



МИНОБРНАУКИ РОССИИ
Федеральное государственное бюджетное образовательное учреждение
высшего образования
«МИРЭА –Российский технологический университет»
РТУ МИРЭА

Институт кибербезопасности и цифровых технологий направление
«Киберразведка и противодействие угрозам с применением технологий
искусственного интеллекта»

Кафедра КБ-4 «Интеллектуальные системы информационной безопасности»

ЛАБОРАТОРНЫЙ ПРАКТИКУМ (отчет)
по дисциплине «Методы сбора и обработки данных
из открытых источников»

Выполнил студент 1 курса
Аверьянов Павел Дмитриевич
Группа: ББМО-01-24
Шифр: 24Б1493

Проверил: Литвин Игорь
Анатольевич


Практическая работа 5

Цель работы:

1. Научиться работать с sshprank

Задача:

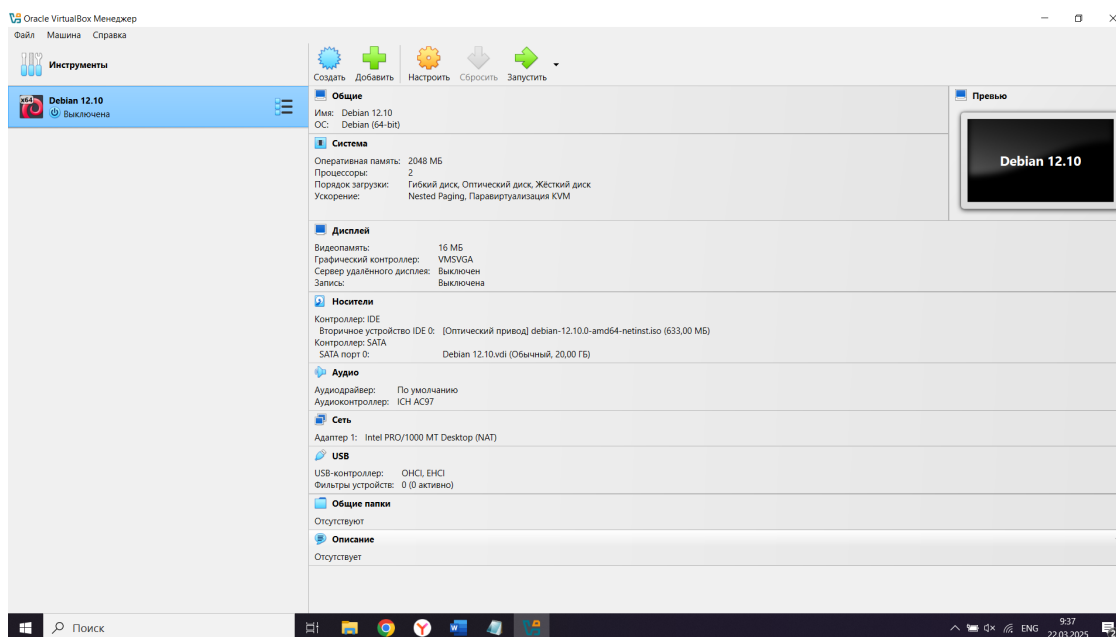
1. Развернуть сервер SSH;
2. Просканировать свою сеть sshprank;
3. Составить словари с помощью:

 Практическая работа "pwdlogy & crunch"
4. Подобрать логин:пароль к SSH;
5. Разобраться, что такое masscan;
6. Просканировать свою сеть с помощью masscan по портам 22,80,2222,8080;
7. Разобраться как masscan работает совместно с nmap, произвести сканирование минимум 10 аргументами из masscan --nmap.

Быстрый массовый сканер SSH, взломщик входа (брут-форс учётных данных) и сборщик баннеров. Этот инструмент использует модуль python-masscan.

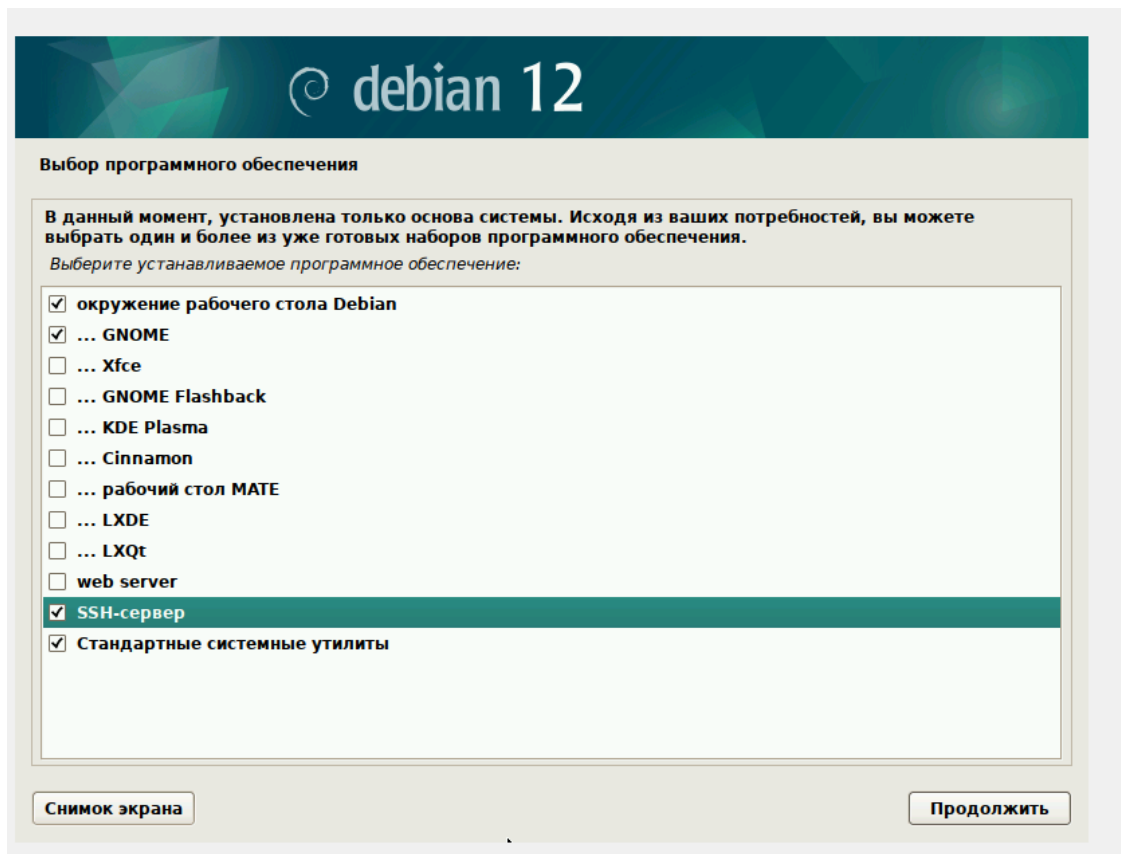
1. Развернуть сервер SSH

Скачаем образ Debian 12.10 и запустим его на виртуальной машине.

