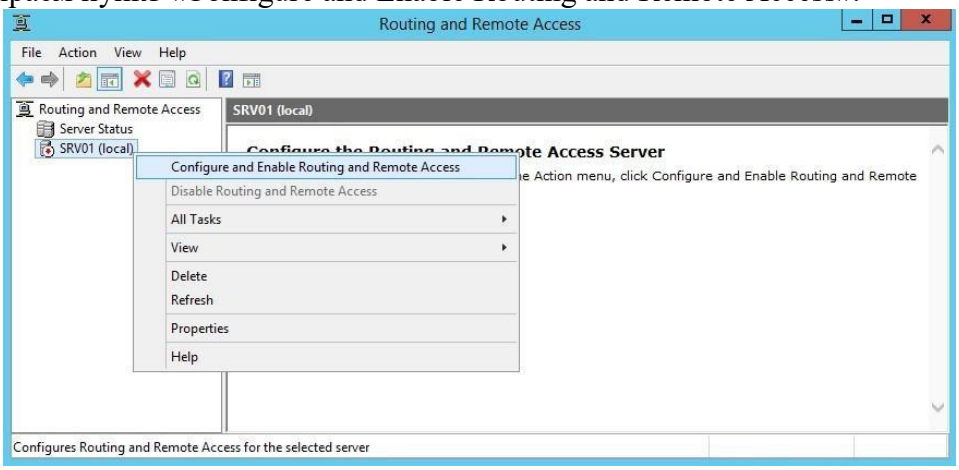


Рис. 138

В окне мастера настройки выбираем пункт «Custom configuration».

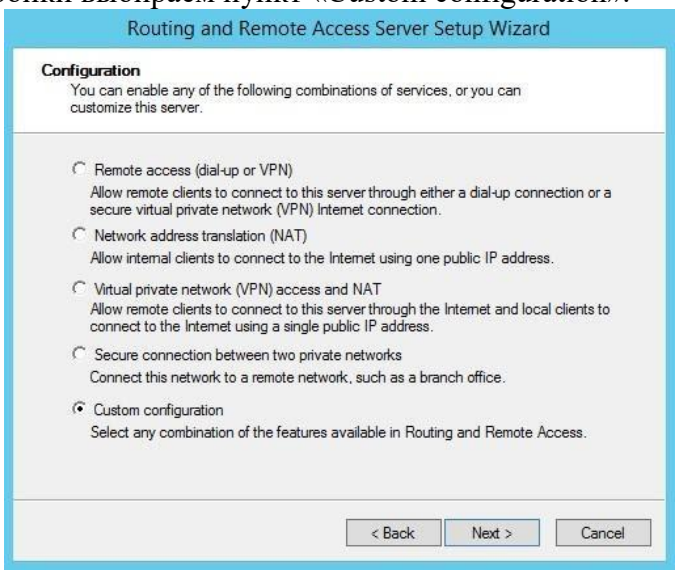


Рис. 139

И отмечаем сервис «VPN access».

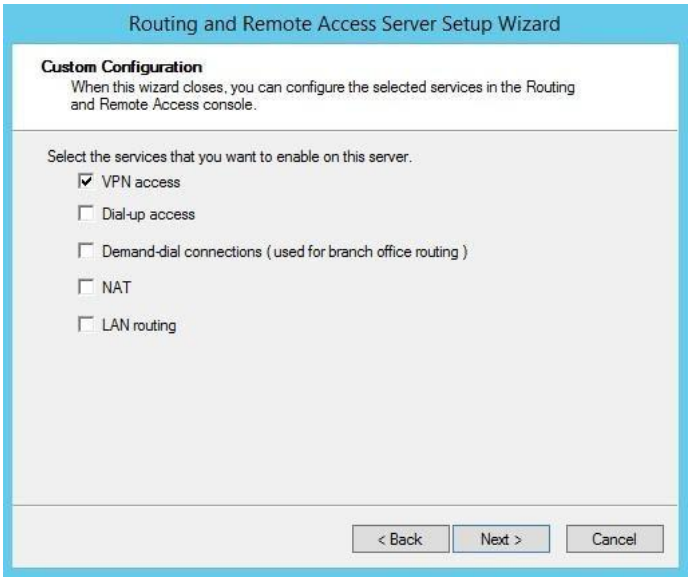


Рис. 140

В завершение настройки стартуем сервис удаленного доступа.

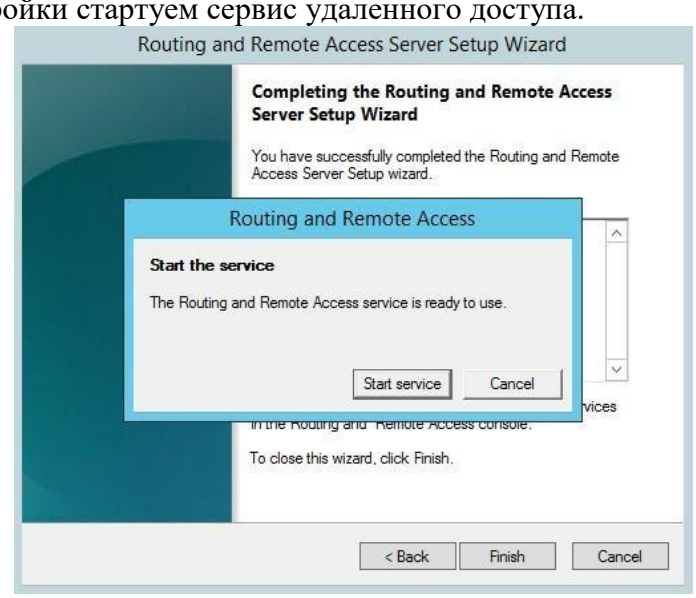


Рис. 141

Сервис VPN установлен и включен, теперь необходимо сконфигурировать его нужным нам образом. Опять открываем меню и выбираем пункт «Properties».

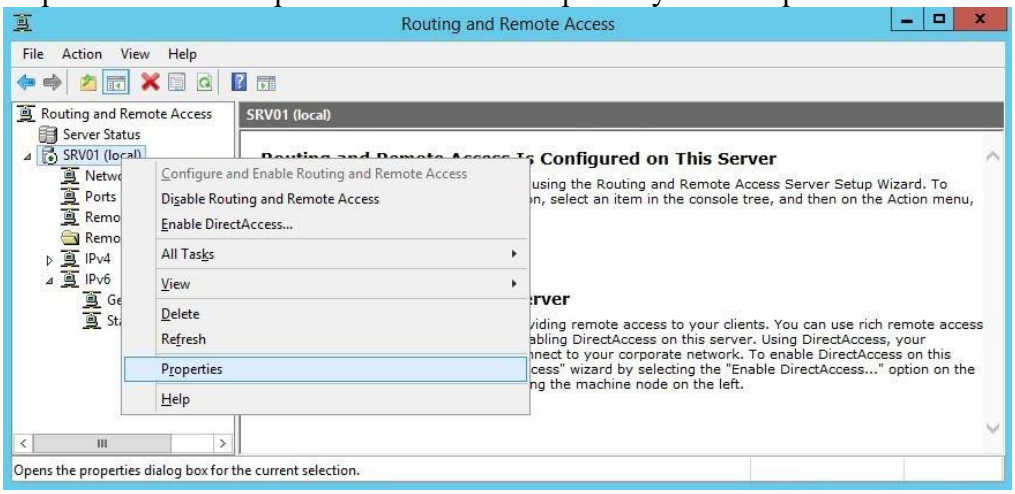


Рис. 142

Переходим на вкладку IPv4. Если у вас в сети нет DHCP сервера, то здесь надо задать диапазон IP адресов, которые будут получать клиенты при подключении к серверу.

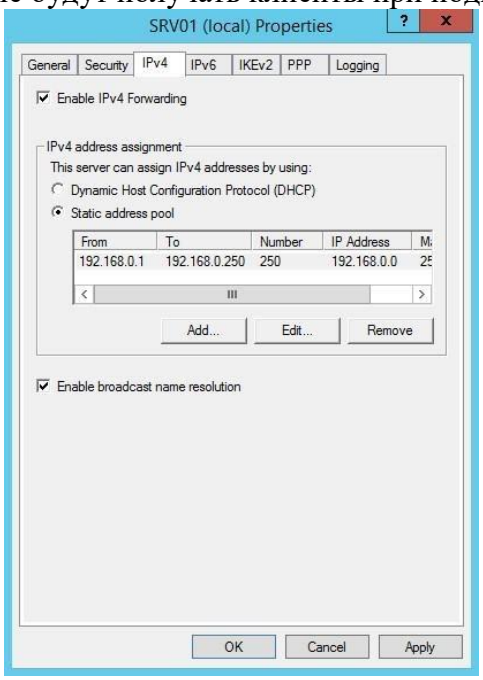


Рис. 143

Дополнительно на вкладке «Security» можно настроить параметры безопасности — выбрать тип аутентификации, задать предварительный ключ (preshared key) для L2TP или выбрать сертификат для SSTP.

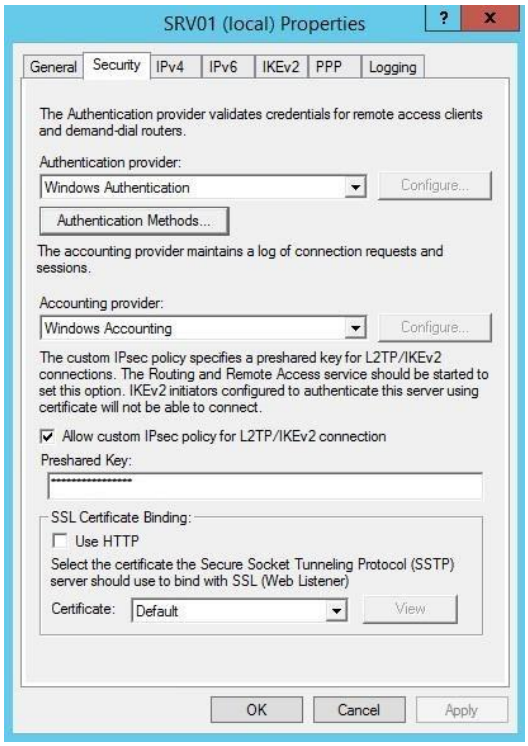


Рис. 144

Сделайте скриншоты (фотографии) процесса организации доступа к сети и вставьте в отчёт.

2.5 Практическая работа№ 5
Установка и сопровождение сетевых сервисов

Задание:

1. Запустить PowerShell. Подождать когда загрузится командная строка.
2. Набрать команду Get-Service.
   Можно при наборе команд набрать начало команды и далее воспользоваться Tab.
   Если вы не помните команду можно набрать get-command и далее маску для поиска, например *-servi*

PowerShell, как правило, не чувствителен к регистру.

В отчёт вставьте скриншот.

Каждая строка представляет собой объект службы (service object).Каждый сервисный объект, как правило, имеет свои свойства.

3. Откроем свойства служб, передав объекты в другую команду, Get-Member

PS C:\> get-service | get-member

Параметр Typename сверху говорит о том, что за объект перед нами;
в данном случае это System.ServiceProcess.ServiceController.

4. Посмотрим информацию о Windows Update через Get-Service. Получим информацию о некоторых ее свойствах.

PS C:\> get-service wuauserv | select Displayname,Status,Can*

Следует выключить данную службу.
Для этого нужно узнать нужную команду, воспользуемся get-command *-servi*
Найдите команду и остановите службу wuauserv.
Если у вас появляется ошибка, значит вы скорее всего запустили PowerShell не под администратором.

В отчёт скриншот с командами.

5. С помощью команды

PS C:\> help get-service –full

просмотрите подробный список параметром команды (передвигаться по тексту с помощью кнопки Enter)

6. Информацию о службе можно получить по ее имени или даже начальным буквам имени

PS C:\> get-service wi*

На экране отобразятся все службы, соответствующие этой маске, статус служб и поясняющая информация об этой службе.

В отчёт скриншот с командами.

7. Если вам необходимо узнать статус службы на удаленных компьютерах, то есть команда

PS C:\> get-service spooler -ComputerName имя_ПК

где spooler — это служба.
— выполните эту команду на своём ПК, поставив вместо имени_ПК localhost;
— выполните эту команду с любым другим ПК в аудитории.
С помощью команды test-connection имя_ПК можно протестировать есть ли соединение по локальной сети с ПК.

В отчёт скриншот с командами.

8. Фильтрация служб осуществляется с помощью командлета Where-Object (where – сокращение для командлета). Все, что нужно от PowerShell в этом случае, так это получить только те службы, у которых статус равен “stopped”.

PS C:\> get-service | where {$_.status -eq 'stopped'}

($_. — текущая переменная конвейера;
eq — оператор сравнения, означающий «равно»)
PowerShell получает информацию обо всех службах и передает их (с помощью “|”)
в следующую команду, которая осуществляет просмотр каждого объекта. Если свойство статуса объекта равно “stopped”, она остается в конвейере (pipeline), в противном случае она из него исключается.

В отчёт скриншот с командами.

9. Теперь давайте попробуем найти одну службу на нескольких машинах.

Вывод отформатируем в таблицу.

PS C:\> get-service -computername @(*'chi-dc01','chi-dc02','chi-dc03'*) | where {$_.name -eq 'wuauserv'} | format-table Name,Status,Machinename –autosize

(*'chi-dc01','chi-dc02','chi-dc03' — одномерный массив имён ПК в сети)*

В отчёт скриншот с командами.

В отчёте написать что делает данная команда.

10. Можно комбинировать запрос отдельных служб с их фильтрацией.

PS C:\> get-service "win*" -comp chi-dc03 | where {$_.status -eq 'running'}

Эта команда находит все службы на компьютере CHI-DC03, которые начинаются с ‘WIN’, но отображает только те, которые запущены.

Выполните команду с любым ПК в сети. Если не работает данная команда с ПК в сети можно использовать localhost. В этом случае получите информацию по данному ПК.

11. Можно сгруппировать объекты по свойству статуса (status property).

PS C:\> $dc03 = get-service -computername chi-dc03 | Group-Object -Property Status

Переменная $dc03 является объектом GroupInfo.

Выполните команду с любым ПК в сети. Если не работает данная команда с ПК в сети можно использовать localhost. В этом случае получите информацию по данному ПК.

12. Проверьте на удалённом ПК или на своём ПКстатус службы dns

PS C:\> get-service dns -ComputerName имя_ПК –RequiredServices

Параметр -RequiredServices передаст объект в конвейер для каждой требуемой службы. Вы можете даже пойти дальше и проверить требуемые службы для работы данной службы.

В отчёт скриншот с командами.

13. Для получения всех зависимых служб воспользуйтесь командой:

PS C:\> get-service –DependentServices

Это не даст вам особо полезную информацию, лучше осуществлять запрос по конкретным службам.

PS C:\> get-service служба -comp localhost -RequiredServices | Sort Machinename,Name | Format-table -GroupBy machinename

Выберите любые две запущенные службы из списка и выясните зависимости этих служб от других служб.

В отчёт скриншот с командами.

14. Проверьте статус службы wuauserv

get-service wuauserv

Если служба остановлена, то запустите её:

get-service wuauserv | start-service

Проверьте статус

Далее остановите службу.

В отчёт скриншот с командами.

15. Выберите из списка служб любую остановленную службу, проверьте статус её, далее запустите, потом остановите.

В отчёт скриншот с командами.

16. Для рестарта служб используется команда

restart-service имя_службы

например, сделайте рестарт для службы spooler

Вы не сможете убедиться, что служба снова запущена.

Для этого добавьте параметр –Passthru в конец команды.

Сделайте рестарт любым двум службам.

В отчёт скриншот с командами.

17. С помощью команды

PS C:\> get-service bits | select *

просмотреть данные о процессе bits.Данный процесс нужно приостановить.

Если значение свойства CanPauseAndContinue равно **True**, значит мы можем приостанавливать и возобновлять работу службы.

Процесс bits приостановить невозможно.

18. С помощью команды:

PS C:\> get-service | where {$_.CanPauseandContinue}

Мы увидим службы с нужным свойством

С помощью команды suspend-service имя_службы –PassThru

приостановите найденные службы
В отчёт скриншот с командами.
19. Для возобновления работы служб воспользуйтесь командой resume-service:
Все приостановленные службы запустите
В отчёт скриншот с командами.
Для рестарта служб на удалённых компьютерах (например, серверах)
используют команду
PS C:\> Invoke-Command {restart-service dns –passthru} –comp имя_ПК
Попробуйте воспользоваться данной командой и перезапустить какую-нибудь службу на удалённом ПК. В случае неудачи этой же командой перезапустите службу на localhost.
20. Для остановки службы на удалённом ПК используют следующую команду:
PS C:\> set-service wuauserv -ComputerName имя_ПК -Status stopped –WhatIf
Попробуйте воспользоваться данной командой и остановить службу на удалённом ПК. В случае неудачи этой же командой остановите службу на localhost.
В отчёт скриншот с командами.
21. Для установления типа запуска службы используется команда:
PS C:\> set-service remoteregistry -StartupType Manual –WhatIf
В отчёте ответьте на вопрос: какой тип запуска вы выставили и какие типы запуска бывают?
Выберите любые две службы и выставите им любой тип запуска.
В отчёт скриншот с командами.
22. С помощью команды:
PS C:\> get-service remoteregistry | select *
просмотрите свойства службы Удалённый реестр, обратите внимание на статус.
С помощью команды из предыдущего пункта поставьте тип запуска на Запущен.
В отчёт скриншот с командами.

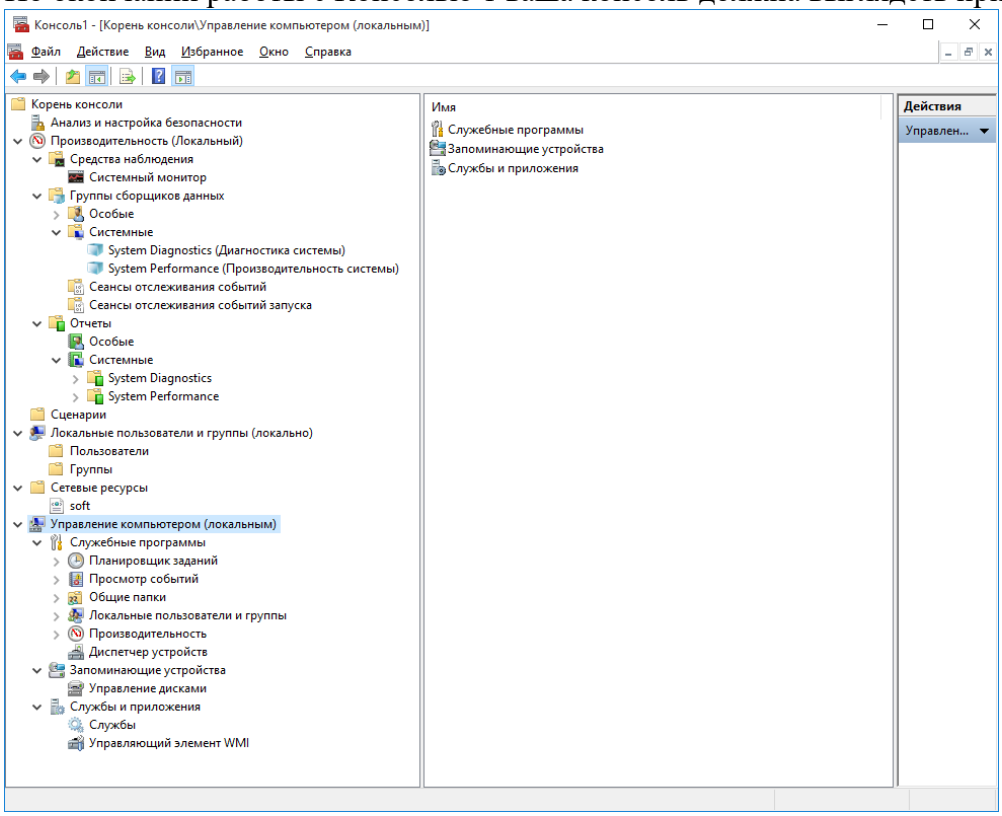
2.6 Практическая работа№ 6
Сбор данных для анализа использования программно-технических средств компьютерных сетей

Задание:

1. На вашей виртуальной машине с ОС Windows открыть список запущенных служб и вставить скриншот в отчёт;
2. В ОС Windows изменить список программ в Автозагрузке и вставить скриншот в отчёт;
3. В ОС Windows открыть события системы и вставить скриншот в отчёт.
4. В ОС Windows создать пользователя с простыми паролем и запретить пользователю смену пароля;
5. Внести пользователя в группу Операторы архива и сделать скриншот.
6. Дать пользователю привилегию на изменение системного времени на компьютере, сделать скриншот в отчёт.
7. В ОС Windows в реестре выбрать любую маленькую ветвь и экспортировать на Рабочий стол, сделать скриншот.
8. Открыть для редактирования в Блокноте экспортированный файл из реестра и сделать скриншот в отчёт.
9. В ОС Windows запустить консоль администрирования, сохраните её на рабочий стол под именем Консоль 1, скриншот в отчёт;
10. Добавить в корень консоли оснастку Системный монитор и Анализ и настройка безопасности (Файл → Добавить или удалить оснастку). В отчёт скриншот.
11. В консоли открыть Производительность → Средства наблюдения → Системный монитор. На диаграмме выбрать «плюс» и добавить два счётчика: браузер и кэш. В отчёт скриншот.

12. Просмотрите графики кэша: чтений с копированием, синхронный чтений. В отчёт скриншоты.
13. Поменяйте тип графиков и скриншот в отчёт.
14. Добавить две новые оснастки: Папка и Локальные пользователи и группы.
15. Папку переименовать в Сценарии. Скриншот в отчёт.
16. С помощью контекстного меню открыть Новый вид панели задач. Установить Стиль для панели результатов «без списка», Стиль для описания заданий «текст». Скриншот в отчёт. Далее всё по умолчанию и Готово. Автоматически запуститься Мастер создания задачи.
17. В окне Мастер создания задачи выбираем Команда операционной системы. Далее набираем команду powershell, Параметры C:\Users\«Имя вашего пользователя»\Desktop\Консоль 1\script\new-user.ps1. Далее «по умолчанию», выбираем любой значок и Готово. Скриншот в отчёт.
18. Добавьте ещё раз оснастку Папка. При её добавлении выберете Дополнительно → Разрешить изменять родительскую оснастку. Далее выбрать в качестве Родительской оснастки папку, затем в неё добавить оснастку Ссылка на веб-ресурсы. Далее в пути написать \\server1\soft. Далее и Готово. Папку переименовать в Сетевые ресурсы. Скриншот в отчёт.
19. Добавьте оснастку Управление компьютером. Попробуйте найти в сети компьютеры. Если ничего не находит, то Локальным компьютером. Скриншот в отчёт.

По окончании работы с Консолью 1 ваша консоль должна выглядеть примерно так:



2.7 Практическая работа№ 7
Обеспечение сетевой безопасности

Исходные данные

Топология

Таблица адресации

Устройство	Интерфейс	IP-адрес	Маска подсети	Шлюз по умолчанию
R1	G0/1	192.168.1.1	255.255.255.0	—
S1	VLAN 1	192.168.1.11	255.255.255.0	192.168.1.1
PC-A	NIC	192.168.1.3	255.255.255.0	192.168.1.1

Все сетевые устройства рекомендуется настраивать с использованием хотя бы минимального набора эффективных команд обеспечения безопасности. Это относится к устройствам конечных пользователей, серверам и сетевым устройствам, таким как маршрутизаторы и коммутаторы.

В ходе лабораторной работы вы должны будете настроить сетевые устройства в топологии таким образом, чтобы разрешать SSH-соединения для удаленного управления. Кроме того, вы должны будете настроить основные эффективные меры обеспечения безопасности через интерфейс командной строки операционной системы Cisco IOS. Затем вам необходимо будет протестировать меры обеспечения безопасности и убедиться в том, что они правильно внедрены и работают без ошибок.

Примечание. В практических лабораторных работах CCNA используются маршрутизаторы с интегрированными сервисами Cisco 1941 (ISR) под управлением Cisco IOS версии 15.2(4) M3 (образ universalk9). Также используются коммутаторы Cisco Catalyst 2960 с операционной системой Cisco IOS версии 15.0(2) (образ lanbasek9). Можно использовать другие маршрутизаторы, коммутаторы и версии Cisco IOS. В зависимости от модели устройства и версии Cisco IOS доступные команды и результаты их выполнения могут отличаться от тех, которые показаны в лабораторных работах. Точные идентификаторы интерфейса см. в сводной таблице по интерфейсам маршрутизаторов в конце лабораторной работы.

Примечание. Убедитесь, что у всех маршрутизаторов и коммутаторов была удалена начальная конфигурация. Если вы не уверены, обратитесь к инструктору.

Необходимые ресурсы

- 1 маршрутизатор (Cisco 1941 с ПО Cisco IOS версии 15.2(4)M3 с универсальным образом или аналогичная модель)
- 1 коммутатор (Cisco 2960 с ПО Cisco IOS версии 15.0(2) с образом lanbasek9 или аналогичная модель)
- 1 ПК (под управлением Windows 7 или 8 с программой эмуляции терминала, например, Tera Term)
- Консольные кабели для настройки устройств Cisco IOS через консольные порты
- Кабели Ethernet, расположенные в соответствии с топологией.

Часть 1: Настройка основных параметров устройств

В части 1 потребуется настроить топологию сети и основные параметры, такие как IP-адреса интерфейсов, доступ к устройствам и пароли на устройствах.

Задание

Шаг 1: Создайте сеть согласно топологии.

Подключите устройства, показанные в топологии, и кабели соответствующим образом.

Шаг 2: Выполните инициализацию и перезагрузку маршрутизатора и коммутатора.

Шаг 3: Выполните настройку маршрутизатора и коммутатора.

a. Подключитесь к устройству с помощью консольного подключения и активируйте привилегированный режим EXEC.

b. Назначьте устройству имя в соответствии с таблицей адресации.

c. Отключите поиск DNS, чтобы предотвратить попытки маршрутизатора неверно преобразовывать введенные команды таким образом, как будто они являются именами узлов.

d. Назначьте **class** в качестве зашифрованного пароля привилегированного режима EXEC.

e. Назначьте **cisco** в качестве пароля консоли и включите вход в систему по паролю.

f. Назначьте **cisco** в качестве пароля VTY и включите вход в систему по паролю.

g. Создайте баннер с предупреждением о запрете несанкционированного доступа к устройству.

h. Настройте и активируйте на маршрутизаторе интерфейс G0/1, используя информацию, приведенную в таблице адресации.

i. Задайте для используемого по умолчанию интерфейса SVI сведения об IP-адресе согласно таблице адресации.

j. Сохраните текущую конфигурацию в файл загрузочной конфигурации.

Часть 2: Настройка базовых мер безопасности на маршрутизаторе

Шаг 1: Зашифруйте открытые пароли.

R1(config)# **service password-encryption**

Шаг 2: Установите более надежные пароли.

Администратор должен следить за тем, чтобы пароли отвечали стандартным рекомендациям по созданию надежных паролей. В рекомендациях должны быть определены сочетания в пароле букв, цифр и специальных символов и его минимальная длина.

Примечание. Согласно данным рекомендациям по лучшим практическим методикам надежные пароли, примеры которых приведены в этой лабораторной работе, необходимо всегда использовать в реальной работе. Однако для упрощения выполнения работы в остальных лабораторных работах данного курса используются пароли cisco и class.

a. Измените зашифрованный пароль привилегированного режима EXEC в соответствии с рекомендациями.

R1(config)# **enable secret Enablep@55**

b. Установите минимальную длину 10 символов для всех паролей. R1(config)# **security**

passwords min-length 10

Шаг 3: Разрешите подключения по протоколу SSH.

a. В качестве имени домена укажите **CCNA-lab.com**.

R1(config)# **ip domain-name CCNA-lab.com**

b. Создайте в базе данных локальных пользователей запись, которая будет использоваться при подключении к маршрутизатору через SSH. Пароль должен соответствовать стандартам надежных паролей, а пользователь — иметь права доступа уровня EXEC. Если уровень привилегий не задан в команде, то пользователь по умолчанию будет иметь права доступа EXEC (уровень 15).

R1(config)# **username SSHadmin privilege 15 secret Admin1p@55**

c. Настройте транспортный вход для линий VTY таким образом, чтобы они могли разрешать подключения по протоколу SSH, но не разрешали подключения по протоколу Telnet.

R1(config)# **line vty 0 4**
R1(config-line)# **transport input ssh**

d. Аутентификация на линиях VTY должна выполняться с использованием базы данных локальных пользователей.

```
R1(config-line)# login local
R1(config-line)# exit
```

e. Создайте ключ шифрования RSA с длиной 1024 бит.

```
R1(config)# crypto key generate rsa modulus 1024
```

Шаг 4: Обеспечьте защиту консоли и линий VTY.

a. Маршрутизатор можно настроить таким образом, чтобы он завершал сеанс подключения в случае отсутствия активности в течение заданного времени. Если сетевой администратор вошел в систему сетевого устройства, а потом был внезапно вынужден покинуть рабочее место, то по истечении установленного времени эта команда автоматически завершит сеанс подключения. Приведенные ниже команды обеспечивают закрытие сеанса линии связи через пять минут отсутствия активности.

```
R1(config)# line console 0
R1(config-line)# exec-timeout 5 0
R1(config-line)# line vty 0 4
R1(config-line)# exec-timeout 5 0
R1(config-line)# exit
R1(config)#
```

b. Команда, приведенная ниже, не разрешает вход в систему с использованием метода полного перебора. Маршрутизатор блокирует попытки входа в систему на 30 секунд, если в течение 120 секунд будет дважды введен неверный пароль. Низкое значение этого таймера установлено специально для данной лабораторной работы.

R1(config)# **login block-for 30 attempts 2 within 120** Что означает **2**

within 120 в приведенной выше команде?

2_____

неудачные попытки в течение 120 секунд

Что означает **block-for 30** в приведенной выше команде?

Шаг 5: Убедитесь, что все неиспользуемые порты отключены.

Порты маршрутизатора отключены по умолчанию, однако рекомендуется лишний раз убедиться, что все неиспользуемые порты отключены администратором. Для этого можно воспользоваться командой **show ip interface brief**. Все неиспользуемые порты, не отключенные администратором, необходимо отключить с помощью команды **shutdown** в режиме конфигурации интерфейса.

```
R1# show ip interface brief
Interface                  IP-Address      OK? Method Status                Protocol
Embedded-Service-Engine0/0 unassigned      YES NVRAM  administratively down down
GigabitEthernet0/0         unassigned      YES NVRAM  administratively down down
GigabitEthernet0/1         192.168.1.1     YES manual up                    up
Serial0/0/0                unassigned      YES NVRAM  administratively down down    Serial0/0/1                unassigned
YES NVRAM  administratively down down      R1#
```

Шаг 6: Убедитесь, что все меры безопасности внедрены правильно.

a. С помощью программы Tera Term подключитесь к маршрутизатору R1 по протоколу Telnet.

Разрешает ли R1 подключение по протоколу Telnet? Дайте пояснение.

_____

Нет, Telnet не был активирован во время настройки маршрутизатора.

_____

b. С помощью программы Tera Term подключитесь к маршрутизатору R1 по протоколу SSH.

Разрешает ли R1 подключение по протоколу SSH? _____Да.

c. Намеренно укажите неверное имя пользователя и пароль, чтобы проверить, будет ли заблокирован доступ к системе после двух неудачных попыток.

Что произошло после ввода неправильных данных для входа в систему во второй раз?

Маршрутизатор отклоняет входящие соединения по протоколу SSH.

d. Из сеанса подключения к маршрутизатору с помощью консоли отправьте команду **show login**, чтобы проверить состояние входа в систему. В приведенном ниже примере команда **show login** была введена в течение 30-секундной блокировки доступа к системе и показывает, что маршрутизатор находится в режиме Quiet. Маршрутизатор не будет разрешать попытки входа в систему в течение еще 14 секунд.

```
R1# show login
    A default login delay of 1 second is applied.
    No Quiet-Mode access list has been configured.

    Router enabled to watch for login Attacks.
    If more than 2 login failures occur in 120 seconds or less,  logins will be disabled for 30
seconds.

    Router presently in Quiet-Mode.
    Will remain in Quiet-Mode for 14 seconds.
    Denying logins from all sources.
R1#
```

e. По истечении 30 секунд повторите попытку подключения к R1 по протоколу SSH и войдите в систему, используя имя **SSHadmin** и пароль **Admin1p@55**.

Что отобразилось после успешного входа в систему? Баннер MOTD и интерпретатор .

f. Войдите в привилегированный режим EXEC и введите в качестве пароля **Enablep@55**.

Если вы неправильно вводите пароль, прерывается ли сеанс SSH после двух неудачных попыток в течение 120 секунд? Дайте пояснение.

Нет, так как login block-for защищает вход в консоль, а не в привилегированный режим EXEC.

g. Введите команду **show running-config** в строке приглашения привилегированного режима EXEC для просмотра установленных параметров безопасности.

Часть 3: Настройка базовых мер безопасности на коммутаторе

Шаг 1: Зашифруйте открытые пароли.

```
S1(config)# service password-encryption
```

Шаг 2: Установите более надежные пароли на коммутаторе.

Измените зашифрованный пароль привилегированного режима EXEC в соответствии с рекомендациями по установке надежного пароля.

```
S1(config)# enable secret Enablep@55
```

Примечание. Команда безопасности **password min-length** на коммутаторах модели 2960 недоступна.

Шаг 3: Разрешите подключения по протоколу SSH.

a. В качестве имени домена укажите **CCNA-lab.com**.

```
S1(config)# ip domain-name CCNA-lab.com
```

b. Создайте в базе данных локальных пользователей запись, которая будет использоваться при подключении к коммутатору через SSH. Пароль должен соответствовать стандартам надежных паролей, а пользователь — иметь права доступа уровня EXEC. Если уровень привилегий не задан в команде, то пользователь по умолчанию будет иметь права доступа EXEC (уровень 1).

S1(config)# **username SSHadmin privilege 1 secret Admin1p@55**

c. Настройте транспортный вход для линий VTY таким образом, чтобы они могли разрешать подключения по протоколу SSH, но не разрешали подключения по протоколу Telnet.

S1(config)# **line vty 0 15**
S1(config-line)# **transport input ssh**

d. Аутентификация на линиях VTY должна выполняться с использованием базы данных локальных пользователей.

S1(config-line)# **login local**
S1(config-line)# **exit**

e. Создайте ключ шифрования RSA с длиной 1024 бит.

S1(config)# **crypto key generate rsa modulus 1024**

Шаг 4: Обеспечьте защиту консоли и линий VTY.

a. Настройте коммутатор таким образом, чтобы он закрывал линию через десять минут отсутствия активности.

S1(config)# **line console 0**
S1(config-line)# **exec-timeout 10 0** S1(config-line)# **line vty 0 15**
S1(config-line)# **exec-timeout 10 0**
S1(config-line)# **exit**
S1(config)#

b. Чтобы помешать попыткам входа в систему с использованием метода полного перебора, настройте коммутатор таким образом, чтобы он блокировал доступ к системе на 30 секунд после двух неудачных попыток входа в течение 120 секунд. Низкое значение этого таймера установлено специально для данной лабораторной работы.

S1(config)# **login block-for 30 attempts 2 within 120** S1(config)# **end**

Шаг 5: Убедитесь, что все неиспользуемые порты отключены.

По умолчанию порты коммутатора включены. Отключите на коммутаторе все неиспользуемые порты. а.

Состояние портов коммутатора можно проверить с помощью команды **show ip interface brief**.

S1# **show ip interface brief**

Interface	IP-Address	OK?	Method	Status	Protocol
Vlan1	192.168.1.11	YES	manual	up	up
FastEthernet0/1	unassigned	YES	unset	down	down
FastEthernet0/2	unassigned	YES	unset	down	down
FastEthernet0/3	unassigned	YES	unset	down	down
FastEthernet0/4	unassigned	YES	unset	down	down
FastEthernet0/5	unassigned	YES	unset	up	up
FastEthernet0/6	unassigned	YES	unset	up	up
FastEthernet0/7	unassigned	YES	unset	down	down
FastEthernet0/8	unassigned	YES	unset	down	down
FastEthernet0/9	unassigned	YES	unset	down	down
FastEthernet0/10	unassigned	YES	unset	down	down
FastEthernet0/11	unassigned	YES	unset	down	down
FastEthernet0/12	unassigned	YES	unset	down	down
FastEthernet0/13	unassigned	YES	unset	down	down
FastEthernet0/14	unassigned	YES	unset	down	down
FastEthernet0/15	unassigned	YES	unset	down	down
FastEthernet0/16	unassigned	YES	unset	down	down
FastEthernet0/17	unassigned	YES	unset	down	down

FastEthernet0/18 unassigned YES unset down down
FastEthernet0/19 unassigned YES unset down down FastEthernet0/20 unassigned YES unset down down
FastEthernet0/21 unassigned YES unset down down
FastEthernet0/22 unassigned YES unset down down
FastEthernet0/23 unassigned YES unset down down
FastEthernet0/24 unassigned YES unset down down
GigabitEthernet0/1 unassigned YES unset down down GigabitEthernet0/2 unassigned YES unset down down

S1#

b. Чтобы отключить сразу несколько интерфейсов, воспользуйтесь командой **interface range**.

S1(config)# **interface range f0/1–4 , f0/7-24 , g0/1-2**
S1(config-if-range)# **shutdown**
S1(config-if-range)# **end**
S1#

c. Убедитесь, что все неактивные интерфейсы отключены администратором.

S1# **show ip interface brief**

Interface	IP-Address	OK?	Method	Status	Protocol
Vlan1	192.168.1.11	YES	manual	up	up
FastEthernet0/1	unassigned	YES	unset	administratively down	down
FastEthernet0/2	unassigned	YES	unset	administratively down	down
FastEthernet0/3	unassigned	YES	unset	administratively down	down
FastEthernet0/4	unassigned	YES	unset	administratively down	down
FastEthernet0/5	unassigned	YES	unset	up	up
FastEthernet0/6	unassigned	YES	unset	up	up
FastEthernet0/7	unassigned	YES	unset	administratively down	down
FastEthernet0/8	unassigned	YES	unset	administratively down	down
FastEthernet0/9	unassigned	YES	unset	administratively down	down
FastEthernet0/10	unassigned	YES	unset	administratively down	down
FastEthernet0/11	unassigned	YES	unset	administratively down	down
FastEthernet0/12	unassigned	YES	unset	administratively down	down
FastEthernet0/13	unassigned	YES	unset	administratively down	down
FastEthernet0/14	unassigned	YES	unset	administratively down	down
FastEthernet0/15	unassigned	YES	unset	administratively down	down
FastEthernet0/16	unassigned	YES	unset	administratively down	down
FastEthernet0/17	unassigned	YES	unset	administratively down	down
FastEthernet0/18	unassigned	YES	unset	administratively down	down
FastEthernet0/19	unassigned	YES	unset	administratively down	down
FastEthernet0/20	unassigned	YES	unset	administratively down	down
FastEthernet0/21	unassigned	YES	unset	administratively down	down
FastEthernet0/22	unassigned	YES	unset	administratively down	down
FastEthernet0/23	unassigned	YES	unset	administratively down	down
FastEthernet0/24	unassigned	YES	unset	administratively down	down
GigabitEthernet0/1	unassigned	YES	unset	administratively down	down
GigabitEthernet0/2	unassigned	YES	unset	administratively down	down

S1#

Шаг 6: Убедитесь, что все меры безопасности внедрены правильно.

a. Убедитесь, что протокол Telnet на коммутаторе отключен.

b. Подключитесь к коммутатору по протоколу SSH и намеренно укажите неверное имя пользователя и пароль, чтобы проверить, будет ли заблокирован доступ к системе.

c. По истечении 30 секунд повторите попытку подключения к R1 по протоколу SSH и войдите в систему, используя имя пользователя **SSHadmin** и пароль **Admin1p@55**.

Появился ли баннер после успешного входа в систему? _____

d. Войдите в привилегированный режим EXEC, используя **Enablep@55** в качестве пароля.

e. Введите команду **show running-config** в строке приглашения привилегированного режима EXEC для просмотра установленных параметров безопасности.

Вопросы для повторения

1. В части 1 для консоли и линий VTY в вашей базовой конфигурации была введена команда **password cisco**. Когда используется этот пароль после применения наиболее эффективных мер обеспечения безопасности?
2. Распространяется ли команда **security passwords min-length 10** на настроенные ранее пароли, содержащие меньше десяти символов?

Сводная таблица по интерфейсам маршрутизаторов

Сводная таблица по интерфейсам маршрутизаторов				
Модель маршрутизатора	**Интерфейс Ethernet № 1**	**Интерфейс Ethernet №2**	**Последовательный интерфейс № 1**	**Последовательный интерфейс №2**
1800	Fast Ethernet 0/0 (F0/0)	Fast Ethernet 0/1 (F0/1)	Serial 0/0/0 (S0/0/0)	Serial 0/0/1 (S0/0/1)
1900	Gigabit Ethernet 0/0 (G0/0)	Gigabit Ethernet 0/1 (G0/1)	Serial 0/0/0 (S0/0/0)	Serial 0/0/1 (S0/0/1)
2801	Fast Ethernet 0/0 (F0/0)	Fast Ethernet 0/1 (F0/1)	Serial 0/1/0 (S0/0/0)	Serial 0/1/1 (S0/0/1)
2811	Fast Ethernet 0/0 (F0/0)	Fast Ethernet 0/1 (F0/1)	Serial 0/0/0 (S0/0/0)	Serial 0/0/1 (S0/0/1)
2900	Gigabit Ethernet 0/0 (G0/0)	Gigabit Ethernet 0/1 (G0/1)	Serial 0/0/0 (S0/0/0)	Serial 0/0/1 (S0/0/1)

Примечание. Чтобы определить конфигурацию маршрутизатора, можно посмотреть на интерфейсы и установить тип маршрутизатора и количество его интерфейсов. Перечислить все комбинации конфигураций для каждого класса маршрутизаторов невозможно. Эта таблица содержит идентификаторы для возможных комбинаций интерфейсов Ethernet и последовательных интерфейсов на устройстве. Другие типы интерфейсов в таблице не представлены, хотя они могут присутствовать в данном конкретном маршрутизаторе. В качестве примера можно привести интерфейс ISDN BRI. Строка в скобках — это официальное сокращение, которое можно использовать в командах Cisco IOS для обозначения интерфейса.

2.8 Практическая работа№ 8
Проведение мониторинга сети

Задание:

Часть 1. Проверка вашего понимания мониторинга сети
Часть 2. Исследование средств мониторинга сети
Часть 3. Выбор средства мониторинга сети

Исходные данные/сценарий

Мониторинг необходим для сетей любого размера. Профилактическое наблюдение за сетевой инфраструктурой поможет администраторам сети выполнять ежедневные обязанности. Существуют различные средства мониторинга сети, которые отличаются по стоимости в зависимости от возможностей, количества сетей и поддерживаемых узлов.
В этой лабораторной работе будет исследоваться доступное ПО для наблюдения за сетью. Вашей задачей будет сбор информации о программных продуктах и функциях этих продуктов. Один продукт вы рассмотрите более подробно и перечислите некоторые из его основных функций.

Необходимые ресурсы:

ПК с доступом к Интернету.

Часть 1: Проверка знакомства с процессом наблюдения за сетью

Опишите, в чём, по вашему мнению, заключается процесс мониторинга сети. Приведите пример его использования в производственной сети.

Часть 2: Изучение средств мониторинга сети

Шаг 1: Проведите исследование и найдите три средства мониторинга сети.

Перечислите эти три найденных средства.

Шаг 2: Заполните следующую форму для выбранных средств мониторинга сети.

Поставщик	**Название продукта**	**Функциональные возможности**

Часть 3: Выберите средство мониторинга сети

Шаг 1: Выберите одно или несколько средств мониторинга из исследования.

Укажите одно или несколько средств из исследования, которые бы вы выбрали для мониторинга сети. Назовите эти средства и объясните свой выбор, перечислив конкретные функциональные возможности, которые по вашему мнению важны.

Шаг 2: Изучите средство мониторинга сети PRTG.

Перейдите на веб-страницу www.paessler.com/prtg.

В следующих полях приведите примеры некоторых функций PRTG.

2.9 Практическая работа№ 9
Принятие мер по восстановлению работоспособности локальной сети при сбоях или выходе из строя сетевого оборудования

Задание:

1. Настраиваем доменную аутентификацию на сетевом оборудовании

При обслуживании больших сетей системные администраторы часто сталкиваются с проблемами аутентификации на сетевом оборудовании. В частности, довольно сложно организовать нормальную работу нескольких сетевых администраторов под индивидуальными учетными записями на большом количестве оборудования (приходится вести и поддерживать в актуальном состоянии базу локальных учетных записей на каждом устройстве). Логичным решение было бы использовать для авторизации уже существующей базы учетных записей — Active Directory. В этой статье мы разберемся, как настроить **доменную (Active Directory) аутентификацию на активном сетевом оборудовании** (коммутаторы, маршрутизаторы).

Не все сетевое оборудование популярных вендоров (CISCO, HP, Huawei) поддерживает функционал для непосредственного обращения к каталогу LDAP, и такое решение не будет универсальным. Для решения нашей задачи подойдет протокол **AAA (Authentication**

Authorization and Accounting), фактически ставший стандартом де-факто для сетевого оборудования. Клиент AAA (сетевое устройство) отправляет данные авторизующегося пользователя на сервер **RADIUS** и на основе его ответа принимает решение о предоставлении / отказе доступа.
Протокол **Remote Authentication Dial In User Service (RADIUS)** в Windows Server 2012 R2 включен в роль **NPS (Network Policy Server)**. В первой части статьи мы установим и настроим роль Network Policy Server, а во второй покажем типовые конфигурации сетевого устройств с поддержкой RADUIS на примере **коммутаторов HP Procurve** и **оборудования Cisco**.

2. Установка и настройка сервера с ролью Network Policy Server

Как правило, сервер с ролью NPS рекомендуется устанавливать на выделенном сервере (не рекомендуется размещать эту роль на контроллере домена). В данном примере роль NPS мы будем устанавливать на сервере с Windows Server 2012 R2.
Откройте консоль **Server Manager** и установите роль **Network Policy Server** (находится в разделе **Network Policy and Access Services**).

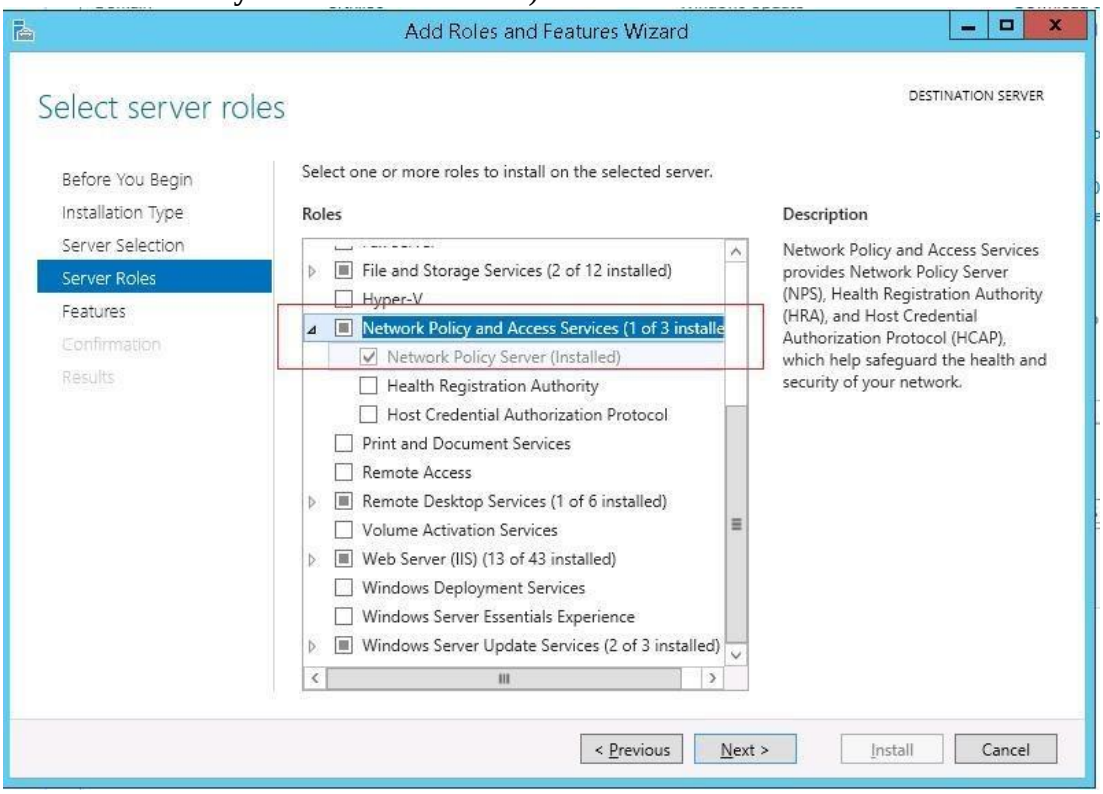


Рис. 104

После окончания установки запустите mmc-консоль управления Network Policy Server. Нас интересуют три следующих раздела консоли:

1. **RADIUS Clients** — содержит список устройств, которые могут аутентифицироваться на сервере
2. **Connection Request Policies** – определяет типы устройств, которые могут аутентифицироваться
3. **Network Polices** – правила аутентификации

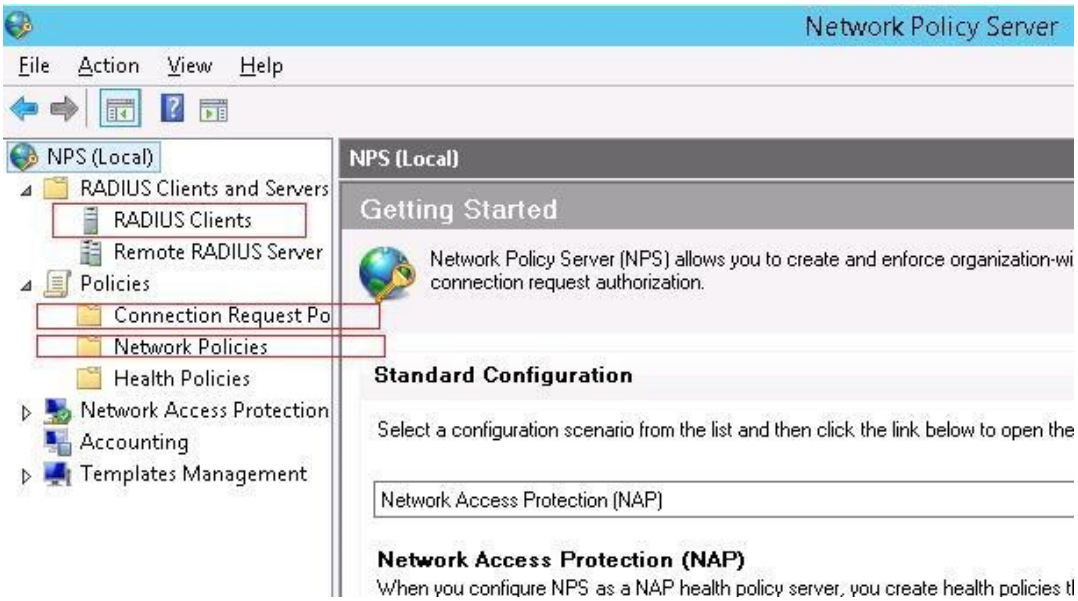


Рис. 105

Добавим нового клиента RADIUS (это будет коммутатор HP ProCurve Switch 5400zl), щелкнув ПКМ по разделу **RADIUS Clients** и выбрав **New**. Укажем:

- **Friendly Name**:sw-HP-5400-1
- **Address (IP or DNS)**: 10.10.10.2
- **Shared secret** (пароль/секретный ключ): пароль можно указать вручную (он должен быть достаточно сложным), либо сгенерировать с помощью специальной кнопки (сгенерированный пароль необходимо скопировать, т.к. в дальнейшем его придется указать на сетевом устройстве).

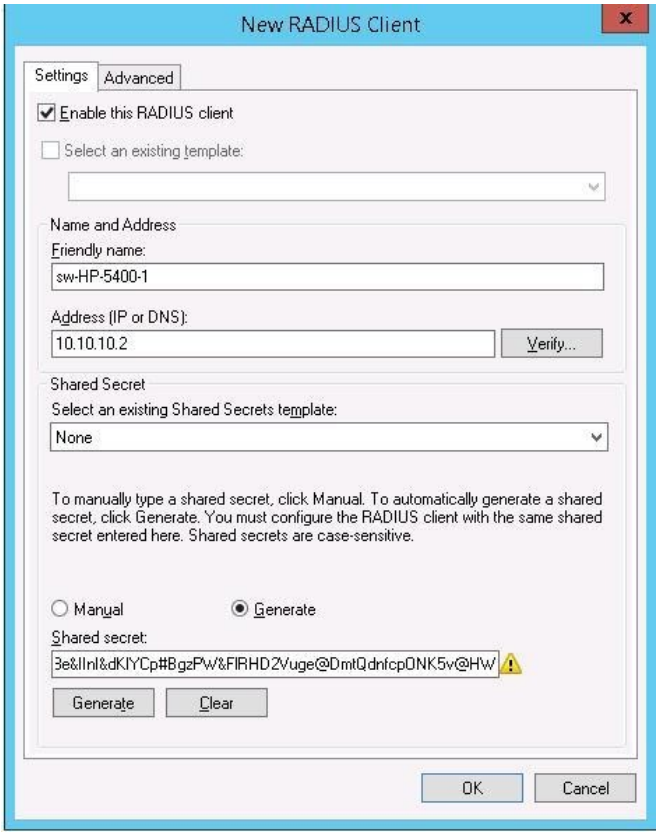


Рис. 106

Отключим стандартную политику (**Use Windows authentication for all users**) в разделе Connection Request Policies, щелкнув по ней ПКМ и выбрав **Disable**.

Создадим новую политику с именем **Network-Switches-AAA** и нажимаем далее. В разделе **Condition** создадим новое условие. Ищем раздел **RADIUS Client Properites** и выбираем **Client Friendly Name**.

