

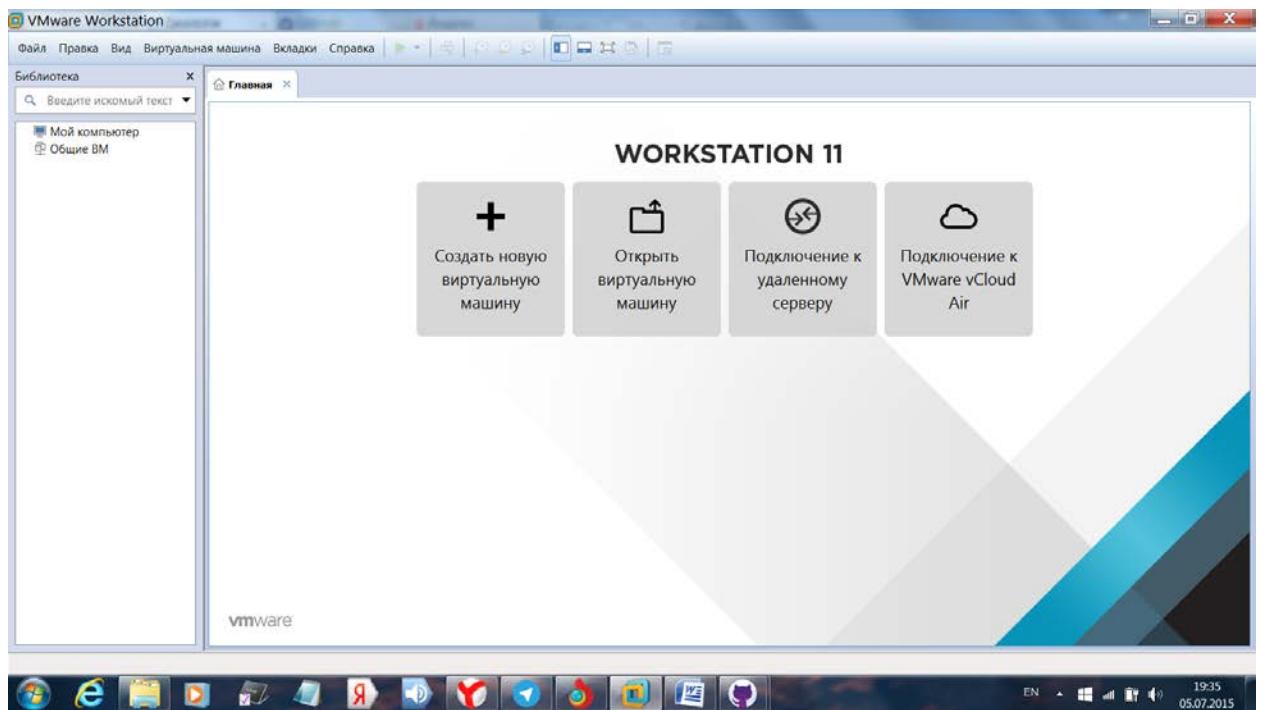
Практика

Задание 1

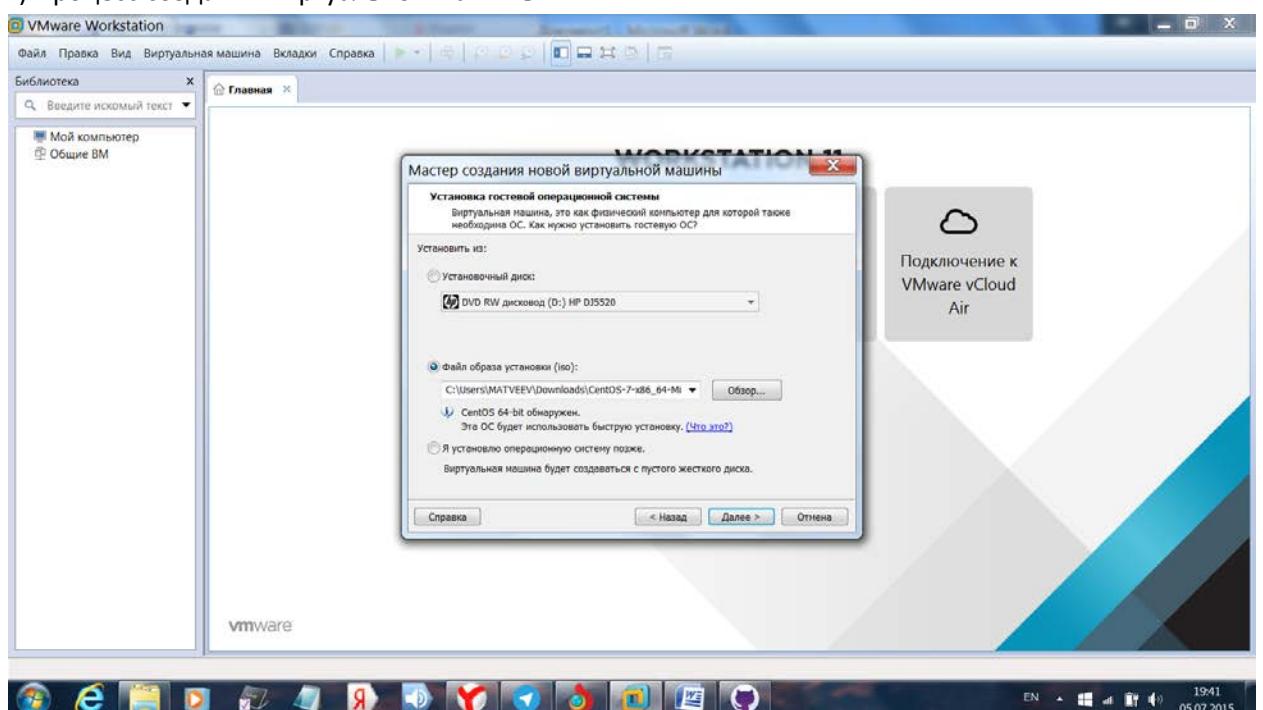
Матвеев М.И.

VMware Workstation создает полностью изолированные безопасные виртуальные машины. Уровень виртуализации VMware сопоставляет ресурсы физического оборудования с ресурсами виртуальной машины. Таким образом, каждая виртуальная машина получает собственные ЦП, память, диски и устройства ввода-вывода и является полным эквивалентом стандартного компьютера x86. CentOS является дистрибутивом GNU/Linux, основанном на свободных исходных текстах коммерческого дистрибутива Red Hat Enterprise Linux компании Red Hat.

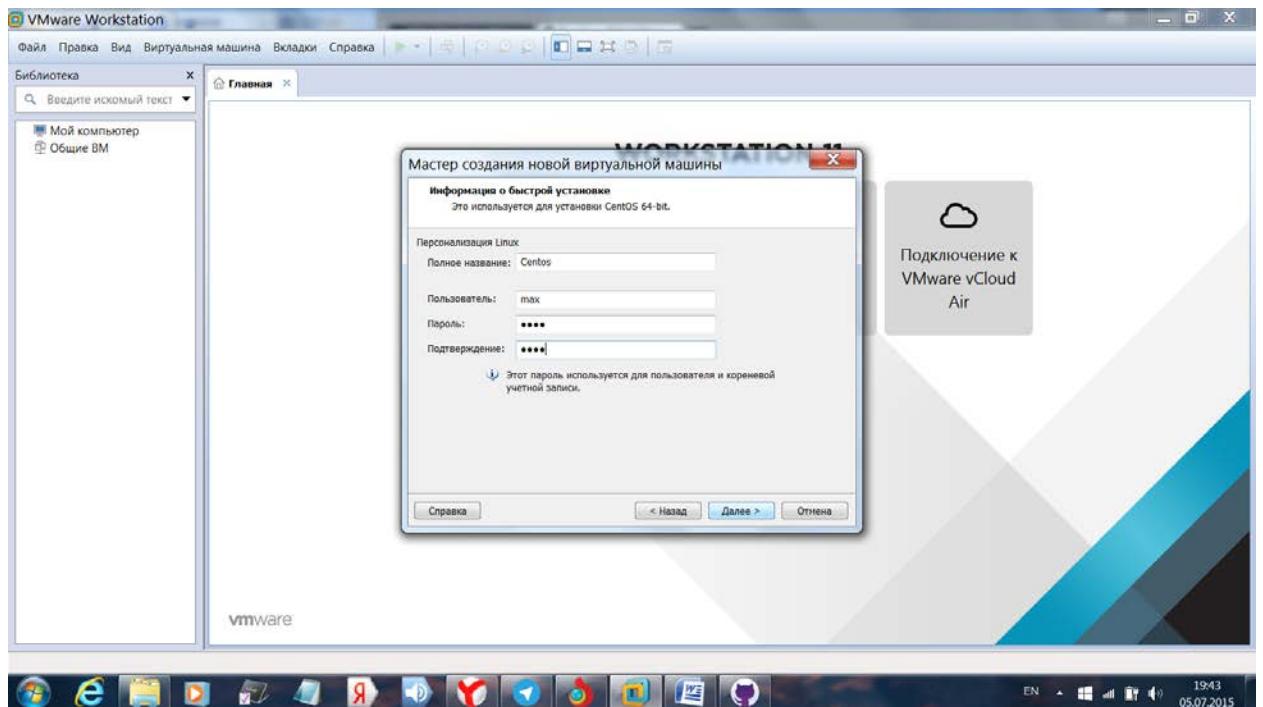
1) Установка VMware Workstation



2) Процесс создания виртуальной машины



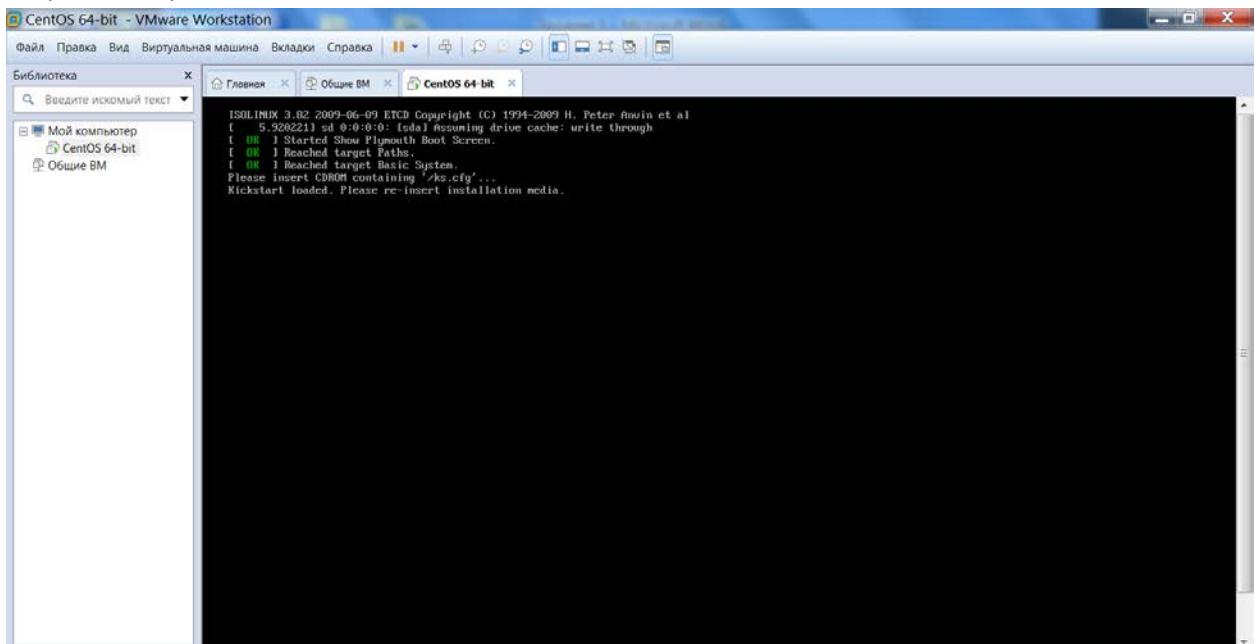
Выбор пароля и логина:



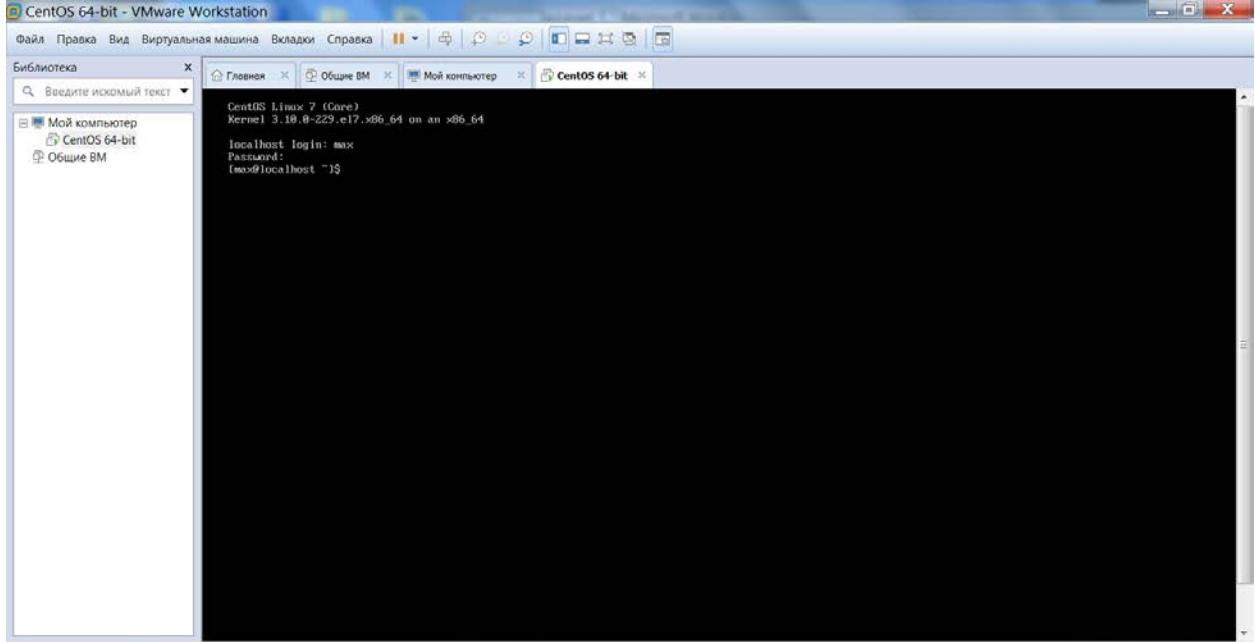
Настройка дополнительных параметров:



Запуск и авторизация



Для входа в эту виртуальную машину, щелкните внутри окна или нажмите Ctrl+G.