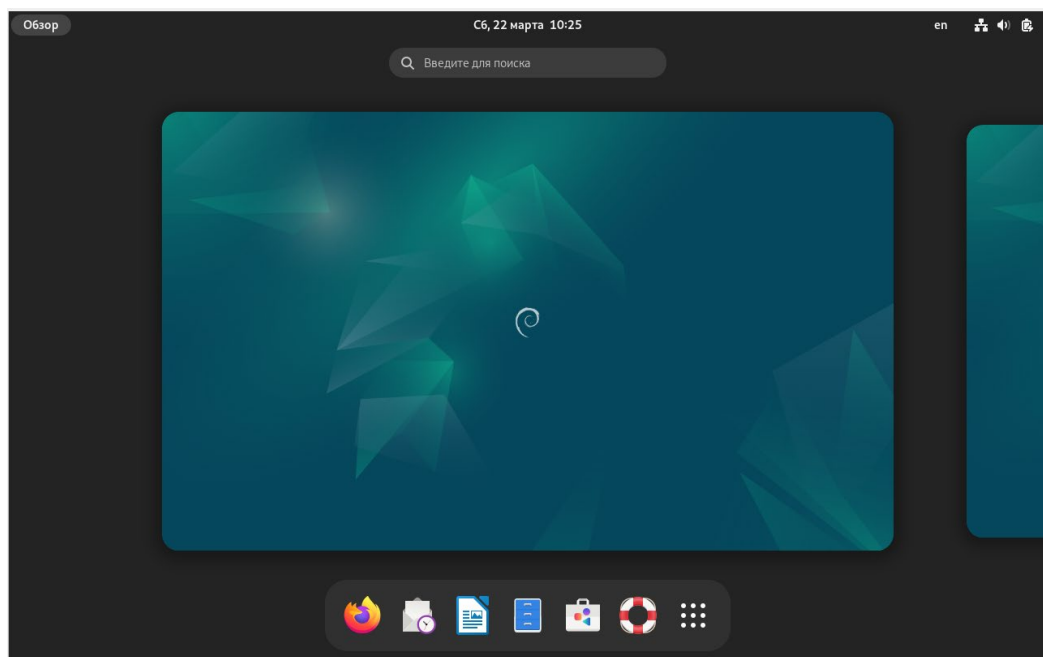


Далее выполняем базовую настройку.

Устанавливаем SSH-сервер



Успешно установили ОС.



Далее проверим работоспособность SSH-сервера:

```
sudo systemctl status ssh
```

```
pavel@debian1210: ~  
pavel@debian1210:~$ sudo systemctl status ssh  
[sudo] пароль для pavel:  
pavel is not in the sudoers file.  
pavel@debian1210:~$ su root  
Пароль:  
root@debian1210:/home/pavel# sudo systemctl status ssh  
● ssh.service - OpenBSD Secure Shell server  
   Loaded: loaded (/lib/systemd/system/ssh.service; enabled; preset: enabled)  
   Active: active (running) since Sat 2025-03-22 10:24:31 MSK; 24min ago  
     Docs: man:sshd(8)  
           man:sshd_config(5)  
  Process: 559 ExecStartPre=/usr/sbin/sshd -t (code=exited, status=0/SUCCESS)  
 Main PID: 576 (sshd)  
    Tasks: 1 (limit: 2282)  
  Memory: 5.1M  
     CPU: 131ms  
   CGroup: /system.slice/ssh.service  
           └─576 "sshd: /usr/sbin/sshd -D [listener] 0 of 10-100 startups"  
  
map 22 10:24:30 debian1210 systemd[1]: Starting ssh.service - OpenBSD Secure Sh  
map 22 10:24:31 debian1210 sshd[576]: Server listening on 0.0.0.0 port 22.  
map 22 10:24:31 debian1210 sshd[576]: Server listening on :: port 22.  
map 22 10:24:31 debian1210 systemd[1]: Started ssh.service - OpenBSD Secure She  
lines 1-17/17 (END)
```

2. Просканировать свою сеть sshprank

Для этого сначала выполним установку sshprank:

Первым делом установим менеджер пакетов PIP для Python 3

Sudo apt install python3-pip

```
pavel@debian1210: ~  
map 22 10:24:30 debian1210 systemd[1]: Starting ssh.service - OpenBSD Secure Sh  
map 22 10:24:31 debian1210 sshd[576]: Server listening on 0.0.0.0 port 22.  
map 22 10:24:31 debian1210 sshd[576]: Server listening on :: port 22.  
map 22 10:24:31 debian1210 systemd[1]: Started ssh.service - OpenBSD Secure She  
  
root@debian1210:/home/pavel# sudo apt install python3-pip  
Чтение списков пакетов... Готово  
Построение дерева зависимостей... Готово  
Чтение информации о состоянии... Готово  
Будут установлены следующие дополнительные пакеты:  
  binutils binutils-common binutils-x86-64-linux-gnu build-essential dpkg-dev  
  fakeroot g++ g++-12 gcc gcc-12 libalgorithm-diff-perl  
  libalgorithm-diff-xs-perl libalgorithm-merge-perl libasan8 libbinutils  
  libc-dev-bin libc-devtools libc6-dev libcc1-0 libcrypt-dev libctf-nobfd0  
  libctf0 libdpkg-perl libexpat1-dev libfakeroot libfile-fcntllock-perl  
  libgcc-12-dev libgprofng0 libitm1 libjs-jquery libjs-sphinxdoc  
  libjs-underscore liblsan0 libnsl-dev libpython3-dev libpython3.11-dev  
  libstdc++-12-dev libtirpc-dev libtsan2 libubsan1 linux-libc-dev make  
  manpages-dev patch python3-dev python3-distutils python3-lib2to3  
  python3-setuptools python3-wheel python3.11-dev rpcsvc-proto zlib1g-dev  
Предлагаемые пакеты:  
  binutils-doc debian-keyring g++-multilib g++-12-multilib gcc-12-doc  
  gcc-multilib autoconf automake libtool flex bison gdb gcc-doc
```

