Linux Process Analysis

Each component has a distinct role within the system's overall functionality, from executing scheduled tasks to providing services to automate routine operations and enable user interaction. However, their crucial roles bring the potential for exploitation by malicious actors that have gained a foothold in the system.

We start by using verified binaries to avoid risk of altered utilities by the actor. This can be done by mounting a USB drive and exporting the path and shared libraries.

> export PATH=/mnt/usb/bin:/mnt/usb/sbin

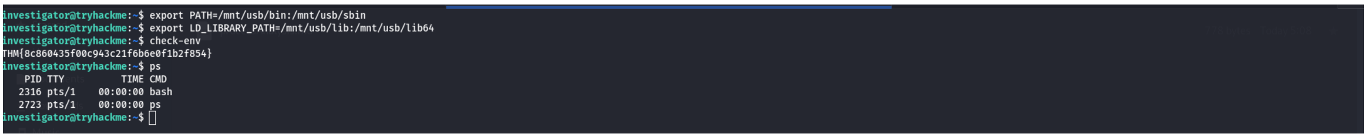
> export LD_LIBRARY_PATH=/mnt/usb/lib:/mnt/usb/lib64

> check-env

Processes

This is a running state of a program. A process (parent) can spawn another process(child). This helps in resource allocation by the OS.

We can use ***ps*** to check running processes.



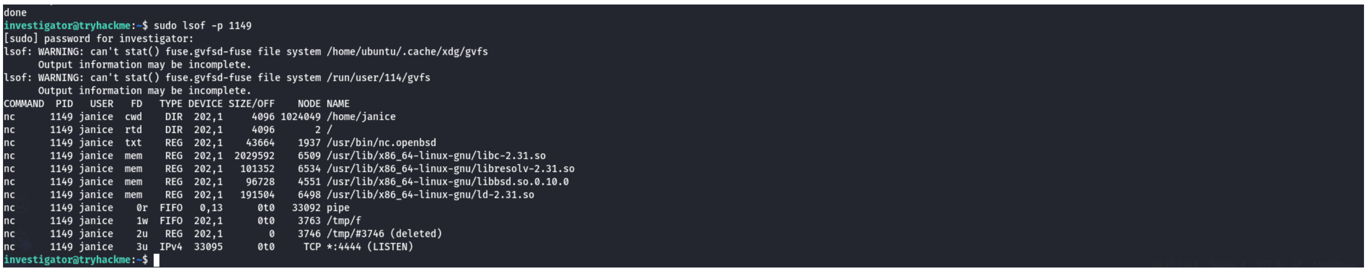
```
investigator@tryhackme:~$ export PATH=/mnt/usb/bin:/mnt/usb/sbin
investigator@tryhackme:~$ export LD_LIBRARY_PATH=/mnt/usb/lib:/mnt/usb/lib64
investigator@tryhackme:~$ check-env
THM{8c860435f00c943c21f6b6e0f1b2f854}
investigator@tryhackme:~$ ps
    PID TTY          TIME CMD
   2316 pts/1    00:00:00 bash
   2723 pts/1    00:00:00 ps
investigator@tryhackme:~$ 
```

> ps -eFH #provides a comprehensive process list which are ordered

LSOF

This utility lists open files and the process associated with it.

