Работу выполнил студент

Абраамян Александр Манвелович,
группа 5130904/10101,
4 курс

Ответы на вопросы

1. Что такое IP адрес, MAC адрес, маска подсети, порт

IP Адрес

IP адрес - это уникальный идентификатор компьютера или другого устройства в сети, который позволяет устройствам общаться между собой. IP адрес состоит из четырех чисел, разделенных точками (например, 192.0.2.1).

Простыми словами - это адрес компьютера. В разных сетях он может быть разным.

MAC Адрес

MAC адрес - это уникальный идентификатор сетевого интерфейса, который присваивается производителем устройства. MAC адрес состоит из шести групп из двух шестнадцатеричных цифр, разделенных двоеточиями (например, 00:11:22:33:44:55).

Маска подсети

Маска подсети - это уникальный идентификатор, который указывает на то, какие часть IP адреса является адресом сети, а какие - адресом хоста. Маска подсети состоит из четырех чисел, разделенных точками (например, 255.255.255.0).

Маска подсети нам нужна чтобы определить какие части IP адреса являются адресом сети, а какие - адресом хоста. Например роутер может посмотреть на адрес куда хочет обратиться пользователь и определить - он хочет обратиться в локанькой сети или куда то на удаленную сеть

Порт

Порт - это уникальный идентификатор, который позволяет компьютеру или другому устройству определять, какие данные предназначены для какой программы. Порт - это целое положительное число от 0 до 65535 (например, 80).

Порты обычно используют приложения для сетевого взаимодействия.

2. Чем хорош и чем плох Telnet

Telnet

Telnet - это протокол, который позволяет компьютеру или устройству общаться с другим устройством, используя текстовые команды. Telnet позволяет эмулировать терминал, что позволяет контролировать устройство, как если бы вы сидели перед ним.

Плюсы

- Telnet позволяет эмулировать терминал, что позволяет контролировать устройство, как если бы вы сидели перед ним.
- Telnet позволяет получать информацию о состоянии устройства.

Минусы

- Telnet обладает слабой защищенностью (man in the middle)
- Telnet не предоставляет безопасных методов аутентификации
- Telnet не предоставляет шифрование данных

3. Как можно улучшить безопасность подключения при использовании SSH

SSH

SSH (Secure Shell) - это протокол, который используется для безопасного подключения к удалённым устройствам и управления ими через незащищённые сети. SSH обеспечивает шифрование данных, аутентификацию и целостность соединения.

Способы улучшения безопасности подключения при использовании SSH

- **Использование SSH-ключей**: Вместо паролей можно использовать SSH-ключи для аутентификации. Это более безопасный метод, так как ключи сложнее перехватить и взломать.
- **Ограничение доступа**: Можно настроить доступ только с определённых IP-адресов и отключить рутовый доступ по SSH.
- **Обновление SSH-сервера**: Регулярно обновлять SSH-сервер, чтобы устранить уязвимости и защититься от новых угроз.
- **Изменение стандартного порта**: Можно изменить стандартный порт 22 на другой, чтобы уменьшить количество атак методом подбора.

4. Когда стоит использовать авторизацию по паролю, а когда по ключам, при подключении по SSH

Авторизация по паролю

Авторизация по паролю - это способ аутентификации, при котором пользователь вводит пароль, заранее установленный на сервере. Это простой способ, но у него есть свои недостатки.

Недостатки

- Пароль может быть легко взломан при помощи метода подбора.
- Пароль может быть легко забыт.

Авторизация по ключам

Авторизация по ключам - это способ аутентификации, при котором пользователь использует пару приватного и публичного ключей. Это более безопасный способ, так как ключи сложнее перехватить и взломать.

Задания

1. Какую информацию можно узнать с помощью команд ifconfig (или ip) и netstat. Приведите примеры

Наиболее часто используемые формы команды ip:

```
ip l
ip a
ip r
```

Выполнив эти команды по очереди на своём полурабочем компьютере я увидел следующие результаты:

ip l

```
➜  ~ ip l
1: lo: <LOOPBACK,UP,LOWER_UP> mtu 65536 qdisc noqueue state UNKNOWN mode
DEFAULT group default qlen 1000
    link/loopback 00:00:00:00:00:00 brd 00:00:00:00:00:00
2: enp3s0: <NO-CARRIER,BROADCAST,MULTICAST,UP> mtu 1500 qdisc fq_codel
state DOWN mode DEFAULT group default qlen 1000
    link/ether 00:d8:61:e4:fb:fe brd ff:ff:ff:ff:ff:ff
3: wlo1: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc noqueue state UP
mode DORMANT group default qlen 1000
    link/ether 58:96:1d:16:f7:10 brd ff:ff:ff:ff:ff:ff
    altname wlp0s20f3
4: ham0: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1404 qdisc fq_codel state
UNKNOWN mode DEFAULT group default qlen 1000
    link/ether 7a:79:19:21:68:7d brd ff:ff:ff:ff:ff:ff
5: br-0ca37458415e: <NO-CARRIER,BROADCAST,MULTICAST,UP> mtu 1500 qdisc
noqueue state DOWN mode DEFAULT group default
    link/ether 02:42:ba:9a:ec:f9 brd ff:ff:ff:ff:ff:ff
6: docker0: <NO-CARRIER,BROADCAST,MULTICAST,UP> mtu 1500 qdisc noqueue
state DOWN mode DEFAULT group default
    link/ether 02:42:8a:55:16:38 brd ff:ff:ff:ff:ff:ff
7: br-57fda32ea20b: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc
noqueue state UP mode DEFAULT group default
    link/ether 02:42:92:aa:48:21 brd ff:ff:ff:ff:ff:ff
```

```
8: br-f9c36026e0cf: <NO-CARRIER,BROADCAST,MULTICAST,UP> mtu 1500 qdisc
noqueue state DOWN mode DEFAULT group default
    link/ether 02:42:34:92:ca:79 brd ff:ff:ff:ff:ff:ff
10: veth904c2bb@if9: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc
noqueue master br-57fda32ea20b state UP mode DEFAULT group default
    link/ether a2:fa:0d:1c:d5:df brd ff:ff:ff:ff:ff:ff link-netnsid 1
12: veth324457a@if11: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc
noqueue master br-57fda32ea20b state UP mode DEFAULT group default
    link/ether 9e:ce:67:61:fa:84 brd ff:ff:ff:ff:ff:ff link-netnsid 0
13: neko-tun: <POINTOPOINT,MULTICAST,NOARP,UP,LOWER_UP> mtu 9000 qdisc
fq_codel state UNKNOWN mode DEFAULT group default qlen 500
    link/none
15: tun0: <POINTOPOINT,MULTICAST,NOARP,UP,LOWER_UP> mtu 1420 qdisc fq_codel
state UNKNOWN mode DEFAULT group default qlen 500
    link/none
```

Команда показала список доступных сетевых интерфейсов некоторые из них в состоянии UP, некоторые в состоянии DOWN, некоторые в состоянии UNKNOWN.

ip a

Более подробное описание интерфейсов можно получить выполнив команду `ip a`

```
➜  ~ ip a
1: lo: <LOOPBACK,UP,LOWER_UP> mtu 65536 qdisc noqueue state UNKNOWN group
default qlen 1000
    link/loopback 00:00:00:00:00:00 brd 00:00:00:00:00:00
    inet 127.0.0.1/8 scope host lo
       valid_lft forever preferred_lft forever
    inet6 ::1/128 scope host
       valid_lft forever preferred_lft forever
2: enp3s0: <NO-CARRIER,BROADCAST,MULTICAST,UP> mtu 1500 qdisc fq_codel
state DOWN group default qlen 1000
    link/ether 00:d8:61:e4:fb:fe brd ff:ff:ff:ff:ff:ff
3: wlo1: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc noqueue state UP
group default qlen 1000
    link/ether 58:96:1d:16:f7:10 brd ff:ff:ff:ff:ff:ff
    altname wlp0s20f3
    inet 192.168.0.104/24 brd 192.168.0.255 scope global noprefixroute wlo1
       valid_lft forever preferred_lft forever
    inet6 fe80::16cd:5e36:4dae:41d3/64 scope link noprefixroute
       valid_lft forever preferred_lft forever
4: ham0: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1404 qdisc fq_codel state
UNKNOWN group default qlen 1000
    link/ether 7a:79:19:21:68:7d brd ff:ff:ff:ff:ff:ff
    inet 25.33.104.125/8 brd 25.255.255.255 scope global ham0
       valid_lft forever preferred_lft forever
    inet6 2620:9b::1921:687d/96 scope global
       valid_lft forever preferred_lft forever
    inet6 fe80::7879:19ff:fe21:687d/64 scope link
       valid_lft forever preferred_lft forever
```

```
5: br-0ca37458415e: <NO-CARRIER,BROADCAST,MULTICAST,UP> mtu 1500 qdisc
noqueue state DOWN group default
    link/ether 02:42:ba:9a:ec:f9 brd ff:ff:ff:ff:ff:ff
    inet 172.19.0.1/16 brd 172.19.255.255 scope global br-0ca37458415e
       valid_lft forever preferred_lft forever
6: docker0: <NO-CARRIER,BROADCAST,MULTICAST,UP> mtu 1500 qdisc noqueue
state DOWN group default
    link/ether 02:42:8a:55:16:38 brd ff:ff:ff:ff:ff:ff
    inet 172.17.0.1/16 brd 172.17.255.255 scope global docker0
       valid_lft forever preferred_lft forever
7: br-57fda32ea20b: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc
noqueue state UP group default
    link/ether 02:42:92:aa:48:21 brd ff:ff:ff:ff:ff:ff
    inet 172.20.0.1/16 brd 172.20.255.255 scope global br-57fda32ea20b
       valid_lft forever preferred_lft forever
    inet6 fe80::42:92ff:feaa:4821/64 scope link
       valid_lft forever preferred_lft forever
8: br-f9c36026e0cf: <NO-CARRIER,BROADCAST,MULTICAST,UP> mtu 1500 qdisc
noqueue state DOWN group default
    link/ether 02:42:34:92:ca:79 brd ff:ff:ff:ff:ff:ff
    inet 172.18.0.1/16 brd 172.18.255.255 scope global br-f9c36026e0cf
       valid_lft forever preferred_lft forever
10: veth904c2bb@if9: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc
noqueue master br-57fda32ea20b state UP group default
    link/ether a2:fa:0d:1c:d5:df brd ff:ff:ff:ff:ff:ff link-netnsid 1
    inet6 fe80::a0fa:dff:fe1c:d5df/64 scope link
       valid_lft forever preferred_lft forever
12: veth324457a@if11: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc
noqueue master br-57fda32ea20b state UP group default
    link/ether 9e:ce:67:61:fa:84 brd ff:ff:ff:ff:ff:ff link-netnsid 0
    inet6 fe80::9cce:67ff:fe61:fa84/64 scope link
       valid_lft forever preferred_lft forever
13: neko-tun: <POINTOPOINT,MULTICAST,NOARP,UP,LOWER_UP> mtu 9000 qdisc
fq_codel state UNKNOWN group default qlen 500
    link/none
    inet 172.19.0.1/28 brd 172.19.0.15 scope global neko-tun
       valid_lft forever preferred_lft forever
    inet6 fe80::4e7:961b:5fc5:af40/64 scope link stable-privacy
       valid_lft forever preferred_lft forever
15: tun0: <POINTOPOINT,MULTICAST,NOARP,UP,LOWER_UP> mtu 1420 qdisc fq_codel
state UNKNOWN group default qlen 500
    link/none
    inet 192.168.204.31/24 scope global tun0
       valid_lft forever preferred_lft forever
    inet6 fe80::b9dd:cf0f:95e1:29df/64 scope link stable-privacy
       valid_lft forever preferred_lft forever
```