Установка пакетов выполнена успешно

```
pavel@debian1210: ~  
Настраивается пакет libjs-sphinxdoc (5.3.0-4) ...  
Настраивается пакет libc6-dev:amd64 (2.36-9+deb12u10) ...  
Настраивается пакет binutils-x86-64-linux-gnu (2.40-2) ...  
Настраивается пакет libstdc++-12-dev:amd64 (12.2.0-14) ...  
Настраивается пакет binutils (2.40-2) ...  
Настраивается пакет dpkg-dev (1.21.22) ...  
Настраивается пакет libexpat1-dev:amd64 (2.5.0-1+deb12u1) ...  
Настраивается пакет gcc-12 (12.2.0-14) ...  
Настраивается пакет zlib1g-dev:amd64 (1:1.2.13.dfsg-1) ...  
Настраивается пакет g++-12 (12.2.0-14) ...  
Настраивается пакет gcc (4:12.2.0-3) ...  
Настраивается пакет libpython3.11-dev:amd64 (3.11.2-6+deb12u5) ...  
Настраивается пакет g++ (4:12.2.0-3) ...  
update-alternatives: используется /usr/bin/g++ для предоставления /usr/bin/c++ (с++) в автоматическом режиме  
Настраивается пакет build-essential (12.9) ...  
Настраивается пакет libpython3-dev:amd64 (3.11.2-1+b1) ...  
Настраивается пакет python3.11-dev (3.11.2-6+deb12u5) ...  
Настраивается пакет python3-dev (3.11.2-1+b1) ...  
Обрабатываются триггеры для man-db (2.11.2-2) ...  
Обрабатываются триггеры для libc-bin (2.36-9+deb12u10) ...  
root@debian1210: /home/pavel#
```

Далее скопируем репозиторий

Но для этого необходимо установить git

```
pavel@debian1210: ~  
Все пакеты имеют последние версии.  
root@debian1210: /tmp# git clone https://github.com/noptrix/sshprank  
bash: git: команда не найдена  
root@debian1210: /tmp# sudo apt install git  
Чтение списков пакетов... Готово  
Построение дерева зависимостей... Готово  
Чтение информации о состоянии... Готово  
Будут установлены следующие дополнительные пакеты:  
  git-man liberror-perl  
Предлагаемые пакеты:  
  git-daemon-run | git-daemon-sysvinit git-doc git-email git-gui gitk gitweb  
  git-cvs git-mediawiki git-svn  
Следующие НОВЫЕ пакеты будут установлены:  
  git git-man liberror-perl  
Обновлено 0 пакетов, установлено 3 новых пакетов, для удаления отмечено 0 пакетов,  
и 0 пакетов не обновлено.  
Необходимо скачать 9 342 kB архивов.  
После данной операции объем занятого дискового пространства возрастёт на 48,2 MB  
.  
Хотите продолжить? [Д/н] Д  
Пол:1 http://deb.debian.org/debian bookworm/main amd64 liberror-perl all 0.17029-2 [29,0 kB]  
Пол:2 http://deb.debian.org/debian bookworm/main amd64 git-man all 1:2.39.5-0+deb12u2 [2 053 kB]
```

Теперь склонируем репозиторий

```
pavel@debian1210: ~
Получено 9 342 kB за 25с (373 kB/s)
Выбор ранее не выбранного пакета liberror-perl.
(Чтение базы данных ... на данный момент установлено 157863 файла и каталога.)
Подготовка к распаковке .../liberror-perl_0.17029-2_all.deb ...
Распаковывается liberror-perl (0.17029-2) ...
Выбор ранее не выбранного пакета git-man.
Подготовка к распаковке .../git-man_1%3a2.39.5-0+deb12u2_all.deb ...
Распаковывается git-man (1:2.39.5-0+deb12u2) ...
Выбор ранее не выбранного пакета git.
Подготовка к распаковке .../git_1%3a2.39.5-0+deb12u2_amd64.deb ...
Распаковывается git (1:2.39.5-0+deb12u2) ...
Настраивается пакет liberror-perl (0.17029-2) ...
Настраивается пакет git-man (1:2.39.5-0+deb12u2) ...
Настраивается пакет git (1:2.39.5-0+deb12u2) ...
Обрабатываются триггеры для man-db (2.11.2-2) ...
root@debian1210:/tmp# git clone https://github.com/noptrix/sshprank
Клонирование в «sshprank»...
remote: Enumerating objects: 231, done.
remote: Counting objects: 100% (70/70), done.
remote: Compressing objects: 100% (40/40), done.
remote: Total 231 (delta 28), reused 68 (delta 28), pack-reused 161 (from 1)
Получение объектов: 100% (231/231), 52.61 КиБ | 1.10 МБ/с, готово.
Определение изменений: 100% (109/109), готово.
root@debian1210:/tmp#
```

Далее установим masscan, shodan, paramiko

```
pavel@debian1210: ~
note: If you believe this is a mistake, please contact your Python installation
or OS distribution provider. You can override this, at the risk of breaking your
Python installation or OS, by passing --break-system-packages.
hint: See PEP 668 for the detailed specification.
root@debian1210:/tmp/sshprank# sudo pip3 install -r docs/requirements.txt --brea
k-system-packages
Collecting paramiko
  Downloading paramiko-3.5.1-py3-none-any.whl (227 kB)
    227.3/227.3 kB 1.9 MB/s eta 0:00:00
Collecting python-masscan
  Downloading python-masscan-1.0.0.tar.gz (7.7 kB)
  Preparing metadata (setup.py) ... done
Collecting shodan
  Downloading shodan-1.31.0.tar.gz (57 kB)
    57.9/57.9 kB 2.5 MB/s eta 0:00:00
  Preparing metadata (setup.py) ... done
Collecting bcrypt>=3.2
  Downloading bcrypt-4.3.0-cp39-abi3-manylinux_2_34_x86_64.whl (284 kB)
    284.2/284.2 kB 2.7 MB/s eta 0:00:00
Requirement already satisfied: cryptography>=3.3 in /usr/lib/python3/dist-packag
es (from paramiko->-r docs/requirements.txt (line 1)) (38.0.4)
Collecting pynacl>=1.5
  Downloading PyNaCl-1.5.0-cp36-abi3-manylinux_2_17_x86_64.manylinux2014_x86_64.
```

Далее запустим sshprank и посмотрим справку с помощью флага -H

По умолчанию программа не выводит никакой информации, даже не показывает взломанные хосты, логины и пароли. Чтобы эта информация выводилась на экран, используйте опцию -v:

[illegible]

3. Составить словари с помощью pwdlogy и crunch

Начну с pwdlogy

Pwdloggy поможет нам создавать файлы с паролями, скажем, для одной жертвы, о которой необходимо собрать как можно больше личной информации.

