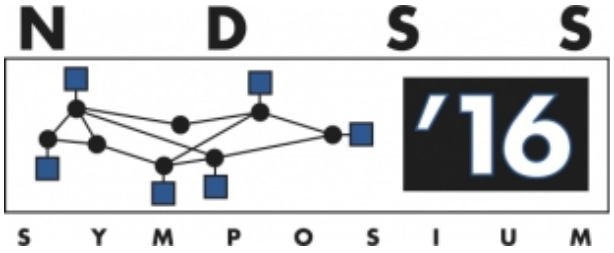
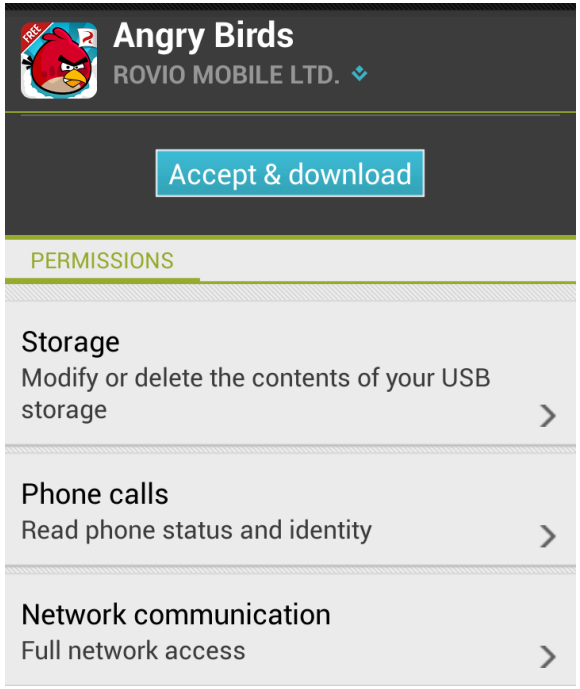
Life after App Uninstallation: Are the Data Still Alive?
Data Residue Attacks on Android