Моя основная сеть это `wlo1`, локальная wi-fi сеть которую я использую для выхода в сеть. Можно увидеть, что роутер выдал моему ноутбуку IPv4 адрес `192.168.0.104` и IPv6 адрес `fe80::16cd:5e36:4dae:41d3`.

Также у меня есть сети которые созданы программами `docker` и `nekoray`. Первая - это технология виртуализации. Вторая - это впн технология основанная на vless.

ip r

Данная команда отображает таблицу маршрутизации.

```
➜  ~ ip r
default via 192.168.0.1 dev wlo1 proto dhcp metric 600
25.0.0.0/8 dev ham0 proto kernel scope link src 25.33.104.125
169.254.0.0/16 dev wlo1 scope link metric 1000
172.17.0.0/16 dev docker0 proto kernel scope link src 172.17.0.1 linkdown
172.18.0.0/16 dev br-f9c36026e0cf proto kernel scope link src 172.18.0.1
linkdown
172.19.0.0/28 dev neko-tun proto kernel scope link src 172.19.0.1
172.19.0.0/16 dev br-0ca37458415e proto kernel scope link src 172.19.0.1
linkdown
172.20.0.0/16 dev br-57fda32ea20b proto kernel scope link src 172.20.0.1
192.168.0.0/24 dev wlo1 proto kernel scope link src 192.168.0.104 metric
600
192.168.1.1 via 192.168.204.1 dev tun0 metric 101
192.168.18.2 via 192.168.204.1 dev tun0 metric 101
192.168.200.0/24 via 192.168.204.1 dev tun0 metric 101
192.168.204.0/24 dev tun0 proto kernel scope link src 192.168.204.31
195.96.77.123 via 192.168.0.1 dev wlo1
```

netstat (ss)

Команда netstat считается устаревшей и не используется в современных сетях, однако ее аналог в виде `ss` отлично справляется с его обязанностями. С помощью нее например можно узнать каким приложением занят тот или иной порт. Открытые подключения сейчас есть. Какие порты в режиме прослушивания и тд...

`ss -4 state listening` - данная команда выводит какие порты сейчас находятся в режиме прослушивания tcp.

```
Netid         Recv-Q         Send-Q                 Local Address:Port
Peer Address:Port         Process
???           0              0                            0.0.0.0:icmp
0.0.0.0:*
???           0              0                            0.0.0.0:ipproto-
255                     0.0.0.0:*
tcp           0              4096                      127.0.0.1:42285
0.0.0.0:*
tcp           0              4096                      127.0.0.1:33835
0.0.0.0:*
tcp           0              4096                      127.0.0.1:42671
0.0.0.0:*
tcp           0              128                       127.0.0.1:ipp
```

```
0.0.0.0:*
tcp           0              4096                         0.0.0.0:50500
0.0.0.0:*
tcp           0              4096                     127.0.0.1:43359
0.0.0.0:*
tcp           0              4096                     127.0.0.1:2080
0.0.0.0:*
tcp           0              128                        0.0.0.0:ssh
0.0.0.0:*
tcp           0              4096                     127.0.0.1:38231
0.0.0.0:*
tcp           0              244
127.0.0.1:postgresql                      0.0.0.0:*
tcp           0              4096                 127.0.0.53%lo:domain
0.0.0.0:*
tcp           0              10                         0.0.0.0:7070
0.0.0.0:*
tcp           0              4096                       0.0.0.0:54320
0.0.0.0:*
tcp           0              511                      127.0.0.1:6463
0.0.0.0:*
```

2. Посмотрите файлы /etc/services, /etc/protocols, расскажите на каких портах работают основные сервисы (ssh, ftp, http, smtp и др.)

Файл services содержит в себе список протоколов и портов. Файл protocols содержит список протоколов и их номера.