Взлом паролей не так прост, как создание словаря. Pwdlogy только позволит создать профильный словарь, он генерирует список возможных паролей по заданному набору ключевых слов, дней рождения и т.д. Мы можем использовать Pwdlogy с широким набором инструментов, таких как Burpsuite, THC-Hydra, Hashcat, и многих других.

Склонировем репозиторий


```
pavel@debian1210: ~  
  
pwdlogy > help  
  
Commands:  
set          -- change settings  
start        -- generate wordlist  
settings     -- display current settings  
save         -- save current settings  
load         -- load settings from file  
functions    -- display available functions  
help         -- display help menu  
credits      -- display credits  
custom       -- custom parameters  
exit/quit    -- exit console  
  
pwdlogy > start  
  
Generating format [k]  
  
Generating format [K]  
  
Generating format [kn]  
  
Generating format [Kn]
```

```
pavel@debian1210: ~  
  
Generating format [lkknnns]  
  
Generating format [lKknnns]  
  
Generating format [lkknnss]  
  
Generating format [lKknnss]  
  
Generating format [kkB]  
  
Generating format [KkB]  
  
Generating format [lkkB]  
  
Generating format [lKkB]  
====DONE====  
  
root@debian1210:/tmp/pwdlogy# ls  
birthday.txt      formats_default.conf  keywords.txt  settings_default.conf  
commonPhrases.txt functions.txt          pwdlogy.py  
default.conf      gen.txt              README.md  
root@debian1210:/tmp/pwdlogy#
```

Так же создадим список пользователей

```
pavel@debian1210: ~  
  
Generating format [lKknnns]  
Generating format [lkknss]  
Generating format [lKknss]  
Generating format [kkB]  
Generating format [KkB]  
Generating format [lkkB]  
Generating format [lKkB]  
====DONE====  
  
root@debian1210:/tmp/pwdlogy# ls  
birthday.txt      formats_default.conf  keywords.txt  settings_default.conf  
commonPhrases.txt  functions.txt         pwdlogy.py  
default.conf      gen.txt              README.md  
root@debian1210:/tmp/pwdlogy# touch user.txt  
root@debian1210:/tmp/pwdlogy# nano user.txt  
root@debian1210:/tmp/pwdlogy#
```

```
GNU nano 7.2      user.txt  
root  
pavel  
john  
  
[ Прочитано 3 строки ]  
^G Справка      ^O Записать     ^W Поиск        ^K Вырезать     ^T Выполнить    ^C Позиция  
^X Выход        ^R ЧитФайл     ^\ Замена       ^U Вставить     ^J Выровнять    ^_ К строке
```

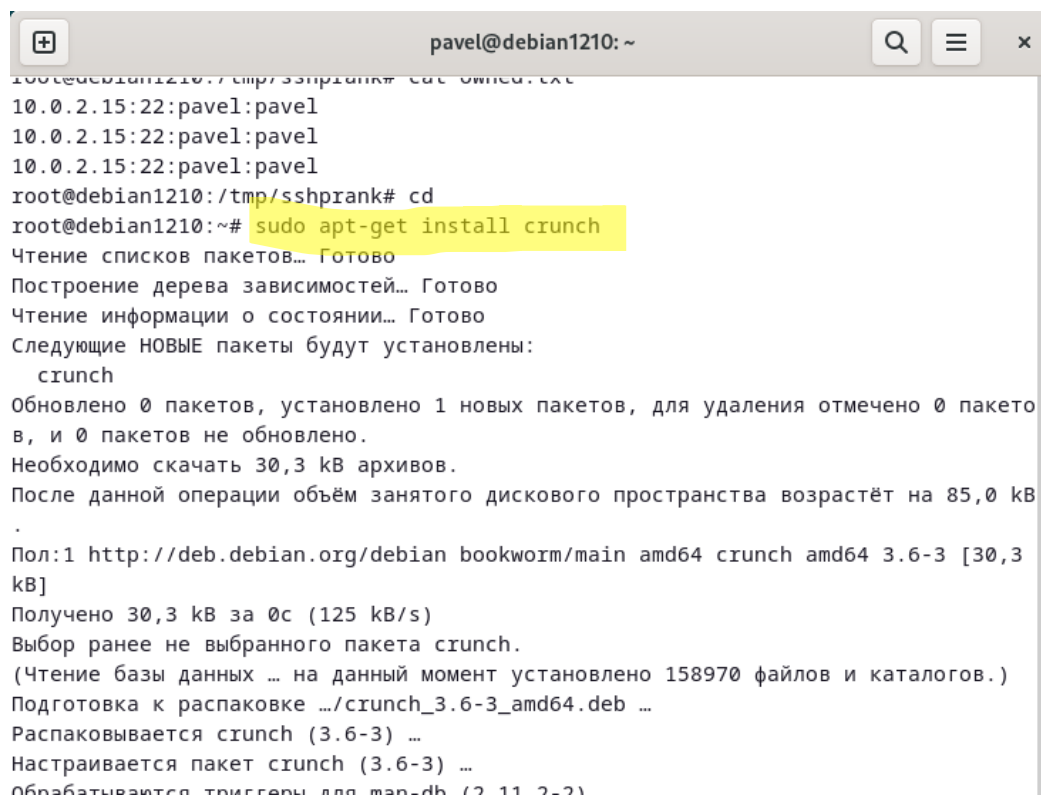
Теперь создадим список паролей с помощью crunch

Crunch – генератор словарей паролей, в которых можно определить стандартную или заданную кодировку. Crunch может произвести все возможные комбинации и перестановки.