Для входа в эту виртуальную машину, щелкните внутри окна или нажмите Ctrl+G.



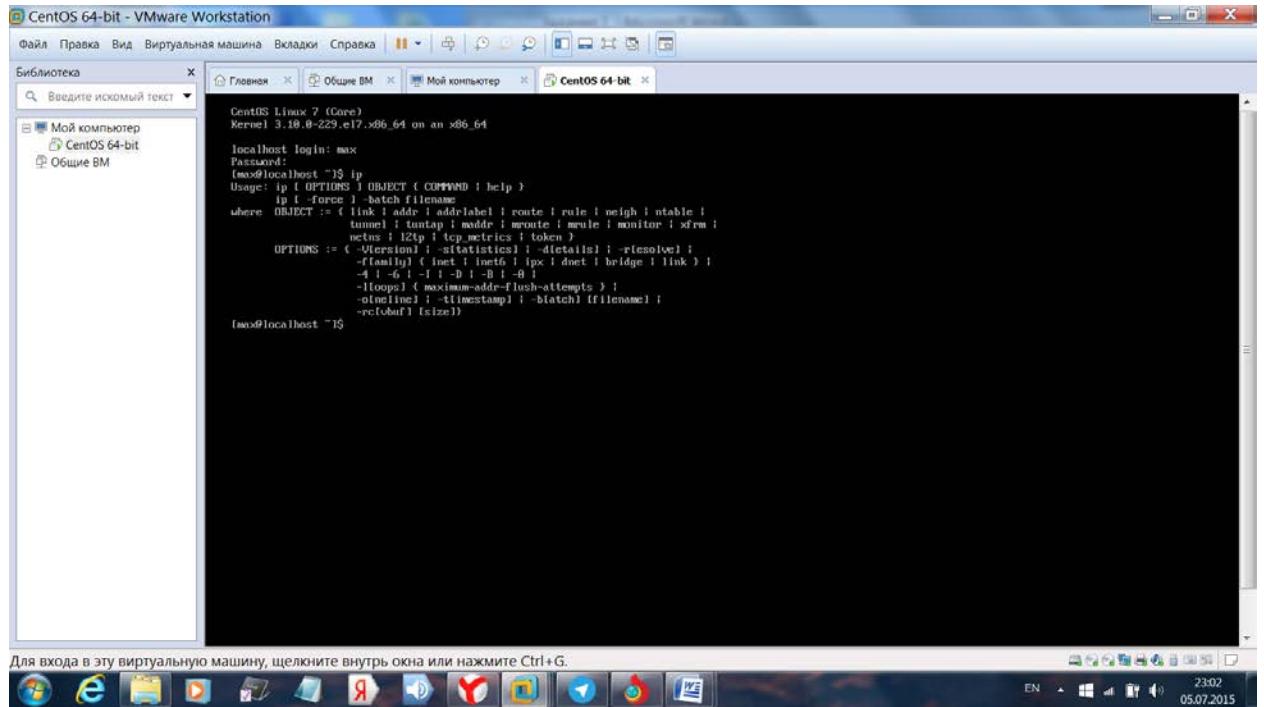
Задание 2

1)Основные команды.

Для взаимодействия с системой, необходимо знать несколько основных команд. Одной из основных команд является команда **su**, которая делает вас пользователем “root”, у которого по умолчанию есть доступ ко всем командам. Ставь пользователем “root”, мы получаем доступ к таким командам как **poweroff**(выключение компьютера), **reboot**(перезагрузка системы),

dir(показывает список файлов в директории, отсортированных вертикально по столбцам), **mkdir**(создать директорию), **rmdir**(удалить директорию).

Команда **ip** помогает узнать информацию о пользователе:



CentOS 64-bit - VMware Workstation

Файл Правка Вид Виртуальная машина Вкладки Справка | Библиотека | Введите искомый текст | Главная | Общие BM | Мой компьютер | CentOS 64-bit |

```
CentOS Linux ? (Core)
Kernel 3.10.0-223.el7.x86_64 on an x86_64
localhost login: max
Password:
(max@localhost ~) $ ip
Usage: ip { OPTIONS } { OBJECT { COMMAND | help } }
      ip { -force } { batch filename }

where OBJECT := { link | addr | addresslabel | route | rule | neigh | ntable |
                 tunnel | tuntap | nudle | mroute | neule | monitor | xfrm |
                 netns | l2tp | tcp_metrics | token }

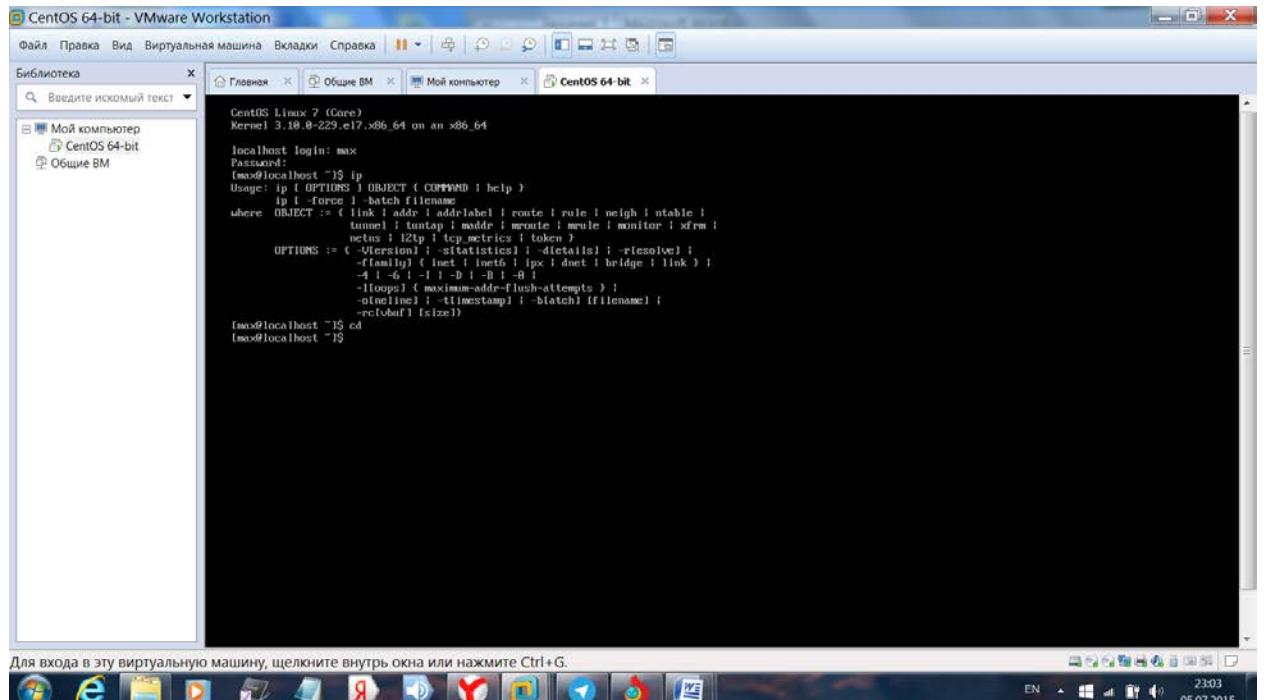
OPTIONS := { -Version| | -sitatistics| | -details| | -resolve| |
            -familij | { inet | inet6 | ipx | dnet | bridge | link } |
            -4 | -6 | -I | -D | -B | -B |
            -lloops | { maximum-addr-flush-attempts } |
            -baseline| | -timestamp| | -batch| {filename} |
            -octobuf| {size} }

(max@localhost ~) $
```

Для входа в эту виртуальную машину, щелкните внутри окна или нажмите Ctrl+G.

EN 23:02 05.07.2015

команда cd: если вы введете эту команду без аргументов, вы попадете в свой домашний каталог; чтобы попасть в любой другой каталог, необходимо указать путь к нему.



CentOS 64-bit - VMware Workstation

Файл Правка Вид Виртуальная машина Вкладки Справка | Библиотека | Введите искомый текст | Главная | Общие BM | Мой компьютер | CentOS 64-bit |

```
CentOS Linux 7 (Core)
Kernel 3.10.0-229.el7.x86_64 on an x86_64
localhost login: max
Password:
(max@localhost ~) $ ip
Usage: ip { OPTIONS } { OBJECT { COMMAND | help } }
      ip { -force } { batch filename }

where OBJECT := { link | addr | addresslabel | route | rule | neigh | ntable |
                 tunnel | tuntap | nudle | mroute | neule | monitor | xfrm |
                 netns | l2tp | tcp_metrics | token }

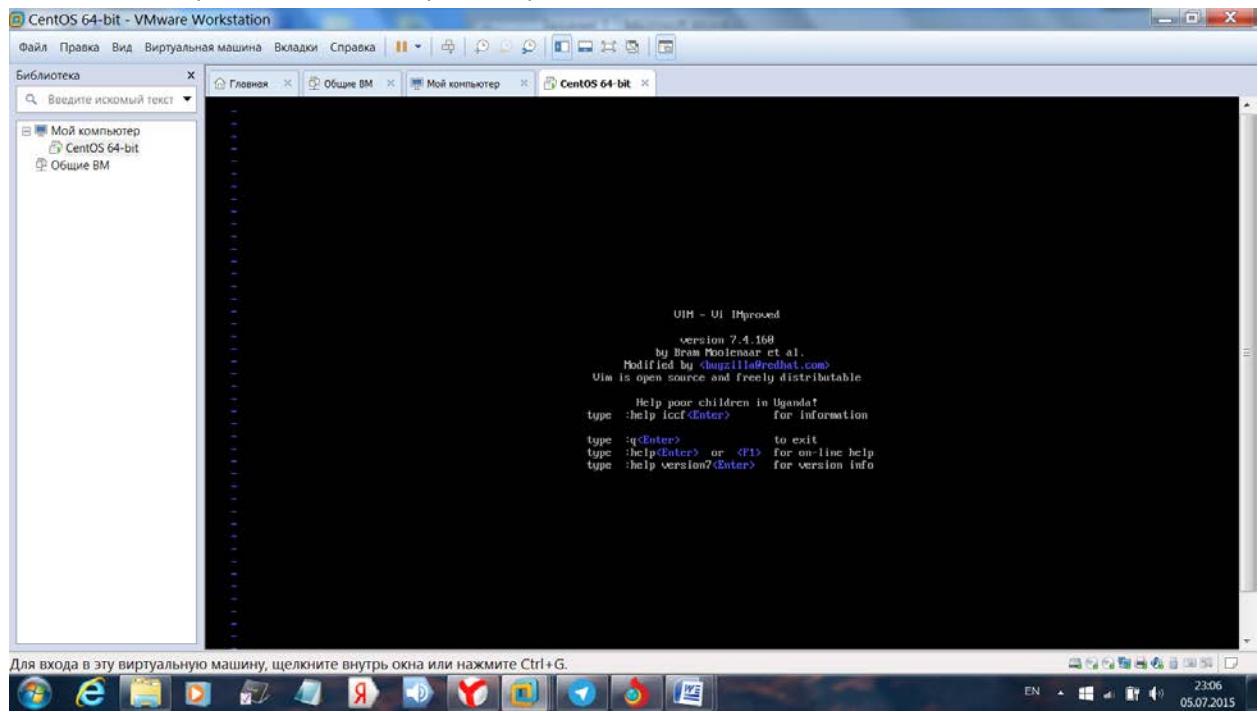
OPTIONS := { -Version| | -sitatistics| | -details| | -resolve| |
            -familij | { inet | inet6 | ipx | dnet | bridge | link } |
            -4 | -6 | -I | -D | -B | -B |
            -lloops | { maximum-addr-flush-attempts } |
            -baseline| | -timestamp| | -batch| {filename} |
            -octobuf| {size} }

(max@localhost ~) $ cd
(max@localhost ~) $
```

Для входа в эту виртуальную машину, щелкните внутри окна или нажмите Ctrl+G.