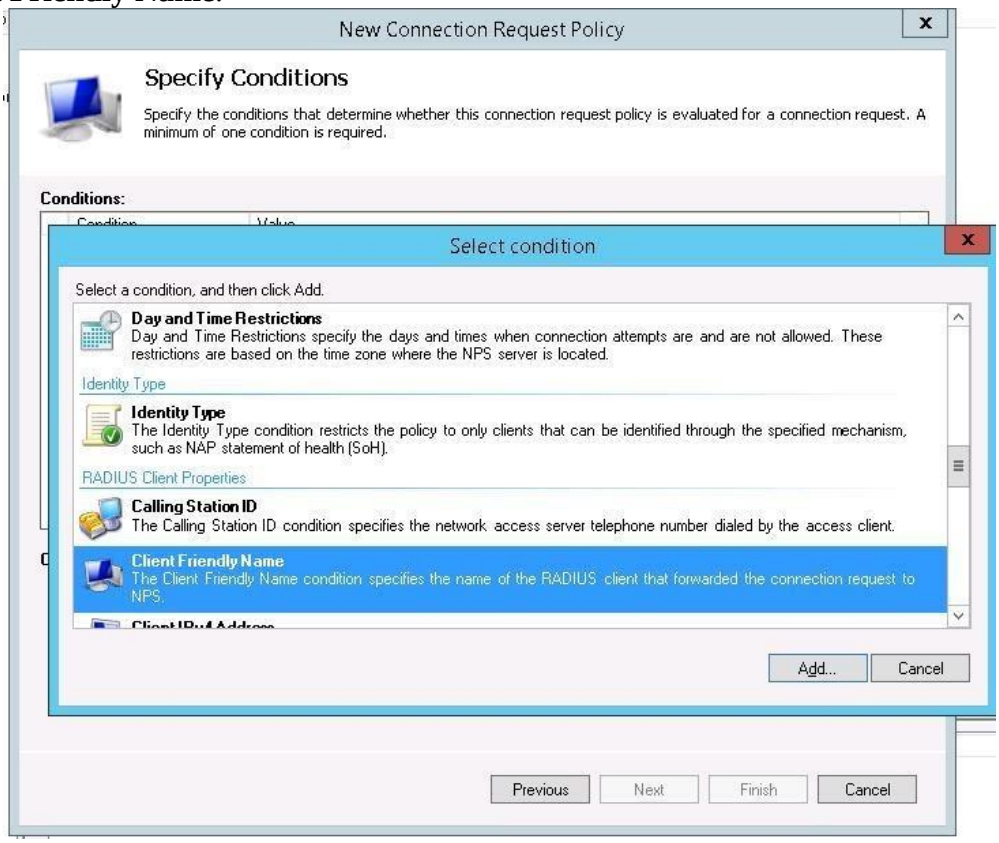


Рис. 107

В качестве значения укажем **sw-?**. Т.е. условие будет применяться для всех клиентов RADIUS, начинающийся с символов :"sw-". Жмем Next->Next-> Next, соглашаясь со всеми стандартными настройками.

Далее в разделе **Network Policies** создадим новую политику аутентификации. Укажите ее имя, например **Network Switch Auth Policy for Network Admins**. Создадим два условия: в первом условии **Windows Groups**, укажем доменную группу, члены которой могут аутентифицироваться (учетные записи сетевых администраторов в нашем примере включены в группу AD Network Admins) Второе условие **Authentication Type**, выбрав в качестве протокола аутентификации PAP.

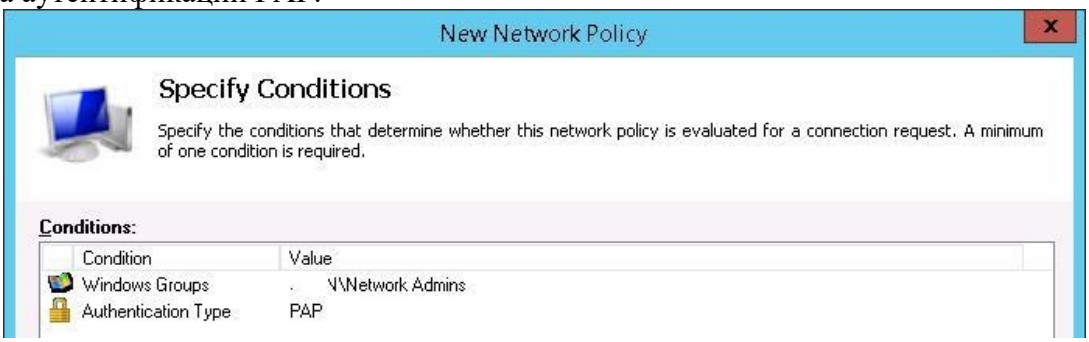


Рис. 108

Далее в окне Configure Authentication Methods снимаем галки со всех типов аутентификации, кроме **Unencrypted authentication (PAP. SPAP)**.

В окне **Configure Settings** изменим значение атрибута **Service-Type** на **Administrative**.

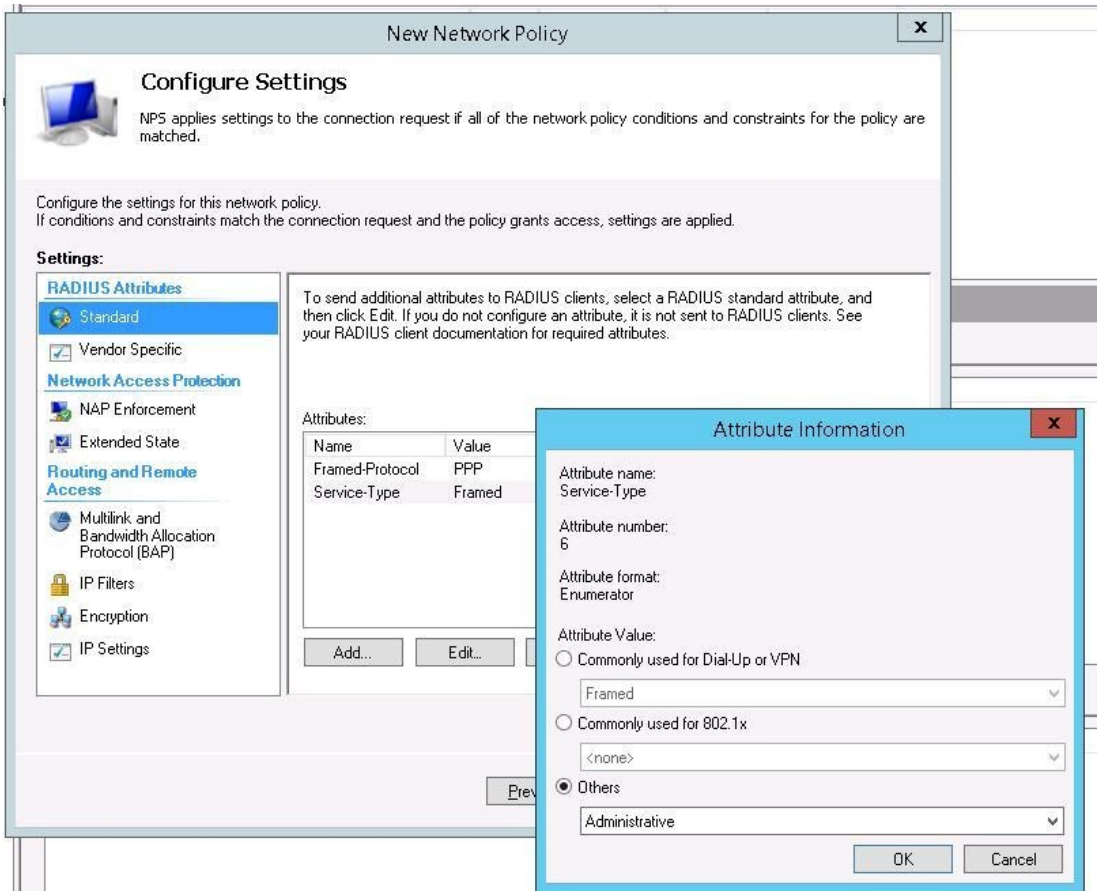


Рис. 109

В остальных случаях соглашаемся со стандартными настройками и завершаем работу с мастером.

И, напоследок, переместим новую политику на первое место в списке политик.

Рис. 110

3. Настройка сетевого оборудования для работы с сервером RADUIS

Осталось настроить наше сетевое оборудование для работы с сервером Radius. Подключимся к нашему коммутатору HP ProCurve Switch 5400 и внесем следующе изменение в его конфигурацию (измените ip адрес сервера Raduis и пароль на свои).

```
aaa authentication console enable radius local
aaa authentication telnet login radius local
aaa authentication telnet enable radius local
aaa authentication ssh login radius local

aaa authentication ssh enable radius local
aaa authentication login privilege-mode
radius-server key YOUR-SECRET-KEY
radius-server host 10.10.10.44 YOUR-SECRET-KEY auth-port 1645 acct-port 1646
radius-server host 10.10.10.44 auth-port 1645
```

```
radius-server host 10.10.10.44 acct-port 1646
```

Совет. Если в целях безопасности вы запретили подключаться к сетевому оборудованию через telnet, эти строки нужно удалить из конфига:

```
aaa authentication telnet login radius local
aaa authentication telnet enable radius local
```

Не закрывая консольное окно коммутатора (**это важно!**, иначе, если что-то пойдет не так, вы более не сможете подключиться к своему коммутатору), откройте вторую telnet-сессию. Должно появиться новое окно авторизации, в котором будет предложено указать имя и пароль учетной записи. Попробуйте указать данные своей учетной записи в AD (она должна входить в группу Network Admins). Если подключение установлено – вы все сделали правильно!

Рис. 111

Для коммутатора Cisco конфигурация, предполагающая использование доменных учетных записей для аутентификации и авторизации, может выглядеть так:

Примечание. В зависимости от модели сетевого оборудования Cisco и версии IOS конфигурация может несколько отличаться.

```
aaa new-model
radius-server host 10.10.10.44 auth-port 1645 acct-port 1646 key YOUR-SECRET-KEY
aaa authentication login default group radius local
aaa authorization exec default group radius local
ip radius source-interface Vlan421
line con 0
line vty 0 4
line vty 5 15
```

Примечание. В такой конфигурации для аутентификации сначала используется сервер RADIUS, а если он не доступен – локальная учетная запись.

Для Cisco ASA конфигурация будет выглядеть так:

```
aaa-server RADIUS protocol radius
aaa-server RADIUS host 10.10.10.44 key YOUR-SECRET-KEY
```

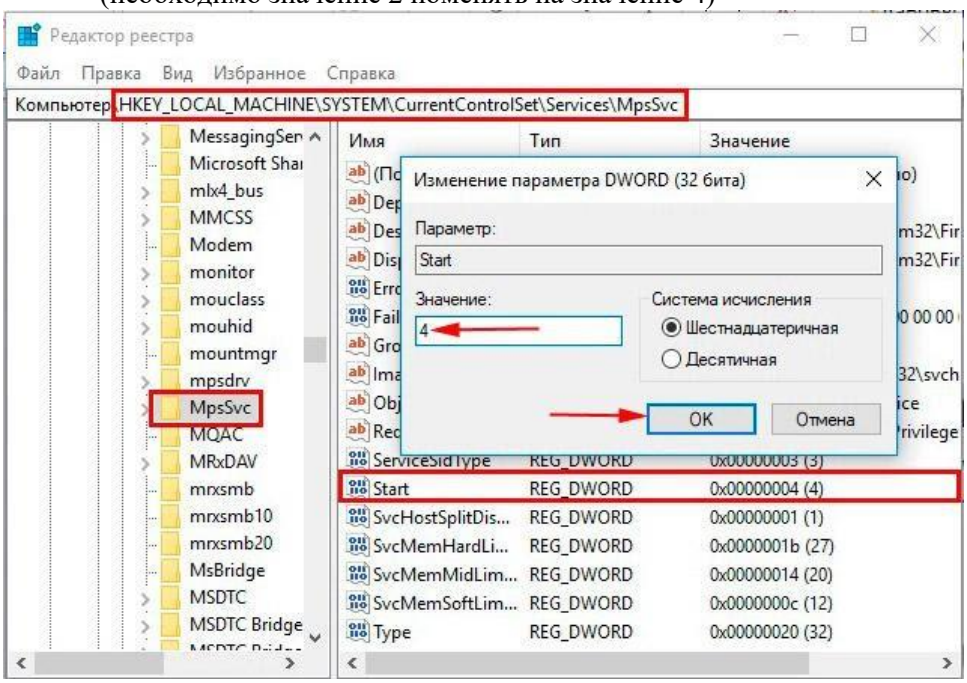
```
radius-common-pw YOUR-SECRET-KEY
aaa authentication telnet console RADIUS LOCAL
aaa authentication ssh console RADIUS LOCAL
aaa authentication http console RADIUS LOCAL
aaa authentication http console RADIUS LOCAL
```

Сделайте скриншоты (фотографии) процесса установки, настройки и устранения неполадок Сетевой политики и вставьте в отчёт.

2.10 Практическая работа№ 10
Выявление ошибок пользователей и программного обеспечения и принятие мер по их исправлению

Задание:

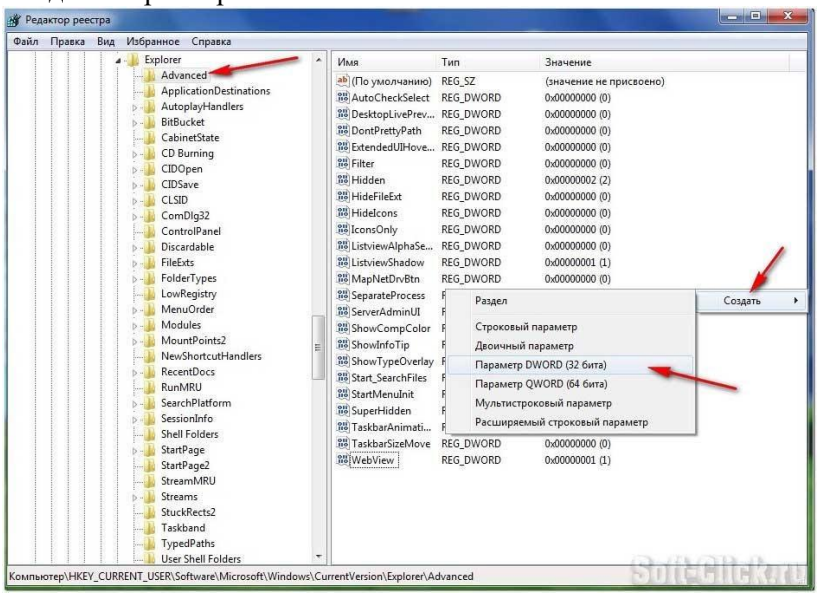
1. Зайти в Редактор реестра → установить курсор на Компьютер → в контекстном меню выбрать Экспортировать →на Рабочий стол Весь реестр с названием Reestr. Сохранить.
2. Зайти в Редактор реестра → HKEY_CURRENT_USER → Control Panel → Desktop.
   С помощью контекстного меню на ветке Desktop экспортировать эту ветку на Рабочий стол, сохранив под именем 1.
   Скриншот с файлом 1 на Рабочем столе в отчёт.
3. Откройте с помощью Блокнота файл реестра под именем 1.
   Измените значение ScreenSaveActive на 0.
   Скриншот с данной информацией в отчёт.
   Сохраните, закройте.
4. В Редакторе реестра в меню Файл → Импорт импортируйте файл 1 в реестр.
5. С помощь клавиши F3 запустите поиск по Реестру, например Брандмауэра. В отчёт вставить скриншот с результатами поиска.
6. С помощью пути, показанного на иллюстрации ниже отключите Брандмауэр:
   (необходимо значение 2 поменять на значение 4)



Скриншот в выполненным заданием в отчёт.

7. Отключение службы Update Orchestrator Service for Windows:
HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Services\UsoSvc
поменять значение параметра Start на 4

8. Для быстрого выключения компьютера изменить значение WaitToKillServiceTimeout на 1000. Пусть: HKEY_LOCAL_MACHINE\SYSTEM\ControlSet001\Control.
Скриншот в выполненным заданием в отчёт.
9. Отключение кэширования изображений: HKEY_CURRENT_USER\Software\Microsoft\Windows\CurrentVersion\Explorer\Advanced. Создаём параметр типа DWORD с именем DisableThumbailCache, значение параметра 1.



Скриншот в выполненным заданием в отчёт
10. Выгрузка из памяти неиспользуемых DLL: HKEY_LOCAL_MACHINE\SOFTWARE\Microsoft\Windows\CurrentVersion|Explorer. Создаём параметр типа DWORD с именем AlwaysUnloadDLL. Значение параместра 1.
Скриншот в выполненным заданием в отчёт
11. Очищение файла подкачки при выключении компьютера: HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Control\Session Manager\Memory Management. Измените значение параметра ClearPageFileShutdown на 1.
Скриншот в выполненным заданием в отчёт.
12. С помощью Интернета найти как запретить смену фона Рабочего стола с помощью реестра. Выполнить найденные действия.
Скриншот в выполненным заданием в отчёт
Задание по Оптимизации

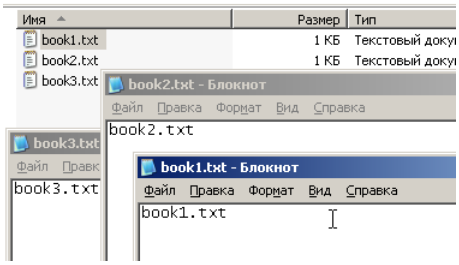
1. Вкладка «Быстродействие» в свойствах системы.
   Чтобы открыть эту вкладку делаем следующее:
   Нажимаем комбинацию клавиш Win+X и выбираем в появившемся окне пункт «Система».

2.11 Практическая работа№ 11
Обеспечение своевременного копирования, архивирования и резервирования данных

Задание:

Создания задания на выполнения архивации данных

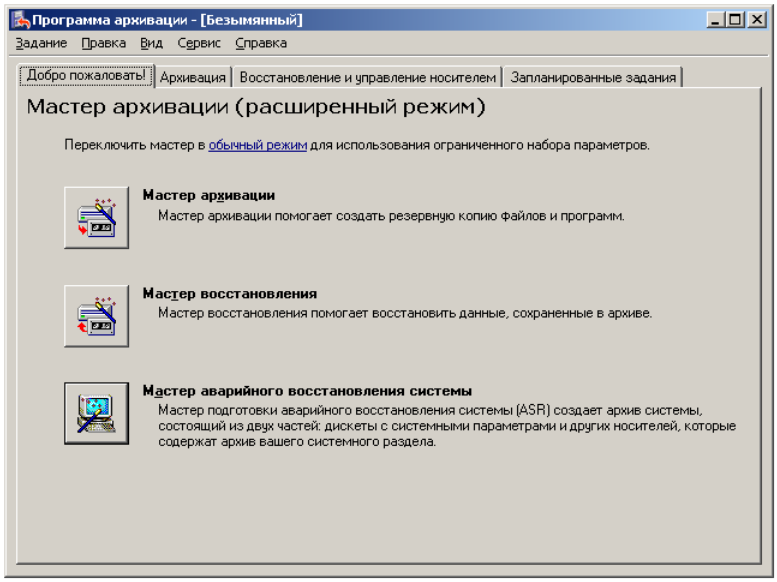
1. Создать на диске «С» Вашего сервера каталог *backup* и *restore*;
2. В папке library, созданной в одной из предыдущих работ создать 3 текстовых файла с наименованиями *book1.txt, book2.txt* и *book3.txt*. Файлы должны содержать свое наименование.



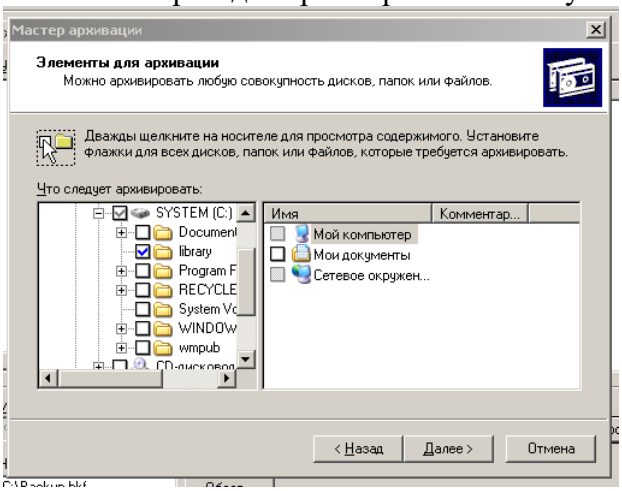
3. Запустить утилиту резервного копирования *ntbackup*.

Эту утилиту можно запустить из Главного меню системы (кнопка «Пуск» — «Все программы» — «Стандартные» — «Служебные» — «Архивация данных»), а можно запустить более быстро из командной строки (кнопка «Пуск» — «Выполнить» — «ntbackup» — кнопка «ОК»). При первом запуске утилиты рекомендуем убрать галочку у поля «Всегда запускать в режиме мастера».

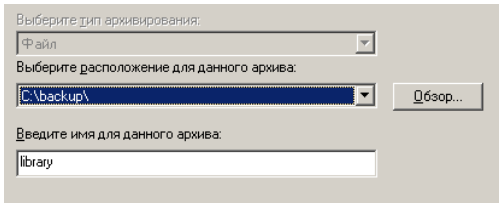
4. Запустить «Мастер архивации» (на закладке «Добро пожаловать» нажать кнопку «Мастер архивации».



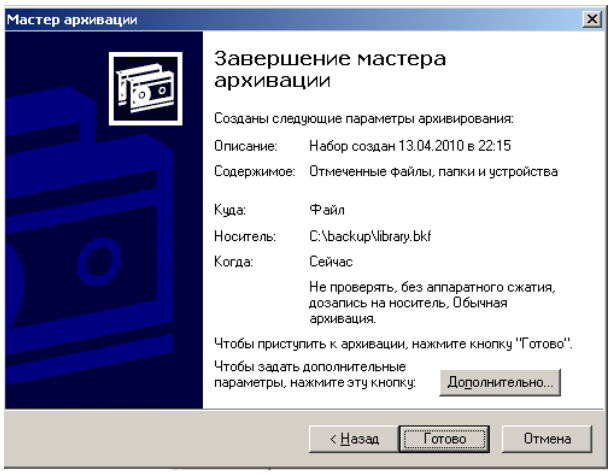
5. После запуска мастера нажмем кнопку «Далее» и выберем, что нам нужно архивировать, в данном примере — «Архивировать выбранные файлы, диски или сетевые данные»
6. Выберем для архивирования папку *library*.



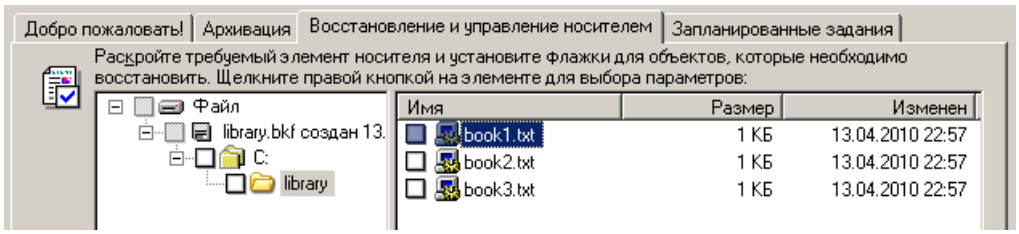
7. Выберем место для создания резервной копии, создадим файл с именем *library*, этому файлу автоматически будет назначено расширение «*.bkf*»



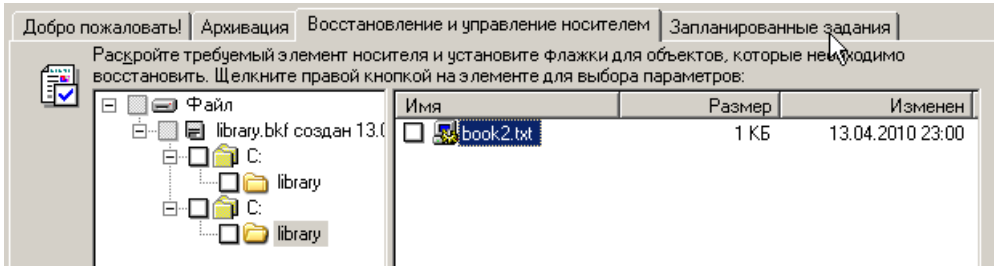
8. На данном этапе нажмем кнопку «Готово».



9. Проверяем полученный результат.



10. Вносим изменение в файл *book1.txt* и *book2.txt*, у файла *book1.txt* убираем атрибут «Файл готов для архивирования», а *book3.txt* - удаляем.
11. Запускаем снова процесс архивации, но на 8 этапе нажмем кнопку «Дополнительно», чтобы задать дополнительные параметры и выбираем тип архивации «Добавочный». Далее все пункты по умолчанию, но при этом не забывайте запоминать, что Вы делаете. Проверяем полученный результат. Почему он такой?



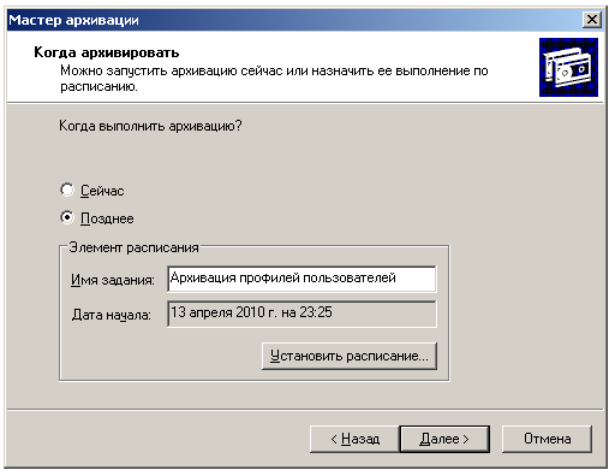
12. Восстановите файл *book3.txt*. Для этого выполните следующие действия:

- Запустим утилиту резервного копирования *ntbackup*.
- Перейдем на закладку "Восстановление и управление носителем".

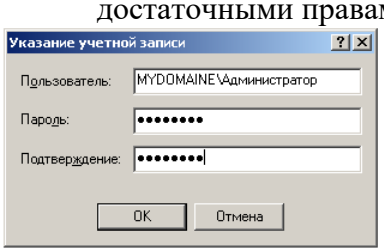
- После появления в списке архивных файлов нужного архива раскроем этот архив и выберем файлы для восстановления из резервной копии. При этом мы можем восстановить файлы в то место, где они были ранее ("Исходное размещение") или выбрать иной путь для их сохранения ("Альтернативное размещение"). Выберите папку *restore*.
- После определения всех параметров восстановления нажмем кнопку "Восстановить", утерянные данные будут восстановлены.

13. Создайте задания на выполнения архивации данных для папки *profiles*, используя выбор дополнительных возможностей:

- Выбираем тип архивирования (выберем «Обычный»).
- Ничего не меняем на странице «Способы архивации».
- На странице «Параметры архивации» можно выбрать замену существующих архивов или добавление архива (если файл с архивной копией уже существует).



14. На странице «Когда архивировать» задайте расписание для автоматического создания резервной копии — выберите вариант «Позднее» и задайте расписание архивирования, чтобы архивирование происходило по всем рабочим дням недели. Время начала установите, исходя из текущего времени системы + пять минут.
15. Нажмите далее. Система запросит имя и пароль пользователя, с чьими полномочиями будет выполняться задание архивирования. Рекомендуем для выполнения заданий резервного копирования создать специальные учетные записи, обладающие достаточными правами (как минимум члены группы «Операторы архива»).



16. Нажмем кнопку «Готово», задание будет создано, и оно появится в списке «Назначенных заданий». Теперь оно будет выполняться регулярно в соответствии с расписанием.
17. Завершите сеанс администратора, ожидайте до завершения задания. После проверьте результат.

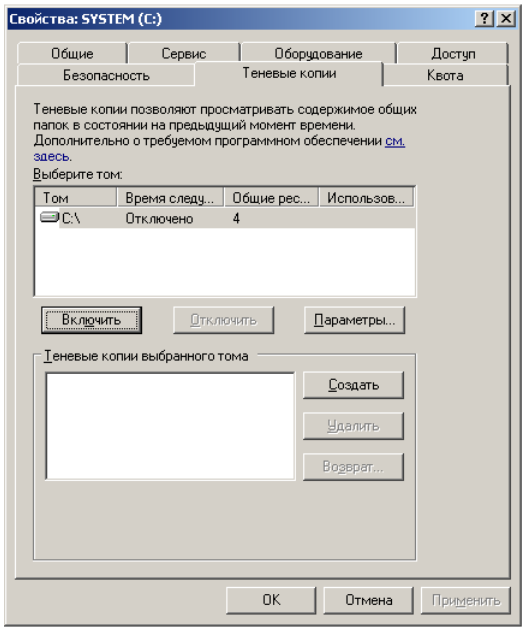
Теневые копии

Эта технология, реализованная в Windows, позволяет архивировать открытые файлы с помощью создания «снимка» файловых ресурсов. По умолчанию теневые копии создаются на том же томе, где хранятся сетевые папки, поэтому они не смогут стать серьезной защи-

той от аппаратных аварий (например, выход из строя диска, на котором размещены эти данные). Можно настроить создание теневых копий на другом томе, что повысит уровень защиты. Теневые копии позволяют восстанавливать данные, ошибочно удаленные или модифицированные пользователями. При этом пользователи могут восстанавливать данные сами, без участия системного администратора. Теневые копии создаются только на томах с файловой системой NTFS.

Рассмотрим пример создания и использования теневых копий тома.

1. Создадим в сетевой папке на сервере файл *document.txt*, содержащий текст: «11111».
2. Откроем Свойства какого-либо тома и перейдем на закладку «Теневые копии». По умолчанию создание теневых копий для всех томов отключено.
3. Включим создание теневых копий для тома «С». При этом автоматически создастся первая теневая копия. В этом окне также можно вручную создать теневую копию данного тома в любой момент времени.

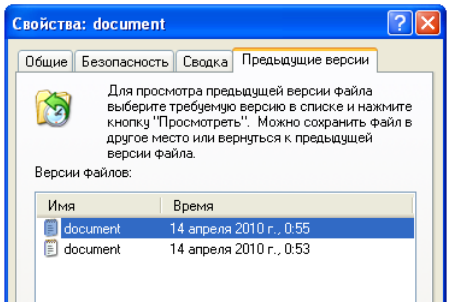


4. Настроим параметры теневого копирования. Для хранения теневых копий на томе требуется не менее 100 МБ дискового пространства, на каждом томе создается максимум 64 копии.
5. Настройте размер пространства для хранения копий в размере 200МБ и расписание создания теневых копий — дважды в день в 14-00 и 24-00.
6. На клиентской машине откройте файл *document.txt* и добавьте новую строку «22222».
7. На сервере вручную создайте еще одну теневую копию данного тома.
8. На клиентской машине откройте файл *document.txt* и добавьте новую строку «33333».

***Замечание**. Теневые копии создаются не для всех файлов тома, а только для тех, которые размещены в папках, выставленных в сеть для общего доступа.*

Использование теневых копий. После создания теневых копий пользователю становятся доступны Предыдущие версии файлов. Для использования этих возможностей нужна клиентская часть для доступа к теневым копиям. Дистрибутив клиента теневых копий хранится на сервере в папке «%SystemRoot%\system32\clients\twclient», в файле twcli32.msi. При установленном клиенте в свойствах файла, открываемого из сетевых папок, становит-

ся доступна закладка «Предыдущие версии». Проверьте, доступна ли данная закладка в Вашей клиентской системе, если нет, то установите необходимое ПО.



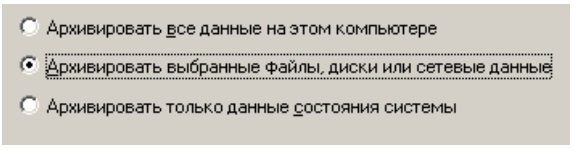
Пользователь теперь может просмотреть предыдущие копии, скопировать их в другой файл или восстановить содержимое файла в одно из предыдущих состояний. Закладка «Предыдущие версии» доступна в Свойствах не только конкретного файла, но и всей сетевой папки. Поэтому можно восстановить не только измененные файлы, но и ошибочно удаленные.

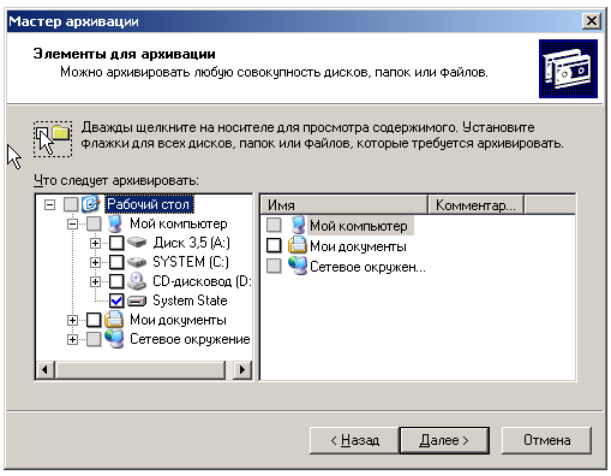
Архивирование и восстановление состояния системы

Большую часть работ по резервному копированию составляют задания на копирование бизнес-информации. Но имеется также возможность создания резервных копий для восстановления функционирования самой операционной системы. Есть два варианта архивирования системных данных — архивирование состояния системы (*System State*) и создания набора для автоматического восстановления системы после аварии (*Automated System Recovery*).

1. **Архивирование и восстановление состояния системы**

Для создания резервной копии состояния системы необходимо в утилите резервного копирования *ntbackup* при создании задания на архивирования отметить галочкой пункт *System State*:





При этом будут архивироваться следующие данные:

- системный реестр;
- база данных зарегистрированных классов объектов (*Class Registration*);
- системные загрузочные файлы;
- база данных служб сертификатов (только на серверах, на которых установлена служба сертификатов);
- база данных *Active Directory* и папка *SYSVOL* (на контроллерах доменов).

Для архивирования состояния системы, а также для последующего восстановления, обязательно нужны права администратора данного компьютера. Восстановление *Active Directory* необходимо выполнять только при загрузке системы в режиме восстановления служб каталогов (запуск меню выбора режимы загрузки операционной системы выбираются в начальный момент загрузки нажатием клавиши F8).

2. **Автоматическое аварийное восстановление системы**

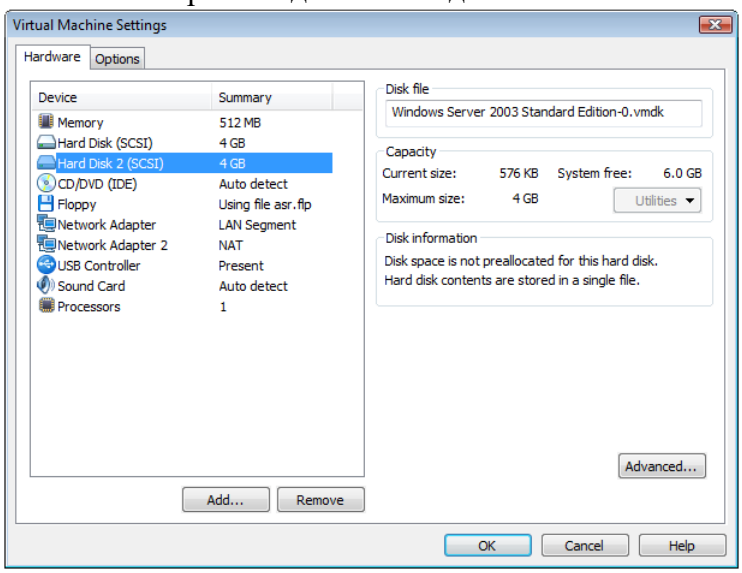
В отличие от резервного копирования состояния системы, при котором сохраняется только часть файлов операционной системы, резервное копирования для автоматического аварийного восстановления системы (*ASR, Automated System Recover*) архивирует больший объем информации — практически весь том, на котором установлена операционная система. И процедура восстановления системы становится более сложной.

3. **Создание ASR-копии**

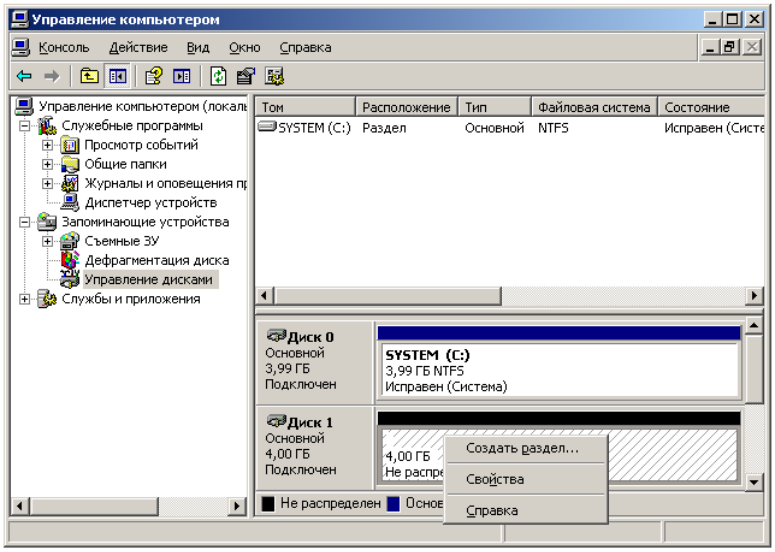
На данном этапе потребуется носитель для создания резервной копии системного тома (порядка нескольких гигабайт), причем в случае восстановления системы этот носитель должен быть доступен мастеру установки операционной системы (т.е. это либо ленточный накопитель с драйверами для контроллера и накопителя, либо дисковый накопитель с соответствующими драйверами), а также чистая отформатированная дискета для сохранения информации о конфигурации резервной копии.

1. Выберем вариант хранения данных на дополнительном дисковом накопителе. Для этого выполним следующие действия:

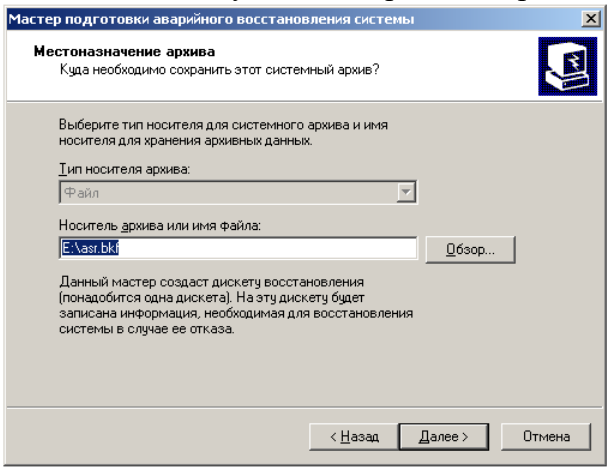
- Завершим работу нашего сервера;
- В настройках данной ОС добавим новый SCSI-винчестер объемом 4Gb;



- Запустим ОС.
- Нажмем правой клавишей мыши на «Мой компьютер» и вызываем «Управление»;
- В управлении дисками инициализируем новый диск;
- Создаем на нем основной NTFS раздел по всему объему диска.

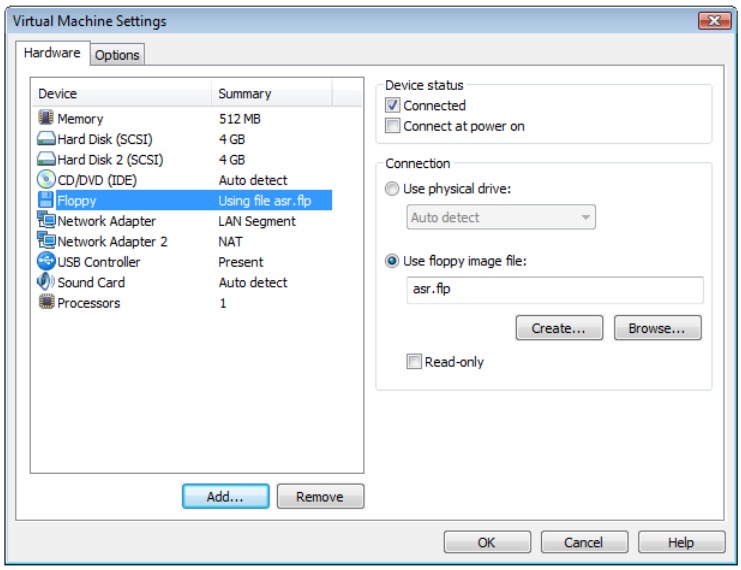


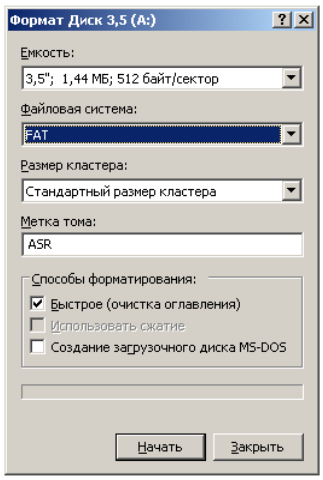
2. Запустим утилиту резервного копирования *ntbackup*.
3. Запустим «Мастер аварийного восстановления системы».
4. Укажем путь для сохранения архива.



5. Нажмем кнопку «Готово». Утилита резервного копирования начнет создание резервной ASR-копии, в нужный момент будет сделан запрос вставить чистую дискету.

Работа с дисководом в WMware имеет определенную специфику. Будем использовать виртуальную дискету. Для этого в свойствах ОС сервера в WMware выберем дискету, выберем «Использовать образ дискеты» и нажмем «Создать». Перед использованием дискеты отформатируйте ее.





После записи конфигурации резервной копии утилита попросит пометить дискету соответствующей информацией (название резервной копии и дата создания).

4. **Восстановление системы с помощью ASR-копии**

1. Подготовим все необходимое для аварийного восстановления системы: установочный CD с дистрибутивом операционной системы, носитель с резервной копией, дискету с конфигурацией ASR-копии.

2. Запустим процесс установки операционной системы с загрузочного компакт-диска для этого в BIOSe виртуальной машины сервера установим загрузку с CD;

3. На первой странице мастера установки системы (после появления синего экрана) нажать клавишу F2 для запуска процесса аварийного восстановления.

4. Далее мастер установки системы выполнит новую установку системы с форматированием системного тома.

5. После выполнения установки операционной системы автоматически запустится утилита резервного копирования, и система попросит вас указать путь к резервной копии для аварийного восстановления и вставить дискету с конфигурацией ASR-копии. Будет выполнено восстановление системы из аварийной резервной копии.
6. После завершения процесса восстановления будет воссоздан работоспособный сервер в той конфигурации, которая была до аварии (при условии, конечно, что, кроме самой системы, будут также восстановлены и данные, необходимые для работы сервера).

7. В BIOSe виртуальной машины сервера установим загрузку с HDD;

Корпорация Microsoft рекомендует использовать данный метод восстановления для серверов, выполняющих особые функции, которые трудно восстановить простой переустановкой и восстановлением данных. Если сервер не исполняет какие-либо особые роли, то Microsoft рекомендует на таких серверах архивировать только данные, а в случае аварии заново переустановить сервер, снова включить его в домен и восстановить данные из резервных копий.

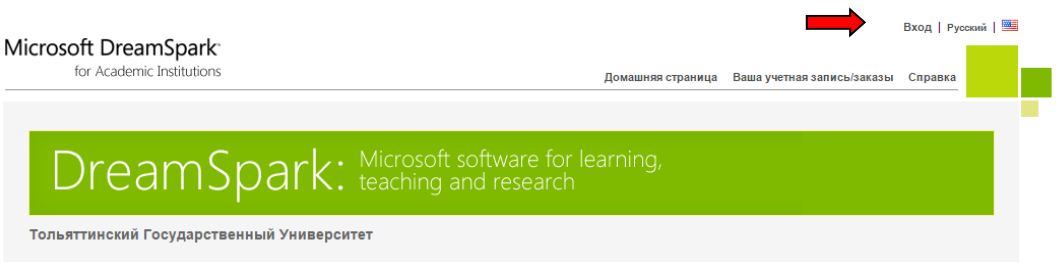
2.12 Практическая работа№ 12
Планирование и реализация стратегии виртуализации серверов

Задание:

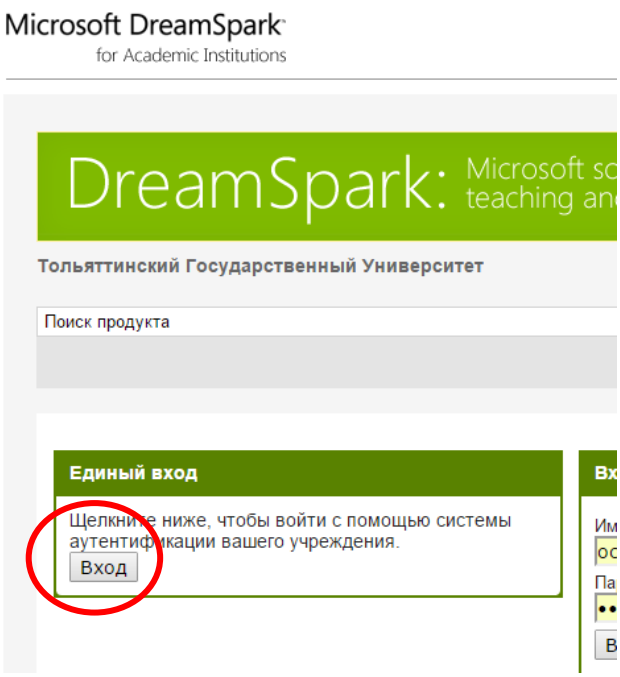
Установите и запустите Oracle VM Virtual Box.
Создайте виртуальную Windows ОС. Для скачивания ISO образа следуйте инструкции, приведенной в конце задания.
Создайте виртуальную Lunux подобную ОС. Для скачивания ISO образа можно использовать следующие ссылки:
http://ubuntu.ru/get
http://www.ubuntu.com/download/desktop
http://www.kubuntu.org/getkubuntu/
http://www.oracle.com/ru/technologies/linux/overview/index.html

Настройте виртуальные машины для выхода в Интернет
Подготовьте отчет по проделанной работе. Отчет должен содержать скриншоты и описание процесса установки и настройки виртуальных ОС.
Представьте отчет о практической работе.

Содержание отчета
Титульный лист.
Название и цель работы.
Результаты выполнения работы
Заключение.
Порядок загрузки ПО от Microsoft
В соответствии с соглашением между ТГУ и Microsoft студенты и преподаватели имеют право бесплатно скачивать ПО для учебных целей. Для скачивания необходимо зайти на сайт DreamSpark по адресу:
http://onthehub.com/
Далее по ссылке *Find your school store for the best software prices*
http://onthehub.com/search/
выбрать *College & University*
http://onthehub.com/search/higher-ed/
Далее выберите страну *Russian Federation* и вуз *Russian State University for Togliatti.*
Перейдите по ссылке: Go to your webStore to save.
В открывшемся окне выберите войти:



Далее выберите *Единый вход*



Введите логин и пароль от образовательного портала ТГУ.
Теперь, следуя инструкциям, Вы можете загружать ПО.

2.13 Практическая работа№ 13
Планирование и реализация сетевой инфраструктуры и систем хранения данных для виртуализации

Задание

В отчете укажите ответы на вопросы:

1. Задачи, которые решает виртуализация СХД
2. В чем заключается Экономия на расширении инфраструктуры
3. Упрощенный перенос данных и зеркальное копирование при виртуализации достигается за счет чего?
4. Привидите пример успешного внедрения виртуальной инфраструктуры (данные взять из открытых источников)

2.14 Практическая работа№ 14
Планирование и развертывание виртуальных машин

Задание:

4.1 Возможности Virtual Box.

Программа Virtual Box позволяет эмулировать x86-совместимый ПК, на который можно установить большое число гостевых ОС, среди которых все семейство Windows, начиная с Windows 3.x и заканчивая Vista, DOS, Linux на основе ядра версий 2.4 и 2.6 и OpenBSD. Сама же виртуальная машина может предоставлять системе доступ в сеть, позволять работать с периферийными устройствами. Помимо прочего стоит отметить достаточно удобные инструменты для обмена файлами. Есть возможность выбора языка ин-

терфейса (поддерживается и русскоязычный интерфейс).

4.2 Запуск Virtual Box.

Запуск программы осуществляется выбором пунктов меню «Пуск»: Пуск — Все программы — Oracle VM VirtualBox — Oracle VM VirtualBox. Программа требует бесплатную регистрацию, которая заключается в простом вводе своего имени и E-mail, ее можно отменить.

4.3 Создание виртуальной машины.

Главное меню программы (рис. 1) состоит из трех пунктов:

- Файл
- Машина
- Справка

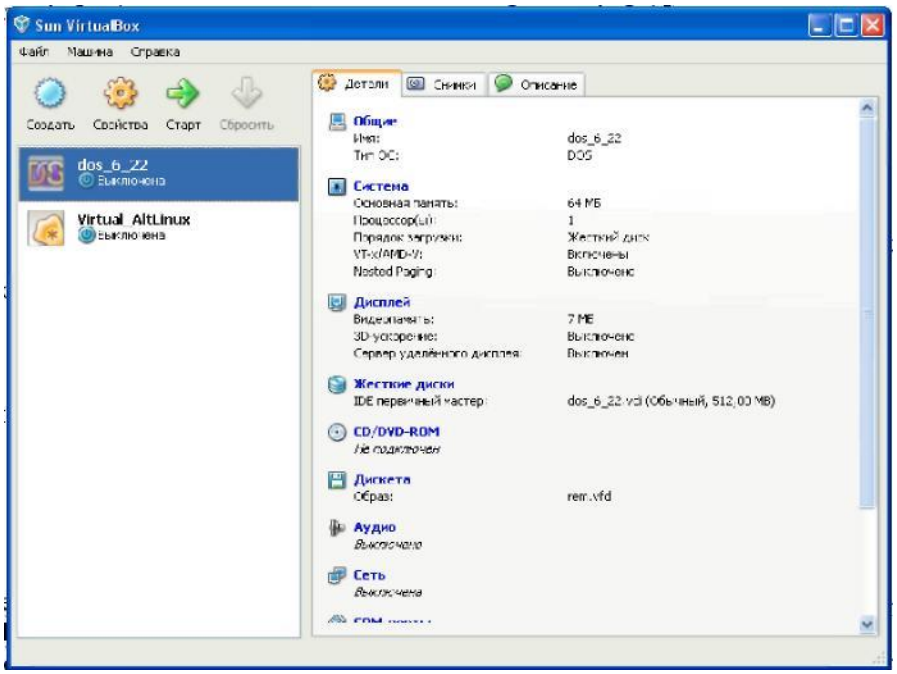


Рис. 1. Главное окно программы

Меню «Файл», в свою очередь, делится на «Менеджер виртуальных дисков», «Настройки» и «Выход».

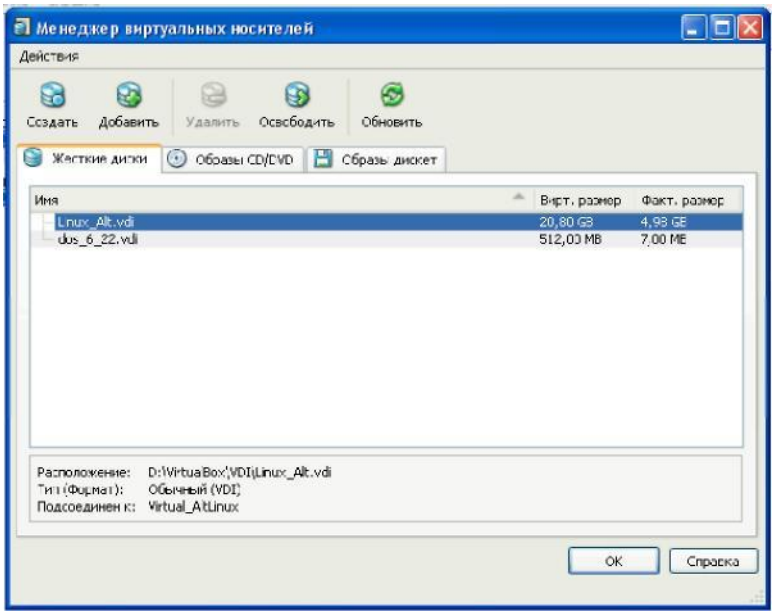


Рис. 2. Менеджер виртуальных дисков

Рассмотрим их поподробнее. Первый пункт - это менеджер виртуальных дисков (рис. 2), при помощи которого можно создавать, добавлять, освобождать или удалять «виртуалки». Также можно подключать образы CD/DVD для последующей установки системы из образа.

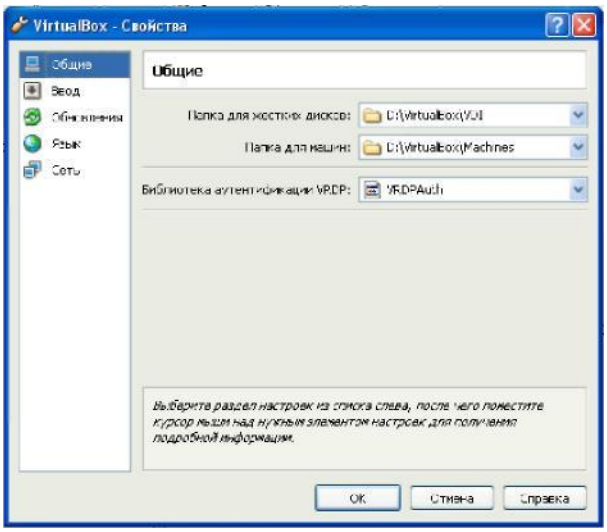


Рис. 3. Настройки

Второй пункт - это «Настройки» (рис. 3). Большого количества настроек здесь нет, но те, которые присутствуют, помогают настроить «индивидуальность» программы.

Также при помощи пункта **«Настройки»** можно выбрать так называемую хост-клавишу, ту клавишу, при нажатии на которую можно активизировать курсор Windows и впоследствии переключаться на другие запущенные программы (или просто открытые окна) в среде Windows. Еще в пункте **«Настройки»** можно выбрать язык программы.

Следующий пункт в Главном меню программы - это **«Машина»**. При помощи данного пункта можно управлять и следить за состоянием вашей виртуальной машины. Также при помощи данного пункта можно создать новую «виртуалку» (рис. 4).

Создание новой «виртуалки» происходит в режиме мастера. То есть делится на несколько регламентированных шагов при помощи которых и создается виртуальная машина.

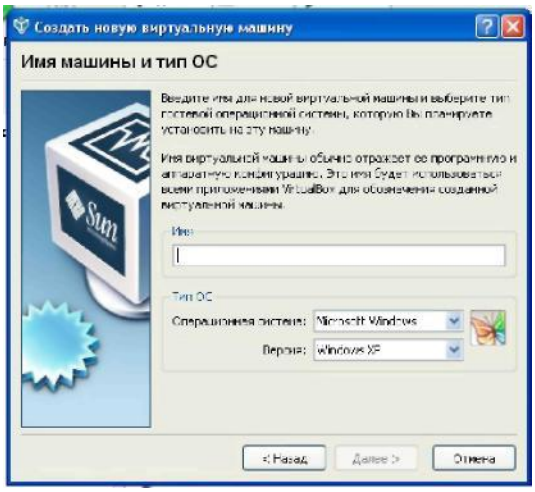


Рис. 4. Мастер создания новой виртуальной машины

Рассмотрим эти пункты:

- Выбор названия виртуальной машины и типа ОС;
- Выбор основной памяти для выделения виртуальной машине;
- Создание/подключение виртуального диска;
- Диалог с указанием успеха. После создания виртуального раздела общий вид программы меняется и становится

более функциональным.