Xiao Zhang, Kailiang Ying, Yousra Aafer,
Zhenshen Qiu, and Wenliang Du

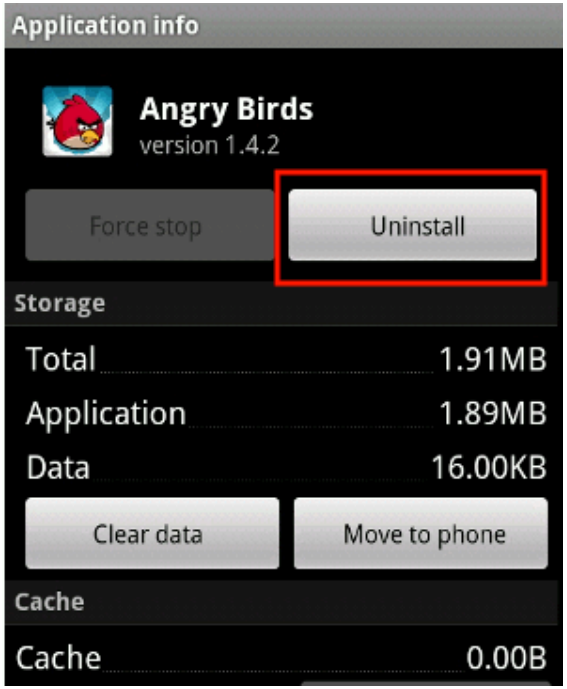


App Life



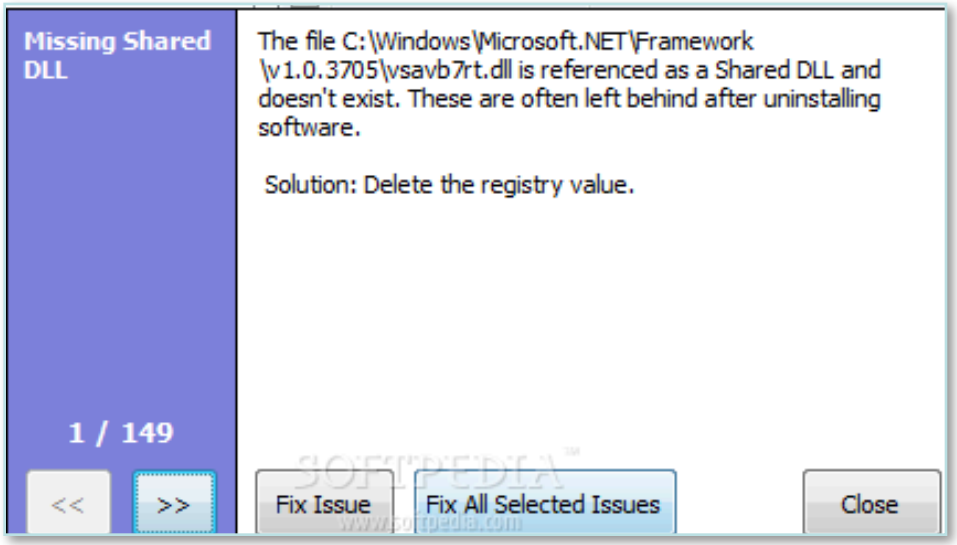
Installation

Interaction

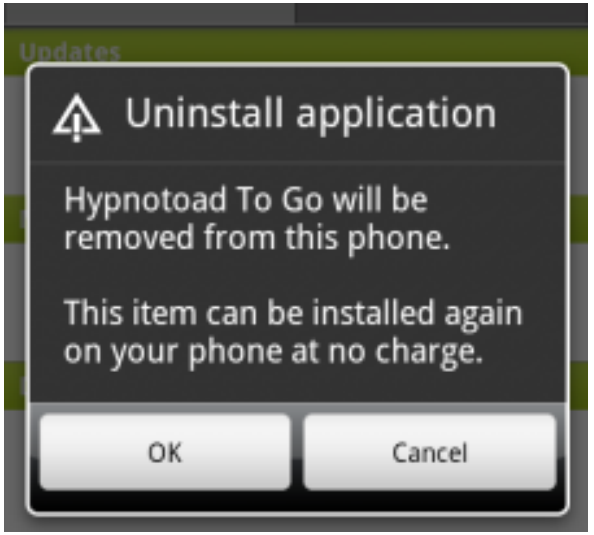


Uninstallation

But, what if …



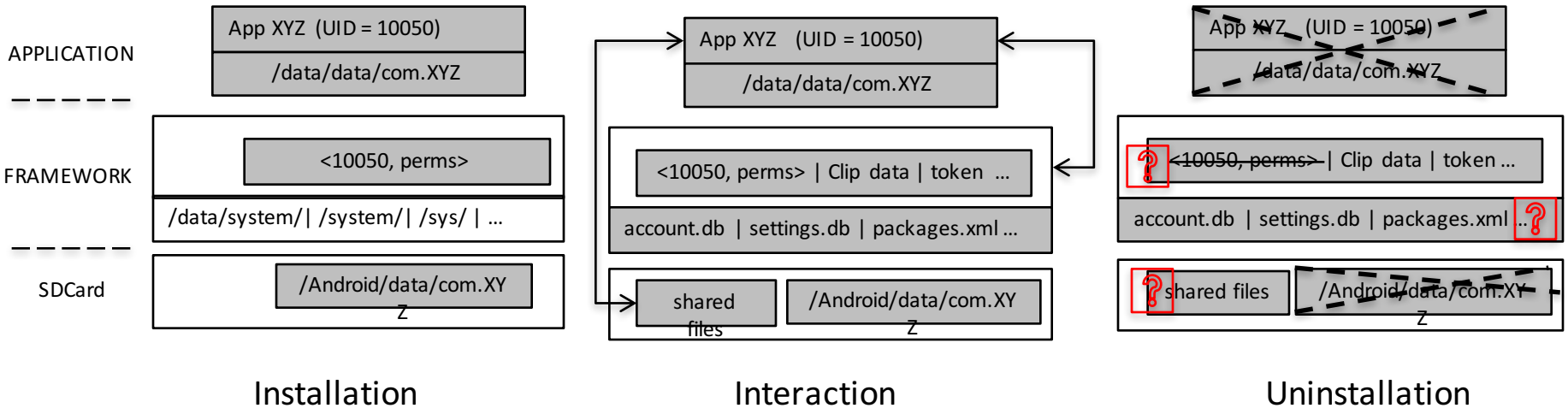
Windows Residue



Android App Uninstallation

Are there any data left after application uninstallation on Android?