EN 23:03 05.07.2015

Команда vi открывает текстовый редактор



Команда cat показывает содержимое текстовых файлов, а также "cat" можно использовать для реализации следующих инструкций:

cat /proc/cpuinfo - отобразить информацию о процессоре

cat /proc/interrupts - показать прерывания

cat /proc/meminfo - проверить использование памяти

cat /proc/swaps - показать файл(ы) подкачки

cat /proc/version - вывести версию ядра

cat /proc/net/dev - показать сетевые интерфейсы и статистику по ним

cat /proc/mounts - отобразить смонтированные файловые системы

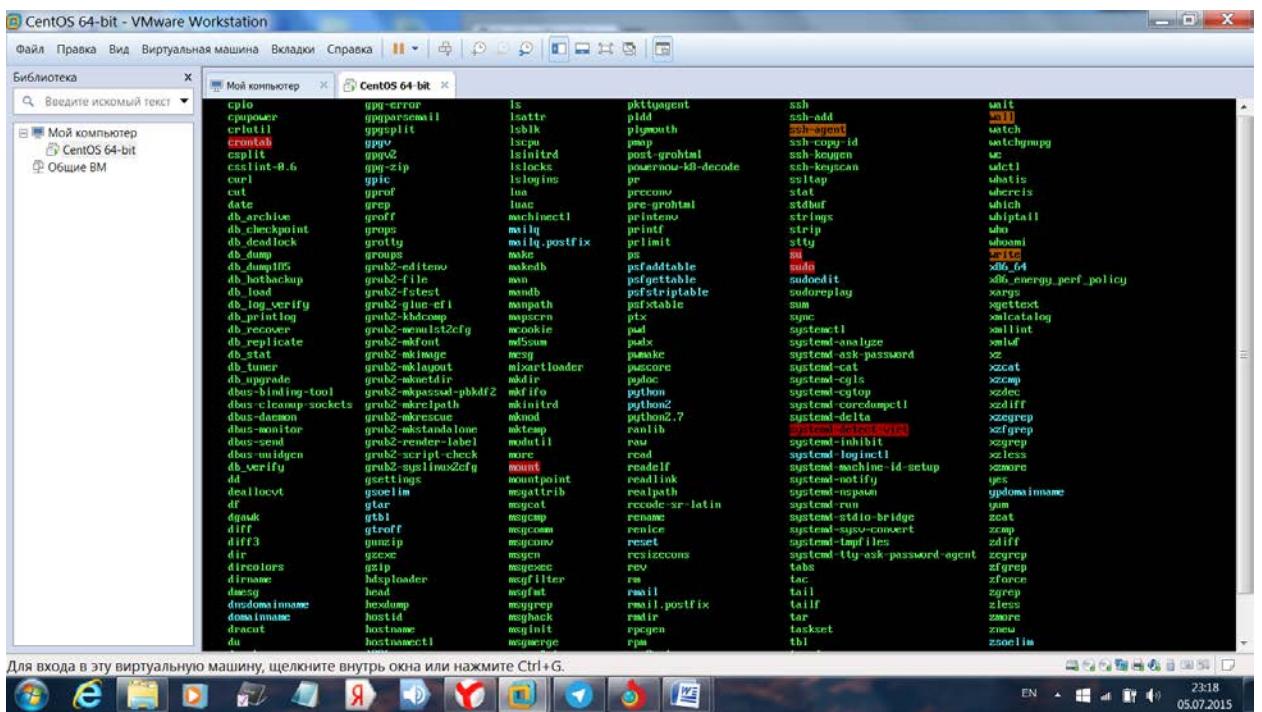
Команда cp позволяет копировать файл

-cp file1 file2 копировать файл file1 в файл file2

-cp dir/* . копировать все файлы директории dir в текущую директорию

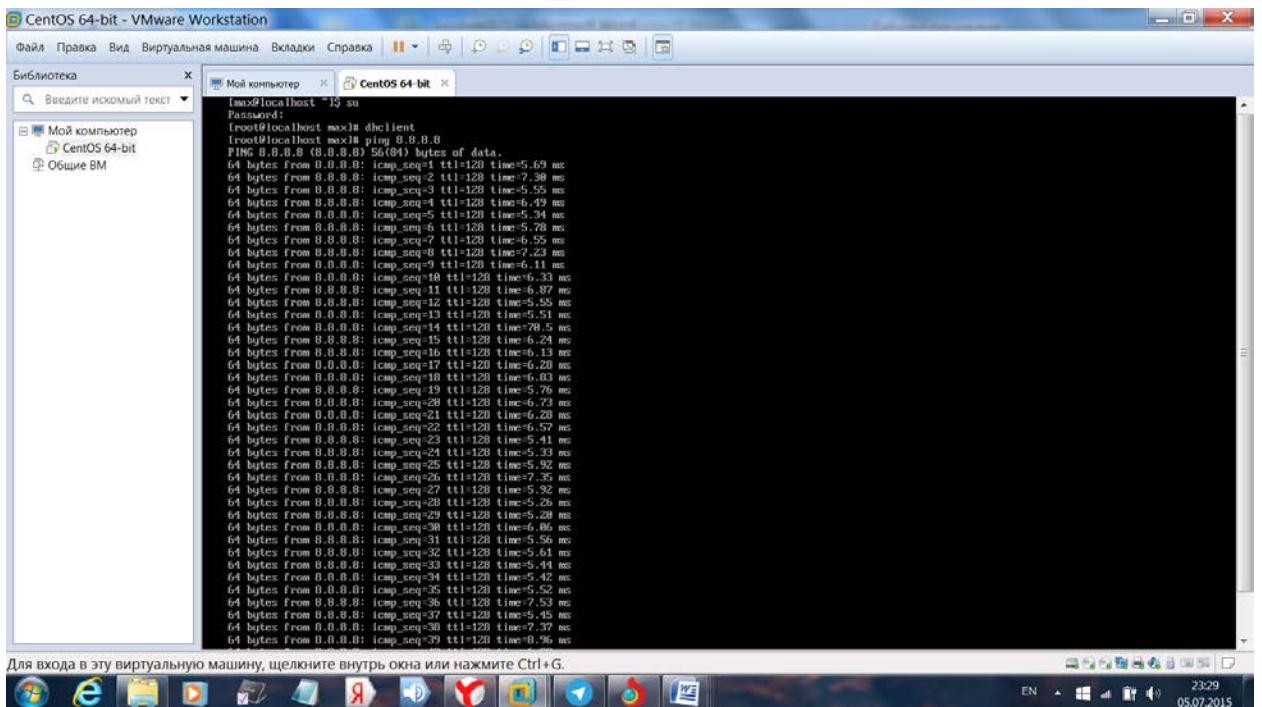
-cp -a /tmp/dir1 . копировать директорию dir1 со всем содержимым в текущую директорию

Команда ls показывает список файлов в текущей директории



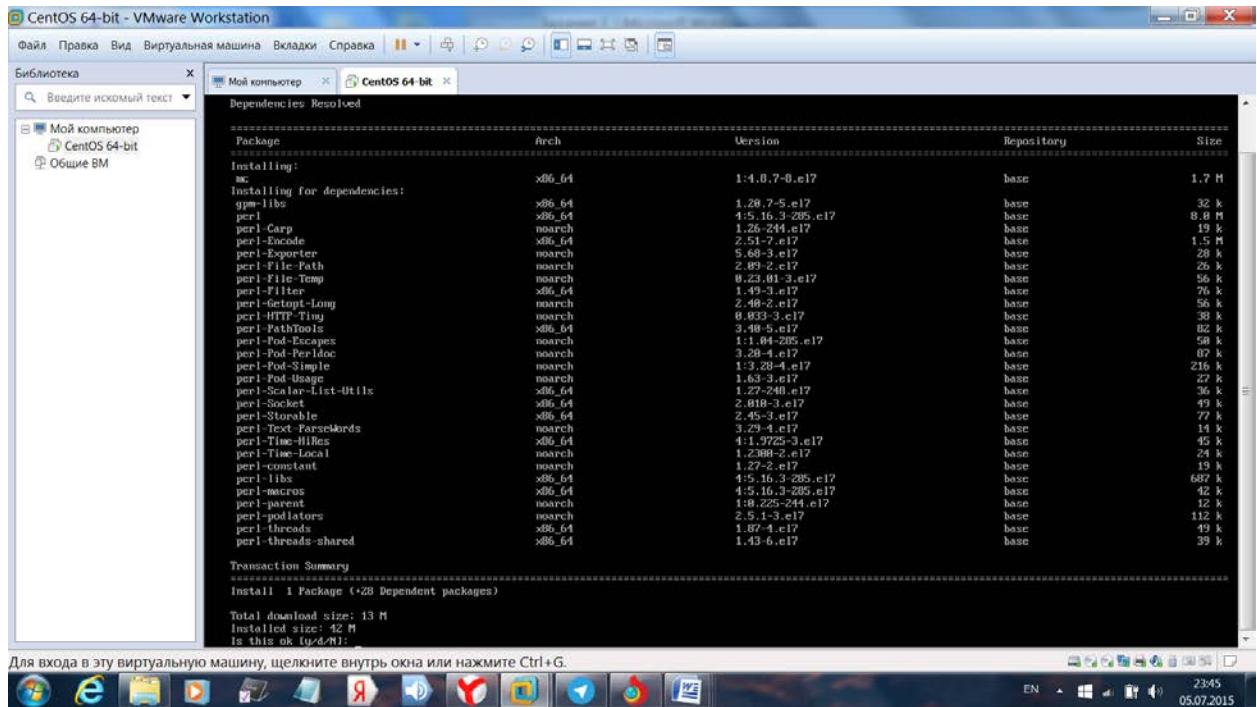
2)Подключение к сети

Чтобы настроить сеть, необходимо получить права рута, то есть вводим команду "su", для подключения к сети вводим "dhclient", а дальше проверяем подключение к интернету, введя "ping 8.8.8.8"



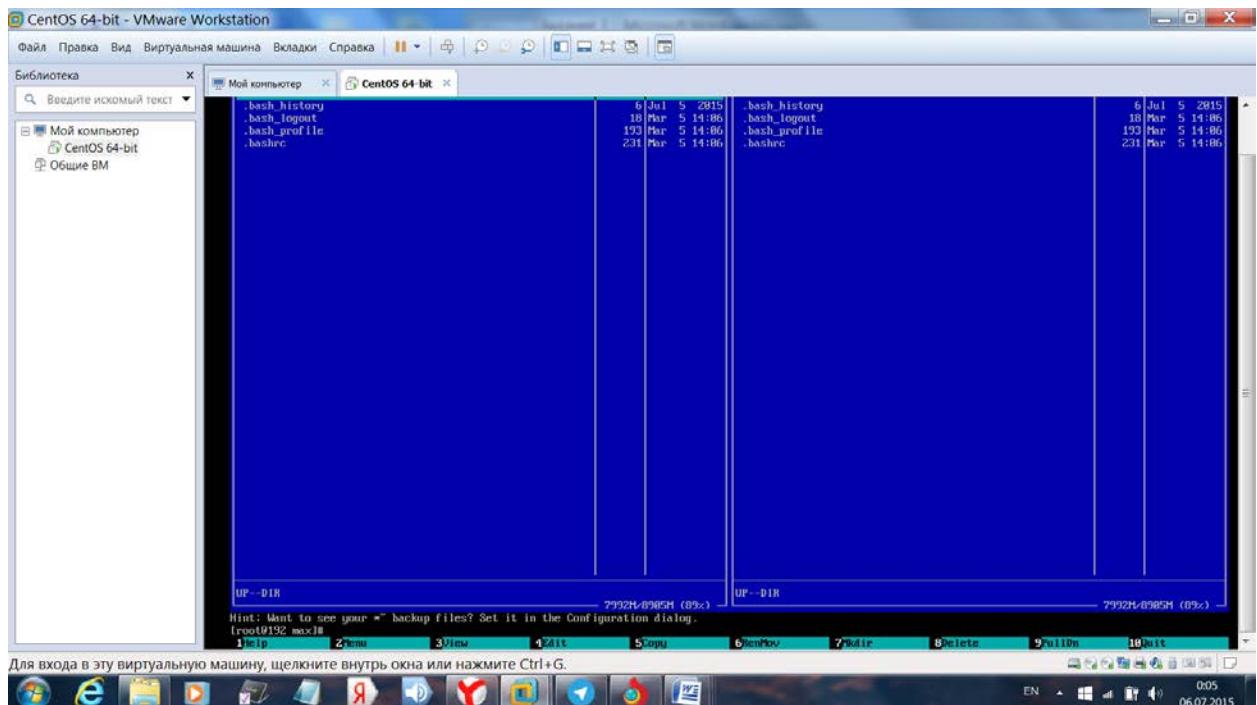
3) Установка дополнительных программ с помощью yum.

-установка mc при помощи команды “yum install mc”:



Для входа в эту виртуальную машину, щелкните внутри окна или нажмите Ctrl+G.

EN 23:45 05.07.2015



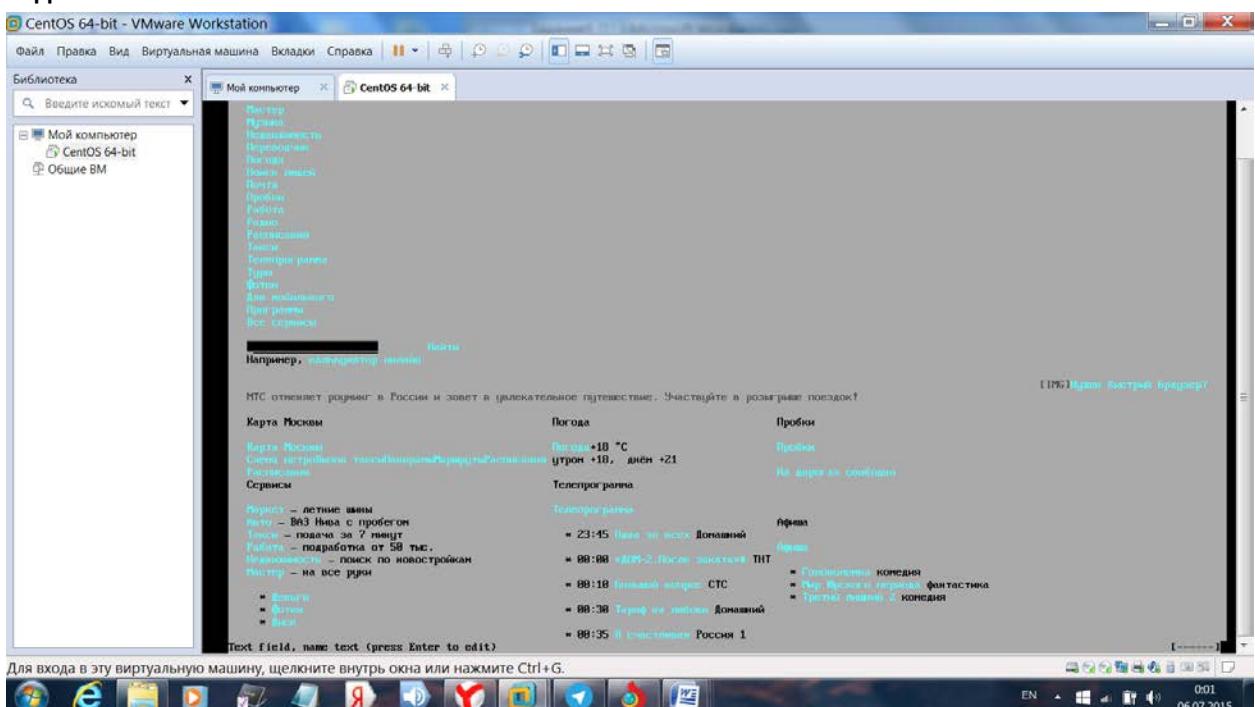
Для входа в эту виртуальную машину, щелкните внутри окна или нажмите Ctrl+G.

