Closer Look at Mobile Hybrid Apps Configurations: Statistics & Implications

Abeer AlJarrah*, Mohamed Shehab†

* Department of Computer Science
San Francisco State University

† Department of Software & Information Systems
University of North Carolina at Charlotte
Outline

- Introduction
- Cordova Apps Configurations
- Threat Model & Configurations role
- Statistics & Implications
- Concluding Remarks
Outline

• Introduction

• Problem Statement and Hypothesis

• The tool: CordovaConfig

• User study and Evaluation

• Concluding Remarks
Mobile Apps Are Not the Same

Native Apps
- Apps are platform specific
- ~90% of all mobile internet traffic
- Full potential of the platform can be leveraged
- Advanced Graphics and smooth UI transitions

HTML5 Apps
- Use standard web technology - not really apps…
- Minimal access to device resources (browser)
- Many Libraries to enhance the look-and-feel

Hybrid Apps
- Mix of Both!
- Same code-base runs on different platforms
- Web-based code have access to device resources
Cordova Library

- Cordova is a popular mobile application development framework originally created by Nitobi. Adobe Systems purchased Nitobi in 2011, rebranded it as PhoneGap, and later released an open source version of the software called Apache Cordova.

- Many other tools and frameworks are also built on top of Cordova, including Ionic, Monaca, TACO, the Intel XDK, and the Telerik Platform.

- Supports 8 platforms, 3726 plugins
Platform Security Model

- Model is based on Domain Whitelisting

- Each app contains a global configuration file (*config.xml*) that controls several aspects of the application behavior

- The file contains policies to controls access to external domains over which the application has no control

- Content-Security policy in HTML pages

- The latest version supports whitelisting:
  - Set of external domains the app is allowed to communicate with.
  - URLs the WebView itself can be navigated to.
  - URLs the app is allowed to ask the system to open
  - Set of plugins to be accessed
Default config.xml

```xml
<widget id="com.phonegap.helloworld" version="1.0.0">
  <name>Hello Cordova</name>
  <description>A sample Apache Cordova app</description>
  <author email="aaljarra@uncc.edu" href="http://liisp.uncc.edu">
    LIISP Team
  </author>
  <content src="index.html" />
  <plugin name="cordova-plugin-whitelist" spec="1" />
  <access origin="*" />
  <allow-intent href="http://*/*" />
  <allow-intent href="https://*/*" />
  <allow-intent href="tel:*" />
  <allow-intent href="sms:*" />
  <allow-intent href="mailto:*" />
  <allow-intent href="geo:*" />
  <platform name="android">
    <allow-intent href="market:*" />
  </platform>
  <platform name="ios">
    <allow-intent href="itms:*" />
    <allow-intent href="itms-apps:*" />
  </platform>
</widget>
```

Default index page that contains a default CSP

- ✓ Browser navigation to everything
- ✓ Dialer
- ✓ SMS
- ✓ Built-in Email
- ✓ Built-in Maps

Target Android & iOS
config.xml Evolution

version 1.5

More fine-grained settings

Towards More Least Privilege

Default CSP

version 8.x
Outline

• Introduction

• Problem Statement and Hypothesis

• The tool: CordovaConfig

• User study and Evaluation

• Concluding Remarks
App Configuration suffers the following limitations:

- Coarse-grained
- Risky Defaults

Developers seem to lack the knowledge/time/willingness to securely configure their apps.

<table>
<thead>
<tr>
<th>OWASP Top-10 Application Security Risks-2017</th>
</tr>
</thead>
<tbody>
<tr>
<td>A1: Injection</td>
</tr>
<tr>
<td>A2: Broken Authentication &amp; Session Mgmt.</td>
</tr>
<tr>
<td>A3: Sensitive Data Exposure</td>
</tr>
<tr>
<td>A4: XML External Entities (XXE)</td>
</tr>
<tr>
<td>A5: Broken Access Control</td>
</tr>
<tr>
<td>A6: Security Misconfiguration</td>
</tr>
<tr>
<td>A7: Cross-Site Scripting (XSS)</td>
</tr>
<tr>
<td>A8: Insecure Deserialization</td>
</tr>
<tr>
<td>A9: Using Components with Known Vulns</td>
</tr>
<tr>
<td>A10: Insufficient Logging &amp; Monitoring</td>
</tr>
</tbody>
</table>
We proposed and developed a configuration tool called CORDOVACONFIG, that guides developers to enhancing the securing and configuring hybrid apps.