В правой части окна можно посмотреть текущее состояние машины, а также наличие подключения CD/DVD привода, Аудио, Сети, USB. В тот момент, когда виртуальная машина неактивна, можно выбрать дополнительные параметры и свойства, нажав на кнопочку вверху окна по имени **«Свойства»**. Например, выбрать порядок загрузки, тип буфера (двунаправленный - из системы в виртуальную машину, или только внутри «виртуалки»).

Можно подключить звук из основной системы, который будет эмулироваться с помощью Windows Direct Sound. Можно включить контроллер USB.

Во время работы системы у каждого пользователя есть возможность делать так называемые снимки состояния системы. С возможным последующим откатом к раннему состоянию системы. Если во время работы с виртуальной системой у вас возникла необходимость срочно (или не совсем срочно) выйти из нее, то по нажатии на хост-клавишу и затем по нажатии комбинации **ALT+F4** можно выбрать один из пунктов завершения работы системы:

- Сохранить состояние машины;
- Послать сигнал завершения (равносильно завершению работы с последующим выключением компьютера в среде Windows);
- Выключить машину.

Удаление «виртуалки» не вызывает никаких трудностей. После выделения нужной

виртуальной машины нажимаете на кнопку **«Удалить»**, и вам задают следующий вопрос: хотите ли вы удалить текущую виртуальную машину? Если вы уверены, то нажимайте **«Удалить»**.

5. **Контрольные задания**

1. Создайте на папки D:\HardDisks\<Ваш логин> и D:\Machines\<Ваш логин>.
2. Запустите VirtualBox.
3. Виберите пунк меню Файл — Настройки.
4. На вкладке Свойства, в разделе общие измените: а) Папка для жестких дисков — D:\HardDisks\<Ваш логин>; б) Папка для машин — D:\Machines\<Ваш логин>. Сохраните изменения.
5. Создайте новую виртуальную машину, со следующими параметрами: Имя: <Ваш логин>_lr1; Операционная система: Other; Версия: DOS; Память: 128 Мб; Загрузочный диск: Первичный (мастер), Создать новый жесткий диск; Динамически расширяющийся образ; Размер 2 Гб.
6. Измените параметры виртуального компьютера, на вкладке Детали. Порядок загрузки: Дискета, Жесткий диск; Аудио: Выключено, Сеть: Выключена, USB: Выключен.

2.15 Практическая работа№ 15
Планирование и реализация решения по администрированию виртуализации

Задание:

1. Выполните миграиию физического сервеа Linux в вертуальную среду
2. Выполните миграиию физического сервеа Windows в вертуальную среду, при помощи disk2vhd
3. Для созданных виртуальных сред настройте резервное копирование
4. Выполните оптимизацию виртуальных жестких дисков.

Сделайте скриншоты (фотографии) процесса установки, настройки и вставьте в отчёт.

2.16 Практическая работа№ 16
Установка сервера Debian. Настройка web-сервера в ОС Debian

Задание 1:
Запускаем VirtualBox и создаем виртуальную машину с именем deb-serv_1

1. **Установка Debian.**

Выберите "Graphical install" курсорными клавишами и нажмите "Enter" для запуска установщика.

Рис. 218

Язык мастера установки и раскладка клавиатуры
Из предлагаемого списка выберите язык, который будет использоваться установщиком Debian для отображения инструкций. Для перехода к следующему шагу мастера установки ОС щелкните по кнопке «Continue».

Рис. 219

Определите местоположение, которое станет использоваться мастером установки операционной системы для определения часового пояса. Нажмите «Продолжить» и в дальнейшем щелкайте по этой кнопке для перехода к следующему шагу установки системы.

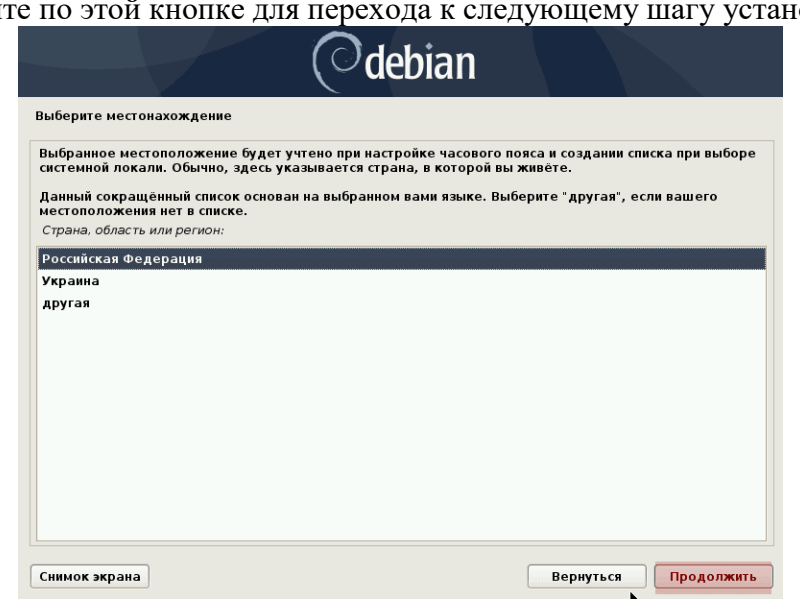


Рис. 220

Выберите клавиатурную раскладку.

Рис. 221

Из перечня выберите клавиатурную комбинацию или клавишу, с помощью которой вы станете переключаться между раскладками клавиатуры. Удобными считаются:

1. Правая клавиша «Alt».
2. Сочетание «Alt и Shift».

При выборе «Alt и Shift», сочетание не сможет использоваться в программах для других задач.

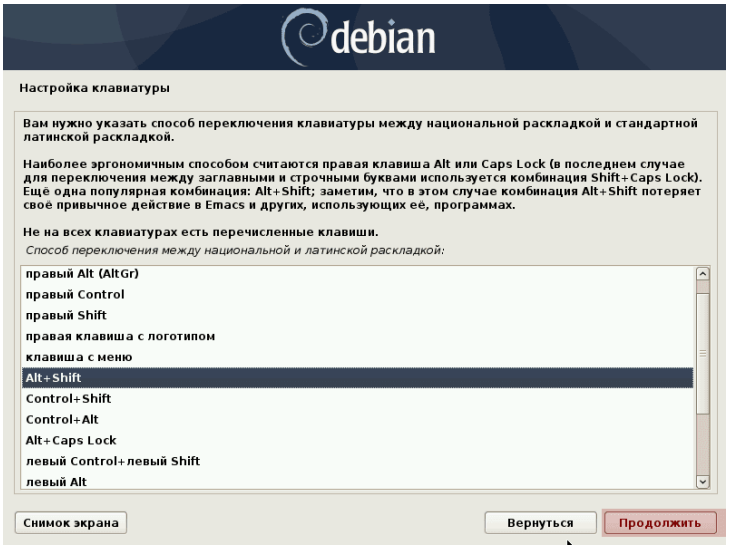


Рис. 222

Параметры установки
Стартует загрузка дополнительных компонентов. Дождитесь ее завершения.

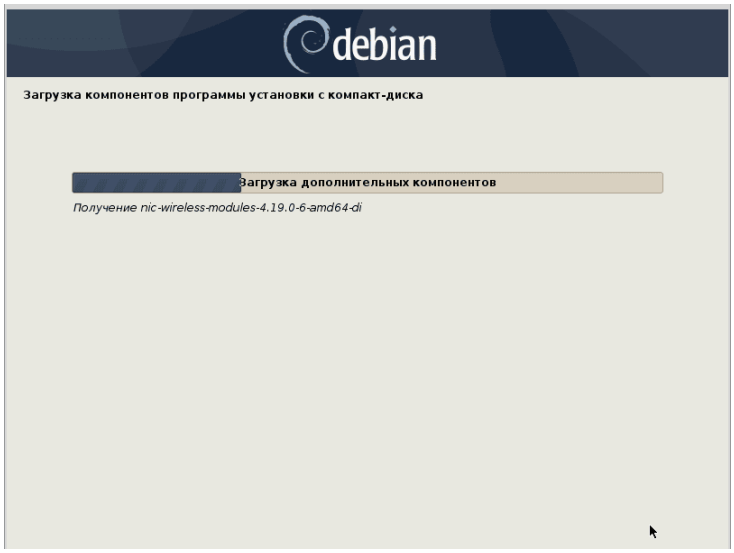


Рис. 223

Чтобы ПК мог быть идентифицирован в сети, укажите имя пользователя, состоящее из одного слова, введенного буквами латинского алфавита. В домашних условиях — любое удобное вам. На работе — определяется администратором сети.

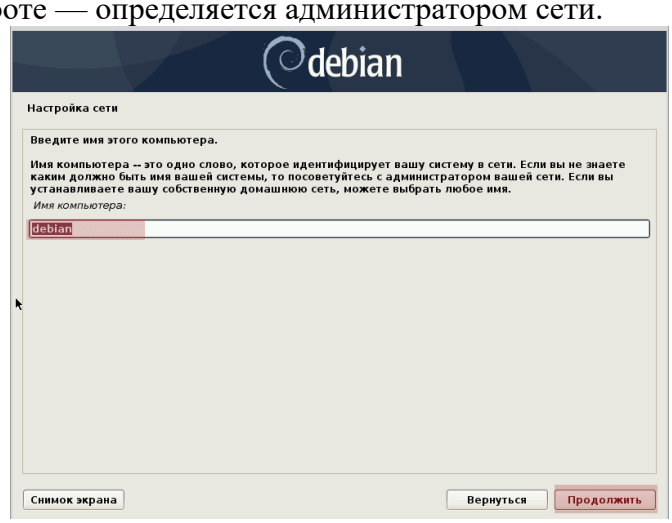


Рис. 224

Введите имя домена — часть интернет-адреса после имени пользователя. Необходимо, чтобы оно было одинаковым для всех домашних устройств. При настройке домашней сети — произвольное.

Рис. 225

Создайте пароль root:

1. Предназначен для задач администрирования системы.
2. Может включать в себя знаки препинания, цифры и латинские буквы.
3. Необходимо периодически менять.
4. Поле нельзя оставить пустым.

Повторите его в дополнительном поле.

Рис. 226

Дайте имя пользовательской учетной записи:

1. Используется вместо учетной записи root для действий, не связанных с администрированием.
2. Указывается в поле «От кого» отправляемых писем.
3. Используется всеми программами, которым необходимо реальное имя пользователя ПК.

Рис. 227

Укажите имя пользователя, под которым будете известны системе.

Рис. 228

Придумайте пароль. Может состоять из латинских букв, знаков препинания и цифр. Подтвердите его повторным вводом в дополнительное поле.

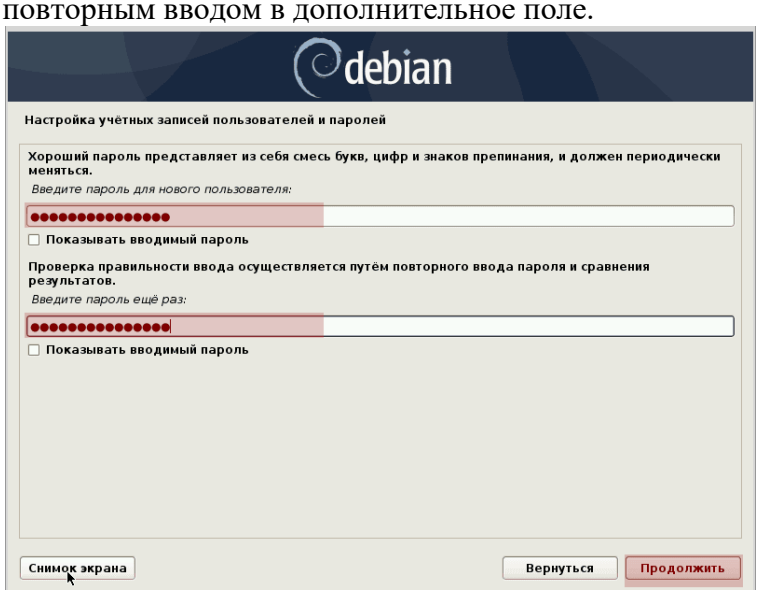


Рис. 229

Выберите часовой пояс из списка.

Рис. 230

Выберите пункт «Авто — использовать весь диск» для разметки диска, на который будет установлена ОС Debian. Все данные будут удалены с накопителя. Убедитесь, что важные файлы сохранены на дополнительных носителях.

Опытным пользователям предлагаются другие варианты разметки диска. Используйте их, если знаете, какого результата хотите достичь.

Рис. 231

Подтвердите внесение изменений.

Рис. 232

Подтвердите, что все файлы будут размещаться в одном разделе.
Предусмотрены два других подхода с созданием отдельных разделов для каталогов:

1. /home
2. /home, /var и /tmp

Рис. 233

Если вы не планируете делать другие настройки, оставьте предлагаемый по умолчанию пункт «Закончить разметку и записать изменения на диск».

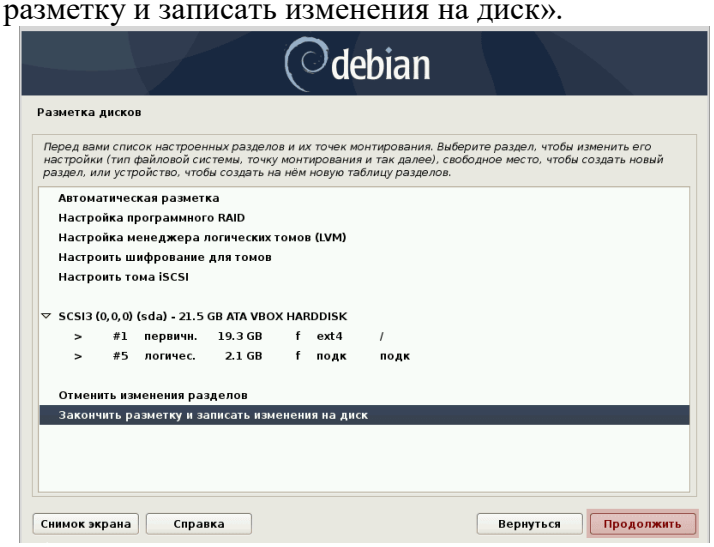


Рис. 234

На экране отобразится перечень изменений, которые будут записаны на диски. Вы можете выбрать:

1. «Нет» и вернуться к ручной разметке.
2. «Да» и продолжить установку системы.

Рассматриваем второй вариант.

Рис. 235

Установка ОС, интерфейса и ПО

Дождитесь завершения установки базовой системы.

Рис. 236

Согласитесь на использование зеркала архива. Позволит настроить графическое окружение рабочего стола и устанавливать дополнительное ПО.
Помните:

1. Необходимо соединение с интернетом.
2. Используется трафик согласно тарифам вашего провайдера (оператора связи).

Рис. 237

Из списка выберите зеркало архива Debian в ближайшей к вам сети.

Рис. 238

«deb.debian.org» — оптимальный выбор в случаях, когда нет точного знания о том, с каким зеркалом связь лучше.

Рис. 239

Укажите HTTP-прокси, если необходимо. Если такой необходимости нет, оставьте поле пустым и перейдите к следующему шагу.

Рис. 240

Дождитесь завершения автоматической настройки менеджера пакетов…

Рис. 241

…а также выбора и установки ПО.

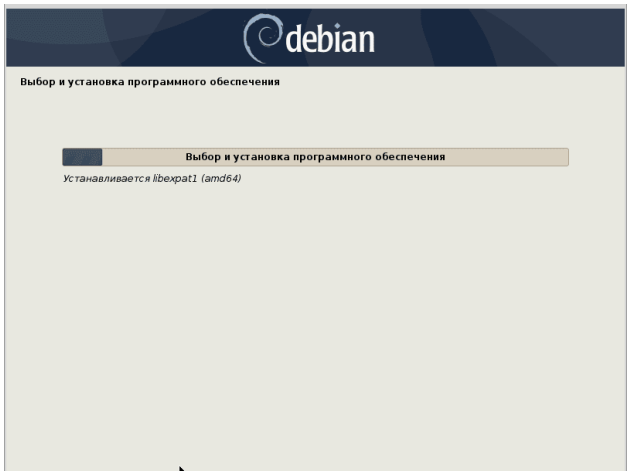


Рис. 242

Определите, позволите ли вы системе отправлять разработчикам данные о наиболее часто используемых пакетах. На основании этой информации определяется, какие пакеты добавляются на первый CD дистрибутива. Принимайте решение, внимательно ознакомившись с информацией, выведенной на экран в этом шаге установки.

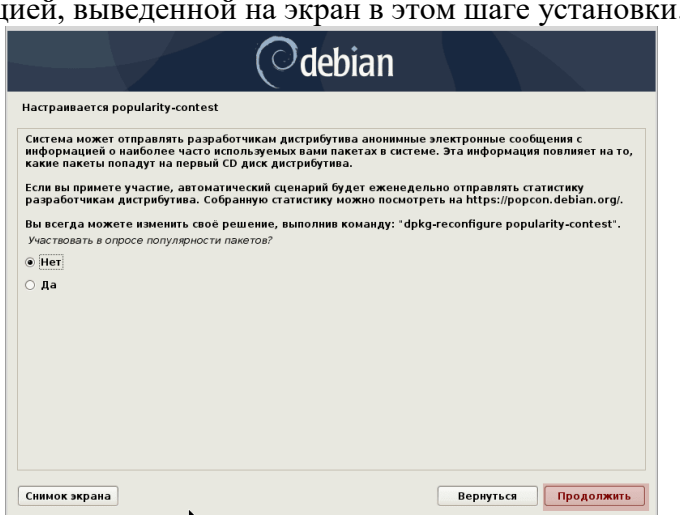


Рис. 243

Поставьте «птички» возле ПО, которое будет установлено в дополнение к базовой системе. Оставляем «птички» напротив SSH-сервер и Стандартные системные утилиты.

Рис. 244

Автоматическая стадия: выбор и установка программного обеспечения. Никаких действий производить не нужно. Дождитесь завершения.

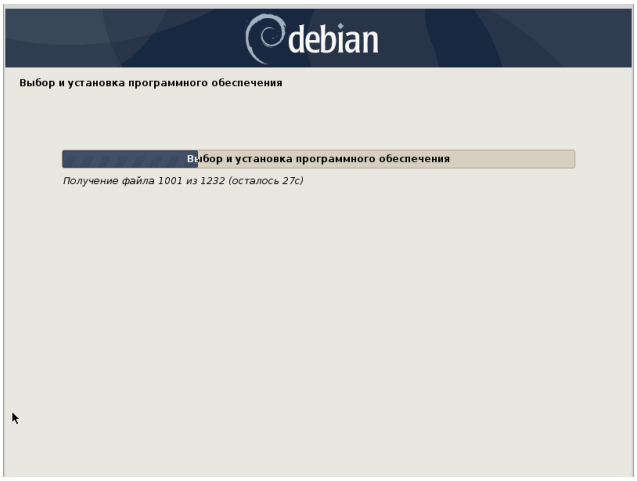


Рис. 245

Согласитесь на установку системного загрузчика GRUB. Рассматриваю этот вариант, предполагая, что Debian будет единственной ОС компьютера. Если на ПК установлена другая система, ее не получится использовать до тех пор, пока GRUB не будет настроен для ее загрузки.

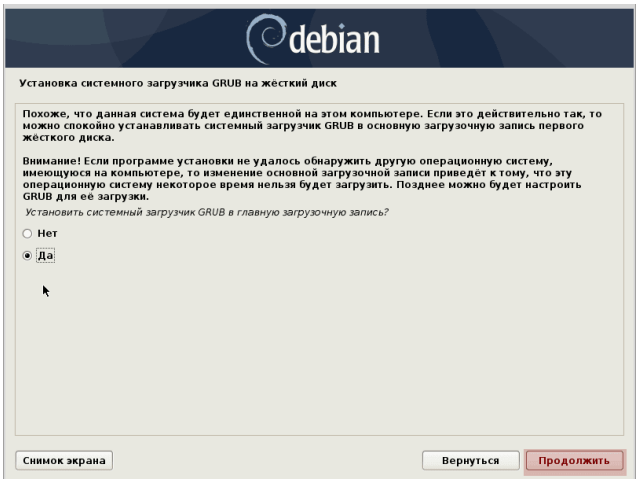


Рис. 246

Подтвердите установку системного загрузчика на жесткий диск ПК.

Рис. 247

Автоматическая установка загрузчика на жесткий диск.

Рис. 248

Перезагружаем сервер. Установка завершена, он полностью готов к работе.

Сделайте скриншоты (фотографии) процесса установки сервера Debian и вставьте в отчёт.

Задание 2:

1. Установка Nginx

Несмотря на то, что Nginx присутствует в репозиториях основных дистрибутивов, рекомендую использовать версию от разработчиков, это позволит более оперативно получать новые версии и новые возможности. Существует две ветки Nginx, основная и стабильная, первая имеет нечетную, вторая четную нумерацию. Разработка происходит следующим образом, все изменения основной ветки, скажем 1.7 фиксируются и переходят в стабильную 1.8, которая перестает разрабатываться и получает только обновления безопасности, основная ветка после этого получает номер 1.9 и в нее вносятся все изменения.

Сами разработчики рекомендуют использовать основную ветку, если только нет каких-то особых требований по совместимости. По своему опыту можем сказать, что основная ветка достаточно стабильна и может быть использована на рабочих серверах.

Для подключения репозиториев Nginx создадим в папке **/etc/apt/sources.list.d** файл **nginx.list**:

```
touch /etc/apt/sources.list.d/nginx.list
```

Потом добавим в него строки. Для Debian:

```
deb http://nginx.org/packages/mainline/debian/ codename nginx
deb-src http://nginx.org/packages/mainline/debian/ codename nginx
```

где **codename** - кодовое имя дистрибутива, например, **jessie** для Debian 8. Если вам не нужны исходные тексты, то репозитории **deb-src** можно не подключать (т.е. не добавлять эти строки).

Затем скачаем и установим PGP-ключ, необходимый для проверки подлинности:

```
cd
wget http://nginx.org/keys/nginx_signing.key
apt-key add nginx_signing.key
```

После чего можно обновить список пакетов и установить nginx:

```
apt-get update
apt-get install nginx
```

Теперь, если набрать в браузере адрес нашего сервера, вы увидите стандартную заглушку Nginx.

Рис. 249

Также проверить состояние веб-сервера можно командой:

```
service nginx status
```

```
root@debian-www:/etc/apt/sources.list.d# service nginx status
• nginx.service - LSB: Stop/start nginx
   Loaded: loaded (/etc/init.d/nginx)
   Active: active (running) since Вс 2015-11-15 23:25:11 MSK; 19s ago
   CGroup: /system.slice/nginx.service
           ├─1177 nginx: master process /usr/sbin/nginx -c /etc/nginx/nginx.conf
           └─1178 nginx: worker process
```

Рис. 250

Теперь перейдем к настройке. Для этого перейдем в папку **/etc/nginx** и откроем файл **nginx.conf**, мы будем перечислять настройки в порядке их нахождения в файле, если данной опции нет - ее следует добавить.

Прежде всего изменим пользователя, от имени которого работает nginx, в Debian/Ubuntu веб-сервер работает от пользователя **www-data** и чтобы избежать в будущем возможных коллизий с правами доступа приведем строку к виду:

```
user  www-data;
```

Затем укажем количество рабочих процессов, рекомендуется выбирать их количество по числу доступных процессорных ядер, в нашем случае 2:

```
worker_processes  2;
```

Приведем секцию **events** к виду:

```
events {
 worker_connections 1024;
 use epoll;
}
```

Первая опция задает количество соединений на рабочий процесс, вторая задает метод обработки соединений, явно укажем наиболее эффективный для Linux.

Теперь перейдем в секцию http и после строки

```
access_log  /var/log/nginx/access.log  main;
```

зададим следующие опции:

```
client_header_timeout 30;
client_body_timeout 30;
reset_timedout_connection on;
```

Они задают таймаут (в секундах) на чтение клиентом тела и заголовка запроса, последняя опция разрешает сброс соединений по таймауту.

```
client_max_body_size 32m;
client_body_buffer_size 128k;
```

Эти параметры ограничивают максимальный размер тела запроса клиента и задают буфер для чтения заголовка запроса. Максимальный размер тела запроса ограничивает размер файла, который может быть загружен веб-сервером.

```
sendfile on;
tcp_nopush on;
```

Также разрешим передачу файлов и оптимизируем этот процесс.

Изменим параметр:

```
keepalive_timeout  30;
```

Он задает таймаут постоянных (keep-alive) соединений, которые позволяют повысить производительность протокола HTTP/1.1, но незакрытое соединений впустую использует ресурсы сервера и поэтому такие соединения следует принудительно завершать.

Ниже зададим параметры **gzip-сжатия**:

```
gzip on;
gzip_disable "msie6";
gzip_proxied any;
gzip_min_length 1024;
gzip_comp_level 4;
gzip_types text/plain text/css application/json application/javascript application/x-javascript
text/xml application/xml application/xml+rss text/javascript application/atom+xml applica-
tion/rdf+xml;
```

Первая опция включает gzip-сжатие, затем отключаем его для младших версий IE (6 и ниже), если такие вдруг зайдут на наш сервер, разрешим сжимать проксированные запросы, это нужно для сжатия динамического содержимого, затем укажем минимальный размер сжимаемого ответа, чтобы не тратить ресурсы сервера на сжатие коротких ответов. Ниже задается уровень сжатия и типы сжимаемых данных.

В самом конце, после

```
include /etc/nginx/conf.d/*.conf;
```

добавим

```
include /etc/nginx/sites-enabled/*;
```

Это позволит подключать конфигурации виртуальных хостов из папки sites-enabled.

Сохраним и проверим конфиг командой:

```
nginx -t
```

После чего можно перезапустить nginx:

```
service nginx restart
```

Теперь можно перейти к настройке виртуальных хостов, создадим две папки:

```
mkdir /etc/nginx/sites-available
mkdir /etc/nginx/sites-enabled
```

В первой будут хранится настройки сайтов, а во второй мы будем создавать символьные ссылки для того, чтобы подключить настройки сайта к конфигурационному файлу nginx.

Перед тем как описывать виртуальные хосты, создадим структуру папок для их хранения:

```
mkdir /var/www
mkdir /var/www/example.org
```

Затем создадим конфигурационный файл для нашего первого сайта:

```
touch /etc/nginx/sites-available/example.org.conf
```

Какого-либо стандарта по названию файлов у nginx нет, поэтому можете придерживаться своей системы, главное, чтобы вам было понятно, какой файл за какой сайт отвечает.

Теперь откроем его и внесем следующий текст:

```
server {
  listen 80;

  server_name example.org;
```

```
  charset utf-8;

  root /var/www/example.org;
  index index.html index.htm index.php;

  access_log /var/log/nginx/example.org_access.log;
  error_log /var/log/nginx/example.org_error.log;
}

server {

  listen 80;

  server_name www.example.org;
  rewrite ^(.*) http://example.org$1 permanent;
}
```

Его синтаксис достаточно прост и понятен, первая секция **server** задает основные параметры сайта, его имя, кодировку, расположение корневой директории и файлов логов. Вторая секция нужна для перенаправления сайта **с www** на **без www**.

Если вы хотите сделать данный виртуальный хост сайтом по умолчанию, т.е. тем на который будут переадресовываться все запросы, для которых nginx не нашел подходящего виртуального хоста или без имени сервера вообще, например, по IP-адресу, то добавьте к директиве **listen** опцию **default**, начиная с версии 0.8.1 можно использовать опцию **default_server**:

```
 listen 80 default;
```

Директива **index** указывает индексные файлы, которые будет искать в данном расположении веб-сервер в порядке их перечисления, так если в директории имеются одновременно **index.html** и **index.php** - использоваться всегда будет первый. Указанная конструкция универсальна, но на практике лучше указать один тип индексного файла, тот что реально используется.

Сохраняем конфигурацию и подключаем ее к nginx:

```
ln -s /etc/nginx/sites-available/example.org.conf  /etc/nginx/sites-enabled/
```

Проверяем конфигурацию и заставим nginx ее перечитать:

```
nginx -t
service nginx reload
```

Теперь поместим в корневую директорию сайта файл **index.html** со следующим содержимым:

```
<body><h1>OK!</h1></body>
```

Теперь набираем в браузере имя нашего сайта и убеждаемся, что все работает.

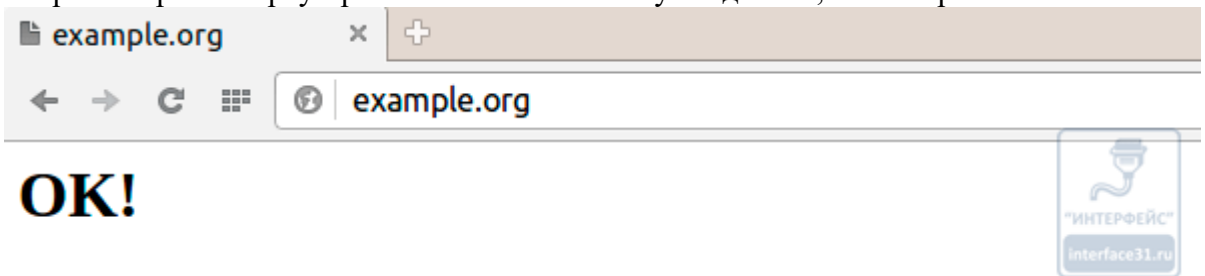


Рис. 251

2. Устанавливаем PHP-FPM

Для работы с современными веб-приложениями вам потребуется поддержка популярного скриптового языка PHP, Nginx поддерживает работу через FastCGI, но не имеет соб-

ственного менеджера процессов, поэтому мы будем использовать для этой цели PHP-FPM.

Важно! В современных дистрибутивах используется более новая версия PHP 7, чтобы работать с новой версией языка вместо **php5** в приведенных ниже командах следует указывать **php7.x** или просто **php** например, вместо **php5-imagick** нужно набрать **php7.0-imagick** или **php-imagick.**

Установим его:

```
apt-get install php5-fpm
```

Все необходимые пакеты и интерпретатор PHP будут установлены по зависимостям. Также имеет смысл сразу установить некоторые модули PHP, например, для работы с графикой:

```
apt-get install php5-gd php5-imagick
```

Настройки PHP-FPM по умолчанию достаточно оптимальны и никаких вмешательств в них не требуется, однако следует подправить некоторые опции PHP, для этого откроем **/etc/php5/fpm/php.ini** и найдем там следующие опции:

```
post_max_size = 8M
```

этот параметр задает максимальный размер данных загружаемых методом POST, влияет, например, на размер загружаемых средствами PHP файлов. По умолчанию 8 МБ, можем изменить по собственным потребностям.

Если вы будете использовать PHP-приложения (CMS) работающие в кодировке отличной от UTF-8, то приведите к следующему виду опцию:

```
default_charset = ""
```

Затем раскоменнтируйте и установите опцию:

```
cgi.fix_pathinfo=0
```

Это закроет возможную уязвимость в PHP.

Еще ниже надо найти и увеличить размер максимально загружаемого файла:

```
upload_max_filesize = 8M
```

Данное значение должно быть больше или равно значению **post_max_size**, иначе вы будете ограничены в загрузке файлов меньшим из указанных в этих опциях размером.

Сохраним изменения и перезапустим PHP-FPM:

```
service php5-fpm restart
```

Теперь следует научить Nginx работать с PHP-FPM, для этого в файл конфигурации виртуального хоста нужно добавить настройки, которые будут перенаправлять (проксировать) все запросы к динамическому содержимому на FastCGI-шлюз.

Если сайтов несколько, то аналогичные настройки потребуется добавить каждому виртуальному хосту, поэтому, чтобы не делать лишних действий, имеет смысл вынести данные настройки в шаблон и подключать к виртуальному хосту уже его. Такой подход имеет еще один плюс, если вам потребуется изменить настройки, то делать это придется только в одном месте.

Создадим директорию для хранения шаблонов:

```
mkdir /etc/nginx/templates
```

После чего создадим в ней шаблон для работы с PHP-FPM:

```
touch /etc/nginx/templates/php-fpm.conf
```

Откроем его и добавим следующий текст:

```
location ~ \.php$ {
  try_files $uri =404;
  fastcgi_pass unix:/var/run/php5-fpm.sock;
  fastcgi_index index.php;
  include fastcgi_params;
  fastcgi_param  SCRIPT_FILENAME  $document_root$fastcgi_script_name;
}
```

Указанный нами блок **location** будет обрабатывать все запросы к php-файлам, первая директива в нем проверяет наличие запрошенного файла, в противном случае отдавая ошибку 404. Вторая устанавливает параметры соединения с FastCGI-шлюзом, в нашем случае с PHP-FPM, соединение устанавливается через UNIX-сокет, как наиболее производительный способ соединения. Затем указывается индексный файл и подгружаются настройки Nginx для FastCGI.

Важно! Обратите внимание, что PHP 7 имеет иной путь к UNIX-сокету, поэтому следует указывать **/var/run/php/php7.0-fpm.sock**

В принципе этого уже достаточно, чтобы работать с динамическим содержимым, но не будем спешить и добавим в файл еще несколько блоков.

```
location ~ /\.ht {
  deny all;
}
```

Несмотря на то, что Nginx не использует htaccess-файлы, они, вместе с файлами htpasswd могут находиться в директории сайта, особенно если до этого он работал на Apache и будет правильно запретить доступ к ним в целях безопасности.

Также следует настроить кэширование статического содержимого:

```
location ~*
\.(gif|jpeg|jpg|txt|png|tif|tiff|ico|jng|bmp|doc|pdf|rtf|xls|ppt|rar|rpm|swf|zip|bin|exe|dll|deb|cur)$ {
  expires 168h;
}
```

Данная конструкция включает кэширование на стороне браузера, сообщая тому, что "срок годности" указанных файлов - 168 часов (1 неделя) и при последующих обращениях на ваш сайт данные файлы следует брать из локального кэша. Мы привели примерный список, вы можете самостоятельно добавить в него нужные расширения файлов.

Также зададим кэширование для скриптов и стилей:

```
location ~* \.(css|js)$ {
  expires 180m;
}
```

Для них установим срок кэширования в 3 часа, что позволит соблюсти баланс между скоростью применения возможных изменений в этих файлах и уменьшением количества запросов к вашему сайту.

Теперь откроем файл конфигурации виртуального хоста и в конце первой секции **server** добавим строку подключения шаблона:

```
include /etc/nginx/templates/php-fpm.conf;
```

Сохраним все настройки, проверим конфигурацию и перезапустим Nginx.

```
nginx -t
service nginx reload
```

Чтобы проверить работу PHP создадим в корневой директории сайта файл **test.php** со следующим содержимым:

```
<?php
phpinfo();
?>
```

Теперь, если обратиться к данному файлу через браузер вы должны увидеть стандартную страницу с информацией о PHP.

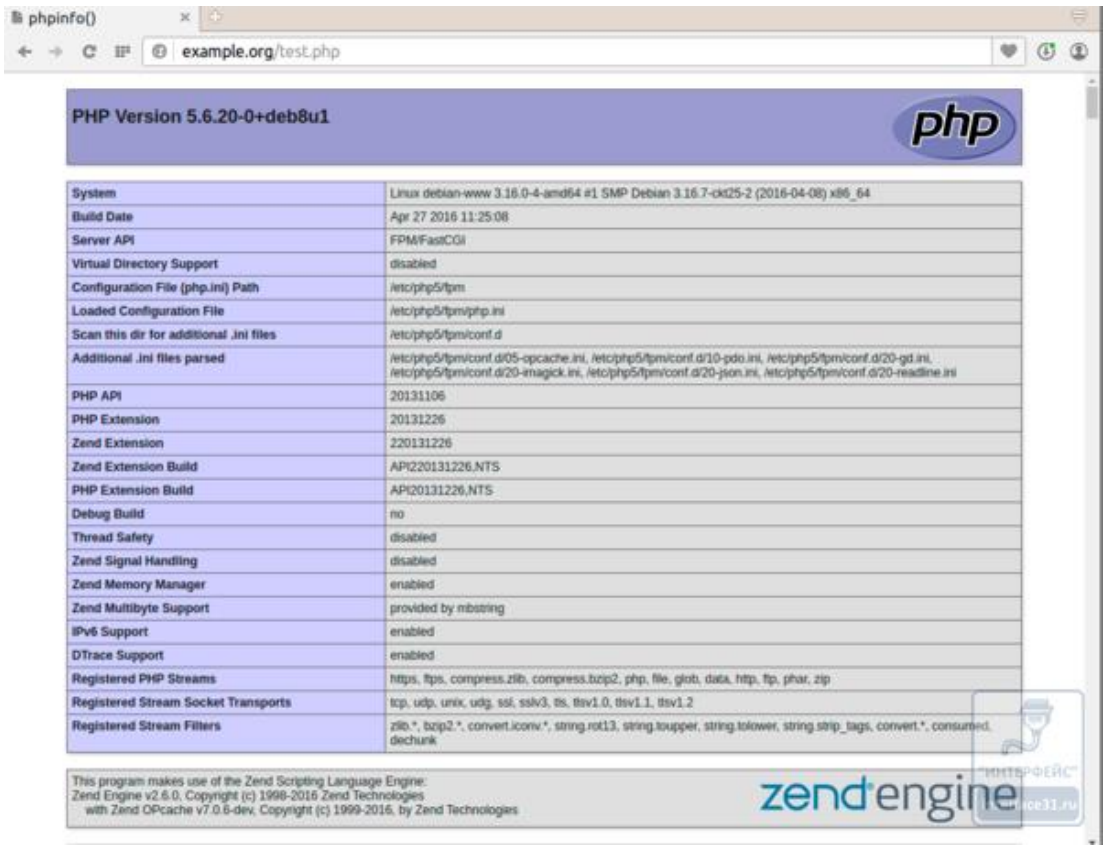


Рис. 252

3. Установка MySQL и phpMyAdmin

СУБД MySQL широко используется для хранения информации в современных веб-приложениях. Это один из самых важных компонентов веб-сервера, так как в базе данных обычно хранится вся информация сайта, кроме статического содержимого (изображений, файлов и т.п.).

Для установки MySQL выполните:

```
apt-get install mysql-server php5-mysql
```

Важно! В свежих выпусках Debian (и его производных) вместо пакета **mysql-server** следует установить **mariadb-server**, который полностью совместим с MySQL.

Данная команда установит MySQL сервер и модуль PHP для работы с ним. В процессе установки вас попросят ввести пароль суперпользователя СУБД (root), не путать с суперпользователем системы.

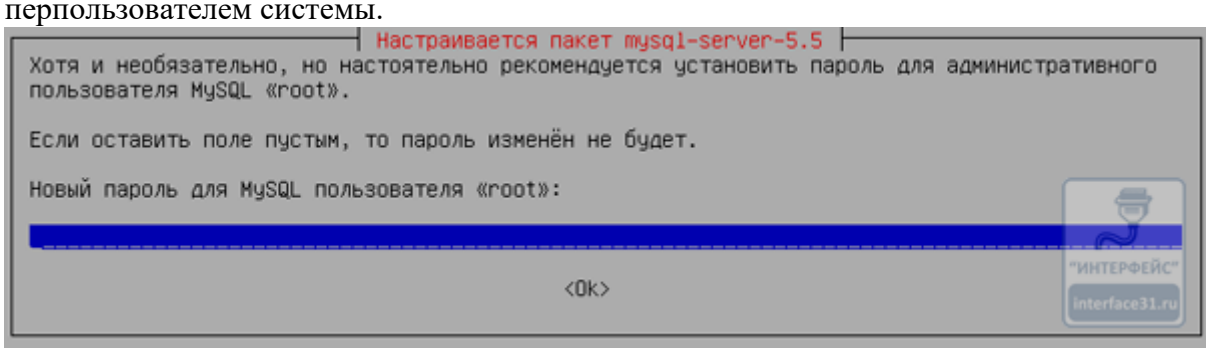


Рис. 253

Для повседневной работы с MySQL удобно использовать веб-приложение администрирования phpMyAdmin, установим его:

```
apt-get install phpmyadmin
```

Установщик phpMyAdmin не умеет конфигурировать Nginx для работы с ним, поэтому ничего не выбираем на данном этапе, а все настройки выполним позже вручную.

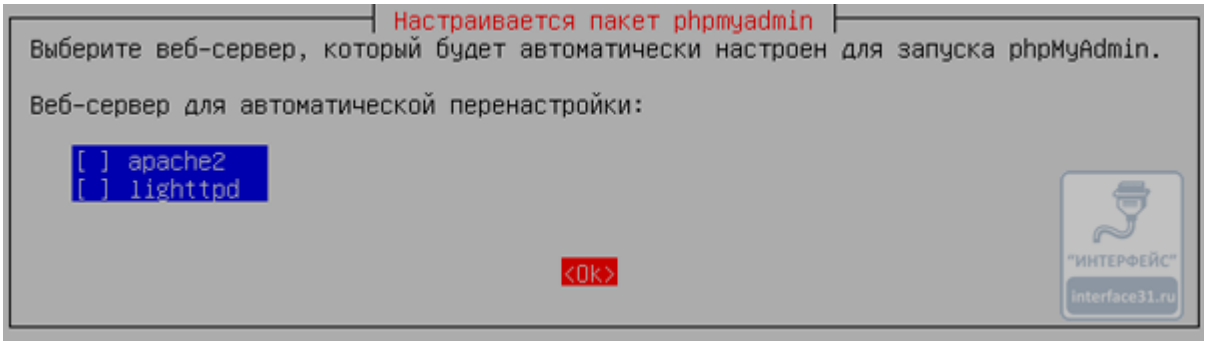


Рис. 254

Для этого создадим еще один файл шаблона:

```
touch /etc/nginx/templates/phpmyadmin.conf
```

и внесем в него следующий текст:

```
location /phpmyadmin {
  root /usr/share/;
  index index.php;

  location ~ ^/phpmyadmin/(.+\.php)$ {
    try_files $uri =404;
    root /usr/share/;
    fastcgi_pass unix:/var/run/php5-fpm.sock;
    fastcgi_index index.php;
    include fastcgi_params;
    fastcgi_param SCRIPT_FILENAME $document_root$fastcgi_script_name;
  }

  location ~* ^/phpmyadmin/(.+\.(jpg|jpeg|gif|css|png|js|ico|html|xml|txt))$ {
    root /usr/share/;
    expires 1M;
  }
}

location /phpMyAdmin {
  rewrite ^/* /phpmyadmin last;
 }
```

На первый взгляд синтаксис может показаться довольно сложным, но если разобрать правила по частям, то мы увидим, что ничего сложного нет. Все это мы уже обсуждали выше. Самый последний **location** осуществляет перенаправление на phpMyAdmin с адресов вида **имя_домена/phpmyadmin**.

Для подключения phpMyAdmin к сайту в описание виртуального хоста добавьте включение еще одного шаблона:

```
include /etc/nginx/templates/phpmyadmin.conf;
```

Проверьте конфигурацию и перезапустите Nginx, после чего наберите в браузере имя вашего сайта, добавив после него **/phpmyadmin**, если все сделано правильно, то вы попадете в админ-панель приложения.

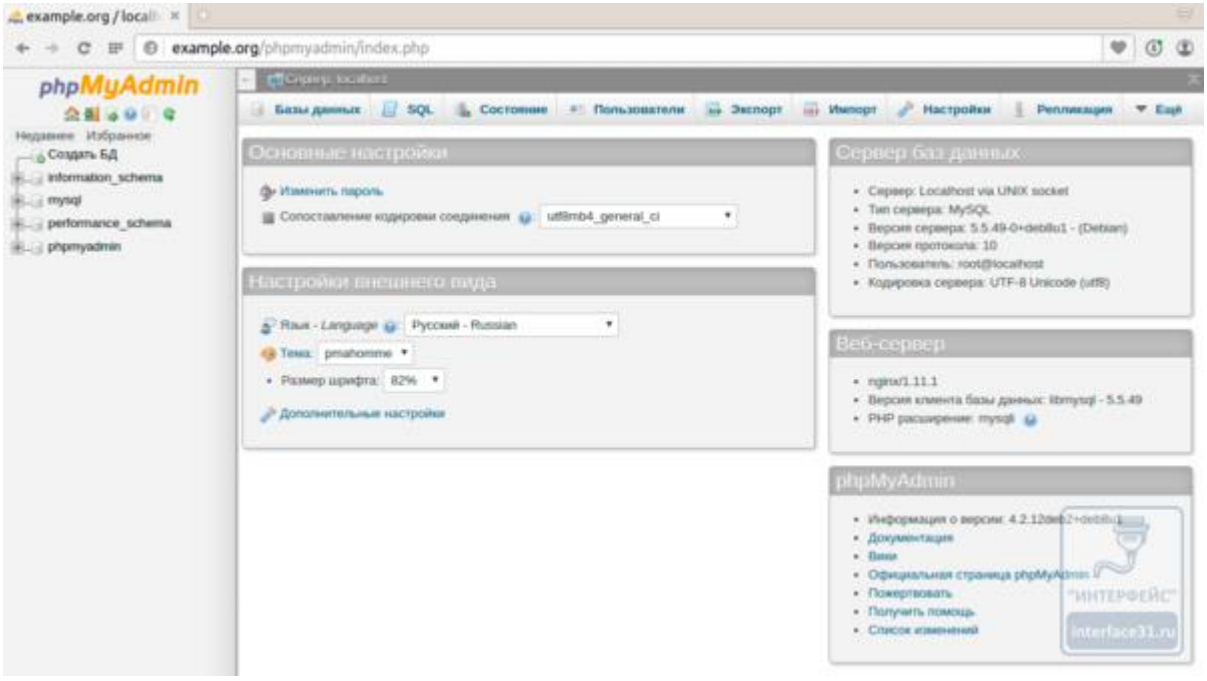


Рис. 255

Для ее устранения выполните:

```
ln -s /etc/php5/mods-available/mcrypt.ini /etc/php5/fpm/conf.d/
service php5-fpm restart
```

На этом настройку нашего сервера можно считать законченной. При переносе или размещении на нем новых сайтов не забывайте правильно устанавливать права и владельца. Стандартный набор прав: 644 для файлов и 755 для папок, его можно быстро установить командой:

```
chmod -R u=rw,g=r,o=r,a+X /var/www/example.org
```

Но учтите, что некоторые CMS требуют нестандартных прав на некоторые папки и файлы, поэтому уточните этот вопрос в документации.

Также не забывайте устанавливать правильного владельца, им должен быть пользователь, от имени которого работает веб-сервер, в нашем случае **www-data**, владелец устанавливается командой:

```
chown -R www-data:www-data /var/www/example.org
```

Обычно данных мер достаточно, но встречаются CMS которые требуют дополнительной настройки для работы с Nginx, в этом случае следует обратиться к документации или на форум поддержки движка. Поэтому мы еще раз повторимся, что данное решение требует определенного опыта и квалификации и не рекомендуется начинающим, а также тем, кто не хочет возиться с настройками, а хочет быстро получить работающий сайт.

Сделайте скриншоты (фотографии) процесса установки и настройки веб-сервера на базе Nginx + PHP-FPM и вставьте в отчёт.

2.17 Практическая работа№ 17
Настройка сервера DNS в ОС Debian. Настройка сервера DHCP в ОС Debian

Задание 1:

Одной из реализацией в Linux DNS серверов является BIND. Текущая реализация это BIND9. Все настроечные файлы находятся в каталоге /etc/bind/. Основной файл конфигурации - named.conf. Установим и настроим сервер DNS BIND9 на основе операционной системы Debian.

1. Для начала откроем терминал. Все действия по установке и настройке DNS сервера производятся с правами root или с помощью sudo.

Прописываем необходимые репозитории для обновления системы, а также установки нужных пакетов:

```
# nano /etc/apt/sources.list


# security updates
deb http://security.debian.org/ jessie/updates main contrib non-free
deb-src http://security.debian.org/ jessie/updates main contrib non-free
# binary and source packages
deb http://ftp.ru.debian.org/debian/ jessie main contrib non-free
deb-src http://ftp.ru.debian.org/debian/ jessie main contrib non-free
# jessie-updates
deb http://ftp.ru.debian.org/debian/ jessie-updates main contrib non-free
deb-src http://ftp.ru.debian.org/debian/ jessie-updates main contrib non-free
```

Для **Debian 9** вместо **jessie** указываем **stretch**.

Сохраняем файл, далее выполняем следующие команды:

```
# apt-get update
# apt-get upgrade
```

Первая команда обновит информацию о пакетах, вторая команда приведет к обновлению нашей системы до актуального состояния.

Устанавливаем пакет **bind9** (dns сервер):

```
# apt-get install bind9
```

Директория, в которой находятся настроечные файлы dns сервера - **/etc/bind/**. Основным настроечным файлом является **named.conf.options**. Настраиваем файл **named.conf.options** (находится в каталоге **/etc/bind**):

```
# cd /etc/bind
# nano named.conf.options
```

Прописываем в файле **named.conf.options**:

```
acl mynetwork { 192.168.91.0/24 ; 127.0.0.1; };
options {
    directory "/var/cache/bind";
    forwarders {
         8.8.8.8;
};
listen-on {
        192.168.91.10;
        192.168.91.20;
};
dnssec-validation auto;
auth-nxdomain no;     # conform to RFC1035
```

```
listen-on-v6 { none; };
allow-query { mynetwork; };
```

Где:
allow-query { mynetwork; }; - список тех, кто имеет право запрашивать информацию, если хотите, чтобы принимать запросы ото всех, вместо **mynetwork** ставим **any**.
acl - ограничивает адреса, которые могут запрашивать зоны с сервера DNS.
forwarders { 8.8.8.8; }; - прописываем DNS сервера, у которого можно получить информацию, если информация о доменах неизвестна нашему серверу.
listen on { 192.168.91.10; 192.168.91.20; }; - прописываем DNS сервера, которые будут использованы для отображения IP адресов в имена и наоборот.
listen-on-v6 { none; }; - если IP 6 версии не используем.
auth-nxdomain no; - параметр для совместимости с RFC1035.
Проверяем:

```
# named-checkconf
```

Далее редактируем файл named.conf.local:

```
# nano /etc/bind/named.conf.local
```

В файле прописываем зону прямого и обратного просмотра для домена. Зона прямого просмотра - тип зоны, в котором в ответ на имя получают IP адрес. Соответственно ответственность зоны обратного просмотра состоит в том, чтобы получить по IP адресу имя компьютера:

```
zone "sigro.ru" {
        type master;
        file "/etc/bind/db.sigro.ru";
};

zone "91.168.192.in-addr.arpa" {
        type master;
        file "/etc/bind/91.168.192.in-addr.arpa.zone";
};
```

91.168.192.in-addr.arpa - обратная зона просмотра, берётся из IP адреса DNS сервера (в данном случае 192.168.91.10);
db.sigro.ru - прямая зона просмотра, домен в данном случае называется sigro.ru.
После каждого изменения конфигурационного файла желательно провести проверку синтаксиса файла, затем переходить к настройке других файлов. Делается это командой:

```
# named-checkconf
```

Далее начинаем прописывать зону прямого просмотра. Чтобы произвести настройку быстрее и по шаблону, копируем фал db.local и изменяем уже вновь созданный файл настройки зоны прямого просмотра (имена для файлов зоны прямого и обратного просмотра можно придумать любые, но принято, чтобы они были читабельны):

```
# cp db.local db.sigro.ru
# nano db.sigro.ru
```

```
;
; BIND data file for local loopback interface
;
$TTL 604800
$ORIGIN sigro.ru.

@       IN      SOA      nic1             admin (
                        2017060100              ; Serial
                         604800                 ; Refresh
                          86400                 ; Retry
                         2419200                ; Expire
                           604800 )             ; Negative Cache TTL
;
@       IN      NS      nic1.sigro.ru.
@       IN      A       192.168.91.10
; @     IN      AAAA    ::1
nic1    IN      A       192.168.91.10
nic2    IN      A       192.168.91.20
cl1     IN      A       192.168.91.101
```

; - после данного знака возможно делать комментарий
$TTL 604800 - time to live (время кэширования из вашей зоны)
$ORIGIN sigro.ru. - при использовании $ORIGIN к именам будет автоматически дописываться в данном случае sigro.ru. (не забываем в конце точку). Например, при считывании зоны прямого просмотра, вместо nic1 автоматически будет подставлено nic1.sigro.ru.
@ IN SOA nic1 admin - запись SOA (начало ответственности)
nic1 - имя первичного dns сервера
admin - почтовый адрес пользователя, отвечающего за эту зону
2017060100 - серийный номер зоны (десятизначное число)
604800 - период обновления
86400 - повтор каждые 86400 с
2419200 - время хранения информации
604800 - время хранения в кэше удаленных серверов негативных ответов
Прописываем зону обратного просмотра. Для этого создаём файл зоны обратного просмотра и производим изменения в вновь созданном файле:

```
# cp db.sigro.ru 91.168.192.in-addr.arpa.zone
# nano 91.168.192.in-addr.arpa.zone
;
; BIND data file for local loopback interface
;
$TTL 604800
$ORIGIN 91.168.192.in-addr.arpa.

@       IN      SOA     nic1.sigro.ru.          admin.sigro.ru. (
                        2017060100              ; Serial
                         604800                 ; Refresh
                          86400                 ; Retry
                         2419200                ; Expire
                           604800 )             ; Negative Cache TTL
;
                NS      nic1.sigro.ru.
```

```
10          PTR     nic1.sigro.ru.
20          PTR     nic2.sigro.ru.
101         PTR     cl1.sigro.ru.
```

Делаем так, чтобы сервер DNS работал с новой конфигурацией:

```
# /etc/init.d/bind9 reload
```

Производим проверки:

```
# named-checkconf
# named-checkconf -z
# named-checkzone 91.168.192.in-addr.arpa 91.168.192.in-addr.arpa.zone
# host 192.168.91.10
# nslookup sigro.ru
# nslookup 192.168.91.10
```

named-checkconf - проверка правильности синтаксиса конфигурационных файлов, рекомендуется делать после каждого изменения в конфигурационном файле.
named-checkconf -z - пытается произвести действия, такие же как **bind** при загрузке зон.
nslookup sigro.ru - должен быть показан адрес проверяемого сервера (т.е. в данном случае **Address: 192.168.91.10**).
nslookup 192.168.91.10 - должен быть показано имя проверяемого сервера (т.е. в данном случае **name = nic1.sigro.ru**).
Если ошибок нет, то сервер DNS сконфигурирован правильно

Сделайте скриншоты (фотографии) процесса настройки сервера времени и лицензирования и вставьте в отчёт.

Задание 2:
Рассмотрим, как установить и настроить DHCP-сервер.