EN 00:5 06.07.2015

-Установка elinks с помощью команды “yum install elinks”:

The screenshot shows a VMware Workstation interface with a CentOS 64-bit virtual machine running. The terminal window displays a root shell session where the user is installing the elinks web browser. The process involves checking dependencies, downloading files from mirror.jhhost.ru, and installing packages like libressl-compat-openssl10, elinks, gpm, js, and nss-compat-openssl10. The transaction summary shows 1 package installed and 3 dependent packages updated. The desktop environment includes a file manager showing a folder named 'CentOS 64-bit' and a taskbar with various application icons.

Яндекс:



-Установка nano с помощью команды “**yum install nano**”:

```
CentOS Linux 7 (Core)
Kernel 3.10.0-229.1.2.1.el7.x86_64 on an x86_64
localhost login: max
Password:
Last login: Sun Jul  5 16:47:36 on ttys0
max@localhost ~]$ su
Password:
(max@localhost max)$ dnf client
(max@localhost max)$ yum install nano
Loaded plugins: fastestmirror
Loading mirror speeds from cached hostfile
 * extras: mirror.hghost.ru
 * updates: mirror.hghost.ru
Resolving Dependencies
--> Running transaction check
--> Package nano.x86_64 0:2.3.1-10.el7 will be installed
--> Finished Dependency Resolution

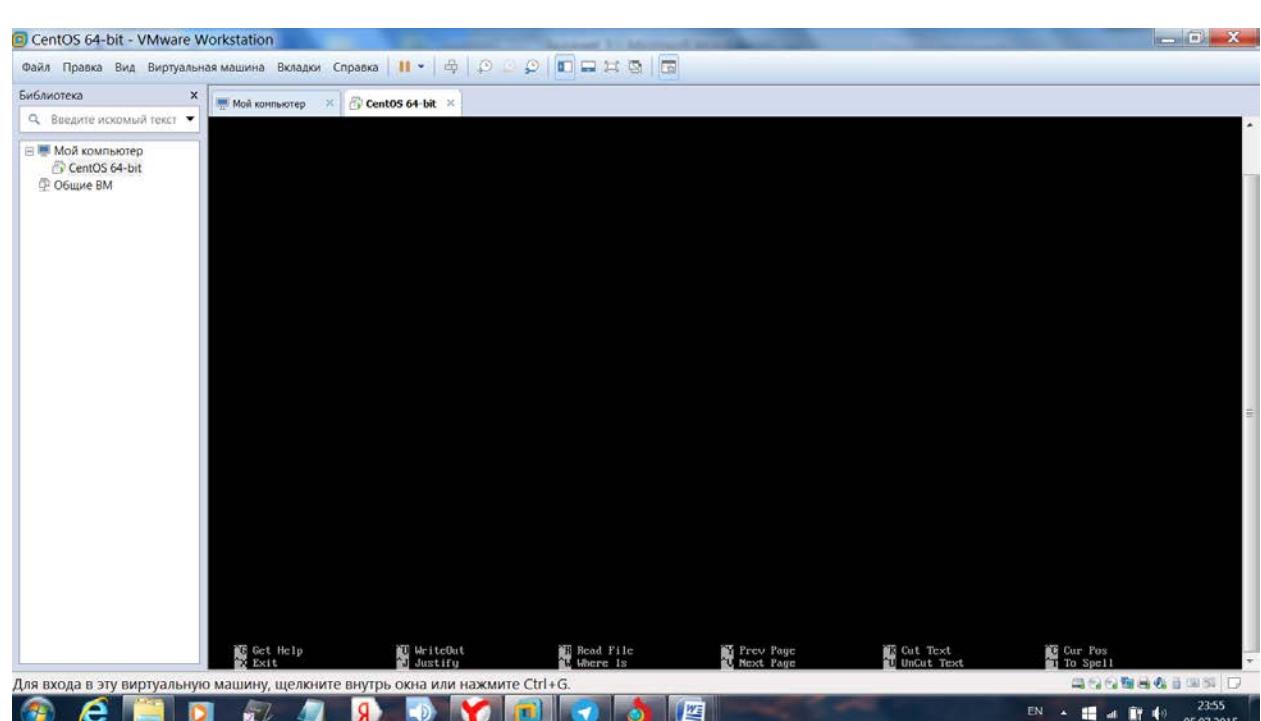
Dependencies Resolved

=====
Package          Arch      Version           Repository  Size
=====
Installing:
nano            x86_64    2.3.1-10.el7      base       440 k

Transaction Summary
=====
Install 1 Package

Total download size: 440 k
Installed size: 1.6 M
Is this ok [y/d/N]: _
```

Для входа в эту виртуальную машину, щелкните внутри окна или нажмите Ctrl+G.

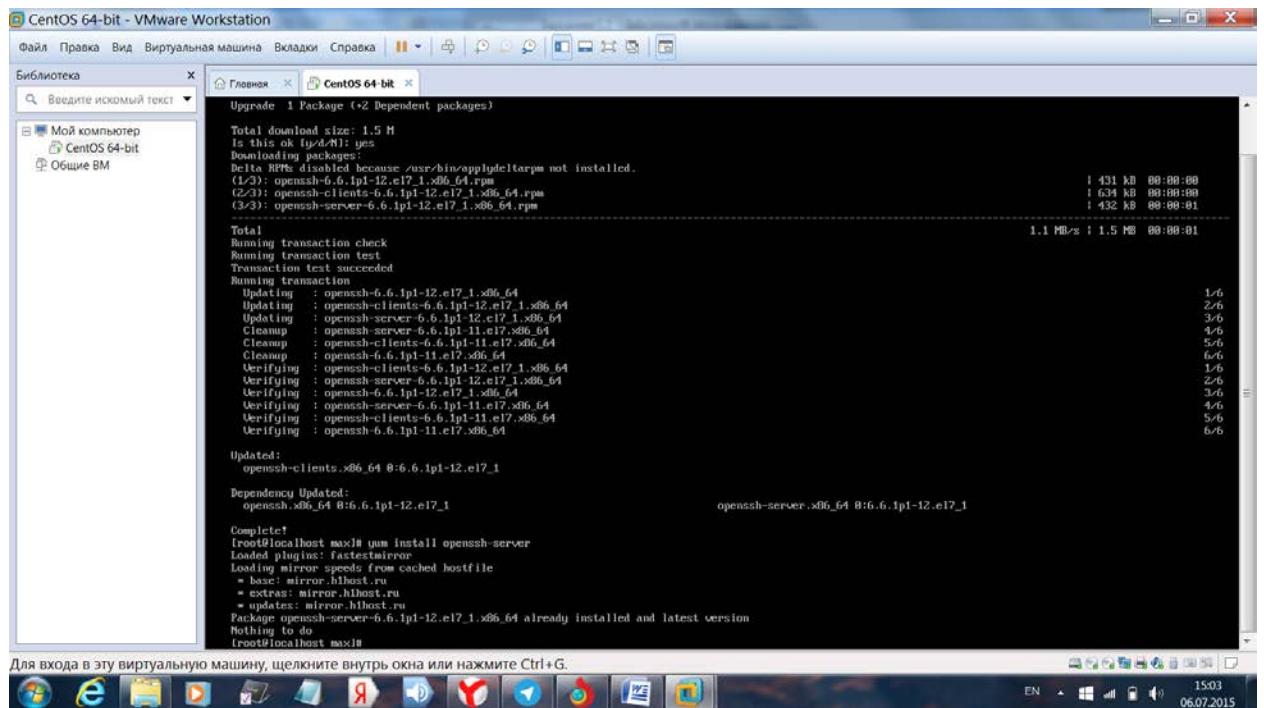


Задание 3

1)SSH это набор программ, которые позволяют регистрироваться на компьютере по сети, удаленно выполнять на нем команды, а также копировать и перемещать файлы между компьютерами. SSH организует защищенное безопасное соединение поверх небезопасных каналов связи.

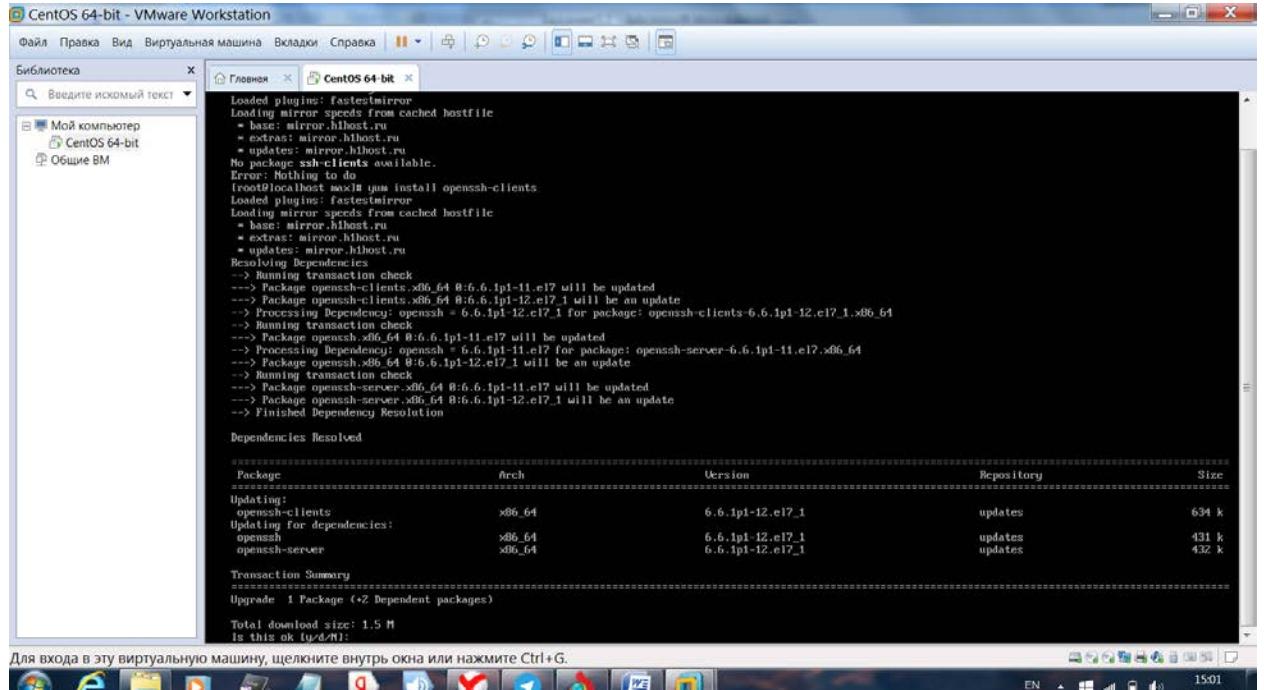
SSH предоставляет замены традиционным г-командам удаленного доступа с тем отличием, что они обладают повышенной безопасностью. Они выполняются поверх защищенных зашифрованных соединений, которые не позволяет прослушивать или подменять трафик. Кроме того, SSH может обеспечивать безопасное соединение для передачи любого другого трафика: например, почтовых сообщений или файлов.