> sudo lsof -p 1149

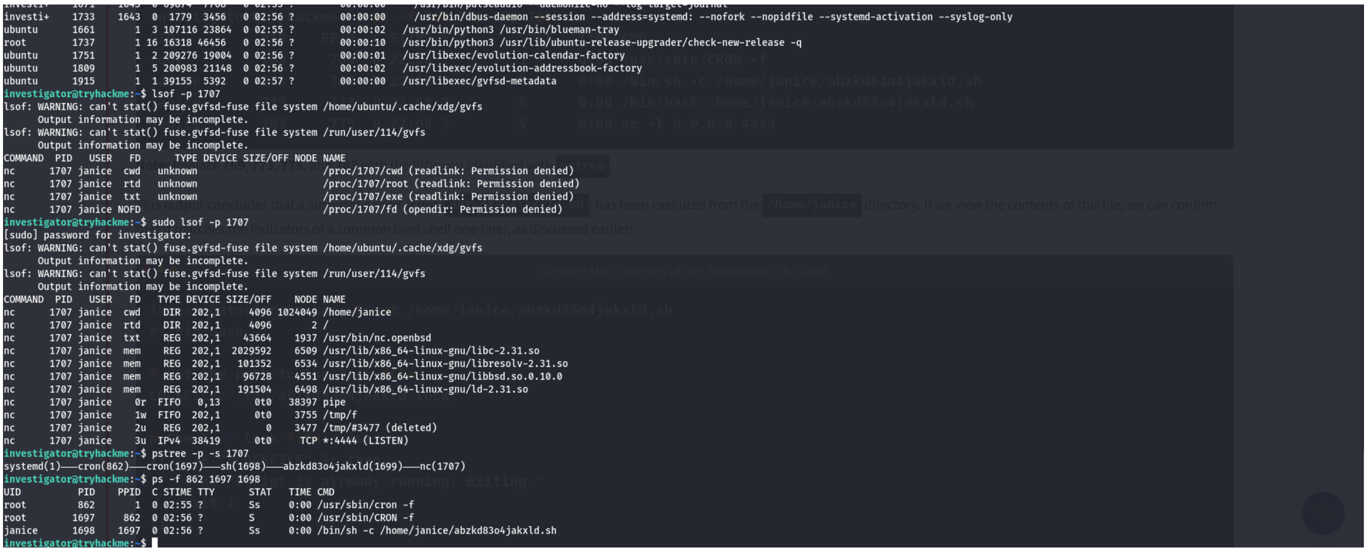


```
done
investigator@tryhackme:~$ sudo lsof -p 1149
[sudo] password for investigator:
lsof: WARNING: can't stat() fuse.gvfsd-fuse file system /home/ubuntu/.cache/xdg/gvfs
      Output information may be incomplete.
lsof: WARNING: can't stat() fuse.gvfsd-fuse file system /run/user/114/gvfs
      Output information may be incomplete.
COMMAND  PID   USER   FD   TYPE DEVICE SIZE/OFF    NODE NAME
nc      1149 janice  cwd    DIR  202,1     4096 1024049 /home/janice
nc      1149 janice  rtd    DIR  202,1     4096       2 /
nc      1149 janice  txt    REG  202,1    43664    1937 /usr/bin/nc.openbsd
nc      1149 janice  mem    REG  202,1  2029592    6509 /usr/lib/x86_64-linux-gnu/libc-2.31.so
nc      1149 janice  mem    REG  202,1   101352    6534 /usr/lib/x86_64-linux-gnu/libresolv-2.31.so
nc      1149 janice  mem    REG  202,1    96728    4551 /usr/lib/x86_64-linux-gnu/libbsd.so.0.10.0
nc      1149 janice  mem    REG  202,1   191504    6498 /usr/lib/x86_64-linux-gnu/ld-2.31.so
nc      1149 janice    0r  FIFO   0,13      0t0   33092 pipe
nc      1149 janice    1w  FIFO  202,1      0t0    3763 /tmp/f
nc      1149 janice    2u   REG  202,1        0    3746 /tmp/#3746 (deleted)
nc      1149 janice    3u  IPv4  33095      0t0     TCP *:4444 (LISTEN)
investigator@tryhackme:~$ 
```

PSTREE

This shows the parent-child relationship tree. Identifying the origin of a process.

