

## BÁO CÁO THỰC HÀNH

Môn học: Pháp chứng kỹ thuật số

Lab 5: Mobile Forensics

GVHD: Đoàn Minh Trung

### 1. THÔNG TIN CHUNG:

(Liệt kê tất cả các thành viên trong nhóm)

Lớp: NT334.P11.ANTT.1

Nhóm: N03

STT	Họ và tên	MSSV	Email
1	Lê Huy Hiệp	21522067	<a href="mailto:21522067@gm.uit.edu.vn">21522067@gm.uit.edu.vn</a>
2	Nguyễn Trần Duy Anh	20520393	<a href="mailto:20520393@gm.uit.edu.vn">20520393@gm.uit.edu.vn</a>

### 2. NỘI DUNG THỰC HIỆN:<sup>1</sup>

STT	Công việc	Kết quả tự đánh giá
1	Kịch bản 0 (Đã làm trên lớp)	100%
2	Kịch bản 1 (Trên lớp làm được một nửa)	100%
3	Kịch bản 2 (Đã làm trên lớp)	100%
4	Kịch bản 3 (Đã làm trên lớp)	100%
5	Kịch bản 4	100%

Phần bên dưới của báo cáo này là tài liệu báo cáo chi tiết của nhóm thực hiện.

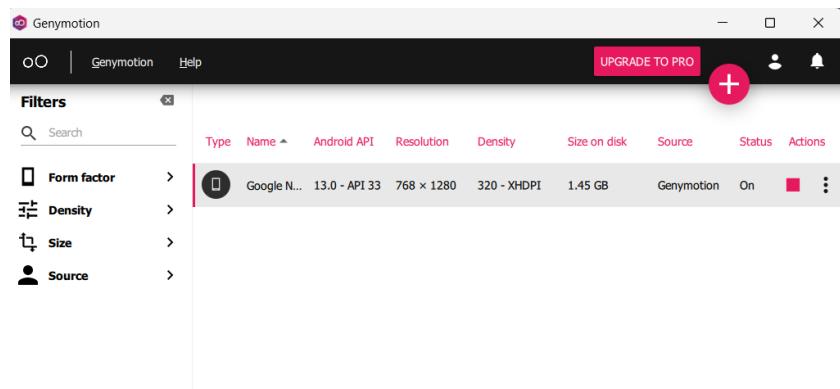
#### Nội dung

Kịch bản 0.....	1
Kịch bản 1.....	9
Kịch bản 2.....	20
Kịch bản 3.....	22
Kịch bản 4.....	26
YÊU CẦU CHUNG.....	36

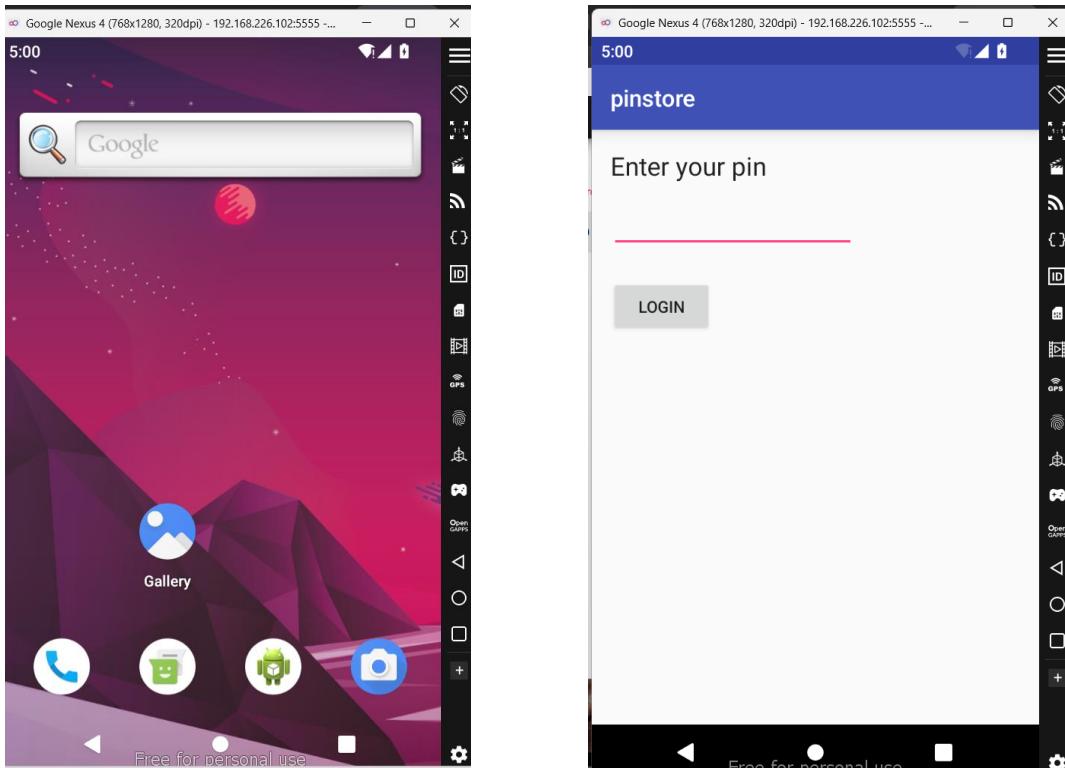
<sup>1</sup> Ghi nội dung công việc, các kịch bản trong bài Thực hành

## Kịch bản 0

Sử dụng Genymotion để tạo thiết bị



Đưa app đã tải về vào và chạy:



Kiểm tra thiết bị đã kết nối bằng lệnh .\adb.exe devices

```
PS D:\DataApp\Genymotion\tools> .\adb.exe devices
List of devices attached
192.168.226.102:5555    device

PS D:\DataApp\Genymotion\tools> |
```

Do Android được phát triển trên nhân Linux do đó, có thể thực hiện các lệnh cơ bản như trọng các hệ điều hành Linux

## Lab 5: Mobile Forensic

```
PS D:\DataApp\Genymotion\tools> .\adb.exe shell
vbox86p:/ $ ls -l
total 72
drwxr-xr-x  2 root  root      4096 2009-01-01 00:00 acct
drwxr-xr-x  27 root  root      560 2024-11-14 04:58 apex
lrw-r--r--  1 root  root      11 2009-01-01 00:00 bin -> /system/bin
lrw-r--r--  1 root  root      50 2009-01-01 00:00 bugreports -> /data/user_de/0/com.android.shell/files/bugreports
drwxrwx---  6 system  cache    4096 2024-11-14 04:58 cache
drwxr-xr-x  3 root  root      0 2024-11-14 04:58 config
lrw-r--r--  1 root  root     17 2009-01-01 00:00 d -> /sys/kernel/debug
drwxrwxr-x  50 system  system   4096 2024-11-14 04:58 data
drwx-----  8 root  system     160 2024-11-14 04:58 data_mirror
drwxr-xr-x  2 root  root     4096 2009-01-01 00:00 debug_ramdisk
drwxr-xr-x  25 root  root     3000 2024-11-14 04:58 dev
lrw-r--r--  1 root  root     11 2009-01-01 00:00 etc -> /system/etc
lrw-r--r--  1 root  shell     16 2009-01-01 00:00 init -> /system/bin/init
-rw-r--r--  1 root  shell     463 2009-01-01 00:00 init.environ.rc
drwxr-xr-x  11 root  root     260 2024-11-14 04:58 linkerconfig
drwx-----  2 root  root     16384 2009-01-01 00:00 lost+found
drwxr-xr-x  16 root  system    340 2024-11-14 04:58 mnt
drwxr-xr-x  2 root  root     4096 2009-01-01 00:00 odm
drwxr-xr-x  2 root  root     4096 2009-01-01 00:00 odm_dlkm
drwxr-xr-x  2 root  root     4096 2009-01-01 00:00 oem
drwxr-xr-x  2 root  root     4096 2009-01-01 00:00 postinstall
dr-xr-xr-x  265 root  root      0 2024-11-14 04:58 proc
lrw-r--r--  1 root  root     15 2009-01-01 00:00 product -> /system/product
lrw-r--r--  1 root  root     11 2009-01-01 00:00 sbin -> /system/bin
lrw-r--r--  1 root  root     21 2009-01-01 00:00 sdcard -> /storage/self/primary
drwxr-xr-x  2 root  root     4096 2009-01-01 00:00 second_stage_resources
drwx--- 4 shell  everybody   80 2024-11-14 04:58 storage
```

### Trích xuất dữ liệu trên android:

Xem các ứng dụng hệ thống/ các ứng dụng cài sẵn trong thư mục app:

```
vbox86p:/system/app # ls
Amaze           CarrierDefaultApp   EasterEgg          NfcNci           SimAppDialog
BasicDreams     CertInstaller       ExtShared         OsuLogin          Superuser
Bluetooth       CompanionDeviceManager GenyDService     PacProcessor      SystemPatcher
BluetoothMidiService CtsShimPrebuilt GenymotionLayout PrintRecommendationService Traceur
BookmarkProvider CubeLiveWallpapers HTMLViewer        PrintSpooler      WAPPushManager
BuiltInPrintService CustomLocale     KeyChain         SecureElement    WallpaperBackup
CaptivePortalLogin DevelopmentSettings LiveWallpapersPicker SettingsService messaging
vbox86p:/system/app # |
```

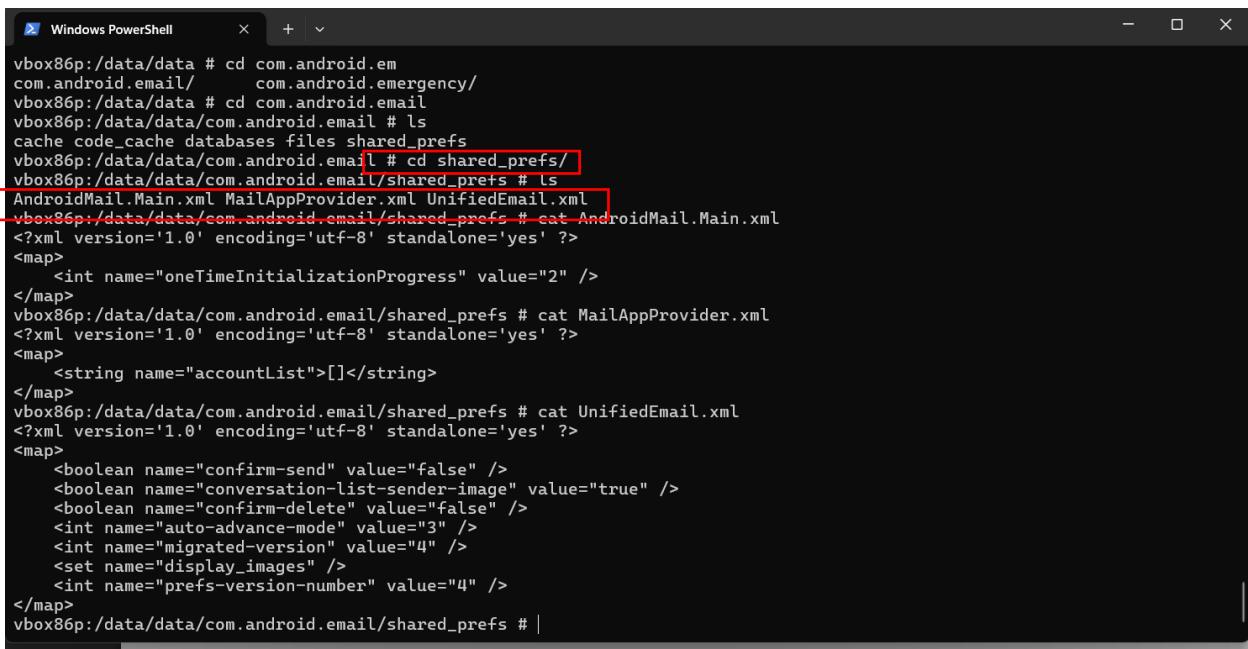
Xem dữ liệu ứng dụng nằm trong thư mục /data/data cần sử dụng quyền root

```
PS C:\Program Files\Genymobile\Genymotion\tools> .\adb root
PS C:\Program Files\Genymobile\Genymotion\tools> .\adb shell
vbox86p:/ # cd data/data
vbox86p:/data/data # ls
android                               com.android.printservice.recommendation
android.ext.services                   com.android.printspooler
android.ext.shared                     com.android.providers.blockednumber
com.amaze.filemanager                 com.android.providers.calendar
com.android.backupconfirm              com.android.providers.contacts
com.android.bips                       com.android.providers.downloads
com.android.bluetooth                  com.android.providers.downloads.ui
com.android.bluetoothmidiservice     com.android.providers.media
com.android.bookmarkprovider         com.android.providers.settings
com.android.calendar                  com.android.providers.telephony
com.android.callogbackup              com.android.providers.userdictionary
com.android.camera2                   com.android.provider
com.android.captiveportallogin       com.android.proxyhandler
com.android.carrierconfig             com.android.quicksearchbox
com.android.carrierdefaultapp        com.android.se
com.android.cellbroadcastreceiver    com.android.server.telecom
com.android.certinstaller            com.android.settings
com.android.companiondevicemanager   com.android.settings.intelligence
com.android.contacts                  com.android.sharedstoragebackup
com.android.cts.ctsshim               com.android.shell
com.android.cts.priv.ctsshim          com.android.simappdialog
com.android.customlocale2            com.android.smspush
com.android.deskclock                com.android.statementservice
```