Особенности:

1. crunch генерирует списки слов (WordList) как методом комбинации, так и методом перестановки;
2. он может разбить вывод по количеству строк или размеру файла;
3. поддерживается возобновление процесса после остановки;
4. образец (паттерн) поддерживает числа и символы;
5. образец поддерживает по отдельности символы верхнего и нижнего регистра;
6. работая с несколькими файлами, выводит отчёт о статусе;
7. новая опция -l для буквальной поддержки, @,% ^;
8. новая опция -d для ограничения дублирования символов, смотрите man-файл для деталей;
9. поддержка unicode.

Скачаем crunch



```
pavel@debian1210: ~  
root@debian1210:/tmp/sshprank# cat /etc/passwd  
10.0.2.15:22:pavel:pavel  
10.0.2.15:22:pavel:pavel  
10.0.2.15:22:pavel:pavel  
root@debian1210:/tmp/sshprank# cd  
root@debian1210:~# sudo apt-get install crunch  
Чтение списков пакетов... Готово  
Построение дерева зависимостей... Готово  
Чтение информации о состоянии... Готово  
Следующие НОВЫЕ пакеты будут установлены:  
  crunch  
Обновлено 0 пакетов, установлено 1 новых пакетов, для удаления отмечено 0 пакетов,  
и 0 пакетов не обновлено.  
Необходимо скачать 30,3 кВ архивов.  
После данной операции объём занятого дискового пространства возрастёт на 85,0 кВ  
.  
Пол:1 http://deb.debian.org/debian bookworm/main amd64 crunch amd64 3.6-3 [30,3  
кВ]  
Получено 30,3 кВ за 0с (125 kB/s)  
Выбор ранее не выбранного пакета crunch.  
(Чтение базы данных ... на данный момент установлено 158970 файлов и каталогов.)  
Подготовка к распаковке .../crunch_3.6-3_amd64.deb ...  
Распаковывается crunch (3.6-3) ...  
Настраивается пакет crunch (3.6-3) ...  
Обрабатываются триггеры для man-db (2.11.2-2)
```

И генерируем пароли

```
pavel@debian1210: ~
подготовка к распаковке .../crunch_3.6-3_amd64.deb ...
Распаковывается crunch (3.6-3) ...
Настраивается пакет crunch (3.6-3) ...
Обрабатываются триггеры для man-db (2.11.2-2) ...
root@debian1210:~# cd /tmp
root@debian1210:/tmp# cd sshprank
root@debian1210:/tmp/sshprank# crunch 5 5 apcvebl -t p@@@ > pass1.txt
Crunch will now generate the following amount of data: 14406 bytes
0 MB
0 GB
0 TB
0 PB
Crunch will now generate the following number of lines: 2401
root@debian1210:/tmp/sshprank# ls
docs  lists  owned.txt  pass1.txt  pass.txt  README.md  sshprank.py  user.txt
root@debian1210:/tmp/sshprank# cat pass1.txt
раааа
рааар
рааас
рааав
рааае
раааб
рааал
раара
рааан
```

4. Подобрать логин: пароль к SSH

При запуске ssh prank с указанием пользователей и большого количества паролей система долго подбирает пароль и ресурсов моей системы не хватает. Поэтому я решил создать небольшой список паролей и пользователей.

```
pavel@debian1210: ~
root@debian1210:/tmp# cd sshprank
root@debian1210:/tmp/sshprank# touch user.txt
root@debian1210:/tmp/sshprank# touch pass.txt
root@debian1210:/tmp/sshprank#
```

```
pavel@debian1210: ~
GNU nano 7.2 user.txt
root
pavel
admin

[ Прочитано 3 строки ]
^G Справка ^O Записать ^W Поиск ^K Вырезать ^T Выполнить ^C Позиция
^X Выход ^R ЧитФайл ^\ Замена ^U Вставить ^J Выводить ^_ К строке
```

```
pavel@debian1210: ~
GNU nano 7.2 pass.txt *
1234
admin
root
qwerty
pavel
1234567890
qazwsx
password

^G Справка ^O Записать ^W Поиск ^K Вырезать ^T Выполнить ^C Позиция
^X Выход ^R ЧитФайл ^\ Замена ^U Вставить ^J Выводить ^_ К строке
```

Запустим sshprank


```
pavel@debian1210: ~  
[!] login failure: 10.0.2.15:22 (auth failed)  
[!] login failure: 10.0.2.15:22 (auth failed)  
[!] login failure: 10.0.2.15:22 (auth failed)  
[!] login failure: 10.0.2.15:22 (auth failed)  
[!] login failure: 10.0.2.15:22 (auth failed)  
[!] login failure: 10.0.2.15:22 (auth failed)  
[!] login failure: 10.0.2.15:22 (auth failed)  
[!] login failure: 10.0.2.15:22 (auth failed)  
[!] login failure: 10.0.2.15:22 (auth failed)  
[!] login failure: 10.0.2.15:22 (auth failed)  
[*] found login: 10.0.2.15:22:pavel:pavel  
[!] login failure: 10.0.2.15:22 (auth failed)  
[!] login failure: 10.0.2.15:22 (auth failed)  
[!] login failure: 10.0.2.15:22 (auth failed)  
[!] login failure: 10.0.2.15:22 (auth failed)  
[!] login failure: 10.0.2.15:22 (auth failed)  
[!] login failure: 10.0.2.15:22 (auth failed)  
[!] login failure: 10.0.2.15:22 (auth failed)  
[!] login failure: 10.0.2.15:22 (auth failed)  
[!] login failure: 10.0.2.15:22 (auth failed)  
^C  
[!] you aborted me  
root@debian1210: /tmp/sshprank#
```

```
pavel@debian1210: ~  
[!] login failure: 10.0.2.15:22 (auth failed)  
[!] login failure: 10.0.2.15:22 (auth failed)  
[!] login failure: 10.0.2.15:22 (auth failed)  
[!] login failure: 10.0.2.15:22 (auth failed)  
[!] login failure: 10.0.2.15:22 (auth failed)  
[!] login failure: 10.0.2.15:22 (auth failed)  
[*] found login: 10.0.2.15:22:pavel:pavel  
[!] login failure: 10.0.2.15:22 (auth failed)  
[!] login failure: 10.0.2.15:22 (auth failed)  
[!] login failure: 10.0.2.15:22 (auth failed)  
[!] login failure: 10.0.2.15:22 (auth failed)  
[!] login failure: 10.0.2.15:22 (auth failed)  
[!] login failure: 10.0.2.15:22 (auth failed)  
[!] login failure: 10.0.2.15:22 (auth failed)  
[!] login failure: 10.0.2.15:22 (auth failed)  
[!] login failure: 10.0.2.15:22 (auth failed)  
^C  
[!] you aborted me  
root@debian1210: /tmp/sshprank# cat owned.txt  
10.0.2.15:22:pavel:pavel  
10.0.2.15:22:pavel:pavel  
10.0.2.15:22:pavel:pavel  
10.0.2.15:22:pavel:pavel  
root@debian1210: /tmp/sshprank#
```