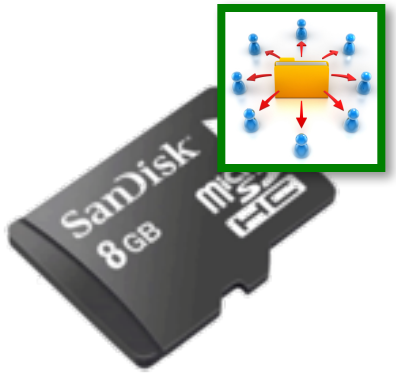
In Details

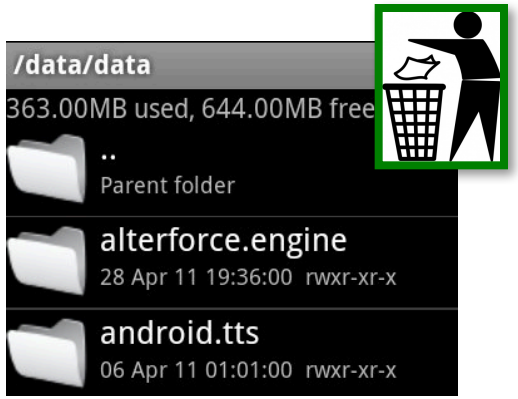


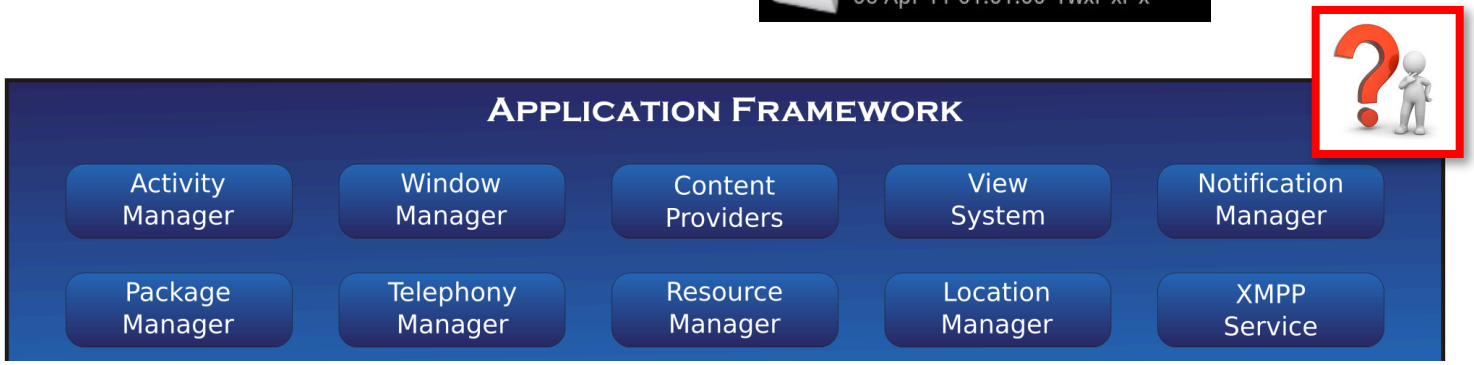
Installation Interaction Uninstallation

Are the data still alive after application uninstallation on Android?

What can go wrong?

