

Debugging Network Application

Tomasz Urban

30 stycznia 2024

1 Task 1

1.1 Clone repository

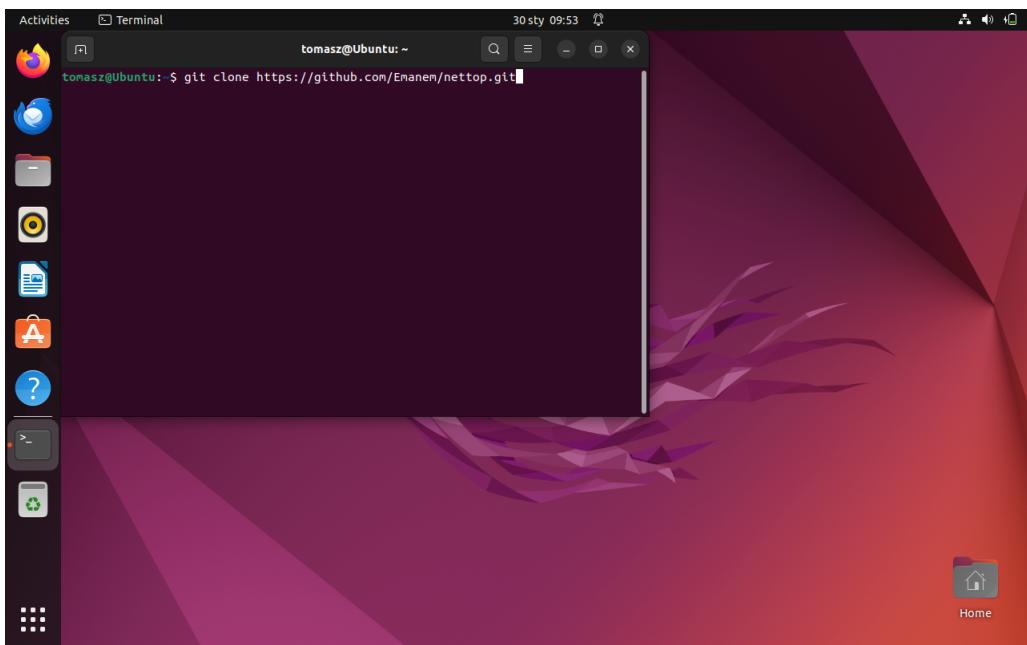
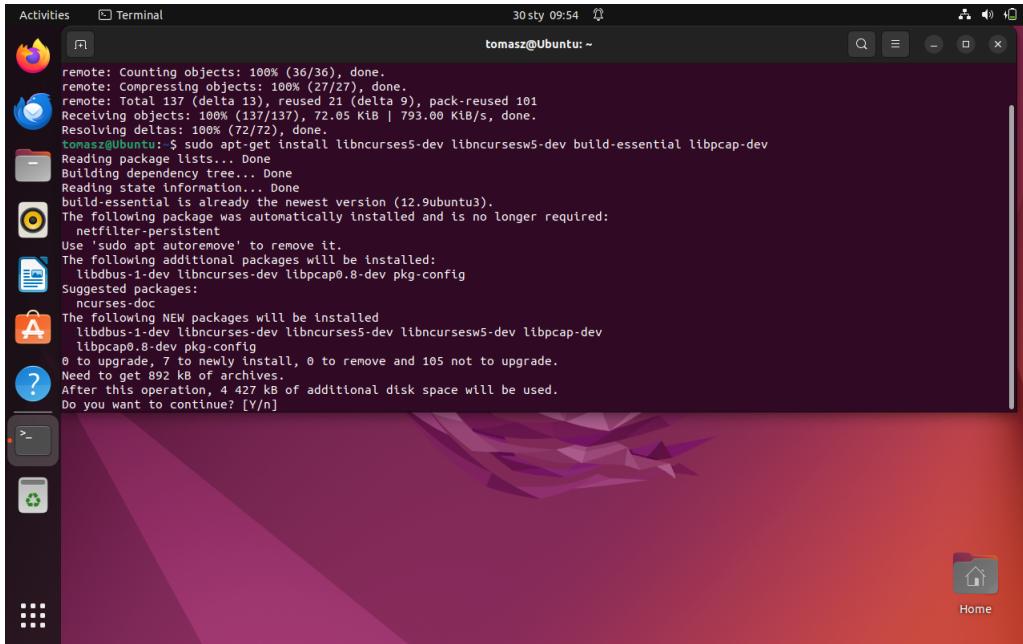


Figure 1: Clone repository with nettop source code

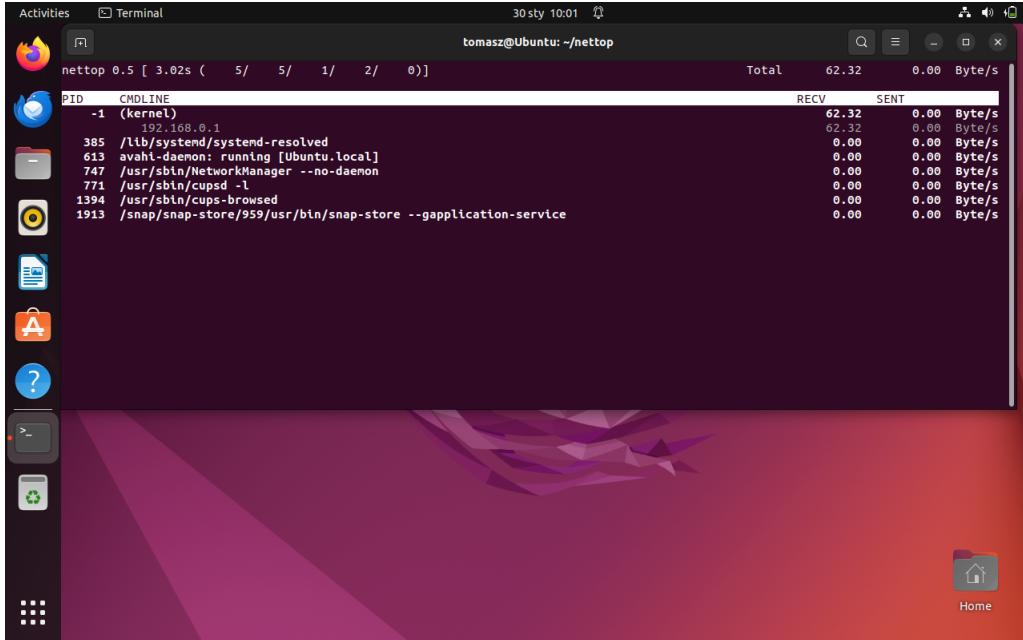
1.2 Install needed dependencies



```
Activities Terminal 30 sty 09:54 tomasz@Ubuntu: ~
remote: Counting objects: 100% (36/36), done.
remote: Compressing objects: 100% (27/27), done.
remote: Total 137 (delta 13), reused 21 (delta 9), pack-reused 101
Receiving objects: 100% (137/137), 72.05 KiB | 793.00 KiB/s, done.
Resolving deltas: 100% (72/72), done.
tomasz@Ubuntu: $ sudo apt-get install libncurses5-dev libncursesw5-dev build-essential libpcap-dev
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
build-essential is already the newest version (12.9ubuntu3).
The following package was automatically installed and is no longer required:
    netfilter-persistent
Use 'sudo apt autoremove' to remove it.
The following additional packages will be installed:
    libdbus-1-dev libncurses-dev libpcap0.8-dev pkg-config
Suggested packages:
    ncurses-doc
The following NEW packages will be installed
    libdbus-1-dev libncurses-dev libncurses5-dev libncursesw5-dev libpcap-dev
    libpcap0.8-dev pkg-config
0 to upgrade, 7 to newly install, 0 to remove and 105 not to upgrade.
Need to get 892 kB of archives.
After this operation, 4 427 kB of additional disk space will be used.
Do you want to continue? [Y/n]
```

Figure 2: Install needed dependencies

1.3 Run nettop



```
Activities Terminal 30 sty 10:01 tomasz@Ubuntu: ~/nettop
nettop 0.5 [ 3.02s ( 5/ 5/ 1/ 2/ 0) ]                                         Total   62.32   0.00 Byte/s
PID  CMDLINE
-1  (kernel)
          192.168.0.1
  385  /lib/systemd/systemd-resolved
  613  avahi-daemon: running [Ubuntu.local]
  747  /usr/sbin/NetworkManager --no-daemon
  771  /usr/sbin/cupsd -l
1394  /usr/sbin/cups-browsed
1913  /snap/snap-store/959/usr/bin/snap-store --gapplication-service
```

Figure 3: Run nettop

2 Task 2

To generate table network interface enp0s8 has been dumped (with global network address). Table with output is in file most_network_traffic.csv.

