Towards Discovering and Understanding Task Hijacking in Android

CHUANGANG REN, YULONG ZHANG, HUI XUE,
TAO WEI, PENG LIU

PENN STATE UNIVERSITY, FIREEYE INC.

PENNSTATE
1855

Android Multi-tasking

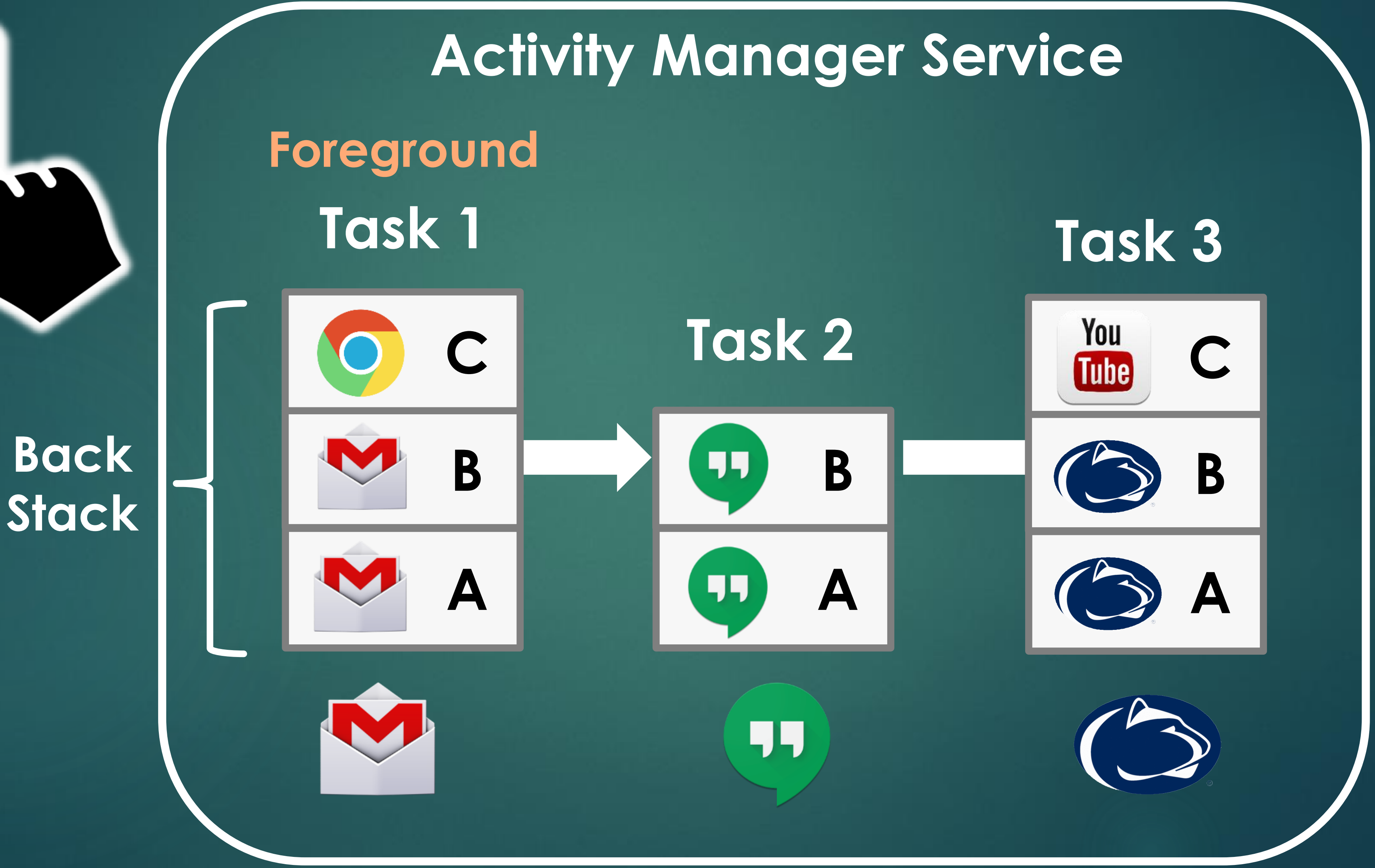
- In **PC world**, multitasking means multiple processes are running at the same period of time.
- In **Android**, multitasking is a different concept:

"*A task is a collection of activities that users interact with when performing a certain job*"