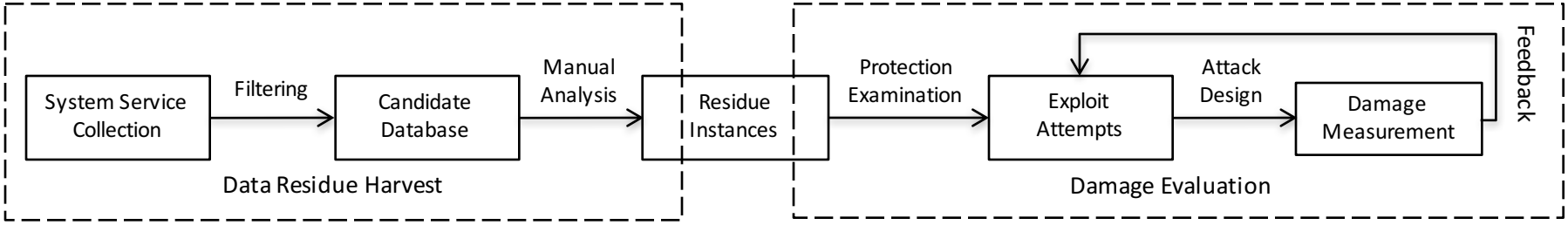


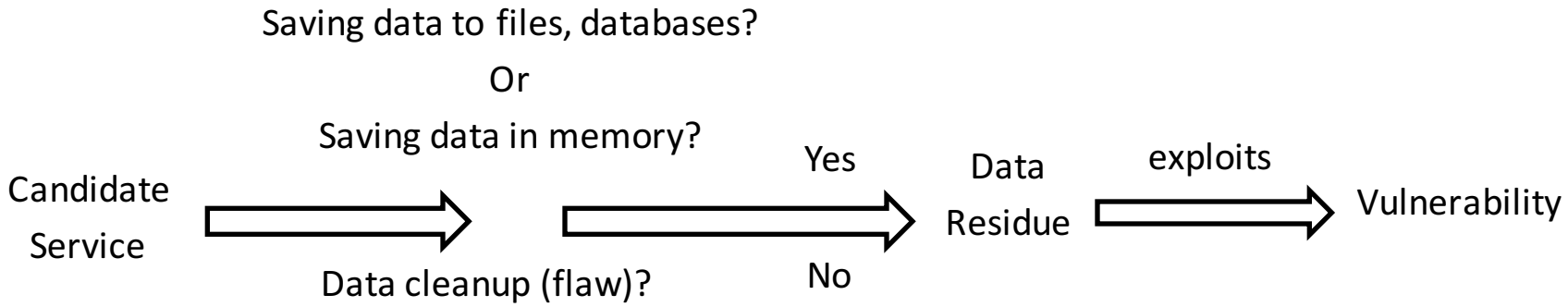




Are the data still alive in Android system services after application uninstallation?

Methodology





Findings

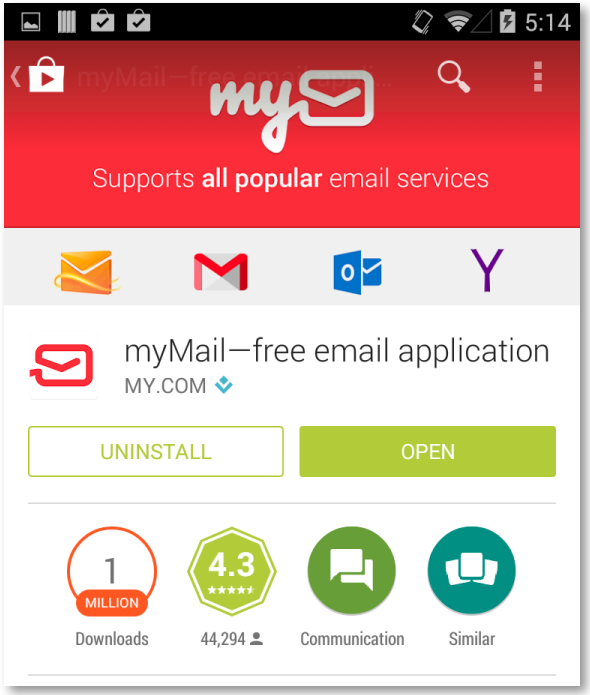
Samples (# Total/Candidate/Residue)	**Category**	**Service Instances**	**Residues**	**Exploitable**
System Services (96/96/10) System-app Services (161/26/2)	Credential Residue	AccountManager Keystore	User Credentials Public/Private Keypairs	✓ ✓†
	Capability Residue	Clipboard ActivityManager	URI PendingIntent	✓ ✗
	Settings Residue	TextService DebugService DreamService TrustAgent LocationManager	User Selected Components	✓ ✓ ✓ ✓ ✓
	History Residue	PrintService DownloadService	Print/Download Information	✓ ✓†
	Permission Residue	PackageManager	Permissions	✓