2) Установка клиента и сервера с помощью команд “yum install openssh-clients” и “yum install openssh-server”



```
CentOS 64-bit - VMware Workstation
Файл Правка Вид Виртуальная машина Вкладки Справка ||| □ ○ X
Библиотека x Главное x CentOS 64-bit x
Установка 1 Package (>2 Dependent packages)
Ведите искомый текст
Мой компьютер CentOS 64-bit Общие ВМ
Total download size: 1.5 M
Is this ok [y/d/N]: yes
Downloading packages:
Delta RPMs disabled because /usr/bin/applydeltarps not installed.
(1/3): openssh-6.6.ip1-12.e17.1.x86_64.rpm
(2/3): openssh-clients-6.6.ip1-12.e17.1.x86_64.rpm
(3/3): openssh-server-6.6.ip1-12.e17.1.x86_64.rpm
1 431 kB 00:00:00
1 634 kB 00:00:00
1 432 kB 00:00:01
1.1 MB/s | 1.5 MB 00:00:01
Total
Running transaction check
Running transaction test
Transaction test succeeded
Running transaction
Upgrading : openssh-6.6.ip1-12.e17.1.x86_64
Upgrading : openssh-clients-6.6.ip1-12.e17.1.x86_64
Upgrading : openssh-server-6.6.ip1-12.e17.1.x86_64
Upgrading : openssh-6.6.ip1-11.e17.x86_64
Cleanup : openssh-6.6.ip1-11.e17.x86_64
Verifying : openssh-clients-6.6.ip1-12.e17.1.x86_64
Verifying : openssh-server-6.6.ip1-12.e17.1.x86_64
Verifying : openssh-6.6.ip1-12.e17.1.x86_64
Verifying : openssh-server-6.6.ip1-11.e17.x86_64
Verifying : openssh-clients-6.6.ip1-11.e17.x86_64
Verifying : openssh-6.6.ip1-11.e17.x86_64
Updated:
openssh-clients.x86_64 0:6.6.ip1-12.e17_1
Dependency Updated:
openssh.x86_64 0:6.6.ip1-12.e17_1
openssh-server.x86_64 0:6.6.ip1-12.e17_1
Complete!
[root@localhost max]# yum install openssh-server
Loaded plugins: fastestmirror
Loading mirror speeds from cached hostfile
 * base: mirror.hhost.ru
 * extras: mirror.hhost.ru
 * updates: mirror.hhost.ru
Package openssh-server-6.6.ip1-12.e17.1.x86_64 already installed and latest version
Nothing to do
[root@localhost max]
```

Для входа в эту виртуальную машину, щелкните внутри окна или нажмите Ctrl+G.



```
CentOS 64-bit - VMware Workstation
Файл Правка Вид Виртуальная машина Вкладки Справка ||| □ ○ X
Библиотека x Главное x CentOS 64-bit x
Скачано 0 из 0 МБ
Скачивание завершено
Loaded plugins: fastestmirror
Loading mirror speeds from cached hostfile
 * base: mirror.hhost.ru
 * extras: mirror.hhost.ru
 * updates: mirror.hhost.ru
No package ssh-clients available.
Error: Nothing to do
[root@localhost max]# yum install openssh-clients
Loaded plugins: fastestmirror
Loading mirror speeds from cached hostfile
 * base: mirror.hhost.ru
 * extras: mirror.hhost.ru
 * updates: mirror.hhost.ru
Resolving Dependencies
--> Running transaction check
--> Package openssh-clients.x86_64 0:6.6.ip1-11.e17 will be updated
--> Package openssh-clients.x86_64 0:6.6.ip1-12.e17_1 will be an update
--> Processing Dependency: openssh = 6.6.ip1-12.e17_1 for package: openssh-clients-6.6.ip1-12.e17.1.x86_64
--> Running transaction check
--> Package openssh-6.6.ip1-11.e17.x86_64 will be updated
--> Package openssh-6.6.ip1-11.e17.x86_64 is needed by dependency for package: openssh-server-6.6.ip1-11.e17.x86_64
--> Package openssh.x86_64 0:6.6.ip1-12.e17_1 will be an update
--> Running transaction check
--> Package openssh-server.x86_64 0:6.6.ip1-11.e17 will be updated
--> Package openssh-server.x86_64 0:6.6.ip1-12.e17_1 will be an update
--> Finished Dependency Resolution

Dependencies Resolved

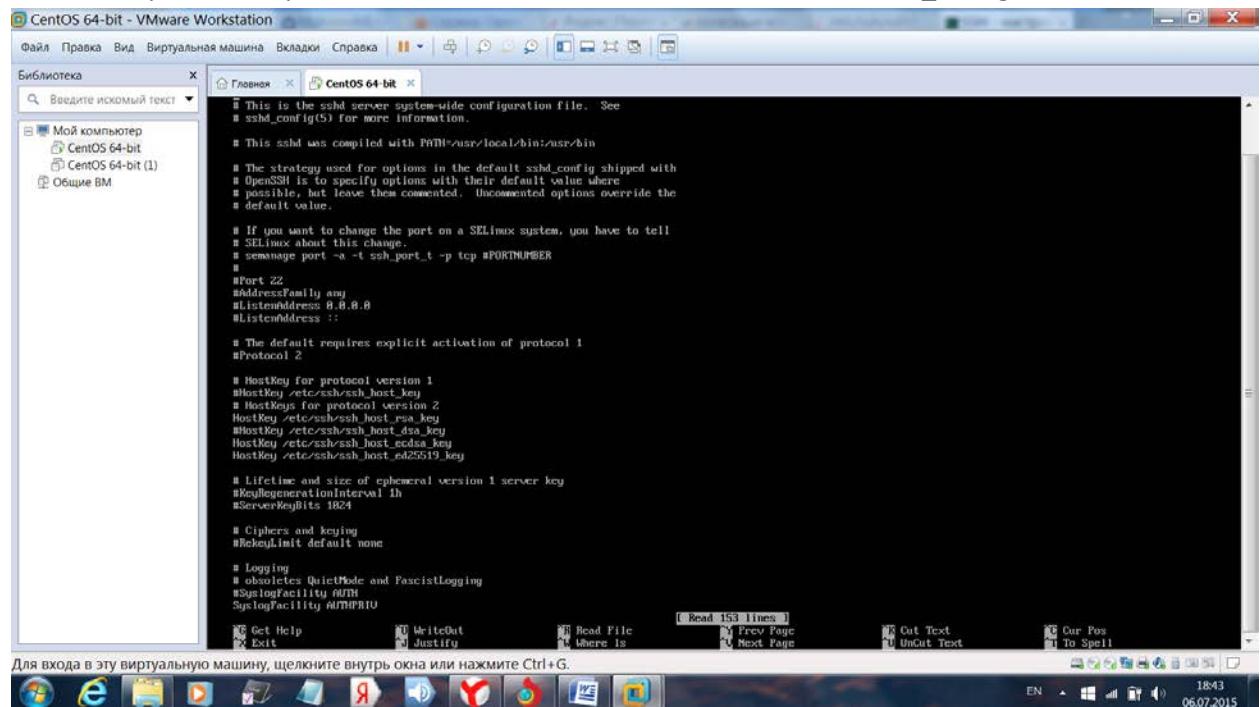
-----
Package           Arch      Version            Repository        Size
-----
Upgrading:
openssh-clients   x86_64   6.6.ip1-12.e17_1   updates          634 k
Updating for dependencies:
openssh           x86_64   6.6.ip1-12.e17_1   updates          431 k
openssh-server    x86_64   6.6.ip1-12.e17_1   updates          432 k

Transaction Summary
-----
Upgrade 1 Package (>2 Dependent packages)
Total download size: 1.5 M
Is this ok [y/d/N]:
```

Для входа в эту виртуальную машину, щелкните внутри окна или нажмите Ctrl+G.



Заходим в файл с настройками с помощью команды “`nano /etc/ssh/sshd_config`”



Видим следующие директивы:

Port 22 — указывает порт по которому сервер будет ожидать входящего соединения. Заменим его на 2496

AddressFamily-семейство адресов которое должна использовать служба `sshd(8)`. Допустимые значения: ``any'' ``inet'' (только IPv4) и ``inet6'' (только IPv6). Значение по умолчанию - ``any''

Далее идут строки:

“ListenAddress 0.0.0.0”

“ListenAddress ::”

Эти строки отвечают за настройку разграничений по сетевым интерфейсам, сетевому адресу или имени компьютера. По умолчанию сервер «слушает» (принимает подключения) на всех сетевых интерфейсах.

Protocol — позволяет выбрать версию протокола 1 или 2. Рекомендуется протокол 2.

Строки HostKey необходимы для второй версии протокола SSH и отвечают за названия файлов ключей и их расположение, эти ключи используются при аутентификации с ключом хоста

Следующие строки относятся к версии протокола 1 :

KeyRegenerationInterval 1h

ServerKeyBits 1024

Следующая группа параметров относится к аутентификации, первый параметр(`LoginGraceTime`) означает, что соединение будет разорвано через указанное количество секунд, если пользователь не войдёт в систему .

Второй параметр(**PermitRootLogin**) разрешает или запрещает вход по SSH под суперпользователем(root)

Третий параметр (**StrictModes**) включает проверку демоном ssh прав и владение домашним каталогом пользователя, который пытается получить удалённый доступ к компьютеру.

3) Заходим с одной виртуальной машины на другую используя ssh-соединение.

Для этого:

- создаём вторую виртуальную машину, производим соответствующие настройки сети и установку софта, аналогично первой ВМ

- для настройки ssh-соединения используем openssh-server, устанавливаем на обе ВМ пользуясь командами “**yum install openssh-server**” и “**yum install openssh-clients**”

```
root@localhost max# dchclient
root@localhost max# yum install openssh-server
Loaded plugins: fastestmirror
base
extras
updates
Loading mirror speeds from cached hostfile
* base: mirror.hhost.ru
* extras: mirror.hhost.ru
* updates: mirror.hhost.ru
Resolving Dependencies
--> Running transaction check
--> Package openssh-server.x86_64 0:6.6.1p1-12.el7_1 will be installed
--> Finished Dependency Resolution

Dependencies Resolved

=====
Package          Arch      Version           Repository      Size
=====
Installing:
openssh-server   x86_64    6.6.1p1-12.el7_1      updates       432 k

Transaction Summary
=====
Install 1 Package

Total download size: 432 k
Installed size: 916 k
Is this ok [y/d/N]: y
Downloading packages:
openssh-server-6.6.1p1-12.el7_1.x86_64.rpm
Running transaction check
Running transaction test
Transaction test succeeded
Running transaction
  Installing : openssh-server-6.6.1p1-12.el7_1.x86_64
  Verifying  : openssh-server-6.6.1p1-12.el7_1.x86_64

Installed:
  openssh-server.x86_64 0:6.6.1p1-12.el7_1

Complete!
root@localhost max# yum install openssh-clients
```

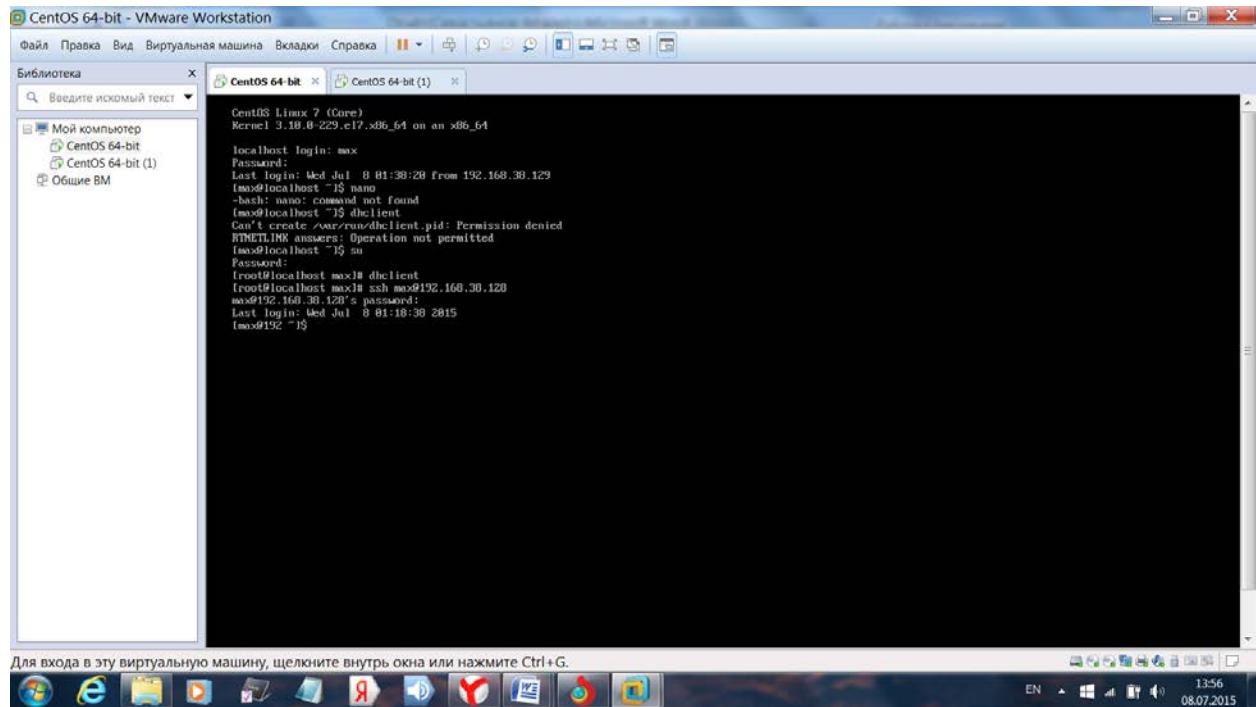
- с помощью текстового редактора nano и команды “**nano /etc/ssh/sshd_config**” открываем и редактируем конфигурационные файлы.