- Android developer documentation

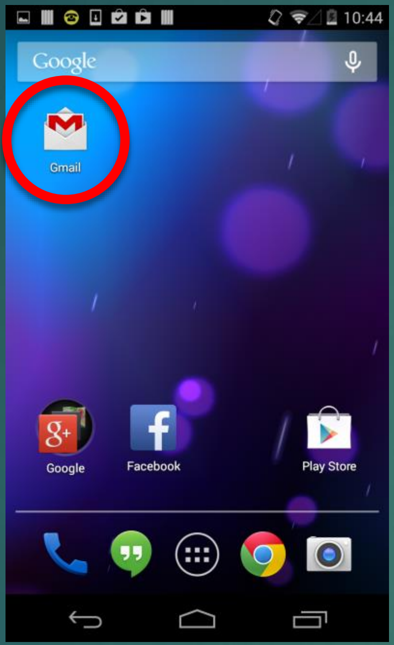
PENNSTATE 1855

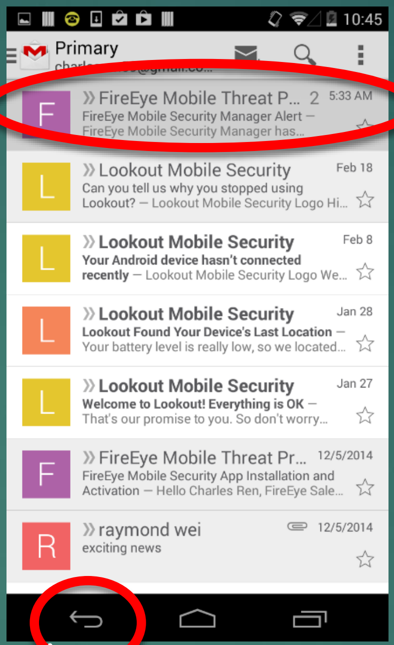
Android Multi-tasking



Lifecycle of Android Task

User Screen





» FireEye Mobile Threat P... 2 5:33 AM
FireEye Mobile Security Manager Alert – FireEye Mobile Security Manager has...

» Lookout Mobile Security Feb 18
Can you tell us why you stopped using Lookout? – Lookout Mobile Security Logo Hi...

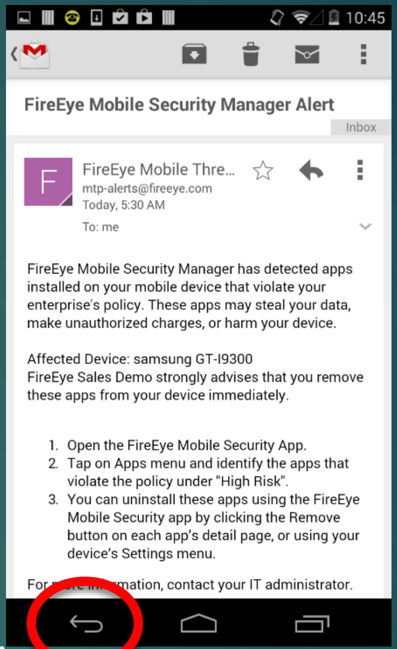
» Lookout Mobile Security Feb 8
Your Android device hasn't connected recently – Lookout Mobile Security Logo We...

» Lookout Mobile Security Jan 28
Lookout Found Your Device's Last Location – Your battery level is really low, so we located...

» Lookout Mobile Security Jan 27
Welcome to Lookout! Everything is OK – That's our promise to you. So don't worry...

» FireEye Mobile Threat Pr... 12/5/2014
FireEye Mobile Security App Installation and Activation – Hello Charles Ren, FireEye Sale...

» raymond wei 12/5/2014
exciting news



FireEye Mobile Security Manager Alert

FireEye Mobile Thre...
mtp-alerts@fireeye.com
Today, 5:30 AM
To: me

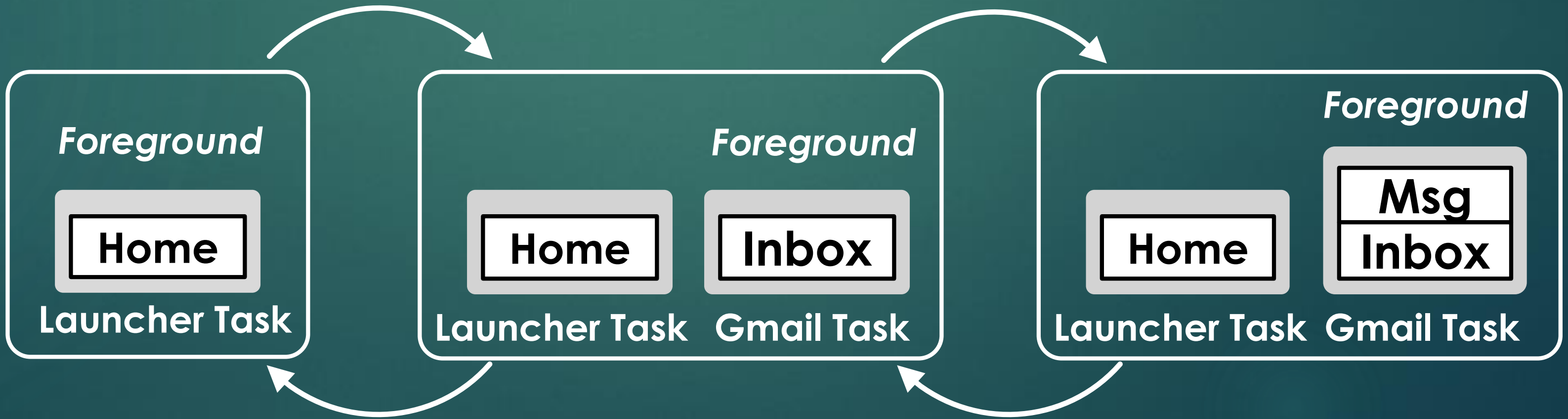
FireEye Mobile Security Manager has detected apps installed on your mobile device that violate your enterprise's policy. These apps may steal your data, make unauthorized charges, or harm your device.