Xem các giá trị tham chiếu quan trọng thì tìm trong thư mục của nó và được lưu dưới dạng tập tin .xml: /data/data/shared\_prefs

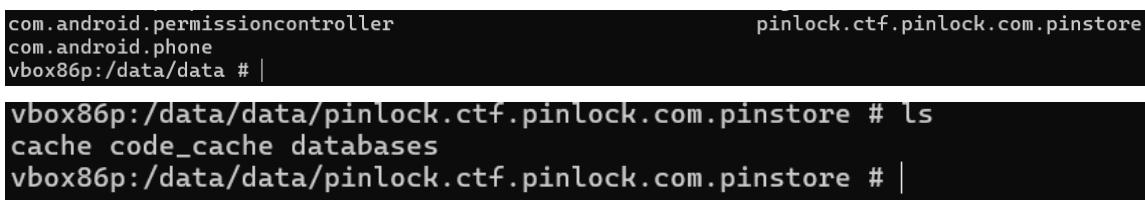
Xem ứng dụng email: /data/data/com.android.email

## Lab 5: Mobile Forensic



```
vbox86p:/data/data # cd com.android.email
com.android.email/
com.android.emergency/
vbox86p:/data/data # cd com.android.email
vbox86p:/data/data/com.android.email # ls
cache code_cache databases files shared_prefs
vbox86p:/data/data/com.android.email # cd shared_prefs/
vbox86p:/data/data/com.android.email/shared_prefs # ls
AndroidMail.Main.xml MailAppProvider.xml UnifiedEmail.xml
vbox86p:/data/data/com.android.email/shared_prefs # cat AndroidMail.Main.xml
<?xml version='1.0' encoding='utf-8' standalone='yes' ?>
<map>
    <int name="oneTimeInitializationProgress" value="2" />
</map>
vbox86p:/data/data/com.android.email/shared_prefs # cat MailAppProvider.xml
<?xml version='1.0' encoding='utf-8' standalone='yes' ?>
<map>
    <string name="accountList">[]</string>
</map>
vbox86p:/data/data/com.android.email/shared_prefs # cat UnifiedEmail.xml
<?xml version='1.0' encoding='utf-8' standalone='yes' ?>
<map>
    <boolean name="confirm-send" value="false" />
    <boolean name="conversation-list-sender-image" value="true" />
    <boolean name="confirm-delete" value="false" />
    <int name="auto-advance-mode" value="3" />
    <int name="migrated-version" value="4" />
    <set name="display_images" />
    <int name="prefs-version-number" value="4" />
</map>
vbox86p:/data/data/com.android.email/shared_prefs # |
```

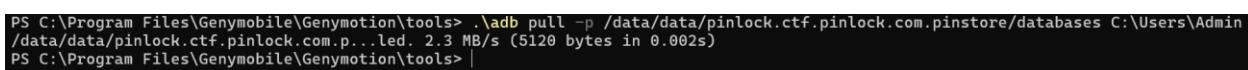
Xem dữ liệu cục bộ của ứng dụng:



```
com.android.permissioncontroller
com.android.phone
vbox86p:/data/data # |

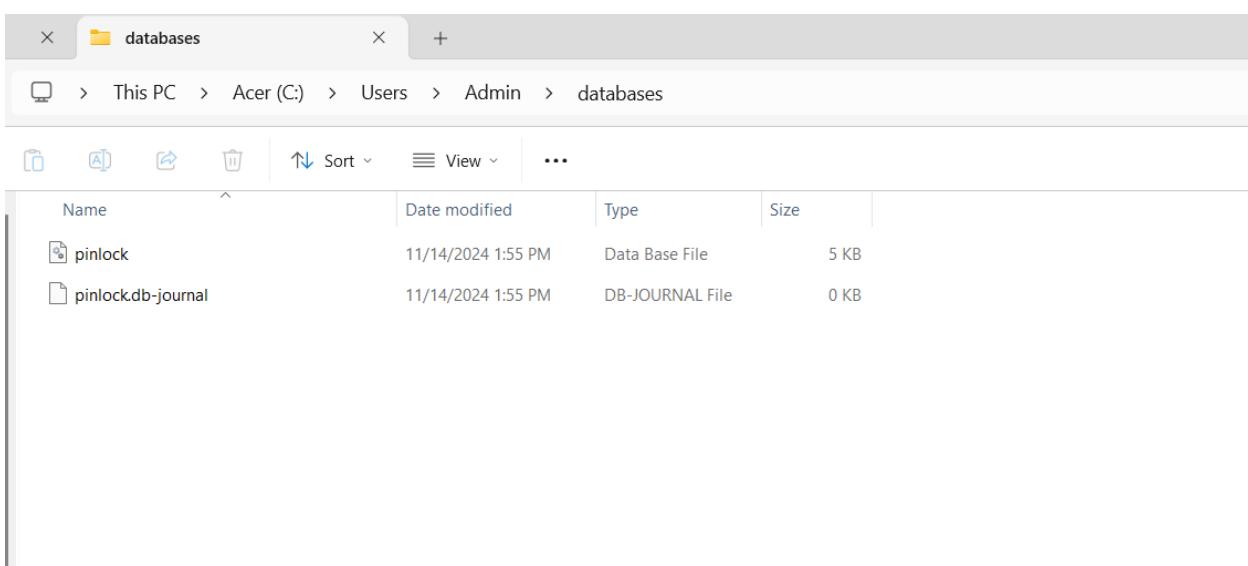
vbox86p:/data/data/pinlock.ctf.pinlock.com.pinstore # ls
cache code_cache databases
vbox86p:/data/data/pinlock.ctf.pinlock.com.pinstore # |
```

Tải dữ liệu từ thiết bị di động về để điều tra:



```
PS C:\Program Files\Genymobile\Genymotion\tools> .\adb pull -p /data/data/pinlock.ctf.pinlock.com.pinstore/databases C:\Users\Admin
\data\data/pinlock.ctf.pinlock.com.p...led. 2.3 MB/s (5120 bytes in 0.002s)
PS C:\Program Files\Genymobile\Genymotion\tools> |
```

Kết quả:



Ý nghĩa của câu lệnh:

[adb pull -p data/data/ \Cases\Case\\_003](#)

Câu lệnh này sử dụng adb: Đây là viết tắt của **Android Debug Bridge** - một công cụ dòng lệnh để giao tiếp với thiết bị Android từ máy tính với tùy chọn **pull** là để tải dữ liệu từ Android về máy tính, tùy chọn **-p** hiển thị chi tiết về tiến trình sao chép, **data/data/** là đường dẫn của thư mục muốn sao chép và **\Cases\Case\_003** là đường dẫn trên máy tính mà ta sao chép dữ liệu vào.

**Tìm hiểu ADB Dumpsys (không đòi hỏi phải có quyền root), để xem các dịch vụ đang chạy trên máy:**

```
PS C:\Program Files\Genymobile\Genymotion\tools> .\adb shell service list
Found 164 services:
 0     gsiservice: []
 1     ions: [com.android.internal.telephony.IOns]
 2     sip: [android.net.sip.ISipService]
 3     carrier_config: [com.android.internal.telephony.ICarrierConfigLoader]
 4     phone: [com.android.internal.telephony.ITelephony]
 5     isms: [com.android.internal.telephony.ISms]
 6     iphonesubinfo: [com.android.internal.telephony.IPhoneSubInfo]
 7     simphonebook: [com.android.internal.telephony.IIccPhoneBook]
 8     ircs: [android.telephony.ims.aidl.IRcs]
 9     isub: [com.android.internal.telephony.ISub]
10     secure_element: [android.se.omapi.ISecureElementService]
11     SystemPatcher: [com.genymotion.systempatcher.ISystemPatcher]
12     telecom: [com.android.internal.telecom.ITelecomService]
13     network_stack: [android.net.INetworkStackConnector]
14     contexthub: [android.hardware.location.IContextHubService]
15     netd_listener: [android.net.metrics.INetdEventListener]
16     connmetrics: [android.net.IIpConnectivityMetrics]
17     bluetooth_manager: [android.bluetooth.IBluetoothManager]
18     app_binding: []
19     clipboard: [android.content.IClipboard]
20     autofill: [android.view.autofill.IAutoFillManager]
21     imms: [com.android.internal.telephony.IMms]
22     incidentcompanion: [android.os.IIncidentCompanion]
23     statscompanion: [android.os.IStatsCompanionService]
24     media.camera.proxy: [android.hardware.ICameraServiceProxy]
25     slice: [android.app.slice.ISliceManager]
26     media_projection: [android.media.projection.IMediaProjectionManager]
27     crossprofileapps: [android.content.pm.ICrossProfileApps]
28     launcherapps: [android.content.pm.ILauncherApps]
29     shortcut: [android.content.pm.IShortcutService]
30     media_router: [android.media.IMediaRouterService]
31     media_resource_monitor: [android.media.IMediaResourceMonitor]
```

Xem thông tin batterystats

## Lab 5: Mobile Forensic

```
PS C:\Program Files\Genymobile\Genymotion\tools> .\adb shell dumpsys battery stats
Battery History (0% used, 0 used of 4096KB, 0 strings using 0):

Per-PID Stats:
  PID 0 wake time: +3s702ms
  PID 684 wake time: +2m41s207ms
  PID 1158 wake time: +1ms
  PID 684 wake time: +3ms
  PID 1158 wake time: +3s299ms
  PID 684 wake time: 0
  PID 891 wake time: +2s909ms
  PID 684 wake time: 0
  PID 0 wake time: +24s680ms
  PID 684 wake time: 0
  PID 684 wake time: 0
  PID 684 wake time: +1ms
  PID 684 wake time: +3ms
  PID 684 wake time: +3ms
  PID 684 wake time: +1ms
  PID 684 wake time: 0
  PID 684 wake time: +1ms
  PID 684 wake time: +1ms
  PID 684 wake time: 0
  PID 684 wake time: +11ms
  PID 2011 wake time: +12ms
  PID 684 wake time: 0
  PID 684 wake time: 0
  PID 684 wake time: 0
  PID 684 wake time: +1ms
```

## Lab 5: Mobile Forensic

```

        (nothing executed)
u0a93:
    Wake lock *dexopt* realtime
u0a94:
    Wake lock *dexopt* realtime
u0a95:
    Wake lock *dexopt* realtime
u0a96:
    Wake lock *dexopt* realtime
u0a97:
    Wake lock *dexopt* realtime
    Wake lock AlarmAlertWakeLock realtime
u0a98:
    Wake lock *dexopt* realtime
u0a99:
    Wake lock *dexopt* realtime
    Apk com.android.inputmethod.latin:
        Service com.android.inputmethod.latin.LatinIME:
            Created for: 0ms uptime
            Starts: 0, launches: 1
u0a100:
    Wake lock *dexopt* realtime
    Apk com.android.email:
        (nothing executed)
u0a101:
    Wake lock *dexopt* realtime
    Wake lock *launch* realtime
    Wake lock WindowManager realtime

Total cpu time reads: 0
Batched cpu time reads: 0
Batching Duration (min): 56
All UID cpu time reads since the later of device start or stats reset: 20
UIDs removed since the later of device start or stats reset: 1
PS C:\Program Files\Genymobile\Genymotion\tools> |

```

Xem thông tin các dịch vụ khác:

Procstats

```

PS C:\Program Files\Genymobile\Genymotion\tools> .\adb shell dumpsys iphonesubinfo
PS C:\Program Files\Genymobile\Genymotion\tools> .\adb shell dumpsys procstats
CURRENT STATS:
* system / 1000 / v29:
    TOTAL: 100% (113MB-116MB-120MB/94MB-97MB-99MB/251MB-260MB-269MB over 3)
    Persistent: 100% (113MB-116MB-120MB/94MB-97MB-99MB/251MB-260MB-269MB over 3)
* com.android.bluetooth / 1002 / v29:
    TOTAL: 100% (17MB-18MB-18MB/13MB-13MB-13MB/104MB-106MB-107MB over 3)
    Persistent: 100% (17MB-18MB-18MB/13MB-13MB-13MB/104MB-106MB-107MB over 3)
* com.android.systemui / u0a84 / v29:
    TOTAL: 100% (41MB-53MB-86MB/27MB-41MB-75MB/130MB-180MB-216MB over 15)
    Persistent: 100% (41MB-53MB-86MB/27MB-41MB-75MB/130MB-180MB-216MB over 15)
* com.android.networkstack / 1073 / v290000000:
    TOTAL: 100% (6.8MB-15MB-27MB/3.7MB-11MB-23MB/88MB-100MB-115MB over 3)
    Persistent: 100% (6.8MB-15MB-27MB/3.7MB-11MB-23MB/88MB-100MB-115MB over 3)
* com.android.phone / 1001 / v29:
    TOTAL: 100% (32MB-43MB-51MB/26MB-37MB-45MB/127MB-140MB-148MB over 3)
    Persistent: 100% (32MB-43MB-51MB/26MB-37MB-45MB/127MB-140MB-148MB over 3)
* com.android.inputmethod.latin / u0a99 / v28:
    TOTAL: 100% (12MB-17MB-25MB/7.3MB-11MB-19MB/104MB-111MB-126MB over 3)
    Imp Fg: 0.53%
    Imp Bg: 99% (12MB-17MB-25MB/7.3MB-11MB-19MB/104MB-111MB-126MB over 3)
* android.ext.services / u0a38 / v290000000:
    TOTAL: 100% (6.3MB-6.8MB-7.2MB/3.3MB-3.6MB-3.7MB/81MB-98MB-106MB over 3)
    Imp Fg: 100% (6.3MB-6.8MB-7.2MB/3.3MB-3.6MB-3.7MB/81MB-98MB-106MB over 3)
* com.genymotion.settings / 1000 / v29:

```

Userappops:

```
PS C:\Program Files\Genymobile\Genymotion\tools> .\adb shell dumpsys userappops  
Can't find service: userappops  
PS C:\Program Files\Genymobile\Genymotion\tools> |
```

Wi-fi:

```
can't find service: userappops  
PS C:\Program Files\Genymobile\Genymotion\tools> .\adb shell dumpsys Wi-Fi  
Can't find service: Wi-Fi  
PS C:\Program Files\Genymobile\Genymotion\tools> DDD|
```

Notification:

```
PS C:\Program Files\Genymobile\Genymotion\tools> .\adb shell dumpsys notification  
Current Notification Manager state:  
    mUseAttentionLight=false  
    mHasLight=false  
    mNotificationPulseEnabled=true  
    mSoundNotificationKey=null  
    mVibrateNotificationKey=null  
    mDisableNotificationEffects=true  
    mCallState=CALL_STATE_IDLE  
    mSystemReady=true  
    mMaxPackageEnqueueRate=5.0  
    mArchive=Archive (1 notification)  
        StatusBarNotification(pkg=android user=UserHandle{0} id=13 tag=PreBootBroadcaster key=0|android|13|PreBootBroadca  
ster|1000: Notification(channel=UPDATES pri=0 contentView=null vibrate=null sound=null tick defaults=0x0 flags=0x2 co  
lor=0xff607d8b vis=PUBLIC)  
  
Snoozed notifications:  
  
Ranking Config:  
    mSignalExtractors.length = 11  
        NotificationChannelExtractor  
        NotificationAdjustmentExtractor  
        BubbleExtractor  
        ValidateNotificationPeople  
        PriorityExtractor  
        ZenModeExtractor  
        ImportanceExtractor  
        NotificationIntrusivenessExtractor  
        VisibilityExtractor
```

Dumpsys procstats: tình trạng sử dụng bộ xử lý của các ứng dụng đang chạy

## Lab 5: Mobile Forensic

```
PS C:\Program Files\Genymobile\Genymotion\tools> .\adb shell dumpsys procstats
CURRENT STATS:
* system / 1000 / v29:
    TOTAL: 100% (112MB-115MB-120MB/94MB-97MB-99MB/251MB-258MB-269MB over 4)
    Persistent: 100% (112MB-115MB-120MB/94MB-97MB-99MB/251MB-258MB-269MB over 4)
* com.android.bluetooth / 1002 / v29:
    TOTAL: 100% (17MB-18MB-18MB/13MB-13MB-13MB/104MB-106MB-107MB over 4)
    Persistent: 100% (17MB-18MB-18MB/13MB-13MB-13MB/104MB-106MB-107MB over 4)
* com.android.systemui / u0a84 / v29:
    TOTAL: 100% (41MB-52MB-86MB/27MB-41MB-75MB/130MB-179MB-216MB over 16)
    Persistent: 100% (41MB-52MB-86MB/27MB-41MB-75MB/130MB-179MB-216MB over 16)
* com.android.networkstack / 1073 / v2900000000:
    TOTAL: 100% (6.8MB-19MB-30MB/3.7MB-15MB-26MB/88MB-105MB-118MB over 4)
    Persistent: 100% (6.8MB-19MB-30MB/3.7MB-15MB-26MB/88MB-105MB-118MB over 4)
* com.android.phone / 1001 / v29:
    TOTAL: 100% (29MB-40MB-51MB/22MB-33MB-45MB/126MB-136MB-148MB over 4)
    Persistent: 100% (29MB-40MB-51MB/22MB-33MB-45MB/126MB-136MB-148MB over 4)
* com.android.inputmethod.latin / u0a99 / v28:
    TOTAL: 100% (12MB-19MB-26MB/7.3MB-13MB-19MB/104MB-115MB-126MB over 4)
    Imp Fg: 0.49%
    Imp Bg: 99% (12MB-19MB-26MB/7.3MB-13MB-19MB/104MB-115MB-126MB over 4)
* android.ext.services / u0a38 / v2900000000:
    TOTAL: 100% (6.3MB-7.1MB-7.9MB/3.3MB-3.6MB-3.7MB/81MB-100MB-106MB over 4)
    Imp Fg: 100% (6.3MB-7.1MB-7.9MB/3.3MB-3.6MB-3.7MB/81MB-100MB-106MB over 4)
* com.genymotion.settings / 1000 / v29:
    TOTAL: 100% (5.7MB-5.9MB-6.3MB/3.3MB-3.5MB-3.6MB/77MB-77MB-77MB over 4)
    Persistent: 100% (5.7MB-5.9MB-6.3MB/3.3MB-3.5MB-3.6MB/77MB-77MB-77MB over 4)
* com.genymotion.systempatcher / 1000 / v29:
    TOTAL: 100% (6.3MB-6.5MB-6.9MB/3.8MB-4.1MB-4.2MB/76MB-77MB-77MB over 4)
    Persistent: 100% (6.3MB-6.5MB-6.9MB/3.8MB-4.1MB-4.2MB/76MB-77MB-77MB over 4)
* com.android.se / 1068 / v29:
```

Dumpsys user: hiển thị thông tin người dùng đang sử dụng thiết bị

```
PS C:\Program Files\Genymobile\Genymotion\tools> .\adb shell dumpsys user
Users:
UserInfo{0:null:13} serialNo=0
State: RUNNING_UNLOCKED
Created: <unknown>
Last logged in: +1h3m28s836ms ago
Last logged in fingerprint: google/vbox86p/vbox86p:10/QQ1D.200105.002/715:userdebug/test-keys
Start time: +1h3m34s61ms ago
Unlock time: +1h3m31s959ms ago
Has profile owner: false
Restrictions:
    none
Device policy global restrictions:
    null
Device policy local restrictions:
    null
Effective restrictions:
    none

Device owner id:-10000

Guest restrictions:
    no_sms
    no_install_unknown_sources
    no_config_wifi
    no_outgoing_calls

Device managed: false
Started users state: {0=3}

Max users: 4
```

Dumpsys App Ops: thông tin về quyền hạn có thể truy cập bởi các ứng dụng

```
PS C:\Program Files\Genymobile\Genymotion\tools> .\adb shell dumpsys appops
Current AppOps Service state:
Settings:
    top_state_settle_time=+30s0ms
    fg_service_state_settle_time=+10s0ms
    bg_state_settle_time=+1s0ms

Op mode watchers:
    Op COARSE_LOCATION:
        #0: ModeCallback{f66db78 watchinguid=-1 flags=0x1 from uid=1000 pid=684}
    Op READ_CALL_LOG:
        #0: ModeCallback{2d3f992 watchinguid=-1 flags=0x0 from uid=1000 pid=684}
    Op WRITE_CALL_LOG:
        #0: ModeCallback{2d3f992 watchinguid=-1 flags=0x0 from uid=1000 pid=684}
    Op READ_SMS:
        #0: ModeCallback{2d3f992 watchinguid=-1 flags=0x0 from uid=1000 pid=684}
    Op RECEIVE_SMS:
        #0: ModeCallback{2d3f992 watchinguid=-1 flags=0x0 from uid=1000 pid=684}
    Op RECEIVE_MMS:
        #0: ModeCallback{2d3f992 watchinguid=-1 flags=0x0 from uid=1000 pid=684}
    Op RECEIVE_WAP_PUSH:
        #0: ModeCallback{2d3f992 watchinguid=-1 flags=0x0 from uid=1000 pid=684}
    Op SEND_SMS:
        #0: ModeCallback{2d3f992 watchinguid=-1 flags=0x0 from uid=1000 pid=684}
    Op SYSTEM_ALERT_WINDOW:
        #0: ModeCallback{8f83a6d watchinguid=-1 flags=0x0 from uid=1000 pid=684}
    Op PLAY_AUDIO:
        #0: ModeCallback{1edbe3e watchinguid=-1 flags=0x0 from uid=1041 pid=456}
    Op TOAST_WINDOW:
        #0: ModeCallback{8f83a6d watchinguid=-1 flags=0x0 from uid=1000 pid=684}
    Op PROCESS_OUTGOING_CALLS:
```

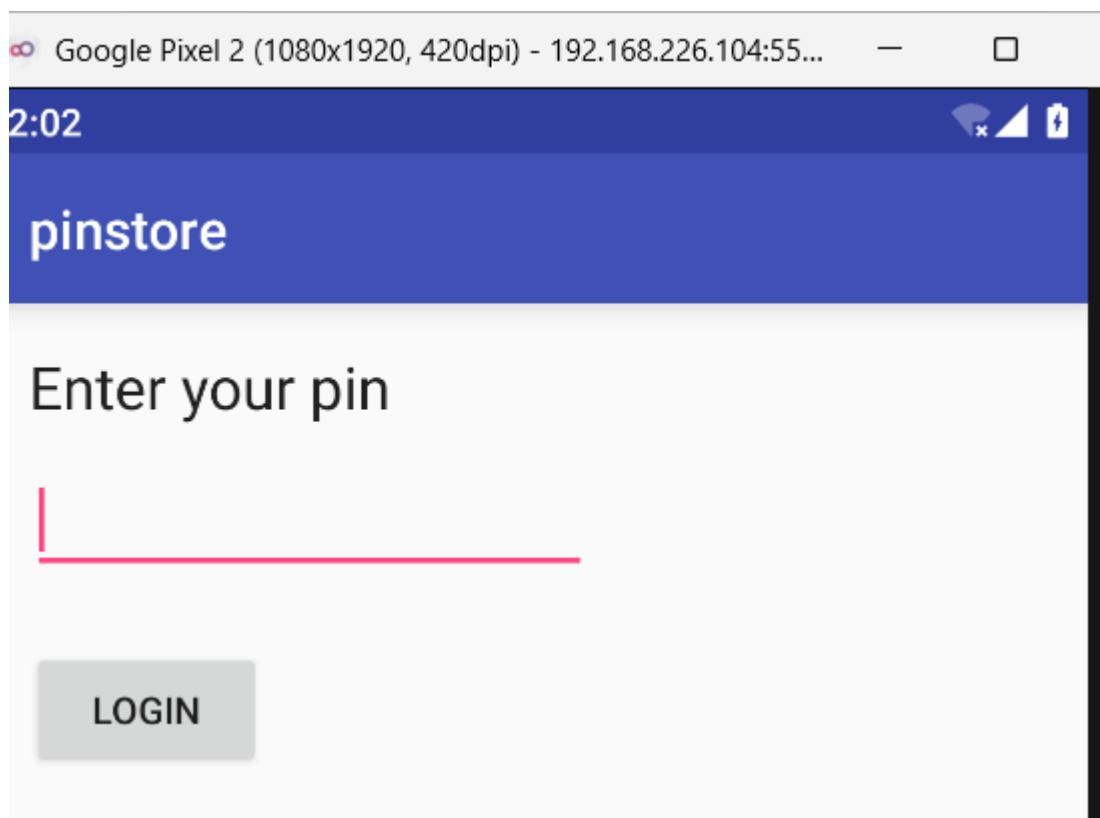
## Kịch bản 1

### 1. Kịch bản 01. Thực hiện phân tích ứng dụng Android

- Mô tả: Phân tích ứng dụng Android, tìm mã PIN trong ứng dụng để tìm flag.
- Tài nguyên thực hiện: pinstore.zip
- Yêu cầu – Gợi ý: Sử dụng các công cụ dịch ngược (decompile) trên mã nguồn Android để phân tích.

