Androwarn

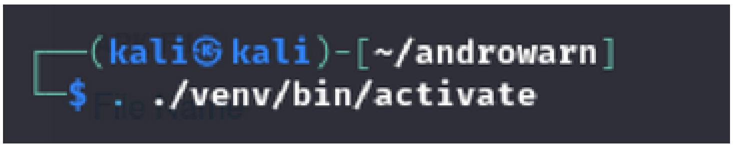
Aakash R
CB.SC.P2CYS23011

Androwarn is a tool designed to help in the analysis of Android applications by generating analysis reports. It detects potential security issues and vulnerabilities in Android APK files. Here are the steps to use Androwarn for Android malware analysis:

Step 1: Install Dependencies Androwarn requires certain dependencies to be installed. Ensure you have Python installed, and then install the necessary Python packages.

Install Python version 3.7.1

Activate Python Virtual Environment

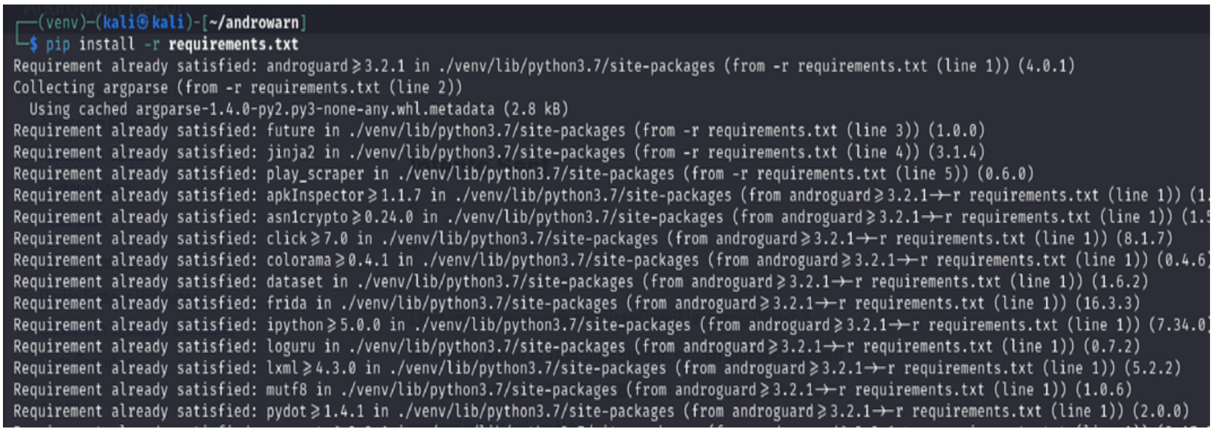


```
┌──(kali㉿kali)-[~/androwarn]
└─$ . ./venv/bin/activate
```

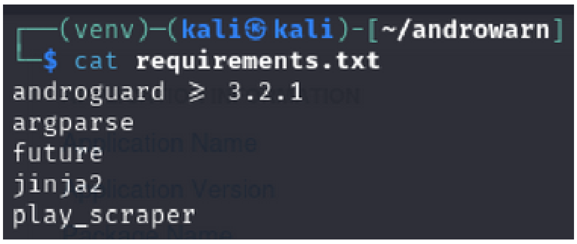
Step 2: Download Androwarn Download the Androwarn tool from its official GitHub repository.