Affected Device: samsung GT-I9300
FireEye Sales Demo strongly advises that you remove these apps from your device immediately.

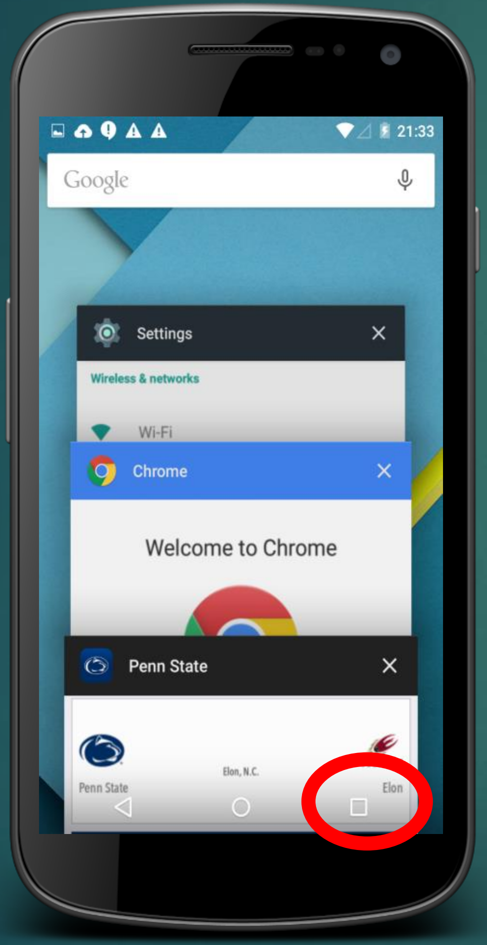
1. Open the FireEye Mobile Security App.
2. Tap on Apps menu and identify the apps that violate the policy under "High Risk".
3. You can uninstall these apps using the FireEye Mobile Security app by clicking the Remove button on each app's detail page, or using your device's Settings menu.

For [illegible]mation, contact your IT administrator.

Task State



Android Multi-tasking



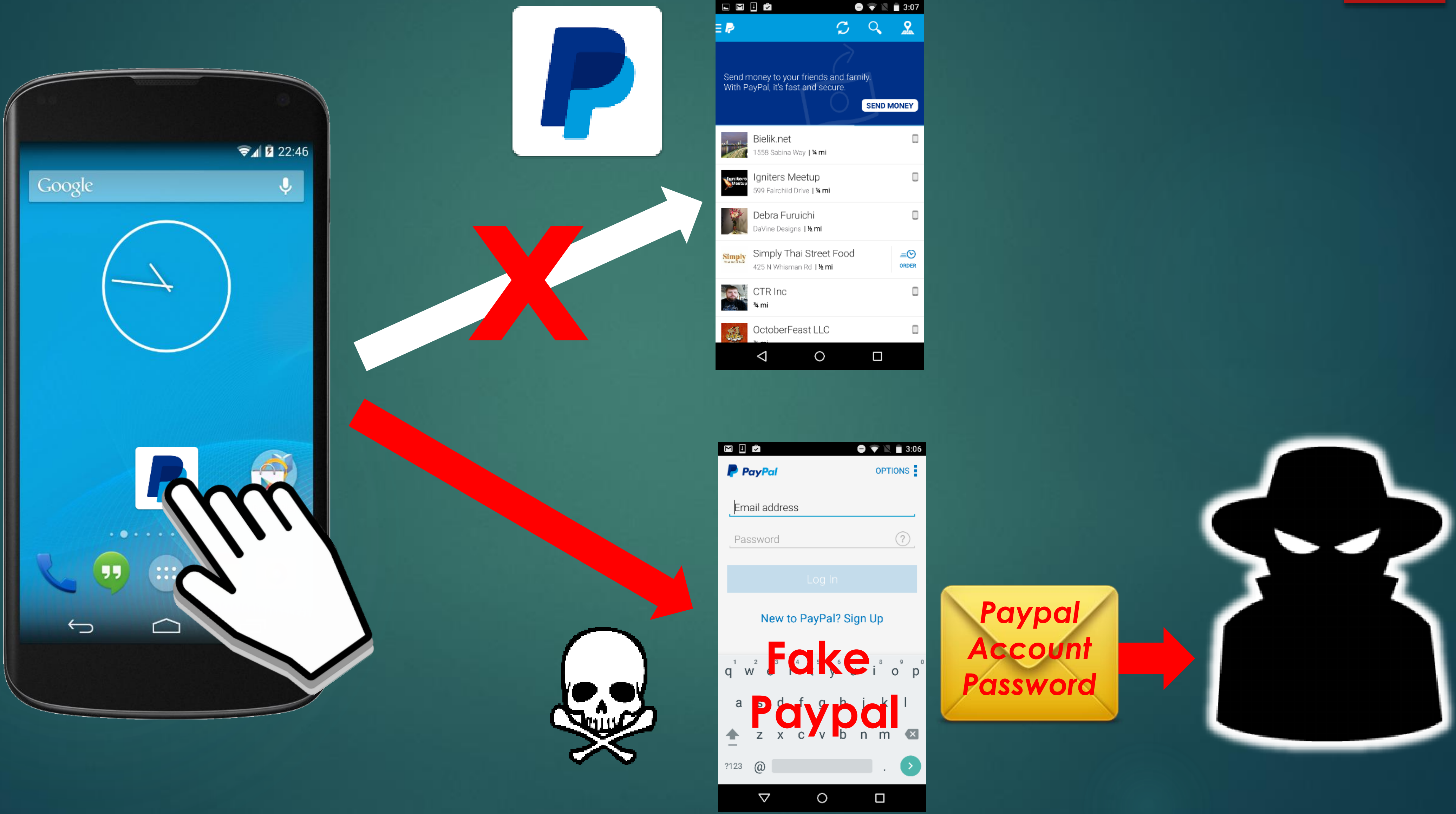
- ✓ Task switching
- ✓ UI navigation
- ✓ Task customization