*Đáp án:*

Ứng dụng:

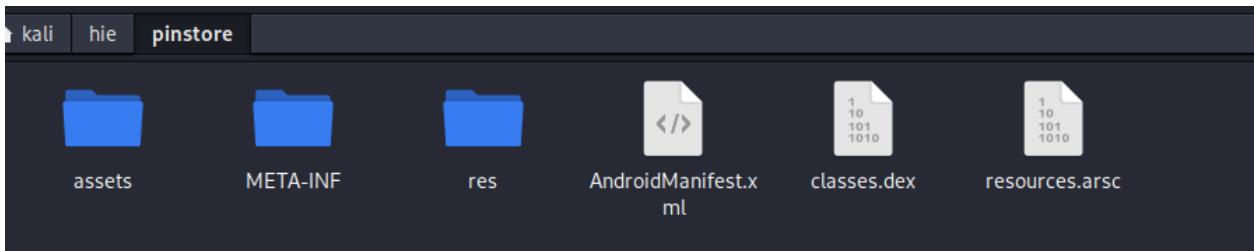


Giải nén tập tin .apk vào thư mục pinstore:

```
kali㉿kali:[~/hie]
$ unzip pinstore.apk -d ./pinstore

Archive: pinstore.apk
inflating: ./pinstore/AndroidManifest.xml
inflating: ./pinstore/assets/README
inflating: ./pinstore/assets/pinlock.db
inflating: ./pinstore/res/anim/abc_fade_in.xml
inflating: ./pinstore/res/anim/abc_fade_out.xml
inflating: ./pinstore/res/anim/abc_grow_fade_in_from_bottom.xml
inflating: ./pinstore/res/anim/abc_popup_enter.xml
inflating: ./pinstore/res/anim/abc_popup_exit.xml
inflating: ./pinstore/res/anim/abc_shrink_fade_out_from_bottom.xml
inflating: ./pinstore/res/anim/abc_slide_in_bottom.xml
inflating: ./pinstore/res/anim/abc_slide_in_top.xml
inflating: ./pinstore/res/anim/abc_slide_out_bottom.xml
inflating: ./pinstore/res/anim/abc_slide_out_top.xml
inflating: ./pinstore/res/color-v11/abc_background_cache_hint_selector_material_dark.xml
inflating: ./pinstore/res/color-v11/abc_background_cache_hint_selector_material_light.xml
inflating: ./pinstore/res/color-v23/abc_color_highlight_material.xml
inflating: ./pinstore/res/color/abc_primary_text_disable_only_material_dark.xml
inflating: ./pinstore/res/color/abc_primary_text_disable_only_material_light.xml
```

## Lab 5: Mobile Forensic

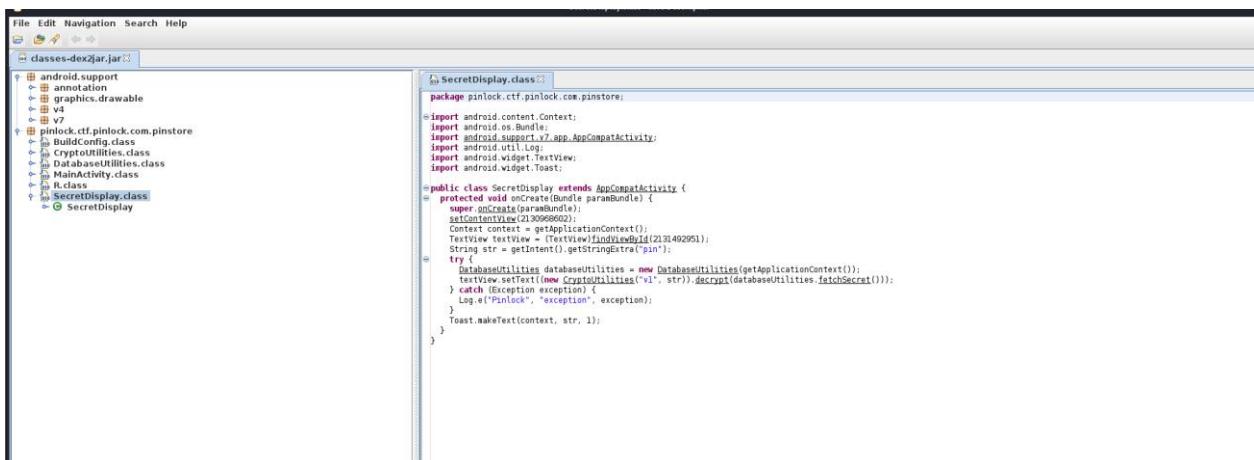


Tập tin classes.dex là tập tin **Dalvik Executable** chứa mã bytecode, là nơi lưu trữ tất cả các lớp Java và phương thức của ứng dụng được biên dịch

Sau khi thu được tập tin classes.dex thì chuyển đổi thành tập tin JAR.



Dùng JD-GUI để mở tập tin .jar, quan sát:



Xem MainActivity.class

## Lab 5: Mobile Forensic

```

public void onClick(View param1View) {
    String str1;
    Intent intent;
    String str3 = MainActivity.this.pinEditText.getText().toString();
    param1View = null;
    String str2 = null;
    try {
        String str = (new DatabaseUtilities(MainActivity.this.getApplicationContext())).fetchPin(
        str1 = str;
    } catch (IOException iOException) {
        iOException.printStackTrace();
    }
    try {
        String str = CryptoUtilities.getHash(str3);
        str2 = str;
        if (str1.equalsIgnoreCase(str2)) {
            intent = new Intent((Context)MainActivity.this, SecretDisplay.class);
            intent.putExtra("pin", str3);
            MainActivity.this.startActivity(intent);
            return;
        }
    } catch (NoSuchAlgorithmException noSuchAlgorithmException) {
        noSuchAlgorithmException.printStackTrace();
        if (intent.equalsIgnoreCase(str2)) {
            intent = new Intent((Context)MainActivity.this, SecretDisplay.class);
            intent.putExtra("pin", str3);
            MainActivity.this.startActivity(intent);
            return;
        }
    } catch (UnsupportedEncodingException unsupportedEncodingException) {
        unsupportedEncodingException.printStackTrace();
        if (intent.equalsIgnoreCase(str2)) {
            intent = new Intent((Context)MainActivity.this, SecretDisplay.class);
            intent.putExtra("pin", str3);
        }
    }
}

```

Thấy str3 chứa mã pin do người dùng nhập vào sau đó nó được hash và gán lại vào str2

Còn mã pin đã hash lấy từ CSDL thông qua phương thức fetchPin() được gán vào str1

Phương thức fetchPin() của class **DatabaseUtilities**

Xem tiếp **DatabaseUtilities.class**

```

package pinlock.ctf.pinlock.com.pinstore;

import android.content.Context;
import android.database.Cursor;
import android.database.sqlite.SQLiteDatabase;
import android.database.sqlite.SQLiteException;
import android.database.sqlite.SQLiteOpenHelper;
import android.util.Log;
import java.io.FileOutputStream;
import java.io.IOException;
import java.io.InputStream;

```

```

public class DatabaseUtilities extends SQLiteOpenHelper {
    private static String dbName;

```

```

    private static String pathToDB = "/data/data/pinlock.ctf.pinlock.com.pinstore/databases/";
    private final Context appcontext;
    private SQLiteDatabase db;

```

```

    static {
        dbName = "pinlock.db";
    }

```

```

    public DatabaseUtilities(Context paramContext) throws IOException {
        super(paramContext, dbName, null, 1);
        this.appcontext = paramContext;
        createDB();
    }

```

```

    public void close() {
        // Byte code:
        //   0: aload_0

```

Dữ liệu được lưu trong **pinlock.db**. Lớp này quản lý CSDL SQLite, CSDL SQLite được lưu dưới dạng tệp có phần mở rộng .db hoặc .sqlite. Dữ liệu trong SQLite được lưu trữ trong các bảng và có thể truy vấn bằng ngôn ngữ SQL.

Xem phương thức `fetchPin()`, mã pin được lưu trong bảng pinDB

```

public String fetchPin() throws IOException {
    openDB();
    Cursor cursor = this.db.rawQuery("SELECT pin FROM pinDB", null);
    String str = "";
    if (cursor.moveToFirst())
        str = cursor.getString(0);
    cursor.close();
    return str;
}

```

Xem thông tin:

```

root@kali: /home/kali/hie/pinstore/assets
kali@kali: ~/hie/pinstore/assets

sqlite3 pinlock.db
SQLite version 3.46.0 2024-05-23 13:25:27
Enter ".help" for usage hints.
sqlite> .table
android_metadata pinDB;           secretsDBv1      secretsDBv2
sqlite> SELECT * FROM pinDB;
1|d8531a519b3d4dfbece0259f90b466a23efc57b
sqlite> SELECT * FROM secretsDBv1;

```

Do hàm mã hóa mã pin do người dùng nhập vào là SHA-1 nên có thể đoán pin trong DB cũng băm từ SHA-1

```

public static String getHash(String paramString) throws NoSuchAlgorithmException, UnsupportedEncodingException {
    MessageDigest messageDigest;
    byte[] arrayOfByte = paramString.getBytes();
    paramString = null;
    try {
        MessageDigest messageDigest1 = MessageDigest.getInstance("SHA-1");
        messageDigest = messageDigest1;
    } catch (NoSuchAlgorithmException noSuchAlgorithmException) {}
    messageDigest.update(arrayOfByte, 0, arrayOfByte.length);
    return getHex(messageDigest.digest());
}

```

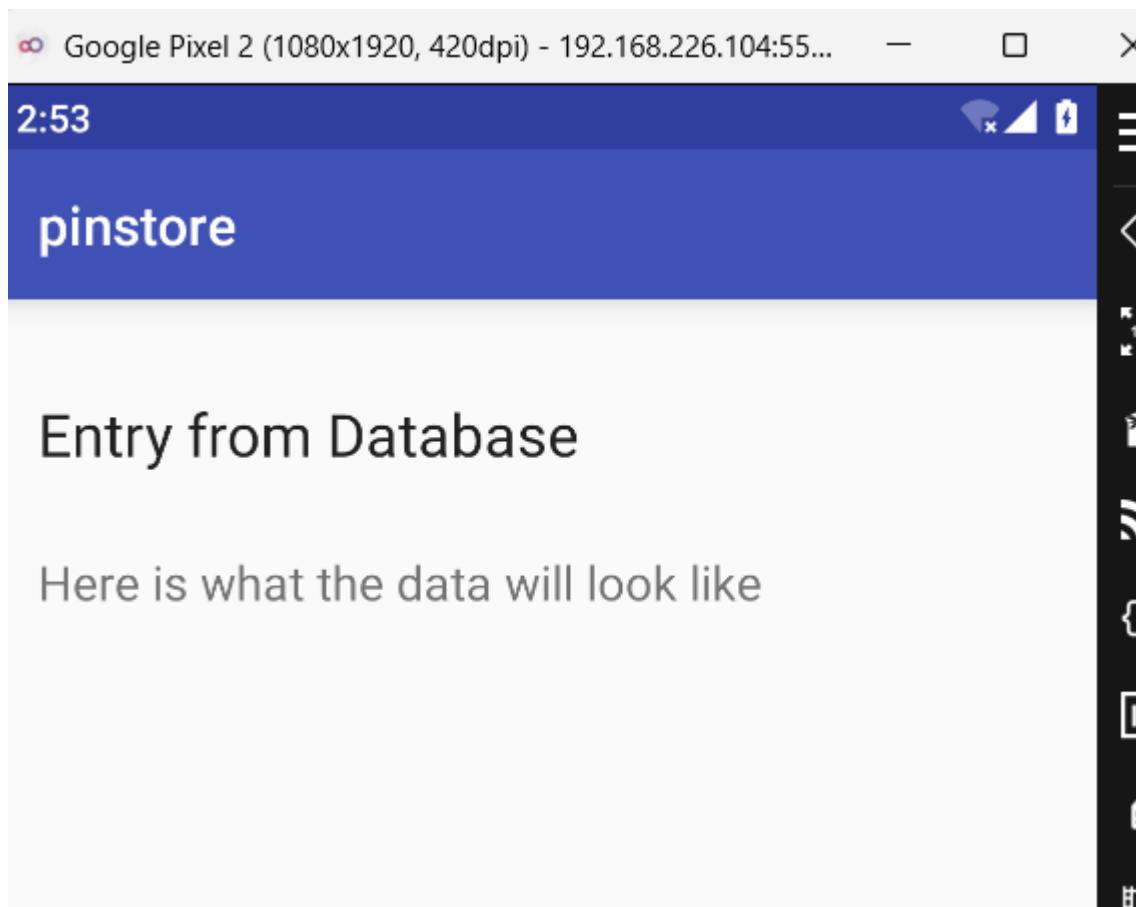
## Lab 5: Mobile Forensic

Thực hiện decode hash: [Decrypt MD5, SHA1, MySQL, NTLM, SHA256, MD5 Email, SHA256 Email, SHA512, Wordpress, Bcrypt hashes for free online](#)

The screenshot shows a web browser window with the URL <https://hashes.com/en/decrypt/hash>. The page displays a search result for the hash `d8531a519b3d4dfbece0259f90b466a23efc57b:7498`. A blue banner at the top says "Proceeded! 1 hashes were checked: 1 found 0 not found". Below this, a green box indicates "Found:" with the same hash value. There are buttons for "SEARCH AGAIN" and "CLEAR". At the bottom, there are sections for HASHES.COM (Support, API), DECRYPT HASHES (Free Search, Mass Search, Reverse Email MD5), TOOLS (Hash Identifier, Hash Verifier, Email Extractor, \*2john Hash Extractor, Hash Generator, File Parser, List Matching), and ESCROW (View jobs, Upload new list, Manage your lists).