Output has been generated with use of jttop command. Command is provided on figure 4.

```

Activities Terminal 30 sty 10:41
tomasz@Ubuntu: ~
tomasz@Ubuntu: $ sudo jnettop -i enp0s8 --display text -t 30 --format '$src$, $srcport$, $dst$, $dstport$, $proto$, $totalbytes$' >most_
network_traffic.csv; sed -i 's/,/;/g' most_network_traffic.csv
Could not read/find config file /root/.jnettop: No such file or directory.
Could not get HW address of interface any: No such device
Could not get HW address of interface bluetooth-monitor: No such device
Could not get HW address of interface nflog: No such device
Could not get HW address of interface nfqueue: No such device
Could not get HW address of interface dbus-system: No such device
Could not get HW address of interface dbus-session: No such device
tomasz@Ubuntu: $ cat most_network_traffic.csv
192.168.0.1 59415 255.255.255.255 7437 UDP 2150
192.168.0.195 1 192.168.0.1 1 IP 248
192.168.0.195 137 192.168.0.1 56001 UDP 92
192.168.0.195 137 192.168.0.1 60885 UDP 92
0.0.0.0 0 0.0.0.0 0 ARP 102
192.168.0.195 36902 34.107.243.93 443 TCP 324
192.168.0.195 36523 192.168.0.1 53 UDP 313
192.168.0.195 54654 192.168.0.1 53 UDP 381
192.168.0.195 36812 192.168.0.1 53 UDP 246
192.168.0.195 56751 192.168.0.1 53 UDP 168
192.168.0.195 40517 192.168.0.1 53 UDP 168
192.168.0.195 36921 192.168.0.1 53 UDP 168
192.168.0.195 43329 192.168.0.1 53 UDP 168
192.168.0.195 137 192.168.0.1 50923 UDP 92
192.168.0.195 137 192.168.0.1 46800 UDP 92
192.168.0.195 42563 192.168.0.1 53 UDP 366
192.168.0.195 59909 192.168.0.1 53 UDP 370
tomasz@Ubuntu: $
```

Figure 4: jttop output

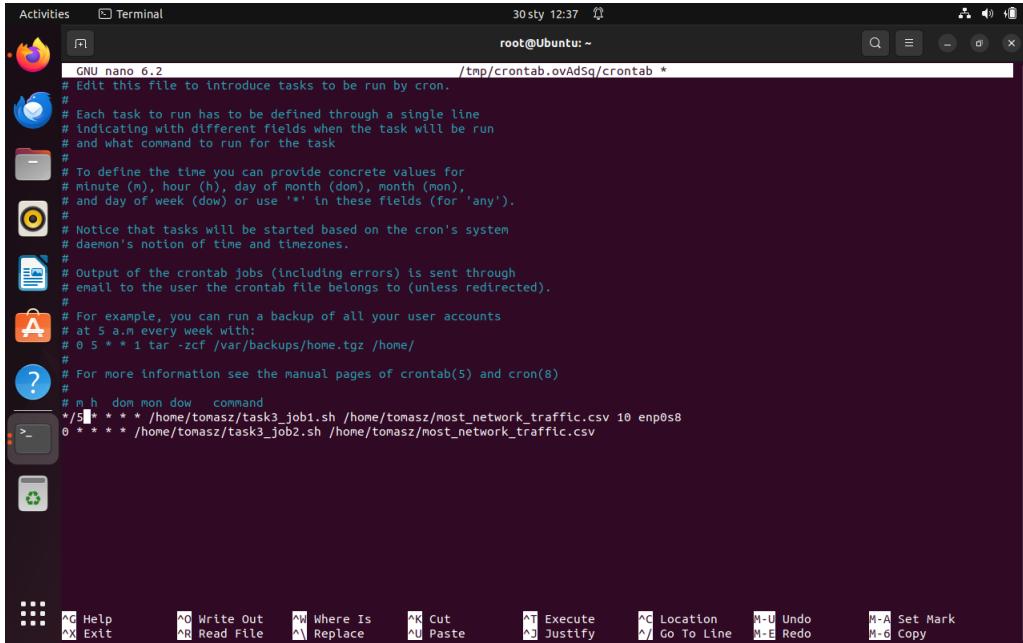
2.1 First cron job

- job must be executed every 5 minutes on every hour,
- job include 10 seconds of gathering network traffic,
- traffic is recorded to file
- number of bytes are captured
- script is provided in file task3_job1.sh

2.2 Second cron job

- at the beginning of hour in the output file of first job, a most networking consuming connection is found
- found value is written to syslog
- at the end file is clear
- script is provided in file task3_job2.sh

2.3 Crontab entries



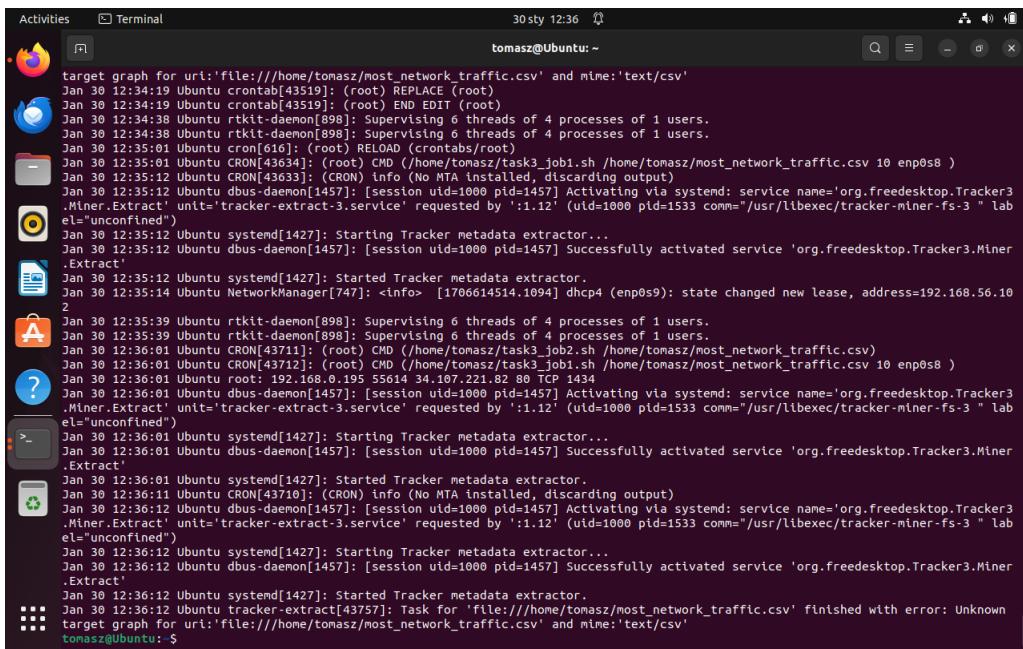
The screenshot shows a terminal window titled "root@Ubuntu: ~". The window displays the contents of a crontab file. The file contains several cron entries, including one that runs a script every week at 5 AM to backup user accounts and another that runs a script every minute to monitor network traffic.

```
GNU nano 6.2
/tmp/crontab.ovAdSq/crontab *
# Edit this file to introduce tasks to be run by cron.
#
# Each task to run has to be defined through a single line
# indicating with different fields when the task will be run
# and what command to run for the task
#
# To define the time you can provide concrete values for
# minute (m), hour (h), day of month (dom), month (mon),
# and day of week (dow) or use '*' in these fields (for 'any').
#
# Notice that tasks will be started based on the cron's system
# daemon's notion of time and timezone.
#
# Output of the crontab jobs (including errors) is sent through
# email to the user the crontab file belongs to (unless redirected).
#
# For example, you can run a backup of all your user accounts
# at 5 a.m. every week with:
# 0 5 * * * tar -cf /var/backups/home.tgz /home/
#
# For more information see the manual pages of crontab(5) and cron(8)
#
# m h dom mon dow command
*/5 * * * * /home/tomasz/task3_job1.sh /home/tomasz/most_network_traffic.csv 10 enp0s8
0 * * * * /home/tomasz/task3_job2.sh /home/tomasz/most_network_traffic.csv
```

Figure 5: Crontab entries

2.4 Syslog

To check the functionality of cron jobs with sudo nano /var/log/syslog content of file has been examined to find out whether the proper line is inside. A new record at 6 PM has been found. Crontab is assumed to work correctly.