1. Установка сервера ISC DHCP

Установим пакет.

```
sudo apt install isc-dhcp-server
```

На всякий случай сделаем резервную копию конфигурационного файла

```
cp /etc/dhcp/dhcpd.conf{,.backup}
cat /dev/null > /etc/dhcp/dhcpd.conf
```

2. Настройка DHCP

Задаем настройки сети, диапазон выдаваемых адресов, маску сети и выдаваемый DNS

```
nano /etc/dhcp/dhcpd.conf
subnet 192.168.38.0 netmask 255.255.255.0 {
 range 192.168.38.100 192.168.38.254;
 option subnet-mask 255.255.255.0;
 option broadcast-address 192.168.38.255;
 option domain-name-servers 8.8.8.8, 8.8.4.4;
 option domain-name "workgroup";
```

```
option routers 192.168.38.1;
default-lease-time 7200;
max-lease-time 480000;
}
```

Все строки параметров в файле конфигурации dhcp заканчиваются точкой с запятой (;). Некоторые параметры могут иметь не одно значение, например, domain-name-servers, у которого два IP-адреса, разделенные запятой. Строки, начинающиеся с '#', являются комментариями и не обрабатываются сервером dhcp.
Некоторые общие параметры сервера DHCP:
subnet— Параметр объявляет подсеть (в нашем случае 192.168.38.0 с маской 255.255.255.0)
range – Диапазон выдаваемых адресов (от 192.168.38.100 до 192.168.38.254).
option subnet-mask – Маска сети. (255.255.255.0)
option broadcast-address – Широковещательный адрес. (192.168.38.255)
domain-name-servers – Адреса серверов DNS. (8.8.8.8, 8.8.4.4)
option domain-name – Доменное имя.(workgroup)
option routers – Определяет IP-адрес вашего шлюза или точки выхода в сеть. (192.168.38.1)
После того как вы отредактировали основной файл конфигурации и объявили диапазоны IP, откройте файл **/etc/default/isc-dhcp-server** и замените параметр ***INTERFACESv4*** на имя сетевого интерфейса, который смотрит внутрь сети. Чтобы узнать его имя воспользуйтесь командами ipconfig или **ip.**

```
INTERFACESv4 = "enp1s8"
```

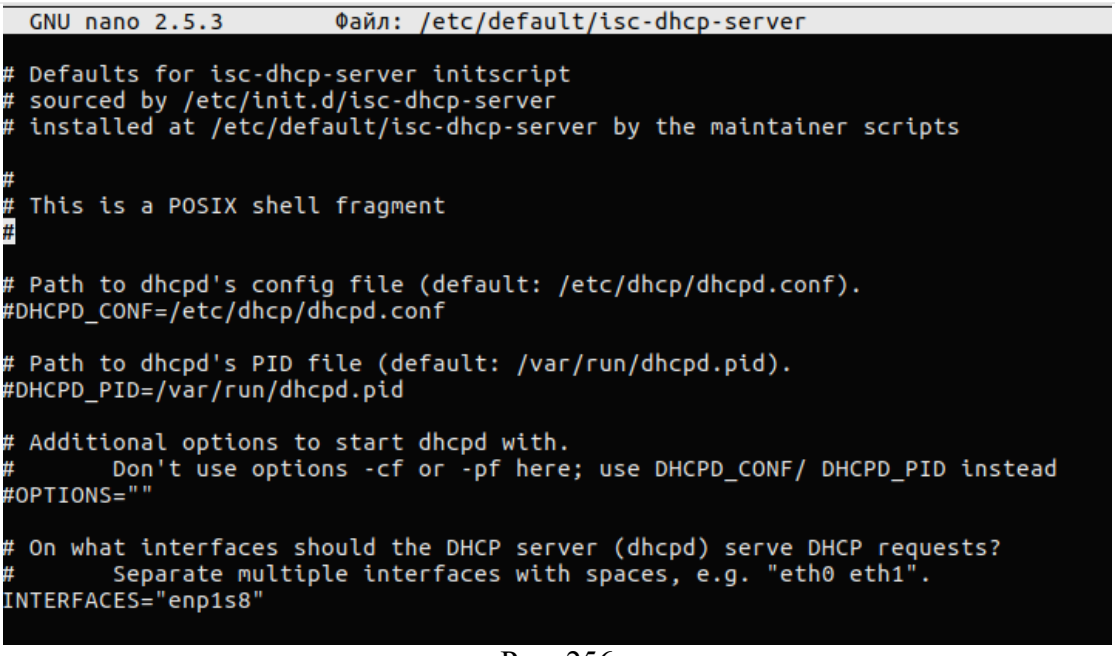


Рис. 256

Наконец, после внесения всех изменений перезапустите сервер dhcp, чтобы применить новую конфигурацию и проверить статус службы, выполнив следующие команды:

```
systemctl restart isc-dhcp-server
systemctl status isc-dhcp-server
```

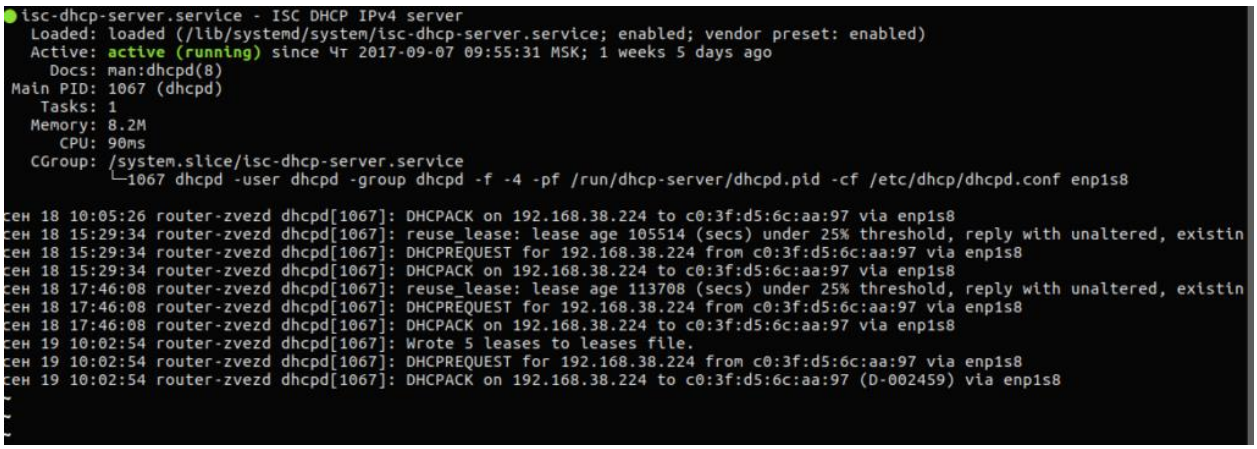


Рис. 257

3. Настройка DHCP-сервера с резервированием IP-адреса.

Часто возникает необходимость зарезервировать за устройством (сервером, принтером и т.д.) постоянный IP-адрес. В этом случае вам нужно знать его MAC- адрес.

```
nano /etc/dhcp/dhcpd.conf
```

Пример резервирования IP-адреса 192.168.38.5 за компьютером SERVER:

```
subnet 192.168.38.0 netmask 255.255.255.0 {
range 192.168.38.100….
.....
host SERVER {
  hardware ethernet 08:60:6e:d6:5e:ff;
  fixed-address 192.168.38.5;}
}
```

После того, как вы внесли изменения в конфигурационный файл, перезапустите сервер DHCP следующей командой:

```
systemctl restart isc-dhcp-server
```

Вы успешно установили и настроили DHCP-сервер.

Сделайте скриншоты (фотографии) процесса настройки сервера DHCP и вставьте в отчёт.

2.18 Практическая работа№ 18
Настройка файловых серверов в ОС Debian

Задание:

1. Подготовка системы

Необходимо использовать виртуальную машину с двумя жесткими дисками, один для системы, второй для данных, точку монтирования диска для данных мы указали как **/samba**.

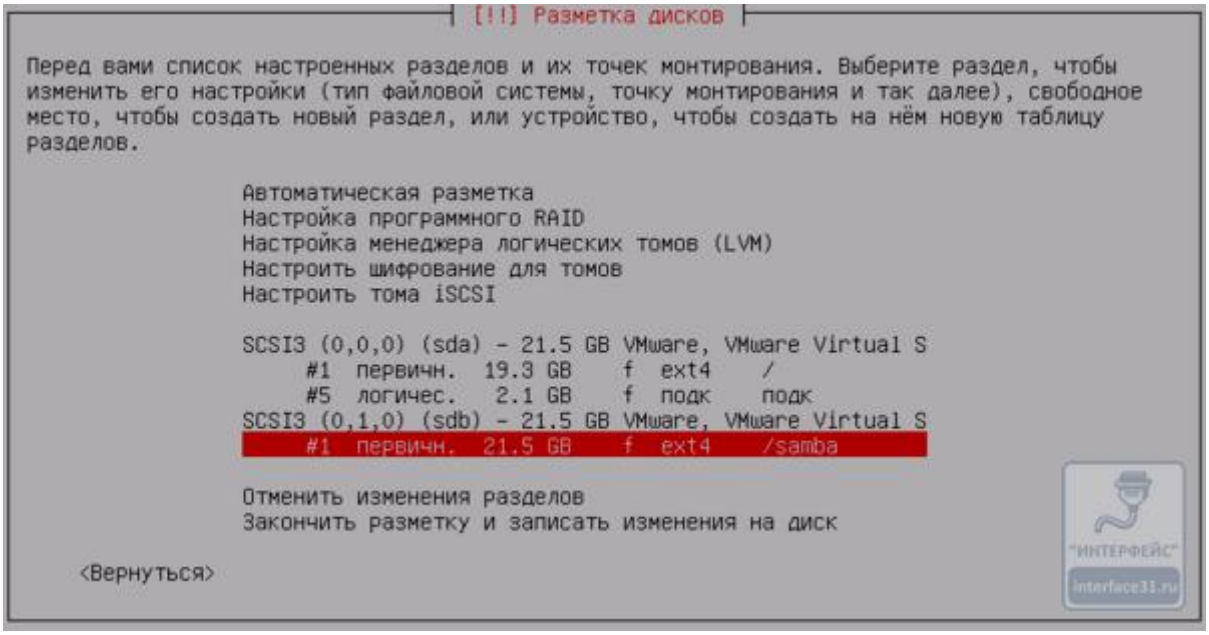


Рис. 258

Также обратите внимание на имя компьютера, Samba 4 будет использовать его в качестве NetBIOS имени.

После установки ОС следует изменить настройку лимита на количество одновременно открытых файлов, в Linux это 1024, а в Windows 16384. Для этого откройте файл **/etc/security/limits.conf** и добавьте в конце две строки:

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

После чего сервер следует перезагрузить.

2. Установка и базовая настройка Samba 4

Установка Samba предельно проста:

```
apt install samba
```

После чего откроем файл **/etc/samba/smb.conf** и выполним общие настройки. Большинство указанных опций в файле уже есть, многие из них даже не потребуется менять, но их назначение будет полезно знать, поэтому мы прокомментируем наиболее важные из них.

За общие настройки сервера отвечает секция **[global]**, которая, кстати, прекрасно прокомментирована. Обратите внимание на два вида комментариев опций, если для этого используется символ **#** - то указанное значение применяется по умолчанию, а символ **;** обозначает предлагаемый вариант настройки.

Начнем, опции перечисляются в порядке их следования в файле:

```
workgroup = WORKGROUP
```

Обозначает рабочую группу Windows, по умолчанию WORKGROUP.

```
;  interfaces = 127.0.0.0/8 eth0
```

Предлагаемая опция, которые определяет интерфейсы или подсети, с которыми будет работать Samba. Допускается смешанная запись, как в примере выше, либо можно указать только интерфейсы:

```
interfaces = lo ens33
```

Или только подсети:

```
interfaces = 127.0.0.0/8 192.168.38.0/24
```

Но само по себе указание интерфейсов не ограничивает Samba, для того чтобы ограничения начали действовать нужно включить следующую опцию:

```
bind interfaces only = yes
```

Следующая опция указывает расположение логов:

```
 log file = /var/log/samba/log.%m
```

По умолчанию лог выключен, для того чтобы его включить добавьте в файл опцию:

```
log level = 1
```

Если вам нужен более подробный лог - установите более высокий уровень, минимальное значение - 1, максимальное - 5.

Также закомментируйте опцию:

```
# syslog = 0
```

В настоящий момент она является не рекомендованной (deprecated).

```
server role = standalone server
```

Обозначает простой файловый сервер, не требующий подключения к домену.

```
map to guest = bad user
```

Определяет способ определения гостевого доступа, при указанном значении гостем будет считаться любой пользователь, который отсутствует в базе Samba. Также могут использоваться значения **never** - не использовать гостевой доступ и **bad password** - в этом случае гостем будет считаться в том числе, и существующий пользователь если он неправильно введет пароль. Данное значение использовать не рекомендуется, так как при ошибке в пароле пользователь все равно получит доступ, но с гостевыми правами.

На этом общая настройка сервера закончена. Проверим конфигурацию на ошибки:

```
testparm
```

И перезапустим сервер

```
service smbd restart
```

Настройка общего ресурса с гостевым доступом

Начнем с самого простого варианта - создадим общий ресурс, доступ к которому может иметь любой пользователь. Для этого добавим в конец файла **/etc/samba/smb.conf** следующие строки.

```
[public]
comment = Shared for all
path = /samba/public
read only = no
guest ok = yes
```

В квадратных скобках задаем имя ресурса, все что ниже скобок - секция этого ресурса. В ней мы указали следующие опции:

- **comment** - описание ресурса, необязательный параметр;
- **path** - путь к директории;
- **read only** - режим только чтения, указываем **no**;
- **guest ok** - разрешен ли гостевой доступ, указываем **yes**;

Теперь создадим саму директорию:

```
mkdir /samba/public
```

и установим на нее необходимые права, для гостевого ресурса это 777:

```
chmod 777 /samba/public
```

Перезапускаем Samba и пробуем получить доступ с любого Windows-клиента.

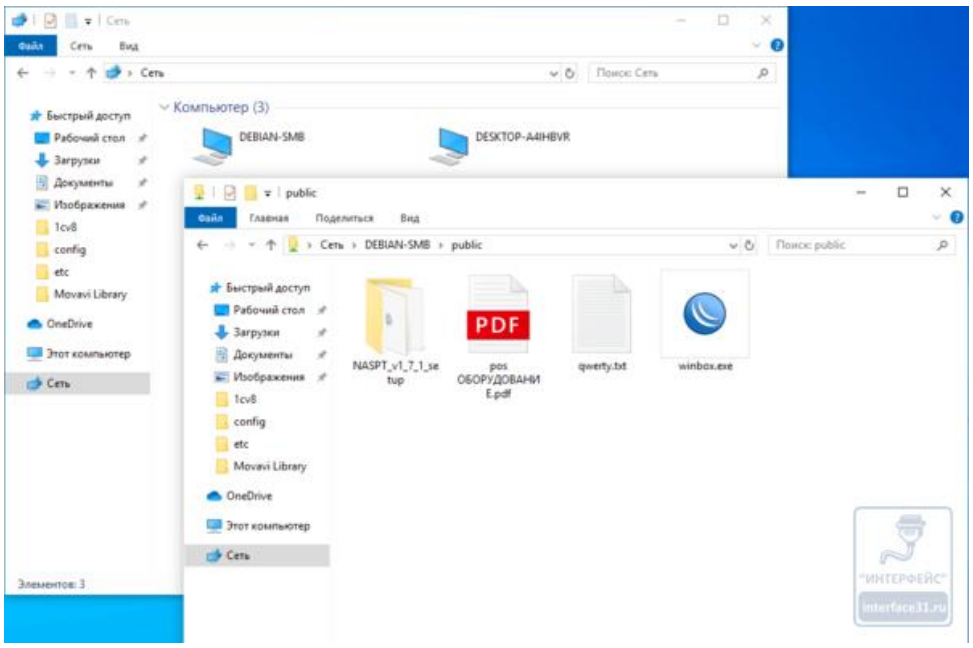


Рис. 259

Если все сделано правильно, то сервер появится в сетевом окружении, и вы без проблем получите доступ к созданной нами общей папке.

3. Настройка общего ресурса с парольным доступом

Гостевой доступ — это просто и удобно, но не всегда приемлемо. Существуют ситуации, когда доступ к общему ресурсу должны иметь только определенные пользователи. В нашем примере создадим два таких ресурса: для бухгалтерии и для IT-отдела.

Снова откроем конфигурационный файл и добавим в него две секции:

```
[buch]
path = /samba/buch
read only = no
guest ok = no
[adm]
path = /samba/adm
read only = no
guest ok = no
```

Они предельно просты и отличаются запретом гостевого доступа - **guest ok = no**. Для того, чтобы разделить доступ к ресурсам будем использовать группы пользователей, создадим две новые группы для наших подразделений:

```
groupadd smbbuch
groupadd smbadm
```

Теперь создадим каталоги:

```
mkdir /samba/buch
mkdir /samba/adm
```

и изменим группу владельца:

```
chgrp smbbuch /samba/buch
chgrp smbadm /samba/adm
```

Затем установим права:

```
chmod 2770  /samba/buch
chmod 2770 /samba/adm
```

Значение 2770 обозначает что мы предоставляем полные права владельцу и группе, для остальных доступ запрещен. А первая двойка устанавливает SGID для каталога, что обеспечивает присвоение группы каталога каждому создаваемому в нем файлу.

В некоторых случаях определенный интерес представляет выставление дял каталога **sticky bit**, который означает, что удалить или переименовать файл может только его

владелец, но работать с ним, в том числе изменять, может любой пользователь, имеющий права записи в каталог. Для этого вместо набора прав **2770** используйте права **3770**.
На этом настройки закончены, не забываем перезапустить Samba. Но в наших группах пока нет пользователей, давайте добавим их туда.
Начнем с уже существующих пользователей, в нашем случае это пользователь andrey, который является главным администратором и должен иметь доступ к обоим ресурсам. Поэтому добавим его в обе группы:

```
usermod -aG smbbuch andrey
usermod -aG smbadm andrey
```

Затем добавим его в базу Samba:

```
smbpasswd -a andrey
```

При этом потребуется установить пароль для доступа к Samba-ресурсам, он должен совпадать с основным паролем пользователя. После чего включим эту учетную запись:

```
smbpasswd -e andrey
```

Проверяем, после ввода пароля мы должны получить доступ к созданным нам ресурсам. Также обратите внимание, после аутентификации в списке общих ресурсов появилась папка с именем пользователя, подключенная только на чтение.

Рис. 260

С настройками по умолчанию Samba предоставляет каждому существующему пользователю доступ только на чтение к его домашнему каталогу. На наш взгляд это довольно удобно и безопасно. Если вас не устраивает такое поведение - удалите из конфигурационного файла секцию **[homes]**.
Теперь о других пользователях. Скажем у нас есть бухгалтер Иванова и админ Петров, каждый из которых должен иметь доступ к своему ресурсу. В тоже время иметь доступ к самому Samba-серверу им необязательно, поэтому создадим новых пользователей следующей командой:

```
useradd -M -s /sbin/nologin ivanova
useradd -M -s /sbin/nologin petrov
```

Ключ **-M** заводит пользователя без создания домашнего каталога, а **-s /sbin/nologin** исключает возможность входа такого пользователя в систему.
Поместим каждого в свою группу:

```
usermod -aG smbbuch ivanova
usermod -aG smbadm petrov
```

Затем добавим их в базу Samba, при этом потребуется установить им пароли:

```
smbpasswd -a ivanova
smbpasswd -a petrov
```

И включим эти учетные записи

```
smbpasswd -e ivanova
smbpasswd -e petrov
```

Если все сделано правильно, то пользователь будет иметь доступ к своим ресурсам и не иметь к чужим.

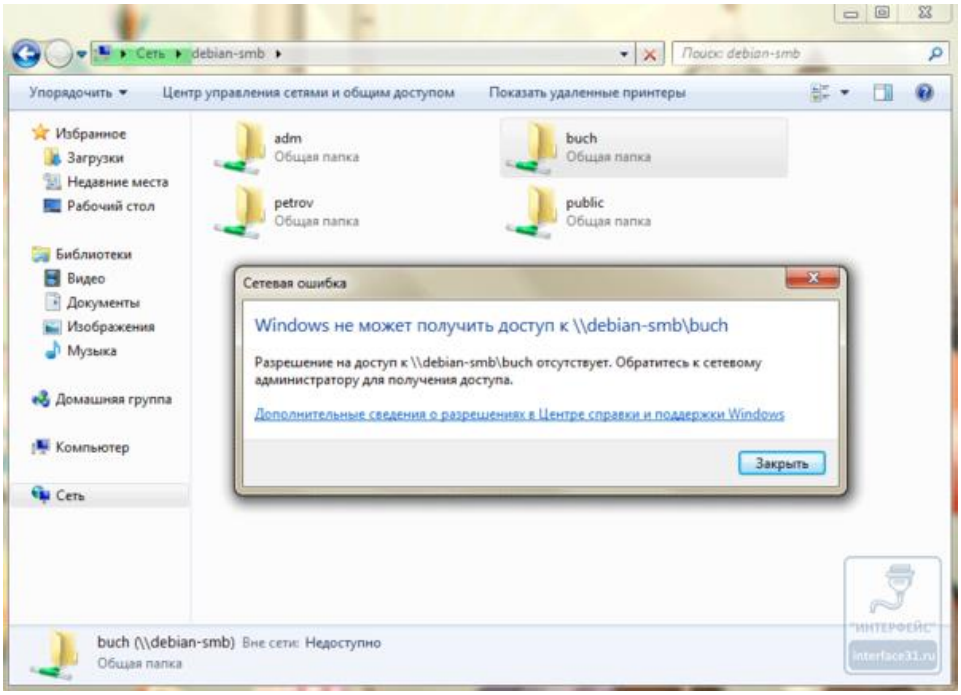


Рис. 261

Также обратите внимание, что несмотря на то, что общий ресурс с именем пользователя создан, доступ он к нему получить не сможет, так как физически его домашняя директория не существует.

4. Настройка корзины для общего ресурса

Полезность корзины на файловом сервере, пожалуй, не будет отрицать никто. Человеку свойственно ошибаться и будет очень обидно, если ценой ошибки окажется несколько часов работы, но, к счастью Samba позволяет помещать удаленные файлы в корзину.

Для активации корзины добавьте в секцию к общему ресурсу следующие строки:

```
vfs objects = recycle
recycle:repository = .recycle
recycle:versions = yes
recycle:keeptree = yes
```

Первая опция добавит в общий ресурс новый объект - корзину, вторая укажет ее расположение - скрытая папка в корне. Две следующих включают сохранение структуры папок при удалении и сохранение нескольких версий файла с одним и тем же именем. Это нужно в тех случаях, когда разные пользователи удалят разные файлы с одним и тем же именем.

Перезапустим Samba и попробуем что-нибудь удалить.

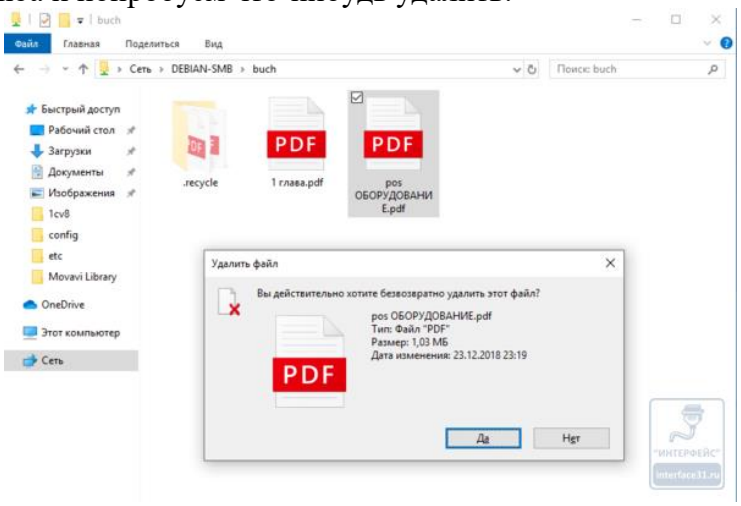


Рис. 262

Несмотря на грозное предупреждение Проводника удаляемые файлы перемещаются в корзину, откуда мы их можем восстановить.
Как видим, работать с Samba не просто, а очень просто.

Сделайте скриншоты (фотографии) процесса настройки файловых серверов и вставьте в отчёт.

2.19 Практическая работа№ 19
Настройка контейнеров Docker

Задание:

1. Установка Docker

Для установки обновим индекс пакетов и установим необходимые зависимости:
root@dedicated:~# sudo apt update
root@dedicated:~# sudo apt install apt-transport-https ca-certificates curl gnupg2 software-properties-common
Подключим репозиторий Docker, предварительно добавив GPG-ключ, и обновим индексы:
root@dedicated:~# curl -fsSL https://download.docker.com/linux/debian/gpg | sudo apt-key add -
root@dedicated:~# sudo add-apt-repository "deb [arch=amd64] https://download.docker.com/linux/debian $(lsb_release -cs) stable"
root@dedicated:~# sudo apt update
По умолчанию, вы сейчас подключены в репозиторий Debian, для дальнейшей установки необходимо переключиться в репозиторий Docker с помощью команды:
root@dedicated:~#apt-cache policy docker-ce
В терминале вы должны увидеть следующее:
После чего можно приступить к установке Docker:
root@dedicated:~# sudo apt install docker-ce
По окончании установки, добавьте его в автозагрузку и проверьте статус:
root@dedicated:~#sudo systemctl enable docker
root@dedicated:~# sudo systemctl status docker
В окне терминала мы должны увидеть следующее:

```
root@dedicated:~# sudo systemctl enable docker
Synchronizing state of docker.service with SysV service script with /lib/systemd/systemd-sysv-install.
Executing: /lib/systemd/systemd-sysv-install enable docker
root@dedicated:~#
root@dedicated:~# sudo systemctl status docker
● docker.service - Docker Application Container Engine
   Loaded: loaded (/lib/systemd/system/docker.service; enabled; vendor preset: enabled)
   Active: active (running) since Wed 2020-01-22 16:09:36 EET; 7min ago
     Docs: https://docs.docker.com
 Main PID: 3304 (dockerd)
    Tasks: 8
   Memory: 44.5M
   CGroup: /system.slice/docker.service
           └─3304 /usr/bin/dockerd -H fd:// --containerd=/run/containerd/containerd.sock

Jan 22 16:09:35 dedicated dockerd[3304]: time="2020-01-22T16:09:35.173955365+02:00" level=warning msg="Your kernel does not supp
Jan 22 16:09:35 dedicated dockerd[3304]: time="2020-01-22T16:09:35.174303055+02:00" level=warning msg="Your kernel does not supp
Jan 22 16:09:35 dedicated dockerd[3304]: time="2020-01-22T16:09:35.174573646+02:00" level=warning msg="Your kernel does not supp
Jan 22 16:09:35 dedicated dockerd[3304]: time="2020-01-22T16:09:35.175070036+02:00" level=info msg="Loading containers: start."
Jan 22 16:09:35 dedicated dockerd[3304]: time="2020-01-22T16:09:35.644387792+02:00" level=info msg="Default bridge (docker0) is
Jan 22 16:09:35 dedicated dockerd[3304]: time="2020-01-22T16:09:35.786353761+02:00" level=info msg="Loading containers: done."
Jan 22 16:09:36 dedicated dockerd[3304]: time="2020-01-22T16:09:36.098174605+02:00" level=info msg="Docker daemon" commit=633a0e
Jan 22 16:09:36 dedicated dockerd[3304]: time="2020-01-22T16:09:36.100594609+02:00" level=info msg="Daemon has completed initial
Jan 22 16:09:36 dedicated systemd[1]: Started Docker Application Container Engine.
Jan 22 16:09:36 dedicated dockerd[3304]: time="2020-01-22T16:09:36.195207838+02:00" level=info msg="API listen on /var/run/docke
lines 1-20/20 (END)
```

Рис. 263

На этом установка Docker завершена, и мы можем приступить к созданию и управлению контейнерами.

2. Команда Docker

Управление контейнерами осуществляется с помощью команды «docker», она имеет следующую структуру:
root@dedicated:~# docker [OPTIONS] COMMAND
Чтобы посмотреть полный список команд введите:
root@dedicated:~# docker
Мы увидим полный список команд в рассматриваемой версии Docker 19.03.5. Для примера, возьмем первую команду по списку («attach») и посмотрим более полную информацию о ней, доступные опции и аргументы:
root@dedicated:~# docker attach --help
Обратите внимание, без 'sudo', команды docker выполняются исключительно под пользователем root. Если в системе присутствуют другие учетные записи, которые работают с docker, не забываем про приставку 'sudo' к командам. Если Вы хотите работать с docker под другим пользователям, не используя постоянно 'sudo', необходимо создать группу docker, если она отсутствует, и добавить в неё нужного пользователя.
root@dedicated:~# sudo groupadd docker
root@dedicated:~# sudo gpasswd -a ${USER} docker
root@dedicated:~# sudo service docker restart
где *${USER}* - имя вашего пользователя. Так же добавьте пользователя в sudoers.
3. Образы контейнеров и управление контейнерами
По умолчанию, все доступные образы контейнеров хранятся в публичном docker-репозитории DockerHub. DockerHub представляет собой публичный реестр с доступными контейнерами, в нем размещены всевозможные образы, от отдельных компонентов до операционных систем. Запуск контейнера выполняется с помощью команды «run».
Чтобы ещё раз сделать акцент на мультиплатформенности, скачаем и запустим в нашем docker (который установлен на операционную систему Debian) контейнер с операционной системой CentOS:
root@dedicated:~# docker pull centos
root@dedicated:~# docker run -i -t centos
Команду 'run' мы запустили с аргументами, залогинившись непосредственно в shell-окружение контейнера f9b7becca15c. Находясь внутри контейнера, выполним три команды: узнаем версию нашей CentOS, отобразим список директорий внутри контейнера и запустим команду top, которая отобразит все запущенные процессы внутри контейнера:
[root@f9b7becca15c /]# cat /etc/system-release
[root@f9b7becca15c /]# ls -la
[root@f9b7becca15c /]# top
Внутри контейнера вы можете работать как в обычной терминальной среде. Командой «yum install mc» установим популярный файловый менеджер. Обратите на это внимание, к этому вопросу мы скоро вернемся.
В данном примере, командой pull мы загрузили последнюю, latest версию CentOS. Мы можем загружать с dockerhub любые другие версии, все что нам для этого нужно, это знать название репозитория и названия образа. Команда будет выглядеть так:
root@dedicated:~# docker pull [ОПЦИИ] [ПУТЬ/]ИМЯ_ОБРАЗА[:ТЕГ]
root@dedicated:~# docker pull centos:centos7.4.1708
Посмотреть список загруженных образов:
root@dedicated:~# docker images
Посмотреть запущенные контейнеры:
root@dedicated:~# docker container ls

Загрузим ещё несколько контейнеров. Они нам понадобятся для наглядности, в заключительной главе этой статьи.
root@dedicated:~# docker pull ubuntu
root@dedicated:~# docker pull ubuntu

Посмотреть список загруженных контейнеров, и список всех контейнеров (без опции '-a' команда отобразит список всех активных контейнеров) :
root@dedicated:~# docker ps -a
Вы, наверное, обратили внимание, что командой 'run', мы с образа CentOS сделали один контейнер, а в списке всех доступных их значится два. На этом моменте нужно понять особенность контейнеризации. Первоначальный образ контейнера CentOS (CONTAINER ID: f9b7becca15c) был пуст. Мы ранее установили в него файловый менеджер 'mc'. Но фактически, мы установили 'mc' не в этот контейнер, мы как бы наложили на него слой, создав на основе CentOS (CONTAINER ID: f9b7becca15c) дополнительный контейнер CentOS+mc (CONTAINER ID: 1d09ac8b4e79). У нас получилось два управляемых, автономных контейнера с разными container id.
Запустить контейнер можно по его container id:
root@dedicated:~# docker start 1d09ac8b4e79
Попасть внутрь контейнера можно с помощью команды 'exec':
root@dedicated:~# docker exec -it 1d09ac8b4e79 bash
Остановить или удалить контейнер можно так же по его container id, или по имени, которое видно на скриншоте выше. Команды для примера, контейнеры удалять пока не будем, они нам ещё будут нужны.
root@dedicated:~# docker stop 1d09ac8b4e79
или root@dedicated:~# docker stop keen_carson
root@dedicated:~# docker rm 1d09ac8b4e79
или root@dedicated:~# docker rm keen_carson

4. Dockerfile

В предыдущем разделе мы работали с уже готовыми образами Docker, которые мы загружали с DockerHub. В этом разделе рассмотрим процесс создания собственного сценария по созданию образа. Этот сценарий пишется в текстовом формате и называется Dockerfile, в нем вы описываете набор инструкций по созданию образа.
Создадим простейший файл Dockerfile:
root@dedicated:~# mkdir /opt/freehost-imag
root@dedicated:~# cd /opt/freehost-image
root@dedicated:/opt/freehost-image# nano Dockerfile
содержимое Dockerfile:
FROM debian:latest
MAINTAINER Dmitry Shestak <shestak@freehost.com.ua>
RUN apt-get update
RUN apt-get install -y nginx mc curl atop
EXPOSE 80
Сохраните файл, и находясь в каталоге с ним, запустите создание образа:
docker build . -t freehost-image
В консоли Вы увидите подробный вывод того, что происходит внутри контейнера. А происходит следующее: поле FROM указывает исходный образ операционной системы, на основе которого будет сформирован наш собственный образ. Поле MAINTAINER указывает автора образа. Поле RUN запускает оболочку командной строки, внутри которой система сначала обновит индекс пакетов, а затем установит перечисленное программное обеспечение.
Теперь посмотрим список доступных образов:
root@dedicated:~# docker images
В списке мы видим наш образ, который мы создали: freehost-image (id 21f738109e09). Выполним в него вход и запустим что-нибудь из программного обеспечения, которое мы перечислили в сценарии.
root@dedicated:~# docker run -it freehost-imag
root@5199cfa793af:/# atop

Сложность Dockerfile ограничена лишь вашими целями, мы рассмотрели самый простой пример. Со всеми доступными переменными и инструкциями, которые можно использовать при создании образа, вы можете ознакомиться в разделе официальной документации.

5. Установка Wordpress в Docker с помощью Docker-compose

Docker-compose это инструмент, который используют для запуска нескольких контейнеров. Он является своего рода сценарием, позволяющий описать взаимодействие нескольких контейнеров для работы одного сложного приложения. Например, если нам нужен только nginx, запустить его можно командой 'docker run nginx'. Если нам потребуется более сложное приложение, включающее в себя nginx+php+mysql, нам потребуется Docker-compose.

В этом примере мы создадим собственный docker-compose для сайта на популярном движке WordPress и запустим его.

Для начала установим приложение Docker-compose:

root@dedicated:~# curl -s https://api.github.com/repos/docker/compose/releases/latest | grep browser_download_url | grep docker-compose-Linux-x86_64 | cut -d '"' -f 4 | wget -qi -
root@dedicated:~# chmod +x docker-compose-Linux-x86_64
root@dedicated:~# sudo mv docker-compose-Linux-x86_64 /usr/local/bin/docker-compose
root@dedicated:~# docker-compose version
docker-compose version 1.25.2, build 698e2846
docker-py version: 4.1.0
CPython version: 3.7.5
OpenSSL version: OpenSSL 1.1.0l 10 Sep 2019

Создадим наш файл сценария

root@dedicated:~# mkdir /opt/wordpress && cd /opt/wordpress
root@dedicated:/opt/wordpress# mcedit docker-compose.yml

...и добавим в него содержимое:

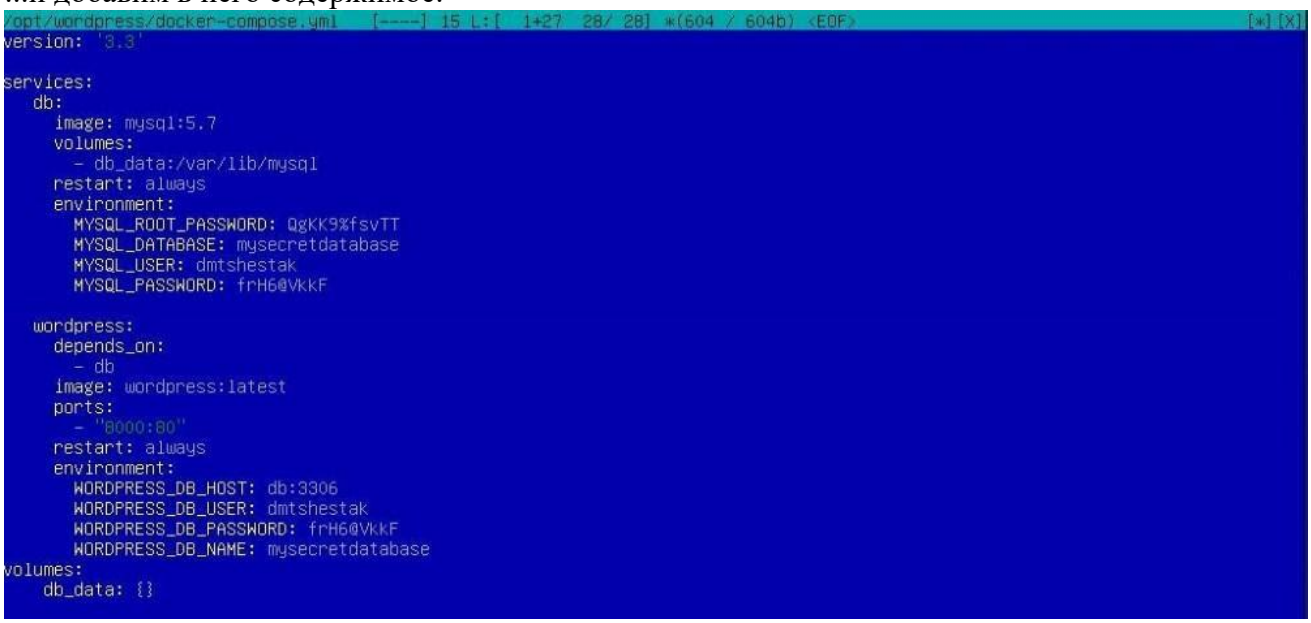


Рис. 264

Внимание! Файл в формате *.yml, который чувствителен к синтаксису. Отступы должны иметь чётное количество пробелов. Если это условие не будет соблюдено, он будет неработоспособен.

После чего, находясь в каталоге с docker-compose.yml выполните команду:

root@dedicated:~# docker-compose up -d

В консоли вы должны увидеть следующее:

Creating network "wordpress_default" with the default driver
Creating volume "wordpress_db_data" with default driver

Pulling db (mysql:5.7)...
5.7: Pulling from library/mysql
804555ee0376: Pull complete
c53bab458734: Pull complete

d054b015f084: Pull complete
Digest:
sha256:73e8d8adf491c7a358ff94c74c8ebe35cb5f8857e249eb8ce6062b8576a01465
Status: Downloaded newer image for wordpress:latest
Creating wordpress_db_1 ... done
Creating wordpress_wordpress_1 ... done

В сценарии нами били описаны сервисы которые нужно установить (mysql и wordpress), мы указали пароли и пробросили порт, по которому мы можем получить доступ. Если вы все сделали верно, то для того чтобы увидеть окно с первоначальной настройкой Wordpress, достаточно в оке браузера ввести: https://ваш_ip_адрес:8000

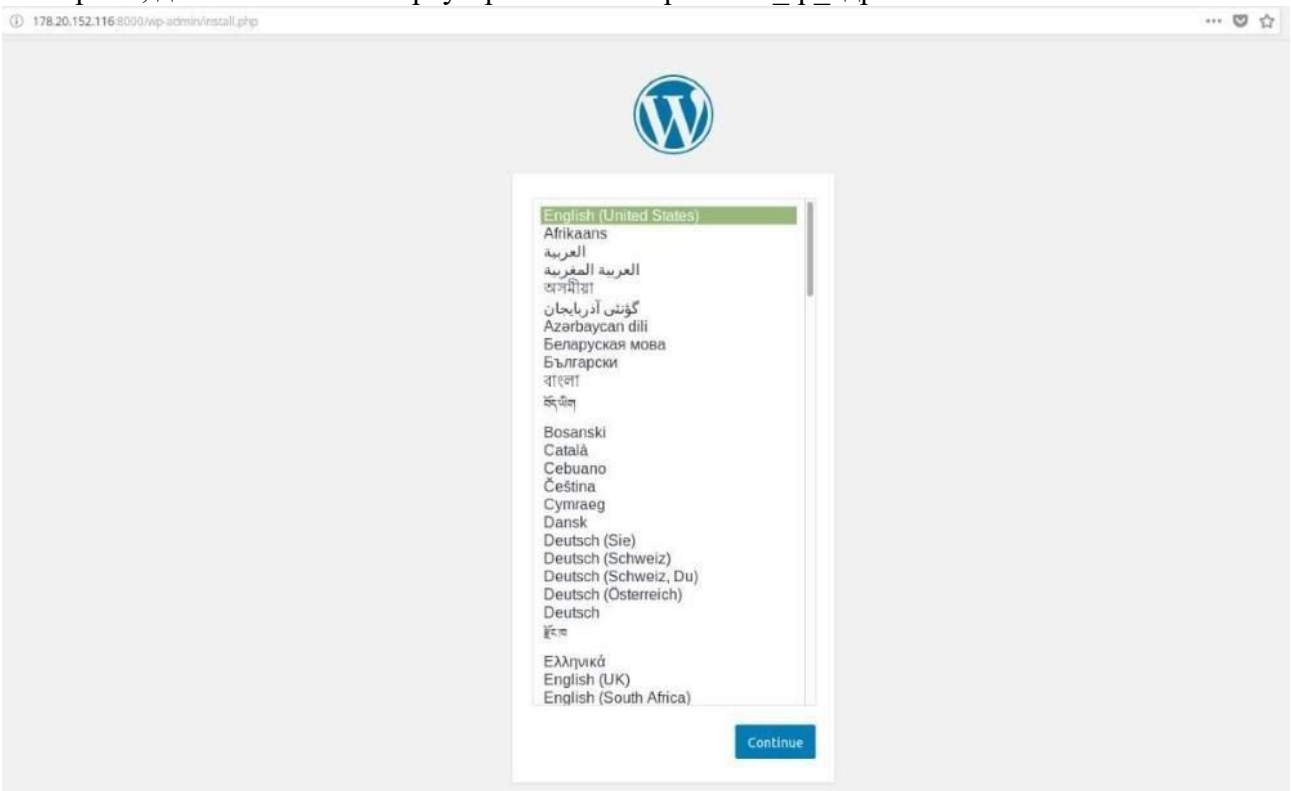


Рис. 265

Уже известной нам командой посмотрим список запущенных контейнеров:

6. Установка и управление Portainer

В предыдущих разделах мы научились загружать, создавать собственные контейнеры, а также комбинировать контейнеры в сложные приложения с помощью docker-compose. Эта глава посвящена удобному приложению для управления Docker хостами и контейнерами, имеющее на вооружении множество возможностей. Вот некоторые из них:

- Web-панель. Управление осуществляется в окне браузера с помощью web-интерфейса.
- Управление сервисами. Portainer позволяет в графической оболочке указать реестр, с которого будет загружен контейнер, название образа, даст возможность пробросить необходимые порты, настроить сеть, указать рабочие директории, подключить диски.

- Управление контейнерами и сбор статистики. Вы сможете анализировать логи каждого отдельного контейнера, собирать статистику по используемым ресурсам, работать в контейнере подключившись к нему через bash в интерактивном окне.
- Кластеризация. Portainer поддерживает кластеризацию Docker Swarm.

Установить его не сложно, так как Portainer сам представлен в виде контейнера.
Первым делом создадим каталог, где будут размещены данные:
root@dedicated:~# mkdir /root/portainer/data
Запустим контейнер следующей командой:
root@dedicated:~# docker run --name portainer --restart always -d -p 9000:9000 -v /root/portainer/data:/data -v /var/run/docker.sock:/var/run/docker.sock portainer/portainer
Так же мы можем установить Portainer через уже знакомый нам Docker-compose. Содержание файла будет следующее:

```
portainer:
    image: portainer/portainer
    container_name: portainer
    hostname: portainer
    restart: always
    command: --no-auth --no-analytics
    volumes:
        - /var/run/docker.sock:/var/run/docker.sock
    ports:
        - "9000:9000"
```

По окончании установки, web-интерфейс Portainer будет доступен по ссылке: https://ваш_ip_адрес:9000
Первым делом вы увидите окно регистрации, с предложением ввести пароль администратора. Введите сложный пароль и подтвердите его.
Перед вами появится с окно с выбором окружения, Local или Remote. Так как Docker у нас установлен локально, выбираем Local и подключаемся нажатием Connect. После чего вы увидите рабочее окружение Portainer.
На первый взгляд UI Portainer может показаться перегруженным и запутанным, но это только на первый взгляд. Меню и функционал интуитивно понятен. Предлагаем вам самостоятельно пройтись по всевозможным меню и вкладкам.
На следующем скриншоте, в разделе Containers мы можем увидеть все контейнеры, которые мы создали ранее в этой статье:

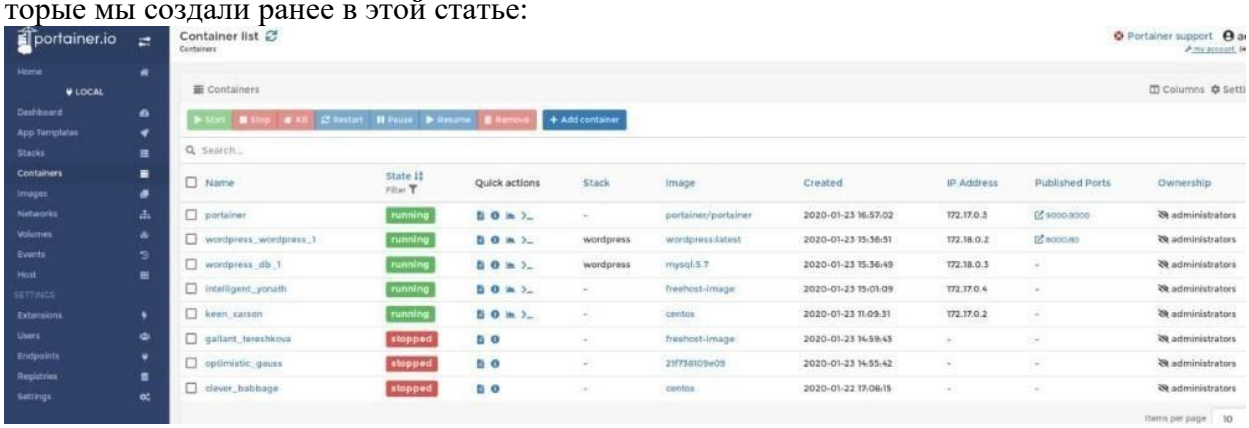


Рис. 266

Portainer достаточно хорошо документирован. Мы не видим смысла перегружать статью графическим контентом и скриншотами, все это доступно и подробно описано в официальном руководстве, с которым рекомендуем ознакомиться для решения более сложных задач.

Сделайте скриншоты (фотографии) процесса настройки контейнеров Dockeru вставьте в отчёт.

2.20 Практическая работа№ 20
Установка сервера CentOS. Настройка web-сервера в CentOS

Задание 1:

1. Установка CentOS 8 с помощью ISO-образа

В рекомендуемых требования указано, что для установки CentOS 8 необходимо минимум 10 Гб места на диске и 512 Мб RAM на одно ядро процессора.
Первым шагом, вам будет предложено выбрать дальнейшие действия. Так как вы выполняете установку, нам интересен первый пункт меню:

```
Install CentOS Linux 8.0.1905
```

Рис. 267

Выбрав его, у вас запустится процесс установки:

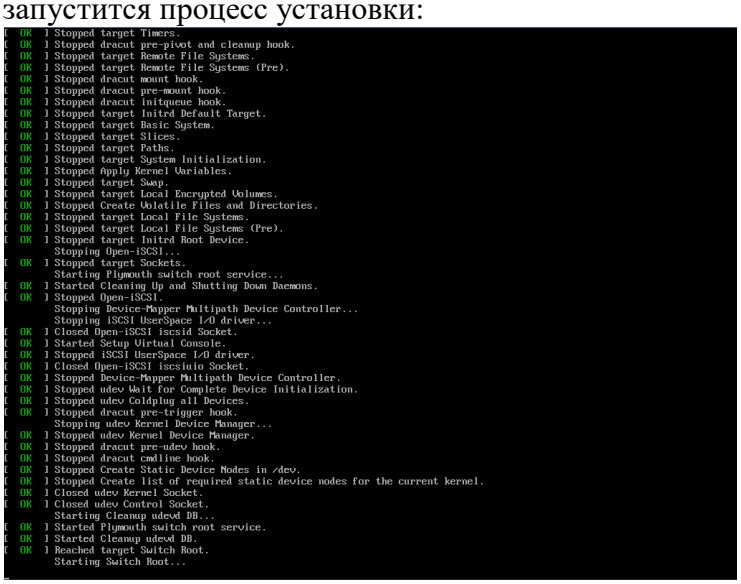


Рис. 268

В процессе пока все необходимое ПО загружается, можно просто наблюдать за процессом, от вас не требуется никаких действий.

И когда черный экран сменится на интерактивный с логотипом **CentOS**, пора брать в руки мышь и клавиатуру.
В приветствии, система попросит вас выбрать язык.
Нажмите кнопку **“Continue”**. В следующем меню нужно выбрать основные настройки для установки CentOS.
Для запуска установки, обязательно настроить только один пункт **“Installation Destination”**, там вы указываете разбивку диска, но давайте сразу настроим сеть и дату со временем.
В зависимости от вашего часового пояса, вы устанавливаете свои параметры, для нас это Москва.
Чтобы настроить сеть, переходим в пункт **“Network & Host Name”**
В поле **“Host Name”** указываем имя сервера и для конфигурации сетевых интерфейсов нажимаем **“Configure”**
В главной вкладке, нужно отметить галочкой **“Automatically connect to this network when it is available”**, это нужно для того, чтобы сетевой интерфейс поднимался автоматически.
Перейдите во вкладку **“IPv4 Settings”** (либо **IPv6** если вы используете данный протокол) настройте **IP**-адрес, маску подсети и шлюз.
Для ввода конкретного **IP (192.168.1.N, где N – номер вашего пользователя)**, нужно выбрать метод **“Manual”** и нажать кнопку **“Add”**, после чего у вас появится возможность ввести нужные данные. Сохранив все, мы вернемся к начальному окну настроек сети.
Как можно увидеть на скриншоте, **IP** адрес добавился и сетевой интерфейс уже поднят (состояние Connected).
Следующим шагом мы перейдем к разбивке диска:

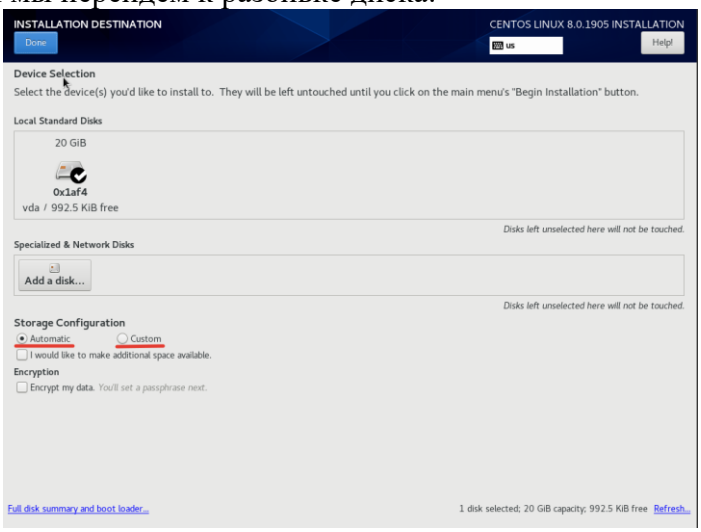


Рис. 269

Для примера установки воспользуемся автоматической разбивкой от разработчиков **CentOS**. Если вы хотите разбить диск по-своему, нужно выбрать пункт **“Custom”.**
Если контроллер вашего сервера не поддерживает аппаратный RAID, в этом пункте меню вы можете собрать программный RAID из дисков сервера на базе mdadm.
Список пакетов для установки выбирается в пункте Software Selection. Если вы планируете использовать CentOS 8 в качестве сервера, достаточно выбрать Minimal Install, а из добавлений Standard и Guest Agents (если вы ставите гостевую ОС в виртуальной машине).
После вышеописанных действий, можно запускать установку кнопкой **“Begin Installation”**

В процессе уже самой распаковки и установки необходимых компонентов, вам потребуется указать пароль для **root**-пользователя и можно создать дополнительного пользователя, но это не обязательный пункт.
Нажмите на кнопку **“Root Password”**, введите и повторите пароль и нажмите **“Done”**, чтобы вернуться к установке:

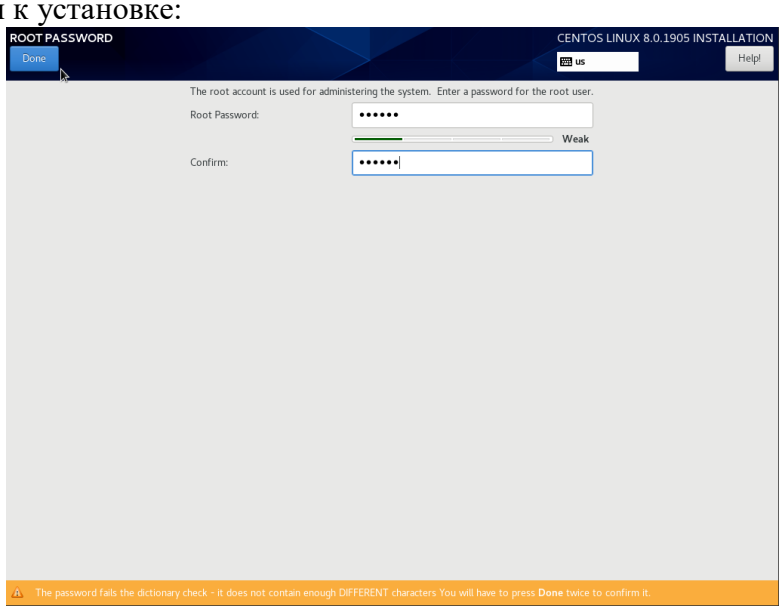


Рис. 270

Изначально предлагаю установить простой пароль, чтобы в случае проблем с сетью, вы могли легко его вспомнить и исправить проблемы. После того, как система будет установлена, пароль рекомендуется изменить на более сложный.

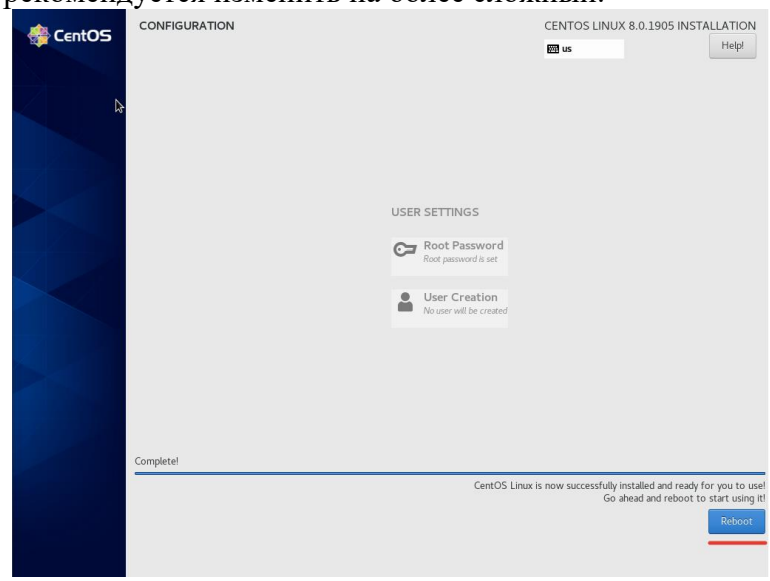


Рис. 271

На этом установка **CentOS 8** завершена.