Biết được mã pin là: **7498**

Nhập vào ứng dụng thấy kết quả:



Nhưng chưa tìm thấy flag

Tùy đoạn code trong SecretDisplay.class

```

public class SecretDisplay extends AppCompatActivity {
    protected void onCreate(Bundle paramBundle) {
        super.onCreate(paramBundle);
        setContentView(2130968602);
        Context context = getApplicationContext();
        TextView textView = (TextView) findViewById(2131492951);
        String str = getIntent().getStringExtra("pin");
    }
    try {
        DatabaseUtilities databaseUtilities = new DatabaseUtilities(getApplicationContext());
        textView.setText((new CryptoUtilities("v1", str)).decrypt(databaseUtilities.fetchSecret()));
    } catch (Exception exception) {
        Log.e("Pinlock", "exception", exception);
    }
    Toast.makeText(context, str, 1);
}
}

```

- Mã pin từ người dùng sẽ được đưa vào biến string ‘str’

- Khởi tạo đối tượng ‘databaseUtilities’ từ class DatabaseUtilities và truyền tham số getApplicationContext()

- Sau đó **textview** được sử dụng để hiển thị văn bản trong giao diện người dùng, nó lấy giá trị từ biểu thức

**(new CryptoUtilities("v1", str)).decrypt(databaseUtilities.fetchSecret())**

Biểu thức này khởi tạo và sử dụng phương thức từ 2 class là CryptoUtilities và databaseUtilities

Tiến hành xem từng thành phần của biểu thức

✚ Xem xét CryptoUtilities.class

Hàm khởi tạo:

```

public class CryptoUtilities {
    private Cipher cipher;
    private SecretKeySpec key;
    private String pin;
    public CryptoUtilities(String paramString1, String paramString2) throws Exception {
        this.pin = paramString2;
        this.key = getKey(paramString1);
        this.cipher = Cipher.getInstance("AES/ECB/PKCS5Padding");
    }
}

```

Hàm khởi tạo nhận 2 tham số là paramString1 và paramString2 tương ứng với chuỗi “v1” và str

Sau đó sử dụng phương thức getKey cho paramString1

```

public SecretKeySpec getKey(String paramString) throws Exception {
    if (paramString.equalsIgnoreCase("v1")) {
        Log.d("Version", paramString);
        arrayOfByte = "t0ps3kr3tk3y".getBytes("UTF-8");
        return new SecretKeySpec(Arrays.copyOf(MessageDigest.getInstance("SHA-1").digest(arrayOfByte), 16), "AES");
    }
    Log.d("Version", (String)arrayOfByte);
    byte[] arrayOfByte = "SampleSalt".getBytes();
    char[] arrayOfChar = this.pin.toCharArray();
    return new SecretKeySpec(SecretKeyFactory.getInstance("PBKDF2withHmacSHA1").generateSecret(new PBEKeySpec(arrayOfChar, arrayOfByte, 1000, 128)).getEncoded(), "AES");
}

```

- Nếu như là “v1” thì sẽ thực hiện băm chuỗi ‘t0ps3kr3tk3y’ bằng SHA-1 lấy 16byte đầu để làm khóa AES sau đó trả về giá trị vào đối tượng SecretKeySpec.

- Nếu không thì sử dụng paramString2 (chuỗi **pin** do người dùng nhập vào) và salt để tạo key AES trả về giá trị vào đối tượng SecretKeySpec.

Kết quả sau khi khởi tạo ta sẽ có:

- Chuỗi pin = paramString2 (chuỗi pin người dùng nhập vào)
- Chuỗi key AES ( là 1 trong 2 trường hợp tạo key ở trên )
- Biến cypher có thể dùng cho mã hóa AES với mode ECB

➡ Tiếp theo xem xét phương thức decrypt

```
public String decrypt(String paramString) throws Exception {
    byte[] arrayOfByte = Base64.decode(paramString.getBytes(), 2);
    Log.d("Status", arrayOfByte.toString());
    this.cipher.init(2, this.key);
    return new String(this.cipher.doFinal(arrayOfByte), "UTF-8");
}
```

- Input: một string
- Phương thức này sẽ lấy key AES ở trên

Sử dụng key AES ở trên để giải mã và chuyển thành string sử dụng mã hóa UTF-8

- Output: có thể là Flag
- ➡ Đầu vào của decrypt trong biểu thức là databaseUtilities.fetchSecret()

Phương thức fetchSecret() trong lấy giá trị đầu tiên từ cột **entry** trong bảng secretsDBv1

```
public String fetchSecret() throws IOException {
    openDB();
    Cursor cursor = this.db.rawQuery("SELECT entry FROM secretsDBv1", null);
    String str = "";
    if (cursor.moveToFirst())
        str = cursor.getString(0);
    Log.d("secret", str);
    cursor.close();
    return str;
}
```

➡ Để xem được database sử dụng công cụ sqlite3:

Có tổng cộng 4 table

Xem giá trị từ cột entry

```

kali@kali: ~/hie/pinstore/assets
agedigest.update(arrayOfByte, 0, arrayOfByte.length);
File Actions Edit View Help);

(kali㉿kali)-[~/hie/pinstore/assets]
$ sqlite3 pinlock.db
SQLite version 3.46.0 2024-05-23 13:25:27
Enter ".help" for usage hints.
sqlite> .table.format("%02x", new Object[] { Byte.valueOf(paramArrayOfbyte[i]) })
android_metadata      secretsDBv1      secretsDBv2
sqlite> SELECT entry FROM secretsDBv1;
hcsvUnln5jMdw3GeI4o/txB5vaEf1PFAKQ3kPsRW2o5rR0a1JE54d0BLkzXPtqB
sqlite> SELECT entry FROM secretsDBv2;
Bi528nDlNBcX9BcCC+ZqGQo10z01+GOWSmvxRj7jg1g=option {
sqlite> fbyte = Base64.decode(paramString.getBytes(), 2);
d("Status", arrayOfByte.toString());
.cipher.init(2, this.key);
new String(this.cipher.doFinal(arrayOfByte), "UTF-8");

String encrypt(String paramString) throws Exception {
arrayOfByte = paramString.getBytes();
.cipher.init(1, this.key);
yOfByte = this.cipher.doFinal(arrayOfByte);
d("Status", Base64.encodeToString(arrayOfByte, 2));
Base64.encodeToString(arrayOfByte, 2);

SecretKeySpec getKey(String paramString) throws Exception {
paramString.equalsIgnoreCase("v1")) {
}

```

Thấy có 2 chuỗi được mã hóa trong 2 table

Chuỗi trong bảng secretDBv1 được mặc định giải mã theo key tạo theo kiểu thứ nhất

```

if (paramString.equalsIgnoreCase("v1")) {
    Log.d("Version", paramString);
    arrayOfByte = "tOp$3kr3tk3y".getBytes("UTF-8");
    return new SecretKeySpec(Arrays.copyOf(MessageDigest.getInstance("SHA-1").digest(arrayOfByte), 16), "AES");
}

```

Chuỗi trong bảng secretDBv2 có vẻ khác nên em sẽ giải mã theo kiểu tạo key thứ hai

```

Log.d("Version", (String)arrayOfByte);
byte[] arrayOfByte = "SampleSalt".getBytes();
char[] arrayOfChar = this.pin.toCharArray();
return new SecretKeySpec(SecretKeyFactory.getInstance("PBKDF2withHmacSHA1").generateSecret(new PBEKeySpec(arrayOfChar, arrayOfByte, 1000, 128)).getEncoded(), "AES");
}

```

Dựa vào các phương thức đã có sẵn ở trên viết đoạn code giải mã

```
Main.java +  
1 import javax.crypto.Cipher;  
2 import javax.crypto.SecretKeyFactory;  
3 import javax.crypto.spec.PBEKeySpec;  
4 import javax.crypto.spec.SecretKeySpec;  
5 import java.security.MessageDigest;  
6 import java.util.Arrays;  
7 import java.util.Base64;  
8  
9 public class Main {  
10     public static void main(String[] args) throws Exception {  
11         String pin = "7498";  
12         String secretDB1 = "hcsvUnln5jMdw3GeI4o/txB5vaEf1PFAAnQ3kPsRW2o5rR0a1JE54d0BLkzXPtqB";  
13         String secretDB2 = "Bi528nDlNBcX9BcCC+ZqGQo10z01+GOWSmvxRj7jg1g=";  
14         char[] arrayOfChar = pin.toCharArray();  
15  
16         Cipher cipher = Cipher.getInstance("AES/ECB/PKCS5Padding");  
17         // key cho chuoi o DB1  
18         byte[] arrayOfByte1 = "t0ps3kr3tk3y".getBytes("UTF-8");  
19         SecretKeySpec key1 = new SecretKeySpec(  
20             Arrays.copyOf(MessageDigest.getInstance("SHA-1")  
21                 .digest(arrayOfByte1), 16),  
22             "AES");  
23  
24         // key cho chuoi o DB2  
25         byte[] arrayOfByte2 = "SampleSalt".getBytes();  
26         SecretKeySpec key2 = new SecretKeySpec(  
27             SecretKeyFactory.getInstance("PBKDF2WithHmacSHA1")  
28                 .generateSecret(new PBEKeySpec(arrayOfChar, arrayOfByte2, 1000, 128))  
29                 .getEncoded(),  
30             "AES"  
31         );  
32  
33         // Giai ma  
34         byte[] arrayOfByteFlag1 = Base64.getDecoder().decode(secretDB1);  
35         byte[] arrayOfByteFlag2 = Base64.getDecoder().decode(secretDB2);  
36         cipher.init(2, key1);  
37         String Flag1 = new String(cipher.doFinal(arrayOfByteFlag1), "UTF-8");  
38         cipher.init(2, key2);  
39         String Flag2 = new String(cipher.doFinal(arrayOfByteFlag2), "UTF-8");  
40  
41         System.out.println("Flag1: " + Flag1);  
42         System.out.println("Flag2: " + Flag2);  
43     }  
44 }  
45 |
```

Kết quả sau khi chạy xong:

```
Run Share Command Line Arguments

Flag1: Here is what the data will look like
Flag2: Flag:OnlyAsStrongAsWeakestLink
** Process exited - Return Code: 0 **
```

Kết quả: Flag:OnlyAsStrongAsWeakestLink

## Kịch bản 2

### 2. Kịch bản 02. Thực hiện phân tích tập tin ứng dụng thu được.

- Mô tả: Ứng dụng kb02 cần được phân tích thành mã smali để tìm flag.
- Tài nguyên thực hiện: kb02\_zha.apk
- Yêu cầu – Gợi ý: sử dụng công cụ APKTool/ JADX/ dex2jar/ jdgui/ Android Studio, flag có dạng CTF{....}

Đáp án:

```
<manifest android:versionCode="1" android:versionName="1.0" package="com.example.blink" platformBuildVersionCode="1"
platformBuildVersionName="1">
<uses-sdk android:minSdkVersion="15" android:targetSdkVersion="27"/>
<application android:theme="@style/AppTheme" android:label="@string/app_name" android:icon="@mipmap/ic_launcher"
    android:debuggable="true" android:allowBackup="true" android:supportsRtl="true" android:roundIcon="@mipmap/ic_launcher_round">
    <activity android:theme="@style/AppTheme.NoActionBar" android:label="@string/title_activity_r2d2" android:name="com.example.blink.r2d2"/>
    <activity android:name="com.example.blink.MainActivity">
        <intent-filter>
            <action android:name="android.intent.action.MAIN"/>
            <category android:name="android.intent.category.LAUNCHER"/>
        </intent-filter>
    </activity>
</application>
</manifest>
```

Tiến hành compile file kb02\_zha.apk bằng jadx, xem trong tập tin AndroidManifest.xml, ta thấy được những tập tin liên quan là r2d2 và MainActivity.