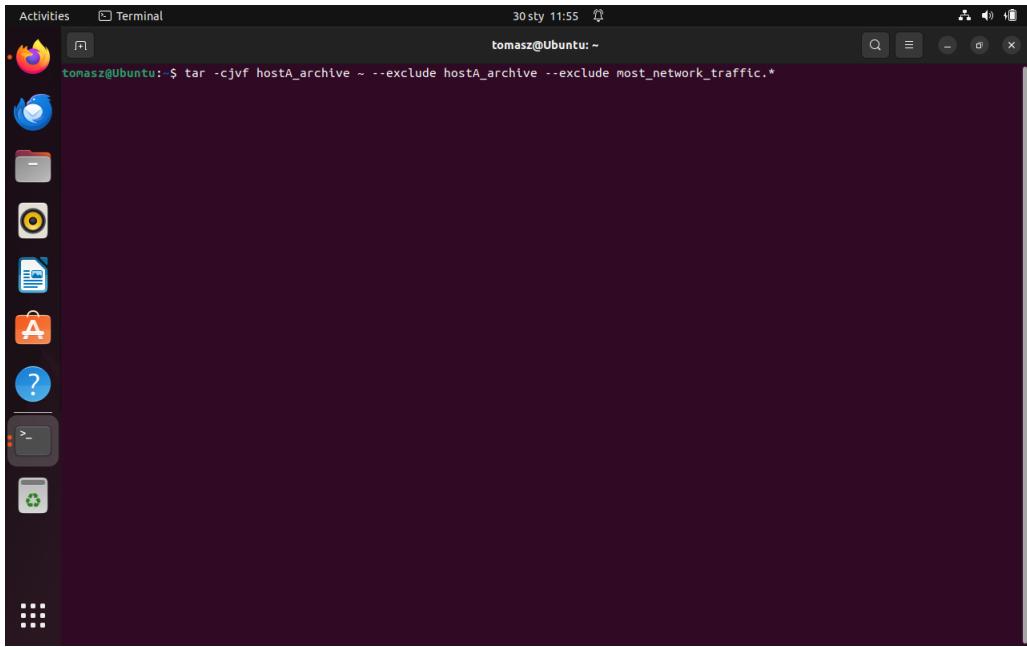
The screenshot shows a terminal window titled "tomasz@Ubuntu: ~". The window displays a log entry from the syslog. The log entry shows a cron job running at 12:35 PM on January 30, 2019, which supervises 6 threads of 4 processes of 1 users. It also shows a message from the systemd service 'org.freedesktop.Tracker3.Miner.Extract' starting a metadata extractor.

```
tomasz@Ubuntu: ~
target graph for url:'file:///home/tomasz/most_network_traffic.csv' and mime:'text/csv'
Jan 30 12:34:19 Ubuntu crontab[43519]: (root) REPLACE (root)
Jan 30 12:34:19 Ubuntu crontab[43519]: (root) END EDIT (root)
Jan 30 12:34:38 Ubuntu rtkit-daemon[398]: Supervising 6 threads of 4 processes of 1 users.
Jan 30 12:34:38 Ubuntu rtkit-daemon[398]: Supervising 6 threads of 4 processes of 1 users.
Jan 30 12:35:01 Ubuntu cron[610]: (root) RELOAD (crontabs/root)
Jan 30 12:35:01 Ubuntu CRON[43634]: (root) CMD (/home/tomasz/task3_job1.sh /home/tomasz/most_network_traffic.csv 10 enp0s8 )
Jan 30 12:35:12 Ubuntu dbus-daemon[1457]: [session uid=1000 pid=1457] Activating via systemd: service name='org.freedesktop.Tracker3.Miner.Extract' unit='tracker-extract-3.service' requested by ':1.12' (uid=1000 pid=1533 comm="/usr/libexec/tracker-miner-fs-3" lab el='unconfined')
Jan 30 12:35:12 Ubuntu systemd[1427]: Starting Tracker metadata extractor...
Jan 30 12:35:12 Ubuntu dbus-daemon[1457]: [session uid=1000 pid=1457] Successfully activated service 'org.freedesktop.Tracker3.Miner.Extract'
Jan 30 12:35:12 Ubuntu systemd[1427]: Started Tracker metadata extractor.
Jan 30 12:35:14 Ubuntu NetworkManager[747]: <info> [1706614514.1094] dhcpc4 (enp0s9): state changed new lease, address=192.168.56.10
2
Jan 30 12:35:39 Ubuntu rtkit-daemon[398]: Supervising 6 threads of 4 processes of 1 users.
Jan 30 12:35:39 Ubuntu rtkit-daemon[398]: Supervising 6 threads of 4 processes of 1 users.
Jan 30 12:36:01 Ubuntu CRON[43711]: (root) CMD (/home/tomasz/task3_job2.sh /home/tomasz/most_network_traffic.csv)
Jan 30 12:36:01 Ubuntu CRON[43712]: (root) CMD (/home/tomasz/task3_job1.sh /home/tomasz/most_network_traffic.csv 10 enp0s8 )
Jan 30 12:36:01 Ubuntu dbus-daemon[1457]: [session uid=1000 pid=1457] Activating via systemd: service name='org.freedesktop.Tracker3.Miner.Extract' unit='tracker-extract-3.service' requested by ':1.12' (uid=1000 pid=1533 comm="/usr/libexec/tracker-miner-fs-3" lab el='unconfined')
Jan 30 12:36:01 Ubuntu systemd[1427]: Starting Tracker metadata extractor...
Jan 30 12:36:01 Ubuntu dbus-daemon[1457]: [session uid=1000 pid=1457] Successfully activated service 'org.freedesktop.Tracker3.Miner.Extract'
Jan 30 12:36:01 Ubuntu systemd[1427]: Started Tracker metadata extractor.
Jan 30 12:36:11 Ubuntu CRON[43710]: (CRON) info (No MTA installed, discarding output)
Jan 30 12:36:12 Ubuntu dbus-daemon[1457]: [session uid=1000 pid=1457] Activating via systemd: service name='org.freedesktop.Tracker3.Miner.Extract' unit='tracker-extract-3.service' requested by ':1.12' (uid=1000 pid=1533 comm="/usr/libexec/tracker-miner-fs-3" lab el='unconfined')
Jan 30 12:36:12 Ubuntu systemd[1427]: Starting Tracker metadata extractor...
Jan 30 12:36:12 Ubuntu dbus-daemon[1457]: [session uid=1000 pid=1457] Successfully activated service 'org.freedesktop.Tracker3.Miner.Extract'
Jan 30 12:36:12 Ubuntu systemd[1427]: Started Tracker metadata extractor.
Jan 30 12:36:12 Ubuntu tracker-extract[43757]: Task for 'file:///home/tomasz/most_network_traffic.csv' finished with error: Unknown target graph for url:'file:///home/tomasz/most_network_traffic.csv' and mime:'text/csv'
```

Figure 6: Syslog record

3 Task 4

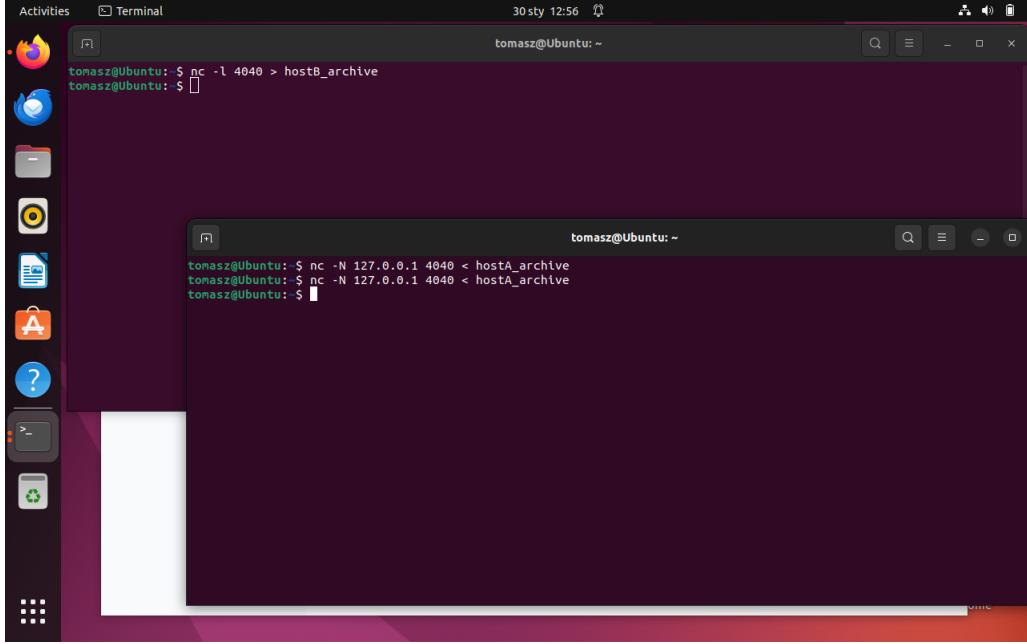
3.1 Create and compress archive of home directory



```
tomasz@Ubuntu:~$ tar -cjvf hostA_archive ~ --exclude hostA_archive --exclude most_network_traffic.*
```