2. Базовая настройка CentOS 8 после установки

Базовая настройка **CentOS 8** практически не отличается от настройки CentOS, я делаю базовые настройки одинаковые на всех серверах.

1. CentOS 8: Установка обновлений и инструментов администратора

В **CentOS 8** на замену **yum**, пришел **dnf**.

Dnf – это следующее поколение приложения **YUM**, менеджер пакетов для дистрибутивов Linux на основе **RPM**-пакетов. Ранее **dnf** использовался в дистрибутивах **Fedora**, а теперь и в **CentOS 8**.

Первое действие, которое необходимо выполнить на вновь установленном сервере, это обновление системы:

dnf update -y
Если образ свежий, то скорее всего у вас не будет пакетов для обновлений:
[root@centos var]# dnf update

```
Last metadata expiration check: 0:21:47 ago on Wed 09 Oct 2020 02:36:45 PM +06.
Dependencies resolved.
Nothing to do.
Complete!
```

Если у вас обнаружатся обновления, обязательно их установите.
Следующим шагом, подключим репозиторий **EPEL** и установим необходимые утилиты для удобного администарирования сервера:
dnf install epel-release –y
dnf install mc wget screen nano net-tools bind-utils curl lsof vim -y
Для комфортного администрирования хватает этого набора утилит, вы можете установить свои привычные утилиты.
Автоматическое обновление системы не включаем, так как всегда требуется устанавливать необходимые обновления вручную. Если вы хотите настроить автоматическое обновление, установите пакет **dnf-automatic**:
dnf install -y dnf-automatic
Чтобы проверить активные задания на обновления системы введите:
systemctl list-timers *dnf-*
2. Отключение SELinux
На начальном этапе необходимо отключить **SELinux** (для применения изменения нужно перезагрузить сервер):
nano /etc/sysconfig/selinux
reboot
Отключение **SELinux** налету, можно выполнить командой:
setenforce 0
3. Настройка сети в CentOS 8
Так как сеть мы настроили на этапе установки системы, настройка ее в данный момент не требуется. Необходимо добавить, что в **CentOS 8**, сеть управляется только через **Network Manager и утилиту** nmcl. **Network-scripts** по умолчанию не поддерживаются.
Проверка статуса сети:
[root@server ~]# systemctl status NetworkManager

```
● NetworkManager.service - Network Manager
Loaded: loaded (/usr/lib/systemd/system/NetworkManager.service; enabled; vendor preset: enabled)
Active: active (running) since Mon 2019-10-07 08:23:11 MSK; 3h 37min ago
Docs: man:NetworkManager(8)
Main PID: 870 (NetworkManager)
Tasks: 3 (limit: 5060)
Memory: 4.7M
CGroup: /system.slice/NetworkManager.service
└─870 /usr/sbin/NetworkManager --no-daemon
```

4. Установка и смена hostname
Если вы не задали корректный **hostname** сервера при установке или просто хотите изменить, это можно выполнить несколькими способами. Измените его в файле **/etc/hostname** или поменяйте с помощью команды:
hostnamectl set-hostname нужный_хостнейм
5. Настройка firewalld в CentOS 8

Добавим в доверенные зоны на **firewalld**, нужные для начальной работы сервисы (SSH и HTTP/HTTPS):

```
firewall-cmd –-add-service=ssh
firewall-cmd --permanent –-add-service=http
firewall-cmd --permanent –-add-service=https
```

6. Настройка времени и часового пояса (time-zone)

Чтобы посмотреть текущее время и time-zone, нужно ввести команду **date**:

```
[root@centos var]# date
Wed Oct 9 13:03:00 MSK 2020
```

Мы указали **time-zone** при установке самой системы, поэтому у нас время по Москве. Чтобы поменять **time-zone**, нужно воспользоваться соответствующей командой:

```
timedatectl set-timezone Europe/Moscow
```

Где вместо **Europe/Moscow** вы можете указать свой вариант, например:

```
[root@server network-scripts]# date
Mon Oct 7 12:46:09 MSK 2019
[root@server network-scripts]# timedatectl set-timezone Asia/Almaty
[root@server network-scripts]# date
Mon Oct 7 15:46:22 +06 2019
```

Для синхронизации времени используется **chronyd**, мы включим его и добавим в автозагрузку через systemctl:

```
dnf install chrony
systemctl enable chronyd
systemctl start chronyd
 [root@server network-scripts]# systemctl status chronyd
● chronyd.service - NTP client/server
Loaded: loaded (/usr/lib/systemd/system/chronyd.service; enabled; vendor preset: enabled)
Active: active (running) since Mon 2019-10-07 16:13:48 +06; 9s ago
Docs: man:chronyd(8)
man:chrony.conf(5)
Main PID: 31700 (chronyd)
Tasks: 1 (limit: 5060)
Memory: 1.1M
CGroup: /system.slice/chronyd.service
└─31700 /usr/sbin/chronyd
```

7. Настройка истории команда в bash_history

Для удобного просмотра истории, предлагаю добавить пару строк в **.bashrc**, чтобы в последствии можно было легко ориентироваться в отчетах.

При настройке по умолчанию, вывод **history** выглядит следующим образом:

```
[root@centos ~]# history
1 dnf repolist
2 dnf install epel-release
```

То есть мы видим, что выполнялось на сервере, но не видим время и точную дату, для нас это критично, так как доступ к серверам могут иметь несколько специалистов. Поэтому приведем **history** к приятному виду:

```
export HISTSIZE=10000
export HISTTIMEFORMAT="%h/%d/%y - %H:%M:%S "
```

Теперь при проверке **history,** мы видим точное время выполнения той или иной команды:

```
[root@centos ~]# history
```

```
1 Oct/07/19 - 16:16:29 dnf repolist
2 Oct/07/19 - 16:16:29 dnf install epel-release
```

8. Cockpit: Веб-интерфейс управления сервером в CentOS 8

В CentOS 8 предустановлен веб-интерфейс управления сервером cockpit. Он также управляется через systemctl. Вы можете запустить его и добавить в автозгрузку:

```
# systemctl enable cockpit.socket
# systemctl start cockpit.socket
```

По–умолчанию веб сервер Cockpit слушает на порту 9090. Добавьте этот порт в разрешенные:

```
# firewall-cmd --get-active-zones
# firewall-cmd --add-port=9090/tcp --zone=MY_ACTIVE_ZONE
# firewall-cmd --reload
```

Для доступа к веб-интерфейсу Cockpit, откройте в браузере URL адрес https://your-CentOS8-IP:9090 и авторизуйтесь.

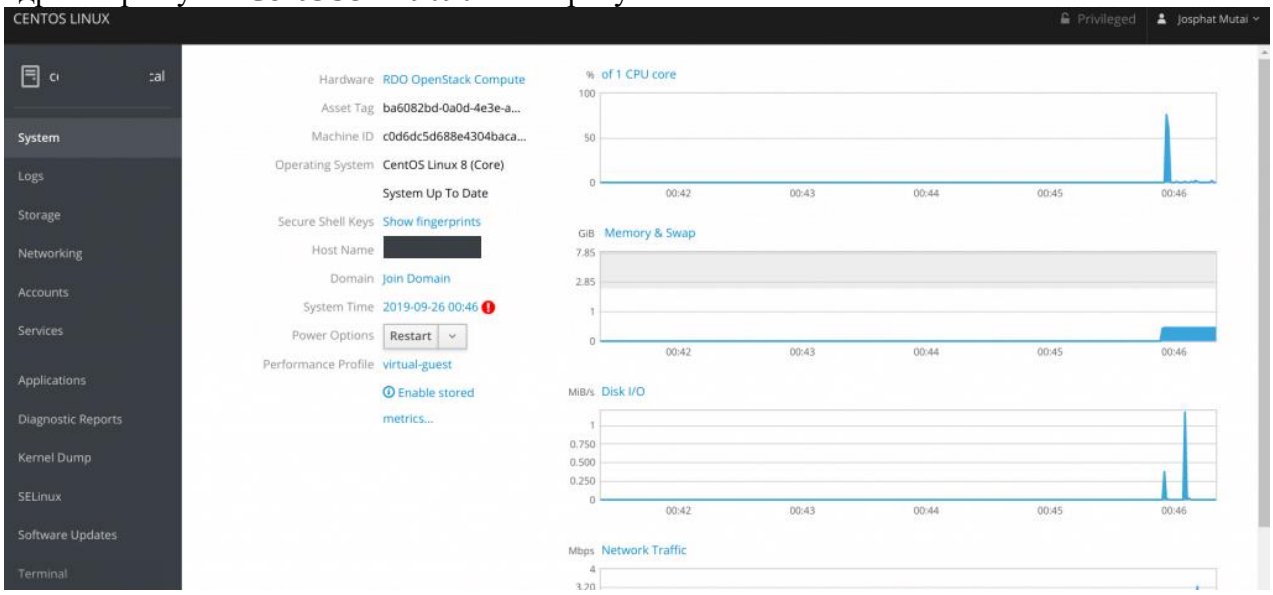


Рис. 272

С помощью веб-интерфейса Cockpit вы можете смотреть загрузку своего сервера, управлять сетями и хранилищами, контейнерами, смотреть логи.

На этом типовая настройка CentOS сервера закончена.

Сделайте скриншоты (фотографии) процесса установка сервера CentOS и вставьте в отчёт.

Задание 2:

1. Установка пакетов
1. Обновляем CentOS:

```
dnf update
```

2. Устанавливаем дополнительные пакеты для загрузки и распаковки:

```
dnf install unzip
```

3. Настройка безопасности

Открываем необходимые порты в брандмауэре:

```
firewall-cmd --permanent --add-port={80,443,8080}/tcp
firewall-cmd --permanent --add-port={20,21,60000-65535}/tcp
firewall-cmd --permanent --add-port={25,465,587}/tcp
firewall-cmd --reload
```

** 80, 443 и 8080 порты для веб-сервера; 20, 21 порты нужны для работы FTP;* 60000-65535 *также необходимы для работы FTP (динамические порты для пассивного режима); 25, 465 и 587 порты нужны для работы почтового сервера по SMTP; последняя команда перезапускает firewalld, чтобы применить новые правила.*

2. Установка NGINX
    1. Устанавливаем NGINX:

```
dnf install nginx
```

2. Внесем небольшую корректировку в файл nginx.conf:

```
nano /etc/nginx/nginx.conf
```

3. В секцию http добавим строку:

```
http {
   ...
   server_names_hash_bucket_size 64;
   ....
}
```

** на практике, может встретиться ошибка* ***could not build server_names_hash, you should increase server_names_hash_bucket_size: 32****. Она возникает при большом количестве виртуальных серверов или если один из них будет иметь длинное название. Данная строка в конфиге исправит ситуацию.*

4. Разрешаем автозапуск сервиса и запустим его:

```
systemctl enable nginx
systemctl start nginx
```

5. Проверим, что веб-сервер работает. Для этого открываем браузер на другом компьютере, который находится в одной сети и вводим в адресной строке IP-адрес сервера. В итоге мы должны увидеть заголовок «Welcome to nginx!»:

Рис. 273

** обратите внимание, что данное приветствие может иметь и другой вид.*

3. Установка PHP и PHP-FPM
    1. Устанавливаем PHP и php-fpm следующей командой:

```
dnf install php php-fpm
```

** В CentOS 8 будет установлена версия php 7.2 и выше*

2. Запускаем php-fpm и разрешаем его автозапуск:

```
systemctl enable php-fpm --now
```

4. Настройка связки NGINX + PHP
    1. Открываем файл для настройки виртуального домена по умолчанию:

```
nano /etc/nginx/nginx.conf
```

В секции **location** редактируем параметр **index** на следующее значение:

```
location / {
    index  index.php index.html index.htm;
}
```

** добавляем **index.php** в начало списка. Если параметра **index** нет, создаем его.*

2. Внутри секции **server** добавим следующее:

```
location ~ \.php$ {
    set $root_path /usr/share/nginx/html;
    fastcgi_pass unix:/run/php-fpm/www.sock;
    fastcgi_index index.php;
    fastcgi_param SCRIPT_FILENAME $root_path$fastcgi_script_name;
    include fastcgi_params;
    fastcgi_param DOCUMENT_ROOT $root_path;
}
```

** где **/usr/share/nginx/html** — корневой путь хранения скриптов; **unix:/run/php-fpm/www.sock** — файл для взаимодействия с php-fpm.*

3. Открываем настройки php-fpm:

```
nano /etc/php-fpm.d/www.conf
```

4. Проверяем, что параметр listen настроен так:

```
listen = /run/php-fpm/www.sock
```

... иначе, меняем значение. После перезагружаем php-fpm:

```
systemctl restart php-fpm
```

** в данном примере мы указываем, что php-fpm будет использовать сокетный файл **/run/php-fpm/www.sock** для взаимодействия. Этот файл мы указали выше в настройке NGINX.*

5. Проверяем правильность настроек nginx:

```
nginx -t
```

И перезагружаем его:

```
systemctl restart nginx
```

6. Создаем index.php в каталоге сайта по умолчанию со следующим содержимым:

```
nano /usr/share/nginx/html/index.php
```

```
<?php phpinfo(); ?>
```

Открываем в браузере IP-адрес нашего сервера. Теперь мы должны увидеть сводную информацию по PHP и его настройкам, например:

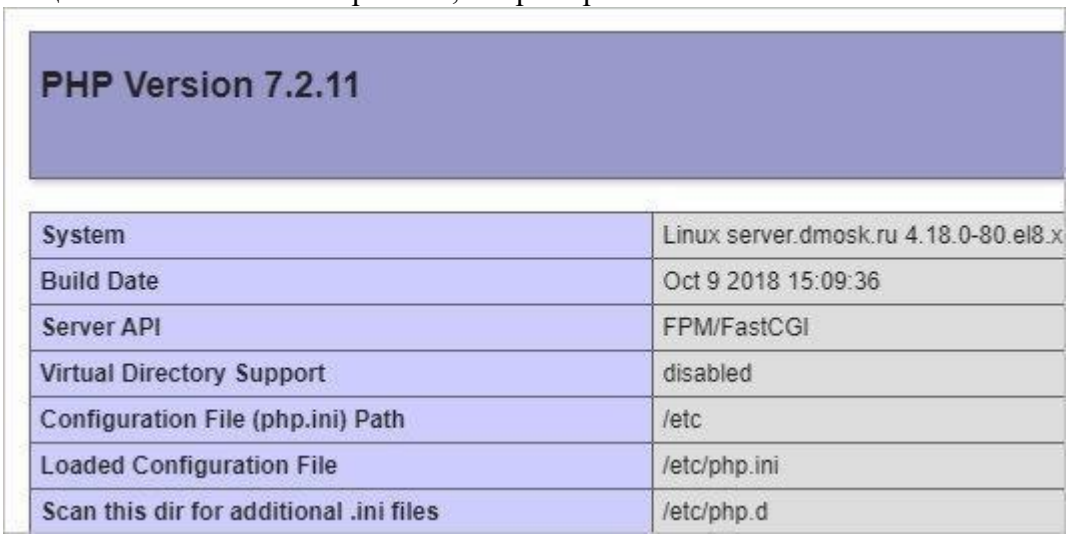


Рис. 274

1. Установка MariaDB или MySQL

1. Устанавливаем MariaDB следующей командой:

```
dnf install mariadb mariadb-server
```

** для установки mysql выполняем команду **dnf install mysql***

2. Разрешаем автозапуск и запускаем СУБД:

```
systemctl enable mariadb --now
```

** для работы с mysql меняем **mariadb** на **mysql**.*

3. Сразу создаем пароль для учетной записи root:

```
mysqladmin -u root password
```

2. PHP + MariaDB (MySQL)
    1. Для возможности подключаться к базе данных скриптами PHP необходимо установить следующие модули:

```
dnf install php-mysqli
```

2. Если мы установили php5, также ставим php-mysql:

```
dnf install php-mysql
```

После перезагружаем php-fpm:

```
systemctl restart php-fpm
```

3. Открываем наш сайт в браузере. В phpinfo появится новая секция MySQL:

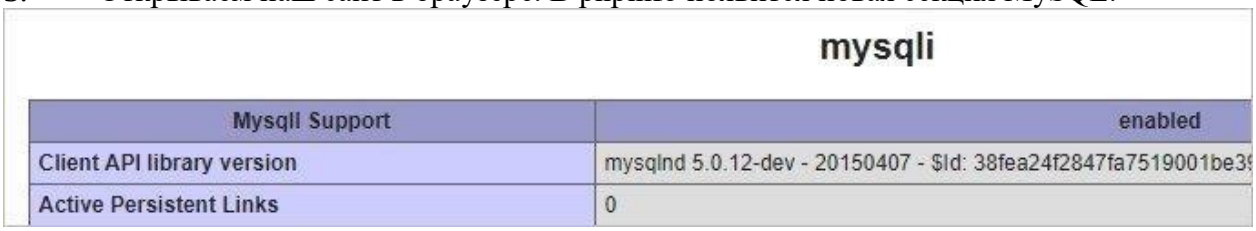


Рис. 275

** нас не должно смущать, что установили мы mariadb, а заголовок mysql. Если посмотреть в таблицу, можно увидеть ячейку **Client API version**, в которой указано, что используется именно mariadb.*

3. Установка phpMyAdmin

Переходим на сайт разработчика phpMyAdmin и копируем ссылку на нужную нам версию, например, последнюю:

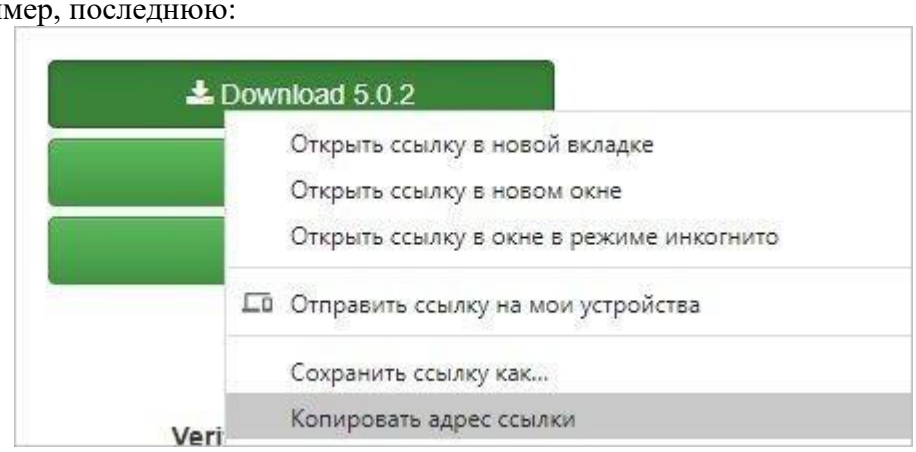


Рис. 276

Воспользовавшись скопированной ссылкой, скачиваем архив с установочными файлами:

```
wget https://files.phpmyadmin.net/phpMyAdmin/5.0.2/phpMyAdmin-5.0.2-all-languages.zip
```

Распаковываем скачанный архив:

```
unzip phpMyAdmin-*-all-languages.zip
```

Создаем каталог для phpmyadmin:

```
mkdir /usr/share/phpMyAdmin
```

... и переносим в него содержимое распакованного архива:

```
mv phpMyAdmin-*-all-languages/* /usr/share/phpMyAdmin/
```

Задаем владельца для каталога:

```
chown -R apache:apache /usr/share/phpMyAdmin
```

** как правило, сервис, которых обрабатываем php-запросы работает от пользователя **apache**.*

Устанавливаем модули php, необходимые для корректной работы phpMyAdmin:

```
dnf install php-json php-mbstring  php-mysqli
```

Внесем небольшую настройку в конфигурацию phpMyAdmin.
Сгенерируем случайную последовательность символов:

```
head /dev/urandom | tr -dc A-Za-z0-9 | head -c 32 ; echo ''
```

Откроем на редактирование или создадим файл:

```
nano /usr/share/phpMyAdmin/config.inc.php
```

Внесем в него строку:

```
<?php
...
$cfg['blowfish_secret'] = 'jd7n6yIcHOl55ikE7l5HAdNaWwunSHvR';

?>
```

** где jd7n6yIcHOl55ikE7l5HAdNaWwunSHvR — последовательность, которую нам выдала команда head **/dev/urandom ...**; Также обратите внимание на* **<?php ?>** — если мы создали новый файл, необходимо указать данные теги, так как они открывают код PHP. В противном случае, настройка не применится.
Теперь создадим для phpmyadmin отдельный виртуальный домен в NGINX:

```
nano /etc/nginx/conf.d/phpMyAdmin.conf
```

И добавим в него следующее содержимое:

```
server {
	listen	80;
	server_name  phpmyadmin.wbsh.local;
	set $root_path /usr/share/phpMyAdmin;

	location / {
		root $root_path;
		index index.php;
	}

	location ~ \.php$ {
		fastcgi_pass unix:/run/php-fpm/www.sock;
		fastcgi_index index.php;
		fastcgi_param SCRIPT_FILENAME $root_path$fastcgi_script_name;
		include fastcgi_params;
		fastcgi_param DOCUMENT_ROOT $root_path;
		fastcgi_read_timeout 300;
	}
}
```

** где **phpmyadmin.wbsh.local** — адрес для виртуального домена, именно этот адрес должен быть введен в адресную строку браузера, чтобы открылся нужный сайт. Поэтому если нет возможности зарегистрировать домен и имя узла в DNS, можно воспользоваться локальным файлом hosts. **/usr/share/phpMyAdmin** — это каталог, в который по умолчанию устанавливается phpMyAdmin.*
После перезапускаем NGINX:

```
systemctl reload nginx
```

Также нужно перезапустить php-fpm, так как в процессе установки были добавлены модули для PHP:

```
systemctl restart php-fpm
```

И открываем в браузере наш домен, в данном примере, http://phpmyadmin.wbsh.local. Откроется форма для авторизации — вводим логин root и пароль, который мы указали после установки и запуска mariadb.

4. Установка Memcached

Первым этапом мы установим и настроим сервис memcached.

Вторым — модуль php-memcached.

1. Сервис memcached

Выполняем установку пакетов:

```
dnf install memcached libmemcached
```

Создаем или открываем на редактирование конфигурационный файл для запуска сервиса:

```
nano /etc/sysconfig/memcached
```

Приводим его к виду:

```
PORT="11211"
USER="memcached"
MAXCONN="1024"
CACHESIZE="512"
OPTIONS="-l 127.0.0.1 -U 0"
```

** где **PORT** указываем на каком порту будет слушать сервис кэширования; **USER** — пользователь, под которым должен запускаться сервис; **MAXCONN** — максимальное число одновременных подключений; **CACHESIZE** — размер под кэш в мегабайтах; **OPTIONS** — параметры запуска (в данном примере наш сервис будет принимать запросы только с адреса локальной петли).*

После разрешаем автозапуск и запускаем сервис кэширования:

```
systemctl enable memcached --now
```

2. Модуль для php

Переходим на страницу загрузки memcached сайта pecl.php.net и копируем ссылку на стабильную версию memcached:

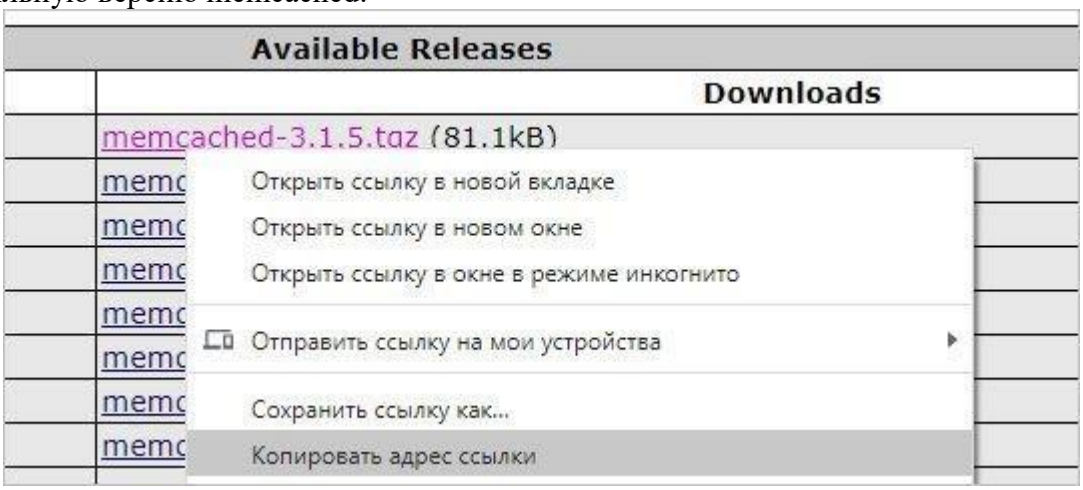


Рис. 277

Обратите внимание, что у каждой версии пакета есть свои требования к версии PHP. Внимательно изучаем, подойдет ли версия php-memcached для нашего сервера.

Скачиваем архив, ссылку на который мы скопировали:

```
wget http://pecl.php.net/get/memcached-3.1.5.tgz
```

Устанавливаем пакеты, необходимые для сборки php-pecl-memcached:

```
dnf install php-devel zlib-devel make
dnf --enablerepo=PowerTools install libmemcached-devel
```

Распаковываем скачанный архив:

```
tar -xvzf memcached-*.tgz
```

Переходим в распакованный каталог:

```
cd memcached-*/
```

Запускаем компиляцию php-расширения:

```
phpize
```

Конфигурируем исходник:

```
./configure
```

Собираем расширение:

```
make
```

Копируем созданный модуль в каталог php-модулей:

```
cp modules/memcached.so /usr/lib64/php/modules/
```

Создаем конфигурационной файл для подключения расширения:

```
nano /etc/php.d/20-memcached.ini
```

```
extension=memcached.so
```

После установки модуля перезапускаем php-fpm:

```
systemctl restart php-fpm
```

Чтобы проверить, что модуль memcached работаем, открываем наш сайт в браузере — в phpinfo должна появиться новая секция:

memcached

memcached support	enabled
Version	3.1.5
libmemcached version	1.0.18
SASL support	yes

... или вводим команду:

```
php -m | grep memcached
```

Мы должны получить:

```
memcached
```

3. Установка и настройка FTP-сервера

В качестве FTP-сервера будем использовать ProFTPd, так как он позволяет авторизовываться под uid системных учетных записей.

ProFTPd можно устанавливать командой:

```
dnf install proftpd
```

Загружаем скрипт ftpasswd:

```
wget http://www.castaglia.org/proftpd/contrib/ftpasswd -P /etc/proftpd
```

Разрешаем запуск на выполнение скрипта:

```
chmod +x /etc/proftpd/ftpasswd
```

Создаем виртуального пользователя:

```
/etc/proftpd/ftpasswd --passwd --file=/etc/proftpd/ftpd.passwd --name=ftpwww --uid=48 --gid=48 --home=/var/www --shell=/sbin/nologin
```

** где*

- ***/etc/proftpd/ftpd.passwd*** *— путь до файла, в котором хранятся пользователи;*
- ***ftpwww*** *— имя пользователя (логин);*
- ***uid*** *и* ***gid*** *— идентификаторы пользователя и группы системной учетной записи (apache);*
- ***/var/www*** *— домашний каталог пользователя;*
- ***/sbin/nologin*** *— оболочка, запрещающая локальный вход пользователя в систему.*

Изменим права для созданного файла с паролями:

```
chmod 440 /etc/proftpd/ftpd.passwd
```

** в противном случае, при запуске proftpd мы получим ошибку «...fatal: AuthUserFile: unable to use /etc/proftpd.d/ftpd.passwd: Operation not permitted...»*

Открываем на редактирование конфигурационный файл proftpd:

```
nano /etc/proftpd.conf
```

И редактируем следующее (комментируем):

```
#AuthOrder ...
```

Создадим конфигурационный файл со своими настройками:

```
nano /etc/proftpd/conf.d/custom.conf
```

И добавим следующее:

```
UseIPv6 off
IdentLookups off
PassivePorts 60000 65535

RequireValidShell off
AuthUserFile /etc/proftpd/ftpd.passwd
AuthPAM off
LoadModule mod_auth_file.c
AuthOrder mod_auth_file.c
```

** где **60000 - 65535** — диапазон динамических портов для пассивного режима.*

Разрешаем автозапуск FTP-серверу и запускаем его:

```
systemctl enable proftpd --now
```

Пробуем подключиться к серверу, использую любые FTP-клиенты, например, FileZilla, Total Commander или тот же браузер.

Это базовая и самая простая настройка ProFTPd, но если необходимо настроить TLS или хранить виртуальных пользователей в базе MySQL, читайте подробнее инструкцию по настройке ProFTPd на CentOS.

4. Apache (httpd)

Несмотря на то, что мы установили и настроили PHP-FPM, Apache нам понадобится, как минимум, по двум причинам. Во-первых, многие сайты используют файл .htaccess, который читает Apache. Во-вторых, последний включает большое число модулей, которые может использовать портал.

И так, устанавливаем httpd:

```
dnf install httpd
```

Заходим в настройки:

```
nano /etc/httpd/conf/httpd.conf
```

И редактируем следующее:

```
Listen 8080
```

** наш веб-сервер будет слушать на порту **8080**, так как на 80 уже работает NGINX.*

```
<IfModule dir_module>
    DirectoryIndex index.php index.html
</IfModule>
```

** если не указан конкретный скрипт, сначала веб-сервер пытается найти и запустить **index.php**, затем **index.html***

Добавляем:

```
<Directory /var/www/*/www>
    AllowOverride All
    Options Indexes ExecCGI FollowSymLinks
    Require all granted
</Directory>
```

** где **Directory** — разрешенные каталоги для запуска из apache; **Options** — разрешенные опции; **Require** — с каких IP-адресов можно открывать сайты, определенные в данном каталоге. Итого, мы разрешаем все каталоги в **/var/www**, но только если следующий каталог будет **www**; разрешаем опции **Indexes** (возвращает список файлов, если нет индексного файла, например, index.php), **ExecCGI** (разрешены сценарии CGI), **FollowSymLinks** (включены символические ссылки в этом каталоге); доступ для данных каталогов разрешен со всех адресов (**all granted**).*

Проверяем синтаксис конфигурационного файла httpd:

```
apachectl configtest
```

И если получаем ответ:

```
Syntax OK
```

... разрешаем автозапуск и запускаем службу:

```
systemctl enable httpd
systemctl start httpd
```

Создаем php-файл со следующим содержимым:

```
nano /var/www/html/index.php
```

```
<?php phpinfo(); ?>
```

Открываем браузер и вводим в адресную строку IP-адрес нашего сервера и добавляем :8080 (http://<IP-адрес нашего сервера>:8080). Откроется привычная нам страница с информацией о PHP. В разделе «PHP Variables» мы должны увидеть Apache для опции $_SERVER['SERVER_SOFTWARE']:

Рис. 278

5. NGINX + Apache

Ранее нами была настроена связка nginx + php-fpm. Теперь проверяем совместную работу первого с apache.

Открываем конфигурационный файл nginx:

```
nano /etc/nginx/nginx.conf
```

Находим наш настроенный location для php-fpm:

```
...
   location ~ \.php$ {
         set $root_path /usr/share/nginx/html;
         fastcgi_pass unix:/run/php-fpm/www.sock;
         fastcgi_index index.php;
         fastcgi_param SCRIPT_FILENAME $root_path$fastcgi_script_name;
         include fastcgi_params;
         fastcgi_param DOCUMENT_ROOT $root_path;
   }
...
```

и меняем на:

```
...
   location ~ \.php$ {
      proxy_pass http://127.0.0.1:8080;
      proxy_redirect     off;
      proxy_set_header   Host             $host;
      proxy_set_header   X-Forwarded-Proto $scheme;
      proxy_set_header   X-Real-IP        $remote_addr;
      proxy_set_header   X-Forwarded-For  $proxy_add_x_forwarded_for;
   }
...
```

Проверяем, есть ли файл:

```
nano /etc/nginx/default.d/php.conf
```

... и если есть, комментируем его содержимое:

```
#index index.php index.html index.htm;
#
#location ~ \.(php|phar)(/.*)?$ {
#    fastcgi_split_path_info ^(.+\.(?:php|phar))(/.*)$;
#
```

```
#    fastcgi_intercept_errors on;
#    fastcgi_index  index.php;
#    include        fastcgi_params;
#    fastcgi_param  SCRIPT_FILENAME  $document_root$fastcgi_script_name;
#    fastcgi_param  PATH_INFO $fastcgi_path_info;
#    fastcgi_pass   php-fpm;
#}
```

** в данном примере мы отключили обработку всех php-файлов с помощью php-fpm, так как это у нас должен делать apache.*

Проверяем и перезапускаем nginx:

```
nginx -t
systemctl restart nginx
```

Пробуем снова открыть в браузере адрес http://<IP-адрес нашего сервера> (уже без 8080) — должна открыться та же страница, что при проверке Apache (с добавлением 8080):

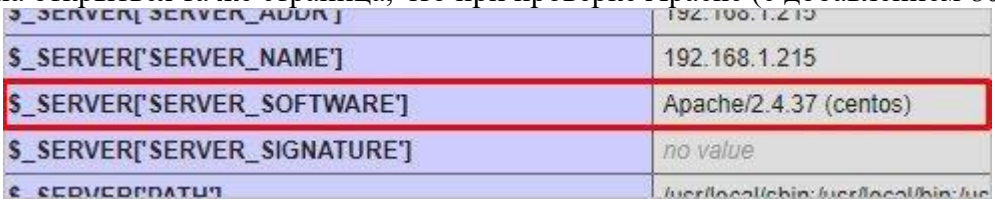


Рис. 279

6. Apache Real IP

Так как все запросы на httpd приходят от NGINX, они воспринимаются как от IP-адреса 127.0.0.1. На практике, это может привести к проблемам, так как некоторым сайтам необходимы реальные адреса посетителей.

Для решения проблемы будем использовать модуль mod_rpaf. Устанавливаем набор разработчика для apache:

```
dnf install httpd-devel gcc unzip redhat-rpm-config
```

Переходим в каталог /usr/local/src:

```
cd /usr/local/src
```

Скачиваем модуль:

```
wget https://github.com/gnif/mod_rpaf/archive/stable.zip
```

Распаковываем его:

```
unzip stable.zip
```

Переходим в распакованный каталог:

```
cd mod_rpaf-stable/
```

Собираем модуль и устанавливаем его:

```
make
make install
```

** при возникновении ошибки **./apxs.sh: line 15: -c: command not found**, необходимо поставить **which** командой **dnf install which**.*

Создаем конфигурационный файл со следующим содержимым:

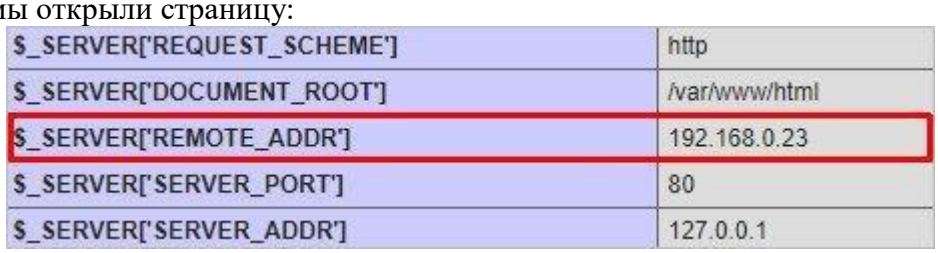
```
nano /etc/httpd/conf.d/mod_rpaf.conf
```

```
LoadModule             rpaf_module modules/mod_rpaf.so
RPAF_Enable            On
RPAF_ProxyIPs          127.0.0.1
RPAF_SetHostName       On
RPAF_SetHTTPS          On
RPAF_SetPort           On
RPAF_ForbidIfNotProxy  Off
```

Перезапускаем httpd:

```
systemctl restart httpd
```

Для проверки открываем нашу страницу с phpinfo и находим $_SERVER['REMOTE_ADDR'] — его значение должно быть равно адресу компьютера, с которого мы открыли страницу:



$_SERVER['REQUEST_SCHEME']	http
$_SERVER['DOCUMENT_ROOT']	/var/www/html
$_SERVER['REMOTE_ADDR']	192.168.0.23
$_SERVER['SERVER_PORT']	80
$_SERVER['SERVER_ADDR']	127.0.0.1

Рис. 280

7. Postfix

Устанавливаем postfix командой:

```
dnf install postfix
```

Теперь нам необходимо сделать несколько простых настроек:

```
nano /etc/postfix/main.cf
```

Редактируем:

```
...
myorigin = $mydomain
...
inet_protocols = ipv4
...
```

Добавляем:

```
smtp_generic_maps = hash:/etc/postfix/generic_map
```

** **myorigin** — имя домена, которое будет подставляться всем отправляемым сообщениям без явного указания оного; **inet_protocols** — задает версию IP, с которой будет работать Postfix (если на нашем сервере используется ipv6, значение параметра стоит оставить **all**); **smtp_generic_maps** указывает на карту с общими правилами пересылки.*

Открываем карту пересылки:

```
nano /etc/postfix/generic_map
```

И добавляем:

```
@wbsh.local    no-reply@wbsh.local
```

** данной настройкой мы будем подставлять всем отправляемым письмам без поля FROM адрес **no-reply@wbsh.local**.*

Создаем карту:

```
postmap /etc/postfix/generic_map
```

Для применения настроек перезагружаем почтовый сервер:

```
systemctl restart postfix
```

5. Тюнинг веб-сервера

1. PHP

Открываем на редактирование следующий файл:

```
nano /etc/php.ini
```

И правим следующее:

```
upload_max_filesize = 512M
...
post_max_size = 512M
...
short_open_tag = On
...
date.timezone = "Europe/Moscow"
```

Перезапускаем php-fpm и httpd:

```
systemctl restart php-fpm
```

```
systemctl restart httpd
```

2. NGINX

Открываем на редактирование следующий файл:

```
nano /etc/nginx/nginx.conf
```

И внутри секции http добавляем:

```
client_max_body_size 512M;
```

После перезапускаем nginx:

```
systemctl restart nginx
```

3. Создание первого сайта

Задаем переменную, значение которой будет домен сайта:

```
TMP_SITE=site1
```

** где **site1** — имя домена. Нам будет намного удобнее копировать и вставлять команды с переменной (не придется править после копипасты).*

Создаем новый файл виртуального домена NGINX:

```
nano /etc/nginx/conf.d/$TMP_SITE.conf
```

** обязательно на конце должен быть **.conf**, так как только такие файлы веб-сервер подгружает в конфигурацию.*

И добавляем следующее содержимое.

Для HTTP:

```
server {
   listen       80;
   server_name  site1.local www.site1.local;
   set $root_path /var/www/site1/www;

   access_log /var/www/site1/log/nginx/access_log;
   error_log /var/www/site1/log/nginx/error_log;

   gzip  on;
   gzip_disable "msie6";
   gzip_min_length 1000;
   gzip_vary on;
   gzip_proxied    expired no-cache no-store private auth;
   gzip_types      text/plain text/css application/json application/x-javascript text/xml applica-
tion/xml application/xml+rss text/javascript application/javascript;

   root   $root_path;

   location / {
      proxy_pass http://127.0.0.1:8080/;
      proxy_redirect     off;
      proxy_set_header   Host             $host;
      proxy_set_header   X-Forwarded-Proto $scheme;
      proxy_set_header   X-Real-IP        $remote_addr;
      proxy_set_header   X-Forwarded-For  $proxy_add_x_forwarded_for;
   }

   location ~*
^.+\.(jpg|jpeg|gif|png|css|zip|tgz|gz|rar|bz2|doc|docx|xls|xlsx|exe|pdf|ppt|tar|wav|bmp|rtf|js)$ {
         expires modified +1w;
   }
}
```

** где **site1.local** — домен, для которого создается виртуальный домен; **/var/www/site1** — каталог, в котором будет размещаться сайт.*
*** все запросы будут переводиться на локальный сервер, порт 8080, на котором работает apache, кроме обращений к статическим файла (jpg, png, css и так далее).*
**** обратите внимание на выделения полужирным — здесь нужно подставить свои данные.*

Для HTTPS:

```
server {
   listen 80;
   server_name site1.local www.site1.local;
   return 301 https://$host$request_uri;
}

server {
   listen       443 ssl;
   ssl on;
   ssl_certificate /etc/nginx/ssl/cert.pem;
   ssl_certificate_key /etc/nginx/ssl/cert.key;

   server_name  site1.local www.site1.local;
   set $root_path /var/www/site1/www;

   access_log /var/www/site1/log/nginx/access_log;
   error_log /var/www/site1/log/nginx/error_log;

   gzip  on;
   gzip_disable "msie6";
   gzip_min_length 1000;
   gzip_vary on;
   gzip_proxied    expired no-cache no-store private auth;
   gzip_types      text/plain text/css application/json application/x-javascript text/xml application/xml application/xml+rss text/javascript application/javascript;

   root   $root_path;

   location / {
      proxy_pass http://127.0.0.1:8080/;
      proxy_redirect     off;
      proxy_set_header   Host             $host;
      proxy_set_header   X-Forwarded-Proto $scheme;
      proxy_set_header   X-Real-IP        $remote_addr;
      proxy_set_header   X-Forwarded-For  $proxy_add_x_forwarded_for;
   }

   location ~* ^.+\.(jpg|jpeg|gif|png|css|zip|tgz|gz|rar|bz2|doc|docx|xls|xlsx|exe|pdf|ppt|tar|wav|bmp|rtf|js)$ {
         expires modified +1w;
   }
}
```

** в первой секции server мы перенаправляем все запросы по незащищенному **http** на **https**.*
*** **ssl_certificate** и **ssl_certificate_key** — пути к публичному и приватному ключам соответственно.*

Теперь настроим виртуальный домен в Apache:

```
nano /etc/httpd/conf.d/$TMP_SITE.conf
<VirtualHost *:8080>
   Define root_domain site1.local
   Define root_path /var/www/site1

   ServerName ${root_domain}
   ServerAlias www.${root_domain}
   DocumentRoot ${root_path}/www

   ErrorLog    ${root_path}/log/apache/error_log
   TransferLog  ${root_path}/log/apache/access_log
</VirtualHost>
```

Создаем каталоги для сайта:

```
mkdir -p /var/www/$TMP_SITE/{www,tmp}
mkdir -p /var/www/$TMP_SITE/log/{nginx,apache}
```

Создаем индексный файл со следующим содержимым:

```
nano /var/www/$TMP_SITE/www/index.php
<?php echo "<h1>Hello from site1</h1>"; ?>
```

Задаем права на папки:

```
chown -R apache:apache /var/www/$TMP_SITE
chmod -R 775 /var/www/$TMP_SITE
```

Проверяем корректность настроек конфигурационных файлов:

```
nginx -t
apachectl configtest
```

Перезапускаем веб-сервер:

```
systemctl reload nginx
systemctl reload httpd
```

Открываем сайт в браузере.

При необходимости, создаем базу данных.

```
mysql -uroot -p
> CREATE DATABASE site1 DEFAULT CHARACTER SET utf8 DEFAULT COLLATE utf8_general_ci;
> GRANT ALL PRIVILEGES ON site1.* TO dbuser@localhost IDENTIFIED BY 'password' WITH GRANT OPTION;
```

** данными sql-командами мы создаем базу данных **site1** и предоставляем к ней доступ для учетной записи **dbuser** с паролем **password**. При желании сделать соединение более безопасным, можно убрать* **WITH GRANT OPTION**.

Сделайте скриншоты (фотографии) процесса настройки web-сервера и вставьте в отчёт.

2.21 Практическая работа№ 21
Настройка сервера DNS в CentOS. Настройка сервера DHCP в CentOS

Задание 1:

1. Инсталляция необходимых пакетов

Перед началом рассмотрения следующих инструкций хотим отметить, что на нашем сайте уже имеется общее руководство по конфигурации стандартного DNS в Linux. Мы рекомендуем задействовать именно тот материал, если следует выставить настройки для

обычного посещения интернет-сайтов. Далее же мы покажем, как инсталлируется основной локальный DNS-сервер с клиентской частью.

В качестве средства создания локального DNS-сервера рекомендуем задействовать **bind9**. Настройка последующих конфигурационных файлов тоже будет базироваться на общих принципах поведения этого компонента. По умолчанию **bind9** отсутствует в операционной системе, поэтому начнем с ее добавления.

1. Введем команду sudo dnf install bind bind-utils -y и нажать на **Enter** для ее активации.

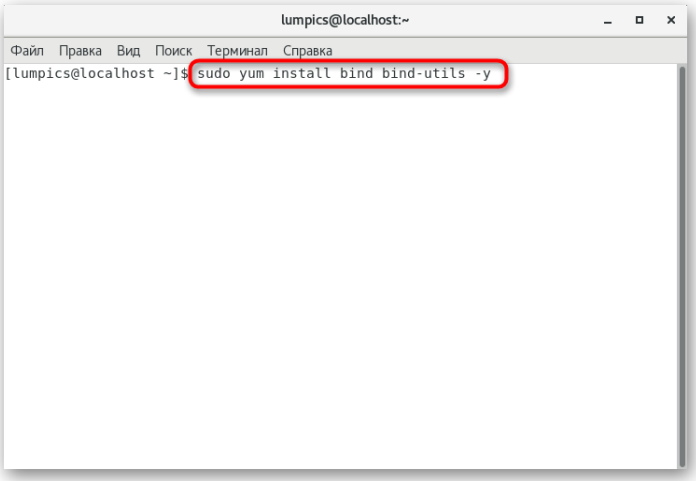


Рис. 281

2. Данное действие выполняется от имени суперпользователя (**sudo**), поэтому придется подтвердить учетную запись, введя пароль в появившуюся строку.

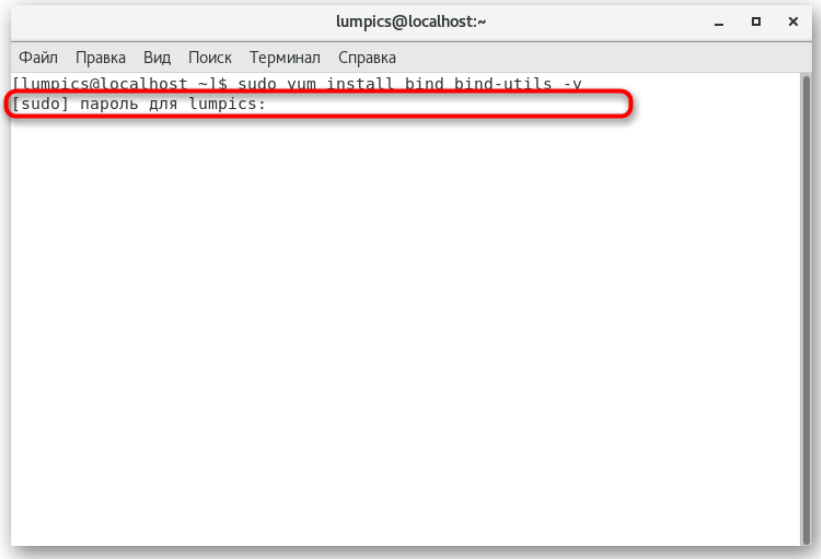


Рис. 282

3. Ожидайте завершения скачивания и установки пакетов.

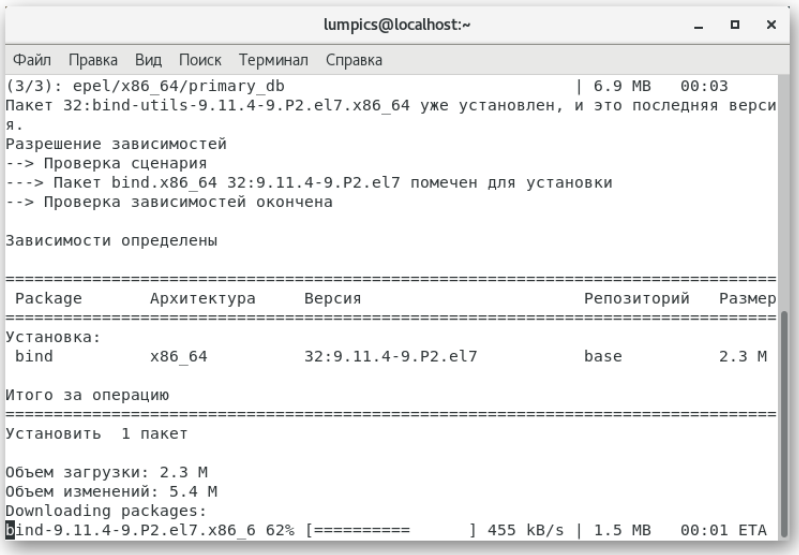


Рис. 283

По окончании данного процесса вы будете уведомлены о том, что все пакеты успешно добавлены в систему. После этого переходите к следующему шагу.

2. Глобальная настройка DNS-сервера

1. Для редактирования конфигурационных объектов можно использовать любой текстовый редактор. Предлагаем инсталлировать удобный **nano**, введя в консоли sudo dnf install nano.

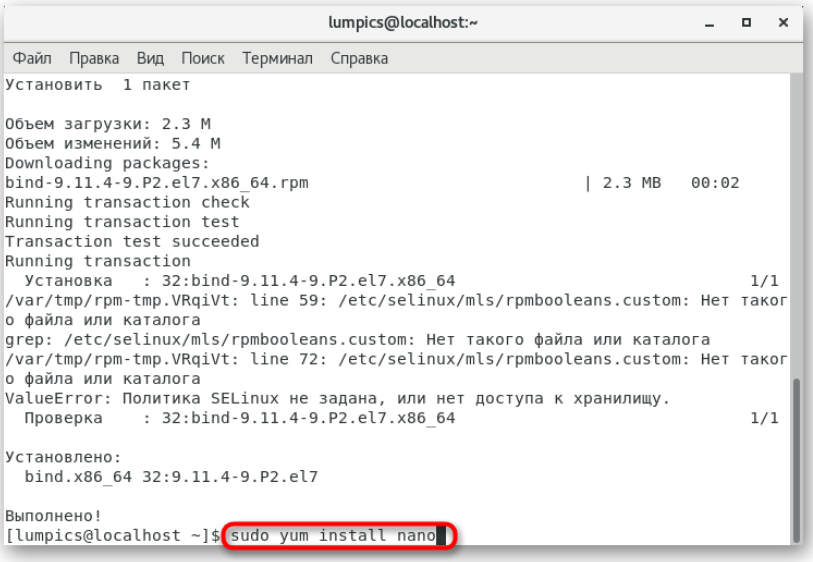


Рис. 284

2. Все необходимые пакеты будут загружены, а если они уже присутствуют в дистрибутиве, вы получите уведомление **«Выполнять нечего»**.

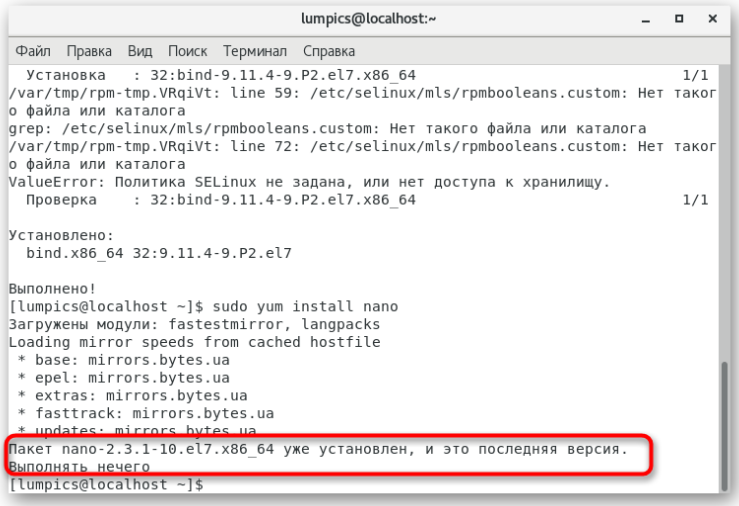


Рис. 285

3. Приступим к редактированию самого файла. Откройте его через sudo nano /etc/named.conf. При необходимости замените желаемый текстовый редактор, тогда строка получится примерно такой: sudo vi /etc/named.conf.

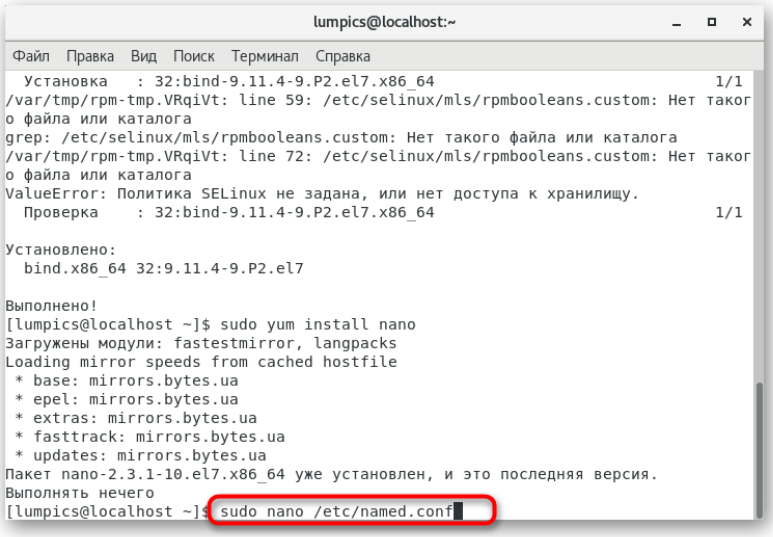


Рис. 286

4. Ниже мы приведем содержимое, которое нужно вставить в открывшийся файл или сверить его с уже существующим, добавив недостающие строки.

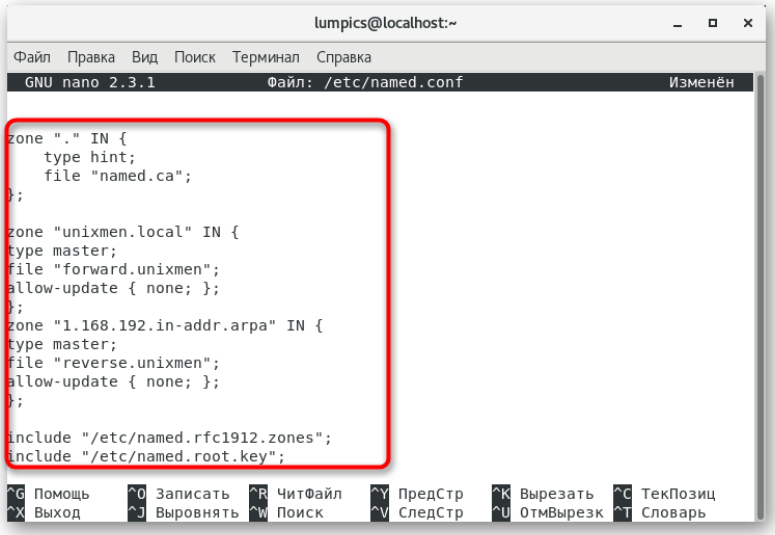


Рис. 287

5. После этого нажмите на **Ctrl + O**, чтобы записать изменения.