Как видим пароль нашёлся успешно

5. Разобраться, что такое masscan

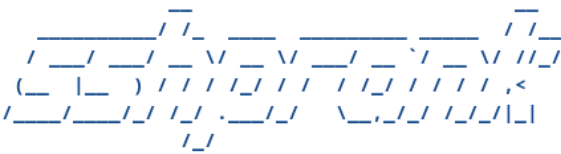
MASSCAN — это массовый сканер IP портов. Это самый быстрый сканер портов Интернета. Отправляя 10 миллионов пакетов в секунду, он может сканировать весь интернет за 6 минут. Этот инструмент полезен для обзора сетей большого масштаба — таких как Интернет или внутренние сети. Хотя скорость по умолчанию ограничена 100 пакетами в секунду, программа может разогнаться до 25 миллионов пакетов в секунду, при такой скорости весь (по одному порту на IP) Интернет будет просканирован за 3 минуты.

Он создаёт результат схожий с nmap, самым знаменитым сканером портов. Внутри он работает подобно scanrand, unicornscan и ZMap, используя асинхронную передачу. Главным отличием является то, что он быстрее этих сканеров. Дополнительно, он более гибкий, позволяет произвольные диапазоны адресов и портов.

6. Просканировать свою сеть с помощью masscan по портам 22,80,2222,8080

Запустим сканирование через опцию -m:

```
root@debian1210:/tmp/sshprank# ./sshprank.py -m '-p22 --range 192.168.1.0/24' -v
```



```
--== [ by nullsecurity.net ] ==--  
  
[+] game started  
[+] scanning and cracking targets  
[/] scanning sshds  
[!] no sshds found :(  
[+] game over  
root@debian1210:/tmp/sshprank# masscan 192.168.1.0/24 -p22,80,2222,8080 > ssh_info.txt  
Starting masscan 1.3.2 (http://bit.ly/14GZzcT) at 2025-03-23 14:26:13 GMT  
Initiating SYN Stealth Scan  
Scanning 256 hosts [4 ports/host]  
root@debian1210:/tmp/sshprank# cat ssh_info.txt  
Discovered open port 22/tcp on 192.168.1.1  
Discovered open port 80/tcp on 192.168.1.14  
root@debian1210:/tmp/sshprank#
```

Хостов не нашлось.

Далее попробую с помощью скачанного masscan через `sudo apt-get install masscan`:

Здесь мы используем скорость сканирования (--rate), порты (-p), диапазон сканирования (--range), интерфейс (-e), сохранение в удобном формате (-oL) и результат сохраняем в masscan.txt.

Далее выполним сканирование с помощью nmap из файла masscan.txt. Указываем, что диапазон IP-адресов нужно брать из файла (-iL). А также флаг -Sv, чтобы определить версии сервисов, работающих на открытых портах. И флаг -oN для записи результата в файл res.txt

```
pavel@debian1210: ~  
root@debian1210:/tmp/sshprank# nmap -iL masscan.txt -sV -p 1-65535 -v -oN res.txt  
Starting Nmap 7.93 ( https://nmap.org ) at 2025-03-23 21:31 MSK  
NSE: Loaded 45 scripts for scanning.  
Failed to resolve "open".  
Failed to resolve "tcp".  
Initiating Ping Scan at 21:31  
Scanning 2 hosts [4 ports/host]  
Completed Ping Scan at 21:31, 3.02s elapsed (2 total hosts)  
Nmap scan report for 62078 (0.0.242.126) [host down]  
Nmap scan report for 1742753279 (103.224.77.255) [host down]  
Initiating ARP Ping Scan at 21:31  
Scanning 192.168.1.13 [1 port]  
Completed ARP Ping Scan at 21:31, 1.42s elapsed (1 total hosts)  
Nmap scan report for 192.168.1.13 [host down]  
Read data files from: /usr/bin/./share/nmap  
Nmap done: 3 IP addresses (0 hosts up) scanned in 4.93 seconds  
Raw packets sent: 18 (664B) | Rcvd: 0 (0B)  
root@debian1210:/tmp/sshprank# cat res.txt  
# Nmap 7.93 scan initiated Sun Mar 23 21:31:33 2025 as: nmap -iL masscan.txt -sV  
-p 1-65535 -v -oN res.txt  
Failed to resolve "open".  
Failed to resolve "tcp".  
Nmap scan report for 62078 (0.0.242.126) [host down]
```

```
pavel@debian1210: ~  
Initiating Ping Scan at 21:31  
Scanning 2 hosts [4 ports/host]  
Completed Ping Scan at 21:31, 3.02s elapsed (2 total hosts)  
Nmap scan report for 62078 (0.0.242.126) [host down]  
Nmap scan report for 1742753279 (103.224.77.255) [host down]  
Initiating ARP Ping Scan at 21:31  
Scanning 192.168.1.13 [1 port]  
Completed ARP Ping Scan at 21:31, 1.42s elapsed (1 total hosts)  
Nmap scan report for 192.168.1.13 [host down]  
Read data files from: /usr/bin/./share/nmap  
Nmap done: 3 IP addresses (0 hosts up) scanned in 4.93 seconds  
Raw packets sent: 18 (664B) | Rcvd: 0 (0B)  
root@debian1210:/tmp/sshprank# cat res.txt  
# Nmap 7.93 scan initiated Sun Mar 23 21:31:33 2025 as: nmap -iL masscan.txt -sV  
-p 1-65535 -v -oN res.txt  
Failed to resolve "open".  
Failed to resolve "tcp".  
Nmap scan report for 62078 (0.0.242.126) [host down]  
Nmap scan report for 1742753279 (103.224.77.255) [host down]  
Nmap scan report for 192.168.1.13 [host down]  
Read data files from: /usr/bin/./share/nmap  
# Nmap done at Sun Mar 23 21:31:38 2025 -- 3 IP addresses (0 hosts up) scanned in 4.93 seconds  
root@debian1210:/tmp/sshprank#
```