Figure 7: Syslog record

3.2 Send with nc



```
tomasz@Ubuntu:~$ nc -l 4040 > hostB_archive
tomasz@Ubuntu:~$ 
```



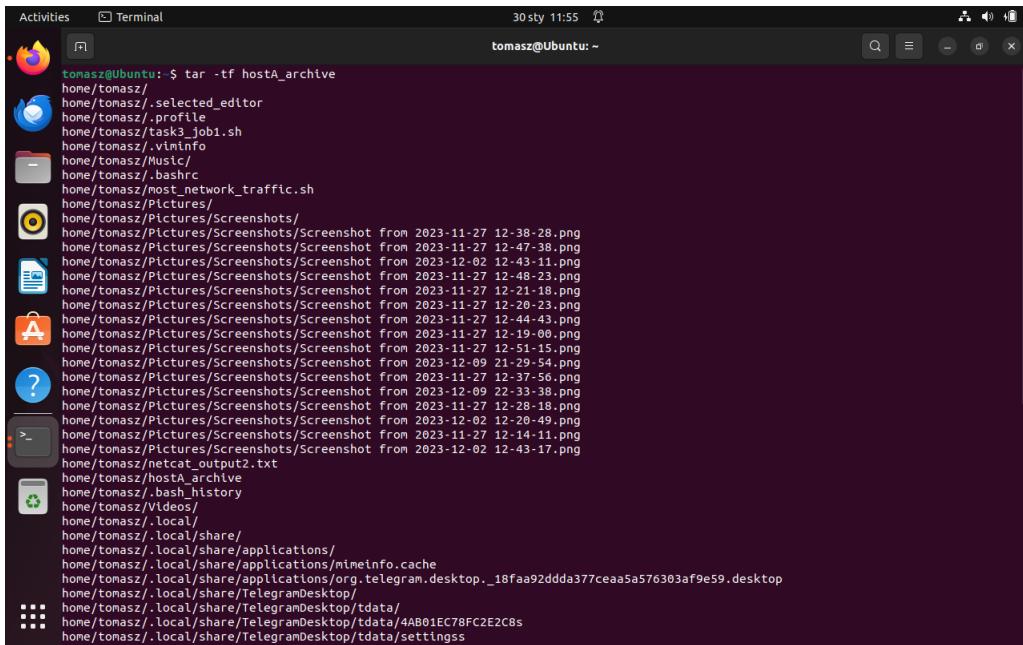
```
tomasz@Ubuntu:~$ nc -N 127.0.0.1 4040 < hostA_archive
tomasz@Ubuntu:~$ nc -N 127.0.0.1 4040 < hostA_archive
tomasz@Ubuntu:~$ 
```

Figure 8: Send home archive via nc

On hostB received archive is saved as hostB_archive.

3.3 Content of archive

Both archives (hostB_archive and hostA_archive) are provided on a disk for evaluation.

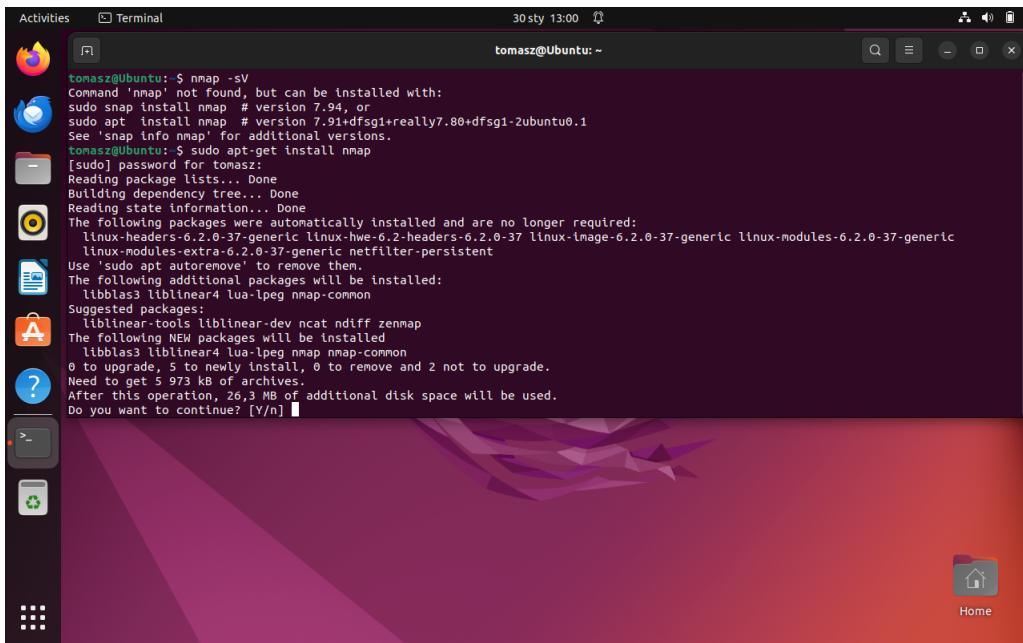


```
tomasz@Ubuntu: $ tar -tf hostA_archive
home/tomasz/
home/tomasz/.selected_editor
home/tomasz/.profile
home/tomasz/task3_job1.sh
home/tomasz/.viminfo
home/tomasz/Music/
home/tomasz/.bashrc
home/tomasz/host-network_traffic.sh
home/tomasz/Pictures/
home/tomasz/Pictures/Screenshots/Screenshot from 2023-11-27 12-38-28.png
home/tomasz/Pictures/Screenshots/Screenshot from 2023-11-27 12-47-38.png
home/tomasz/Pictures/Screenshots/Screenshot from 2023-12-02 12-43-11.png
home/tomasz/Pictures/Screenshots/Screenshot from 2023-11-27 12-48-23.png
home/tomasz/Pictures/Screenshots/Screenshot from 2023-11-27 12-21-18.png
home/tomasz/Pictures/Screenshots/Screenshot from 2023-11-27 12-20-23.png
home/tomasz/Pictures/Screenshots/Screenshot from 2023-11-27 12-44-43.png
home/tomasz/Pictures/Screenshots/Screenshot from 2023-11-27 12-19-00.png
home/tomasz/Pictures/Screenshots/Screenshot from 2023-11-27 12-51-15.png
home/tomasz/Pictures/Screenshots/Screenshot from 2023-12-02 21-29-54.png
home/tomasz/Pictures/Screenshots/Screenshot from 2023-11-27 12-37-56.png
home/tomasz/Pictures/Screenshots/Screenshot from 2023-12-02 22-33-38.png
home/tomasz/Pictures/Screenshots/Screenshot from 2023-11-27 12-28-18.png
home/tomasz/Pictures/Screenshots/Screenshot from 2023-12-02 12-20-49.png
home/tomasz/Pictures/Screenshots/Screenshot from 2023-11-27 12-14-11.png
home/tomasz/Pictures/Screenshots/Screenshot from 2023-12-02 12-43-17.png
home/tomasz/netcat_output2.txt
home/tomasz/hostA_archive
home/tomasz/.bash_history
home/tomasz/Videos/
home/tomasz/.local/
home/tomasz/.local/share/
home/tomasz/.local/share/applications/
home/tomasz/.local/share/applications/mimeinfo.cache
home/tomasz/.local/share/applications/org.telegram.desktop._18faa92ddd377ceaa5a576303af9e59.desktop
home/tomasz/.local/share/TelegramDesktop/
home/tomasz/.local/share/TelegramDesktop/tdata/
home/tomasz/.local/share/TelegramDesktop/tdata/4A801EC78FC2E2C8S
home/tomasz/.local/share/TelegramDesktop/tdata/settingsss
```

Figure 9: Content of archives

4 Task 5

4.1 Install nmap



```
tomasz@Ubuntu: $ nmap -sv
Command 'nmap' not found, but can be installed with:
  sudo snap install nmap # version 7.94, or
  sudo apt install nmap # version 7.91+dfsg1+really7.80+dfsg1-2ubuntu0.1
See 'snap info nmap' for additional versions.
tomasz@Ubuntu: $ sudo apt-get install nmap
[sudo] password for tomasz:
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
The following packages were automatically installed and are no longer required:
  linux-headers-6.2.0-37-generic linux-hwe-6.2.0-headers-6.2.0-37 linux-image-6.2.0-37-generic linux-modules-6.2.0-37-generic
  linux-modules-extra-6.2.0-37-generic netfilter-persistent
Use 'sudo apt autoremove' to remove them.
The following additional packages will be installed:
  libbbfs3 libbblinear4 lua-lpeg nmap-common
Suggested packages:
  libbblinear-tools libbblinear-dev ncat ndiff zenmap
The following NEW packages will be installed:
  libbbfs3 libbblinear4 lua-lpeg nmap nmap-common
0 to upgrade, 5 to newly install, 0 to remove and 2 not to upgrade.
Need to get 5 973 kB of additional disk space.
After this operation, 26,3 MB of additional disk space will be used.
Do you want to continue? [Y/n] ■
```