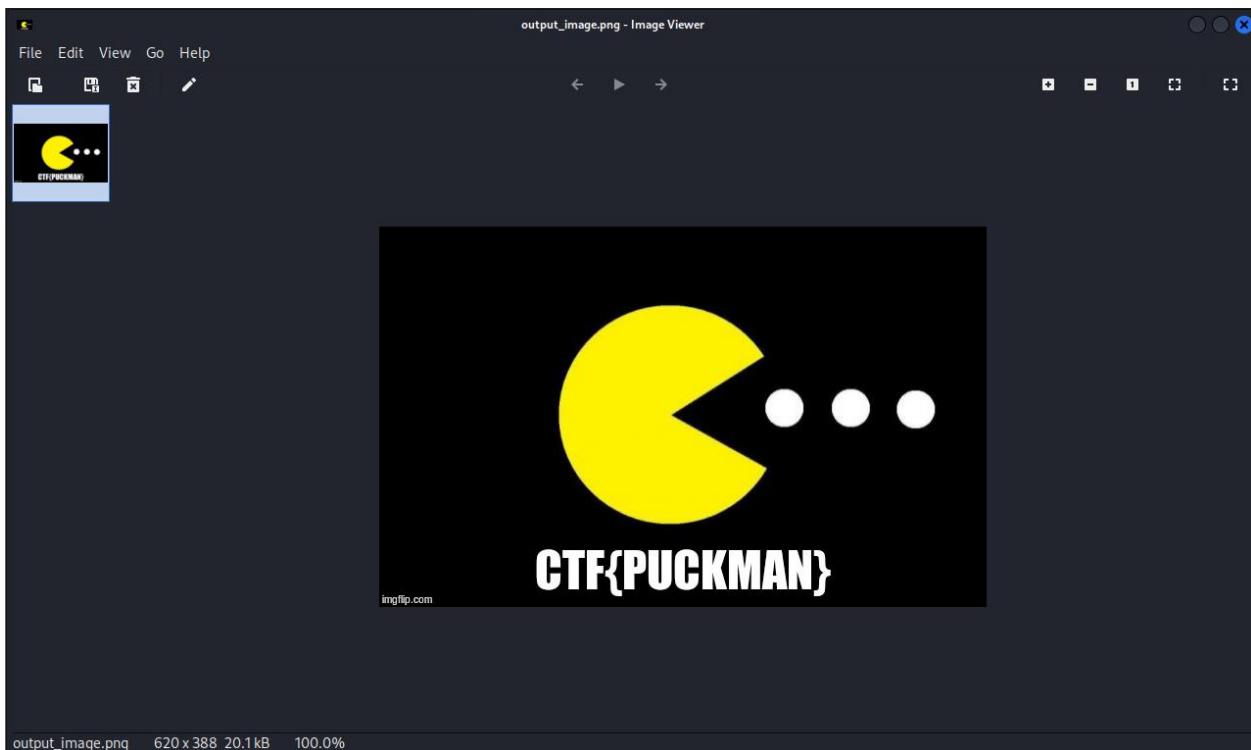
# Lab 5: Mobile Forensic

```
File Actions Edit View Help
zsh: corrupt history file /home/kali/.zsh_history
└─(kali㉿kali)-[~/Desktop/resources-session05]
$ ./jadex/bin/jadx -d kb02 kb02_zha.apk
Picked up _JAVA_OPTIONS: -Dawt.useSystemAAFontSettings=on -Dswing.aatext=true
INFO - loading ...
INFO - processing ...
ERROR - finished with errors, count: 4
```

Mở xem xét tập tin r2d2 ta thấy chuỗi base64 nghi vấn

Tiến hành copy chuỗi ra 1 file text và chuyển thành hình ảnh, ta có được flag

```
(kali㉿kali)-[~/Desktop/resources-session05/kb02]
$ base64 -d kb02.txt > output_image.png
```



Flag: CTF{PUCKMAN}

### Kịch bản 3

#### 3. Kịch bản 03. Thực hiện phân tích tập tin ứng dụng thu được.

- Mô tả: Một ứng dụng có tính năng ghi nhớ các địa điểm mà người dùng muốn hay không muốn tham quan chỉ bằng dấu tick đơn giản trên bản đồ. Tìm flag.
- Tài nguyên: kb03\_yon.apk
- Yêu cầu – Gợi ý: Decompile, chú ý CSDL của ứng dụng.

Gợi ý:

# Lab 5: Mobile Forensic

This XML file does not appear to have any style information associated with it. The document tree is shown below.

```
<manifest android:versionCode="1" android:versionName="1.0" package="com.example.yayornay" platformBuildVersionCode="1" platformBuildVersionName="1">
<uses-sdk android:minSdkVersion="24" android:targetSdkVersion="27"/>
<uses-permission android:name="android.permission.ACCESS_FINE_LOCATION"/>
<uses-permission android:name="android.permission.ACCESS_NETWORK_STATE"/>
<uses-permission android:name="android.permission.INTERNET"/>
<uses-feature android:glEsVersion="0x20000" android:required="true"/>
<application android:theme="@style/AppTheme" android:label="@string/app_name" android:icon="@mipmap/ic_launcher" android:debuggable="true" android:allowBackup="true"
    android:supportsRtl="true" android:roundIcon="@mipmap/ic_launcher_round">
    <meta-data android:name="com.google.android.geo.API_KEY" android:value="@string/google_maps_key"/>
    <activity android:label="@string/title_activity_maps" android:name="com.example.yayornay.MapsActivity"/>
    <activity android:name="com.example.yayornay.MainActivity">
        <intent-filter>
            <action android:name="android.intent.action.MAIN"/>
            <category android:name="android.intent.category.LAUNCHER"/>
        </intent-filter>
    </activity>
    <activity android:theme="@android:style/Theme.Translucent.NoTitleBar" android:name="com.google.android.gms.common.api.GoogleApiActivity" android:exported="false"/>
    <meta-data android:name="com.google.android.gms.version" android:value="@integer/google_play_services_version"/>
    <meta-data android:name="android.support.VERSION" android:value="26.1.0"/>
</application>
</manifest>
```

Tiến hành compile file kb03\_yon.apk bằng jadx, xem trong tập tin AndroidManifest.xml, ta thấy được những tập tin liên quan là MapsActivity

The screenshot shows a code editor window with the following details:

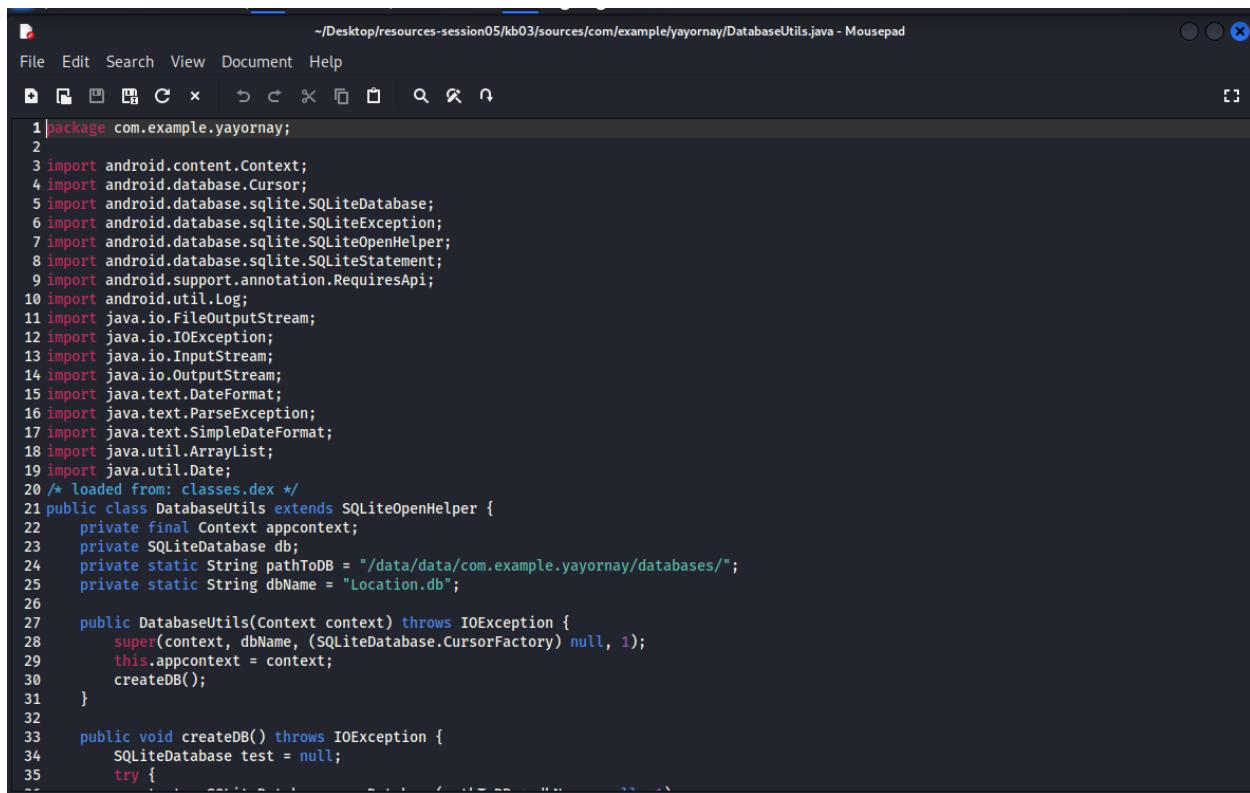
- Title Bar:** The title bar displays the path `~/Desktop/resources-session05/kb03/sources/com/example/yayornay/MapsActivity.java` and the application name `Mousepad`.
- Menu Bar:** The menu bar includes File, Edit, Search, View, Document, and Help.
- Toolbar:** The toolbar contains icons for New, Open, Save, Cut, Copy, Paste, Find, Replace, and Delete.
- Code Area:** The main area contains Java code for a `MapsActivity`. The code handles map click events and adds markers to the map based on location data from a database. It uses `MarkerOptions` to set marker positions and titles, and `BitmapDescriptorFactory` to define marker colors.

```
33     this.mMap.setOnMapClickListener(this);
34     try {
35         DatabaseUtils dbUtil = new DatabaseUtils(getApplicationContext());
36         ArrayList<Location> locations = dbUtil.fetchLocations();
37         Iterator<Location> it = locations.iterator();
38         while (it.hasNext()) {
39             Location location = it.next();
40             LatLng temp = new LatLng(location.latitude, location.longitude);
41             float color = 120.0f;
42             String label = "Yay!";
43             if (location.color == 0.0d) {
44                 color = 0.0f;
45                 label = "Nay!";
46             }
47             this.mMap.addMarker(new MarkerOptions().position(temp).title(label).icon(BitmapDescriptorFactory.defaultMarker(color)));
48         }
49         LatLng bSidesSF = new LatLng(37.7842927d, -122.4037178d);
50         this.mMap.moveCamera(CameraUpdateFactory.newLatLng(bSidesSF));
51         this.mMap.animateCamera(CameraUpdateFactory.zoomTo(10.0f));
52     } catch (IOException e) {
53         e.printStackTrace();
54     }
55 }
56
57 @Override // com.google.android.gms.maps.GoogleMap.OnMapLongClickListener
58 public void onMapLongClick(LatLng point) {
59     GoogleMap googleMap = this.mMap;
60     if (googleMap != null) {
61         googleMap.addMarker(new MarkerOptions().position(point).title("Nay!").icon(BitmapDescriptorFactory.defaultMarker(0.0f)));
62     }
63     try {
64         DatabaseUtils dbUtil = new DatabaseUtils(getApplicationContext());
65         Location location = new Location(new Date(System.currentTimeMillis()), point.latitude, point.longitude, 0.0f);
66         dbUtil.insertLocation(location);
67     } catch (IOException e) {
```

Xem xét trong file MapActivity.java ta thấy được database có vẻ liên quan đến file DatabaseUtils

Từ đó ta mở file DatabaseUtils.java, ta thấy database được lưu trong 1 tập tin tên là Location.db