> pstree -p -s 1149



```
investi+    1671    1643  0 69874  7768   0 02:55 ?        00:00:00    /usr/bin/pulseaudio --daemonize=no --log-target=journal
investi+    1733    1643  0  1779  3456   0 02:56 ?        00:00:00    /usr/bin/dbus-daemon --session --address=systemd: --nofork --nopidfile --systemd-activation --syslog-only
ubuntu      1661       1  3 107116 23864  0 02:55 ?        00:00:02   /usr/bin/python3 /usr/bin/blueman-tray
root        1737       1 16 16318 46456   0 02:56 ?        00:00:10   /usr/bin/python3 /usr/lib/ubuntu-release-upgrader/check-new-release -q
ubuntu      1751       1  2 209276 19004  0 02:56 ?        00:00:01   /usr/libexec/evolution-calendar-factory
ubuntu      1809       1  5 200983 21148  0 02:56 ?        00:00:02   /usr/libexec/evolution-addressbook-factory
ubuntu      1915       1  1 39155  5392   0 02:57 ?        00:00:00   /usr/libexec/gvfsd-metadata
investigator@tryhackme:~$ lsof -p 1707
lsof: WARNING: can't stat() fuse.gvfsd-fuse file system /home/ubuntu/.cache/xdg/gvfs
      Output information may be incomplete.
lsof: WARNING: can't stat() fuse.gvfsd-fuse file system /run/user/114/gvfs
      Output information may be incomplete.
COMMAND  PID   USER   FD      TYPE DEVICE SIZE/OFF NODE NAME
nc      1707 janice  cwd   unknown                      /proc/1707/cwd (readlink: Permission denied)
nc      1707 janice  rtd   unknown                      /proc/1707/root (readlink: Permission denied)
nc      1707 janice  txt   unknown                      /proc/1707/exe (readlink: Permission denied)
nc      1707 janice NOFD                                /proc/1707/fd (opendir: Permission denied)
investigator@tryhackme:~$ sudo lsof -p 1707
[sudo] password for investigator:
lsof: WARNING: can't stat() fuse.gvfsd-fuse file system /home/ubuntu/.cache/xdg/gvfs
      Output information may be incomplete.
lsof: WARNING: can't stat() fuse.gvfsd-fuse file system /run/user/114/gvfs
      Output information may be incomplete.
COMMAND  PID   USER   FD   TYPE DEVICE SIZE/OFF    NODE NAME
nc      1707 janice  cwd    DIR  202,1     4096 1024049 /home/janice
nc      1707 janice  rtd    DIR  202,1     4096       2 /
nc      1707 janice  txt    REG  202,1    43664    1937 /usr/bin/nc.openbsd
nc      1707 janice  mem    REG  202,1  2029592    6509 /usr/lib/x86_64-linux-gnu/libc-2.31.so
nc      1707 janice  mem    REG  202,1   101352    6534 /usr/lib/x86_64-linux-gnu/libresolv-2.31.so
nc      1707 janice  mem    REG  202,1    96728    4551 /usr/lib/x86_64-linux-gnu/libbsd.so.0.10.0
nc      1707 janice  mem    REG  202,1   191504    6498 /usr/lib/x86_64-linux-gnu/ld-2.31.so
nc      1707 janice    0r  FIFO   0,13      0t0   38397 pipe
nc      1707 janice    1w  FIFO  202,1      0t0    3755 /tmp/f
nc      1707 janice    2u   REG  202,1        0    3477 /tmp/#3477 (deleted)
nc      1707 janice    3u  IPv4  38419      0t0     TCP *:4444 (LISTEN)
investigator@tryhackme:~$ pstree -p -s 1707
systemd(1)───cron(862)───cron(1697)───sh(1698)───abzkd83o4jakxld(1699)───nc(1707)
investigator@tryhackme:~$ ps -f 862 1697 1698
UID          PID    PPID  C STIME TTY      STAT   TIME CMD
root         862       1  0 02:55 ?        Ss     0:00 /usr/sbin/cron -f
root        1697     862  0 02:56 ?        S      0:00 /usr/sbin/CRON -f
janice      1698    1697  0 02:56 ?        Ss     0:00 /bin/sh -c /home/janice/abzkd83o4jakxld.sh
investigator@tryhackme:~$ 
```

Cronjobs

Cronjobs are scheduled tasks executed automatically at predefined intervals by the cron daemon. The cron daemon is a background process responsible for managing cronjobs based on configuration files known as crontabs. Users can have their crontab file stored in the /var/spool/cron/crontabs directory. The main crontab file at /etc/crontab governs system-wide cronjobs

system wide configurations

> /etc/crontab

Additional configs can be found here with diff names, /etc/cron.d, /etc/cron.daily and son on.
We can view crons at user lever with their assigned permissions.

