APK MAPPER DOCUMENTATION

By williams

Motivation (why ?):

The following tool is created because of the developer curiosity in android application penetration testing and also the developer is challenged by his mentor whether he is capable to create a tool that competent to mapped component inside multiple android applications and convert it into a single report. It took two days for developer to create the tool and the developer emphasize to whoever read documentation to improve the tool capability for education purpose.

Background :

According to developer android website, there are 4 vital components in android application:

1. Activities
2. Service
3. Broadcast Receiver
4. Content provider

All of the components is inscribed inside a file called “AndroidManifest.xml” and used as a guideline by the application in order to run properly. Furthermore, not just components, configuration like permissions and version of the application is also listed inside the file thus make it an elegant place to start doing penetration testing because any vulnerability in application level can be spotted by simply analyzing the xml file.

To know more about android permission check the following link:

1. https://developer.android.com/guide/topics/permissions/overview
2. https://developer.android.com/guide/topics/manifest/manifest-intro

Requirements:

- Python version 2.x
- Library:
    - https://pypi.org/project/AxmlParserPY/
        - Command: pip install AxmlParserPY
        - Info: used to decode the android manifest file inside the apk
    - https://pypi.org/project/python-magic/
        - Command: pip install python-magic

- Info: used to detect file extension

Implementations:

1. Overview of the library
    a. AxmlParserPY => decode apk file to get the androidmanifest.xml file
    Example:

    ```
    import axmlparserpy.apk as apk # to import the library
    ap = apk.APK('_PATH_TO_APK') # initiate an object to load apk of interest to be analyzed
    print ap.get_package() # used to get the package name
    print ap.get_androidversion_name() # used to get the package version

    #to get the application permission
    print ap.get_permissions() # this will return a list of all the android application permission in list type

    # to get the application component
    print ap.get_activities() #list all the activities
    print ap.get_providers() #list all the content provider
    print ap.get_receivers() #list all the broadcast receiver
    print ap.get_services() #list all the services

    #alternative way
    print app.get_elements("<xml tag>","<attribute>")
    # i use the get_elements approach because it can also retrieve the exported tag of the components
    ```

    b. Python-magic => check the extension of the file
    Example:

    ```
    import python-magic

    magic.from_file("test.apk",mime=True) # result = application/zip
    ```

    c. To list all of the apk file, it is to use "os" library from the python library
    Example:

    ```
    import os
    os.listdir(os.getcwd()) #return list all of the folder and file
    ```

d. To generate report for now just simply used the csv library from python

2. How to build it
    a. Now that we already know all the functionality it is time to put it into one program. To make it easier for tracing bug i create 6 functions that have own predefined task.
        i. List_service
        ii. List_activities
        iii. List_provider
        iv. List_receiver
        v. Check_backup
        vi. List_permission

Source code:

```
import axmlparserpy.apk as apk
import os
import csv
import magic

def list_service(app):
    result = []
    name = app.get_elements("service","android:name")
    export = app.get_elements("service","android:exported")

    for n,e in zip(name,export):
        if "true" in e:
            n += ":true"
            result.append(n)
        else:
            n += ":false"
            result.append(n)

    return ' \n'.join(result)


def list_activities(app):
    result = []

    name = app.get_elements("activity","android:name")
    export = app.get_elements("activity","android:exported")

    for n,e in zip(name,export):
        if "true" in e:
            n += ":true"
```

```
                result.append(n)
            else:
                n += ":false"
                result.append(n)

        return ' \n'.join(result)

def list_provider(app):
        result = []

        name = app.get_elements("provider","android:name")
        export = app.get_elements("provider","android:exported")

        for n,e in zip(name,export):
            if "true" in e:
                n += ":true"
                result.append(n)
            else:
                n += ":false"
                result.append(n)

        return ' \n'.join(result)

def list_receiver(app):
        result = []
        name = app.get_elements("receiver","android:name")
        export = app.get_elements("receiver","android:exported")

        for n,e in zip(name,export):
            if "true" in e:
                n += ":true"
                result.append(n)
            else:
                n += ":false"
                result.append(n)

        return " \n".join(result)

def check_backup(app):
        backup = app.get_elements("application","android:allowBackup")
        if "true" in backup[0]:
            return "true"
        else:
            return "false"

def list_permission(app):
        perm = app.get_permissions()
        return " \n".join(perm)

list_file =  os.listdir(os.getcwd()) #get all apk file in directory
header = ['Package Name','Version','Backup','Permission',"Activities","Content Provider",
"Service","Broadcast Receiver"]

result = open("apk_result.csv","w")
```

```
if len(list_file) == 0:
    print "HELP: you need to put it inside a directory that contain multiple .apk files"
else:
    print "#####################################################################"
    print "apk_mapper v 1.0: tools to mapping multiple apk manifest file for analysis"
    print "PLEASE DONT change the name of the python file !"
    print "CREATED BY: williams"
    print "#####################################################################"

    writer = csv.writer(result) #put a header in csv
    writer.writerow(header)

    for x in list_file:
        file_ext = magic.from_file(x,mime=True)
        if file_ext == "application/zip":
            result = []
            app = apk.APK(x)
            result.append(app.get_package()) #print app name
            result.append(app.get_androidversion_name()) #print app version
            result.append(check_backup(app))
            result.append(list_permission(app)) #print permission
            result.append(list_activities(app)) #print activity
            result.append(list_provider(app)) #print provider
            result.append(list_service(app))
            result.append(list_receiver(app))
            writer.writerow(result)
```