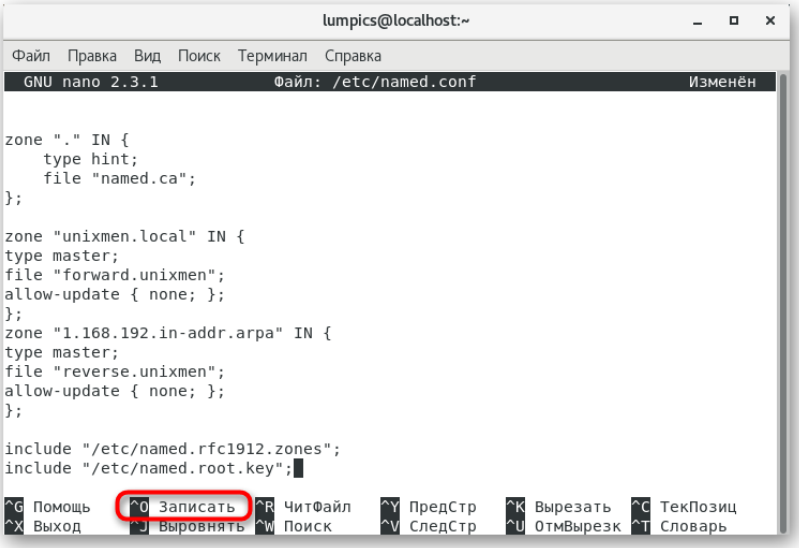


Рис. 288

6. Менять название файла не нужно, достаточно просто нажать на **Enter**.

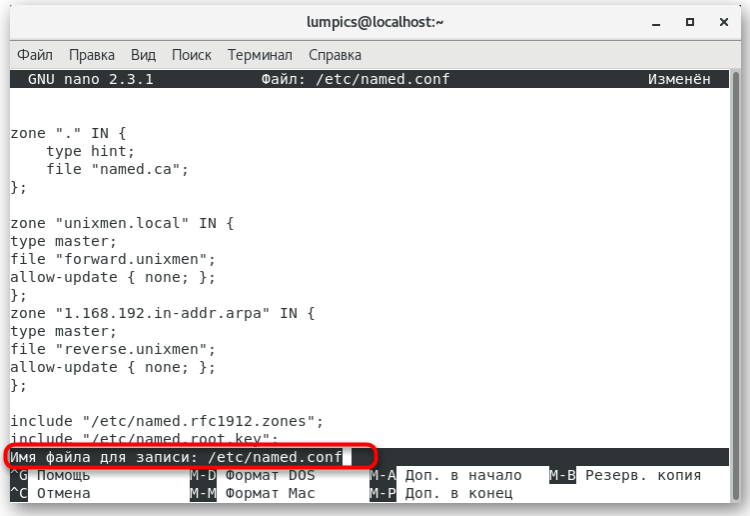


Рис. 289

7. Покиньте текстовый редактор через **Ctrl + X**.

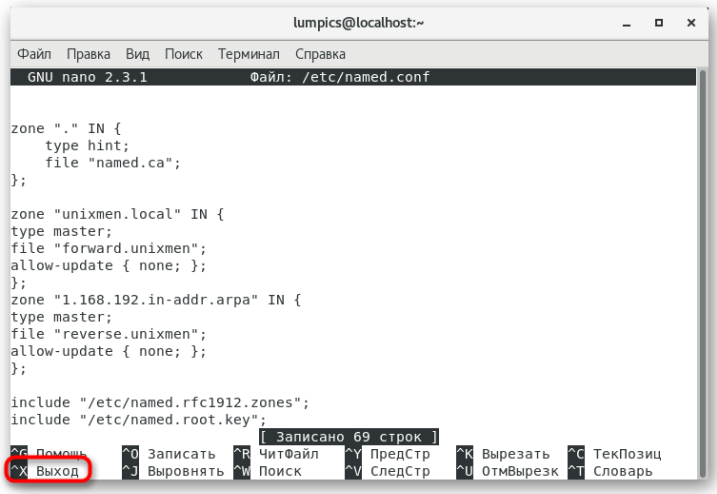


Рис. 290

Как уже было сказано ранее, в конфигурационный файл потребуется вставить определенные строки, задающие общие правила поведения DNS-сервера.

```
//
// named.conf
//
// Provided by Red Hat bind package to configure the ISC BIND named(8) DNS
// server as a caching only nameserver (as a localhost DNS resolver only).
//
// See /usr/share/doc/bind*/sample/ for example named configuration files.
//
options {
listen-on port 53 { 127.0.0.1; 192.168.1.101;}; ### Master DNS IP ###
# listen-on-v6 port 53 { ::1; };
directory "/var/named";
dump-file "/var/named/data/cache_dump.db";
statistics-file "/var/named/data/named_stats.txt";
memstatistics-file "/var/named/data/named_mem_stats.txt";
allow-query { localhost; 192.168.1.0/24;}; ### IP Range ###
allow-transfer{ localhost; 192.168.1.102; }; ### Slave DNS IP ###
/*
- If you are building an AUTHORITATIVE DNS server, do NOT enable recursion.
- If you are building a RECURSIVE (caching) DNS server, you need to enable
recursion.
- If your recursive DNS server has a public IP address, you MUST enable access
control to limit queries to your legitimate users. Failing to do so will
cause your server to become part of large scale DNS amplification
attacks. Implementing BCP38 within your network would greatly
reduce such attack surface
*/
recursion yes;
dnssec-enable yes;
dnssec-validation yes;
dnssec-lookaside auto;
/* Path to ISC DLV key */
bindkeys-file "/etc/named.iscdlv.key";
managed-keys-directory "/var/named/dynamic";
```

```
pid-file "/run/named/named.pid";
session-keyfile "/run/named/session.key";
};
logging {
channel default_debug {
file "data/named.run";
severity dynamic;
};
};
zone "." IN {
type hint;
file "named.ca";
};
zone "unixmen.local" IN {
type master;
file "forward.unixmen";
allow-update { none; };
};
zone "1.168.192.in-addr.arpa" IN {
type master;
file "reverse.unixmen";
allow-update { none; };
};
include "/etc/named.rfc1912.zones";
include "/etc/named.root.key";
```

** где **192.168.1.101** — IP-адрес нашего NS-сервера, на котором он будет принимать запросы; **allow-query** из соображений безопасности можно ограничить доступ для конкретной сети, например, вместо **any** написать **192.168.1.0/24**.*

Убедитесь, что все выставлено в точности так, как показано выше, а уже потом переходите к следующему шагу.

3. Создание прямой и обратной зоны

Для получения информации об источнике DNS-сервер использует прямые и обратные зоны. Прямая позволяет получать IP-адрес по имени хоста, а обратная через IP выдает доменное имя. Корректная работа каждой зоны должна быть обеспечена специальными правилами, созданием которых мы и предлагаем заняться далее.

1. Для прямой зоны создадим отдельный файл через тот же текстовый редактор. Тогда строка будет выглядеть так: sudo nano /var/named/forward.unixmen.

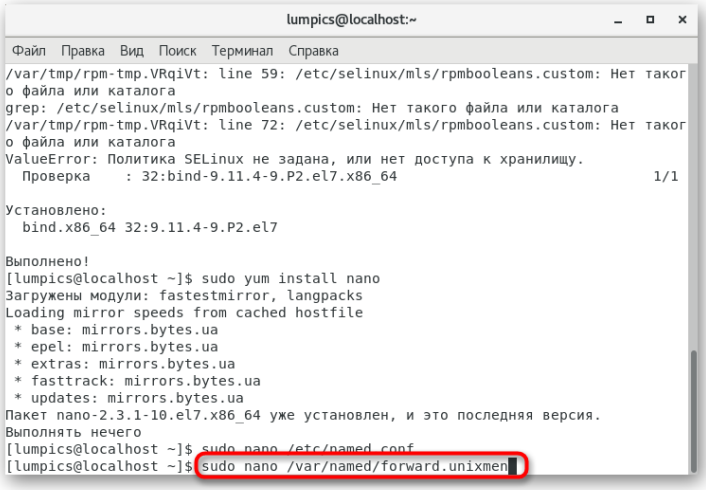


Рис. 291

2. Вы будете уведомлены о том, что это пустой объект. Вставьте туда указанное ниже содержимое:

```
$TTL 86400
@ IN SOA masterdns.unixmen.local. root.unixmen.local. (
2011071001 ;Serial
3600 ;Refresh
1800 ;Retry
604800 ;Expire
86400 ;Minimum TTL
)
@ IN NS masterdns.unixmen.local.
@ IN NS secondarydns.unixmen.local.
@ IN A 192.168.1.101
@ IN A 192.168.1.102
@ IN A 192.168.1.103
masterdns IN A 192.168.1.101
secondarydns IN A 192.168.1.102
client IN A 192.168.1.103
```

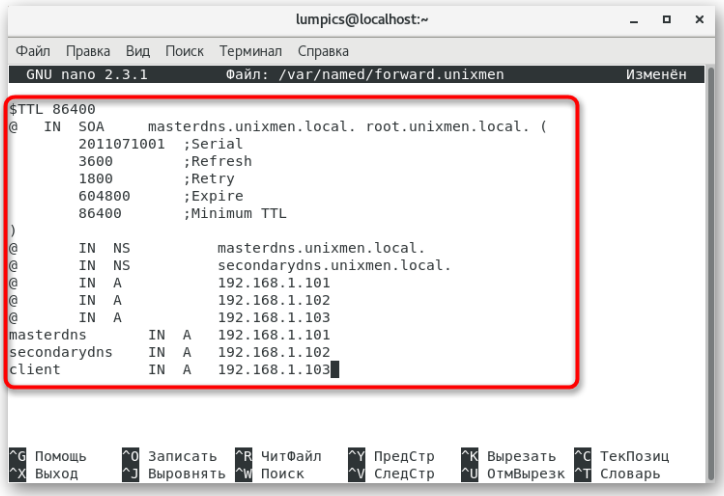


Рис. 292

3. Сохраните изменения и закройте текстовый редактор.

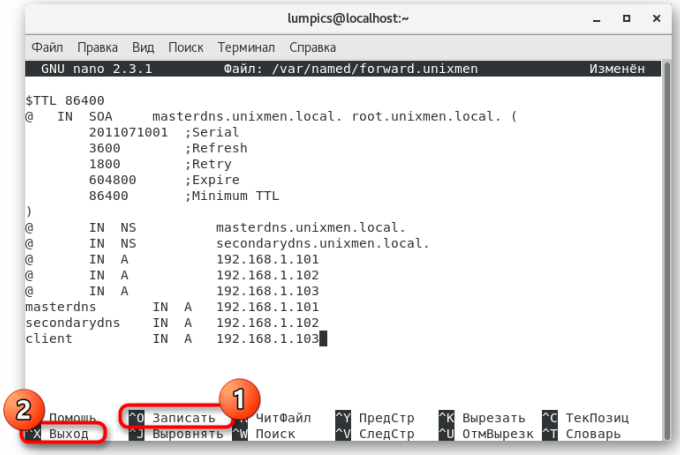


Рис. 293

4. Теперь перейдем к обратной зоне. Для нее требуется файл /var/named/reverse.unixmen.

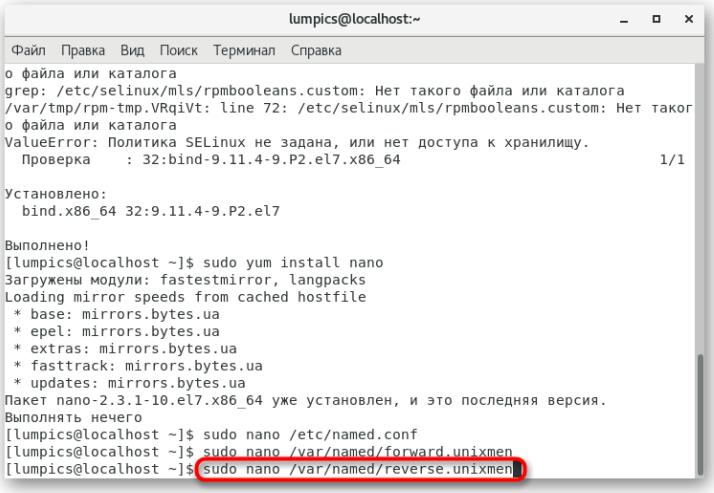


Рис. 294

5. Это тоже будет новый пустой файл. Вставьте туда:

```
$TTL 86400
@ IN SOA masterdns.unixmen.local. root.unixmen.local. (
2011071001 ;Serial
3600 ;Refresh
1800 ;Retry
604800 ;Expire
86400 ;Minimum TTL
)
@ IN NS masterdns.unixmen.local.
@ IN NS secondarydns.unixmen.local.
@ IN PTR unixmen.local.
masterdns IN A 192.168.1.101
secondarydns IN A 192.168.1.102
client IN A 192.168.1.103
101 IN PTR masterdns.unixmen.local.
102 IN PTR secondarydns.unixmen.local.
103 IN PTR client.unixmen.local.
```

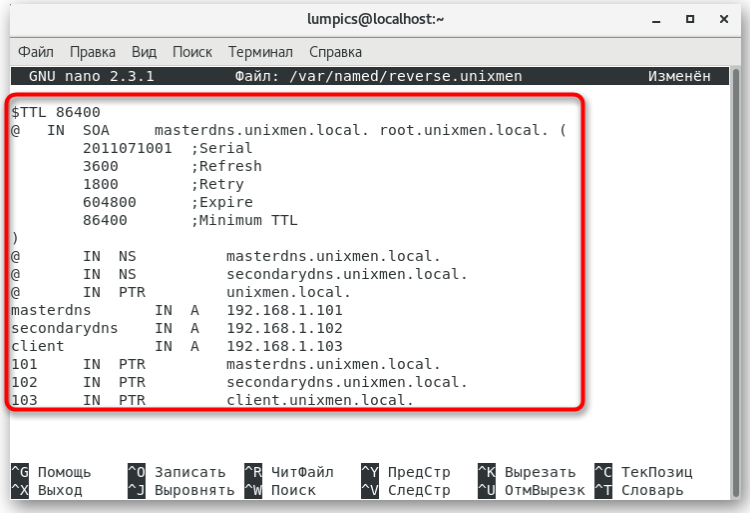


Рис. 295

6. При сохранении не изменяйте название объекта, а просто нажмите на клавишу **Enter**.

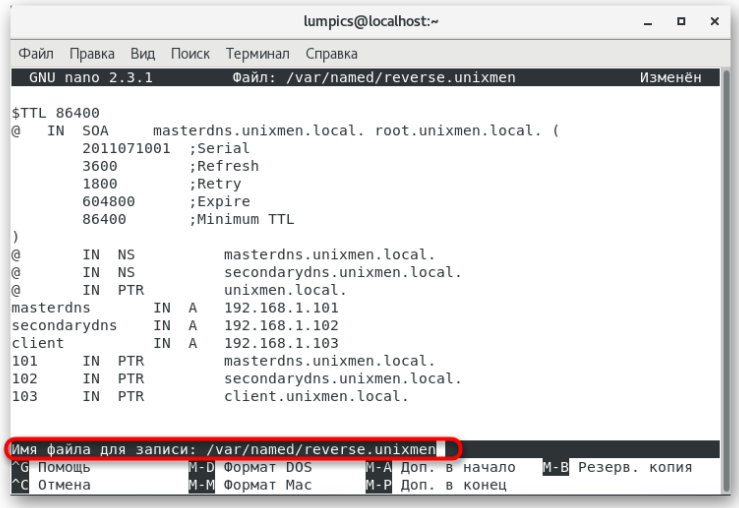


Рис. 296

Теперь указанные файлы будут использованы для прямой и обратной зоны. При необходимости следует редактировать именно их, чтобы изменить какие-то параметры. Об этом вы тоже можете прочесть в официальной документации.

4. Запуск DNS-сервера

После выполнения всех предыдущих указаний можно уже запустить DNS-сервер, чтобы в будущем легко проверить его работоспособность и продолжить настройку важных параметров. Осуществляется поставленная задача следующим образом:

1. В консоли введите sudo systemctl enable named, чтобы добавить DNS-сервер в автозагрузку для автоматического запуска при старте операционной системы.

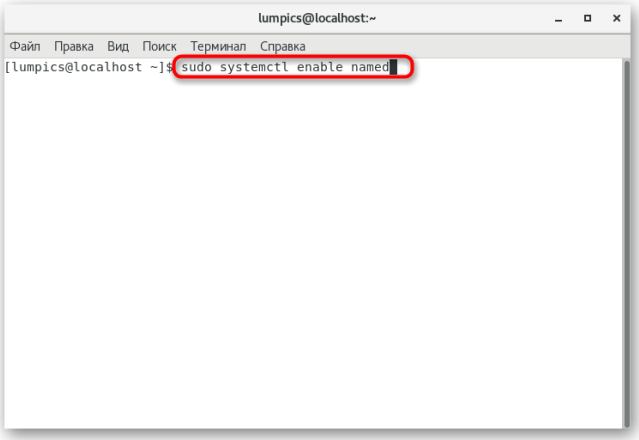


Рис. 297

2. Подтвердите это действие, введя пароль суперпользователя.

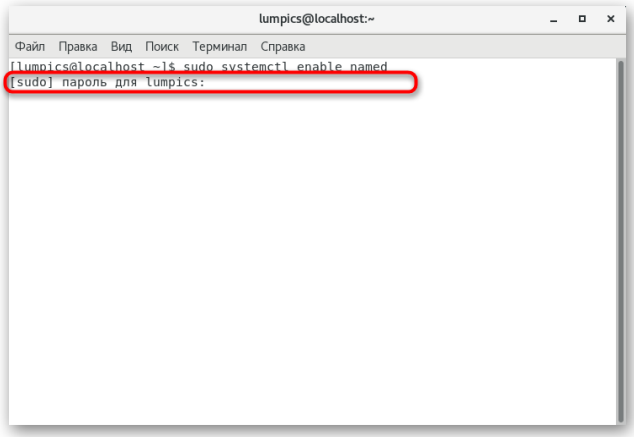


Рис. 298

3. Вы будете уведомлены о создании символической ссылки, а значит, действие выполнено успешно.

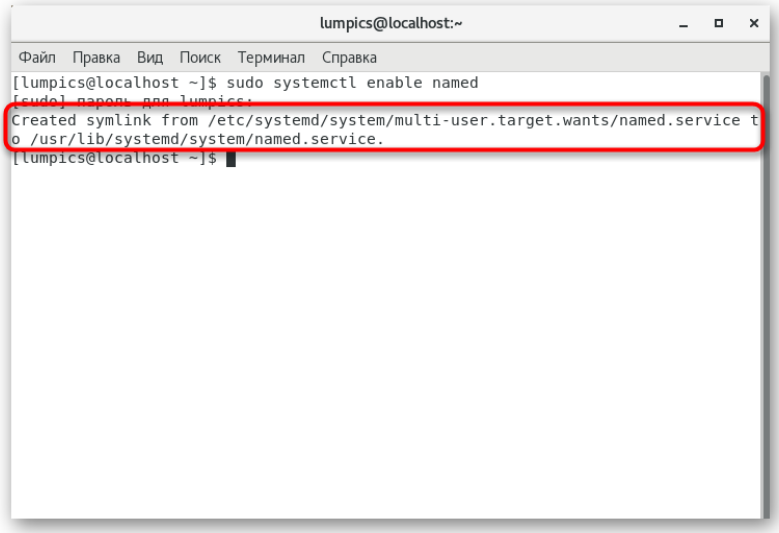


Рис. 299

4. Запустите утилиту через systemctl start named. Остановить ее можно так же, только заменив опцию **start** на stop.

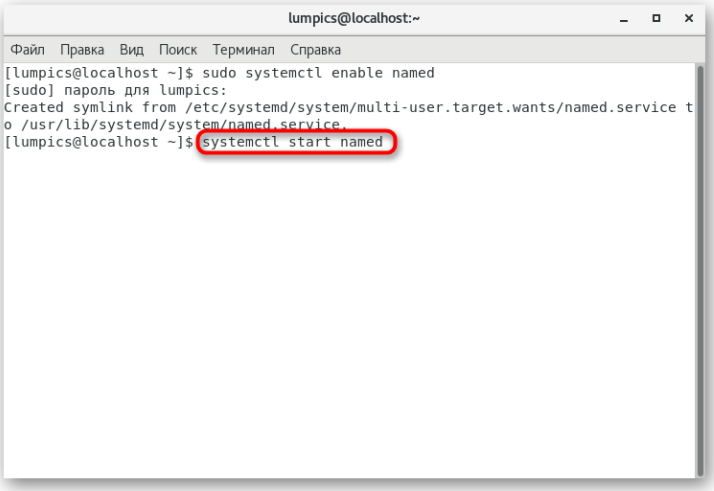


Рис. 300

5. При отображении всплывающего окна с подтверждением подлинности введите пароль от root.

5. Изменение параметров межсетевого экрана

Для корректного функционирования DNS-сервера потребуется открыть порт 53, что осуществляется через стандартный межсетевой экран FirewallD.

В **«Терминале»** потребуется ввести всего три простых команды:

1. Первая имеет вид firewall-cmd --permanent --add-port=53/tcp и отвечает за открытие порта TCP-протокола. Вставьте ее в консоль и нажмите на **Enter**.

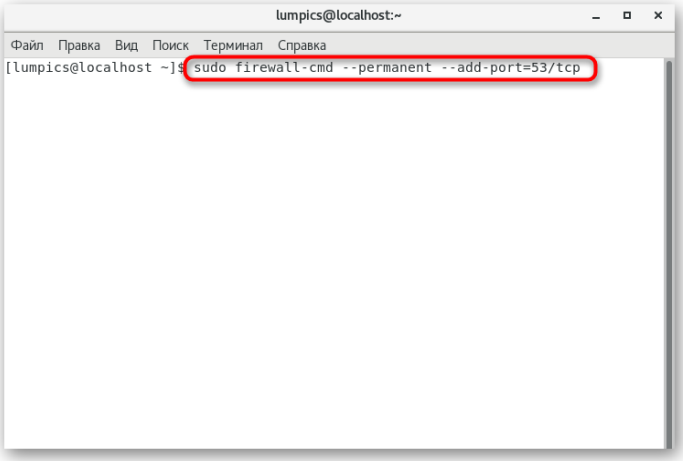


Рис. 301

2. Вы должны получить уведомление **«Success»**, что свидетельствует об успешном применении правила. После этого вставьте строку firewall-cmd --permanent --add-port=53/udp для открытия порта протокола UDP.

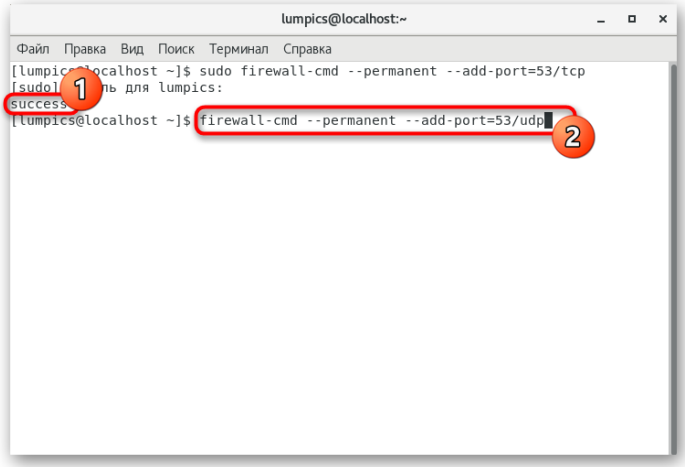


Рис. 302

3. Все изменения будут применены только после перезагрузки межсетевого экрана, что производится через команду firewall-cmd --reload.

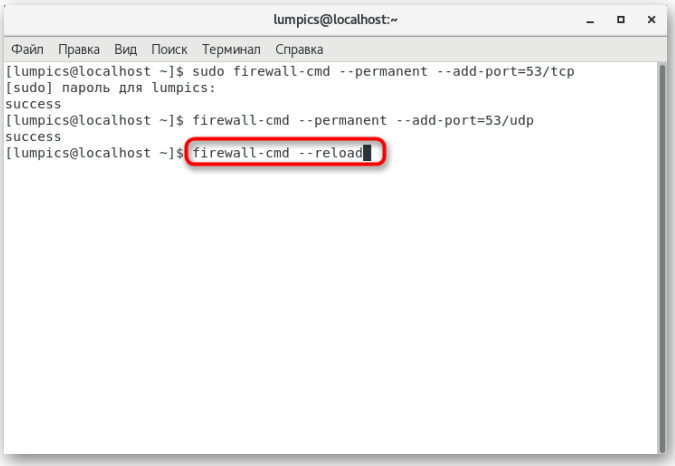


Рис. 303

Больше никаких изменений с брандмауэром производить не придется. Держите его постоянно во включенном состоянии, чтобы не возникло проблем с получением доступа.

6. Настройка прав доступа

Сейчас потребуется выставить основные разрешения и права доступа, чтобы немного обезопасить функционирование DNS-сервера и оградить обычных пользователей от возможности изменять параметры.

Все последующие команды должны быть активированными от имени суперпользователя. Чтобы постоянно не вводить пароль, советую включить перманентный рут-доступ для текущей терминальной сессии. Для этого в консоли введите su.

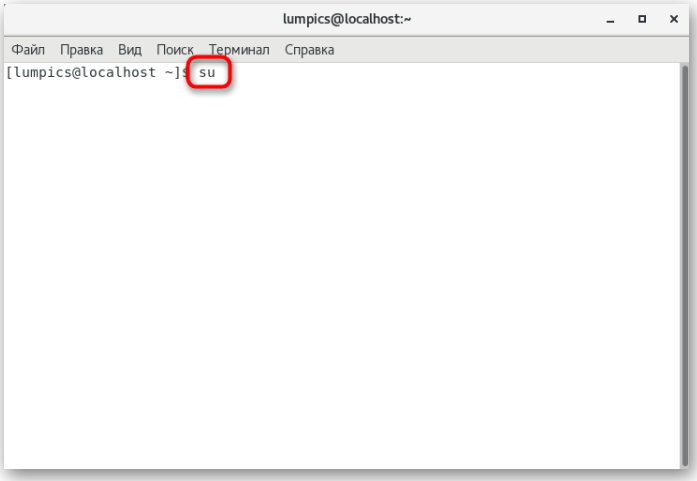


Рис. 304

1. Укажите пароль доступа.

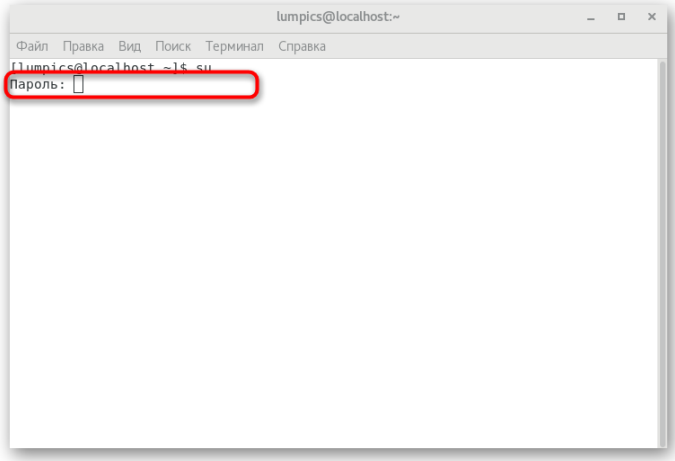


Рис. 305

2. После этого поочередно впишите указанные ниже команды, чтобы создать оптимальную настройку доступа:

chgrp named -R /var/named
chown -v root:named /etc/named.conf
restorecon -rv /var/named
restorecon /etc/named.conf

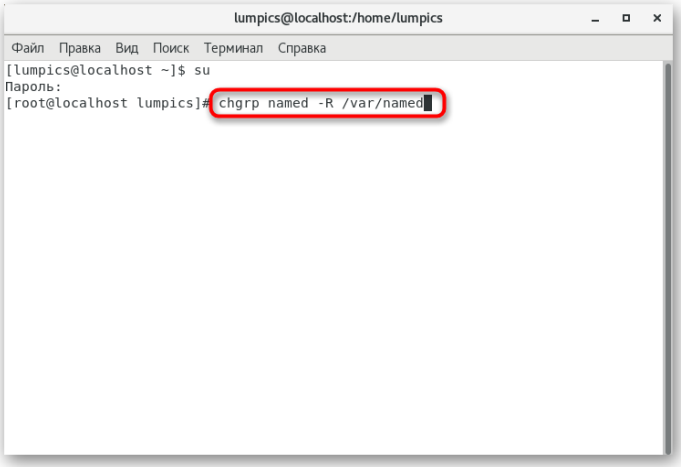


Рис. 306

На этом общая конфигурация главного DNS-сервера закончена. Осталось только отредактировать несколько конфигурационных файлов и произвести тестирование на ошибки. Со всем этим мы и предлагаем разобраться в следующем шаге.

7. Тестирование на ошибки и завершение настройки

Рекомендуем начать с проверок на ошибки, чтобы в будущем не пришлось менять и оставшиеся конфигурационные файлы. Именно поэтому мы и рассмотрим все это в пределах одного шага, а также приведем образцы правильного вывода команд для тестирования.

1. Введите в **«Терминале»** named-checkconf /etc/named.conf. Это позволит проверить глобальные параметры. Если в результате никакого вывода не последовало, значит, все настроено корректно. В противном случае изучите сообщение и, отталкиваясь от него, решите проблему.

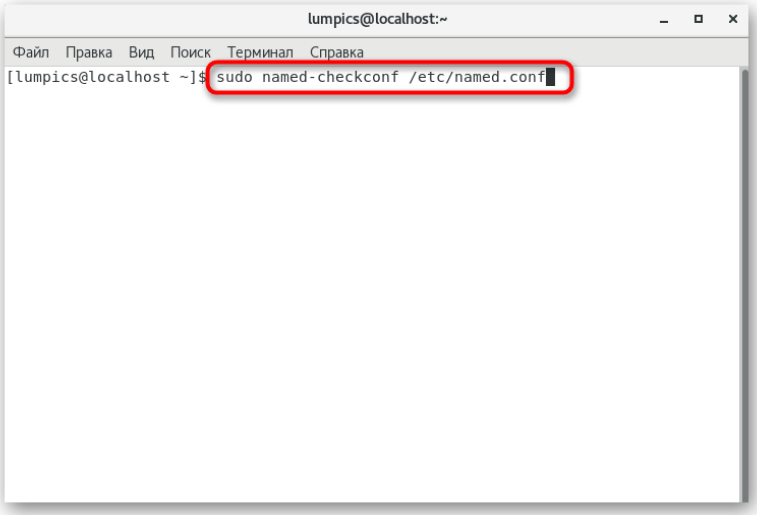


Рис. 307

2. Далее требуется проверить прямую зону, вставив строку named-checkzone unixmen.local /var/named/forward.unixmen.

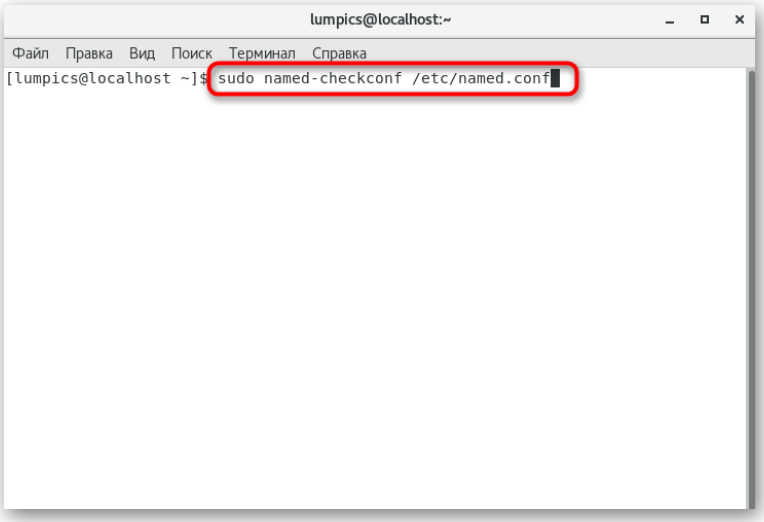


Рис. 308

3. Образец вывода выглядит следующим образом: zone unixmen.local/IN: loaded serial 2011071001 OK.

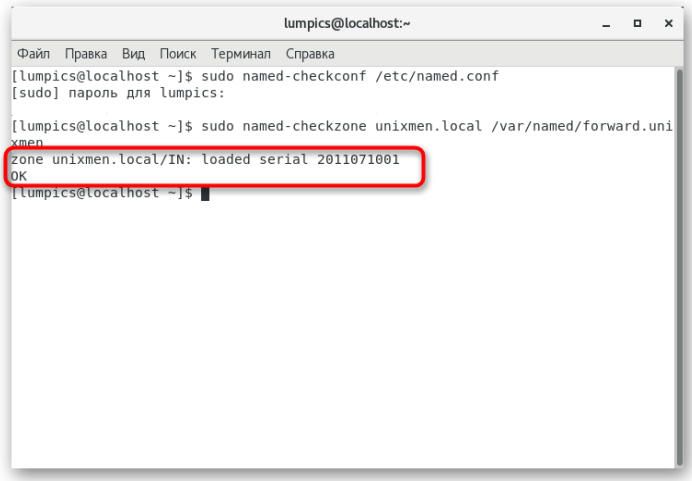


Рис. 309

4. Примерно то же самое осуществляем и с обратной зоной через named-checkzone unixmen.local /var/named/reverse.unixmen.

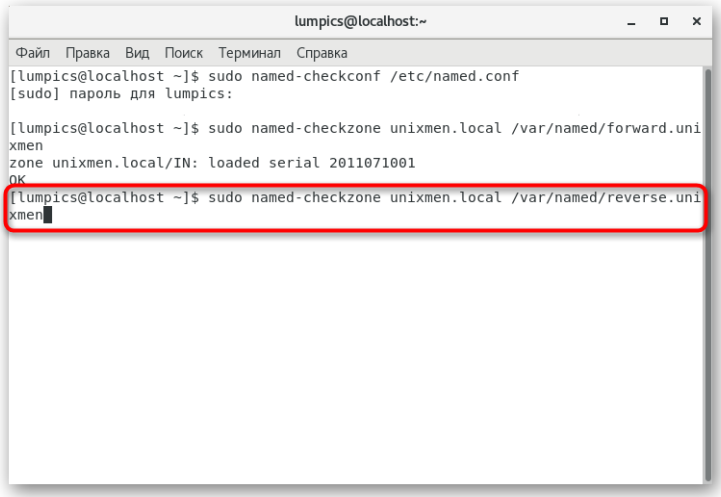


Рис. 310

5. Правильный вывод должен быть таким: zone unixmen.local/IN: loaded serial 2011071001 OK.

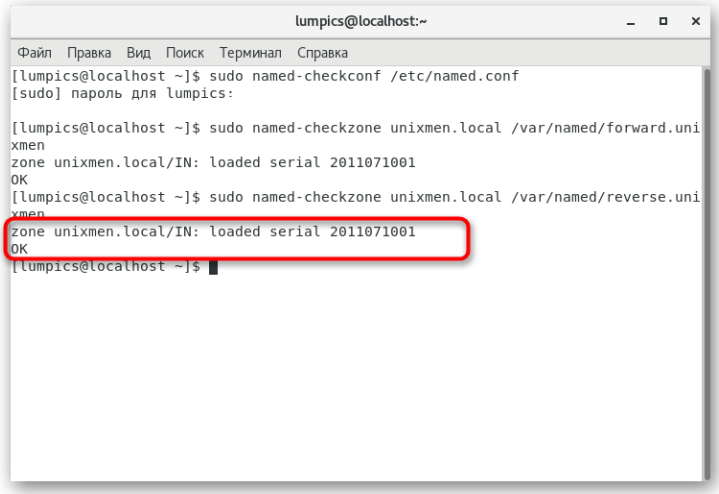


Рис. 311

6. Теперь перейдем к настройкам основного сетевого интерфейса. В него потребуется добавить данные текущего DNS-сервера. Для этого откройте файл /etc/sysconfig/network-scripts/ifcfg-enp0s3.

* enp0s3 – сетевой интерфейс в примере, у вас может отличаться.

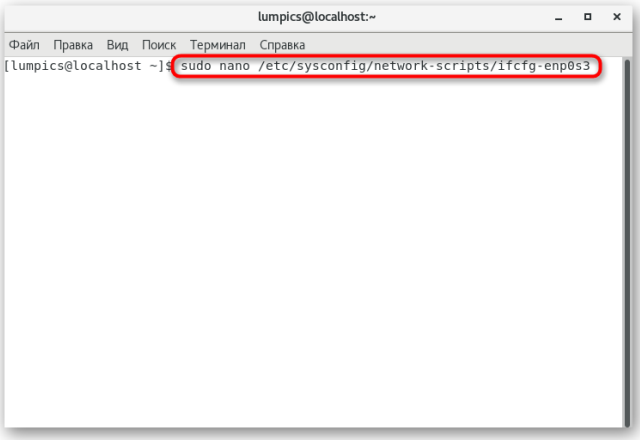


Рис. 312

7. Проверьте, чтобы содержимое было такое, как показано ниже. При необходимости вставьте параметры DNS.

```
TYPE="Ethernet"
BOOTPROTO="none"
DEFROUTE="yes"
IPV4_FAILURE_FATAL="no"
IPV6INIT="yes"
IPV6_AUTOCONF="yes"
IPV6_DEFROUTE="yes"
IPV6_FAILURE_FATAL="no"
NAME="enp0s3"
UUID="5d0428b3-6af2-4f6b-9fe3-4250cd839efa"
ONBOOT="yes"
HWADDR="08:00:27:19:68:73"
IPADDR0="192.168.1.101"
PREFIX0="24"
GATEWAY0="192.168.1.1"
DNS="192.168.1.101"
IPV6_PEERDNS="yes"
IPV6_PEERROUTES="yes"
```

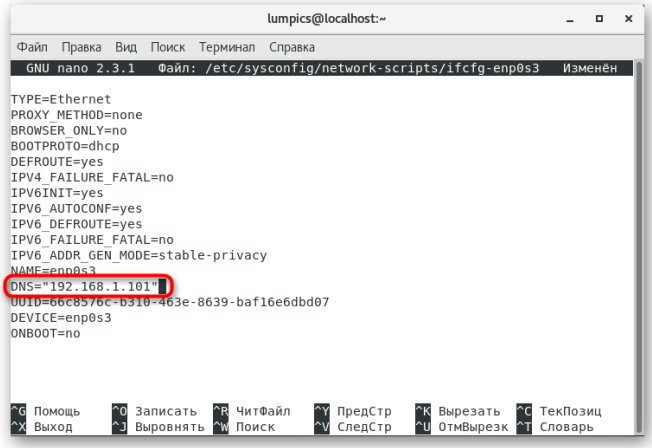


Рис. 313

8. После сохранения изменений переходите к файлу /etc/resolv.conf.

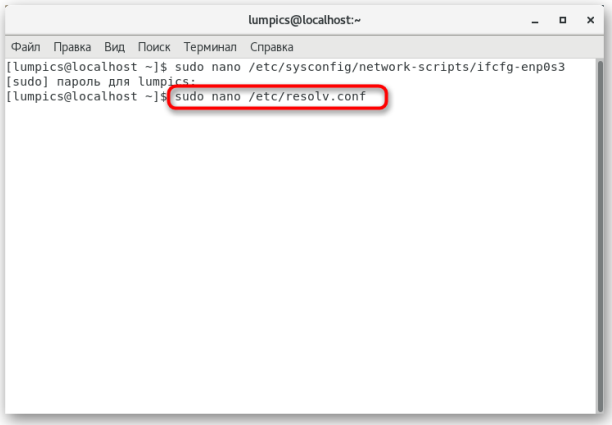


Рис. 314

9. Здесь нужно добавить всего одну строку: nameserver 192.168.1.101.

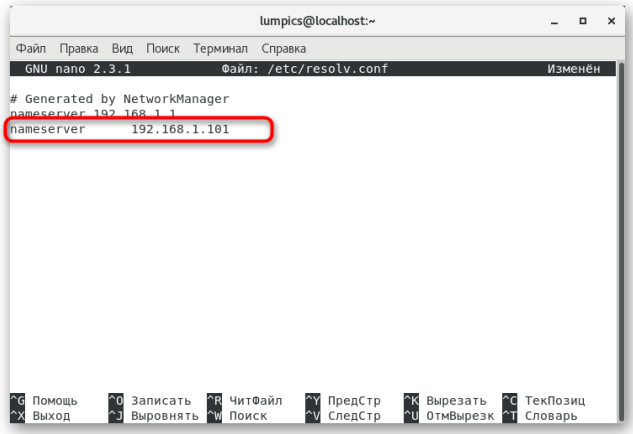


Рис. 315

10. По завершении остается только перезагрузить сеть или компьютер, чтобы обновить конфигурацию. Служба сети перезапускается через команду systemctl restart network.

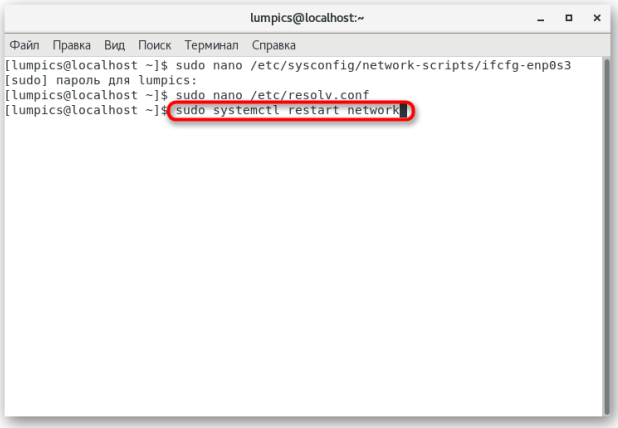


Рис. 316

8. Проверка установленного DNS-сервера

В завершении конфигурации остается только проверить работу имеющегося DNS-сервера после его добавления в глобальную службу сети. Эта операция так же выполняется при помощи специальных команд. Первая из них имеет вид dig masterdns.unixmen.local.

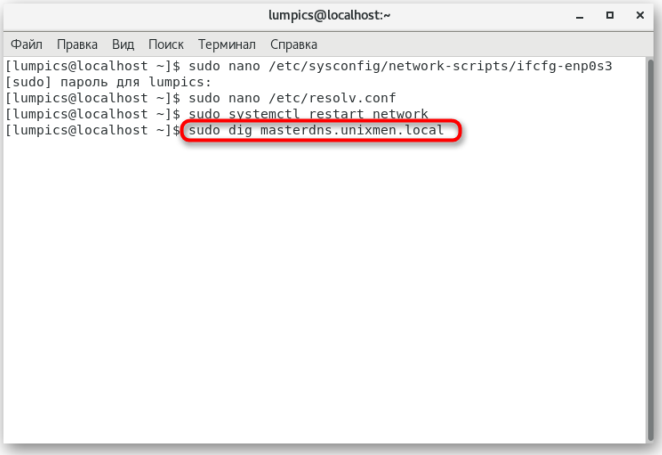


Рис. 317

В результате на экране должен появиться вывод, имеющий схожее представление с указанным ниже содержимым.

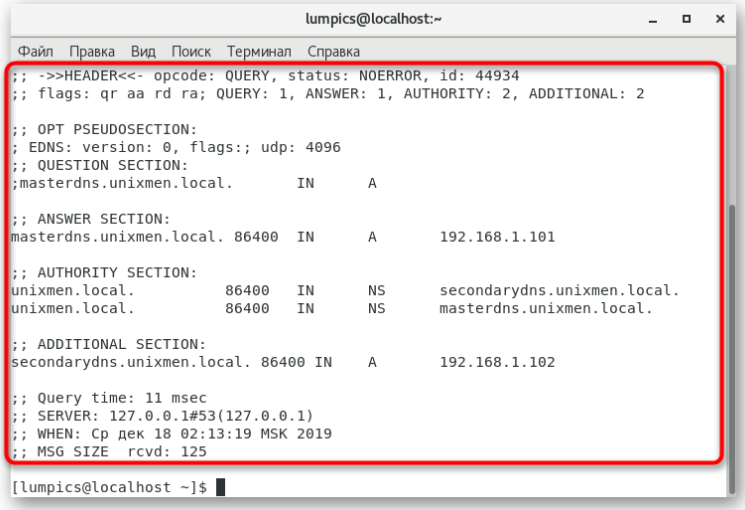


Рис. 318

; <> DiG 9.9.4-RedHat-9.9.4-14.el7 <> masterdns.unixmen.local
;; global options: +cmd
;; Got answer:
;; ->>HEADER<<- opcode: QUERY, status: NOERROR, id: 25179
;; flags: qr aa rd ra; QUERY: 1, ANSWER: 1, AUTHORITY: 2, ADDITIONAL: 2
;; OPT PSEUDOSECTION:
; EDNS: version: 0, flags:; udp: 4096
;; QUESTION SECTION:
;masterdns.unixmen.local. IN A
;; ANSWER SECTION:
masterdns.unixmen.local. 86400 IN A 192.168.1.101
;; AUTHORITY SECTION:
unixmen.local. 86400 IN NS secondarydns.unixmen.local.
unixmen.local. 86400 IN NS masterdns.unixmen.local.
;; ADDITIONAL SECTION:
secondarydns.unixmen.local. 86400 IN A 192.168.1.102
;; Query time: 0 msec
;; SERVER: 192.168.1.101#53(192.168.1.101)
;; WHEN: Wed Aug 20 16:20:46 IST 2014
;; MSG SIZE rcvd: 125

Дополнительная команда позволит узнать о состоянии локальной работы DNS-сервера. Для этого в консоль вставьте nslookup unixmen.local и нажмите на **Enter**.

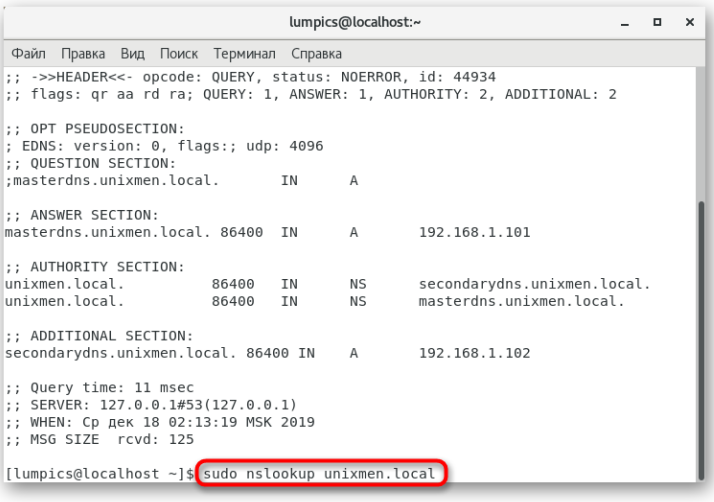


Рис. 319

В результате должно отобразиться три разных представления адресов IP и доменных имен.

Server: 192.168.1.101
Address: 192.168.1.101#53
Name: unixmen.local
Address: 192.168.1.103
Name: unixmen.local
Address: 192.168.1.101
Name: unixmen.local
Address: 192.168.1.102

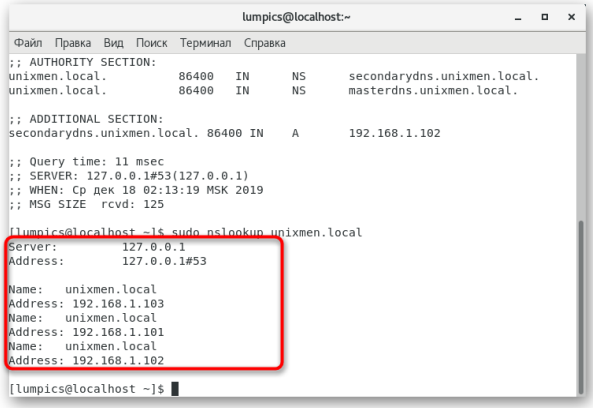


Рис. 320

Если вывод соответствует тому, который указан выше, значит, конфигурация завершена успешно и можно переходить к работе с клиентской частью DNS-сервера.

9. Настройка клиентской части DNS-сервера

Мы не будем разделять эту процедуру на отдельные шаги, поскольку она выполняется путем редактирования всего одного конфигурационного файла. В него необходимо добавить информацию обо всех клиентах, которые будут подключены к серверу, а пример такой настройки выглядит так:

1. Откройте файл /etc/resolv.conf через любой удобный текстовый редактор.

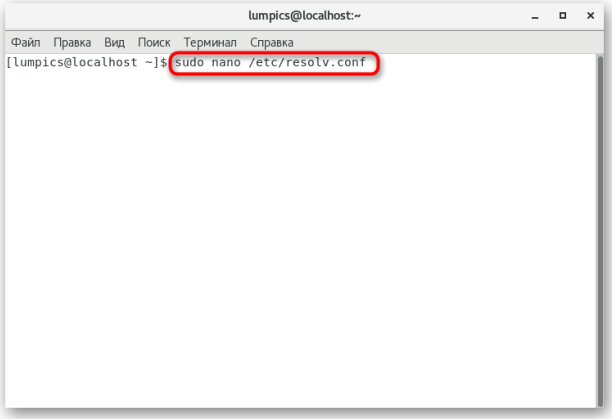


Рис. 321

2. Добавьте туда строки search unixmen.local nameserver 192.168.1.101 и nameserver 192.168.1.102, заменив необходимое на клиентские адреса.

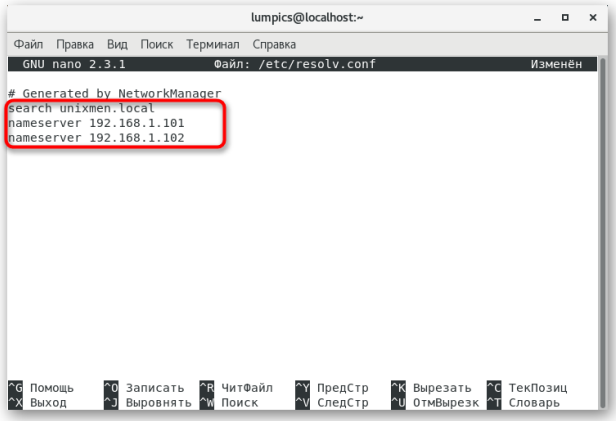


Рис. 322

3. При сохранении не изменяйте имя файла, а просто нажмите на клавишу **Enter**.

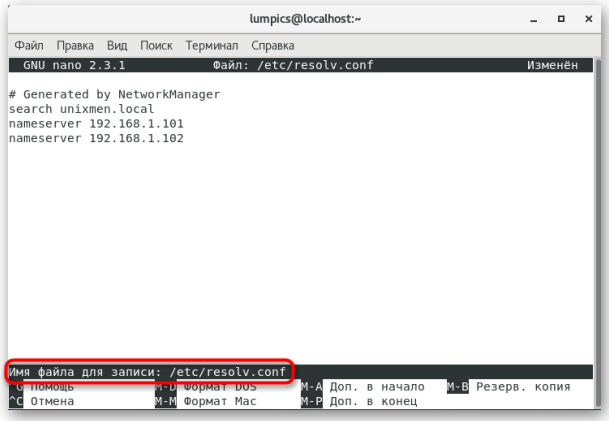


Рис. 323

4. После выхода из текстового редактора в обязательном порядке перезагрузите глобальную сеть через команду systemctl restart network.

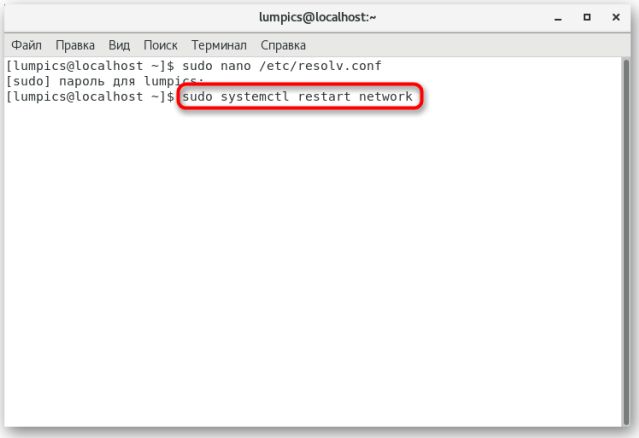


Рис. 324

1. Тестирование DNS-сервера

Последний этап нашего сегодняшнего материала — завершающее тестирование DNS-сервера. Ниже вы видите несколько команд, позволяющий справиться с поставленной задачей. Используйте одну из них, активировав через **«Терминал»**. Если в выводе не наблюдается никаких ошибок, следовательно, весь процесс выполнен верно.

dig masterdns.unixmen.local
dig secondarydns.unixmen.local
dig client.unixmen.local
nslookup unixmen.local

Сделайте скриншоты (фотографии) процесса настройки сервера DNS и вставьте в отчёт.