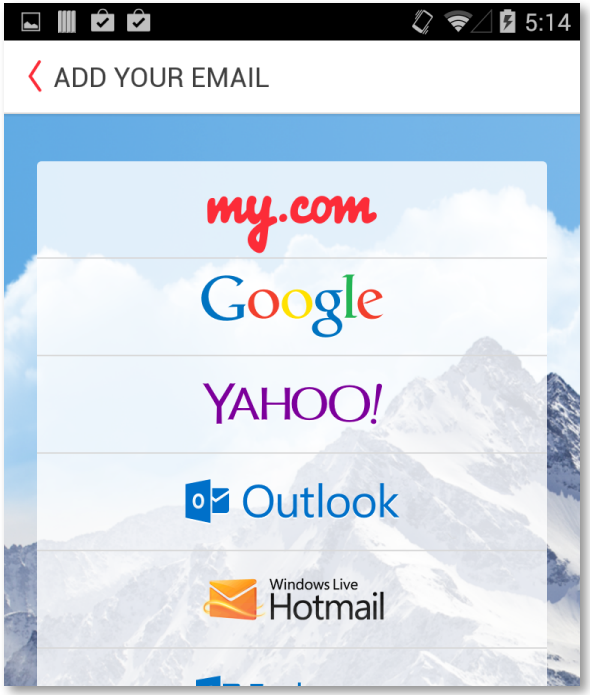
† Resolved on Android Lollipop, but reproducible on KitKat and prior versions

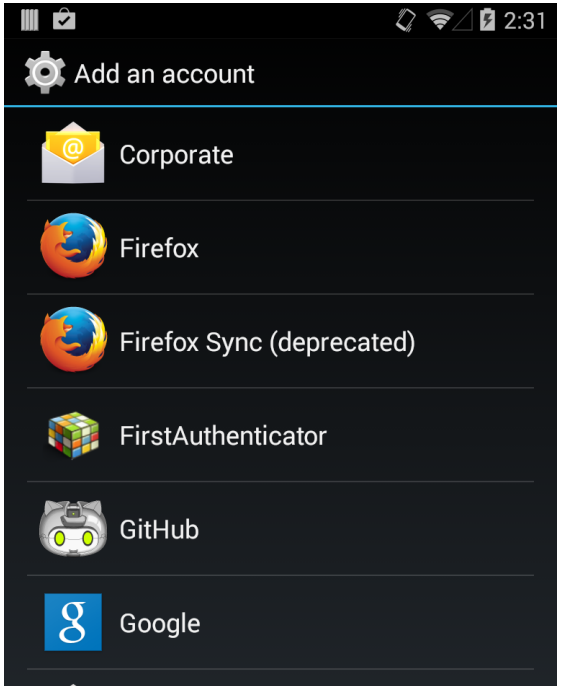
- 7 security vulnerabilities acknowledged by Google with Medium priority