git clone https://github.com/maaaaz/androwarn.git cd androwarn

Install all required packages from requirement.txt

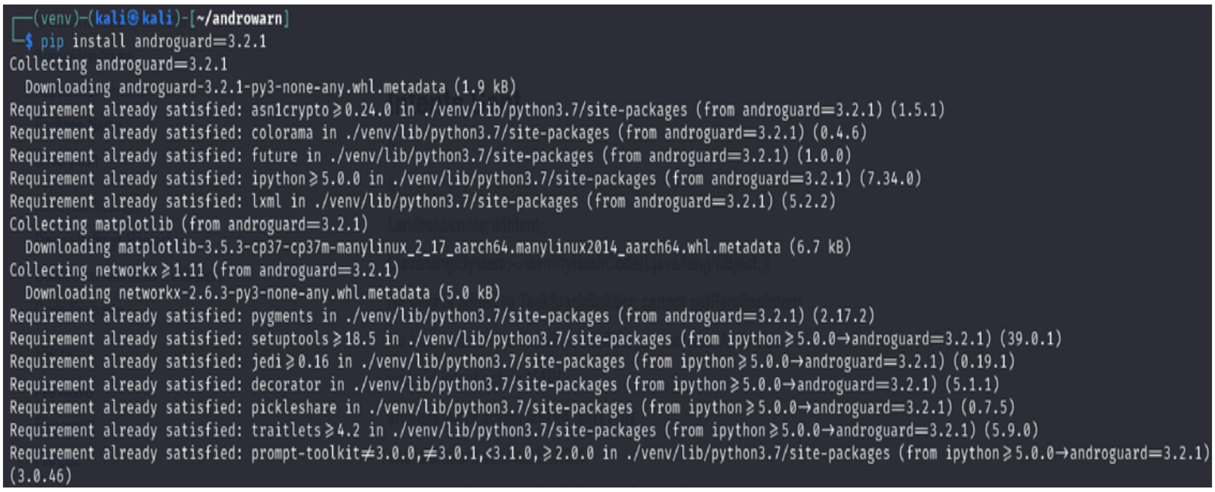


```
┌──(venv)─(kali㉿kali)-[~/androwarn]
└─$ pip install -r requirements.txt
Requirement already satisfied: androguard⩾3.2.1 in ./venv/lib/python3.7/site-packages (from -r requirements.txt (line 1)) (4.0.1)
Collecting argparse (from -r requirements.txt (line 2))
  Using cached argparse-1.4.0-py2.py3-none-any.whl.metadata (2.8 kB)
Requirement already satisfied: future in ./venv/lib/python3.7/site-packages (from -r requirements.txt (line 3)) (1.0.0)
Requirement already satisfied: jinja2 in ./venv/lib/python3.7/site-packages (from -r requirements.txt (line 4)) (3.1.4)
Requirement already satisfied: play_scraper in ./venv/lib/python3.7/site-packages (from -r requirements.txt (line 5)) (0.6.0)
Requirement already satisfied: apkInspector⩾1.1.7 in ./venv/lib/python3.7/site-packages (from androguard⩾3.2.1→-r requirements.txt (line 1)) (1.
Requirement already satisfied: asn1crypto⩾0.24.0 in ./venv/lib/python3.7/site-packages (from androguard⩾3.2.1→-r requirements.txt (line 1)) (1.5
Requirement already satisfied: click⩾7.0 in ./venv/lib/python3.7/site-packages (from androguard⩾3.2.1→-r requirements.txt (line 1)) (8.1.7)
Requirement already satisfied: colorama⩾0.4.1 in ./venv/lib/python3.7/site-packages (from androguard⩾3.2.1→-r requirements.txt (line 1)) (0.4.6
Requirement already satisfied: dataset in ./venv/lib/python3.7/site-packages (from androguard⩾3.2.1→-r requirements.txt (line 1)) (1.6.2)
Requirement already satisfied: frida in ./venv/lib/python3.7/site-packages (from androguard⩾3.2.1→-r requirements.txt (line 1)) (16.3.3)
Requirement already satisfied: ipython⩾5.0.0 in ./venv/lib/python3.7/site-packages (from androguard⩾3.2.1→-r requirements.txt (line 1)) (7.34.0
Requirement already satisfied: loguru in ./venv/lib/python3.7/site-packages (from androguard⩾3.2.1→-r requirements.txt (line 1)) (0.7.2)
Requirement already satisfied: lxml⩾4.3.0 in ./venv/lib/python3.7/site-packages (from androguard⩾3.2.1→-r requirements.txt (line 1)) (5.2.2)
Requirement already satisfied: mutf8 in ./venv/lib/python3.7/site-packages (from androguard⩾3.2.1→-r requirements.txt (line 1)) (1.0.6)
Requirement already satisfied: pydot⩾1.4.1 in ./venv/lib/python3.7/site-packages (from androguard⩾3.2.1→-r requirements.txt (line 1)) (2.0.0)
```

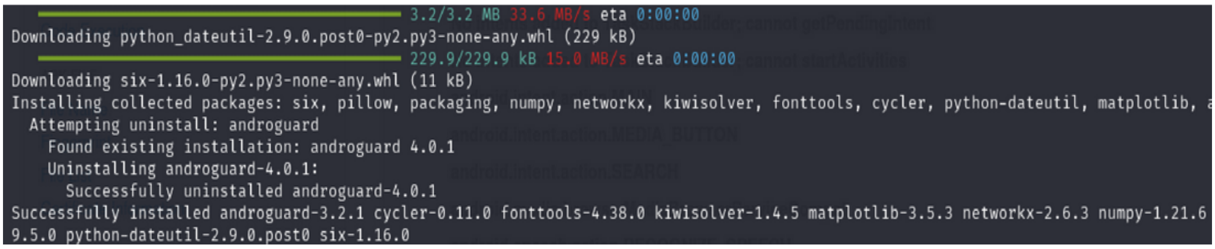


```
┌──(venv)─(kali㉿kali)-[~/androwarn]
└─$ cat requirements.txt
androguard  ⩾ 3.2.1
argparse
future
jinja2
play_scraper
```

