Malware Family	Query
ADRD	Two receivers plus one service. 1. One receiver will launch the service; 2. The service will read deviceId, subscribeId, osVersion, networkInfo and encrypt them all, then send the encrypted string to internet. 3. Another receiver will start on "BOOT_COMPLETE" and it will launch the previous service periodically througth PendingIntent.
AnserverBot	1. Require a large portion of permission; 2. Start with a receiver which requries a high priority and 10 actions. 3. The receiver will abort broadcastor and collect deviceId, SubscribeId, SDK version and send them to file and internet.
BaseBridge	1. The original infected app tries to get root permission but no tainted flow. 2. For the malware being downloaded on runtime, it has one receiver and two services(AdSmsService, BridgeProvider and PhoneService); 3. One receiver has a high priority and an intent filter with 10 actions; 4. The receiver will abortBroadcast and launch another two services; 5. This services will read SMS and SubScribeID,Sim card information, getActiveNetworkInfo, OS version, manufacturer and send them to Internet.
DroidKungfu2	A isolated subgraph with: 1. Start with a receiver, then launch a service, finally launch an activity; 2. Receiver has an intent filter of "BOOT_COMPLETE"; 3. Service component read SDK version, deviceId, linenumber, Model and OS type, then write them all to the file name "mycfg.ini" and internet
DroidKungfu3	A isolated subgraph with: 1. Start with a receiver, then launch a service, finally launch an activity; 2. Receiver has an intent filter of "BOOT_COMPLETE"; 3. Service component read SDK version, deviceId, linenumber, Model and OS type, then send to internet.
Geinimi	1. A graph starts with a receiver, requires intent filter of "BOOT_COMPLETED" and category of "android.intent.category.LAUNCHER"; 2. The receiver then launches a service and this service will start itself again on its destroy() method; 3. The following info will be leaked to internet by the service: deviceId, deviceSoftwareVersion, LineNumber, networkISO, operator, operatorName, VoicemailNum, SubscribeId, SimSerial, SimOperator, SimOperatorName, SimCountryISO, phoneType, NetworkType, android.os.Build(Model, Brand, CPU_ABI, fingerprint, manufacturer, id, host)
GoldDream	1. A graph starts with a receiver, which requires intent-filter of "BOOT_COMPLETED", "SMS_RECEIVED", "PHONE_STATE" and "NEW_OUTGOING_CALL"; 2. The receiver reads SMS and phone number then writes to a file. 3. The service being launched will read the file and leak all the info to internet.
KMin	Type One: 1. A graph starts with a receiver, with high priority intent filters: "SMS_RECEIVED", "WAP_PUSH_RECEIVED"; 2. The receiver will real the SMS and dump to log. Type Two: 1. Start with a receiver on "BOOT_COMPLETE" and launch a service; 2. The MAIN activity will launch another activity which will dump the deviceID, SubscribeId and datetime to the log. Some apps will also leak to internet.
Pjapps	1. A isolated receiver with hight priority and requires action "SMS_RECEIVED" and do "abortBroadcast"; 2. Another receiver with "SIG_STR" and launches a service; 3. The service will register a new receiver listening "SMS_RECEIVE" with high priority. Also it will send IMSI, DeviceId, Sim, linenumber and mobile number to Internet. 4. The new registered receiver will also leak the above info. it also will launch another activity(Dialog screen) which will install a new apk if use clicks YES.
DroidDreamLight	(Apps are seriously obfuscated) 1. A receiver has an intent-filter for "PHONE_STATE" and default category. 2. The receiver will load a service; 3. The service will read deviceId, subscribeId, SDK version, package info of installed apps and then dump them all to a file and internet.(Use android.os.handler to pass sensitive data)