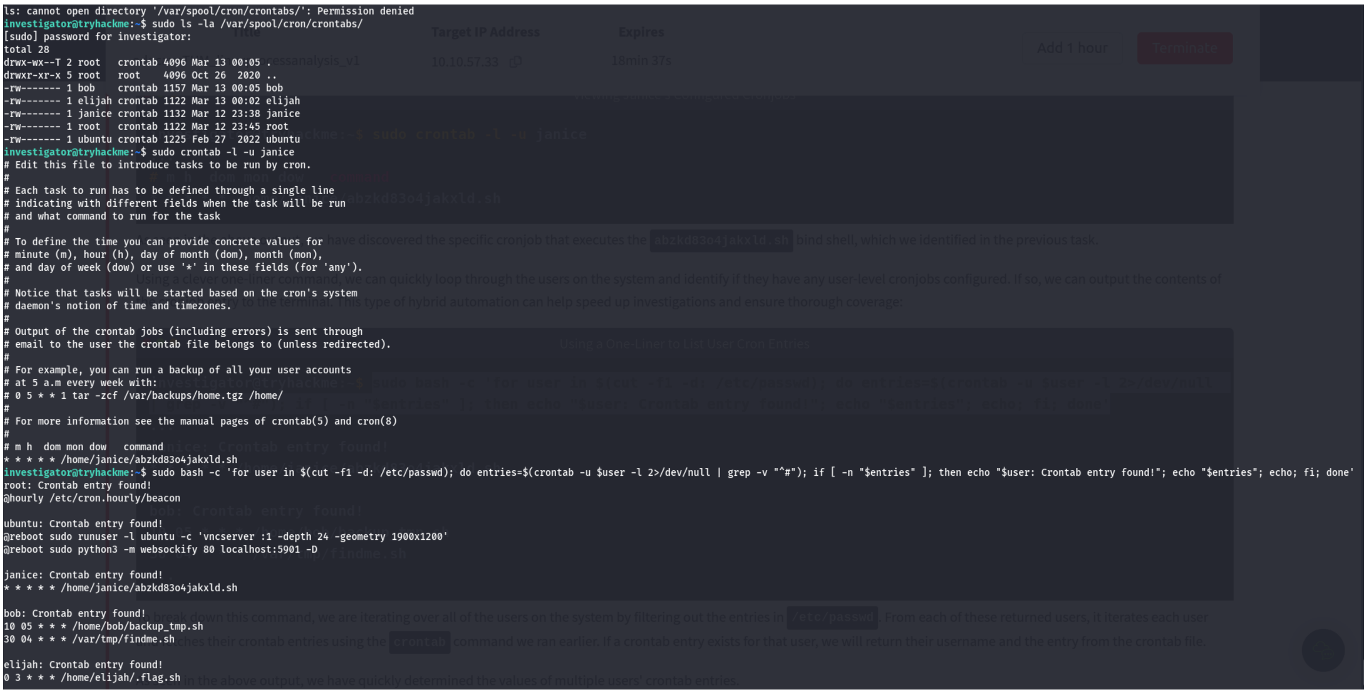
> /var/spool/cron/crontabs/

We can further use cat or crontab to analyse them.

> sudo crontab -l -u janice

We can use a one liner to display all user cronjobs

```
sudo bash -c 'for user in $(cut -f1 -d: /etc/passwd); do entries=$(crontab -u $user -l 2>/dev/null | grep -v "^#"); if [ -n "$entries" ]; then echo "$user: Crontab entry found!"; echo "$entries"; echo; fi; done'
```



```
ls: cannot open directory '/var/spool/cron/crontabs/': Permission denied
investigator@tryhackme:~$ sudo ls -la /var/spool/cron/crontabs/
[sudo] password for investigator:
total 28
drwx-wx--T 2 root   crontab 4096 Mar 13 00:05 .
drwxr-xr-x 5 root   root    4096 Oct 26  2020 ..
-rw------- 1 bob    crontab 1157 Mar 13 00:05 bob
-rw------- 1 elijah crontab 1122 Mar 13 00:02 elijah
-rw------- 1 janice crontab 1132 Mar 12 23:38 janice
-rw------- 1 root   crontab 1122 Mar 12 23:45 root
-rw------- 1 ubuntu crontab 1225 Feb 27  2022 ubuntu
investigator@tryhackme:~$ sudo crontab -l -u janice
# 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 timezones.
#
# 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 * * 1 tar -zcf /var/backups/home.tgz /home/
#
# For more information see the manual pages of crontab(5) and cron(8)
#
# m h  dom mon dow   command
* * * * * /home/janice/abzkd83o4jakxld.sh
investigator@tryhackme:~$ sudo bash -c 'for user in $(cut -f1 -d: /etc/passwd); do entries=$(crontab -u $user -l 2>/dev/null | grep -v "^#"); if [ -n "$entries" ]; then echo "$user: Crontab entry found!"; echo "$entries"; echo; fi; done'
root: Crontab entry found!
@hourly /etc/cron.hourly/beacon

ubuntu: Crontab entry found!
@reboot sudo runuser -l ubuntu -c 'vncserver :1 -depth 24 -geometry 1900x1200'
@reboot sudo python3 -m websockify 80 localhost:5901 -D

janice: Crontab entry found!
* * * * * /home/janice/abzkd83o4jakxld.sh

bob: Crontab entry found!
10 05 * * * /home/bob/backup_tmp.sh
30 04 * * * /var/tmp/findme.sh

elijah: Crontab entry found!
0 3 * * * /home/elijah/.flag.sh
```