/etc/services

```
# Network services, Internet style
#
# Updated from https://www.iana.org/assignments/service-names-port-
numbers/service-names-port-numbers.xhtml .
#
# New ports will be added on request if they have been officially assigned
# by IANA and used in the real-world or are needed by a debian package.
# If you need a huge list of used numbers please install the nmap package.

tcpmux      1/tcp               # TCP port service multiplexer
echo        7/tcp
echo        7/udp
discard     9/tcp       sink null
discard     9/udp       sink null
systat      11/tcp      users
daytime     13/tcp
daytime     13/udp
netstat     15/tcp
qotd        17/tcp      quote
chargen     19/tcp      ttytst source
chargen     19/udp      ttytst source
```

```
ftp-data    20/tcp
ftp     21/tcp
fsp     21/udp      fspd
ssh     22/tcp              # SSH Remote Login Protocol
telnet      23/tcp
smtp        25/tcp      mail
time        37/tcp      timserver
time        37/udp      timserver
whois       43/tcp      nicname
tacacs      49/tcp              # Login Host Protocol (TACACS)
tacacs      49/udp
domain      53/tcp              # Domain Name Server
domain      53/udp
bootps      67/udp
bootpc      68/udp
tftp        69/udp
gopher      70/tcp              # Internet Gopher
finger      79/tcp
http        80/tcp      www     # WorldWideWeb HTTP
kerberos    88/tcp      kerberos5 krb5 kerberos-sec # Kerberos v5
kerberos    88/udp      kerberos5 krb5 kerberos-sec # Kerberos v5
iso-tsap    102/tcp     tsap        # part of ISODE
acr-nema    104/tcp     dicom       # Digital Imag. & Comm. 300
pop3        110/tcp     pop-3       # POP version 3
sunrpc      111/tcp     portmapper  # RPC 4.0 portmapper
sunrpc      111/udp     portmapper
auth        113/tcp     authentication tap ident
nntp        119/tcp     readnews untp   # USENET News Transfer Protocol
ntp     123/udp             # Network Time Protocol
epmap       135/tcp     loc-srv     # DCE endpoint resolution
netbios-ns  137/udp             # NETBIOS Name Service
netbios-dgm 138/udp             # NETBIOS Datagram Service
netbios-ssn 139/tcp             # NETBIOS session service
imap2       143/tcp     imap        # Interim Mail Access P 2 and 4
snmp        161/tcp             # Simple Net Mgmt Protocol
snmp        161/udp
snmp-trap   162/tcp     snmptrap    # Traps for SNMP
snmp-trap   162/udp     snmptrap
cmip-man    163/tcp             # ISO mgmt over IP (CMOT)
cmip-man    163/udp
cmip-agent  164/tcp
cmip-agent  164/udp
mailq       174/tcp         # Mailer transport queue for Zmailer
xdmcp       177/udp         # X Display Manager Control Protocol
bgp     179/tcp             # Border Gateway Protocol
smux        199/tcp             # SNMP Unix Multiplexer
qmtp        209/tcp             # Quick Mail Transfer Protocol
z3950       210/tcp     wais        # NISO Z39.50 database
ipx     213/udp             # IPX [RFC1234]
ptp-event   319/udp
ptp-general 320/udp
pawserv     345/tcp             # Perf Analysis Workbench
zserv       346/tcp             # Zebra server
rpc2portmap 369/tcp
```

```
rpc2portmap 369/udp             # Coda portmapper
codaauth2   370/tcp
codaauth2   370/udp             # Coda authentication server
clearcase   371/udp     Clearcase
ldap        389/tcp         # Lightweight Directory Access Protocol
ldap        389/udp
svrloc      427/tcp             # Server Location
svrloc      427/udp
https       443/tcp             # http protocol over TLS/SSL
https       443/udp             # HTTP/3
snpp        444/tcp             # Simple Network Paging Protocol
microsoft-ds    445/tcp             # Microsoft Naked CIFS
kpasswd     464/tcp
kpasswd     464/udp
submissions 465/tcp     ssmtp smtps urd # Submission over TLS [RFC8314]
saft        487/tcp         # Simple Asynchronous File Transfer
isakmp      500/udp             # IPSEC key management
rtsp        554/tcp         # Real Time Stream Control Protocol
rtsp        554/udp
nqs     607/tcp             # Network Queuing system
asf-rmcp    623/udp     # ASF Remote Management and Control Protocol
qmqp        628/tcp
ipp     631/tcp             # Internet Printing Protocol
ldp     646/tcp             # Label Distribution Protocol
ldp     646/udp
#
# UNIX specific services
#
exec        512/tcp
biff        512/udp     comsat
login       513/tcp
who     513/udp     whod
shell       514/tcp     cmd syslog  # no passwords used
syslog      514/udp
printer     515/tcp     spooler     # line printer spooler
talk        517/udp
ntalk       518/udp
route       520/udp     router routed   # RIP
gdomap      538/tcp             # GNUstep distributed objects
gdomap      538/udp
uucp        540/tcp     uucpd       # uucp daemon
klogin      543/tcp             # Kerberized `rlogin' (v5)
kshell      544/tcp     krcmd       # Kerberized `rsh' (v5)
dhcpv6-client   546/udp
dhcpv6-server   547/udp
afpovertcp  548/tcp             # AFP over TCP
nntps       563/tcp     snntp       # NNTP over SSL
submission  587/tcp             # Submission [RFC4409]
ldaps       636/tcp             # LDAP over SSL
ldaps       636/udp
tinc        655/tcp             # tinc control port
tinc        655/udp
silc        706/tcp
kerberos-adm    749/tcp             # Kerberos `kadmin' (v5)
```