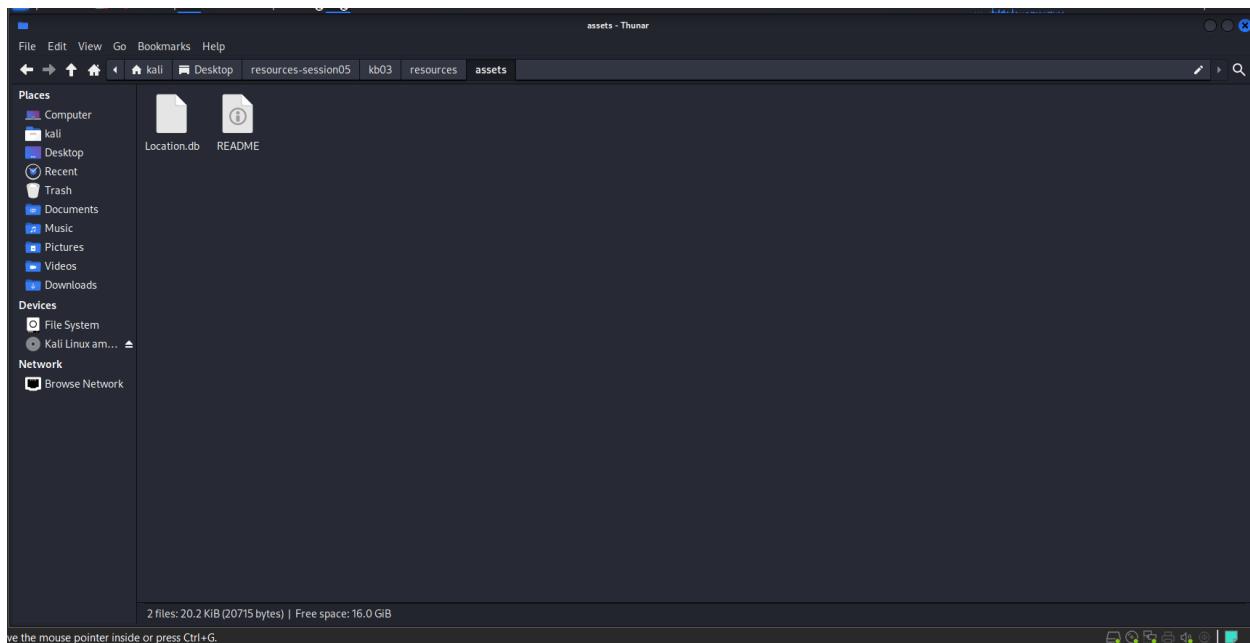
## Lab 5: Mobile Forensic



```

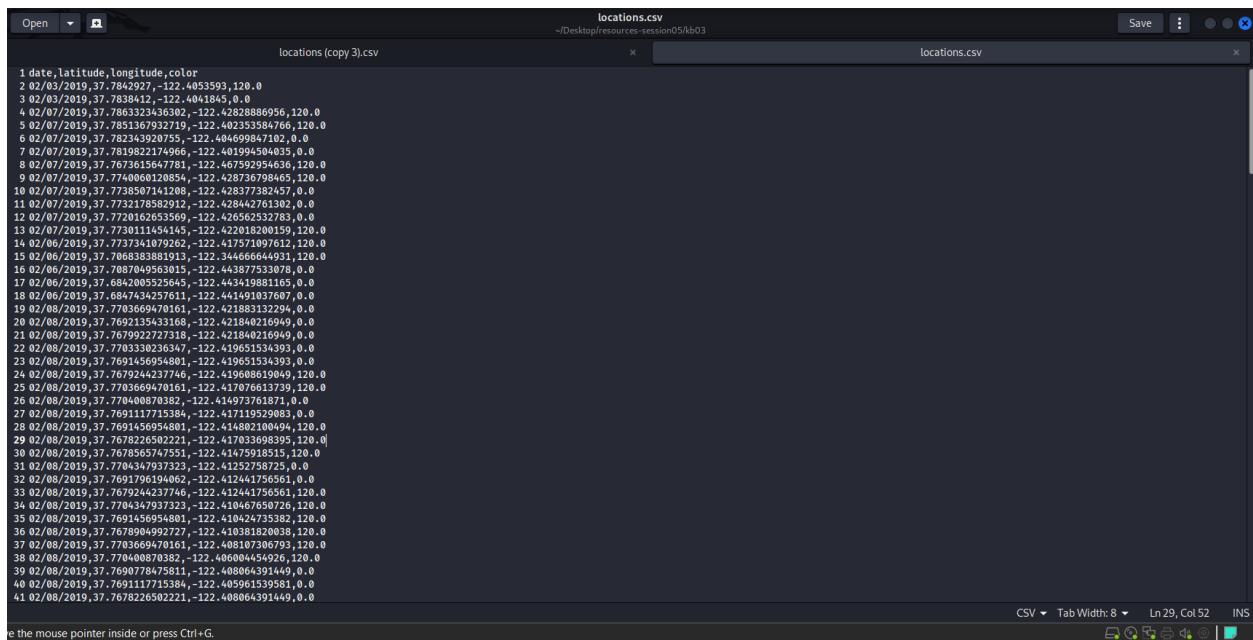
1 package com.example.yayornay;
2
3 import android.content.Context;
4 import android.database.Cursor;
5 import android.database.sqlite.SQLiteDatabase;
6 import android.database.sqlite.SQLiteException;
7 import android.database.sqlite.SQLiteOpenHelper;
8 import android.database.sqlite.SQLiteStatement;
9 import android.support.annotation.RequiresApi;
10 import android.util.Log;
11 import java.io.FileOutputStream;
12 import java.io.IOException;
13 import java.io.InputStream;
14 import java.io.OutputStream;
15 import java.text.DateFormat;
16 import java.text.ParseException;
17 import java.text.SimpleDateFormat;
18 import java.util.ArrayList;
19 import java.util.Date;
20 /* loaded from: classes.dex */
21 public class DatabaseUtils extends SQLiteOpenHelper {
22     private final Context appcontext;
23     private SQLiteDatabase db;
24     private static String pathToDB = "/data/data/com.example.yayornay/databases/";
25     private static String dbName = "Location.db";
26
27     public DatabaseUtils(Context context) throws IOException {
28         super(context, dbName, (SQLiteDatabase.CursorFactory) null, 1);
29         this.appcontext = context;
30         createDB();
31     }
32
33     public void createDB() throws IOException {
34         SQLiteDatabase test = null;
35         try {

```

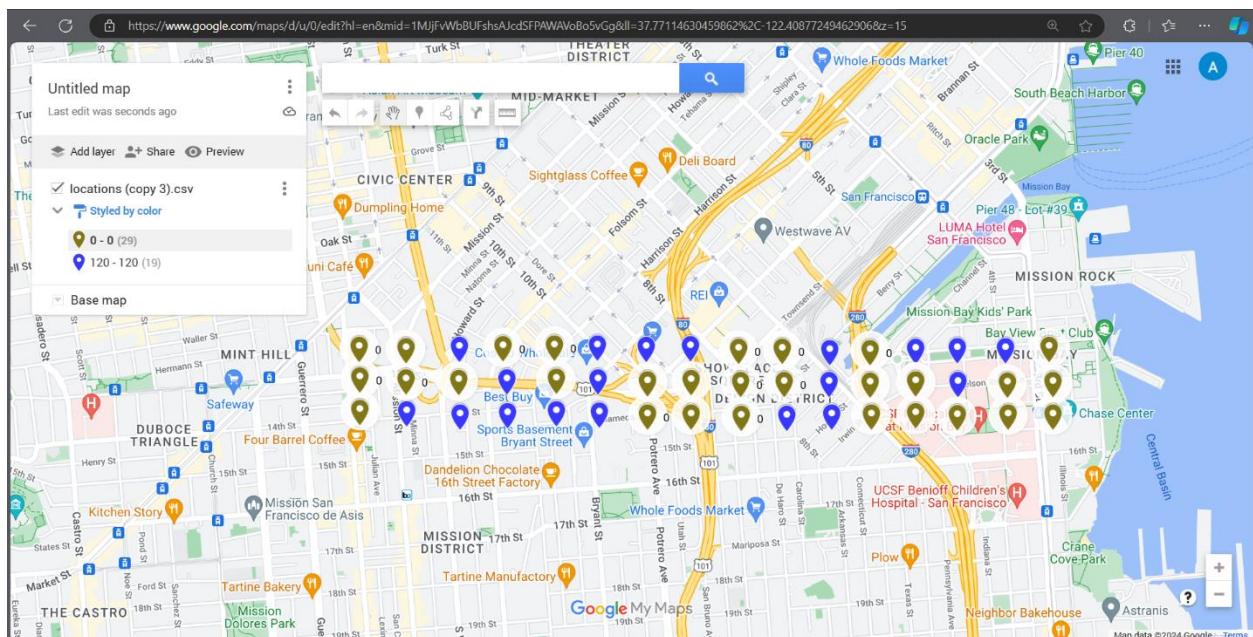


Trích xuất thông tin trong file Location.db, ta có những dòng thông tin gồm ngày tháng, kinh độ, vĩ độ và màu sắc lưu trong google maps

## Lab 5: Mobile Forensic



Sau khi lọc ra từng ngày, ta có được ngày 02/08/2019 có những dấu chấm đáng nghi. Tiếp tục phân loại dấu chấm theo màu sắc được lưu trong database



Liên tưởng tới bài lab trước, em đoán ở đây là ngôn ngữ Braille

## Lab 5: Mobile Forensic

Flag: Z3DLA

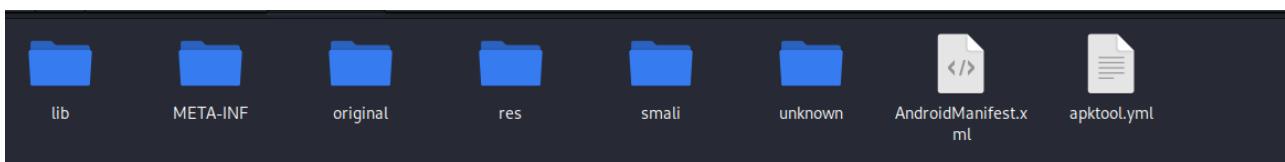
### Kịch bản 4

#### 4. Kịch bản 04. Điều tra trên tập tin ứng dụng thu được.

- Mô tả: Một ứng dụng thời tiết đơn giản có tính năng thu thập và hiển thị thông tin thời tiết.
- Tài nguyên: kb04\_tianqi.apk
- Yêu cầu – Gợi ý: Xác định phiên bản Android đang chạy của ứng dụng. Sử dụng một số công cụ decompile apk như JADX để phân tích code ứng dụng. Flag có định dạng CTF{...}

Đáp án:

Sau khi decompile bằng apktool



```

1 !! brut.androlib.meta.MetaInfo
2 apkFileName: kb04_tianqi.apk
3 compressionType: false
4 doNotCompress:
5 - resources.arsc
6 - png
7 isFrameworkApk: false
8 packageInfo:
9   forcedPackageId: '127'
10  renameManifestPackage: null
11 sdkInfo:
12   minSdkVersion: '26'
13   targetSdkVersion: '27'
14 sharedLibrary: false
15 sparseResources: false
16 unknownFiles:
17   androidsupportmultidexversion.txt: '8'

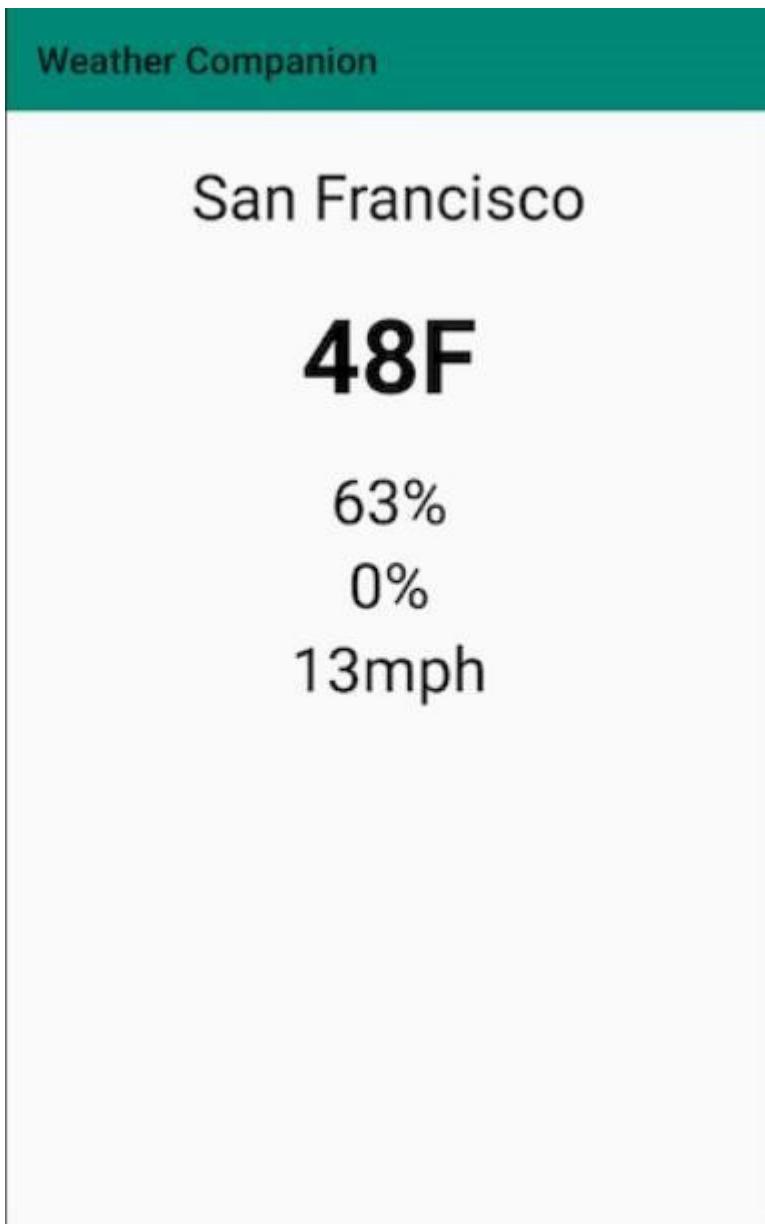
```

minSdkVersion: '26' có nghĩa là ứng dụng này chỉ có thể chạy trên các thiết bị có hệ điều hành Android **8.0** trở lên.

targetSdkVersion: '27' có nghĩa là ứng dụng được thiết kế và tối ưu hóa để chạy tốt trên **Android 8.1** (API Level 27).

Kịch bản em tham khảo từ <https://aadityapurani.com/2019/03/07/bsidessf-ctf-2019-mobile-track/#weather>.

Hình ảnh trích xuất ra được flag và các bước làm chỉ là minh họa cho ý tưởng, bởi vì không có tài khoản truy cập Google Cloud Storage của challenge CTF này.



```

1 <?xml version="1.0" encoding="utf-8" standalone="no"?><manifest xmlns:android="http://schemas.android.com/apk/res/android" package="com.example.myapplication" platformBuildVersionCode="1" platformBuildVersionName="1.0">
2   <uses-permission android:name="android.permission.INTERNET" />
3   <application android:allowBackup="true" android:icon="@mipmap/ic_launcher" android:label="@string/app_name" android:roundIcon="@mipmap/ic_launcher_round" android:supportsRtl="true" android:theme="@style/AppTheme">
4     <activity android:name=".MainActivity" android:label="@string/app_name" android:theme="@style/AppTheme.NoActionBar">
5       <intent-filter>
6         <action android:name="android.intent.action.MAIN" />
7         <category android:name="android.intent.category.LAUNCHER" />
8       </intent-filter>
9     </activity>
10   </application>
11 </manifest>

```

Chương trình có activity là com.example.myapplication.MainActivity, đọc sơ qua thì đoạn mã cố gắng thực thi 1 tác vụ bằng cách sử dụng lớp “a”, truyền hoạt động “this” và ngữ cảnh “getApplicationContext()” làm đối số. Lớp này có thể là 1 lớp thực hiện tùy chỉnh của 1 tác vụ bất đồng bộ AsyncTask để thực hiện các hoạt động nền. Sau đó dòng execute để khởi động thực thi tác vụ.