Logs at cronjobs

> sudo grep cron /var/log/syslog

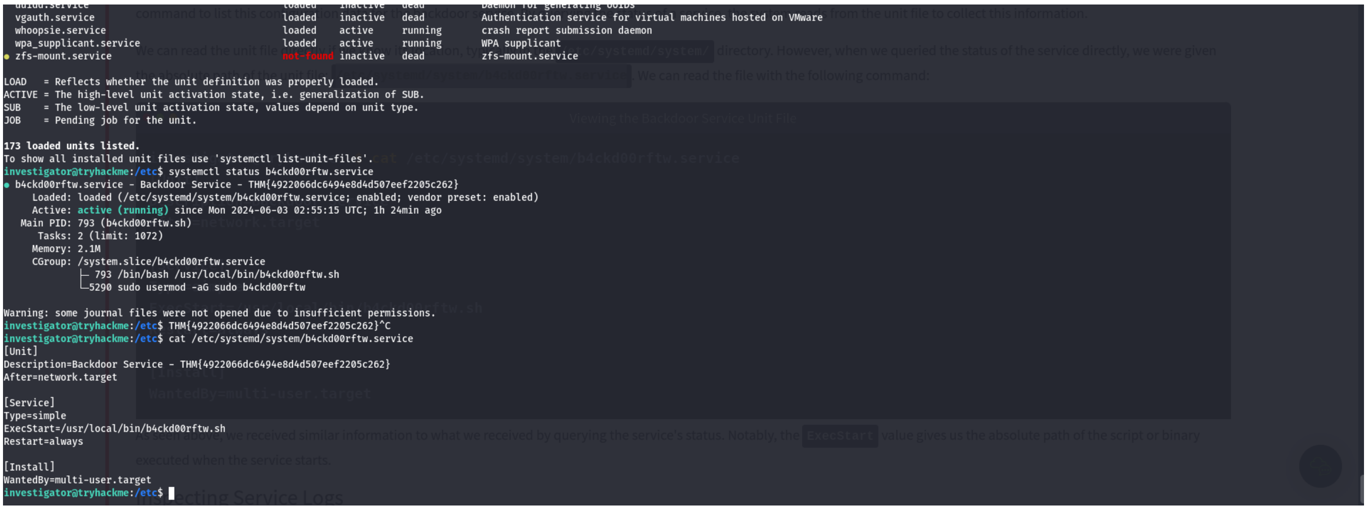
Services

services refer to various background processes or daemons that run continuously, performing tasks such as managing system resources, providing network services, or handling user requests.