PENNSTATE
1855

Security Concerns

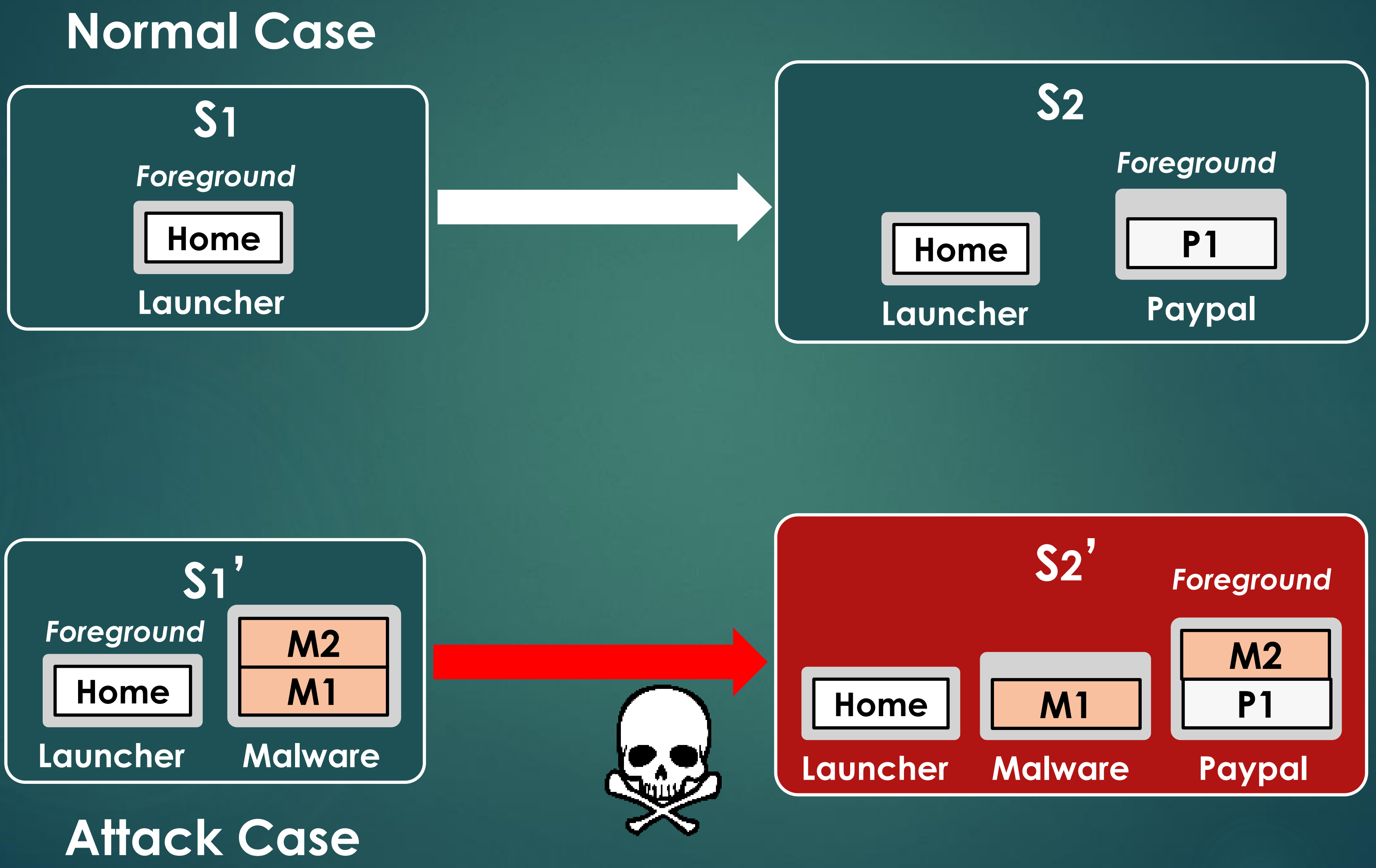
- However, the security implication of Android multitasking remains under-investigated
  - Android allows activities from different apps to reside in the same task (or back stack)
  - Android offers developers great flexibility to customize task behaviors
- We find that Android multitasking is plagued by a serious security risk – ***task hijacking***