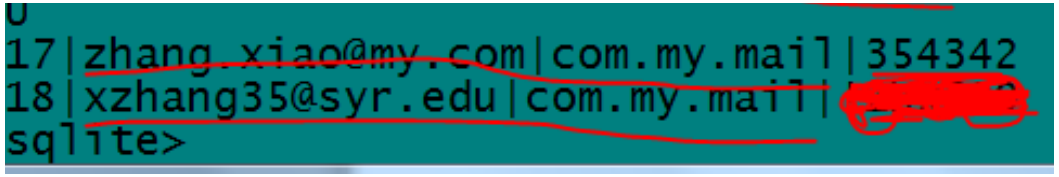
Sample Exploits - I

- ***Credential Stealing***



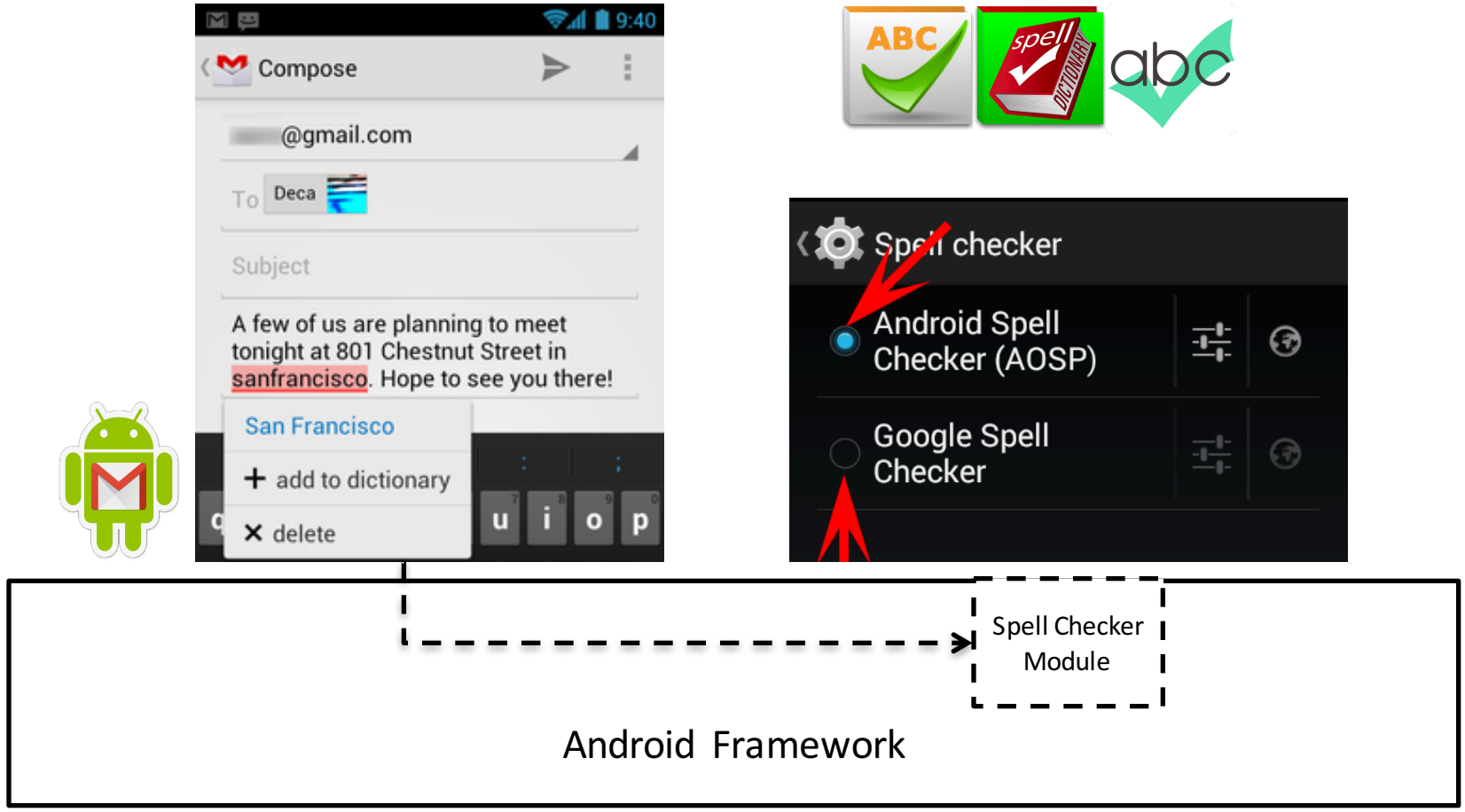






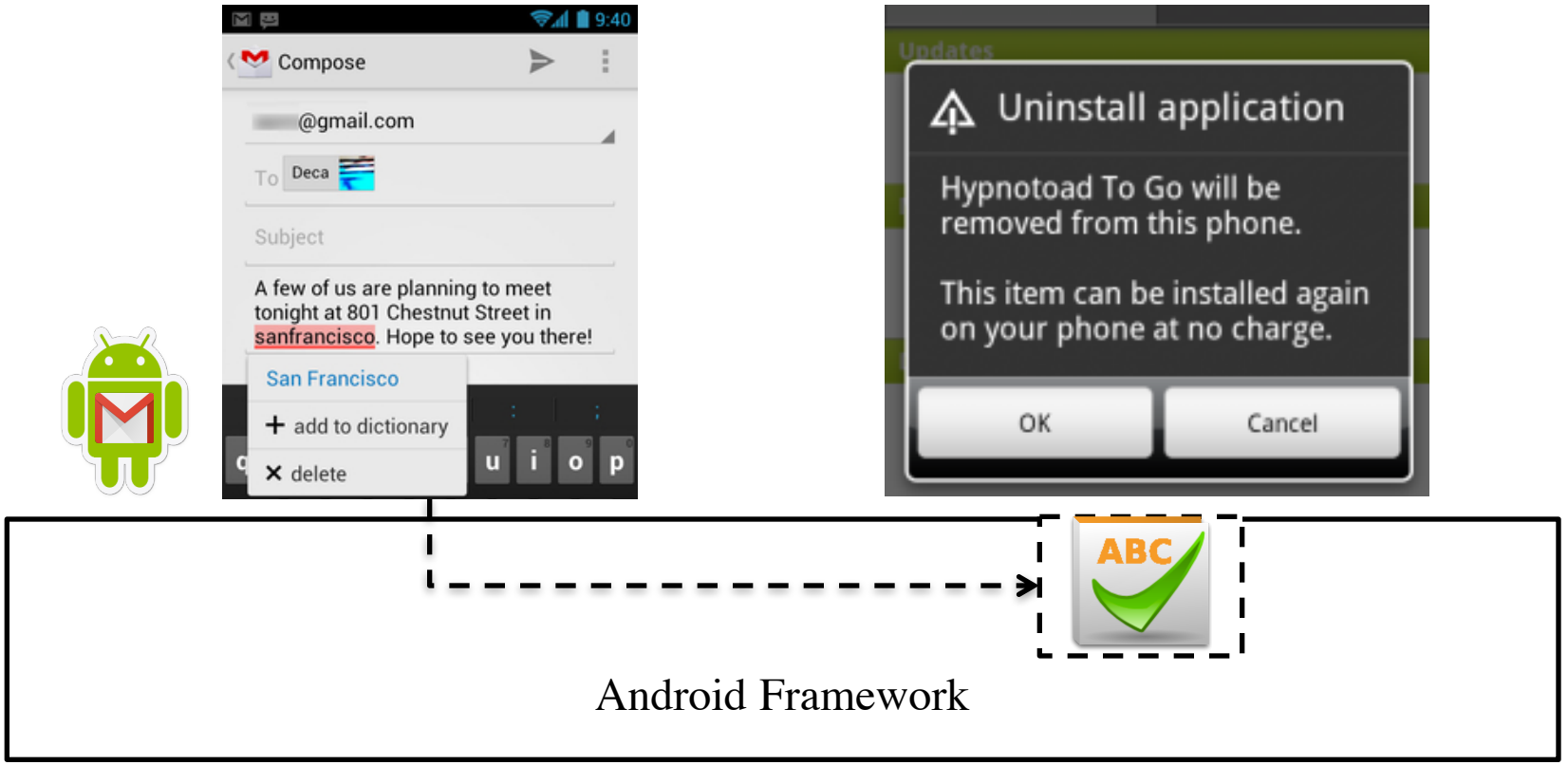
Sample Exploits - II

- ***Settings Impersonating***



Sample Exploits - II

- ***Settings Impersonating***



Even More …

Details are available at:
https://sites.google.com/site/droidnotsecure/

Evaluation

- 2,373 apps
- 10 devices
- 8 Android versions
- 3 play stores

	package	account type	authority
GooglePlay	✗	✓	✓
Amazon Appstore	✗	✓	✓
Samsung Appstore	✗	✓	✓