Step 3: Manually Install Androguard



```
┌──(venv)─(kali㉿kali)-[~/androwarn]
└─$ pip install androguard==3.2.1
Collecting androguard==3.2.1
  Downloading androguard-3.2.1-py3-none-any.whl.metadata (1.9 kB)
Requirement already satisfied: asn1crypto>=0.24.0 in ./venv/lib/python3.7/site-packages (from androguard==3.2.1) (1.5.1)
Requirement already satisfied: colorama in ./venv/lib/python3.7/site-packages (from androguard==3.2.1) (0.4.6)
Requirement already satisfied: future in ./venv/lib/python3.7/site-packages (from androguard==3.2.1) (1.0.0)
Requirement already satisfied: ipython>=5.0.0 in ./venv/lib/python3.7/site-packages (from androguard==3.2.1) (7.34.0)
Requirement already satisfied: lxml in ./venv/lib/python3.7/site-packages (from androguard==3.2.1) (5.2.2)
Collecting matplotlib (from androguard==3.2.1)
  Downloading matplotlib-3.5.3-cp37-cp37m-manylinux_2_17_aarch64.manylinux2014_aarch64.whl.metadata (6.7 kB)
Collecting networkx>=1.11 (from androguard==3.2.1)
  Downloading networkx-2.6.3-py3-none-any.whl.metadata (5.0 kB)
Requirement already satisfied: pygments in ./venv/lib/python3.7/site-packages (from androguard==3.2.1) (2.17.2)
Requirement already satisfied: setuptools>=18.5 in ./venv/lib/python3.7/site-packages (from ipython>=5.0.0->androguard==3.2.1) (39.0.1)
Requirement already satisfied: jedi>=0.16 in ./venv/lib/python3.7/site-packages (from ipython>=5.0.0->androguard==3.2.1) (0.19.1)
Requirement already satisfied: decorator in ./venv/lib/python3.7/site-packages (from ipython>=5.0.0->androguard==3.2.1) (5.1.1)
Requirement already satisfied: pickleshare in ./venv/lib/python3.7/site-packages (from ipython>=5.0.0->androguard==3.2.1) (0.7.5)
Requirement already satisfied: traitlets>=4.2 in ./venv/lib/python3.7/site-packages (from ipython>=5.0.0->androguard==3.2.1) (5.9.0)
Requirement already satisfied: prompt-toolkit!=3.0.0,!=3.0.1,<3.1.0,>=2.0.0 in ./venv/lib/python3.7/site-packages (from ipython>=5.0.0->androguard==3.2.1)
(3.0.46)
```

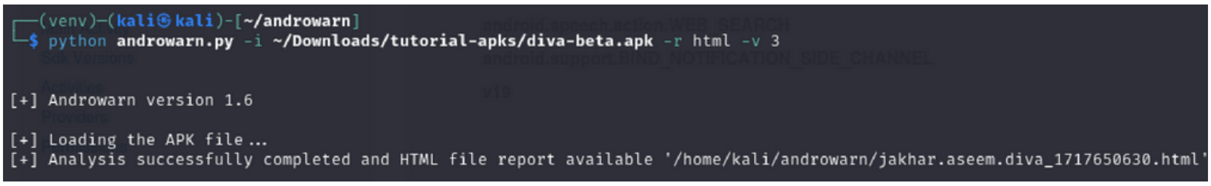


```
    ━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━ 3.2/3.2 MB 33.6 MB/s eta 0:00:00
Downloading python_dateutil-2.9.0.post0-py2.py3-none-any.whl (229 kB)
    ━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━ 229.9/229.9 kB 15.0 MB/s eta 0:00:00
Downloading six-1.16.0-py2.py3-none-any.whl (11 kB)
Installing collected packages: six, pillow, packaging, numpy, networkx, kiwisolver, fonttools, cycler, python-dateutil, matplotlib, a
  Attempting uninstall: androguard
    Found existing installation: androguard 4.0.1
    Uninstalling androguard-4.0.1:
      Successfully uninstalled androguard-4.0.1
Successfully installed androguard-3.2.1 cycler-0.11.0 fonttools-4.38.0 kiwisolver-1.4.5 matplotlib-3.5.3 networkx-2.6.3 numpy-1.21.6
9.5.0 python-dateutil-2.9.0.post0 six-1.16.0
```

Step 4: Run Androwarn

python androwarn.py -i diva-beta.apk -r html -v 3

Execute Androwarn to analyze the APK file. diva-beta.apk with the path to your APK file.



```
┌──(venv)─(kali㉿kali)-[~/androwarn]
└─$ python androwarn.py -i ~/Downloads/tutorial-apks/diva-beta.apk -r html -v 3


[+] Androwarn version 1.6

[+] Loading the APK file...
[+] Analysis successfully completed and HTML file report available '/home/kali/androwarn/jakhar.aseem.diva_1717650630.html'
```

Step 5: Review the Output

Androwarn will generate a detailed report based on the analysis of the APK file. This report will highlight various aspects such as potential security issues, suspicious behaviors, and other vulnerabilities.

Step 6: Interpret the Results

Examine the report to understand the potential threats and vulnerabilities present in the APK file. The report will categorize findings and provide insights into what the APK might be doing.