Example - User Spoofing



PENNSTATE
1855

User Spoofing Attack



PENNSTATE
1855

How does mal-activity migrate?

- The malware tricks the system to relocate the malicious activity (M2) to the Paypal task by manipulating the following task control knobs:
  - **Task affinity**
  - **allowTaskReparenting**

Task Affinity

- An activity attribute defined in each <activity> tag in AndroidManifest.xml
- Task affinity specifies which task that the activity desires to join. By default, all activities in an app have the same affinity – the app package name

```
<manifest xmlns:android="http://schemas.android.com/apk/res/android"
    package="com.example.app“ >

    <application>
        <activity android:name=".ActivityA “/>
        <activity android:name=".ActivityB“ android:taskAffinity="com.example.app:taskB “/>

    </application>
</manifest>
```

Task Affinity

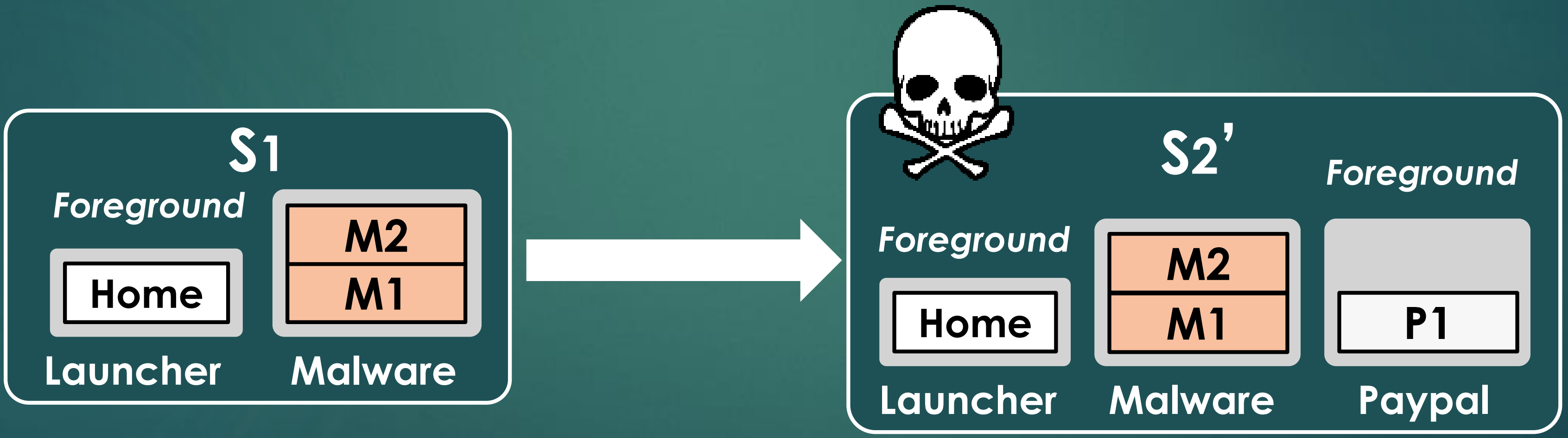
- Developer can re-define the task affinity in order to achieve desirable task behavior
  - Group activities into different tasks
  - Place activities defined in different apps within the same task

- If **<allowTaskReparenting** = “**true”>** for activity A, and when a task with the same affinity as A is brought to the front, the system would move the “relocatable” activity A from its original hosting task to this new foreground task

User Spoofing Attack

Malware abuses the following task control knobs:

1. Activity M2: taskAffinity = **com.paypal.android**
2. Activity M2: allowTaskReparenting = **true**



PENNSTATE

1855

Research Questions

- Question 1: How many types of task hijacking**?**
- Question 2: How to craft the individual attacks**?**
- Question 3: How to assess the vulnerability**?**
- Question 4: How to defend task hijacking**?**

Task Control Knobs

- We find that there are a rich set of task control knobs that can be abused by a task hijacking attack
- Task control knobs in 4 categories:

Intent Flag	Activity Attribute	Call-back Function	Framework API

NEW_TASK
SINGLE_TOP
REORDER_TO_FRONT
NO_HISTORY
CLEAR_TASK
NEW_DOCUMENT
MULTIPLE_TASKS
	launchMode
	taskAffinity
	allowTaskReparenting
	documentLaunchMode
	FinishOnTaskLaunch
		onBackPressed()	TaskStackBuilder class
			startActivity()
			startActivities()

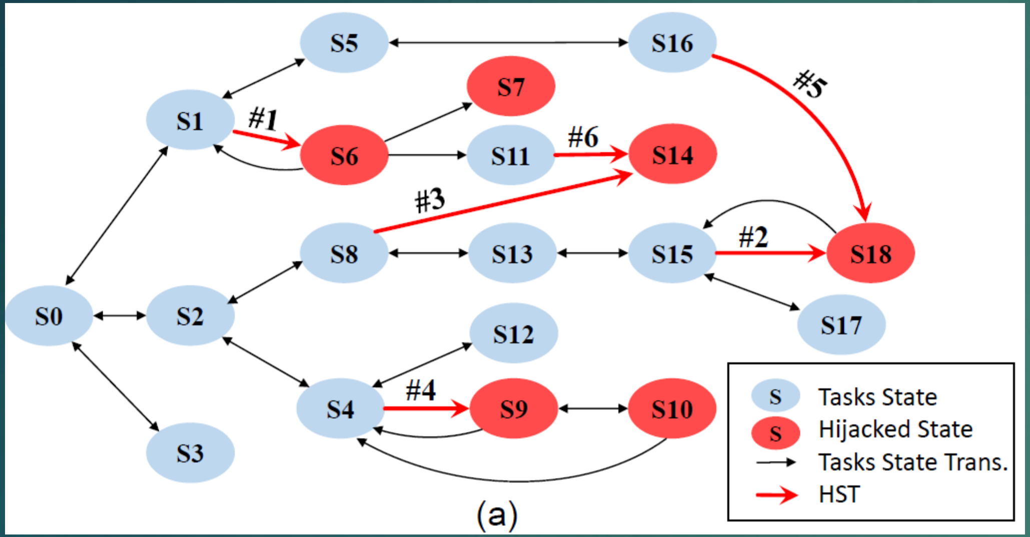
Methodology

- We formalize the task dynamic as a state transition model
  - **Hijacked task state**: at least one task in the system contains both malicious and benign activities
  - **Hijack state transition (HST)**: state transition that leads the system to a hijacked task state
- We simulate an Android system with three apps
  - Two benign apps (A, B), one malware (M)
  - Connect task states and generate task state transition graph
  - Flag the hijacked task states and HST in the graph