- узнаем ip с помощью команды “**ip a**”. Ip: 192.168.38.128

- теперь с основной системы соединяемся удалённым сервером с помощью команды “**ssh max@ip**”

- Затем получаем предупреждение, что подлинность хоста не установлена, видим ESCDA ключ и подтверждаем подключение, получаем уведомление о том, что наш айпи запомнен, вводим пароль

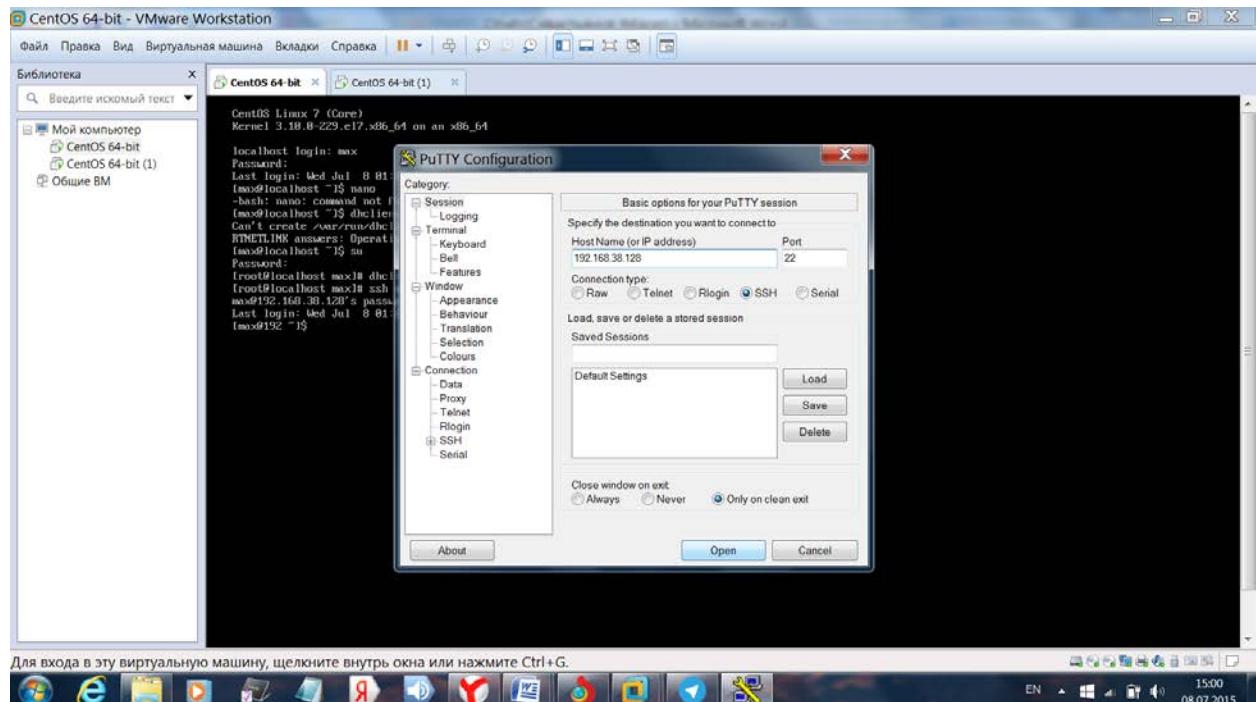
удалённого сервера и таким образом получаем доступ к удаленному хосту.



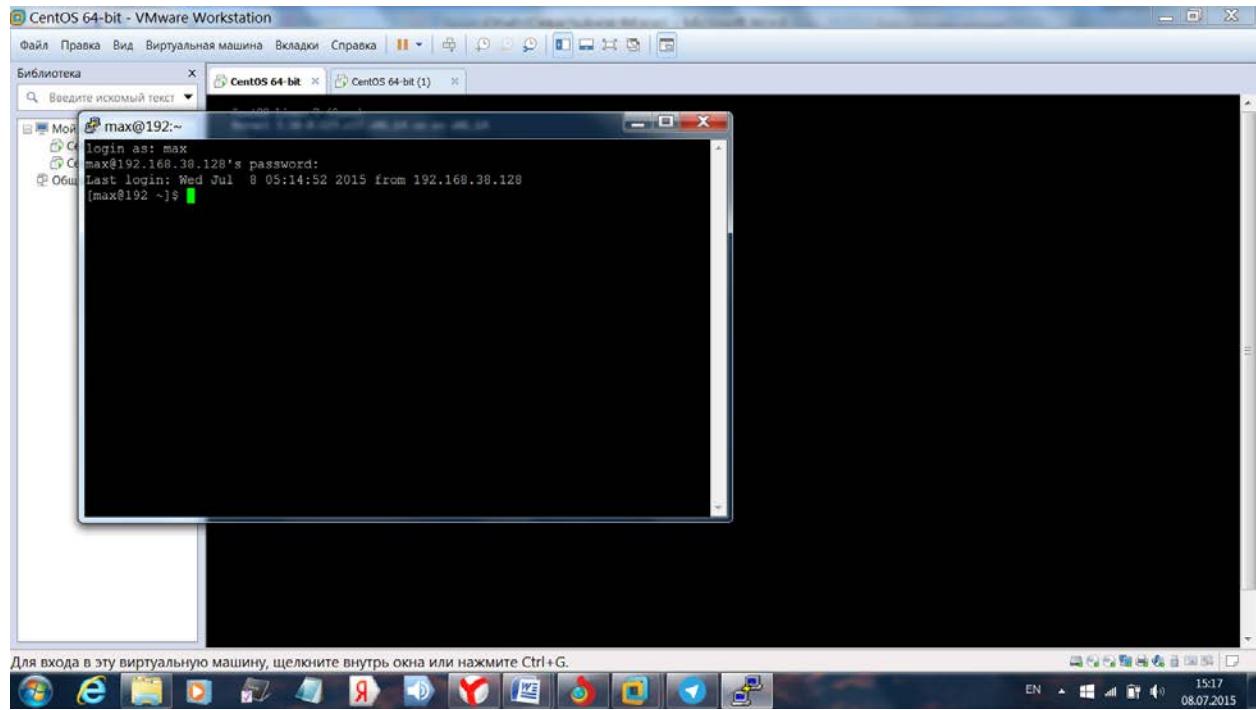
4) Научиться с помощью putty управлять виртуальной машиной с реальной машины

-Скачиваем putty с <http://www.putty.org/>

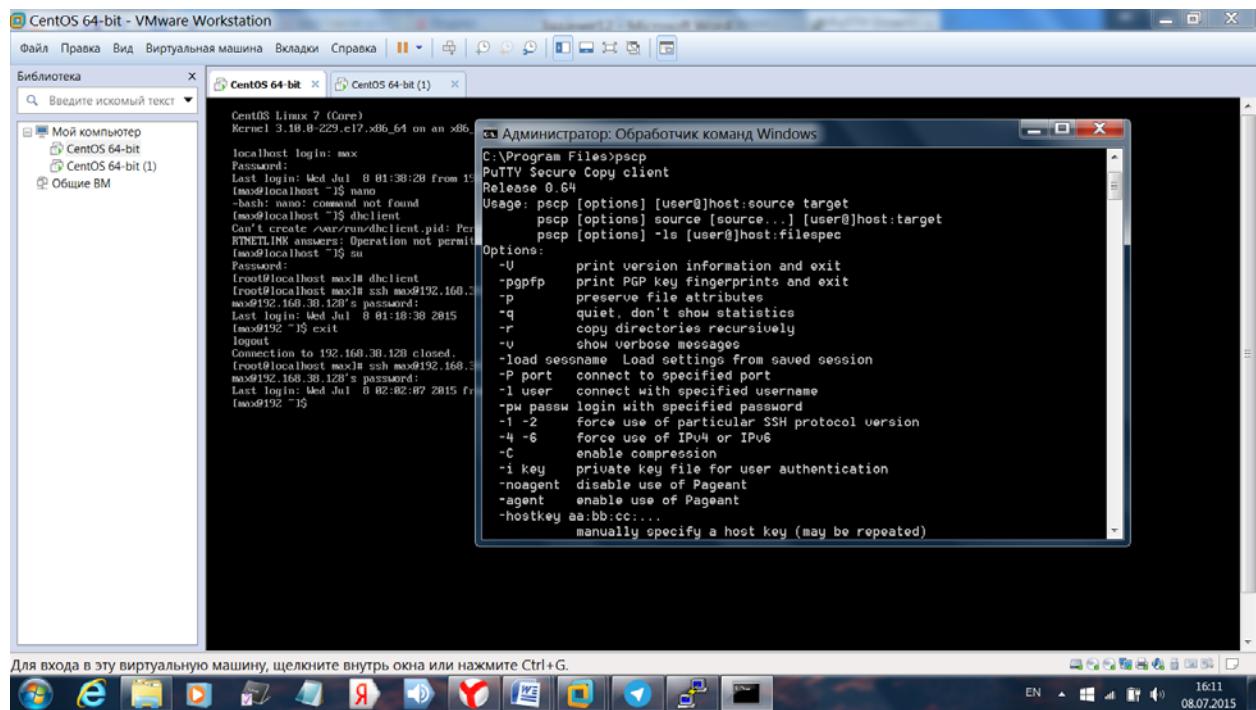
-Открываем putty и вводим ip виртуальной машины



-Открывается окно, в которое мы вводим логин и пароль для подключения к ВМ

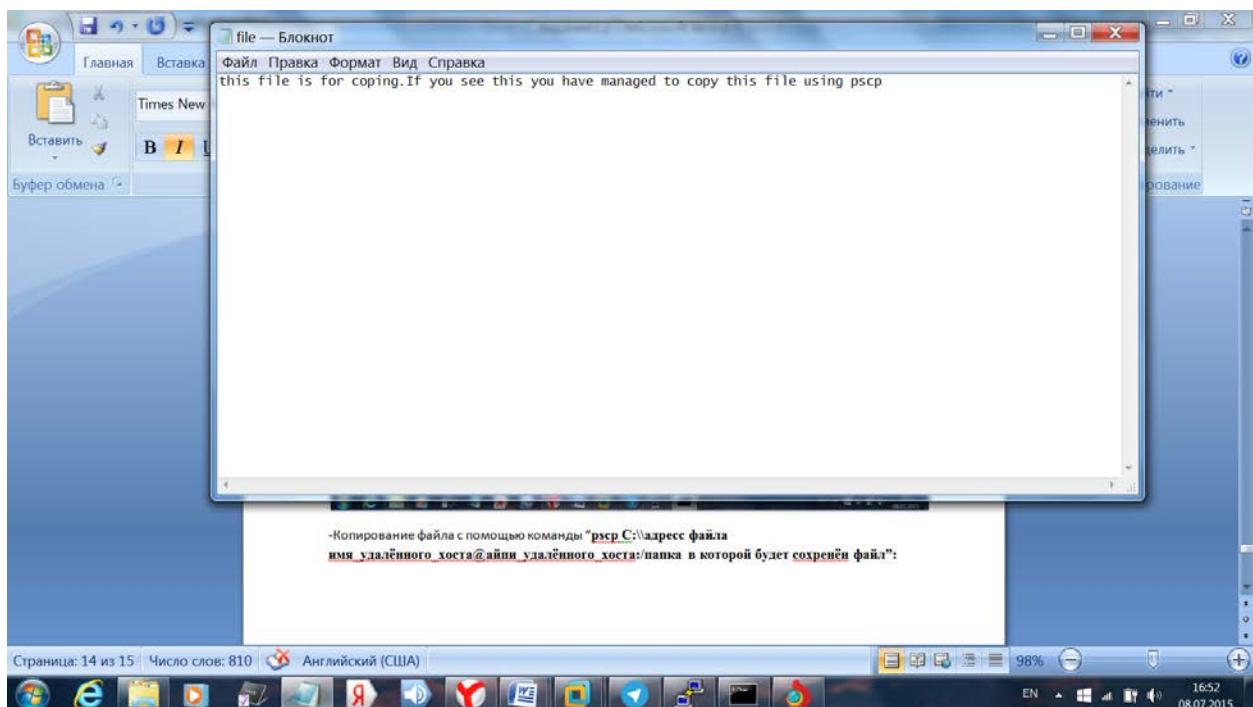


-Скачиваем утилиту pscp и открываем ее через Обработчик команд Windows

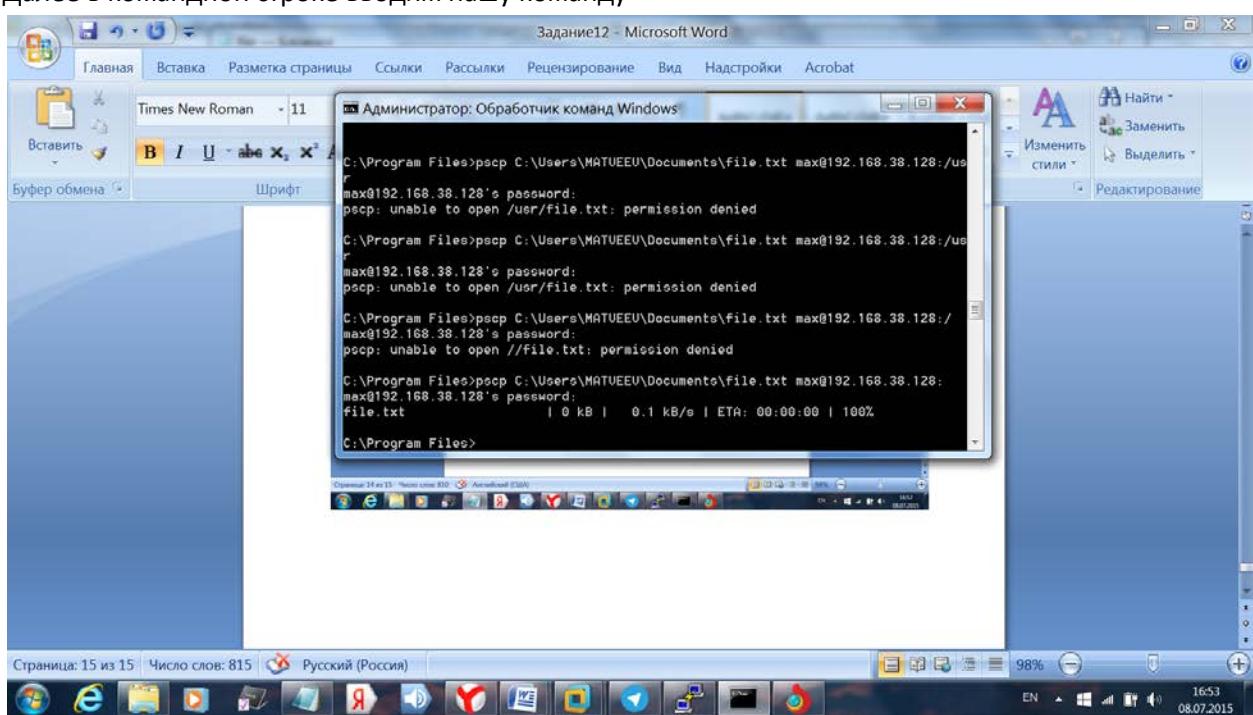


-Копирование файла с помощью команды “**pscp C:\\адрес файла
имя_удалённого_хоста@айпи_удалённого_хоста:/папка в которой будет сохранён файл”:**