Так же в качестве примера выполним полуоткрытое сканирование (-sS)

```
pavel@debian1210: ~
[+] exiting receive thread #0 found=1
[+] all threads have exited
root@debian1210:/tmp/sshprank# masscan -v -sS --range=192.168.1.1-192.168.1.20 -
> 8080
[-] pcap: failed to load: libpcap.so
[-] pcap: failed to load: libpcap.A.dylib
[-] pcap: failed to load: libpcap.dylib
[-] pcap: failed to load: libpcap.so.0.9.5
[-] pcap: failed to load: libpcap.so.0.9.4
[+] pcap: found library: libpcap.so.0.8
[+] interface = enp0s8
[+] if(enp0s8): pcap: libpcap version 1.10.3 (with TPACKET_V3)
[+] if(enp0s8): successfully opened
[+] interface-type = 1
if:enp0s8: type=ethernet(1)
[+] source-mac = 08-00-27-00-b1-bd
[+] source-ip = 192.168.1.16
[+] router-ip = 192.168.1.1
[+] arp: 192.168.1.1 == 78-44-76-a1-0b-d4
[+] router-mac-ipv4 = 78-44-76-a1-0b-d4
Starting masscan 1.3.2 (http://bit.ly/14GZzcT) at 2025-03-23 19:22:48 GMT
Initiating SYN Stealth Scan
Scanning 20 hosts [1 port/host]
[+] starting receive thread #0
```

```
pavel@debian1210: ~
[+] pcap: found library: libpcap.so.0.8
[+] interface = enp0s8
[+] if(enp0s8): pcap: libpcap version 1.10.3 (with TPACKET_V3)
[+] if(enp0s8): successfully opened
[+] interface-type = 1
if:enp0s8: type=ethernet(1)
[+] source-mac = 08-00-27-00-b1-bd
[+] source-ip = 192.168.1.16
[+] router-ip = 192.168.1.1
[+] arp: 192.168.1.1 == 78-44-76-a1-0b-d4
[+] router-mac-ipv4 = 78-44-76-a1-0b-d4
Starting masscan 1.3.2 (http://bit.ly/14GZzcT) at 2025-03-23 19:22:48 GMT
Initiating SYN Stealth Scan
Scanning 20 hosts [1 port/host]
[+] starting receive thread #0
[+] starting transmit thread #0
[+] starting throttler: rate = 100.00-pps
[+] waiting for threads to finish
Discovered open port 8080/tcp on 192.168.1.14
[+] transmit thread #0 complete
[+] exiting transmit thread #0 und=1
[+] exiting receive thread #0 found=1
[+] all threads have exited
root@debian1210:/tmp/sshprank# █
```

Так же можно можно это результат записать в файл


```

root@debian1210:/tmp/sshprank# masscan -v -sS --range=192.168.1.1-192.168.1.20 -
p 8080 --output-filename qwe.txt
[-] pcap: failed to load: libpcap.so
[-] pcap: failed to load: libpcap.A.dylib
[-] pcap: failed to load: libpcap.dylib
[-] pcap: failed to load: libpcap.so.0.9.5
[-] pcap: failed to load: libpcap.so.0.9.4
[+] pcap: found library: libpcap.so.0.8
[+] interface = enp0s8
[+] if(enp0s8): pcap: libpcap version 1.10.3 (with TPACKET_V3)
[+] if(enp0s8): successfully opened
[+] interface-type = 1
if:enp0s8: type=ethernet(1)
[+] source-mac = 08-00-27-00-b1-bd
[+] source-ip = 192.168.1.16
[+] router-ip = 192.168.1.1
[+] arp: 192.168.1.1 == 78-44-76-a1-0b-d4
[+] router-mac-ipv4 = 78-44-76-a1-0b-d4
Starting masscan 1.3.2 (http://bit.ly/14GZzcT) at 2025-03-23 19:28:20 GMT
Initiating SYN Stealth Scan

```

```

pavel@debian1210: ~
[+] starting throttler: rate = 100.00-pps
[+] waiting for threads to finish
[+] transmit thread #0 complete673:16:31 remaining, found=0
[+] exiting transmit thread #0 und=1
[+] exiting receive thread #0 found=1
[+] all threads have exited
root@debian1210:/tmp/sshprank# ls
docs      lists      paused.conf  README.md  result.txt  sshprank.py
hosts.txt masscan.txt qwe.txt     res.txt   ssh_info.txt
root@debian1210:/tmp/sshprank# cat qwe.txt
<?xml version="1.0"?>
<!-- masscan v1.0 scan -->
<nmaprun scanner="masscan" start="1742758100" version="1.0-BETA" xmloutputversi
on="1.03">
<scaninfo type="syn" protocol="tcp" />
<host endtime="1742758100"><address addr="192.168.1.14" addrtype="ipv4"/><ports>
<port protocol="tcp" portid="8080"><state state="open" reason="syn-ack" reason_t
tl="64"/></port></ports></host>
<runstats>
<finished time="1742758117" timestr="2025-03-23 22:28:37" elapsed="17" />
<hosts up="1" down="0" total="1" />
</runstats>
</nmaprun>
root@debian1210:/tmp/sshprank#

```

Так же можем указать исходный IP-адрес

```

root@debian1210:/tmp/sshprank# masscan -p 1-65535 --rate=2500 192.168.1.13 -v --
source-ip=192.168.1.16