```
#
domain-s    853/tcp             # DNS over TLS [RFC7858]
domain-s    853/udp             # DNS over DTLS [RFC8094]
rsync       873/tcp
ftps-data   989/tcp             # FTP over SSL (data)
ftps        990/tcp
telnets     992/tcp             # Telnet over SSL
imaps       993/tcp             # IMAP over SSL
pop3s       995/tcp             # POP-3 over SSL
#
# From ``Assigned Numbers'':
#
#> The Registered Ports are not controlled by the IANA and on most systems
#> can be used by ordinary user processes or programs executed by ordinary
#> users.
#
#> Ports are used in the TCP [45,106] to name the ends of logical
#> connections which carry long term conversations.  For the purpose of
#> providing services to unknown callers, a service contact port is
#> defined.  This list specifies the port used by the server process as its
#> contact port.  While the IANA can not control uses of these ports it
#> does register or list uses of these ports as a convienence to the
#> community.
#
socks       1080/tcp            # socks proxy server
proofd      1093/tcp
rootd       1094/tcp
openvpn     1194/tcp
openvpn     1194/udp
rmiregistry 1099/tcp            # Java RMI Registry
lotusnote   1352/tcp    lotusnotes  # Lotus Note
ms-sql-s    1433/tcp            # Microsoft SQL Server
ms-sql-m    1434/udp            # Microsoft SQL Monitor
ingreslock  1524/tcp
datametrics 1645/tcp    old-radius
datametrics 1645/udp    old-radius
sa-msg-port 1646/tcp    old-radacct
sa-msg-port 1646/udp    old-radacct
kermit      1649/tcp
groupwise   1677/tcp
l2f     1701/udp    l2tp
radius      1812/tcp
radius      1812/udp
radius-acct 1813/tcp    radacct     # Radius Accounting
radius-acct 1813/udp    radacct
cisco-sccp  2000/tcp            # Cisco SCCP
nfs     2049/tcp            # Network File System
nfs     2049/udp            # Network File System
gnunet      2086/tcp
gnunet      2086/udp
rtcm-sc104  2101/tcp            # RTCM SC-104 IANA 1/29/99
rtcm-sc104  2101/udp
gsigatekeeper   2119/tcp
gris        2135/tcp        # Grid Resource Information Server
```

```
cvspserver  2401/tcp            # CVS client/server operations
venus       2430/tcp            # codacon port
venus       2430/udp            # Venus callback/wbc interface
venus-se    2431/tcp            # tcp side effects
venus-se    2431/udp            # udp sftp side effect
codasrv     2432/tcp            # not used
codasrv     2432/udp            # server port
codasrv-se  2433/tcp            # tcp side effects
codasrv-se  2433/udp            # udp sftp side effect
mon     2583/tcp            # MON traps
mon     2583/udp
dict        2628/tcp            # Dictionary server
f5-globalsite   2792/tcp
gsiftp      2811/tcp
gpsd        2947/tcp
gds-db      3050/tcp    gds_db      # InterBase server
icpv2       3130/udp    icp     # Internet Cache Protocol
isns        3205/tcp            # iSNS Server Port
isns        3205/udp            # iSNS Server Port
iscsi-target    3260/tcp
mysql       3306/tcp
ms-wbt-server   3389/tcp
nut     3493/tcp            # Network UPS Tools
nut     3493/udp
distcc      3632/tcp            # distributed compiler
daap        3689/tcp            # Digital Audio Access Protocol
svn     3690/tcp    subversion  # Subversion protocol
suucp       4031/tcp            # UUCP over SSL
sysrqd      4094/tcp            # sysrq daemon
sieve       4190/tcp            # ManageSieve Protocol
epmd        4369/tcp            # Erlang Port Mapper Daemon
remctl      4373/tcp        # Remote Authenticated Command Service
f5-iquery   4353/tcp            # F5 iQuery
ntske       4460/tcp    # Network Time Security Key Establishment
ipsec-nat-t 4500/udp            # IPsec NAT-Traversal [RFC3947]
iax     4569/udp            # Inter-Asterisk eXchange
mtn     4691/tcp            # monotone Netsync Protocol
radmin-port 4899/tcp            # RAdmin Port
sip     5060/tcp            # Session Initiation Protocol
sip     5060/udp
sip-tls     5061/tcp
sip-tls     5061/udp
xmpp-client 5222/tcp    jabber-client   # Jabber Client Connection
xmpp-server 5269/tcp    jabber-server   # Jabber Server Connection
cfengine    5308/tcp
mdns        5353/udp            # Multicast DNS
postgresql  5432/tcp    postgres    # PostgreSQL Database
freeciv     5556/tcp    rptp        # Freeciv gameplay
amqps       5671/tcp            # AMQP protocol over TLS/SSL
amqp        5672/tcp
amqp        5672/sctp
x11     6000/tcp    x11-0       # X Window System
x11-1       6001/tcp
x11-2       6002/tcp
```

```
x11-3       6003/tcp
x11-4       6004/tcp
x11-5       6005/tcp
x11-6       6006/tcp
x11-7       6007/tcp
gnutella-svc    6346/tcp            # gnutella
gnutella-svc    6346/udp
gnutella-rtr    6347/tcp            # gnutella
gnutella-rtr    6347/udp
redis       6379/tcp
sge-qmaster 6444/tcp    sge_qmaster # Grid Engine Qmaster Service
sge-execd   6445/tcp    sge_execd   # Grid Engine Execution Service
mysql-proxy 6446/tcp            # MySQL Proxy
babel       6696/udp            # Babel Routing Protocol
ircs-u      6697/tcp        # Internet Relay Chat via TLS/SSL
bbs     7000/tcp
afs3-fileserver 7000/udp
afs3-callback   7001/udp            # callbacks to cache managers
afs3-prserver   7002/udp            # users & groups database