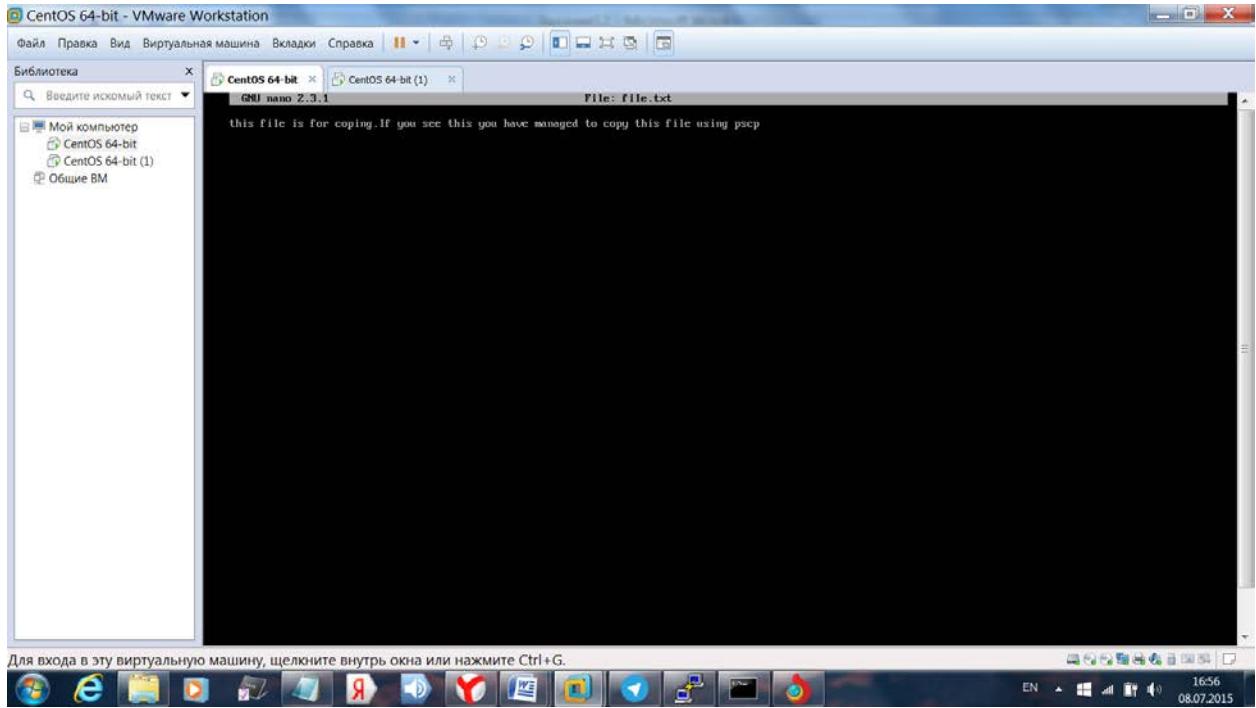
Создание файла на реальной машине:



Далее в командной строке вводим нашу команду

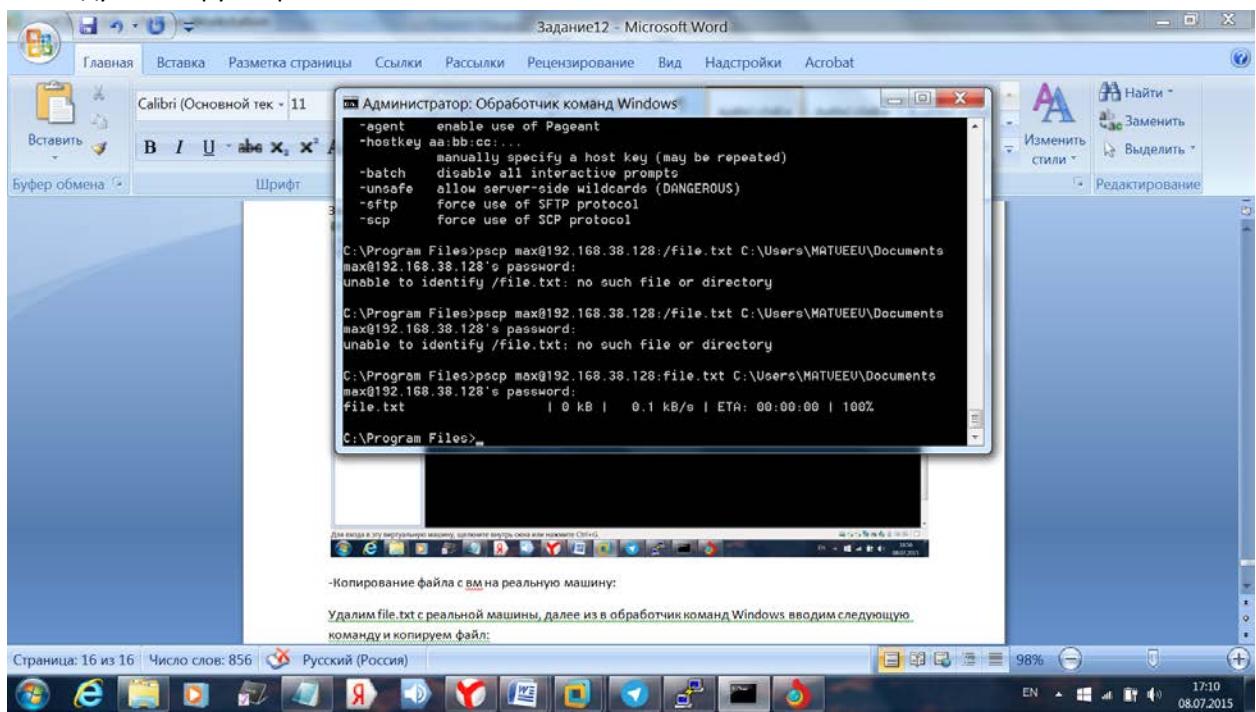


Заходим с вм и открываем скопированный файл с помощью nano:

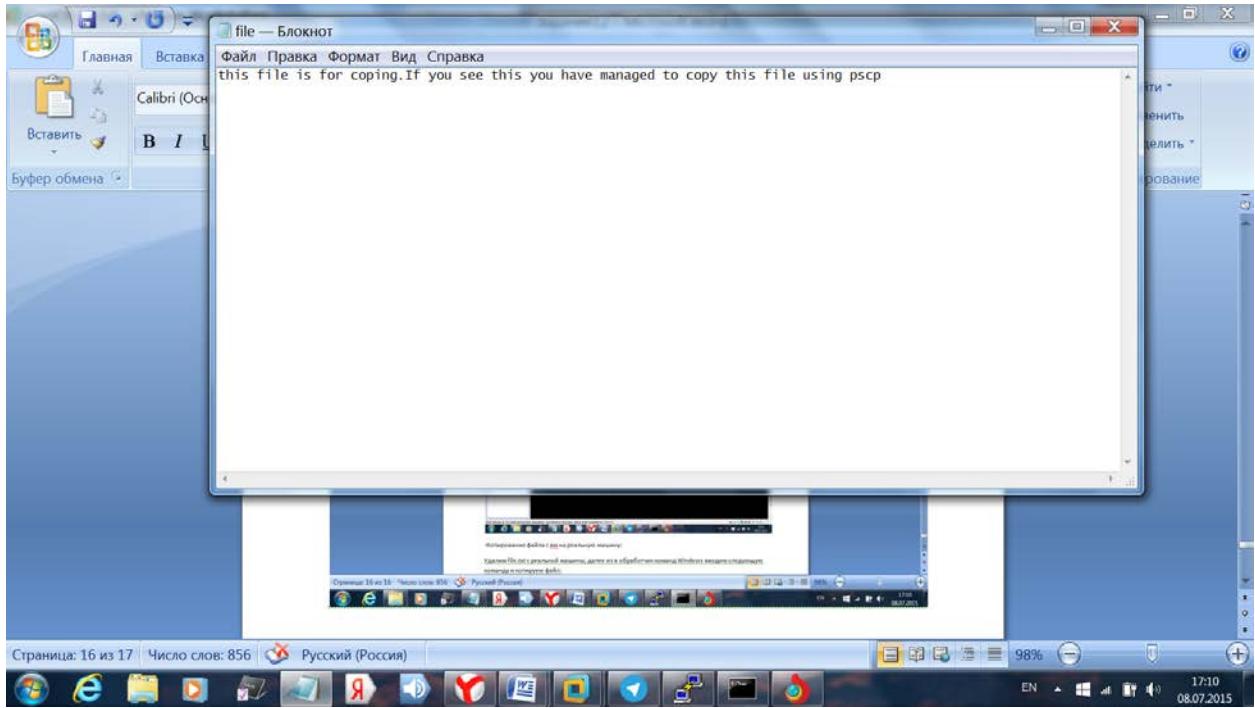


-Копирование файла с вм на реальную машину помошью команды “**pscp**
имя_удалённого_хоста@айпи_удалённого_хоста:/папка в которой харанится файл адрес нового расположения файла на реальной машине”

Удалим file.txt с реальной машины, далее из в Обработчика команд Windows вводим следующую команду и копируем файл:



Затем открываем скопированный файл:



5) SSH ключи

Ключи SSH служат средством идентификации вас при подключении к серверу SSH с использованием [крипtosистемы с открытым ключом](#) и [аутентификации вызов-ответ](#). Одним из непосредственных достоинств этого метода перед традиционной идентификацией с помощью пароля является то, что вы можете быть авторизованы на сервере без регулярной необходимости отсылать ваш пароль через сеть. Даже если кто-либо будет прослушивать ваше соединение, у него не будет возможности перехватить и взломать ваш пароль, поскольку фактически он никогда не передается. Также использование для идентификации ключей SSH устраняет риск, связанный с брут-форс (brute-force) атаками, за счет существенного уменьшения шанса атакующего угадать правильные учетные данные.

Идентификация при помощи ключей SSH предоставляет дополнительную безопасность, а также может быть более удобным способом, чем традиционная идентификация при помощи пароля. При использовании вместе с программой, называемой агентом SSH, ключи SSH могут подключать вас к серверу или нескольким серверам без необходимости помнить и вводить ваш пароль отдельно для каждой системы.

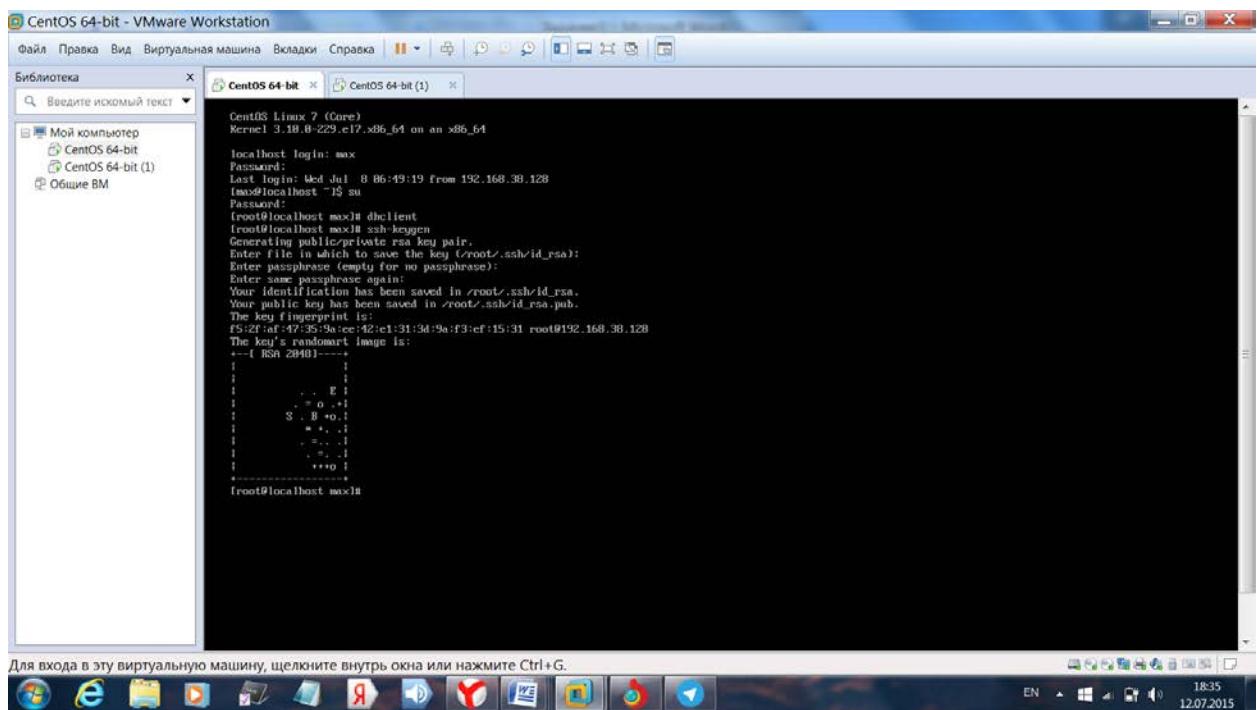
Ключи SSH являются парными: один из них - закрытый, другой - открытый. Закрытый ключ известен только вам, и он должен быть в безопасности. С другой стороны, открытый ключ может свободно раздаваться с любого сервера SSH, к которому вы хотите подключиться.