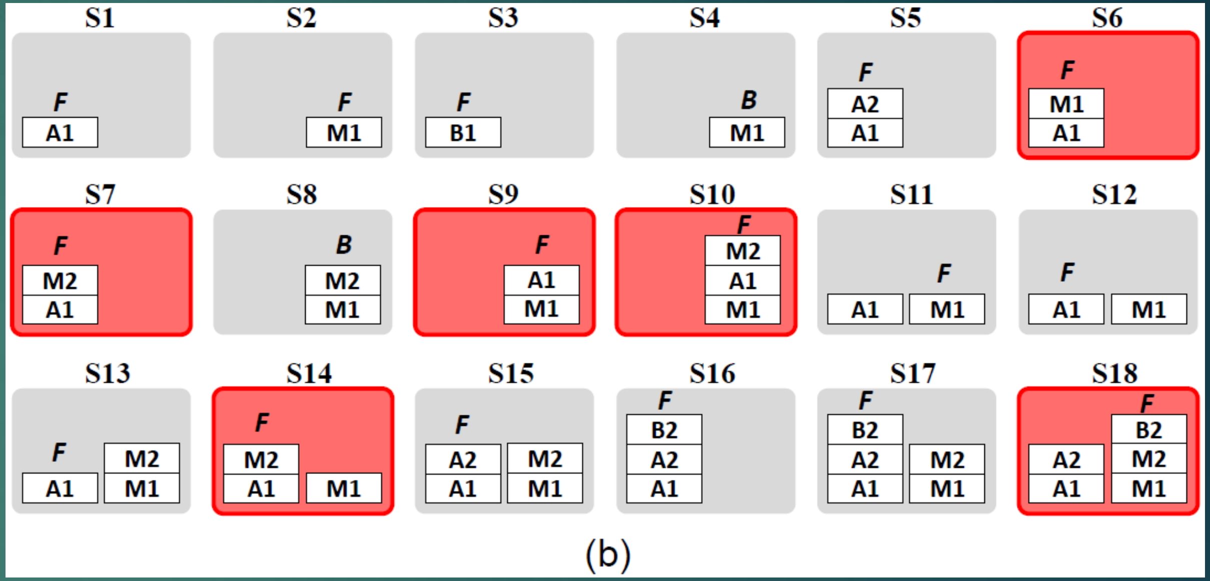
PENNSTATE

Question 1: Types of Task Hijacking

Task State Transition Graph



Task States



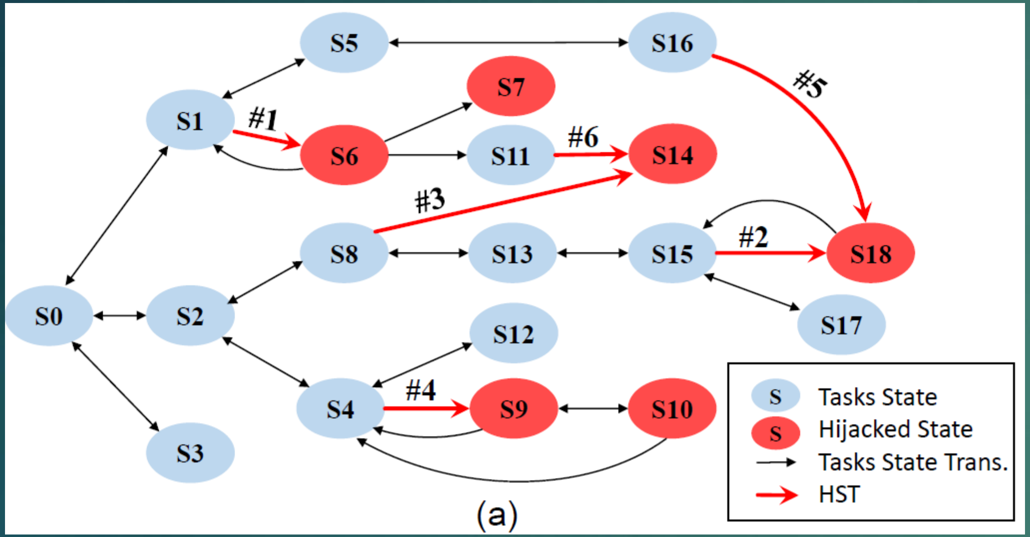
Two types of Hijacking State Transitions (HST):

- Malware activity moves to benign app task
- Benign activity is placed into malware task

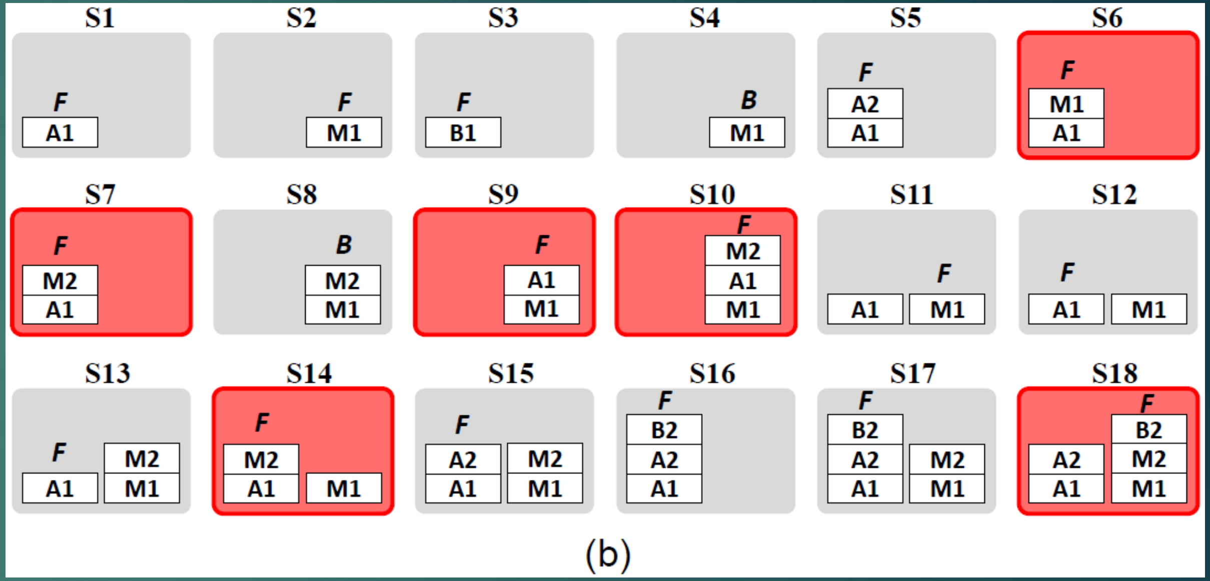
PENNSTATE

Question 1: Types of Task Hijacking

Task State Transition Graph



Task States



Observations:

- There are many possible hijacking state transitions (HSTs)
- Once exploited, the HSTs could result in practical and serious real-world attacks