afs3-vlserver   7003/udp            # volume location database
afs3-kaserver   7004/udp            # AFS/Kerberos authentication
afs3-volser 7005/udp            # volume managment server
afs3-bos    7007/udp            # basic overseer process
afs3-update 7008/udp            # server-to-server updater
afs3-rmtsys 7009/udp            # remote cache manager service
font-service    7100/tcp    xfs     # X Font Service
http-alt    8080/tcp    webcache    # WWW caching service
puppet      8140/tcp            # The Puppet master service
bacula-dir  9101/tcp            # Bacula Director
bacula-fd   9102/tcp            # Bacula File Daemon
bacula-sd   9103/tcp            # Bacula Storage Daemon
xmms2       9667/tcp    # Cross-platform Music Multiplexing System
nbd     10809/tcp           # Linux Network Block Device
zabbix-agent    10050/tcp           # Zabbix Agent
zabbix-trapper  10051/tcp           # Zabbix Trapper
amanda      10080/tcp           # amanda backup services
dicom       11112/tcp
hkp     11371/tcp           # OpenPGP HTTP Keyserver
db-lsp      17500/tcp           # Dropbox LanSync Protocol
dcap        22125/tcp           # dCache Access Protocol
gsidcap     22128/tcp           # GSI dCache Access Protocol
wnn6        22273/tcp           # wnn6

#
# Datagram Delivery Protocol services
#
rtmp        1/ddp           # Routing Table Maintenance Protocol
nbp     2/ddp           # Name Binding Protocol
echo        4/ddp           # AppleTalk Echo Protocol
zip     6/ddp           # Zone Information Protocol

#========================================================================
# The remaining port numbers are not as allocated by IANA.
#========================================================================
```

```
# Kerberos (Project Athena/MIT) services
kerberos4   750/udp     kerberos-iv kdc # Kerberos (server)
kerberos4   750/tcp     kerberos-iv kdc
kerberos-master 751/udp     kerberos_master # Kerberos authentication
kerberos-master 751/tcp
passwd-server   752/udp     passwd_server   # Kerberos passwd server
krb-prop    754/tcp     krb_prop krb5_prop hprop # Kerberos slave
propagation
zephyr-srv  2102/udp            # Zephyr server
zephyr-clt  2103/udp            # Zephyr serv-hm connection
zephyr-hm   2104/udp            # Zephyr hostmanager
iprop       2121/tcp            # incremental propagation
supfilesrv  871/tcp         # Software Upgrade Protocol server
supfiledbg  1127/tcp        # Software Upgrade Protocol debugging

#
# Services added for the Debian GNU/Linux distribution
#
poppassd    106/tcp             # Eudora
moira-db    775/tcp     moira_db    # Moira database
moira-update    777/tcp     moira_update    # Moira update protocol
moira-ureg  779/udp     moira_ureg  # Moira user registration
spamd       783/tcp             # spamassassin daemon
skkserv     1178/tcp            # skk jisho server port
predict     1210/udp            # predict -- satellite tracking
rmtcfg      1236/tcp            # Gracilis Packeten remote config server
xtel        1313/tcp            # french minitel
xtelw       1314/tcp            # french minitel
zebrasrv    2600/tcp            # zebra service
zebra       2601/tcp            # zebra vty
ripd        2602/tcp            # ripd vty (zebra)
ripngd      2603/tcp            # ripngd vty (zebra)
ospfd       2604/tcp            # ospfd vty (zebra)
bgpd        2605/tcp            # bgpd vty (zebra)
ospf6d      2606/tcp            # ospf6d vty (zebra)
ospfapi     2607/tcp            # OSPF-API
isisd       2608/tcp            # ISISd vty (zebra)
fax     4557/tcp            # FAX transmission service (old)
hylafax     4559/tcp            # HylaFAX client-server protocol (new)
munin       4949/tcp    lrrd        # Munin
rplay       5555/udp            # RPlay audio service
nrpe        5666/tcp            # Nagios Remote Plugin Executor
nsca        5667/tcp            # Nagios Agent - NSCA
canna       5680/tcp            # cannaserver
syslog-tls  6514/tcp            # Syslog over TLS [RFC5425]
sane-port   6566/tcp    sane saned  # SANE network scanner daemon
ircd        6667/tcp            # Internet Relay Chat
zope-ftp    8021/tcp            # zope management by ftp
tproxy      8081/tcp            # Transparent Proxy
omniorb     8088/tcp            # OmniORB
clc-build-daemon 8990/tcp           # Common lisp build daemon
xinetd      9098/tcp
git     9418/tcp            # Git Version Control System
```

```
zope        9673/tcp            # zope server
webmin      10000/tcp
kamanda     10081/tcp           # amanda backup services (Kerberos)
amandaidx   10082/tcp           # amanda backup services
amidxtape   10083/tcp           # amanda backup services
sgi-cmsd    17001/udp       # Cluster membership services daemon
sgi-crsd    17002/udp
sgi-gcd     17003/udp           # SGI Group membership daemon
sgi-cad     17004/tcp           # Cluster Admin daemon
binkp       24554/tcp           # binkp fidonet protocol
asp     27374/tcp           # Address Search Protocol
asp     27374/udp
csync2      30865/tcp           # cluster synchronization tool
dircproxy   57000/tcp           # Detachable IRC Proxy
tfido       60177/tcp           # fidonet EMSI over telnet
fido        60179/tcp           # fidonet EMSI over TCP

# Local services
```