The screenshot shows a VMware Workstation interface with several virtual machines running. The windows visible include "Other Linux 5.x kernel", "Windows 10 x64", "Manged NODE", "Ubuntu", "Clone of Ubuntu", and "pfSense". In the foreground, a Java code editor displays the following code:

```
package com.example.myapplication;

import android.os.Bundle;
import android.support.v7.app.c;
import android.support.v7.widget.Toolbar;
import java.io.IOException;
/* loaded from: classes.dex */
public class MainActivity extends c {
    @Override // android.support.v7.app.c, android.support.v4.app.h, android.support.v4.app.z, android.app.Activity
    public void onCreate(Bundle bundle) {
        super.onCreate(bundle);
        setContentView(R.layout.activity_main);
        c().a((Toolbar) findViewById(R.id.toolbar));
        try {
            new a(this, getApplicationContext()).execute(new Void[0]);
        } catch (IOException e) {
            e.printStackTrace();
        }
    }
}
```

- Từ đó ta xem code của a.java

Đoạn mã của code a.java có chức năng lấy dữ liệu thời tiết và hiển thị nó trên màn hình. Bao gồm một số thành phần như sau:

- Lớp `a` được kế thừa từ **AsyncTask**, cho phép thực hiện các tác vụ nền mà không làm gián đoạn giao diện người dùng. Ví dụ :
  - Phương thức **doInBackground** được thực thi trong nền và truy xuất dữ liệu thời tiết. Nó sử dụng lớp **Utils** để thực hiện các chức năng hỗ trợ khác nhau như thao tác chuỗi và giải mã. Tiếp đó xây dựng URL bằng thông tin từ đối tượng `gVar` và lớp `c` rồi thực hiện yêu cầu HTTP GET đến URL được xây dựng. Cuối cùng là đọc phản hồi từ máy chủ và chuyển đổi nó thành chuỗi.
  - Phương thức **onPostExecute** sẽ được gọi sau khi `doInBackground` kết thúc và nhận dữ liệu thời tiết dưới dạng chuỗi. Sau đó, nó phân tích cú pháp phản hồi JSON bằng **JSONObject** rồi trích xuất chi tiết thời tiết như thành phố, nhiệt độ, lượng mưa, độ ẩm và gió. Cuối cùng, nó cập nhật các phần tử **TextView** tương ứng trong bố cục `MainActivity` (ví dụ: `cityName`, `temperatureValue`) bằng cách đặt văn bản của chúng thành các giá trị được trích xuất.



Trường Đại học Công nghệ Thông tin (UIT)

Vậy Json ở đây là gì, ta xem trong file Utils.java, có thể thấy rằng chúng đang giải mã Base64, int thành char, BigInteger thành Hex và các chức năng của ba lớp thư viện native.

## Lab 5: Mobile Forensics

```

public static String a() {
    return new BigInteger("627096631258101466300448072738386213700396112265").toString(16);
}

public static String a(String str, int i) {
    byte[] bytes;
    if (i == 0) {
        bytes = Base64.getDecoder().decode(str);
    } else {
        org.apache.a.a.a.a aVar = new org.apache.a.a.a.a();
        bytes = str.getBytes();
        if (bytes != null && bytes.length != 0) {
            b.a aVar2 = new b.a();
            aVar.a(bytes, 0, bytes.length, aVar2);
            aVar.a(bytes, 0, -1, aVar2);
            bytes = new byte[aVar2.d];
            int length = bytes.length;
            if (aVar2.c != null) {
                int min = Math.min(aVar2.c != null ? aVar2.d - aVar2.e : 0, length);
                System.arraycopy(aVar2.c, aVar2.e, bytes, 0, min);
                aVar2.e += min;
                if (aVar2.e >= aVar2.d) {
                    aVar2.c = null;
                }
            }
        }
    }
    return new String(bytes);
}

public static String a(int[] iArr) {
    String str = "";
    for (int i = 0; i < iArr.length; i++) {
        str = str + ((char) iArr[i]);
    }
    return str;
}

```

```

/* JAD INFO: Access modifiers changed from: package-private */
public native byte[] dks();

/* JAD INFO: Access modifiers changed from: package-private */
public native long gci();

/* JAD INFO: Access modifiers changed from: package-private */
public native byte[] ss(String str, int i);

```

Để rõ hơn chúng ta sẽ hook “toString” và kết xuất giải mã bằng Frida. Quá trình tấn công sẽ bao gồm 3 bước

1. Bypass SSL Unpinning
2. Hook toString
3. Monitor toString

Đoạn mã dưới đây sẽ thực hiện quá trình trên:

```
Java.perform(function() {
    // Step - 1

    var array_list = Java.use("java.util.ArrayList");
    var ApiClient =
        Java.use('com.android.org.conscrypt.TrustManagerImpl');

    ApiClient.checkTrustedRecursive.implementation = function(a1, a2, a3, a4, a5, a6) {
        var k = array_list.$new(); return k;
    }

    // Step - 2

    console.log("Hooking Java");

    const StringBuilder = Java.use('java.lang.StringBuilder');

    StringBuilder.$init.overload('java.lang.String').implementation = function (arg) {
        return arg;
    }
})
```

## Lab 5: Mobile Forensics

```

var partial = ""; var result = this.$init(arg); console.log('new
StringBuilder(" + result + ")');
return result;

}

console.log("Hooking new StringBuilder(java.lang.String)");

// Step - 3

StringBuilder.toString implementation = function () {
    var result = this.toString(); console.log('StringBuilder.toString(); => ' + result)
    return result;
}

console.log("Hooking StringBuilder.toString() hooked");

}, 0);

```

Lưu nó dưới dạng urlconn-hook.js và chạy câu lệnh sau  
frida.exe -U -f com.example.myapplication -l urlconn-hook.js --no-pause  
Kết quả là chuỗi có định dạng JSON, được sử dụng để ủy quyền trong google cloud storage

## Lab 5: Mobile Forensics

```
new StringBuilder("undefined");
StringBuilder.toString(); => {"type": "service_account",
StringBuilder.toString(); => b
StringBuilder.toString(); => bs
StringBuilder.toString(); => bsi
StringBuilder.toString(); => bsid
StringBuilder.toString(); => bsides
StringBuilder.toString(); => bsides-
StringBuilder.toString(); => bsides-s
StringBuilder.toString(); => bsides-sf
StringBuilder.toString(); => bsides-sf-
StringBuilder.toString(); => bsides-sf-c
StringBuilder.toString(); => bsides-sf-ct
StringBuilder.toString(); => bsides-sf-ctf
StringBuilder.toString(); => bsides-sf-ctf-
StringBuilder.toString(); => bsides-sf-ctf-2
StringBuilder.toString(); => bsides-sf-ctf-20
StringBuilder.toString(); => bsides-sf-ctf-201
StringBuilder.toString(); => bsides-sf-ctf-2019
StringBuilder.toString(); => {"type": "service_account", "project_id": "bsides-sf-ctf-2019",
StringBuilder.toString(); => {"type": "service_account", "project_id": "bsides-sf-ctf-2019", "private_key_id": "6dd7fc48a8b1d49edf7f03f74bc47713bec7d989",
StringBuilder.toString(); => -----BEGIN PRIVATE KEY-----
MIIExVAIBADBNBkgkHiG9w0BAQF0AAQCBKWyggSiAgEAAoIBAQCBnaiJXfqZSsc4
SW4ir+yXYJ3IwJz8fwy0Pzoi1b/iTqTCK/ItjP61rJHB5MqKm6vz/WGw7Sm
nd21xMhFqcIwg8h1f7fzhiK0XuvBrB-S+cMEhw0RhwBuc03Zaghffal.ThzPy4x2r
Hh/N8lUYi4T8B8WAGaAzCJ3pui9rTG4+uucxO6pMNz3/ENzyOSmhr9Xb50kHY/Aq
H8tBAwH8oSWR1t1laC8Ch18wDun6fK1NgYYmmcBWxSjropu8f6MR7vMM3F0PEua
YBz1ZpxVvCi1iMS0wihtAa1ZBuWhRenAiTfwOct4r1SfbgySqcTsok5t0ws52tM
tuzt/P0JgMBAEAEcgAEAC97230lu125w1hufXdwTb7n/vLIw7SSyTvhf0dsWkb+5
+19od5SE5rVh79sDR/n4NEkt8b15u1w3jgP08W34qARHORX2TNgxpbpd231rY
ekuj+hW2atF6Afao00k+Bw+7L7lwrs6+j8pgLr4D1L2zebyz2hIMlw106s2pCnMyM
SMn593YfzmNotaoJ3dAwKG8PunRhoqDqlnu0574dXkTFkEePcedyza007Iy
CoNTUyJx007a1bTPVwyqm7m0ewh1SYJuh5oefcfaLIC9eT2nn6+c9qRgG03AwBC8V
s9tMnR9+DwFHrUmPn8AKB4YKov/JGUv29sV10d8YwQKBgQD1obsDSI96xm2s+lnE
Pab2+4t6sg/1o1dpu4w+/0u/RUT+joumswvF5n7KUXVAP8npnu6LYou1lhZ9+zV
ThTINxyTrA5VmPhoEpeY80FbcNw1TxKj9nfvbS2Jw2GLZQlapN0X7YE0axRNJs
CSyC2LDVV/j0abjzx0CCFc05+QkBgoC7vKpSDMjw17FpJm/T9oakdvZNzt3ZU
+DTmpXh/0W56j9vdzGk31mIV/SSzVUfwQaxFBzvQlAI69fru/DSt8h1CpGd
LEz8S0qq7ubbs7g0DK/TrwSQhvbd0eqGzu0ntrUfvaL7wq7yGKVjQSLhVmhHeZ
m94mGME20wKBgxFFSwGRGhM/WxKGVgA0J7WtGwZtnju+rAcRZFZAApfFq1JFhXNe
gkyDwhjadvpiAYB7P+ar1MwtxS1qGFBgYKlw+5oQ/Luh1Lx9vQgEWYQMQ
jE56sQ054IdIMrOCHnCVFzk0r5tVkjvh1M2G05C10BF7NiYG5PdrBT5AoGABEwT
```

Bây giờ chúng ta có thể quan sát tất cả các hàm gọi `toString()`, cuối cùng ta thấy được JSON hoàn chỉnh từ `str4`

Vậy là quá trình ghi đè `toString()` đã thành công, bây giờ ta có thể sử dụng `gsutil` để truy cập Google Cloud Storage từ giao diện dòng lệnh

```
$ gcloud auth activate-service-account --key-file=key.json
```

Activated service account credentials **for**: [weather-companionservice-acco@bsides-sf-ctf-2019.iam.gserviceaccount.com]

```
$ gsutil ls -p bsides-sf-ctf-2019 gs://weather-companion  
gs://weather-companion/flag.txt  
gs://weather-companion/weather.json
```

Chúng ta có thể thấy được flag.txt trong storage , ta sẽ dùng lệnh sau để copy moi thứ vào để đọc flag

```
$ gsutil -m cp -r -p bsides-sf-ctf-2019 gs://weather-companion ./
```

Flag: CTF{buck3t\_s3at5}

---

*Sinh viên đọc kỹ yêu cầu trình bày bên dưới trang này*

## YÊU CẦU CHUNG

- Sinh viên tìm hiểu và thực hiện bài tập theo yêu cầu, hướng dẫn.
- Nộp báo cáo kết quả chi tiết những việc (**Report**) bạn đã thực hiện, quan sát thấy và kèm ảnh chụp màn hình kết quả (nếu có); giải thích cho quan sát (nếu có).
- Sinh viên báo cáo kết quả thực hiện và nộp bài.

### Báo cáo:

- File **.DOCX** và **.PDF**. Tập trung vào nội dung, không mô tả lý thuyết.
- Nội dung trình bày bằng **Font chữ Times New Romans/ hoặc font chữ của mẫu báo cáo này (UTM Neo Sans Intel/UTM Viet Sach)** – cỡ chữ 13. **Canh đều (Justify) cho văn bản. Canh giữa (Center) cho ảnh chụp.**
- Đặt tên theo định dạng: [Mã lớp]-ExeX\_GroupY. (trong đó X là Thứ tự Bài tập, Y là mã số thứ tự nhóm trong danh sách mà GV phụ trách công bô).

*Ví dụ: [NT101.K11.ANTT]-Exe01\_Group03.*

- Nếu báo cáo có nhiều file, nén tất cả file vào file **.ZIP** với cùng tên file báo cáo.
- **Không đặt tên đúng định dạng – yêu cầu, sẽ KHÔNG chấm điểm bài nộp.**
- Nộp file báo cáo trên theo thời gian đã thống nhất tại [courses.uit.edu.vn](http://courses.uit.edu.vn).

### Đánh giá:

- Hoàn thành tốt yêu cầu được giao.
- Có nội dung mở rộng, ứng dụng.

*Bài sao chép, trẽ, ... sẽ được xử lý tùy mức độ vi phạm.*

**HẾT**