How to use it:

1. Put the apk_mapper.py into one directory filled with multiple apk

```
cuckoo@cuckoo-VirtualBox:~/Documents/apk_mapper$ ls -la
total 236472
drwxr-xr-x 2 cuckoo cuckoo      4096 Apr 30 09:14 .
drwxr-xr-x 3 cuckoo cuckoo      4096 Apr 28 14:20 ..
-rw-r--r-- 1 cuckoo cuckoo  64333312 Apr 27 07:47 adobe.apk
-rw-rw-r-- 1 cuckoo cuckoo      2862 Apr 29 20:00 apk_mapper.py
-rw-r--r-- 1 cuckoo cuckoo  48626817 Apr 27 07:47 dict.apk
-rw-r--r-- 1 cuckoo cuckoo 103939967 Apr 27 07:51 gacha.apk
-rw-r--r-- 1 cuckoo cuckoo  25223354 Apr 27 07:49 pou.apk
cuckoo@cuckoo-VirtualBox:~/Documents/apk_mapper$
```

2. Run the apk_mapper.py and in the end of the program it will generate apk_result.csv that contain report from all of the apk.

```
[cuckoo@cuckoo-VirtualBox:~/Documents/apk_mapper$ python apk_mapper.py
######################################################################
apk_mapper v 1.0: tools to mapping multiple apk manifest file for analysis
PLEASE DONT change the name of the python file !
CREATED BY: williams
######################################################################
[cuckoo@cuckoo-VirtualBox:~/Documents/apk_mapper$ ls
adobe.apk  apk_mapper.py  apk_result.csv  dict.apk  gacha.apk  pou.apk
cuckoo@cuckoo-VirtualBox:~/Documents/apk_mapper$
```

Preview:

Package Name	Version	Backup	Permission	Activities
com.youdao.hindict	4.0.10	true	android.permission.INTERNET android.permission.ACCESS_NETWORK_STATE android.permission.SYSTEM_ALERT_WINDOW android.permission.READ_EXTERNAL_STORAGE android.permission.ACCESS_WIFI_STATE android.permission.WRITE_EXTERNAL_STORAGE com.android.launcher.permission.INSTALL_SHORTCUT com.android.launcher.permission.UNINSTALL_SHORTCUT android.permission.RECEIVE_BOOT_COMPLETED android.permission.DISABLE_KEYGUARD android.permission.GET_TASKS android.permission.ACCESS_COARSE_LOCATION android.permission.ACCESS_FINE_LOCATION android.permission.CAMERA android.permission.RECORD_AUDIO android.permission.WAKE_LOCK com.google.android.c2dm.permission.RECEIVE android.permission.CHANGE_CONFIGURATION com.google.android.finsky.permission.BIND_GET_INSTALL_REFERRER_SERVICE android.permission.MOUNT_UNMOUNT_FILESYSTEMS android.permission.FLASHLIGHT android.permission.VIBRATE android.permission.MODIFY_AUDIO_SETTINGS android.permission.BROADCAST_STICKY android.permission.MODIFY_AUDIO_SETTINGS	com.youdao.hindict.activity.HostActivity:false com.youdao.hindict.activity.SplashActivity:false com.youdao.hindict.activity.GuideActivity:false com.youdao.hindict.activity.SearchActivity:false com.youdao.hindict.activity.QuickSearchActivity:true com.youdao.hindict.activity.DefinitionActivity:false com.youdao.hindict.activity.CameraActivity:false com.youdao.hindict.activity.SettingActivity:false com.youdao.hindict.activity.WebActivity:false com.youdao.hindict.activity.QuoteActivity:false com.youdao.hindict.activity.DisplayLanguageActivity:false com.youdao.hindict.activity.CopySettingActivity:false com.youdao.hindict.activity.LockScreenSettingActivity:false com.youdao.hindict.activity.SurveyActivity:false com.youdao.hindict.activity.SuggestActivity:false com.youdao.hindict.activity.MyFavoriteActivity:false com.youdao.hindict.activity.OfflineActivity:false com.youdao.hindict.activity.WordbookActivity:false com.facebook.ads.InterstitialAdActivity:false com.youdao.hindict.activity.YoutubeActivity:false com.youdao.hindict.activity.ClipboardTransActivity:false com.youdao.hindict.activity.PermissionDialogActivity:false com.youdao.hindict.activity.AutoStartDialogActivity:false com.youdao.hindict.activity.LockScreenActivity:false com.youdao.hindict.activity.YoutubeWebViewActivity:false com.youdao.hindict.activity.OcrRegionResultActivity:false com.youdao.hindict.activity.WordLockSettingsActivity:false com.youdao.sdk.common.YouDaoBrowser:false com.youdao.hindict.activity.DictCardDetailActivity:false com.youdao.hindict.activity.DictImageActivity:false com.youdao.hindict.activity.TabActivity:false com.youdao.hindict.activity.DialogueTransActivity:false com.youdao.hindict.activity.TranslationHistoryActivity:false com.youdao.hindict.activity.PrivacyActivity:false com.youdao.hindict.activity.PushTestActivity:false com.youdao.hindict.activity.FishGameActivity:false com.youdao.uclass.activity.ExtendedLiveActivity:false com.youdao.uclass.activity.CourseDetailActivity:false com.youdao.uclass.activity.LoginActivity:false com.youdao.uclass.activity.PhoneLoginActivity:false com.google.android.gms.ads.AdActivity:false com.youdao.cropper.YDCropActivity:false com.facebook.FacebookActivity:false com.facebook.CustomTabMainActivity:false com.facebook.ads.AudienceNetworkActivity:false com.google.android.gms.auth.api.signin.internal.SignInHubActivity:false com.google.firebase.auth.internal.FederatedSignInActivity:true com.google.android.gms.common.api.GoogleApiActivity:false com.youdao.ydaccount.activity.GoogleCallbackActivity:false com.youdao.ydaccount.activity.FacebookCallbackActivity:false com.youdao.ydaccount.activity.ThirdPartyWebLoginActivity:false com.youdao.ydliveplayer.activity.YDLiveActivity:false com.youdao.ydliveplayer.activity.NPSViewActivity:false com.youdao.ydplayerview.SimpleVideoActivity:false