Figure 10: Search for OS and services

4.2 Nmap Security scanning, search for services an daemons

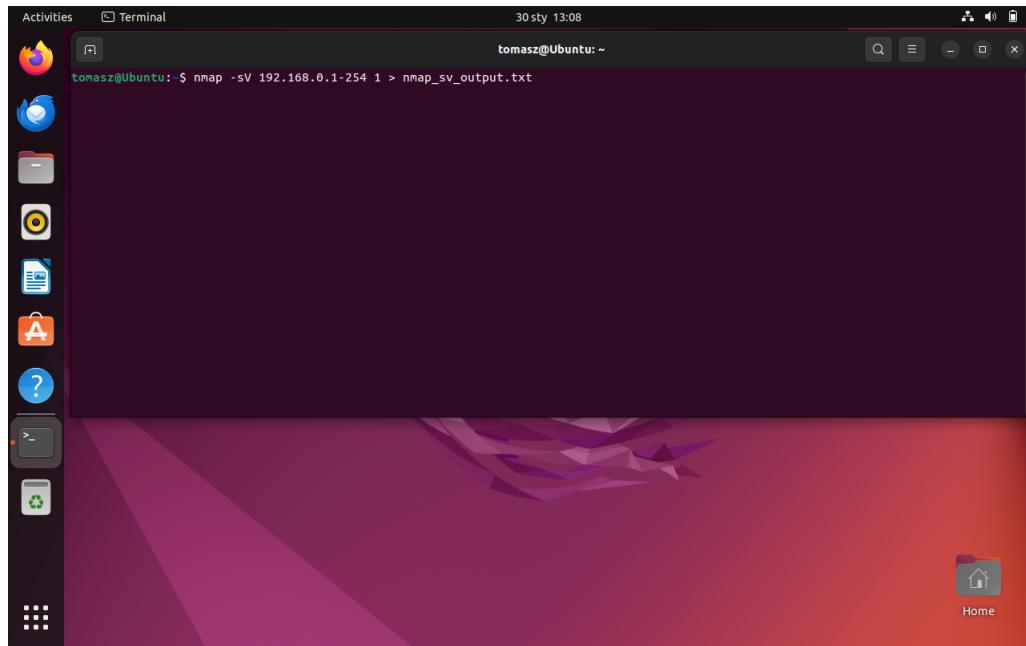


Figure 11: Search for services and daemons

File with search results is provided on the shared disk (nmap_sv_output.txt).

4.2.1 Search results

On gateway (192.168.0.1) there are following services available (all on tcp sockets):

- ssh server on port 22, input is filtered
- dns server on port 53, input is open
- http server on port 80
- netbios-ssn on port 139
- microsoft-ds on port 445
- upnp on port 1900

On host (192.168.0.195) all ports are closed. There are no available services.

4.3 Nmap Security scanning, search for OS and service detection

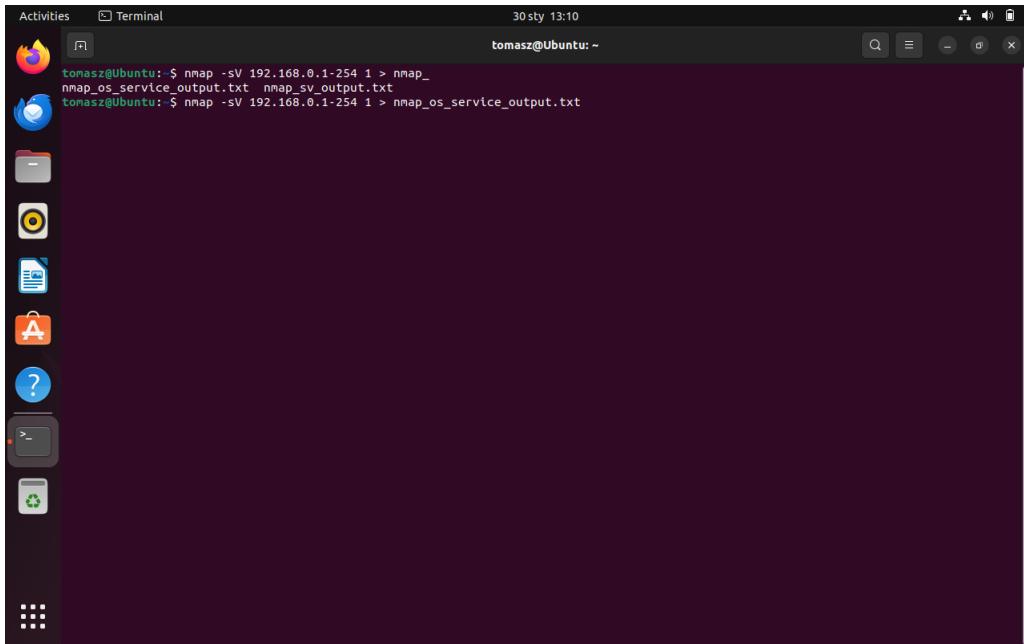
A screenshot of a Linux desktop environment showing a terminal window. The terminal window title is "Terminal" and the status bar shows "tomasz@Ubuntu: ~" and the date/time "30 sty 13:10". The terminal content shows two commands being run: "nmap -sV 192.168.0.1-254 1 > nmap_os_service_output.txt" and "nmap -sV 192.168.0.1-254 1 > nmap_sv_output.txt". The background shows a dark-themed desktop with various icons in the dock.

Figure 12: Search for OS and services

File with search results is provided on the shared disk (nmap_os_service_output.txt).

4.3.1 Search results

On gateway (192.168.0.1) there are following services available (all on tcp sockets):

- ssh server on port 22, input is filtered
- dns server on port 53, input is open
- http server on port 80
- netbios-ssn on port 139
- microsoft-ds on port 445
- upnp on port 1900

On host (192.168.0.195) all ports are closed. There are no available services.

5 Task 6

5.1 NMAP script categories

5.1.1 auth

Deal with authentication credentials on the target system

5.1.2 broadcast

Typically these scripts discover hosts not listed on the command line by broadcasting on local network.

5.1.3 brute

Brute scripts use brute force attack to guess authentication credentials of a remote server.

5.1.4 default

Scripts which are the default set and are run when using the $-sC$ or $-A$. Several factors influence decision about choosing a script:

- speed- default scan must finish quickly
- usefulness- default script must produce useful information
- verbosity
- reliability
- intrusiveness- default scripts are almost always in safe category
- privacy

5.1.5 discovery

Discover more about the network by querying public registries.

5.1.6 dos

Scripts in these categories may cause a denial of service. Sometimes used to test vulnerability to a denial of service method.

5.1.7 exploit

Scripts used to actively exploit some vulnerability.

5.1.8 external

These scripts may send data to third-party database or other network resource.

5.1.9 fuzzer

This category contains scripts which are designed to send server software unexpected or randomized fields in each packet. While this technique can be useful for finding undiscovered bugs and vulnerabilities in software, it is both a slow process and bandwidth intensive.

5.1.10 intrusive

Scripts that cannot be categorized to safe category.

5.1.11 malware

These scripts check, whether the target platform contains malware or backdoors.

5.1.12 safe

Scripts not designed to crash services, use large amounts of network bandwidth or of other resources, exploit security holes.

5.1.13 version

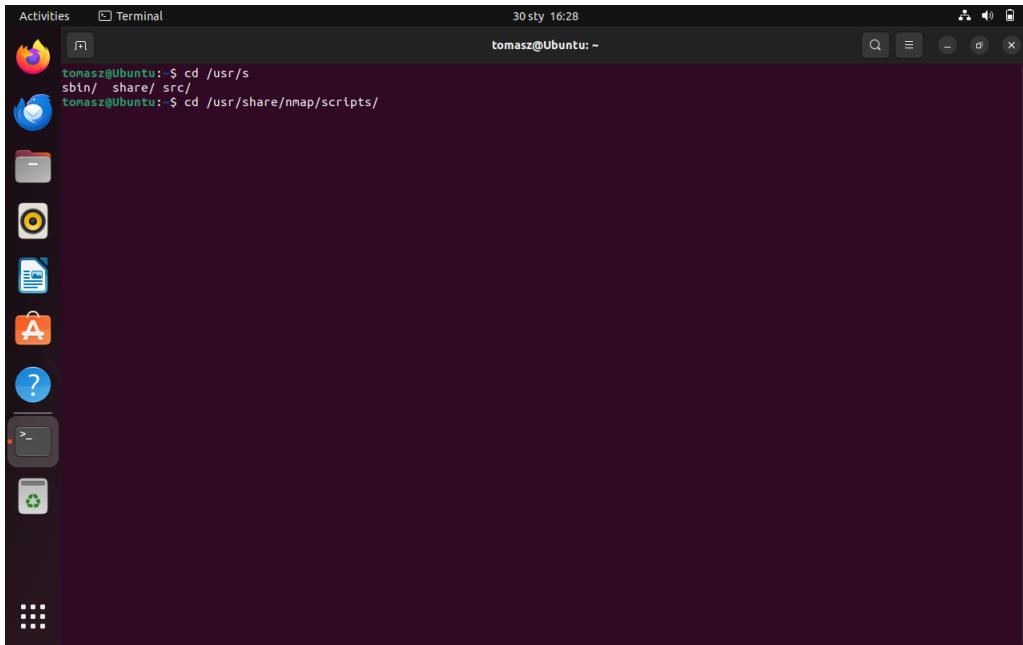
These scripts detect version features. Cannot be selected explicitly.