Когда у сервера SSH есть ваш открытый ключ в файле, и он видит, что вы запрашиваете соединение, он использует этот открытый ключ, чтобы создать и отправить вам т.н. вызов. Этот вызов является чем-то вроде зашифрованного сообщения, на которое должен поступить соответствующий ответ, чтобы сервер предоставил вам доступ. Безопасным это сообщение делает тот факт, что оно может быть прочитано только кем-то, у кого есть закрытый ключ. Открытый ключ может быть использован для зашифровки сообщения, но расшифровать то же самое сообщение он не сможет. Только вы, держатель закрытого ключа, будете иметь возможность корректно принять вызов и создать соответствующий ответ.

Этот этап вызов-ответ проходит незаметно для пользователя. До тех пор, пока у вас есть закрытый ключ, который обычно хранится в каталоге `~/.ssh/`, ваш клиент SSH будет иметь возможность отправить правильный ответ серверу.

Поскольку закрытые ключи считаются конфиденциальной информацией, обычно они хранятся на диске в зашифрованном виде. По этой причине, когда запрашивается закрытый ключ, необходимо ввести пароль для расшифровки этого ключа. Внешне это может быть похоже на ввод пароля непосредственно на сервере SSH, но это не так: этот пароль используется только для расшифровки закрытого ключа в вашей локальной системе. Этот пароль не передается и не должен передаваться через сеть.

Пара ключей SSH может быть сгенерирована при помощи команды `ssh-keygen`:



Пара ключей сгенерирована

Далее зайдем в директорию /root/ssh и сможем увидеть три файла : id_rsa id_rsa.pub и known_hosts.

Подключаемся к удаленному хосту:

CentOS 64-bit - VMware Workstation

Файл Правка Вид Виртуальная машина Вкладки Справка |||

Библиотека x CentOS 64-bit x CentOS 64-bit (1) x

CentOS Linux 7 (Core)
Kernel 3.18.8-229.el7.x86_64 on an x86_64

localhost login: max
Password:
Last login: Wed Jul 8 06:49:19 from 192.168.30.128
max@localhost ~\$ su
Password:
root@localhost ~# sshclient
(root@localhost max) \$ ssh-keygen
Generating public/private key pair.
Enter file in which to save the key (/root/.ssh/id_rsa):
Enter passphrase (empty for no passphrase):
Enter same passphrase again:
Your identification has been saved in /root/.ssh/id_rsa.
Your public key has been saved in /root/.ssh/id_rsa.pub.
The key fingerprint is:
15:2f:af:47:35:9a:ee:42:e1:3d:9a:f3:f1:15:31 root@192.168.30.128
The key's randomart image is:
+---+
| . |
| . |
| . E |
| . = o .+|
S . B +o+.|
| +. . . |
| |
| |
| |
| |
| |
root@localhost ~# cd /root/.ssh
root@localhost .ssh# ll
bash: ll: command not found
root@localhost .ssh# ls
id_rsa id_rsa.pub known_hosts
root@localhost .ssh# cd
root@localhost .ssh# ssh-keygen -t rsa
root@localhost .ssh# ssh max@192.168.30.128
max@192.168.30.128's password:
Last login: Wed Jul 8 12:52:07 2015
max@localhost ~\$

-С помощью команды “`cp /root/.ssh/id_rsa.pub max@192.168.38.128`” копируем файл id_rsa.pub в домашний каталог на удаленный сервер.

-Создаем каталог /root/.ssh(если его нет) на удаленном сервере и добавляем id_rsa.pub в authorized_keys с помощью команды “`cat /root/.ssh/id_rsa.pub >> /root/.ssh/authorized_keys`”.

Теперь можно подключаться следующим образом:

The screenshot shows a VMware Workstation interface with a CentOS 64-bit virtual machine running. The terminal window displays an SSH session log. The log shows the user 'max' logging in from IP 192.168.38.128. It includes standard SSH configuration parameters like LogLevel, Authentication, and RSAAuthentication, followed by a series of login attempts and successes. The session ends with the user exiting and the terminal closing.

```
# Logging
# obsolete QuietMode and FascistLogging
#SyslogFacility AUTH
#SyslogFacility AUTHPRIV
#LogLevel INFO

# Authentication:

#LoginGraceTime 2m
#PermitRootLogin no
#StrictModes yes
#MaxAuthTries 6
#MaxSessions 10

RSAAuthentication yes
PubkeyAuthentication yes

# The default is to check both .ssh/authorized_keys and .ssh/authorized_keys2
# but this is overridden so installations will only check .ssh/authorized_keys
AuthorizedKeysFile .ssh/authorized_keys

#AuthorizedPrincipalsFile none

#AuthorizedKeysCommand none
#AuthorizedKeysCommandUser nobody

[ Wrote 153 times ]

root@localhost max$ exit
exit
(max@localhost ~)$ ssh /root/.ssh/id_rsa.pub max@192.168.38.128
ssh: Could not resolve hostname /root/.ssh/id_rsa.pub: Name or service not known
(max@localhost ~)$ ssh -i /root/.ssh/id_rsa.pub max@192.168.38.128
Warning: Identity file /root/.ssh/id_rsa.pub not accessible: Permission denied.
(max@192.168.38.128 ~)$ Last login: Sun Jul 12 09:44:38 2015 from 192.168.38.128
(max@localhost ~)$ exit
logout
Connection to 192.168.38.128 closed.
(max@localhost ~)$ ssh -i /root/.ssh/id_rsa.pub max@192.168.38.128
Warning: Identity file /root/.ssh/id_rsa.pub not accessible: Permission denied.
(max@192.168.38.128 ~)$ Last login: Sun Jul 12 10:09:07 2015 from 192.168.38.128
(max@localhost ~)$
```

6) FTP

FTP — стандартный [протокол](#), предназначенный для передачи файлов по TCP-сетям (например, Интернет). Использует 21й порт. FTP часто используется для загрузки сетевых страниц и других документов с частного устройства разработки на открытые [сервера хостинга](#).

Протокол построен на архитектуре «[клиент-сервер](#)» и использует разные сетевые соединения для передачи команд и данных между клиентом и сервером. Пользователи FTP могут пройти аутентификацию, передавая логин и пароль [открытым текстом](#), или же, если это разрешено на сервере, они могут подключиться анонимно. Можно использовать протокол [SSH](#) для безопасной передачи, скрывающей (шифрующей) логин и пароль, а также шифрующей содержимое.

Первые клиентские FTP-приложения были интерактивными инструментами командной строки, реализующими стандартные команды и синтаксис. [Графические пользовательские интерфейсы](#) с тех пор были разработаны для многих используемых по сей день операционных систем. Среди этих интерфейсов как программы общего веб-дизайна вроде [Microsoft Expression Web](#), так и специализированные FTP-клиенты (например, FileZilla).

FTP является одним из старейших прикладных протоколов, появившимся задолго до [HTTP](#), и даже до [TCP/IP](#), в 1971 году. В первое время он работал поверх протокола [NCP](#)^[1]. Он и сегодня широко используется для распространения [ПО](#) и доступа к удалённым [хостам](#).

Доступ к файлам на удаленном компьютере по протоколу FTP осуществляется с помощью программ, которые называются **FTP-клиентами** (в качестве примитивного FTP-клиента может использоваться [www-браузер](#), например Opera, Firefox или Microsoft Internet Explorer).

Практически все современные операционные системы включают также FTP-клиент для работы в командной строке, который так и называется «ftp».

Если у вас есть интернет, то вы можете получить доступ к большому количеству информации, расположенной в различных уголках Сети. Для использования FTP необходим так называемый FTP-клиент, подключающийся к FTP-серверу (сервер, откуда скачиваются данные). Анонимный FTP позволяет подключаться к серверу даже не будучи на нем зарегистрированным (не имея на нем логина и пароля). Как правило, в качестве логина (имени пользователя) указывается anonymous, а в качестве пароля — ваш e-mail. Это делается на больших серверах для того, чтобы каждый мог скачать, к примеру, бесплатный дистрибутив Linux или какие-нибудь другие полезные программы.

FTP-клиент общается с FTP-сервером при помощи специальных FTP-команд (в зависимости от сервера они могут незначительно отличаться, но в целом набор команд более-менее стандартен). Тогда почему бы не дать возможность пользователю вводить эти команды, чтобы без посредника просматривать содержимое FTP-сервера, закачивать файлы, устанавливать режимы передачи. Именно так и было на заре интернета. Первопроходцы глобальной паутины торопливо набирали команды для FTP-сервера прямо из консоли. Существует такая возможность и поныне. Правда, если для пользователей Unix-систем такая манера общения с FTP весьма привычна, то обладатели Windows наверняка даже и не подозревают, что их система позволяет это делать. Надо отметить, что использовать консольный вариант FTP-клиента поначалу очень даже увлекательно, а в некоторых случаях и чрезвычайно полезно.

SFTP (SSH File Transfer Protocol) является сетевым протоколом, который обеспечивает функциональность передачи файлов и манипулирование по любому надежному потоку данных. Он обычно используется с протоколом SSH-2 (TCP port 22) для обеспечения безопасной передачи файлов, но предназначен и для работы с другими протоколами.

Плюсы: Хороший стандарт, который строго определяют большинство (если не все) аспектов деятельности. Имеет только одно подключение (нет необходимости подключения к DATA). Подключение всегда защищено .Список каталогов является однородным и машиночитаемым Протокол включает операции для разрешения и манипулирования атрибутом, захвата файла и отличается большей функциональностью

Минусы: Передача двоичная и не удобна для чтения пользователем. Ключами SSH труднее управлять и проверять. Стандарты определяют определенные вещи, как optionalные или рекомендованные, что приводит к определенным проблемам совместимости между разными названиями программного обеспечения от разных производителей. Нет копирования сервер-сервер и рекурсивных операций по удалению каталога Нет встроенной SSH / SFTP поддержки в VCL и .NET фреймворках.

-Устанавливаем FTP с помощью команды “**yum install vsftpd**”:

```

CentOS Linux 7 (Core)
Kernel 3.10.0-223.el7.x86_64 on an x86_64
localhost login: max
Password:
Last login: Sun Jul 12 18:09:29 from 192.168.38.128
tmax@localhost ~]$ su
Password:
[root@localhost ~]# yum install vsftpd
Loaded plugins: fastestmirror
base
extras
updates
Loading mirror speeds from cached hostfile
* base: mirror.hihost.ru
* extras: mirror.hihost.ru
* updates: mirror.hihost.ru
Resolving Dependencies
--> Running transaction check
--> Package vsftpd.x86_64 0:3.0.2-9.el7 will be installed
--> Finished Dependency Resolution

Dependencies Resolved

=====
Package           Arch      Version        Repository  Size
=====
Installing:
vsftpd            x86_64   3.0.2-9.el7    base       165 k

Transaction Summary
=====
Install 1 Package

Total download size: 165 k
Installed size: 343 k
Is this ok [y/d/N]:
```

Для входа в эту виртуальную машину, щелкните внутри окна или нажмите Ctrl+G.

-С помощью команды “**nano /etc/vsftpd/vsftpd.config**” открываем файл с настройками:

```

CentOS 6.4 - VMware Workstation
File Edit View Virtual Machine Bookmarks Help
Library Search Enter search text ...
CentOS 6.4 - CentOS 6.4 (1)
My Computer CentOS 6.4-bit CentOS 6.4-bit (1)
General View
The default compiled-in settings are fairly paranoid. This sample file
loosens things up a bit, to make the ftp daemon more usable.
Please see vsftpd.conf.5 for all compiled-in defaults.

READ THIS: This example file is NOT an exhaustive list of vsftpd options.
Please read the vsftpd.conf.5 manual page to get a full idea of vsftpd's
capabilities.

# Allow anonymous FTP? (Beware - allowed by default if you comment this out).
anonymous_enable=YES

# Uncomment this to allow local users to log in.
# When SELinux is enforcing check for SE bool ftp_home_dir
local_enable=YES

# Uncomment this to enable any form of FTP write command.
write_enable=YES

# Default umask for local users is 077. You may wish to change this to 022,
# If your users expect that (022 is used by most other Ftpd's)
local_umask=022

# Uncomment this to allow the anonymous FTP user to upload files. This only
# has an effect if the above global write enable is activated. Also, you will
# obviously need to create a directory writable by the FTP user.
# When SELinux is enforcing check for SE bool allow_ftpd_anon_write, allow_ftpd_full_access
anon_upload_enable=YES

# Uncomment this if you want the anonymous FTP user to be able to create
# new directories.
anon_mkdir_write_enable=YES

# Activate directory messages - messages given to remote users when they
# go into a certain directory.
dirmessage_enable=YES

# Activate logging of uploads/downloads.
xferlog_enable=YES

# Make sure PORT transfer connections originate from port 20 (ftp-data).
connect_from_port_20=YES
```

For entry in this virtual machine, click inside the window or press Ctrl+G.

Внесем следующие изменения :

Опция	Описание
anonymous_enable=NO	Запрещаем анонимный доступ

local_enable=YES	Разрешаем доступ локальным пользователям
write_enable=YES	Даем пользователям FTP права на запись
connect_from_port_20=NO	Отключаем 20 порт, уменьшает привилегии VSftpd
chroot_local_user=YES	Chroot всех пользователей
local_umask=022	Устанавливаем маску 022, чтобы быть уверенными в том, что для всех файлов (644) и папок (755) которые мы закачиваем, устанавливаются соответствующие права

-Задаем пароль и логин для этого пользователя

- Также установим FTP клиент, для подключения к FTP серверу