Attack Instances		Account	Clipboard	Download	Dream	Keystore	Permission	Print	Spell Checker
I: Analysis on Real-world Applications									
# Targets		131	92	17	24	63	55	49	16
II: Examination on Essential Attributes									
Attributes		account type	authority	UID	package	UID	sharedUserId	UID/package	package
III: Measurement on Device Customization Influence[†]									
LG Nexus 4	5.1.0	✓	✓	✗	✓	✗	✓	✓	✓
Galaxy Nexus	4.3	✓	✓	✓	✓	✓	✓	N/A[1]	✓
ASUS Nexus 7 (2013)	5.1.1	✓	✓	✗	✓	✗	✓	✓	✓
Samsung Nexus S	4.1.2	✓	✓	✓	N/A[1]	N/A[1]	✓	N/A[1]	✓
LG Nexus 5	5.0.1	✓	✓	✗	✓	✗	✓	✓	✓
Samsung Tab 10.1	4.0.4	✓	✓	✓	N/A[1]	✗	✓	N/A[1]	N/A[2]
HuaWei Y321	4.1.2	✓	✓	✓	N/A[1]	N/A[1]	✓	N/A[1]	N/A[2]
Moto X (2014)	5.0.0	✓	✓	✗	✓	✗	✓	✓	✓
Samsung Note 8.0	4.4.2	✓	✖	✓	✓	✓	✓	N/A[1]	N/A[2]
LG G3	5.0.0	✓	✓	✗	✓	N/A[1]	✓	✓	N/A[2]

[†] N/A[1]: feature Not Available because of the low Android version; N/A[2]: feature Not Available because of the vendor customization.

Fundamental Causes

- Data Residue Instances <-> *Mandatory Design Principle in Backend*

- Exploits <-> *Signature-based Frontend*

Layers	Attributes	Assumptions	Protection Effectiveness	Breaking Conditions
Kernel	PID	process isolation	Hard Isolation	—
Framework	UID	UID exclusion	individual device cycle	device rebooting
Application	package	package exclusion	individual device state	(un)installation
Component	account type	customized-id exclusion	Invalid	(un)installation
	authority	customized-id exclusion	individual device state	(un)installation

TABLE IV: Security Examination of Android Attributes Used in Protecting Data Residue

Limitation

- Manual Analysis
- Static Analysis
  - App Level
  - Intelligence
- Dynamic Analysis
  - App Level
  - Exploit Conditions

```
private class TextServicesMonitor extends PackageMonitor {
    @Override
    public void onSomePackagesChanged() {
      synchronized (mSpellCheckerMap) {
        buildSpellCheckerMapLocked(mContext, mSpellCheckerList,
mSpellCheckerMap);
        // TODO: Update for each locale
        SpellCheckerInfo sci = getCurrentSpellChecker(null);
        if (sci == null) return;
        final String packageName = sci.getPackageName();
        final int change = isPackageDisappearing(packageName);
        if (// Package disappearing
            change == PACKAGE_PERMANENT_CHANGE || change ==
PACKAGE_TEMPORARY_CHANGE
            // Package modified
            || isPackageModified(packageName)) {
          sci = findAvailSpellCheckerLocked(null, packageName);
          if (sci != null) {
            setCurrentSpellCheckerLocked(sci.getId());
          }
        }
      }
    }
  }
```

Conclusion

- Data Residue Vulnerability
- Systematic Study
- Comprehensive Evaluation

- Trigger more research efforts

Questions?

xzhang35@syr.edu

https://sites.google.com/site/droidnotsecure/