5.1.14 vuln

Check for special types of vulnerabilities.

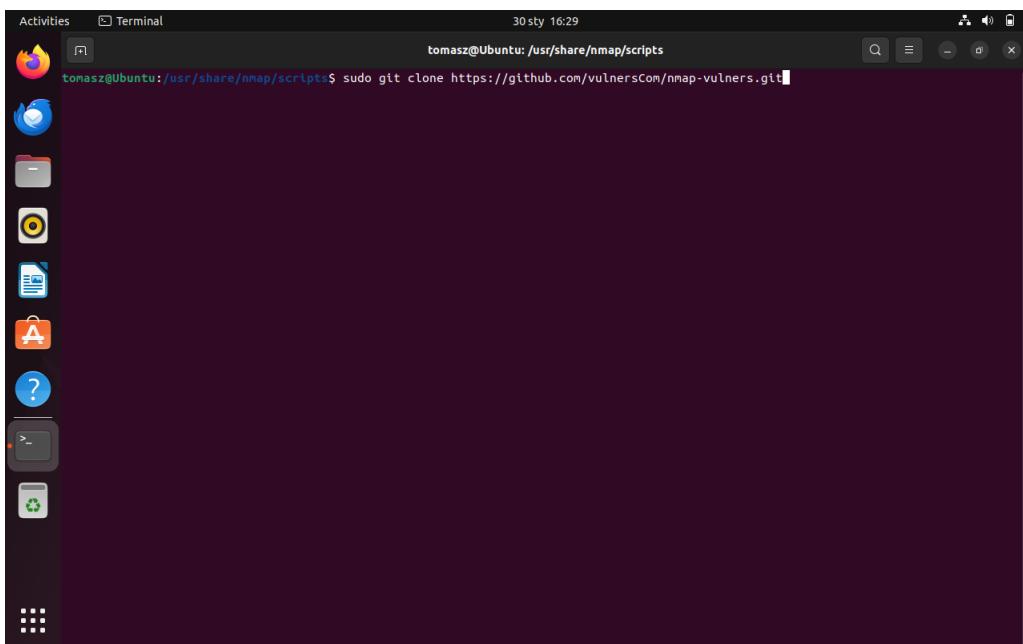
5.2 Perform CVE detection using NMAP

5.2.1 Installing cve scripts



```
Activities Terminal 30 sty 16:28
tomasz@Ubuntu:~$ cd /usr/s
tomasz@Ubuntu:~$ bbin/ share/ src/
tomasz@Ubuntu:~$ cd /usr/share/nmap/scripts/
```

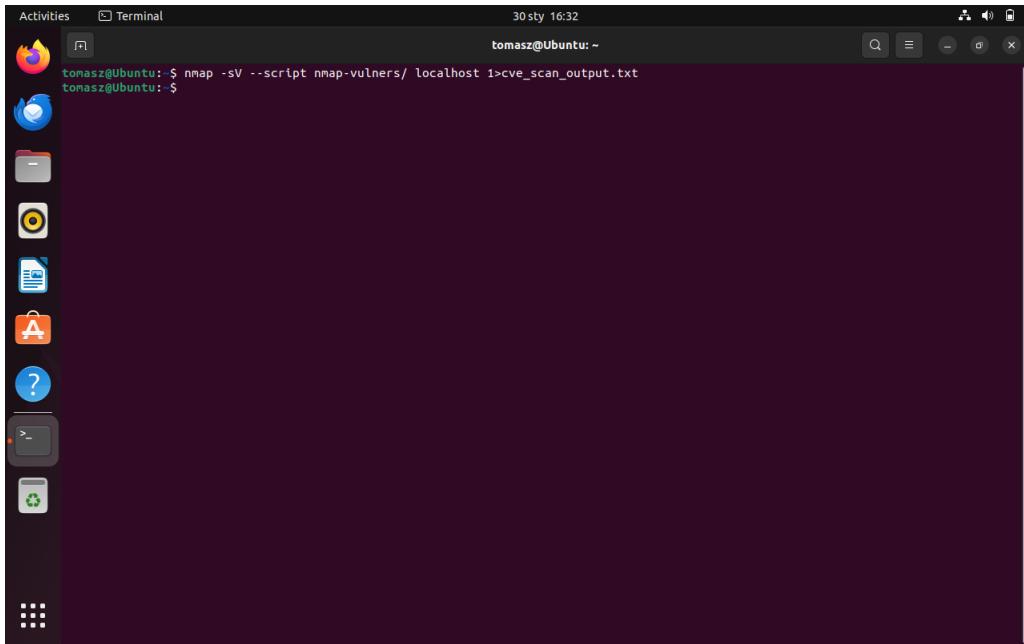
Figure 13: Change directory



```
Activities Terminal 30 sty 16:29
tomasz@Ubuntu:~$ cd /usr/share/nmap/scripts
tomasz@Ubuntu:/usr/share/nmap/scripts$ sudo git clone https://github.com/vulnersCom/nmap-vulners.git
```

Figure 14: Cloning cve script

5.2.2 CVE searching

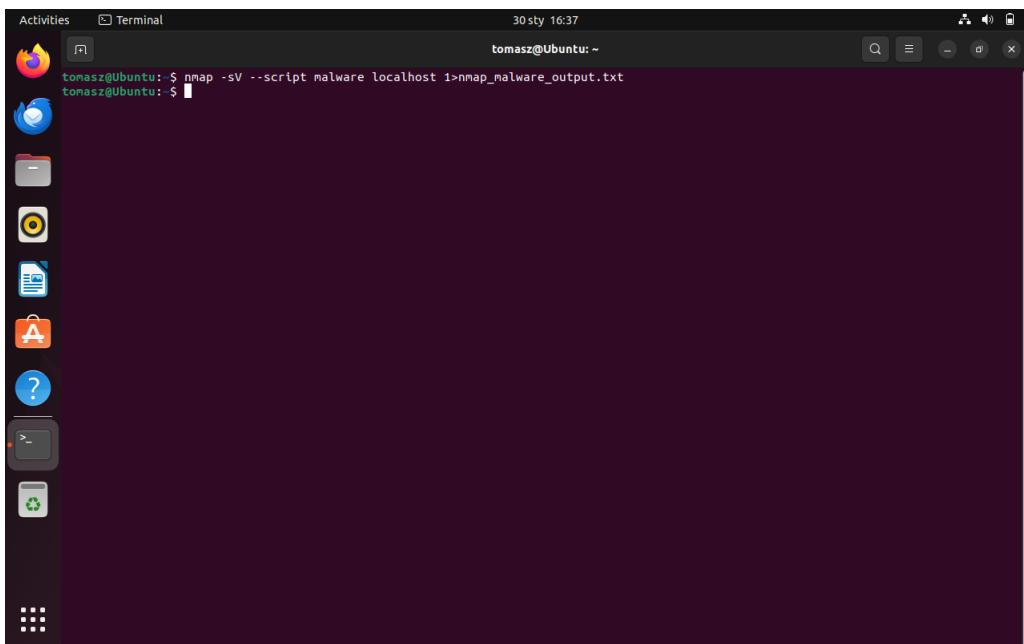


```
Activities Terminal 30 sty 16:32
tomasz@Ubuntu:~$ nmap -sV --script nmap-vulners/ localhost 1>cve_scan_output.txt
tomasz@Ubuntu:~$
```

Figure 15: Searching for CVE vulnerabilities

Output of searching is provided in file cve_scan_output.txt.

5.3 Perform malware and backdoors detection scanning



```
Activities Terminal 30 sty 16:37
tomasz@Ubuntu:~$ nmap -sV --script malware localhost 1>nmap_malware_output.txt
tomasz@Ubuntu:~$
```

Figure 16: Searching for CVE vulnerabilities

Output of searching is provided in file nmap_malware_output.txt.