```

И ещё пару команд. -I ключ позволяет брать диапазон адресов из файла. А excludefile позволяет исключить диапазон адресов из файла.

```
pavel@debian1210: ~  
root@debian1210:/tmp/sshprank# masscan -iL hosts10.txt -p 1-65535 --rate=10000 -  
-excludefile hosts11.txt -v  
[-] pcap: failed to load: libpcap.so  
[-] pcap: failed to load: libpcap.A.dylib  
[-] pcap: failed to load: libpcap.dylib  
[-] pcap: failed to load: libpcap.so.0.9.5  
[-] pcap: failed to load: libpcap.so.0.9.4  
[+] pcap: found library: libpcap.so.0.8  
[+] interface = enp0s8  
[+] if(enp0s8): pcap: libpcap version 1.10.3 (with TPACKET_V3)  
[+] if(enp0s8): successfully opened  
[+] interface-type = 1  
if:enp0s8: type=ethernet(1)  
[+] source-mac = 08-00-27-00-b1-bd  
[+] source-ip = 192.168.1.16  
[+] router-ip = 192.168.1.1  
[+] arp: 192.168.1.1 == 78-44-76-a1-0b-d4  
[+] router-mac-ipv4 = 78-44-76-a1-0b-d4  
Starting masscan 1.3.2 (http://bit.ly/14GZzcT) at 2025-03-23 19:52:14 GMT  
Initiating SYN Stealth Scan  
Scanning 4 hosts [65535 ports/host]  
[+] starting transmit thread #0  
[+] starting throttler: rate = 10000.00-pps  
[+] starting receive thread #0
```

Ответы на вопросы:

1. Что такое SSHPrank?

 **SSHprank** - это инструмент для тестирования на проникновение в систему с использованием протокола SSH. Он может выполнять различные атаки и тестировать уязвимости в SSH-серверах. Это инструмент, который может использоваться разработчиками и администраторами для обнаружения и устранения уязвимостей в SSH-серверах. **Однако обратите внимание, что использование этого инструмента для атаки на несанкционированные системы может быть незаконным.**

2. Какие основные функции SSHPrank?

sshprank позволяет быстро проверять множество учётных записей на наличие уязвимостей, что делает его полезным для тестирования безопасности систем, быстрого сканирования и обнаружения SSH-серверов, что помогает в выявлении потенциальных целей для тестирования.

3. Как установить SSHPrank на Linux?

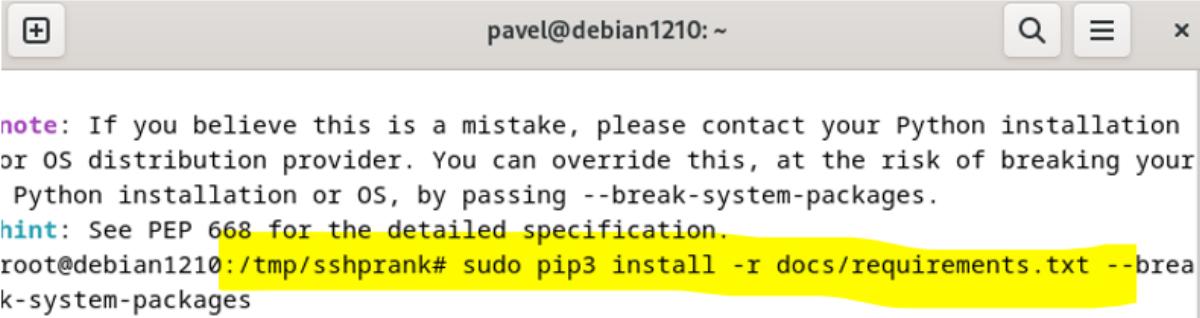
Для этого сначала выполним установку sshprank:

Первым делом установим менеджер пакетов PIP для Python 3

Sudo apt install python3-pip

```
root@debian1210:/tmp# git clone https://github.com/noptrix/sshprank
```

Далее установим masscan, shodan, paramiko



A terminal window titled 'pavel@debian1210: ~' with search, menu, and close buttons. It displays a note about Python installation, a hint to see PEP 668, and the command to install dependencies. The command is highlighted in yellow.

```
note: If you believe this is a mistake, please contact your Python installation
or OS distribution provider. You can override this, at the risk of breaking your
Python installation or OS, by passing --break-system-packages.
hint: See PEP 668 for the detailed specification.
root@debian1210:/tmp/sshprank# sudo pip3 install -r docs/requirements.txt --brea
k-system-packages
```

4. Как создать файл конфигурации для SSHPrank?

Файл конфигурации создаётся по умолчанию при клонировании репозитория. Чтобы его редактировать необходимо ввести: `nano sshprank.py`

5. Как запустить SSHPrank в интерактивном режиме?

По умолчанию программа не выводит никакой информации, даже не показывает взломанные хосты, логины и пароли. Чтобы эта информация выводилась на экран, используйте опцию `-v`:

$$\frac{(\overline{\alpha_1} \dots \overline{\alpha_n})}{\overline{\alpha_1} \dots \overline{\alpha_n}} \cdot \frac{(\overline{\beta_1} \dots \overline{\beta_m})}{\overline{\beta_1} \dots \overline{\beta_m}} = \frac{(\overline{\alpha_1} \dots \overline{\alpha_n}, \overline{\beta_1} \dots \overline{\beta_m})}{\overline{\alpha_1} \dots \overline{\alpha_n}, \overline{\beta_1} \dots \overline{\beta_m}}$$

```
[+] game started
[+] scanning and cracking random targets
[/] scanning sshds
[+] game over
```

--password: Указывает пароль или файл со списком паролей для попыток входа.

--help: Показывает справку по использованию SSHPrank и доступным параметрам.

10. Какие риски существуют при использовании SSHPrank для атаки на сервер S

1. Правовые последствия

Атака на сервер SSH без разрешения является незаконной и может привести к уголовной ответственности. Это может включать штрафы и тюремное заключение, особенно если атака приводит к ущербу или утечке данных.

2. Обнаружение и блокировка

Многие серверы имеют системы обнаружения вторжений (IDS) и системы предотвращения вторжений (IPS), которые могут зафиксировать подозрительную активность, такую как множественные неудачные попытки входа. Это может привести к блокировке IP-адреса атакующего и уведомлению администраторов.

3. Репутационные риски

Если атака будет обнаружена, это может негативно сказаться на репутации атакующего, особенно если он связан с организацией или сообществом. Это может привести к потере доверия со стороны коллег и партнеров.

Вывод: в результате выполнения практической работы были получены и опробованы навыки работы с sshprank, masscan, pwdlogy и crunch. Так же частично поработали с nmap.