Content Provider	Service	Broadcast Receiver
com.youdao.hindict.provider.FavoriteProvider:false com.duapps.ad.stats.DuAdCacheProvider:false androidx.core.content.FileProvider:false com.youdao.sdk.extra.common.AdDownloadProvider:false com.facebook.internal.FacebookInitProvider:false com.crashlytics.android.CrashlyticsInitProvider:false com.google.firebase.provider.FirebaseInitProvider:false com.google.android.gms.ads.MobileAdsInitProvider:false androidx.lifecycle.ProcessLifecycleOwnerInitializer:false	com.google.android.gms.analytics.CampaignTrackingService:false com.youdao.hindict.push.MyFirebaseMessagingService:false com.youdao.hindict.service.ClipboardWatcher:true com.youdao.hindict.service.JobService:true com.youdao.hindict.service.LockScreenService:true com.youdao.sdk.common.YouDaoAppService:false androidx.room.MultiInstanceInvalidationService:false com.google.android.gms.analytics.AnalyticsService:false com.google.android.gms.analytics.AnalyticsJobService:false com.google.firebase.messaging.FirebaseMessagingService:true com.google.android.gms.auth.api.signin.RevocationBoundService:true com.google.firebase.components.ComponentDiscoveryService:false com.google.firebase.iid.FirebaseInstanceIdService:true com.google.android.gms.measurement.AppMeasurementService:false com.google.android.gms.measurement.AppMeasurementJobService:false com.netease.nimlib.service.NimService:false com.netease.nimlib.service.NimService$Aux:false com.netease.nimlib.job.NIMJobService:true com.netease.nimlib.service.ResponseService:false com.youdao.ydplayerview.services.MediaPlayerService:true	com.google.android.gms.analytics.CampaignTrackingReceiver:true com.google.ads.conversiontracking.InstallReceiver:true com.youdao.hindict.receiver.DownloadReceiver:false com.youdao.hindict.receiver.UserPresentReceiver:false com.youdao.hindict.receiver.NetBroadcastReceiver:false com.youdao.hindict.receiver.BootCompletedReceiver:false com.youdao.hindict.receiver.MainProcessReceiver:false com.duapps.ad.base.PackageAddReceiver:false com.youdao.sdk.common.YouDaoTrackerReceiver:true com.youdao.hindict.receiver.TimezoneReceiver:false com.appsflyer.SingleInstallBroadcastReceiver:true com.google.android.gms.analytics.AnalyticsReceiver:false com.google.firebase.iid.FirebaseInstanceIdReceiver:true com.google.android.gms.measurement.AppMeasurementReceiver:false com.google.android.gms.measurement.AppMeasurementInstallReferrerReceiver:true com.netease.nimlib.service.NimReceiver:false com.netease.nimlib.service.ResponseReceiver:false com.youdao.ydliveplayer.receiver.MediaReiceiver:false

air.com.lunime.gachalife	1.0.9	false	android.permission.INTERNET android.permission.WRITE_EXTERNAL_STORAGE android.permission.READ_PHONE_STATE android.permission.ACCESS_NETWORK_STATE com.android.vending.BILLING	air.com.lunime.gachalife.AppEntry:false com.google.android.gms.ads.AdActivity:false	

Next Update:

Generate html report and json document so can be convert into nosql database