Задание 2:

1. Устанавливаем DHCP:

```
dnf install dhcp-server
```

Теперь откроем на редактирование конфигурационный файл:

```
nano /etc/dhcp/dhcpd.conf
```

И внесем в него, примерно, следующее:

```
subnet 192.168.1.0 netmask 255.255.255.0 {
  range 192.168.1.200 192.168.0.250;
  option domain-name-servers 192.168.1.101, 192.168.1.102;
  option domain-name "wbsh.local";
  option routers 192.168.1.1;
  option broadcast-address 192.168.1.255;
  default-lease-time 600;
  max-lease-time 7200;
}
```

** где*

- ***subnet*** *обозначает сеть, в области которой будет работать данная группа настроек;*
- ***range*** *— диапазон, из которого будут браться IP-адреса;*
- ***option domain-name-servers*** *— через запятую перечисленные DNS-сервера;*
- ***option domain-name*** *— суффикс доменного имени;*
- ***option routers*** *— шлюз по умолчанию;*
- ***option broadcast-address*** *— адрес сети для широковещательных запросов;*
- ***default-lease-time***, ***max-lease-time*** *— время и максимальное время в секундах, на которое клиент получит адрес, по его истечению будет выполнено продление срока.*

*** все примеры настроек можно увидеть в файле* ***/usr/share/doc/dhcp*/dhcpd.conf.example*** *(вместо * будет версия установленного dhcp).*

Проверить корректность конфигурационного файла можно командой:

```
dhcpd -t -cf /etc/dhcp/dhcpd.conf
```

Разрешаем автозапуск сервиса:

```
systemctl enable dhcpd
```

и запускаем его:

```
systemctl start dhcpd
```

Добавляем правило в firewalld:

```
firewall-cmd --permanent --add-service=dhcp
firewall-cmd --reload
```

2. Определяем интерфейс для работы

Если в системе присутствует несколько сетевых адаптеров, а сервер DHCP должен работать только для определенных, открываем на редактирование следующий файл:

```
nano /etc/sysconfig/dhcpd
```

И добавляем в него следующее:

```
DHCPDARGS=ens32
```

** в данном примере сервер будет работать только для интерфейса* ***ens32****.*

Перезапускаем сервис:

```
systemctl restart dhcp
```

3. Резервирование IP

Резервирование создается по MAC-адресу сетевого адаптера.
Пример настройки dhcpd.conf:

```
nano /etc/dhcp/dhcpd.conf
host host1 {
   hardware ethernet 28:10:7B:27:C2:A0; fixed-address 192.168.1.201;
}
host host2 {
   hardware ethernet 28:10:7B:27:C2:A1; fixed-address 192.168.1.202;
}
```

** где **host1** — имя узла, для когорого резервируем адрес (не обязательно должен совпадать с реальным); **28:10:7B:27:C2:A0** — mac-адрес; **192.168.1.201** — IP, который будет назначать узлу. Аналогичо, для второго узла.*

4. Подключение конфигурационных файлов

Для удобства, некоторые блоки с настройками можно вынести в отдельные файлы и подключить их в основном конфигурационном файле:

```
nano /etc/dhcp/dhcpd.conf
include "/etc/dhcp/conf.d/subnets.conf";
```

Список арендованных адресов
Для просмотра списка адресов, которые были выданы DHCP-сервером вводим команду:

```
cat /var/lib/dhcpd/dhcpd.leases
```

5. Настройка логов

По умолчанию, сервер dhcp ведет лог в файле /var/log/messages, что не очень удобно, так как это общий лог-файл, в котором может находиться много записей.
Для того, чтобы сервер сохранял записи в отдельный файл, открываем на редактирование rsyslog.conf:

```
nano /etc/rsyslog.conf
```

И добавляем следующее:

```
local6.*/var/log/dhcp.log
```

Далее открываем конфигурационный файл dhcp:

```
nano /etc/dhcp/dhcpd.conf
```

И добавляем:

```
log-facility local6;
```

Перезапускаем сервисы:

```
systemctl restart dhcpd
systemctl restart rsyslog
```

Сделайте скриншоты (фотографии) процесса настройки сервера DHCP и вставьте в отчёт.

2.22 Практическая работа№ 22
Установка и настройка OpenVPN. Применение протокола IP-sec и SSH

Задание 1:

1. Подготовка операционной системы

Мы внесем небольшие правки в настройки. Настроим время для правильного формирования клиентских сертификатов, отключим систему безопасности SELinux, откроем нужные порты брандмауэра.

1. Настройка времени

Установим правильную временную зону:

```
\cp /usr/share/zoneinfo/Europe/Moscow /etc/localtime
```

** в данном примере мы укажем московское время.*

Устанавливаем утилиту для синхронизации времени:

```
dnf install chrony
```

Разрешаем автозапуск службы chronyd и запускаем ее:

```
systemctl enable chronyd
systemctl start chronyd
```

Проверить корректность времени можно командой:

```
date
```

2. Настройка SELinux

В нашей инструкции мы просто отключим SELinux. Если необходимо его настроить и оставить включенным, используем инструкцию Настройка SELinux в CentOS 7 (для CentOS 8 она также подходит).

И так, отключаем Selinux командой:

```
setenforce 0
```

Чтобы Selinux не включился после перезагрузки, открываем на редактирование файл:

```
nano /etc/selinux/config
```

... и редактируем опцию *SELINUX*:

```
...
SELINUX=disabled
...
```

3. Настройка брандмауэра

Создаем правило для firewalld:

```
firewall-cmd --permanent --add-port=443/udp
```

** в данной инструкции мы настроим работу OpenVPN на порту **443** по **UDP**. Если в вашем случае необходим другие порт и протокол, меняем значения на соответствующие.*

Применяем настройку:

```
firewall-cmd --reload
```

4. Установка и создание сертификатов

Использование сертификатов является обязательным условием при использовании VPN. Поэтому сразу после установки мы создадим все необходимые ключи.

2. Установка OpenVPN

Устанавливаем репозиторий epel:

```
dnf install epel-release
```

Устанавливаем необходимые пакеты следующей командой:

```
dnf install openvpn easy-rsa
```

1. Создание сертификатов

Переходим в каталог easy-rsa:

```
cd /usr/share/easy-rsa/3
```

** в зависимости от версии easy-rsa, последний каталог может быть другим. Увидеть точное название каталога можно командой **ls /usr/share/easy-rsa/**.*

Чтобы упростить и ускорить процесс создания ключей, создаем следующий файл:

```
nano vars
```

```
export KEY_COUNTRY="RU"
export KEY_PROVINCE="Sankt-Petersburg"
export KEY_CITY="Sankt-Petersburg"
export KEY_ORG="DMOSK COMPANY"
export KEY_EMAIL="master@dmosk.ru"
export KEY_CN="DMOSK"
export KEY_OU="DMOSK"
export KEY_NAME="name-openvpn-server.dmosk.ru"
export KEY_ALTNAMES="name-openvpn-server"
```

** где **KEY_CN** и **KEY_OU**: рабочие подразделения (например, можно указать название отдела); **KEY_NAME**: адрес, по которому будет выполняться подключение (можно*

указать полное наименование сервера); ***KEY_ALTNAMES*** *— альтернативный адрес.*
** так как мы генерируем самоподписный сертификат, значения данных полей никак не повлияют на работу OpenVPN, однако, для удобства, лучше подставить реальные данные.*

Справка: когда клиенты подключаются к OpenVPN, они используют асимметричное шифрование (также известное как открытый/закрытый ключ) для выполнения TLS-рукопожатия. Однако при передаче шифрованного VPN-трафика сервер и клиенты используют симметричное шифрование, которое также известно как шифрование общедоступного ключа.

Симметричное шифрование требует гораздо меньшего количества вычислений по сравнению с асимметричным: используемые числа гораздо меньше, и современные процессоры имеют инструкции для выполнения оптимизированного симметричного шифрования. Для переключения с асимметричного на симметричное шифрование сервер OpenVPN и клиент будут использовать алгоритм Диффи — Хеллмана на эллиптических кривых для согласования общего секретного ключа в максимально короткие сроки.

Запускаем созданный файл на исполнение:

```
. ./vars
```

2. Генерация ключей

Инициализируем PKI:

```
./easyrsa init-pki
```

Мы должны увидеть:

```
init-pki complete; you may now create a CA or requests.
Your newly created PKI dir is: /usr/share/easy-rsa/3/pki
```

... а в текущем каталоге появится папка pki.

Генерируем корневой сертификат (CA):

```
./easyrsa build-ca
```

... после ввода **Enter** обязательно задаем пароль дважды. На запрос ввести **Common Name** можно просто нажать ввод или написать свое имя:

```
Common Name (eg: your user, host, or server name) [Easy-RSA CA]:
```

Создаем ключ Диффи-Хеллмана:

```
./easyrsa gen-dh
```

Для создания сертификата сервера необходимо сначала создать файл запроса:

```
./easyrsa gen-req vpn-server nopass
```

** на запрос ввода* ***Common Name*** *просто вводим* ***Enter****, чтобы использовать настройку из файла vars;* ***nopass*** *можно упустить, если хотим повысить безопасность с помощью пароля на сертификат.*

... и на его основе — сам сертификат:

```
./easyrsa sign-req server vpn-server
```

После ввода команды подтверждаем правильность данных, введя **yes:**

```
  Confirm request details: yes
```

... и вводим пароль, который указывали при создании корневого сертификата.

Для создания ta ключа используем команду:

```
openvpn --genkey --secret pki/ta.key
```

Сертификаты сервера готовы и находятся в каталоге **pki.**

Создаем каталог в /etc/openvpn, в котором будем хранить сертификаты:

```
mkdir -p /etc/openvpn/server/keys
```

Переходим в каталог pki:

```
cd pki
```

Копируем в него необходимые сертификаты:

```
cp ca.crt issued/vpn-server.crt private/vpn-server.key dh.pem ta.key /etc/openvpn/server/keys/
```

3. Настройка и запуск сервера

Создаем конфигурационный файл для сервера openvpn:

```
nano /etc/openvpn/server/server.conf
```

И вставляем в него следующее:

```
local 192.168.0.15
port 443
proto udp
dev tun
ca keys/ca.crt
cert keys/vpn-server.crt
key keys/vpn-server.key
dh keys/dh.pem
tls-auth keys/ta.key 0
server 172.16.10.0 255.255.255.0
ifconfig-pool-persist ipp.txt
keepalive 10 120
max-clients 32
persist-key
persist-tun
status /var/log/openvpn/openvpn-status.log
log-append /var/log/openvpn/openvpn.log
verb 0
mute 20
daemon
mode server
tls-server
comp-lzo no
```

** где из всех параметров, обязательно, внести изменения нужно в следующие — **local**: IP-адрес, на котором будет обрабатывать запросы OpenVPN; **port**: сетевой порт (443 позволит избежать проблем при использовании Интернета в общественных местах, но может быть уже занят в вашей системе — посмотреть список используемых портов можно командой **ss -tunlp**. Если порт занят, используйте любой из свободных, например 1194).*

Создаем каталог для логов сервера:

```
mkdir /var/log/openvpn
```

Разрешаем автоматический старт сервиса vpn:

```
systemctl enable openvpn-server@server
```

И запускаем его:

```
systemctl start openvpn-server@server
```

4. Настройка OpenVPN-клиента

Для настройки клиента необходимо на сервере сгенерировать сертификаты, а на клиентском компьютере установить программу openvpn и настроить ее.

5. Создание сертификатов

На сервере генерируем сертификаты для клиента. Для этого снова переходим в каталог easy-rsa:

```
cd /usr/share/easy-rsa/3
```

Запускаем еще раз vars:

```
. ./vars
```

Создаем клиентский сертификат:

```
./easyrsa gen-req client1 nopass
./easyrsa sign-req client client1
```

Мы должны увидеть запрос на подтверждение намерения выпустить сертификат — вводим **yes**:

```
  Confirm request details: yes
```

** в данном примере будет создан сертификат для **client1**.*
На сервере скопируем ключи во временную директорию, выполнив последовательно 3 команды:

```
mkdir /tmp/keys
cp pki/issued/client1.crt pki/private/client1.key pki/dh.pem pki/ca.crt pki/ta.key /tmp/keys
chmod -R a+r /tmp/keys
```

** сертификаты скопированы в каталог **/tmp** для удобства переноса их на клиентский компьютер.*
Сертификаты готовы для скачивания.

6. На клиенте

В качестве примера, выполним подключение к нашему серверу с компьютера Windows.
Пошагово, выполняем следующие действия:
Заходим на официальную страницу загрузки openvpn и скачиваем клиента для Windows:

Рис. 325

Запускаем скачанный файл и устанавливаем программу, нажимая «Далее».
Переходим в папку *C:\Program Files\OpenVPN\config*.
Копируем в нее файлы ca.crt, client1.crt, client1.key, dh.pem, ta.key из каталога /tmp/keys на сервере, например, при помощи программы WinSCP.
После переноса файлов, не забываем удалить ключи из временного каталога на сервере:

```
rm -R /tmp/keys
```

Возвращаемся к компьютеру с Windows, открываем блокнот от имени администратора и вставляем следующие строки:

```
client
resolv-retry infinite
nobind
remote 192.168.0.15 443
proto udp
dev tun
comp-lzo no
ca ca.crt
cert client1.crt
key client1.key
dh dh.pem
tls-client
tls-auth ta.key 1
float
keepalive 10 120
persist-key
persist-tun
verb 0
```

** где **192.168.0.15 443** — IP-адрес OpenVPN-сервера и порт, на котором он принимает запросы.*
Сохраняем файл с именем **config.ovpn** в папке *C:\Program Files\OpenVPN\config*.
Запускаем с рабочего стола программу «OpenVPN GUI» от имени администратора.
Нажимаем правой кнопкой по появившемуся в трее значку и выбираем «Подключиться»:

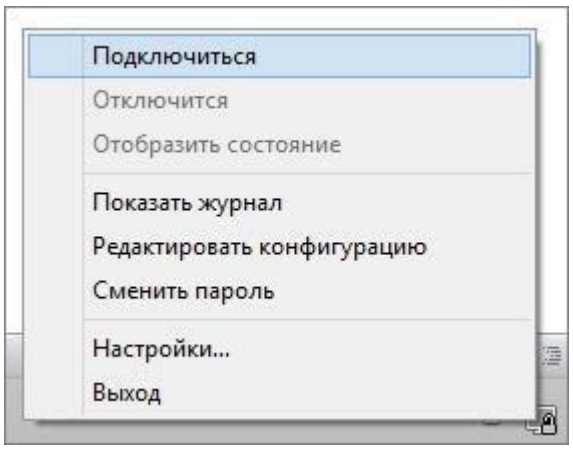


Рис. 326

Произойдет подключение и значок поменяет цвет с серого/желтого на зеленый.
Для автозапуска клиента, открываем службы Windows, находим и настраиваем службу OpenVPNService для автозапуска:

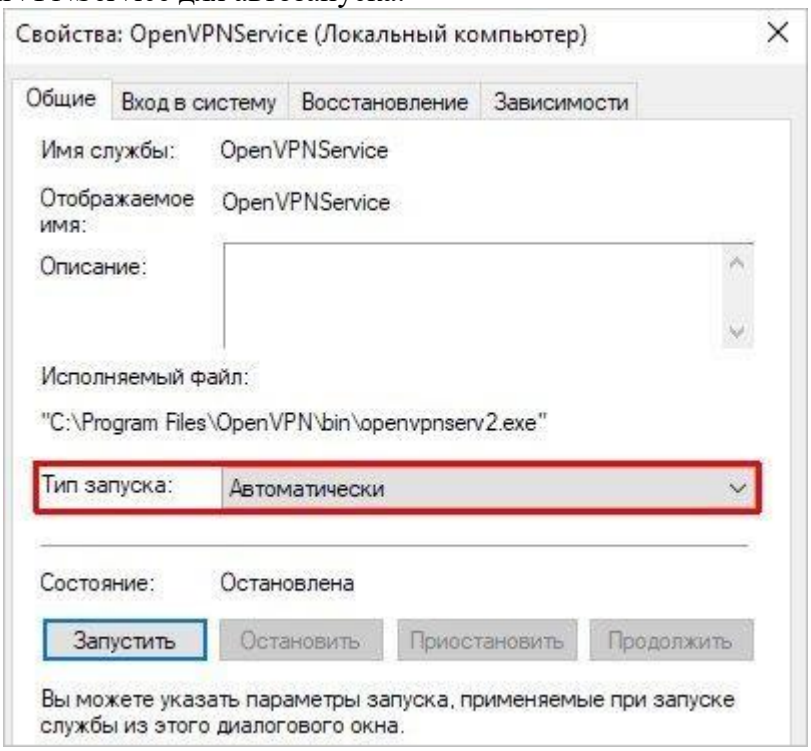


Рис. 327

7. Аутентификация пользователей

Мы можем настроить проверку пользователя по логину и паролю. Это даст дополнительный уровень защиты, а также позволит использовать один и тот же сертификат для всех подключений.

8. Настройка сервера

Открываем конфигурационный файл openvpn:

```
nano /etc/openvpn/server/server.conf
```

И добавляем следующие строчки:

```
username-as-common-name
plugin /usr/lib64/openvpn/plugins/openvpn-plugin-auth-pam.so login
```

** где **username-as-common-name** указывает на то, что openvpn должен использовать логины, как основные идентификаторы клиента; **plugin** указывает на путь к самому плагину и для чего он используется.*

** как путь, так и название файла **openvpn-plugin-auth-pam.so** могут отличаться. Это*

*зависит от версии Linux и OpenVPN. Чтобы найти путь до нужного файла, можно воспользоваться командой **find / -name "openvpn-*auth-pam*" -print**.*

Перезапускаем сервер:

```
systemctl restart openvpn-server@server
```

При необходимости, создаем учетную запись для авторизации:

```
useradd vpn1 -s /sbin/nologin
```

И задаем ей пароль:

```
passwd vpn1
```

9. Настройка на клиенте

В конфигурационный файл клиента добавляем:

```
auth-user-pass
```

Теперь при подключении программа будет запрашивать логин и пароль.

Вход без ввода пароля (сохранение пароля)

Если необходимо настроить авторизацию, но автоматизировать вход клиента, открываем конфигурационный файл последнего и строку для авторизации меняем на:

```
auth-user-pass auth.txt
```

** где **auth.txt** — файл, в котором мы будем хранить логин и пароль.*

Создаем текстовый файл auth.txt в той же папке, где находится файл конфигурации со следующим содержимым:

```
username
password
```

** где **username** — логин пользователя, а **password** — пароль.*

Переподключаем клиента.

Описанный метод аутентификации является базовым и требует наличие обычной системной учетной записи. Если необходима более сложная авторизация на базе LDAP, можно воспользоваться инструкцией настройка OpenVPN сервера с аутентификацией через LDAP (написана на базе Linux Ubuntu).

10. Отзыв сертификата

В случае, когда необходимо прекратить действие определенного сертификата, можно его отозвать (revoke). Чтобы настроить отзыв сертификата в OpenVPN, нужно указать файл для проверки отозванных сертификатов, затем отозвать сам сертификат.

11. Настройка OpenVPN

Открываем конфигурационный файл:

```
nano /etc/openvpn/server/server.conf
```

Добавляем строку:

```
...
crl-verify keys/crl.pem
```

** в данном примере, сервер будет проверять список отозванных сертификатов в файле **keys/crl.pem**.*

12. Отзыв сертификата

В нашем примере мы создали сертификат client1 — сделаем его отзыв. Переходим в каталог:

```
cd /usr/share/easy-rsa/3
```

Отзываем сертификат командой:

```
./easyrsa revoke client1
```

** здесь мы отзываем сертификат для клиента **client1**.*

Подтверждаем наши намерения:

```
  Continue with revocation: yes
```

... и вводим пароль для центра сертификации.

После этого создаем/обновляем файл crl.pem:

```
./easyrsa gen-crl
```

** необходимо будет ввести пароль центра сертификации.*

Копируем файл crl.pem в каталог openvpn:

```
cp pki/crl.pem /etc/openvpn/server/keys/
```

После перезагружаем сервис openvpn:

```
systemctl restart openvpn-server@server
```

13. Доступ клиентам друг к другу

Ранее мы настроили более безопасный сценарий подключения — туннели, которые не позволяют подключенным клиентам видеть друг друга. Но если мы хотим сделать так, чтобы все подключенные к VPN устройства видели друг друга по сети, нам нужно изменить некоторые настройки на сервере и клиенте.

14. Настройка сервера

Открываем файл настроек:

```
nano /etc/openvpn/server/server.conf
```

Добавляем строку:

```
client-to-client
```

** данная настройка как раз и говорит, что клиенты могут видеть друг друга через нашу сеть VPN.*

Теперь находим настройку:

```
dev tun
```

... и меняем ее на:

```
dev tap
```

** туннели создают небольшие подсети на 4 адреса для каждого подключения, таким образом, изолируя клиентов друг от друга. Нам же нужно сделать так, чтобы клиенты были в одной сети VPN. Поэтому мы меняем тип интерфейса на* ***tap*****.**

Перезапускаем нашу службу сервиса:

```
systemctl restart openvpn-server@server
```

15. Настройка клиента

На клиенте нам нужно изменить только тип сетевого интерфейса на tap:

```
dev tap
```

После можно подключаться к серверу.

Сделайте скриншоты (фотографии) процесса установки и настройки OpenVPN и вставьте в отчёт.

Задание 2:

1. Подготовка сервера

Для установки ПО потребуется репозиторий EPEL:

```
dnf install epel-release
```

Настраиваем брандмауэр:

```
firewall-cmd --permanent --add-port=1701/{tcp,udp}
firewall-cmd --permanent --add-service=ipsec
firewall-cmd --reload
```

Отключаем SELinux:

```
setenforce 0
sed -i 's/^SELINUX=.*/SELINUX=disabled/g' /etc/selinux/config
```

2. Настройка VPN-сервера

Для настройки нашего сервера мы настроим следующие компоненты: IPSEC (strongswan), L2TP (xl2tpd), PPP.

3. IPSEC

Для управления IPSec используется пакет strongswan — установим его командой:

```
dnf install strongswan
```

Открываем конфигурационный файл для настройки ipsec:

```
nano /etc/strongswan/ipsec.conf
```

Для **config setup** добавим:

```
config setup
   nat_traversal=yes
   virtual_private=%v4:10.0.0.0/8,%v4:192.168.0.0/16,%v4:172.16.0.0/12
   oe=off
   protostack=netkey
```

** где:*

- ***nat_traversal*** *— обход NAT.*
- ***virtual_private*** *— определяет приватные сети. В данном примере просто перечислены сети, зарезервированные под локальные — мы можем указать и другие.*
- ***oe*** *— <не смог найти описание данного параметра>.*
- ***protostack*** *— определяет стек протоколов, который будет использоваться для подключения.*

... а также вставляем ниже:

```
conn L2TP-PSK-NAT
   rightsubnet=vhost:%priv
   also=L2TP-PSK-noNAT

conn L2TP-PSK-noNAT
   authby=secret
   pfs=no
   auto=add
   keyingtries=3
   rekey=no
   ikelifetime=8h
   keylife=1h
   type=transport
   left=%any
   leftprotoport=udp/1701
   right=%any
   rightprotoport=udp/%any
   ike=aes128-sha1-modp1536,aes128-sha1-modp1024,aes128-md5-modp1536,aes128-md5-
modp1024,3des-sha1-modp1536,3des-sha1-modp1024,3des-md5-modp1536,3des-md5-
modp1024
   esp=aes128-sha1-modp1536,aes128-sha1-modp1024,aes128-md5-modp1536,aes128-md5-
modp1024,3des-sha1-modp1536,3des-sha1-modp1024,3des-md5-modp1536,3des-md5-
modp1024
```

** где:*

- ***authby*** *— способы аутентификации двух узлов. Возможны варианты secret (по паролю) или rsasig (цифровые подписи RSA).*
- ***pfs*** *— расшифровывается как Perfect Forward Secrecy. Позволяет активировать совершенную секретность в канале ключей соединения.*
- ***auto*** *— операция, которая должна запуститься автоматически при старте IPsec.*
- ***keyingtries*** *— число попыток, чтобы «договориться» о соединении или его замене.*
- ***rekey*** *— перепроверить соединение, когда оно истекает.*
- ***ikelifetime*** *— время соединения до повторного согласования ISAKMP или IKE SA.*
- ***keylife*** *— как долго должен длиться конкретный экземпляр соединения.*

- ***type*** *— тип соединения. Возможны варианты tunnel (хост-хост, хост-подсеть или подсеть-подсеть); transport (хост-хост); passthrough (без обработки IPsec).*
- ***left*** *— IP-адрес левого участника (сервера). %any означает, что адрес может быть любой.*
- ***leftprotoport*** *— определяет протокол и порт, на котором будет работать левая сторона (сервер). В данном примере указан UDP и порт 1701.*
- ***right*** *— IP-адрес правого участника (клиента). %any означает, что адрес может быть любой.*
- ***rightprotoport*** *— определяет протокол и порт, на котором будет работать правая сторона (клиент). В данном примере указан UDP и любой порт.*

Создаем секретный ключ — для этого открываем на редактирование файл:

```
nano /etc/strongswan/ipsec.secrets
```

... и добавляем:

```
%any %any : PSK "my_key_password"
```

** в данном примере мы устанавливаем общий пароль **my_key_password** для соединений с любого IP.*

Разрешаем автозапуск strongswan и перезапускаем службу:

```
systemctl enable strongswan
systemctl restart strongswan
```

4. L2TP

Устанавливаем сервер L2TP:

```
dnf install xl2tpd
```

Открываем файл настройки сервера:

```
nano /etc/xl2tpd/xl2tpd.conf
```

Для раздела **[global]** добавляем:

```
[global]
port = 1701
access control = no
ipsec saref = yes
force userspace = yes
auth file = /etc/ppp/chap-secrets
```

где:

- ***port*** *— порт UDP, на котором работает VPN. По умолчанию, 1701.*
- ***access control*** *— принимать или нет запросы только от клиентов с определенными IP, перечисленными в настройках клиентов.*
- ***ipsec saref*** *— указывает использовать или нет ipsec Security Association, позволяющий отслеживать несколько клиентов с одинаковыми IP-адресами.*
- ***force userspace*** *— повышает производительность за счет декапсуляции пакетов L2TP.*
- ***auth file*** *— путь к файлу аутентификации.*

Раздел **[lns default]** можно полностью удалить или закомментировать (символом «;») и заменить на:

```
[lns default]
ip range = 176.16.10.10-176.16.10.200
local ip = 176.16.10.1
require authentication = yes
name = l2tp
pass peer = yes
ppp debug = no
pppoptfile = /etc/ppp/options.xl2tpd
length bit = yes
refuse pap = yes
```

где:

- ***ip range*** *— диапазон адресов, которые назначаются подключенным клиентам.*
- ***local ip*** *— IP-адрес сервера в сети VPN.*
- ***name*** *— имя сервера для процесса согласования.*
- ***pppoptfile*** *— путь к файлу с настройкой pppd.*
- ***flow bit*** *— позволяет добавлять в пакеты порядковые номера.*
- ***exclusive*** *— если поставить в yes, сервер разрешит только одно соединение с клиентом.*
- ***hidden bit*** *— скрывать или нет AVP.*
- ***length bit*** *— использовать ли бит длины, указывающий полезную нагрузку.*
- ***require authentication*** *— требовать ли аутентификацию.*
- ***require chap*** *— требовать ли аутентификацию PPP по протоколу CHAP.*
- ***refuse pap*** *— отказывать ли авторизацию пирам, использующим PAP.*

Разрешаем автозапуск vpn-сервера и перезапускаем его:

```
systemctl enable xl2tpd
systemctl restart xl2tpd
```

5. PPP

Открываем на редактирование конфигурационный файл:

```
nano /etc/ppp/options.xl2tpd
```

Можно закомментировать все, что там есть и вставить:

```
ipcp-accept-local
ipcp-accept-remote
auth
idle 1800
mtu 1200
mru 1200
nodefaultroute
lock
proxyarp
connect-delay 5000
name l2tpd
login
ms-dns 77.88.8.8
ms-dns 8.8.8.8
require-mschap-v2
```

Создаем пользователя. Для этого открываем файл:

```
nano /etc/ppp/chap-secrets
```

И добавляем:

```
user1 * password1 172.16.10.10
user2 * password2 *
user3 l2tpserver password2 *
```

** формат записи — <логин> <имя сервиса> <пароль> <IP клиента (не обязательно)>. Первая учетная запись может подключаться к любому VPN и только с IP* ***172.16.10.10****, вторая — к любому VPN с любого IP, третья — к серверу* ***l2tpserver****, но с любого IP.*

Перезапускаем xl2tpd:

```
systemctl restart xl2tpd
```

6. Настройка клиента

Рассмотрим процесс настройки клиента на базе Windows. Для андроида и устройств Apple параметры заполняются аналогично.

Графический интерфейс

В параметрах сети и Интернет в разделе **VPN** создаем новое соединение:

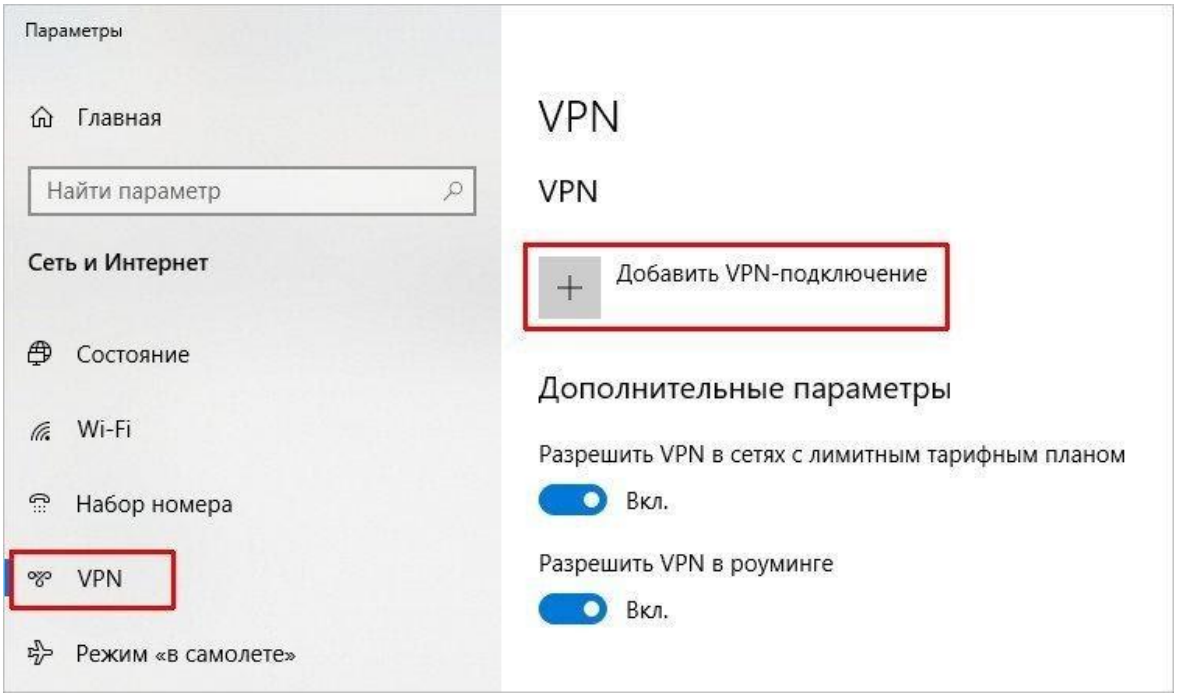


Рис. 328

Задаем настройки:

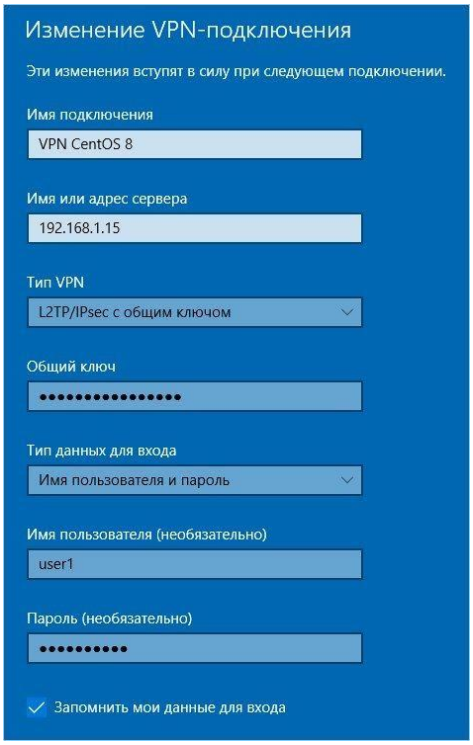


Рис. 329

** где:*

- ***Имя подключения*** *— произвольное имя для подключения.*
- ***Имя или адрес сервера*** *— адрес сервера VPN, к которому мы будем подключаться. В данном примере используем внутреннюю сеть, но в продуктивной среде адрес должен быть внешним.*
- ***Тип VPN*** *— для нашего случая, выбираем L2TP/IPsec с предварительным ключом.*
- ***Общий ключ*** *— ключ, который мы задали в файле /etc/ipsec.secrets.*
- ***Тип данных для входа*** *— выбираем пользователь и пароль.*
- ***Имя пользователя и пароль*** *— логин и пароль, которые мы задали в файле /etc/ppp/chap-secrets.*

Командная строка

Соединение VPN в Windows можно создать с помощью Powershell:

```
Add-VpnConnection -Name "VPN CentOS 8" -ServerAddress "192.168.1.15" -TunnelType "L2tp" -EncryptionLevel "Required" -AuthenticationMethod MsChapv2 -SplitTunneling -DnsSuffix "wbsh.local" -L2tpPsk "my_key_password" -Force -RememberCredential -PassThru
```

** где:*

- ***Name*** *— произвольное имя для подключения.*
- ***ServerAddress*** *— адрес сервера VPN, к которому мы будем подключаться. В данном примере используем внутреннюю сеть, но в продуктивной среде адрес должен быть внешним.*
- ***TunnelType*** *— тип туннеля. В нашем случае это L2TP.*
- ***EncryptionLevel*** *— указание на требование использования шифрованного канала.*
- ***AuthenticationMethod*** *— метод аутентификации. В нашем случае ms-chap-2.*
- ***DnsSuffix*** *— суффикс DNS. Будет автоматически подставляться к коротким именам узлов.*
- ***L2tpPsk*** *— предварительный ключ, который мы задали в файле /etc/ipsec.secrets.*

7. Доступ к локальной сети и сети Интернет

При подключении к нашему серверу VPN у клиента не будет возможности выходить в Интернет и подключаться к ресурсам сети, что делает соединение бессмысленным. Поэтому первым этапом после настройки сервера должна быть настройка маршрутизации сети. Для этого необходимо включить возможность работы в качестве шлюза и настроить правила в брандмауэре.

8. Настройка ядра

Нам нужно разрешить опцию net.ipv4.ip_forward в настройках ядра — для этого откроем файл:

```
nano /etc/sysctl.d/99-sysctl.conf
```

И добавляем в него следующую строку:

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

После применяем настройку:

```
sysctl -p /etc/sysctl.d/99-sysctl.conf
```

В случае с единым сетевым интерфейсом больше ничего делать не потребуется — CentOS начнет работать как Интернет-шлюз.

В случае с несколькими сетевыми адаптерами, настраиваем сетевой экран.

9. Настройка брандмауэра

Настройка выполняется для двух сетевых интерфейсов на примере **ens32** (внутренний) и **ens34** (внешний):

```
firewall-cmd --permanent --zone=public --add-masquerade
firewall-cmd --direct --permanent --add-rule ipv4 filter FORWARD 0 -i ens32 -o ens34 -j ACCEPT
firewall-cmd --reload
```

10. Аутентификация через Active Directory

Проверка подлинности через активный каталог от Microsoft в xl2tp выполняется с помощью winbind и samba.

11. Подготовка сервера

Для корректной работы сервера с Active Directory необходимо задать ему имя (hostname), которое будет доступно в DNS. Также на сервере должно быть задано точное время. Необходимо убедиться, что сервер доступен по своему доменному имени. Если серверу так и не было задано вменяемого имени, вводим команду:

```
hostnamectl set-hostname vpn.wbsh.local
```

** где* ***vpn*** *— имя сервера;* ***wbsh.local*** *— домен.*

После добавляем в DNS наш сервер VPN. Ждем минут 15 (если у нас используется доменная инфраструктура с несколькими сайтами, иначе ждать не нужно).

Задаем временную зону:

```
\cp /usr/share/zoneinfo/Europe/Moscow /etc/localtime
```

** в данном примере мы задаем зону по московскому времени.*

Устанавливаем утилиту для синхронизации времени, разрешаем запуск демона и стартуем его.

```
dnf install chrony
systemctl enable chronyd
systemctl restart chronyd
```

12. Присоединяем сервер к домену

Устанавливаем необходимые компоненты:

```
dnf install samba-client samba-winbind samba-winbind-clients krb5-workstation
```

Открываем конфигурационный файл samba:

```
nano /etc/samba/smb.conf
```

В разделе [global] редактируем следующие опции:

```
workgroup = WBSH
security = ads
```

** где **WBSH** — NETBIOS имя домена; **ads** — указывает, что для samba будет использоваться модель безопасности LDAP Active Directory.*

Также в [global] добавим следующие строки:

```
kerberos method = secrets and keytab
realm = WBSH.LOCAL
winbind enum groups = Yes
winbind enum users = Yes
idmap config * : rangesize = 1000000
idmap config * : range = 1000000-19999999
idmap config * : backend = autorid
```

** где:*

- ***kerberos method** — метод проверки kerberos. В данном примере сначала используется secretts.tdb, а затем системная таблица ключей.*
- ***realm** — сервер Active Directory. В нашем примере прописан домен, так как по нему можно обратиться к любому из серверов AD.*
- ***winbind enum groups** — задает пределы перечисления групп через setgrent(), getgrent() и endgrent().*
- ***winbind enum users** — задает пределы перечисления пользователей через setpwent(), getpwent()и endpwent().*
- ***idmap config * : rangesize** — определяет количество доступных uids и gids в каждом доменном диапазоне.*
- ***idmap config * : range** — определяет доступные совпадающие диапазоны uid и gid, для которых серверная часть является авторитетной.*
- ***idmap config * : backend** — задает idmap плагин для использования в качестве SID/uid/gid подсистемы*

Вводим сервер в домен:

```
net ads join -U Administrator@wbsh.local
```

** где **Administrator** — учетная запись пользователя AD с правами на ввод компьютеров в домен; **wbsh.local** — наш домен.*

Мы должны увидеть, примерно, следующее:

```
Using short domain name -- WBSH
Joined 'SAMBA' to dns domain 'wbsh.local'
```

Разрешаем автозапуск winbind и стартуем его:

```
systemctl enable winbind --now
```

Выбираем профиль для аутентификации:

```
authselect select winbind --force
```

Проверяем, что наш сервер может получить список пользователей Active Directory:

```
wbinfo -u
```

... и групп:

```
wbinfo -g
```

Если мы увидели список пользователей и групп, то присоединение сервера к домену завершено.

После проверяем, что аутентификация в AD через модуль ntlm_auth работает корректно:

```
ntlm_auth --request-nt-key --domain=WBSH.LOCAL --username=Administrator
```

** где **WBSH.LOCAL** — наш домен; **Administrator** — пользователь, под которым будем логиниться для проверки работы модуля.*

13. Настройка PPP для аутентификации через AD

Открываем конфигурационный файл options.xl2tpd:

```
vi /etc/ppp/options.xl2tpd
```

Добавляем в самый низ:

```
...
plugin winbind.so
ntlm_auth-helper '/usr/bin/ntlm_auth --helper-protocol=ntlm-server-1 --require-membership-of="WBSH\\VPN Users"'
```

** где **VPN Users** — группа в AD, пользователи который будут иметь возможность использовать VPN.*

Перезапускаем xl2tpd:

```
systemctl restart xl2tpd
```

14. Проверка

В Active Directory добавляем группу VPN Users (если еще нет). Группа должна быть локальная в домене. В группу добавим пользователей, которым хотим дать доступ для VPN-подключения.

В настройках подключения к серверу меняем пользователя и пароль на доменные.

15. Диагностика проблем

Описанная выше настройка не предполагает наличие лога. Для этого открываем конфигурационный файл для ppp:

```
nano /etc/ppp/options.xl2tpd
```

Добавим:

```
...
logfile /var/log/xl2tpd/xl2tpd.log
debug
```

Создадим каталог для лога:

```
mkdir /var/log/xl2tpd
```

Перезапускаем сервис xl2tpd:

```
systemctl restart xl2tpd
```

Пробуем подключиться к серверу — в случае наличия проблем, наблюдаем за логом:

```
tail -f /var/log/xl2tpd/xl2tpd.log
```

16. Настройка доступа по ssh в centos

SSH-сервер (OpenSSH) позволяет производить удалённое управление операционной системой, а также копирование файлов между компьютерами по шифрованному каналу связи. SSH расшифровывается как Secure Shell. OpenSSH обеспечивает надежную авторизацию и безопасную передачу данных по открытым каналам связи.

Установка SSH-сервера

Для установки SSH-сервера в CentOS необходимо установить пакет openssh-server:

```
# sudo dnf install openssh-server
```

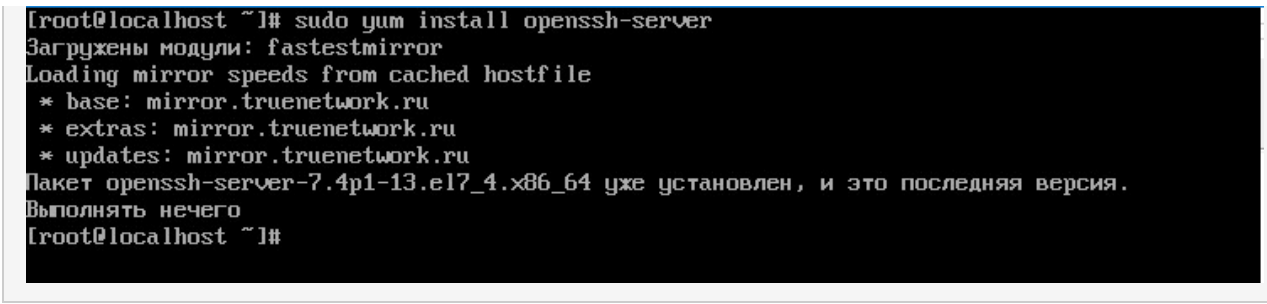


Рис. 330

Добавляем OpenSSH в автозагрузку:

sudo chkconfig sshd on

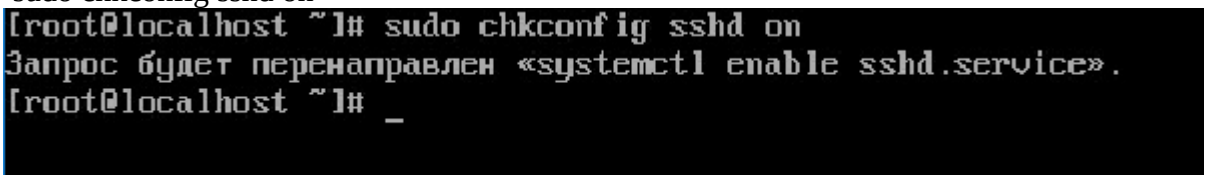


Рис. 331

Для дальнейшей работы нам необходимо запустить сервер OpenSSH.

Запуск OpenSSH:

service sshd start

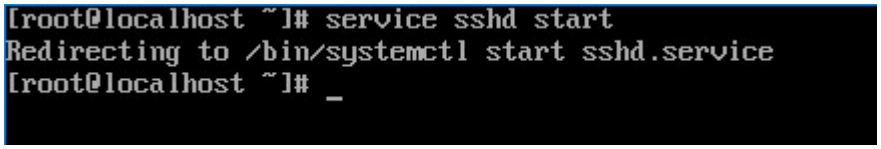


Рис. 332

Настройки SSH-сервера

Настройки SSH-сервера хранятся в файле `/etc/ssh/sshd_config`.

Для приведенного выше примера он может быть следующим:

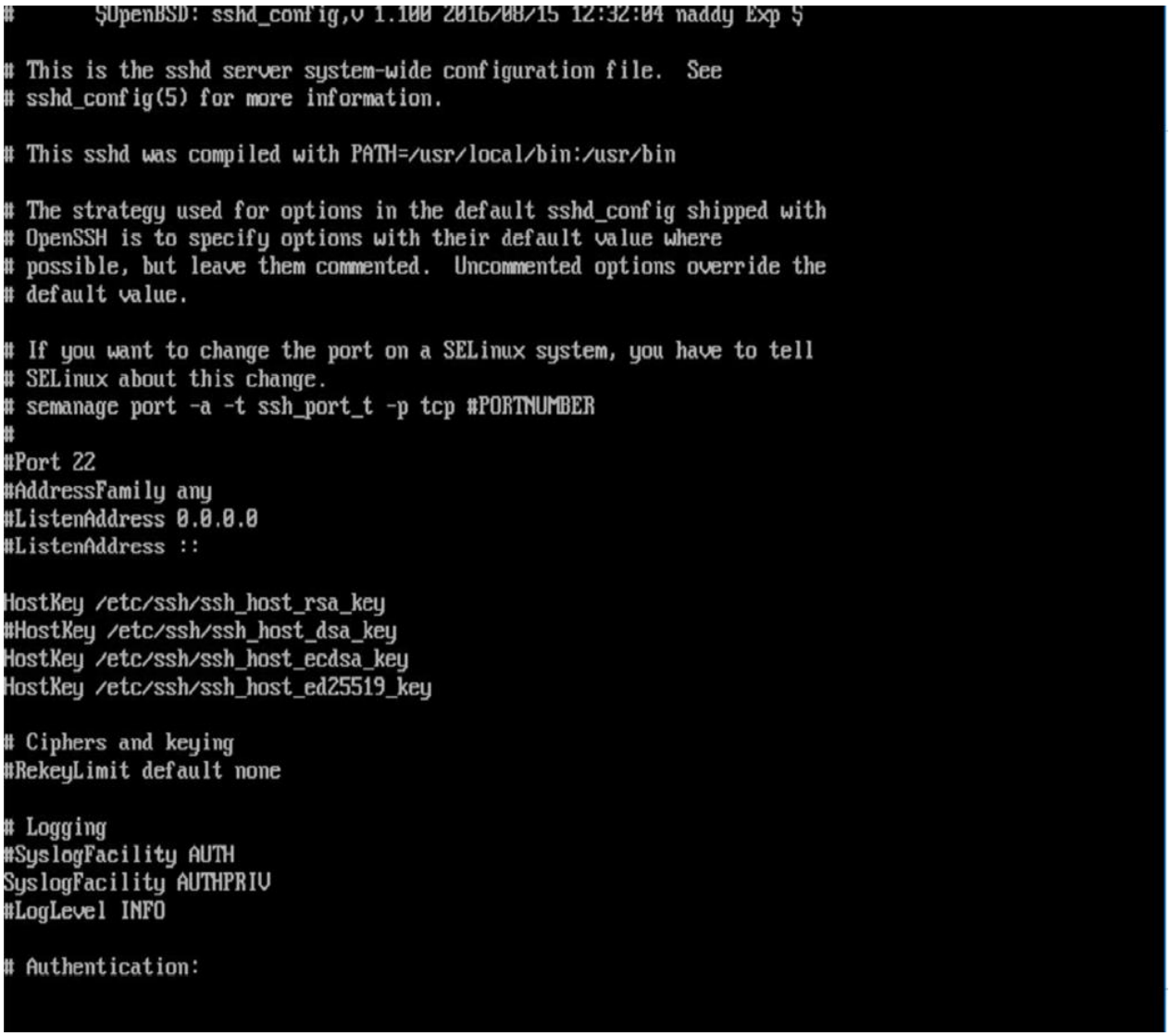


Рис. 333

Наиболее важные опции с точки зрения безопасности:

- Port 22 – Порт по умолчанию.
- Protocol 2,1 – Какая реализация протокола SSH будет использоваться. Рекомендую оставить только 2.
- ListenAddress – По умолчанию SSH сервер прослушивает все доступные интерфейсы, что абсолютно не нужно в большинстве ситуаций. Необходимо прописать сетевой интерфейс, с которого будет осуществляться управление сервером.
- PermitRootLogin yes – По умолчанию разрешает входить по SSH суперпользователю root. Необходимо указать no.
- AllowUsers adminsys – Данный параметр разрешает входить по SSH только перечисленным пользователям.
- AllowGroups wheel – Группа пользователей которой можно входить по SSH, опцию можно не использовать если указана опция AllowUsers.
- DenyUsers baduser – Данная опция запрещает вход по SSH перечисленным пользователям.
- DenyGroups badgroup – Данная опция запрещает вход по SSH перечисленным группам пользователей.
- MaxAuthTries 3 – Сколько раз переспрашивать пароль при неверном вводе. В данном случае SSH-сервер после 3 неверных попыток разорвет соединение с клиентом.

- LoginGraceTime 60 – Через сколько секунд разрывать соединение при отсутствии аутентификации со стороны клиента.
- PermitEmptyPasswords no – Разрешать использовать пустые пароли. По вполне понятным причинам значение этого параметра no.
- PrintLastLog yes – при входе пользователя в систему по SSH ему будет показано когда и откуда последний раз был произведен вход под данным пользователем.
- LogLevel INFO – В качестве параметра этой опции необходимо указать уровень журналирования. Возможные значения QUIET, FATAL, ERROR, INFO, VERBOSE, DEBUG1, DEBUG2, DEBUG3. Чем выше уровень журналирования, тем больше информации появится в файле регистрации событий.
- SyslogFacility AUTHPRIV – Куда будут попадать логи. Возможные значения: DAEMON, USER, AUTH, LOCAL0, LOCAL1, LOCAL2, LOCAL3, LOCAL4, LOCAL5, LOCAL6, LOCAL7.

Вход на сервер через PUTTY PORTABLE

Для начала нам нужно узнать IP-адрес сервера:

ifconfig

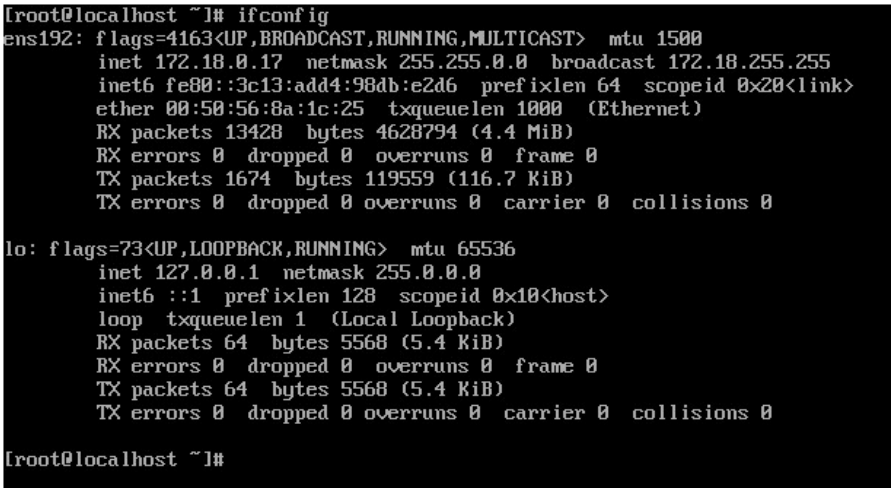


Рис. 334

После чего вводим IP-адрес в PUTTY указав 22 порт (стоит по умолчанию). Указываем SSH соединение и заходим на сервер.

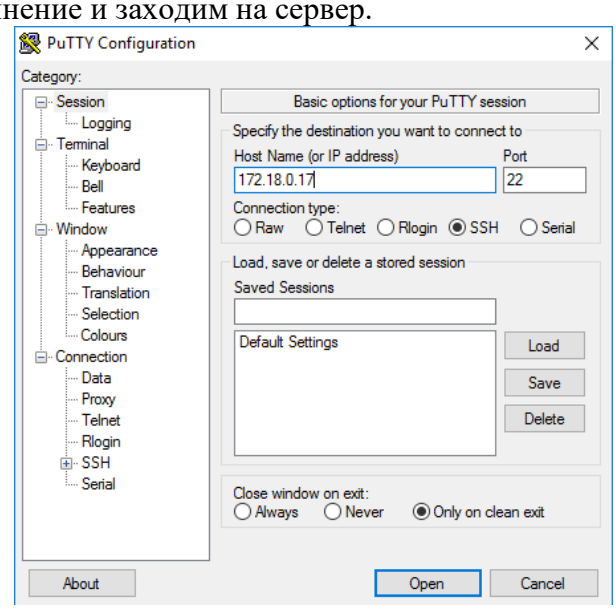


Рис. 335

При входе на сервер он попросит вас зайти под своей учетной записью.

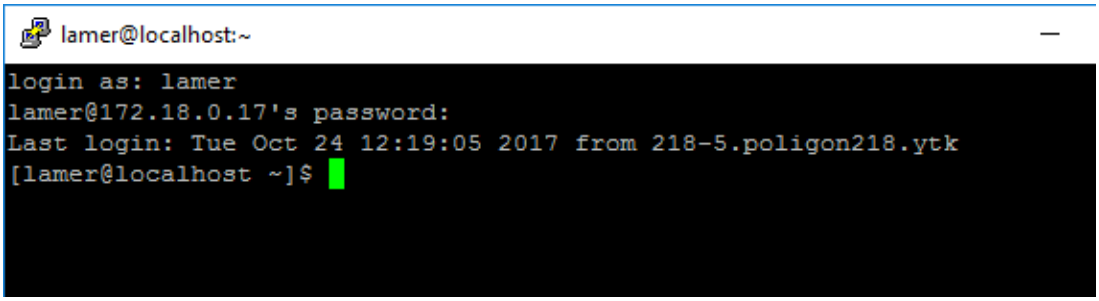


Рис. 336

Сделайте скриншоты (фотографии) процесса настройки протокола IP-sec и SSH и вставьте в отчёт.

2.23 Практическая работа№ 23
Настройка регистрации действий. Установка и настройка OpenLDAP

Задание 1:

В данной практической работе мы рассмотрим основы работы с rsyslog. В примерах мы будем использовать следующие узлы:

Сервер:192.168.241.140

Клиент: 172.31.21.58

1. Установка и настройка сервера Rsyslog

В большинстве дистрибутивов Linux пакет rsyslog предустановлен. Если у вас его нет, установите его при помощи менеджера пакетов:

Для RHEL/CentOS:

```
$ dnf install rsyslog
```

После установки rsyslog нужно запустить службу, активировать автоматический запуск при загрузке и проверить состояние при помощи команды systemctl.

```
$ sudo systemctl start rsyslog
$ sudo systemctl enable rsyslog
$ sudo systemctl status rsyslog
```

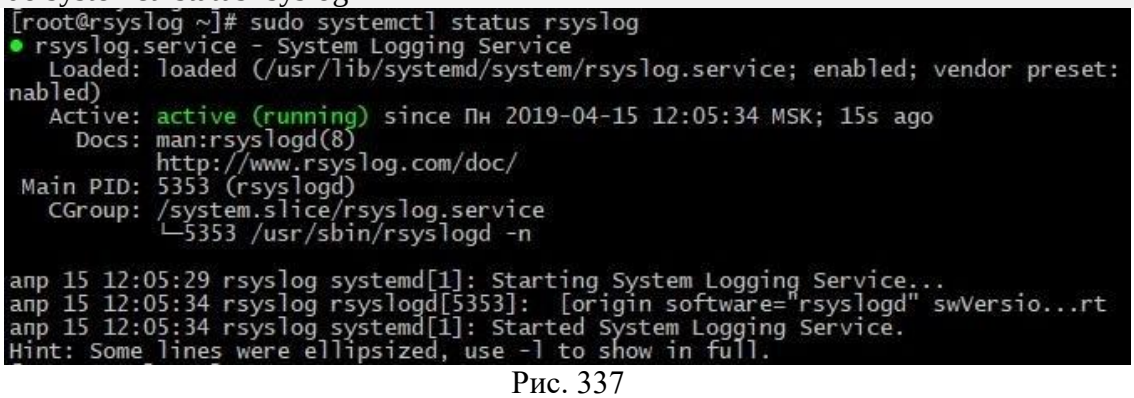


Рис. 337

Главный файл конфигурации rsyslog — /etc/rsyslog.conf. Он загружает модули, определяет глобальные директивы, содержит правила по обработке сообщений логов, а также включает пути ко всем файлам конфигурации в директории /etc/rsyslog.d/ для различных приложений и служб.

```
$ sudo nano /etc/rsyslog.conf
```

По умолчанию rsyslog использует модули imjournal и imusock для импорта структурированных записей логов из журнала systemd и для приема через сокетов Unix сообщений системных логов от приложений, запущенных в локальной системе, соответственно.

