Bachelor's degree in Computer Science and Engineering
Mobile devices security
2022-2023

Practical Case 3

"RAM Memory Analysis"

Luis Daniel Casais Mezquida - 100429021

Professor:
Pedro Peris López

1 Image extraction

After downloading Volatile 2.6.1 from the source repository, I copied the `i9100-CM_3.0.64.zip` file into the `volatile/plugins/overlays/linux/` folder, and executed the following command:

```
$ sudo python2 vol.py -f i9100-CM.bin --profile=Linuxi9100-CM_3_0_64ARM
linux_recover_filesystem --dump-dir ./output
```

Producing the complete image of the device.

2 Hash extraction

In Android devices, the password is stored in the file `/data/system/password.key`, in this case producing the following (concatenated) hashes:

- **SHA1:** `a66a4a34a78aec1a7058c8fa3bb3b0f1cc537dd0`
- **MD5:** `42f0f3f909f87d0706dcf139ab37f86e`

The salt for the hash can be located in the `/data/system/locksettings.db` database. We can access the database with an SQL viewer, for example:

```
$ sqlite3 locksettings.db
```

By using a query we can obtain the salt:

```
sqlite> SELECT * FROM locksettings WHERE name='lockscreen.password_salt';
```

Which returns the number `-6140990771726895285` (in decimal), which translates to `aac6d16df244374b` in hexadecimal (signed 2's complement).

3 Password cracking

I used Hashcat in order to crack this password. As we know the format of the password, we'll use a mask attack, assuming the "X"s can be any ascii character. As MD5 is faster than SHA1, I decided to use that format to crack it.

The attack would therefore be:

```
$ hashcat -m 10 42f0f3f909f87d0706dcf139ab37f86e:aac6d16df244374b
-a 3 'INS{?a?a?a?a?a}' -o out.out
```

The resulting cracked password was `INS{t1MmY}`.