PENNSTATE 1855

Question 2 – Enabled attacks

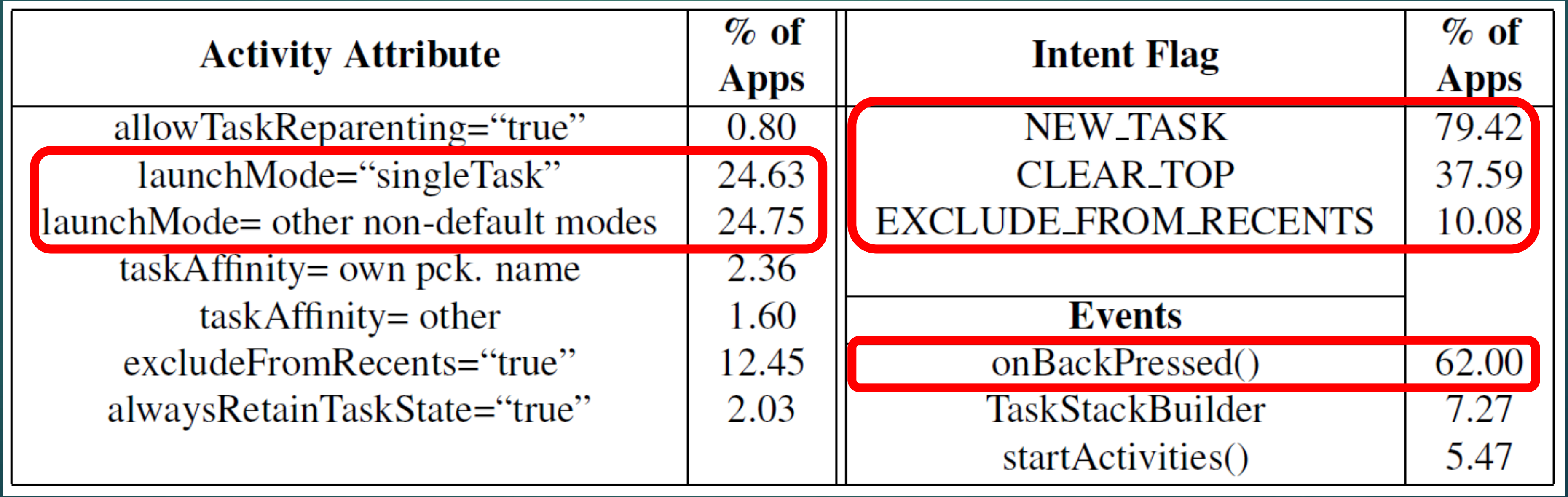
We implemented 6 proof-of-concept attacks in 3 categories:

Attack Category	Consequence	Attack Name	Vulnerable Systems & Apps
User Spoofing	**Sensitive information stolen**	Spoofing attack	**all ; all**
		Phishing attack (I – III)	**all ; some apps**
Denial-of-service	**App function disabled; Restriction of user access**	Ransomware	**>Android 5.0 ; all**
User Monitoring	**User privacy infringement**	Spyware	**>Android 5.0 ; all**

Task hijacking attacks affect all latest Android versions and apps, including the most privileged apps!

Question 3: Vulnerability Assessment

- We would like to first understand the use of security-sensitive task control knobs in real implementation
- We analyze 6.8 million apps from Google Play and other 12 popular third-party app markets



Activity Attribute	**% of Apps**	**Intent Flag**	**% of Apps**
allowTaskReparenting="true"	0.80	NEW_TASK	79.42
launchMode="singleTask"	24.63	CLEAR_TOP	37.59
launchMode= other non-default modes	24.75	EXCLUDE_FROM_RECENTS	10.08
taskAffinity= own pck. name	2.36		
taskAffinity= other	1.60	**Events**	
excludeFromRecents="true"	12.45	onBackPressed()	62.00
alwaysRetainTaskState="true"	2.03	TaskStackBuilder	7.27
		startActivities()	5.47

PENNSTATE

Case Study – Task Affinity

- 1.6% (109K apps) of all apps set the task affinity without containing their own package name
- These apps may interfere with the multitasking behaviors of other apps
  - *Unintentional*: careless app developers who are unaware of the security implications.
  - *Intentional*: task affinity intentionally set to popular app's package name in order to implement legitimate "add-on" feature for these popular apps.
  - We have not found evidence that malware has already abused these task control knobs

PENNSTATE
1855

Question 4: Defense Suggestions

- Detection in app review process
  - App review guideline may contradict with existing app features
  - Challenging to detect stealthy dynamic behaviors of an advanced malware

- More secure multi-tasking mechanism
  - Introduce additional security features for multitasking control
  - For example, task affinity should comply with certain name space specification
  - Introduce additional Boolean attribute to control if the app allow other apps to specify the same task affinity

Proof-of-concept Attack Demo

- Phishing attack
  - A malware can steal user Citi Bank account name and password by hijacking citi bank task with a spoofing Citibank login interface

- Denial of service
  - A malware can disable app uninstallation in a system
  - The similar attack approach could be used to create a ransomware

PENNSTATE
1855

Thank you!