6 Task 7

6.1 Capture traffic on ports 67 and 68

6.1.1 Run tcpdump to capture network on public interface enp0s8

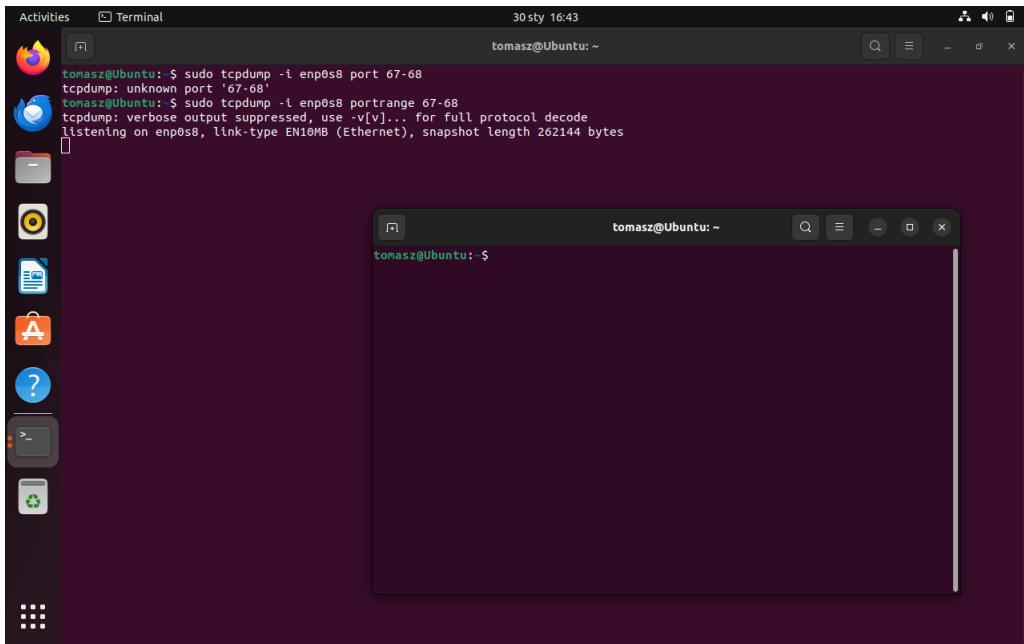


Figure 17: Run tcpdump to listen

6.1.2 Request IP address

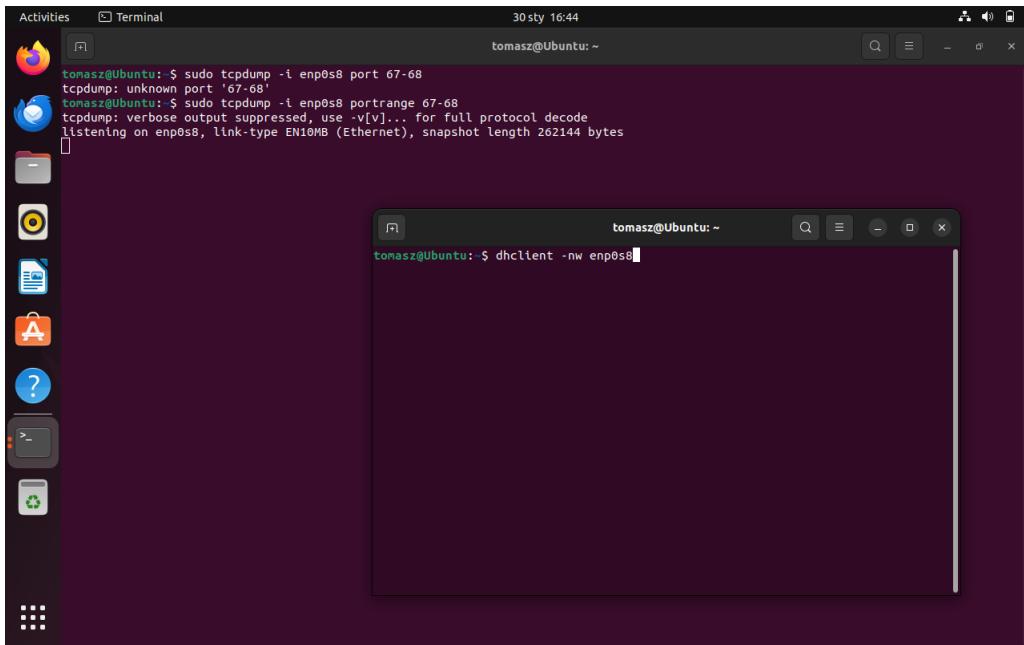
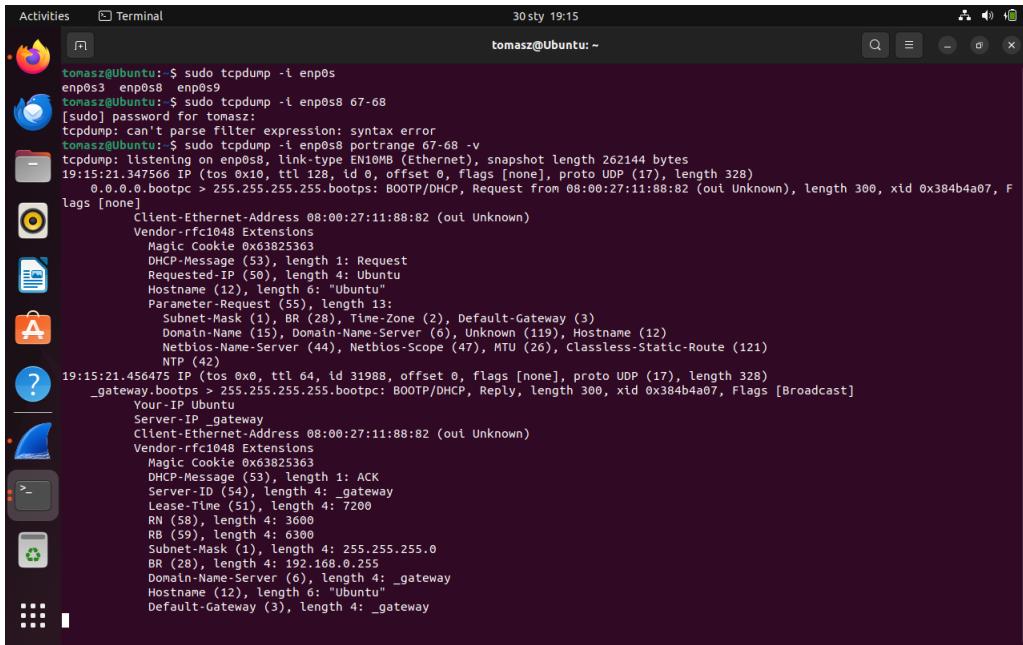


Figure 18: Request IP address

6.1.3 Answer of DHCP server



```
tomasz@Ubuntu:~$ sudo tcpdump -i enp0s8
enp0s3  enp0s8  enp0s9
tomasz@Ubuntu:~$ sudo tcpdump -i enp0s8 67-68
[sudo] password for tomasz:
tcpdump: can't parse filter expression: syntax error
tomasz@Ubuntu:~$ sudo tcpdump -i enp0s8 portrange 67-68 -v
tcpdump: listening on enp0s8, link-type EN10MB (Ethernet), snapshot length 262144 bytes
19:15:21.347566 IP (tos 0x10, ttl 128, id 0, offset 0, flags [none], proto UDP (17), length 328)
    0.0.0.0.bootps > 255.255.255.255.bootps: BOOTP/DHCP, Request From 08:00:27:11:88:82 (oui Unknown), length 300, xid 0x384b4a07, Flags [none]
        Client-Ethernet-Address 08:00:27:11:88:82 (out Unknown)
        Vendor-rfc1048 Extensions
            Magic Cookie 0x63825363
            DHCP-Messaga (53), length 1: Request
            Requested-IP (50), length 4: Ubuntu
            Hostname (12), length 6: "Ubuntu"
            Parameter-Request (55), length 13:
                Subnet-Mask (1), BR (28), Time-Zone (2), Default-Gateway (3)
                Domain-Name (15), Domain-Name-Server (6), Unknown (119), Hostname (12)
                Netbios-Name-Server (44), Netbios-Scope (47), MTU (26), Classless-Static-Route (121)
                NTP (42)
        19:15:21.456475 IP (tos 0x0, ttl 64, id 31988, offset 0, flags [none], proto UDP (17), length 328)
            _gateway.bootps > 255.255.255.255.bootpc: BOOTP/DHCP, Reply, length 300, xid 0x384b4a07, Flags [Broadcast]
                Your-IP Ubuntu
                Server-IP _gateway
                Client-Ethernet-Address 08:00:27:11:88:82 (oui Unknown)
                Vendor-rfc1048 Extensions
                    Magic Cookie 0x63825363
                    DHCP-Messaga (53), length 1: ACK
                    Server-ID (54), length 4: _gateway
                    Lease-Time (51), length 4: 7200
                    RN (58), length 4: 3600
                    RB (59), length 4: 6300
                    Subnet-Mask (1), length 4: 255.255.255.0
                    BR (28), length 4: 192.168.0.255
                    Domain-Name-Server (6), length 4: _gateway
                    Hostname (12), length 6: "Ubuntu"
                    Default-Gateway (3), length 4: _gateway
tomasz@Ubuntu:~$
```