/etc/protocols

```
# Internet (IP) protocols
#
# Updated from http://www.iana.org/assignments/protocol-numbers and other
# sources.
# New protocols will be added on request if they have been officially
# assigned by IANA and are not historical.
# If you need a huge list of used numbers please install the nmap package.

ip  0   IP      # internet protocol, pseudo protocol number
hopopt  0   HOPOPT      # IPv6 Hop-by-Hop Option [RFC1883]
icmp    1   ICMP        # internet control message protocol
igmp    2   IGMP        # Internet Group Management
ggp 3   GGP     # gateway-gateway protocol
ipencap 4   IP-ENCAP    # IP encapsulated in IP (officially ``IP'')
st  5   ST      # ST datagram mode
tcp 6   TCP     # transmission control protocol
egp 8   EGP     # exterior gateway protocol
igp 9   IGP     # any private interior gateway (Cisco)
pup 12  PUP     # PARC universal packet protocol
udp 17  UDP     # user datagram protocol
hmp 20  HMP     # host monitoring protocol
xns-idp 22  XNS-IDP     # Xerox NS IDP
rdp 27  RDP     # "reliable datagram" protocol
iso-tp4 29  ISO-TP4     # ISO Transport Protocol class 4 [RFC905]
dccp    33  DCCP        # Datagram Congestion Control Prot. [RFC4340]
xtp 36  XTP     # Xpress Transfer Protocol
ddp 37  DDP     # Datagram Delivery Protocol
idpr-cmtp 38    IDPR-CMTP   # IDPR Control Message Transport
ipv6    41  IPv6        # Internet Protocol, version 6
ipv6-route 43   IPv6-Route  # Routing Header for IPv6
```

```
ipv6-frag 44    IPv6-Frag   # Fragment Header for IPv6
idrp    45  IDRP        # Inter-Domain Routing Protocol
rsvp    46  RSVP        # Reservation Protocol
gre 47  GRE     # General Routing Encapsulation
esp 50  IPSEC-ESP   # Encap Security Payload [RFC2406]
ah  51  IPSEC-AH    # Authentication Header [RFC2402]
skip    57  SKIP        # SKIP
ipv6-icmp 58    IPv6-ICMP   # ICMP for IPv6
ipv6-nonxt 59   IPv6-NoNxt  # No Next Header for IPv6
ipv6-opts 60    IPv6-Opts   # Destination Options for IPv6
rspf    73  RSPF CPHB   # Radio Shortest Path First (officially CPHB)
vmtp    81  VMTP        # Versatile Message Transport
eigrp   88  EIGRP       # Enhanced Interior Routing Protocol (Cisco)
ospf    89  OSPFIGP     # Open Shortest Path First IGP
ax.25   93  AX.25       # AX.25 frames
ipip    94  IPIP        # IP-within-IP Encapsulation Protocol
etherip 97  ETHERIP     # Ethernet-within-IP Encapsulation [RFC3378]
encap   98  ENCAP       # Yet Another IP encapsulation [RFC1241]
#   99          # any private encryption scheme
pim 103 PIM     # Protocol Independent Multicast
ipcomp  108 IPCOMP      # IP Payload Compression Protocol
vrrp    112 VRRP        # Virtual Router Redundancy Protocol [RFC5798]
l2tp    115 L2TP        # Layer Two Tunneling Protocol [RFC2661]
isis    124 ISIS        # IS-IS over IPv4
sctp    132 SCTP        # Stream Control Transmission Protocol
fc  133 FC      # Fibre Channel
mobility-header 135 Mobility-Header # Mobility Support for IPv6 [RFC3775]
udplite 136 UDPLite     # UDP-Lite [RFC3828]
mpls-in-ip 137  MPLS-in-IP  # MPLS-in-IP [RFC4023]
manet   138         # MANET Protocols [RFC5498]
hip 139 HIP     # Host Identity Protocol
shim6   140 Shim6       # Shim6 Protocol [RFC5533]
wesp    141 WESP        # Wrapped Encapsulating Security Payload
rohc    142 ROHC        # Robust Header Compression
```