Чтобы настроить rsyslog как сетевой централизованный сервер ведения логов, нужно установить протоколы (UDP, TCP или оба), которые будут использоваться для приема удаленных сообщений системных логов, а также прослушиваемые порты.

Если вы хотите использовать UDP-соединение, более быстрое, но ненадежное, найдите в файле конфигурации следующие строки, раскомментируйте их и замените 514 на порт, который вы хотите прослушивать. Этот порт должен соответствовать порту, на который клиенты будут отправлять сообщения, мы рассмотрим это ниже при настройке клиента rsyslog:

```
$ModLoad imudp
$UDPServerRun 514
```

Для использования TCP-соединения (медленнее, но надежнее) найдите и раскомментируйте следующие строки:

```
$ModLoad imtcp
$InputTCPServerRun 514
```

В нашем случае мы будем использовать оба протокола.

Далее вам потребуется определить набор правил для обработки удаленных логов в следующем формате:

```
источник.уровень_важности место_записи_лога
```

источник: тип процесса или приложения, от которого исходит сообщение, значение может быть auth, cron, daemon, kernel, local0..local7. Использование звездочки (*) означает все источники.

уровень_важности: тип сообщения логов: emerg-0, alert-1, crit-2, err-3, warn-4, notice-5, info-6, debug-7. Использование звездочки означает все уровни важности, если ничего не указывать, предполагается отсутствие уровня важности.

- 0, emerg – система не работоспособна
- 1, alert – система требует немедленного вмешательства
- 2, crit – состояние системы критическое
- 3, err – сообщение об ошибке
- 4, warning – предупреждение о возможной проблеме
- 5, notice – нормальное, но важное событие
- 6, info – информационное сообщение
- 7 , debug – отладочное сообщение
- место_записи_лога: локальный файл или удаленный сервер rsyslog (определенный в формате IP-адрес:порт).

Для сбора логов удаленных узлов мы будем использовать следующий набор правил с шаблоном RemoteLogs. Обратите внимание, что эти правила должны предшествовать правилам обработки локальных сообщений.

```
$template RemoteLogs,"/var/log/%HOSTNAME%/%PROGRAMNAME%.log"
*.* ?RemoteLogs
& ~
```

Рассмотрим набор правил более подробно. Первое правило в нем следующее: **«$template RemoteLogs,”/var/log/%HOSTNAME%/%PROGRAMNAME%.log”»**.

Директива *$template* дает демону rsyslog команду собирать полученные сообщения из источников и записывать их в отдельные логи в директории `/var/logs` в соответствии с именем узла (машины клиента) и источником (программой/приложением), от которых были получены сообщения, что определено соответствующим шаблоном.

Вторая строка **«*.* ?RemoteLogs»** означает запись сообщений всех уровней важности от всех источников в соответствии с шаблоном *RemoteLogs*.

Последняя строка **«& ~»** задает rsyslog прекратить обработку сообщений после их записи в файл. Если не указать «& ~», сообщения будут записаны в локальные файлы. Настройка сервера для нашего примера завершена. Теперь нужно сохранить и закрыть

файл конфигурации, а также перезапустить демон rsyslog, чтобы изменения вступили в силу:

```
$ sudo systemctl restart rsyslog
```

Далее требуется проверить сетевые сокеты rsyslog. Воспользуйтесь netstat

```
$ netstat -nap | grep "rsyslog"
```

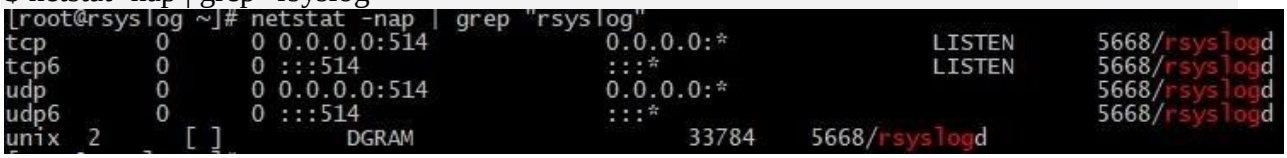


Рис. 338

Если у вас включена служба SELinux, нужно выполнить следующие команды, чтобы разрешить трафик rsyslog:

```
$ sudo semanage -a -t syslogd_port_t -p udp 514
$ sudo semanage -a -t syslogd_port_t -p tcp 514
```

При включенном брандмауэре нужно открыть TCP и UDP порты 514, чтобы разрешить подключение к серверу rsyslog по обоим протоколам:

Для CentOS (брандмауэр firewalld):

```
$ sudo firewall-cmd --permanent --add-port=514/udp
$ sudo firewall-cmd --permanent --add-port=514/tcp
$ sudo firewall-cmd --reload
```

Для Ubuntu (брандмауэр ufw):

```
$ sudo ufw allow 514/udp
$ sudo ufw allow 514/tcp
$ sudo ufw reload
```

2. Настройка клиента Rsyslog для отправки логов на сервер

Проверьте, запущена ли служба rsyslog на клиентской машине, при помощи следующей команды:

```
$ sudo systemctl status rsyslog
```

Если она не установлена, установите и запустите службу точно так же, как для сервера: После запуска службы откройте файл конфигурации:

```
$ sudo nano /etc/rsyslog.conf
```

Чтобы демон rsyslog работал как клиент и отправлял все локальные логи на удаленный сервер rsyslog, добавьте следующее правило перенаправления в конце файла, как показано на скриншоте ниже. Номер порта должен соответствовать номеру порта, прописанному в конфигурации сервера:

```
*. * @@192.168.100.10:514
```

Приведенное правило будет отправлять сообщения всех уровней важности от всех источников. Для отправки сообщений от конкретного источника, например, auth, воспользуйтесь следующим правилом:

```
auth. * @@192.168.100.10:514
```

Сохраните и закройте файл, а также перезагрузите службу rsyslog чтобы изменения вступили в силу.

```
$ sudo systemctl restart rsyslog
```

3. Мониторинг логов на сервере

Последний этап – проверить, действительно ли rsyslog получает сообщения от клиента и сохраняет их в директории /var/log и формате имя_узла/имя_программы.log. Выполните команду ls, чтобы получить список файлов директории логов и проверьте, есть ли там директории под названием ip-172.31.21.58 (или с соответствующим именем узла вашего клиента).

```
$ ls -l /var/log/
```

Если директория существует, проверьте файлы логов в ней следующей командой:

```
$ sudo ls -l /var/log/ip-172-31-21-58/
```

Сделайте скриншоты (фотографии) процесса настройки регистрации действийи вставьте в отчёт.

Задание 2:

1. Установка Open LDAP на CentOS

Установите openldap, openldap-серверы, openldap-клиенты и миграционные *инструменты* из *YUM* .

```
[root@localhost]# yum -y install openldap openldap-servers openldap-clients
migration tools
 Loaded plugins: fastestmirror, langpacks
 updates
 | 3.4 kB  00:00:00
 updates/7/x86_64/primary_db
 | 2.2 MB  00:00:05
 Determining fastest mirrors
 (1/2): extras/7/x86_64/primary_db
 | 121 kB  00:00:01
 (2/2): base/7/x86_64/primary_db
 | 5.6 MB  00:00:16
 Package openldap-2.4.40-13.el7.x86_64 already installed and latest version
 Resolving Dependencies
 --> Running transaction check
 ---> Package openldap-clients.x86_64 0:2.4.40-13.el7 will be installed
 ---> Package openldap-servers.x86_64 0:2.4.40-13.el7 will be installed
 --> Finished Dependency Resolution
 base/7/x86_64/group_gz
 | 155 kB  00:00:00

 Dependencies Resolved

================================================================================
================================================================================
Package                    Arch
Version                    Repository                 Size
================================================================================
================================================================================
Installing:
openldap-clients           x86_64
2.4.40-13.el7              base                       188 k
openldap-servers           x86_64
2.4.40-13.el7              base                       2.1 M

Transaction Summary
================================================================================
================================================================================
Install  2 Packages

Total download size: 2.3 M
Installed size: 5.3 M
Downloading packages:

Installed:
openldap-clients.x86_64 0:2.4.40-13.el7
openldap-servers.x86_64 0:2.4.40-13.el7
Complete!
[root@localhost]#
```

Теперь давайте запустим и *включим* сервис *slapd* —

```
[root@centos]# systemctl start slapd
[root@centos]# systemctl enable  slapd
```

Теперь давайте убедимся, что у нас есть структура *openldap* в */ etc / openldap* .

```
root@localhost]# ls /etc/openldap/
certs  check_password.conf  ldap.conf  schema  slapd.d
[root@localhost]#
```

Затем убедитесь, что наш сервис slapd запущен.

```
root@centos]# netstat -antup | grep slapd
tcp        0      0 0.0.0.0:389             0.0.0.0:*               LISTEN      1641/slapd
tcp6       0      0 :::389                  :::*                    LISTEN      1641/slapd

[root@centos]#
```

Далее, давайте настроим нашу установку *Open LDAP* .
Убедитесь, что наш системный пользователь *ldap* создан.

```
[root@localhost]# id ldap
uid=55(ldap) gid=55(ldap) groups=55(ldap)
[root@localhost]#
```

Создайте наши учетные данные LDAP.

```
[root@localhost]# slappasswd
New password:
Re-enter new password:
{SSHA}20RSyjVv6S6r43DFPeJgASDLlLoSU8g.a10

[root@localhost]#
```

Нам нужно сохранить вывод из slappasswd.

2. Настройка Open LDAP

Шаг 1 — Настройте LDAP для домена и добавьте администратора.
Во-первых, мы хотим настроить нашу среду openLDAP. Ниже приведен шаблон для использования с командой *ldapmodify* .

```
dn: olcDatabase={2}hdb,cn=config
changetype: modify
replace: olcSuffix
olcSuffix: dc=vmnet,dc=local
dn: olcDatabase = {2}hdb,cn=config
changetype: modify
replace: olcRootDN
olcRootDN: cn=ldapadm,dc=vmnet,dc=local
dn: olcDatabase = {2}hdb,cn=config
changetype: modify
replace: olcRootPW
olcRootPW: <output from slap
```

Внесите изменения в /etc/openldap/slapd.d/cn=config/olcDatabase = {1} monitor.ldif с помощью команды ldapmodify.

```
[root@localhost]# ldapmodify -Y EXTERNAL -H ldapi:/// -f /home/rdc/Documents/db.ldif
SASL/EXTERNAL authentication started
SASL username: gidNumber = 0+uidNumber = 0,cn=peercred,cn=external,cn=auth
SASL SSF: 0
modifying entry "olcDatabase = {2}hdb,cn=config"
modifying entry "olcDatabase = {2}hdb,cn=config"
modifying entry "olcDatabase = {2}hdb,cn=config"
```

```
[root@localhost cn=config]#
```

Давайте проверим измененную конфигурацию LDAP.

```
root@linux1 ~]# vi /etc/openldap/slapd.d/cn=config/olcDatabase={2}hdb.ldif

[root@centos]# cat /etc/openldap/slapd.d/cn\=config/olcDatabase\=\{2\}hdb.ldif
 # AUTO-GENERATED FILE - DO NOT EDIT!! Use ldapmodify.
 # CRC32 a163f14c
dn: olcDatabase = {2}hdb
objectClass: olcDatabaseConfig
objectClass: olcHdbConfig
olcDatabase: {2}hdb
olcDbDirectory: /var/lib/ldap
olcDbIndex: objectClass eq,pres
olcDbIndex: ou,cn,mail,surname,givenname eq,pres,sub
structuralObjectClass: olcHdbConfig
entryUUID: 1bd9aa2a-8516-1036-934b-f7eac1189139
creatorsName: cn=config
createTimestamp: 20170212022422Z
olcSuffix: dc=vmnet,dc=local
olcRootDN: cn=ldapadm,dc=vmnet,dc=local
olcRootPW:: e1NTSEF1bUVyb1VzZTRjc2dkYVdGaDY0T0k =
entryCSN: 20170215204423.726622Z#000000#000#000000
modifiersName: gidNumber = 0+uidNumber = 0,cn=peercred,cn=external,cn=auth
modifyTimestamp: 20170215204423Z

[root@centos]#
```

Как вы можете видеть, наши модификации LDAP предприятия были успешными.
Далее мы хотим создать самозаверяющий ssl-сертификат для OpenLDAP. Это защитит связь между корпоративным сервером и клиентами.
Шаг 2 — Создайте самоподписанный сертификат для OpenLDAP.
Мы будем использовать *openssl* для создания ssl-сертификата с собственной подписью. Перейдите к следующей главе « **Создание сертификата SSL LDAP с помощью openssl»,** чтобы получить инструкции по обеспечению безопасности связи с OpenLDAP. Затем, когда SSL-сертификаты будут настроены, мы завершим нашу корпоративную конфигурацию OpenLDAP.
Шаг 3 — Настройте OpenLDAP для использования безопасной связи с сертификатом.
Создайте файл *certs.ldif* в *vim* со следующей информацией —

```
dn: cn=config
changetype: modify
replace: olcTLSCertificateFile
olcTLSCertificateFile: /etc/openldap/certs/yourGeneratedCertFile.pem

dn: cn=config
changetype: modify
replace: olcTLSCertificateKeyFile
olcTLSCertificateKeyFile: /etc/openldap/certs/youGeneratedKeyFile.pem
```

Затем снова используйте команду *ldapmodify* для объединения изменений в конфигурацию OpenLDAP.

```
[root@centos rdc]# ldapmodify -Y EXTERNAL  -H ldapi:/// -f certs.ldif
SASL/EXTERNAL authentication started
SASL username: gidNumber = 0+uidNumber = 0,cn=peercred,cn=external,cn=auth
SASL SSF: 0
modifying entry "cn=config"

[root@centos]#
```

Наконец, давайте проверим нашу конфигурацию OpenLADP.

```
[root@centos]# slaptest -u
config file testing succeeded
[root@centos]#
```

Шаг 4 — Настройте базу данных slapd.

```
cp /usr/share/openldap-servers/DB_CONFIG.example /var/lib/ldap/DB_CONFIG &&
chown ldap:ldap /var/lib/ldap/*
```

Обновляет схему OpenLDAP.
Добавьте косинус и nis схемы LDAP.

```
ldapadd -Y EXTERNAL -H ldapi:/// -f /etc/openldap/schema/cosine.ldif
ldapadd -Y EXTERNAL -H ldapi:/// -f /etc/openldap/schema/nis.ldif
ldapadd -Y EXTERNAL -H ldapi:/// -f /etc/openldap/schema/inetorgperson.ldif
```

Наконец, создайте схему предприятия и добавьте ее в текущую конфигурацию OpenLDAP.
Следующее для домена под названием vmnet. *локальный* с администратором LDAP под названием *ldapadm* .

```
dn: dc=vmnet,dc=local
dc: vmnet
objectClass: top
objectClass: domain

dn: cn=ldapadm ,dc=vmnet,dc=local
objectClass: organizationalRole
cn: ldapadm
description: LDAP Manager

dn: ou = People,dc=vmnet,dc=local
objectClass: organizationalUnit
ou: People

dn: ou = Group,dc=vmnet,dc=local
objectClass: organizationalUnit
ou: Group
```

Наконец, импортируйте это в текущую схему OpenLDAP.

```
[root@centos]# ldapadd -x -W -D "cn=ldapadm,dc=vmnet,dc=local" -f ./base.ldif
 Enter LDAP Password:
adding new entry "dc=vmnet,dc=local"

adding new entry "cn=ldapadm ,dc=vmnet,dc=local"

adding new entry "ou=People,dc=vmnet,dc=local"

adding new entry "ou=Group,dc=vmnet,dc=local"

[root@centos]#
```

Шаг 5 — Настройка пользователей OpenLDAP Enterprise.
Откройте *vim* или ваш любимый текстовый редактор и скопируйте следующий формат. Это настройка для пользователя с именем «entacct» в домене LDAP «vmnet.local».

```
dn: uid=entacct,ou=People,dc=vmnet,dc=local
objectClass: top
objectClass: account
objectClass: posixAccount
```

```
objectClass: shadowAccount
cn: entacct
uid: entacct
uidNumber: 9999
gidNumber: 100
homeDirectory: /home/enyacct
loginShell: /bin/bash
gecos: Enterprise User Account 001
userPassword: {crypt}x
shadowLastChange: 17058
shadowMin: 0
shadowMax: 99999
shadowWarning: 7
```

Теперь импортируйте вышеуказанные файлы, как сохраненные, в схему OpenLdap.

```
[root @ centos] # ldapadd -x -W -D "cn = ldapadm, dc = vmnet, dc = local" -f entuser.ldif
 Введите пароль LDAP:
добавление новой записи "uid = entacct, ou = People, dc = vmnet, dc = local"

[Корень @ CentOS] #
```

Прежде чем пользователи смогут получить доступ к LDAP Enterprise, нам нужно назначить пароль следующим образом:

```
ldappasswd -s password123 -W -D "cn=ldapadm,dc=entacct,dc=local" -x "uid=entacct
,ou=People,dc=vmnet,dc=local"
```

-s указывает пароль для пользователя
-x — имя пользователя, к которому применяется обновленный пароль
-D — это * отличительное имя для аутентификации по схеме LDAP.
Наконец, прежде чем войти в учетную запись Enterprise, давайте проверим нашу запись *OpenLDAP* .

```
[root@centos rdc]# ldapsearch -x cn=entacct -b dc=vmnet,dc=local
 # extended LDIF
 #
 # LDAPv3
 # base <dc=vmnet,dc=local> with scope subtree
 # filter: cn=entacct
 # requesting: ALL
 #
 # entacct, People, vmnet.local
dn: uid=entacct,ou=People,dc=vmnet,dc=local
objectClass: top
objectClass: account
objectClass: posixAccount
objectClass: shadowAccount
cn: entacct
uid: entacct
uidNumber: 9999
gidNumber: 100
homeDirectory: /home/enyacct
loginShell: /bin/bash
gecos: Enterprise User Account 001
userPassword:: e2NyeXB0fXg=
shadowLastChange: 17058
shadowMin: 0
shadowMax: 99999
shadowWarning: 7
```

Преобразование таких вещей, как */ etc / passwd* и */ etc / groups* в аутентификацию OpenLDAP, требует использования инструментов миграции. Они включены в пакет migtools. Затем устанавливается в */ usr / share /igrationtools* .

```
[root@centos openldap-servers]# ls -l /usr/share/migrationtools/
total 128
-rwxr-xr-x. 1 root root  2652 Jun  9  2014 migrate_aliases.pl
-rwxr-xr-x. 1 root root  2950 Jun  9  2014 migrate_all_netinfo_offline.sh
-rwxr-xr-x. 1 root root  2946 Jun  9  2014 migrate_all_netinfo_online.sh
-rwxr-xr-x. 1 root root  3011 Jun  9  2014 migrate_all_nis_offline.sh
-rwxr-xr-x. 1 root root  3006 Jun  9  2014 migrate_all_nis_online.sh
-rwxr-xr-x. 1 root root  3164 Jun  9  2014 migrate_all_nisplus_offline.sh
-rwxr-xr-x. 1 root root  3146 Jun  9  2014 migrate_all_nisplus_online.sh
-rwxr-xr-x. 1 root root  5267 Jun  9  2014 migrate_all_offline.sh
-rwxr-xr-x. 1 root root  7468 Jun  9  2014 migrate_all_online.sh
-rwxr-xr-x. 1 root root  3278 Jun  9  2014 migrate_automount.pl
-rwxr-xr-x. 1 root root  2608 Jun  9  2014 migrate_base.pl
```

Шаг 6 — Наконец, нам нужно разрешить доступ к сервису slapd, чтобы он мог обслуживать запросы.

```
firewall-cmd --permanent --add-service=ldap
firewall-cmd --reload
```

3. Настройте клиентский доступ LDAP

Настройка клиентского доступа LDAP требует наличия следующих пакетов на клиенте: клиенты openldap, open-ldap и nss_ldap.

Настройка аутентификации LDAP для клиентских систем немного проще.

Шаг 1 — Установите зависимые пакеты —

```
# yum install -y openldap-clients nss-pam-ldapd
```

Шаг 2 — Настройте аутентификацию LDAP с помощью *authconfig* .

```
authconfig --enableldap --enableldapauth --ldapserver=10.25.0.1 --
ldapbasedn="dc=vmnet,dc=local" --enablemkhomedir --update
```

Шаг 3 — Перезапустите службу nslcd.

Сделайте скриншоты (фотографии) процесса настройки сервера LDAP и вставьте в отчёт.

2.24 Практическая работа№ 24
Установка и настройка IPtables. Поиск уязвимостей информационных систем

Задание 1:

1. Отключение firewalld

Первым делом отключим firewalld, который присутствует в centos 7 по-умолчанию сразу после установки:

```
# systemctl stop firewalld
```

Теперь удалим его из автозагрузки, чтобы он не включился снова после рестарта:

```
# systemctl disable firewalld
```

После этого на сервере настройки сетевого экрана становятся полностью открытыми. Посмотреть правила iptables можно командой:

```
# iptables -L -v -n
```

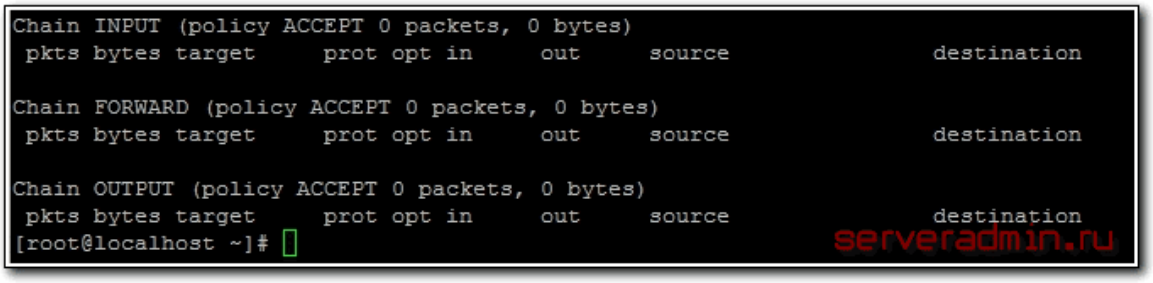


Рис. 339

Дальше пойдет информация исключительно по конфигурированию только iptables. Темы firewalld я больше касаться не буду.

2. Установка iptables

На самом деле фаервол у нас на сервере уже стоит и работает, просто нет никаких правил, все открыто. Установить нам нужно будет дополнительные утилиты управления, без которых конфигурировать iptables невозможно. Например, нельзя будет перезапустить фаервол:

```
# systemctl restart iptables.service
Failed to issue method call: Unit iptables.service failed to load: No such file or directory.
```

Или добавить в автозапуск не получится:

```
# systemctl enable iptables.service
Failed to issue method call: No such file or directory
```

Чтобы подобных ошибок не было, установим необходимый пакет с утилитами:

```
# yum -y install iptables-services
```

Теперь можно добавить iptables в автозагрузку и запустить:

```
# systemctl enable iptables.service
# systemctl start iptables.service
```

2. Настройка фаервола

Для управления правилами фаервола я использую скрипт. Создадим его:

```
# mcedit /etc/iptables.sh
```

Далее будем наполнять его необходимыми правилами. Я буду разбирать все значимые части скрипта, а **полностью его приведу в виде текстового файла в конце статьи**. Правила сделаны в виде картинок, чтобы запретить копирование и вставку. Это может привести к ошибкам в работе правил, с чем я сам столкнулся во время подготовки статьи.

Мы рассмотрим ситуацию, когда сервер является шлюзом в интернет для локальной сети.

Первым делом зададим все переменные, которые будем использовать в скрипте. Это не обязательно делать, но рекомендуется, потому что удобно переносить настройки с сервера на сервер. Достаточно будет просто переназначить переменные.

```
#!/bin/bash
export IPT="iptables"
# Внешний интерфейс
export WAN=eth0
export WAN_IP=85.31.203.127
# Локальная сеть
export LAN1=eth1
export LAN1_IP_RANGE=10.1.3.0/24
```

Перед применением новых правил, очищаем все цепочки:

```
$IPT -F
$IPT -F -t nat
$IPT -F -t mangle
$IPT -X
$IPT -t nat -X
$IPT -t mangle -X
```

Блокируем весь трафик, который не соответствует ни одному из правил:

```
$IPT -P INPUT DROP
$IPT -P OUTPUT DROP
$IPT -P FORWARD DROP
```

Разрешаем весь трафик локалхоста и локалки:

```
$IPT -A INPUT -i lo -j ACCEPT
$IPT -A INPUT -i $LAN1 -j ACCEPT
$IPT -A OUTPUT -o lo -j ACCEPT
$IPT -A OUTPUT -o $LAN1 -j ACCEPT
```

Разрешаем делать ping:

```
$IPT -A INPUT -p icmp --icmp-type echo-reply -j ACCEPT
$IPT -A INPUT -p icmp --icmp-type destination-unreachable -j ACCEPT
$IPT -A INPUT -p icmp --icmp-type time-exceeded -j ACCEPT
$IPT -A INPUT -p icmp --icmp-type echo-request -j ACCEPT
```

Если вам это не нужно, то не добавляйте разрешающие правила для icmp.
Открываем доступ в инет самому серверу:

```
$IPT -A OUTPUT -o $WAN -j ACCEPT
```

Если вы хотите открыть все входящие соединения сервера, то добавляйте дальше правило:

```
$IPT -A INPUT -i $WAN -j ACCEPT
```

Делать это не рекомендуется, привожу просто для примера, если у вас появится такая необходимость.
Дальше разрешим все установленные соединения и дочерние от них. Так как они уже установлены, значит прошли через цепочки правил, фильтровать их еще раз нет смысла:

```
$IPT -A INPUT -p all -m state --state ESTABLISHED,RELATED -j ACCEPT
$IPT -A OUTPUT -p all -m state --state ESTABLISHED,RELATED -j ACCEPT
$IPT -A FORWARD -p all -m state --state ESTABLISHED,RELATED -j ACCEPT
```

Теперь добавим защиту от наиболее распространенных сетевых атак. Сначала отбросим все пакеты, которые не имеют никакого статуса:

```
$IPT -A INPUT -m state --state INVALID -j DROP
$IPT -A FORWARD -m state --state INVALID -j DROP
```

Блокируем нулевые пакеты:

```
$IPT -A INPUT -p tcp --tcp-flags ALL NONE -j DROP
```

Закрываемся от syn-flood атак:

```
$IPT -A INPUT -p tcp ! --syn -m state --state NEW -j DROP
$IPT -A OUTPUT -p tcp ! --syn -m state --state NEW -j DROP
```

Следом за этими правилами рекомендуется поставить правила на запрет доступа с определенных IP, если у вас имеется такая необходимость. Например, вам надоел адрес 84.122.21.197 брутом ssh. Блокируем его:

```
$IPT -A INPUT -s 84.122.21.197 -j REJECT
```

Если вы не ставите ограничений на доступ из локальной сети, то разрешаем всем выход в интернет:

```
$IPT -A FORWARD -i $LAN1 -o $WAN -j ACCEPT
```

Следом запрещаем доступ из инета в локальную сеть:

```
$IPT -A FORWARD -i $WAN -o $LAN1 -j REJECT
```

Чтобы наша локальная сеть пользовалась интернетом, включаем nat:

```
$IPT -t nat -A POSTROUTING -o $WAN -s $LAN1_IP_RANGE -j MASQUERADE
```

Чтобы не потерять доступ к серверу, после применения правил, разрешаем подключения по ssh:

```
$IPT -A INPUT -i $WAN -p tcp --dport 22 -j ACCEPT
```

И в конце записываем правила, чтобы они применились после перезагрузки:

```
/sbin/iptables-save  > /etc/sysconfig/iptables
```

Мы составили простейший конфиг, который блокирует все входящие соединения, кроме ssh и разрешает доступ из локальной сети в интернет. Попутно защитились от некоторых сетевых атак.
Сохраняем скрипт, делаем исполняемым и запускаем:

```
# chmod 0740 /etc/iptables.sh
# /etc/iptables.sh
```

Выполним просмотр правил и проверим, все ли правила на месте:

```
# iptables -L -v -n
```

Обращаю ваше внимание - применять правила нужно лишь в том случае, если у вас имеется доступ к консоли сервера. При ошибке в настройках вы можете потерять доступ.

Убедитесь, что в нештатной ситуации вы сможете отключить фаервол и скорректировать настройки.

3. Открытие портов

Теперь немного расширим нашу конфигурацию и откроем в iptables порты для некоторых сервисов. Допустим, у нас работает веб-сервер и необходимо открыть к нему доступ из интернета. Добавляем правила для веб-трафика:

```
$IPT -A INPUT -p tcp -m tcp --dport 80 -j ACCEPT
$IPT -A INPUT -p tcp -m tcp --dport 443 -j ACCEPT
```

Было добавлено разрешение на входящие соединения по 80-му и 443-му портам, которые использует web сервер в своей работе.

Если у вас установлен почтовый сервер, то нужно разрешить на него входящие соединения по всем используемым портам:

```
$IPT -A INPUT -p tcp -m tcp --dport 25 -j ACCEPT
$IPT -A INPUT -p tcp -m tcp --dport 465 -j ACCEPT
$IPT -A INPUT -p tcp -m tcp --dport 110 -j ACCEPT
$IPT -A INPUT -p tcp -m tcp --dport 995 -j ACCEPT
$IPT -A INPUT -p tcp -m tcp --dport 143 -j ACCEPT
$IPT -A INPUT -p tcp -m tcp --dport 993 -j ACCEPT
```

Для корректной работы DNS сервера, нужно открыть UDP порт 53

```
$IPT -A INPUT -i $WAN -p udp --dport 53 -j ACCEPT
```

И так далее. По аналогии можете открыть доступ для всех необходимых сервисов.

4. Проброс (forward) порта

Рассмотрим ситуацию, когда необходимо выполнить проброс портов с внешнего интерфейса на какой-то компьютер в локальной сети. Допустим, вам необходимо получить rdp доступ к компьютеру 10.1.3.50 из интернета. Делаем проброс TCP порта 3389:

```
$IPT -t nat -A PREROUTING -p tcp --dport 3389 -j $WAN} -j DNAT --to 10.1.3.50
```

Если вы не хотите светить снаружи известным портом, то можно сделать перенаправление с нестандартного порта на порт rdp конечного компьютера:

```
$IPT -t nat -A PREROUTING -p tcp --dport 53243 -j ${WAN} -j DNAT --to 10.1.3.50:3389
```

Если вы пробрасываете порт снаружи внутрь локальной сети, то обязательно закомментируйте правило, которое блокирует доступ из внешней сети во внутреннюю. В моем примере это правило:

```
$IPT -A FORWARD -i $WAN -o $LAN1 -j REJECT
```

Либо перед этим правилом создайте разрешающее правило для доступа снаружи к внутреннему сервису, например вот так:

```
$IPT -A FORWARD -i $WAN -d 10.1.3.50 -p tcp -m tcp --dport 3389 -j ACCEPT
```

5. Включение логов

Во время настройки полезно включить логи, чтобы мониторить заблокированные пакеты и выяснять, почему отсутствует доступ к необходимым сервисам, которые мы вроде бы уже открыли. Я отправляю все заблокированные пакеты в отдельные цепочки (block_in, block_out, block_fw), соответствующие направлению трафика и маркирую в логах каждое направление. Так удобнее делать разбор полетов. Добавляем следующие правила в самый конец скрипта, перед сохранением настроек:

```
$IPT -N block_in
$IPT -N block_out
$IPT -N block_fw

$IPT -A INPUT -j block_in
$IPT -A OUTPUT -j block_out
$IPT -A FORWARD -j block_fw

$IPT -A block_in -j LOG --log-level info --log-prefix "--IN--BLOCK"
$IPT -A block_in -j DROP
$IPT -A block_out -j LOG --log-level info --log-prefix "--OUT--BLOCK"
$IPT -A block_out -j DROP
$IPT -A block_fw -j LOG --log-level info --log-prefix "--FW--BLOCK"
$IPT -A block_fw -j DROP
```

Все заблокированные пакеты вы сможете отследить в файле /var/log/messages.
После того, как закончите настройку, закомментируйте эти строки, отключив логирование. Обязательно стоит это сделать, так как логи очень быстро разрастаются. Практического смысла в хранении подобной информации лично я не вижу.
Как отключить iptables
Если вы вдруг решите, что firewall вам больше не нужен, то отключить его можно следующим образом:

```
# systemctl stop iptables.service
```

Эта команда останавливает фаервол. А следующая удаляет из автозагрузки:

```
# systemctl disable iptables.service
```

Отключив сетевой экран, мы разрешили все соединения.
Сделайте скриншоты (фотографии) процесса настройки IPTABLESи вставьте в отчёт.

Задание 2:
Kali включает в себя очень способный OpenVAS, который является бесплатным и с открытым исходным кодом.
Это просто потому, что сканеры уязвимостей часто имеют слабую репутацию, прежде всего потому, что их роль и цель неправильно поняты.
Сканеры Vulnerabilty сканируют уязвимости – но они не являются волшебными машинами эксплойта и должны быть одним из многих источников информации, используемых в оценке безопасности.
Слепой запуск сканера уязвимостей на цель почти наверняка закончится разочарованием и горем, с десятками (или даже сотнями) результатов низкого уровня или неинформативных результатов.

1. Системные Требования

Основная жалоба, которую получают о OpenVAS (или любом другом сканере уязвимостей), можно резюмировать как «она слишком медленная и сбойная и не работает, и это плохо, и очень плохо».
Почти во всех случаях медленность и / или сбои связаны с недостаточными системными ресурсами.
OpenVAS имеет десятки тысяч сигнатур, и если вы не дадите вашей системе достаточного количества ресурсов, особенно оперативной памяти, вы окажетесь в мире страданий.
Для некоторых коммерческих сканеров уязвимостей требуется как минимум 8 ГБ ОЗУ и рекомендуется еще больше.
OpenVAS не требует около такого объема памяти, но чем больше вы можете предоставить ему, тем более плавно система сканирования будет работать.

Для этого урока наша виртуальная машина Kali имеет 3 процессора и 3 ГБ оперативной памяти, что обычно достаточно для сканирования небольшого количества хостов одновременно.

2. Начальная установка OpenVAS в Кали

У OpenVAS много движущихся частей, и настройка вручную может быть проблемой.

К счастью, Kali содержит простую в использовании утилиту под названием «openvas-setup», которая занимается настройкой OpenVAS, загрузкой сигнатур и созданием пароля для пользователя admin.

Эта первоначальная настройка может занять довольно много времени, даже при быстром подключении к Интернету, можно просто сидеть сложа руки.

В конце настройки будет отображаться автоматически созданный пароль для пользователя admin.

Обязательно сохраните этот пароль где-нибудь в безопасности.

```
root@kali:~# openvas-setup
ERROR: Directory for keys (/var/lib/openvas/private/CA) not found!
ERROR: Directory for certificates (/var/lib/openvas/CA) not found!
ERROR: CA key not found in /var/lib/openvas/private/CA/cakey.pem
ERROR: CA certificate not found in /var/lib/openvas/CA/cacert.pem
ERROR: CA certificate failed verification, see /tmp/tmp.7G2IQWtqwj/openvas-manage-certs.log for details. Aborting.ERROR: Your OpenVAS certificate infrastructure did NOT pass validation.
See messages above for details.
Generated private key in /tmp/tmp.PerU5lG2tl/cakey.pem.
Generated self signed certificate in /tmp/tmp.PerU5lG2tl/cacert.pem.
...
/usr/sbin/openvasmd
User created with password 'xxxxxxx-xxxx-xxxx-xxxx-xxxxxxxxxx'.
```

3. Работа с ошибками установки

Иногда скрипт «openvas-setup» будет отображать ошибки в конце загрузки NVT, аналогичные приведенным ниже.

```
(openvassd:2272): lib kb_redis-CRITICAL **: get_redis_ctx: redis connection error: No such file or directory
(openvassd:2272): lib kb_redis-CRITICAL **: redis_new: cannot access redis at '/var/run/redis/redis.sock'
(openvassd:2272): lib kb_redis-CRITICAL **: get_redis_ctx: redis connection error: No such file or directory
openvassd: no process found
```

Если вам посчастливилось столкнуться с этой проблемой, вы можете запустить «openvas-check-setup», чтобы узнать, какой компонент вызывает проблемы.

В этом конкретном случае мы получаем следующее из скрипта:

```
...
ERROR: The number of NVTs in the OpenVAS Manager database is too low.
FIX: Make sure OpenVAS Scanner is running with an up-to-date NVT collection and run 'openvasmd --rebuild'.
...
```

Скрипт «openvas-check-setup» обнаруживает проблему и даже предоставляет команду для запуска (надеюсь) решения этой проблемы.

После восстановления коллекции NVT рекомендуется пройти все проверки.

```
root@kali:~# openvasmd --rebuild
root@kali:~# openvas-check-setup
openvas-check-setup 2.3.7
Test completeness and readiness of OpenVAS-9
```

```
...
It seems like your OpenVAS-9 installation is OK.
...
```

4. Управление пользователями OpenVAS

Если вам нужно (или хотите) создать дополнительных пользователей OpenVAS, запустите ‘openvasmd’ с параметром -create-user, который добавит нового пользователя и отобразит случайно сгенерированный пароль.

```
root@kali:~# openvasmd --create-user=dookie
User created with password 'yyyyyyyy-yyyy-yyyy-yyyy-yyyyyyyyyy'.
root@kali:~# openvasmd --get-users
admin
dookie
```

К счастью, изменение паролей пользователей OpenVAS легко осуществляется с помощью опции «openvasmd» и «new-password».

```
root@kali:~# openvasmd --user=dookie --new-password=s3cr3t
root@kali:~# openvasmd --user=admin --new-password=sup3rs3cr3t
```

5. Запуск и остановка OpenVAS

Сетевые службы по умолчанию отключены в Kali Linux, поэтому, если вы не настроили OpenVAS для запуска при загрузке, вы можете запустить необходимые службы, запустив «openvas-start».

```
root@kali:~# openvas-start
Starting OpenVas ServicesПосле того, как у вас есть список хостов, вы можете импортировать их в разделе «Цели» в меню «Конфигурация».
```

Когда службы завершают инициализацию, вы должны найти TCP-порты 9390 и 9392, которые прослушивают ваш loopback-интерфейс.

```
root@kali:~# ss -ant
State Recv-Q Send-Q Local Address:Port Peer Address:Port
LISTEN 0 128 127.0.0.1:9390 *:*
LISTEN 0 128 127.0.0.1:9392 *:*
```

Из-за нагрузки на системные ресурсы вы, вероятно, захотите остановить OpenVAS, когда вы закончите использовать его, особенно если вы не используете специальную систему для сканирования уязвимостей.

OpenVAS можно остановить, запустив «openvas-stop».

```
root@kali:~# openvas-stop
Stopping OpenVas Services
```

6. Использование Greenbone Security Assistant

Greenbone Security Assistant – это веб-интерфейс OpenVAS, доступный на вашем локальном компьютере (после запуска OpenVAS) на **https://localhost: 9392.**

После принятия самозаверенного сертификата вам будет представлена страница входа в систему и после аутентификации вы увидите основную панель.

Рис. 340

7. Настройка учетных данных

Сканеры уязвимостей обеспечивают наиболее полные результаты, когда вы можете предоставить механизму сканирования учетные данные для использования на сканируемых системах.

OpenVAS будет использовать эти учетные данные для входа в сканированную систему и выполнения подробного перечисления установленного программного обеспечения, патчей и т. д.

Вы можете добавить учетные данные через запись «Credentials» в меню «Configuration».

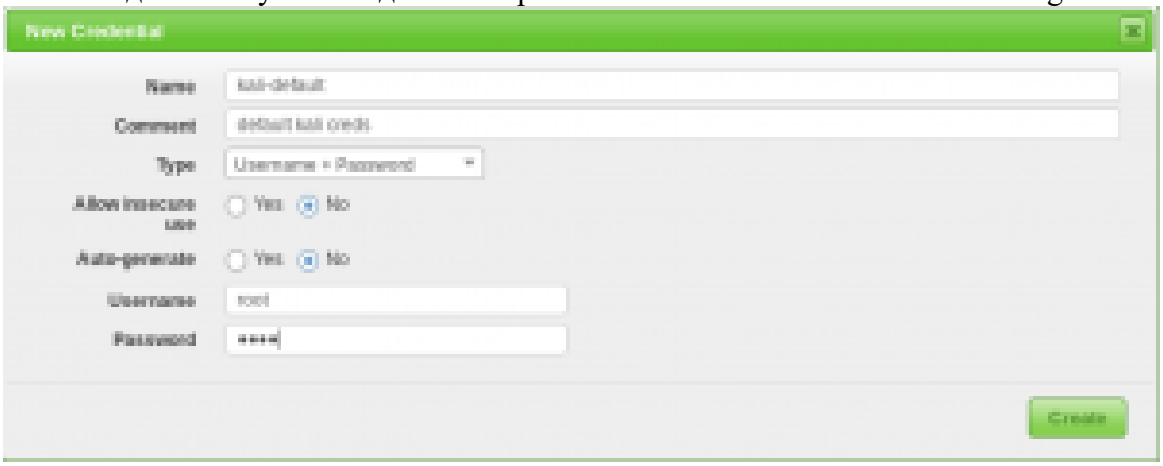


Рис. 341

8. Конфигурация цели

OpenVAS, как и большинство сканеров уязвимостей, может сканировать удаленные системы, но это сканер уязвимостей, а не сканер портов.

Вместо того, чтобы полагаться на сканер уязвимостей для идентификации хостов, вы значительно упростите свою жизнь с помощью специализированного сетевого сканера, такого как Nmap или Masscan, и импортируйте список целей в OpenVAS.

```
root@kali:~# nmap -sn -oA nmap-subnet-86 192.168.86.0/24
root@kali:~# grep Up nmap-subnet-86.gnmap | cut -d " " -f 2 > live-hosts.txt
```

После того, как у вас есть список хостов, вы можете импортировать их в разделе «target» в меню «Configuration».

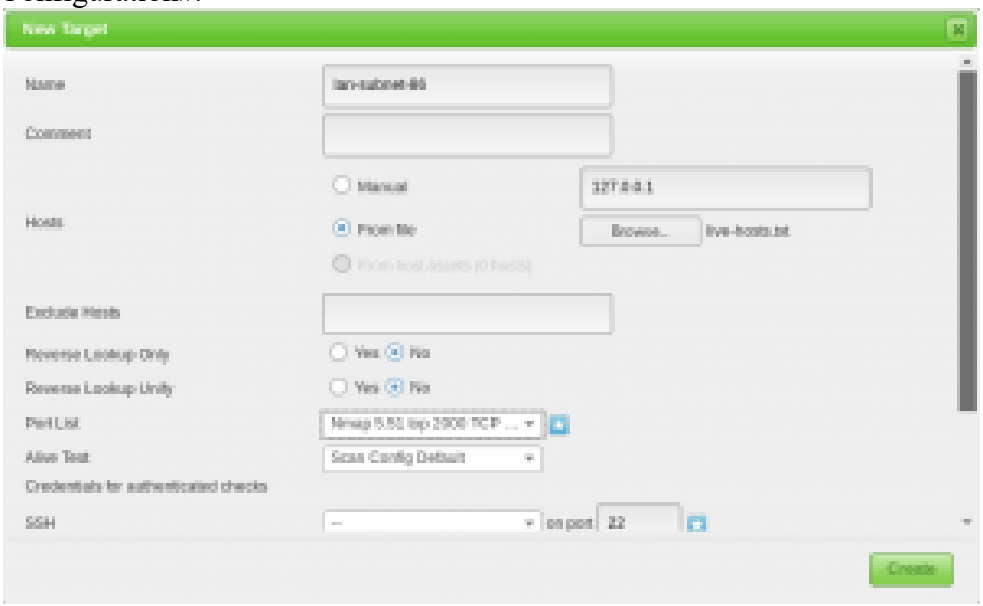


Рис. 342

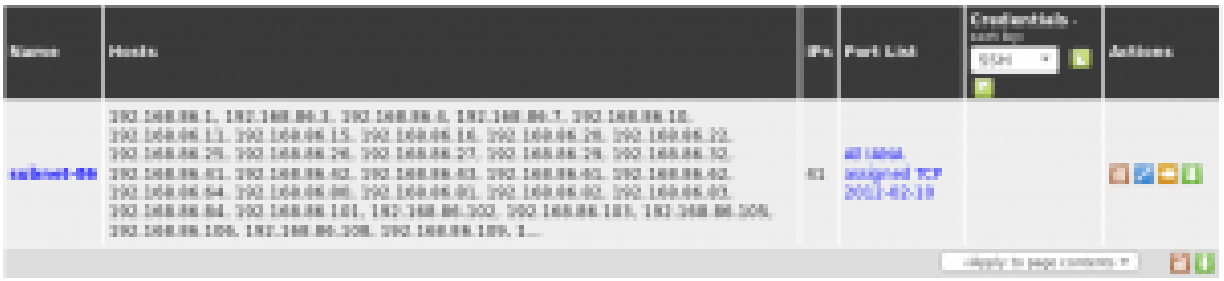


Рис. 343

9. Конфигурация сканирования

Перед запуском сканирования уязвимостей вы должны точно настроить Scan Config/,
Это можно сделать в разделе “Scan Configs” в меню “Config”.

Вы можете клонировать любую конфигурацию сканирования по умолчанию и редактировать ее параметры, отключая любые службы или проверки, которые вам не нужны.

Если вы используете Nmap для проведения предварительного анализа ваших целевых объектов, вы можете сэкономить время сканирования уязвимости.

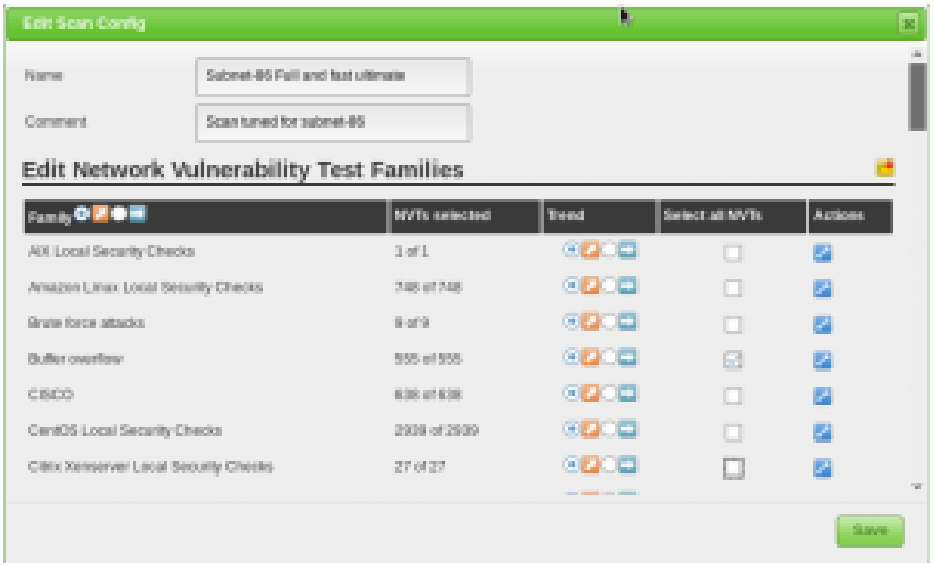


Рис. 344

10. Конфигурация задачи

Ваши учетные данные, цели и конфигурации сканирования настроены таким образом, что теперь вы готовы собрать все вместе и запустить сканирование уязвимостей.

В OpenVAS сканирование уязвимостей проводится как «tasks».

Когда вы настраиваете новую задачу, вы можете дополнительно оптимизировать сканирование путем увеличения или уменьшения одновременных действий, которые происходят.

С нашей системой с 2 ГБ оперативной памяти мы скорректировали наши настройки задач, как показано ниже.

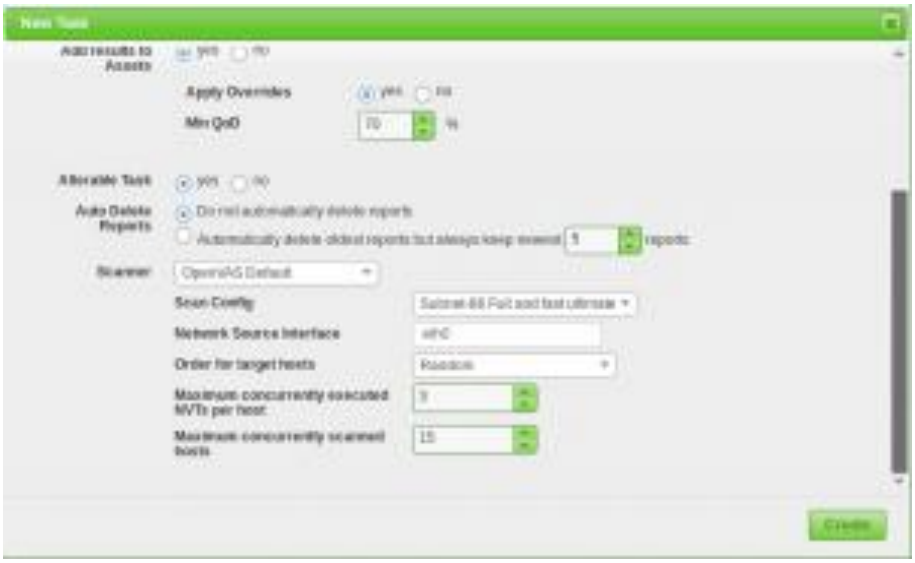


Рис. 345

Благодаря нашим более точно настроенным параметрам сканирования и целевому выбору результаты нашего сканирования намного полезнее.

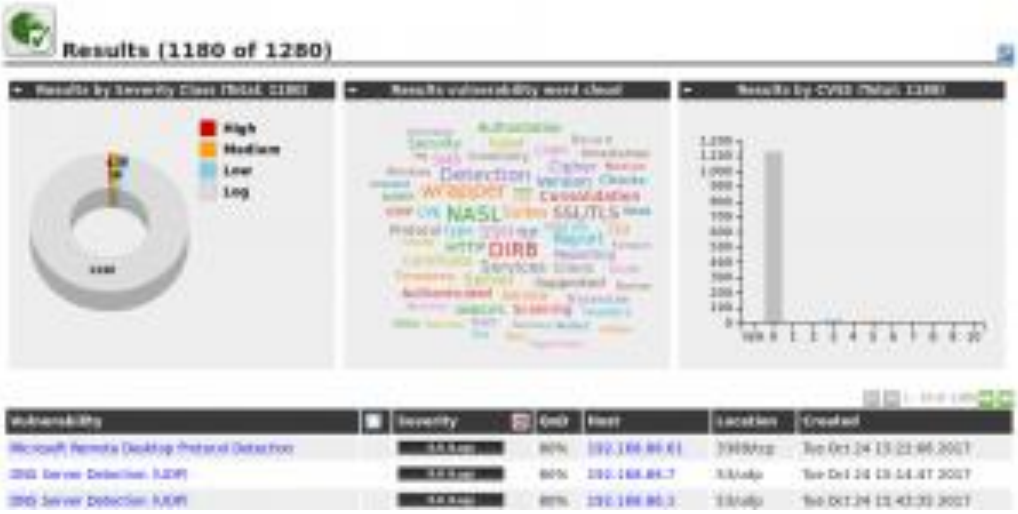


Рис. 346

11. Автоматизация OpenVAS

Одной из менее известных функций OpenVAS является интерфейс командной строки, с которым вы взаимодействуете с помощью команды «omp».

Его использование не совсем интуитивно, но мы не единственные поклонники OpenVAS, и мы столкнулись с несколькими базовыми скриптами, которые вы можете использовать и расширить сканирование для автоматизации OpenVAS.

Первый – openvas-automate.sh от mgeeky, полуинтерактивный скрипт Bash, который предлагает вам тип сканирования и заботится обо всем остальном.

Конфигурации сканирования жестко закодированы в сценарии, поэтому, если вы хотите использовать свои настроенные конфиги, их можно добавить в разделе «targets».

```
root@kali:~# apt -y install pcregrep
root@kali:~# ./openvas-automate.sh 192.168.86.61:: OpenVAS automation script.
mgeeky, 0.1[>] Please select scan type:
1. Discovery
2. Full and fast
3. Full and fast ultimate
4. Full and very deep
5. Full and very deep ultimate
6. Host Discovery
7. System Discovery
9. Exit

------------------------------
Please select an option: 5

[+] Tasked: 'Full and very deep ultimate' scan against '192.168.86.61'
[>] Reusing target...
[+] Target's id: 6ccbb036-4afa-46d8-b0c0-acbd262532e5
[>] Creating a task...
[+] Task created successfully, id: '8e77181c-07ac-4d2c-ad30-9ae7a281d0f8'
[>] Starting the task...
[+] Task started. Report id: 6bf0ec08-9c60-4eb5-a0ad-33577a646c9b
[.] Awaiting for it to finish. This will take a long while...
8e77181c-07ac-4d2c-ad30-9ae7a281d0f8 Running 1% 192.168.86.61
```

Мы также наткнулись на сообщение в блоге по code16, которое представляет и объясняет их скрипт Python для взаимодействия с OpenVAS.

Подобно скрипту Bash выше, вам нужно будет внести некоторые изменения в скрипт, если вы хотите настроить тип сканирования.

```
root@kali:~# ./code16.py 192.168.86.27
-------------------------------------------------------------------------------
code16
-------------------------------------------------------------------------------
small wrapper for OpenVAS 6[+] Found target ID: 19f3bf20-441c-49b9-823d-11ef3b3d18c2
[+] Preparing options for the scan...
[+] Task ID = 28c527f8-b01c-4217-b878-0b536c6e6416
[+] Running scan for 192.168.86.27
[+] Scan started... To get current status, see below:zZzzZzzZzzZzzZzzZzzZzzZzzZzzZzzZzzZzzZzzZzzZzzZzzZzzZzzZzzZzzZzzZzzZzzZzzZzz
...
zZzzZzzZzzZzzZzzZzzZzzZzzZzzZzzZzzZzzZzzZzzZzzZzzZzzZzzZzzZzzZzzZzzZzzZzzZzzZzzZzzZzzZzz
[+] Scan looks to be done. Good.
[+] Target scanned. Finished taskID : 28c527f8-b01c-4217-b878-0b536c6e6416
[+] Cool! We can generate some reports now ... :)
[+] Looking for report ID...
[+] Found report ID : 5ddcb4ed-4f96-4cee-b7f3-b7dad6e16cc6
[+] For taskID : 28c527f8-b01c-4217-b878-0b536c6e6416
[+] Preparing report in PDF for 192.168.86.27
[+] Report should be done in : Report_for_192.168.86.27.pdf
[+] Thanks. Cheers!
```