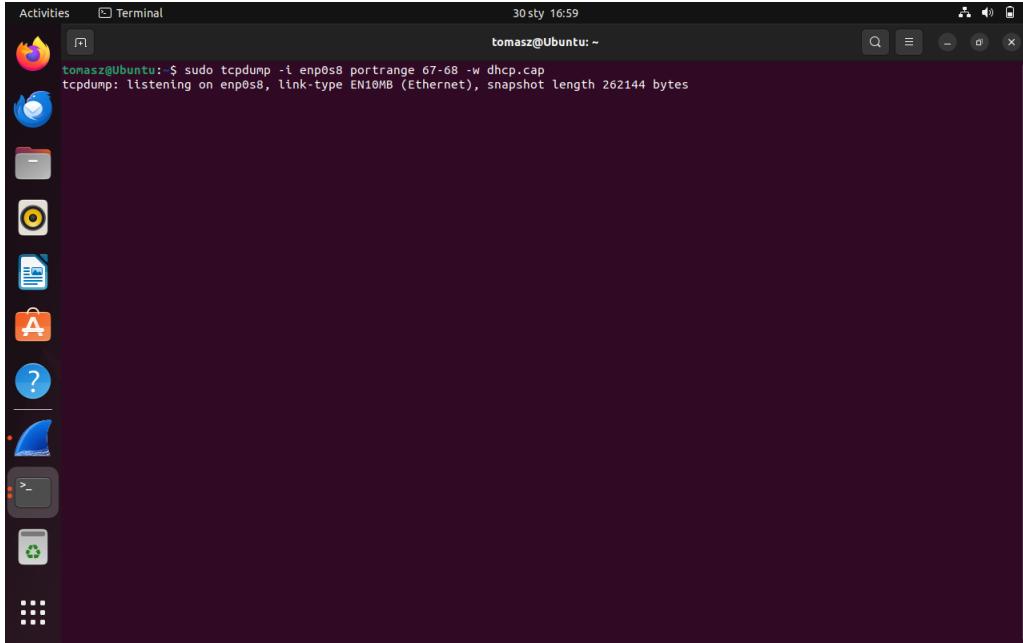
Figure 19: Answer of DHCP request

6.2 Evaluate content

- Client-Ethernet-Address: 08:00:27:11:88:82
- Requested-IP Option 50: Ubuntu
- Hostname Option 12: Ubuntu
- SRC IP Address for the initial Request Message: 0.0.0.0
- DST IP Address for the initial Request Message: 255.255.255.255
- SRC IP Addresses for the ACK Reply Message: 192.168.0.1 (_gateway)
- DST IP Addresses for the ACK Reply Message: 255.255.255.255

6.3 Task 8

6.3.1 Save tcpdump to file of .pcap format



```
tomasz@Ubuntu: ~
tomasz@Ubuntu: $ sudo tcpdump -i enp0s8 portrange 67-68 -w dhcp.cap
tcpdump: listening on enp0s8, link-type EN10MB (Ethernet), snapshot length 262144 bytes
```

Figure 20: Answer of DHCP request

6.3.2 Wireshark output

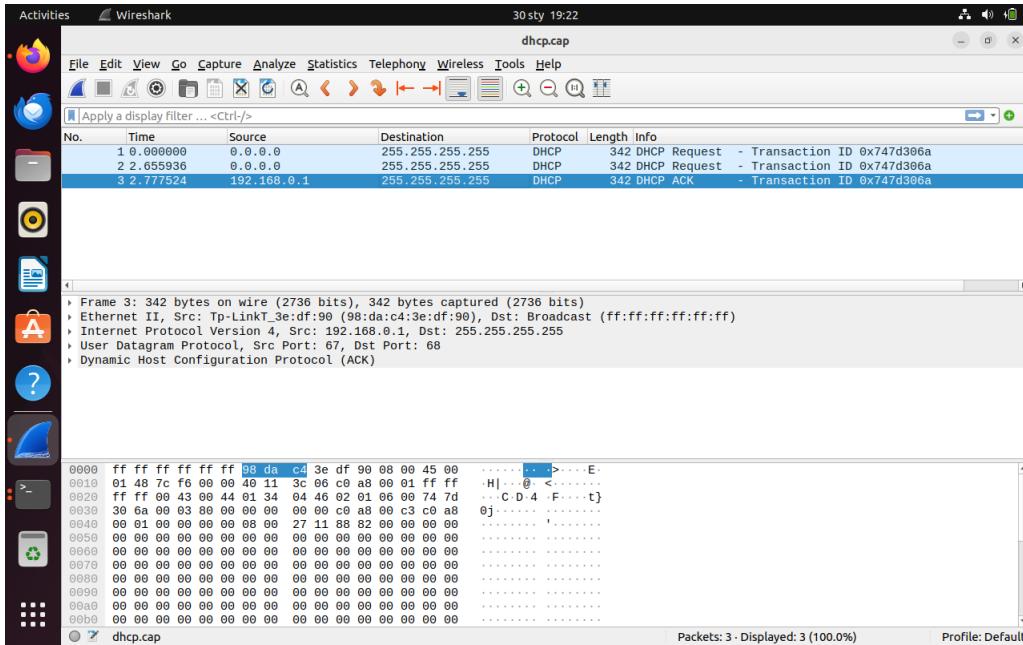


Figure 21: Wireshark output

6.4 Evaluate content

- Client-Ethernet-Address: 08:00:27:11:88:82
- Hostname Option 12: Ubuntu

- SRC IP Address for the initial Request Message: 0.0.0.0
- DST IP Address for the initial Request Message: 255.255.255.255
- SRC IP Addresses for the ACK Reply Message: 192.168.0.1
- DST IP Addresses for the ACK Reply Message: 255.255.255.255

7 Task 9

Task has been made on VM machine and on host machine via host-only adapter.

7.1 Install iperf

On both computers iperf utility has been installed.

```
tomasz@tomasz-Aspire-A515-54G:~$ sudo apt install iperf
[sudo] password for tomasz:
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
The following packages were automatically installed and are no longer required:
  libqt5help5 libqt5sql5 libqt5sql5-sqlite libqt5xml5 libsdl-ttf2.0-0
Use 'sudo apt autoremove' to remove them.
The following NEW packages will be installed:
  iperf
0 upgraded, 1 newly installed, 0 to remove and 19 not upgraded.
Need to get 121 kB of archives.
After this operation, 315 kB of additional disk space will be used.
Get:1 http://pl.archive.ubuntu.com/ubuntu jammy/universe amd64 iperf amd64 2.1.5+dfsg1-1 [121 kB]
Fetched 121 kB in 1s (105 kB/s)
Selecting previously unselected package iperf.
(Reading database ... 276473 files and directories currently installed.)
Preparing to unpack .../iperf_2.1.5+dfsg1-1_amd64.deb ...
Unpacking iperf (2.1.5+dfsg1-1) ...
Setting up iperf (2.1.5+dfsg1-1) ...
Processing triggers for man-db (2.10.2-1) ...
Processing triggers for ufw (0.36.1-4ubuntu0.1) ...
tomasz@tomasz-Aspire-A515-54G:~$
```

Figure 22: Iperf installation

7.2 Measure bandwidth

On host computer iperf has been turned on in server mode. VM has played the role of client. Communication has been established on port 2020. Measurement has been made twice. One time for tcp connection, secondly for udp connection.

```
tomasz@tomasz-Aspire-A515-54G:~$ iperf -s -p 2020 -o iperf_output.txt
Output from stdout and stderr will be redirected to file iperf_output.txt
^Ctomasz@tomasz-Aspire-A515-54G:~$ iperf -s -u -p 2020 -o iperf_output.txt
Output from stdout and stderr will be redirected to file iperf_output.txt
```

Figure 23: Measure bandwidth

```
tomasz@Ubuntu:~$ iperf -c -u -p 2020 localhost
iperf: ignoring extra argument -- localhost
-u: Unknown host
-u: Unknown host
-u: Unknown host
tomasz@Ubuntu:~$ iperf -c -u -p 2020
-----[ 1] Client connecting to localhost, UDP port 2020
-----[ 1] Sending 1470 byte datagrams, IPG target: 11215.21 us (kalman adjust)
-----UDP buffer size: 208 KByte (default)
-----[ 1] local 127.0.0.1 port 50033 connected with 127.0.0.1 port 2020
[ ID] Interval Transfer Bandwidth
[ 1] 0.0000-10.0044 sec 1.25 MBytes 1.05 Mbits/sec
[ 1] Sent 894 datagrams
read failed: Connection refused
```

Figure 24: Iperf for client and udp protocol

Command for server:

- tcp: iperf -s -p 2020 -o iperf_output.txt
- udp: iperf -s -u -p 2020 -o iperf_output_udp.txt

Commands for client:

- tcp: iperf -c -p 2020 localhost
- udp: iperf -c -u -p 2020

Ouput of iperf has been provided on disk:

- iperf_output.txt for tcp

Measured values:

- tcp: $8.14 \frac{Gbits}{sec}$
- udp: $1.05 \frac{Mbits}{sec}$