3. Настройте подключение по ssh к какому-либо серверу (в крайнем случае к localhost) с использованием ключей шифрования.

Для начала установим необходимые утилиты для поднятия ssh сервера. Мой ноутбук работает на операционной системе Ubuntu.

```
sudo apt-get install openssh-server
```

Далее запустим ssh сервис.

```
sudo systemctl start ssh
```

При необходимости можно отредактировать конфигурационный файл.

```
sudo nano /etc/ssh/sshd_config
```

Затем перезагрузим сервис.

```
sudo systemctl reload ssh
```

Чтобы подключиться - напишем

```
➜  ~ ssh hryapusek@localhost
hryapusek@localhost's password:
Welcome to Ubuntu 22.04.5 LTS (GNU/Linux 6.8.0-52-generic x86_64)

 * Documentation:  https://help.ubuntu.com
 * Management:     https://landscape.canonical.com
 * Support:        https://ubuntu.com/pro

Expanded Security Maintenance for Applications is not enabled.

41 updates can be applied immediately.
1 of these updates is a standard security update.
To see these additional updates run: apt list --upgradable

46 additional security updates can be applied with ESM Apps.
Learn more about enabling ESM Apps service at https://ubuntu.com/esm


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

Ubuntu comes with ABSOLUTELY NO WARRANTY, to the extent permitted by
applicable law.

Last login: Sat Oct  5 00:02:35 2024 from 25.56.214.183
xset:  unable to open display ""
Cannot open display "default display"
➜  ~
```

Чтобы отключится нажмем комбинацию CTRL+D

4. Настройте различные параметры подключения по SSH, расскажите, когда их нужно использовать

Основные

Port

Port - это параметр, который указывает, какой порт SSH сервер будет слушать для входящих соединений. По умолчанию он равен 22.

AddressFamily

AddressFamily - это параметр, который указывает, какой тип адресов SSH сервер будет принимать. По умолчанию он равен `any`.

ListenAddress

ListenAddress - это параметр, который указывает, какой адрес SSH сервер будет слушать для входящих соединений.

Авторизация

PubkeyAuthentication

PubkeyAuthentication - это параметр, который указывает, может ли SSH сервер использовать OpenSSH-ключи для аутентификации.

LoginGraceTime

LoginGraceTime - это параметр, который указывает, сколько времени пользователь имеет на аутентификацию. Если аутентификация не будет успешной, то соединение будет закрыто.

PermitRootLogin

PermitRootLogin - это параметр, который указывает, может ли пользователь root логиниться на сервер. Если он равен `prohibit-password`, то root может логиниться только с помощью ключей.

StrictModes

StrictModes - это параметр, который указывает, должны ли права доступа к файлам быть строго проверены. Если он равен `yes`, то права доступа к файлам будут строго проверены.

MaxAuthTries

MaxAuthTries - это параметр, который указывает, сколько раз пользователь может попытаться аутентифицироваться. Если количество попыток превысит это значение, то соединение будет закрыто.

MaxSessions

MaxSessions - это параметр, который указывает, сколько сеансов может быть открыто одновременно.

Algorithms

Key exchange algorithms

Key exchange algorithms - это алгоритмы обмена ключами, которые используются для обмена информацией между клиентом и сервером. Они обеспечивают безопасность соединения, так как ключи шифрования не передаются напрямую.

Самыми популярными алгоритмами обмена ключами являются:

- `diffie-hellman-group-exchange-sha256`
- `diffie-hellman-group14-sha256`
- `ecdh-sha2-nistp256`

Ciphers

Ciphers - это алгоритмы шифрования, которые используются для шифрования данных. Они обеспечивают безопасность соединения, так как данные шифруются перед передачей.

Самыми популярными алгоритмами шифрования являются:

- `aes256-ctr`
- `aes128-ctr`
- `3des-ctr`

MACs

MACs - это алгоритмы аутентификации, которые используются для аутентификации данных. Они обеспечивают безопасность соединения, так как данные аутентифицируются перед передачей.

Самыми популярными алгоритмами аутентификации являются:

- `hmac-sha2-512`
- `hmac-sha2-256`
- `hmac-sha1`

HostKeyAlgorithms

HostKeyAlgorithms - это алгоритмы, которые используются для верификации ключей хоста. Они обеспечивают безопасность соединения, так как ключи хоста аутентифицируются перед передачей.

Самыми популярными алгоритмами аутентификации ключей хоста являются:

- `ssh-rsa`
- `ssh-dss`
- `ecdsa-sha2-nistp256`

KexAlgorithms

KexAlgorithms - это алгоритмы, которые используются для обмена ключами. Они обеспечивают безопасность соединения, так как ключи шифрования не передаются напрямую.

Самыми популярными алгоритмами обмена ключами являются:

- `curve25519-sha256`

- `diffie-hellman-group-exchange-sha256`

Прочее

Ssh в современном программировании встречается повсеместно. На 2 курсе в качестве летней практики я поставил задачу написать Вк бота, который пересылал бы сообщения администрации всем студентам. Для того чтобы бот постоянно работал необходимо было настроить сервер и запустить бота на нём. Все сервера поставляются через ssh, и там было моё первое серьезное знакомство с этим протоколом.

Далее ssh встречался в лабораторных работах по Dev-ops чтобы быстрее управлять виртуальными машинами и подключаться к ним.