Lists all services

> sudo systemctl list-units --all --type=service --no-pager

After we get the path we can read to get more details

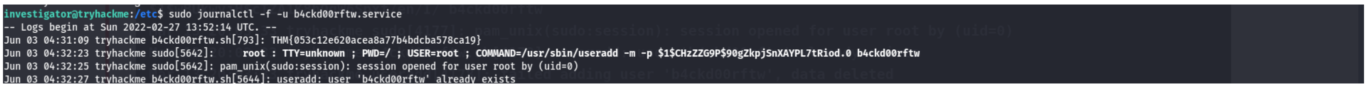


```
  uuidd.service                        loaded    inactive dead      Daemon for generating UUIDs
  vgauth.service                       loaded    inactive dead      Authentication service for virtual machines hosted on VMware
  whoopsie.service                     loaded    active   running   crash report submission daemon
  wpa_supplicant.service               loaded    active   running   WPA supplicant
● zfs-mount.service                    not-found inactive dead      zfs-mount.service

LOAD   = Reflects whether the unit definition was properly loaded.
ACTIVE = The high-level unit activation state, i.e. generalization of SUB.
SUB    = The low-level unit activation state, values depend on unit type.
JOB    = Pending job for the unit.

173 loaded units listed.
To show all installed unit files use 'systemctl list-unit-files'.
investigator@tryhackme:/etc$ systemctl status b4ckd00rftw.service
● b4ckd00rftw.service - Backdoor Service - THM{4922066dc6494e8d4d507eef2205c262}
     Loaded: loaded (/etc/systemd/system/b4ckd00rftw.service; enabled; vendor preset: enabled)
     Active: active (running) since Mon 2024-06-03 02:55:15 UTC; 1h 24min ago
   Main PID: 793 (b4ckd00rftw.sh)
      Tasks: 2 (limit: 1072)
     Memory: 2.1M
     CGroup: /system.slice/b4ckd00rftw.service
             ├─ 793 /bin/bash /usr/local/bin/b4ckd00rftw.sh
             └─5290 sudo usermod -aG sudo b4ckd00rftw

Warning: some journal files were not opened due to insufficient permissions.
investigator@tryhackme:/etc$ THM{4922066dc6494e8d4d507eef2205c262}^C
investigator@tryhackme:/etc$ cat /etc/systemd/system/b4ckd00rftw.service
[Unit]
Description=Backdoor Service - THM{4922066dc6494e8d4d507eef2205c262}
After=network.target

[Service]
Type=simple
ExecStart=/usr/local/bin/b4ckd00rftw.sh
Restart=always

[Install]
WantedBy=multi-user.target
investigator@tryhackme:/etc$ █
```

We then get the absolute path which we use to read the service.

We use journalctl to read the logs
The easiest way to query and view service logs from the systemd journal (the systemd logging service) is through the journalctl command

> sudo journalctl -f -u b4ckd00rftw.service



```
investigator@tryhackme:/etc$ sudo journalctl -f -u b4ckd00rftw.service
-- Logs begin at Sun 2022-02-27 13:52:14 UTC. --
Jun 03 04:31:09 tryhackme b4ckd00rftw.sh[793]: THM{053c12e620acea8a77b4bdcba578ca19}
Jun 03 04:32:23 tryhackme sudo[5642]:     root : TTY=unknown ; PWD=/ ; USER=root ; COMMAND=/usr/sbin/useradd -m -p $1$CHzZZG9P$90gZkpjSnXAYPL7tRiod.0 b4ckd00rftw
Jun 03 04:32:25 tryhackme sudo[5642]: pam_unix(sudo:session): session opened for user root by (uid=0)
Jun 03 04:32:27 tryhackme b4ckd00rftw.sh[5644]: useradd: user 'b4ckd00rftw' already exists
```

AutoStart Scripts

This are scripts that are executed when the systems boots up or user logs in.

- System-wide Autostart Scripts
  - /etc/init.d/ , /etc/rc.d/, /etc/systemd/system/
- User-soecific Auto Scripts
  - ~/.config/autostart/, ~/.config/
  - ls -a /home/*/.config/autostart - to view all users scripts

Application Artifacts

> sudo dpkg -l #lists all installed packages and their versions

```
find /home/ -type f -name ".viminfo" 2>/dev/null
.nano_hsitory
sudo find /home -type d \( -path "*/.mozilla/firefox" -o -path
"*/.config/google-chrome" \) 2>/dev/null
```

Browser artifacts

```
sudo python3 /home/investigator/dumpzilla.py
/home/eduardo/.mozilla/firefox/niijyovp.default-release --Summary --
Verbosity CRITICAL
sudo python3 /home/investigator/dumpzilla.py
/home/eduardo/.mozilla/firefox/niijyovp.default-release --Cookies
```