H1: Using CORDOVACONFIG changes developers’ mental model in regard to understanding the boundaries of the required privileges of an app.

H2: Using CORDOVACONFIG improves developers’ understanding of a specific app’ configurations.
Outline

- Introduction
- Problem Statement and Hypothesis
- The tool: CordovaConfig
- User study and Evaluation
- Concluding Remarks
Cordova Config Tool

- Through both static and dynamic analysis Cordova Config monitors the app usage behavior and extracts the safe configurations during testing and development phase.
  - Generate policies per app page
  - Derive the white list domains
  - Provide page based policies
- We provide a Web-based Tool: To be used by the developer to view/edit/confirm captured behavior.

Cordova Config Tool - Overview
Cordova Config Tool

• The web tool provides educational information about the accesses of the app.

• The tool also provides security controls to edit the derived policy.
• **Educational**: Scan misaligned plugins and explain the impact of these plugins.

---

**Your App Configuration Analysis - Part 1**

**Plugins Found in your app config.xml**

```
<feature name="Geolocation">
  <param name="android-package" value="org.apache.cordova.geolocation.Geolocation" />
</feature>

<feature name="Camera">
  <param name="android-package" value="org.apache.cordova.camera.CameraLauncher" />
</feature>

<feature name="Contacts">
  <param name="android-package" value="org.apache.cordova.contacts.ContactManager" />
</feature>

<feature name="Capture">
  <param name="android-package" value="org.apache.cordova.mediacapture.Capture" />
</feature>

<feature name="Battery">
  <param name="android-package" value="org.apache.cordova.BatteryListener" />
</feature>
```

Which means accessing:

- Geolocation
- Video/Audio capture
- Camera
- Contacts

Next
**Educational**: Scan default navigation and explain impact

**Your App Configuration Analysis - Part 2**

**URLs the app is allowed to ask the system to open:**

1. `<allow-intent href="market:" />
2. `<allow-intent href="http://*:*" />
3. `<allow-intent href="https://*:*" />

Which means accessing:

- Any Android Market
- Allow links to any http page to open in a browser
- Allow links to any https page to open in a browser

**External apps the app is allowed to ask the system to open:**

1. `<allow-intent href="tel:*" />
2. `<allow-intent href="sms:*" />
3. `<allow-intent href="mailto:*" />
4. `<allow-intent href="geo:*" />

Which means accessing:

- Allow to open the dialler
- Allow to open the messages app
- Allow to open the email app
- Allow to open the maps
**Educational:** Scan default network access and native permissions and explain the impact

Your App Configuration Analysis - Part 3

Network Requests (images, JS, etc) are allowed to be made:

```xml
<access origin='*' />
```

Which means accessing:
- Don’t block any request

URLs the WebView can be navigated to:

No configuration found!

- By default, WebView is allowed to navigate to local files only

Android Permissions accesses are:

```xml
<uses-permission android:name="android.permission.ACCESS_COARSE_LOCATION" />
<uses-permission android:name="android.permission.ACCESS_FINE_LOCATION" />
<uses-permission android:name="android.permission.WRITE_EXTERNAL_STORAGE" />
<uses-permission android:name="android.permission.READ_CONTACTS" />
<uses-permission android:name="android.permission.WRITE_CONTACTS" />
<uses-permission android:name="android.permission.GET_ACCOUNTS" />
<uses-permission android:name="android.permission.RECORD_AUDIO" />
<uses-permission android:name="android.permission.RECORD_VIDEO" />
```

Which means accessing Android System:
- Location – An app can use the device’s location. This includes two different location settings, approximate and precise. Using the approximate location option, the app gets the device’s location from the network, and the precise option uses the device’s GPS and network to determine the location. To do this, the permission allows the app to access extra location provider commands and GPS.
- Photos/Media/Files – The application has the ability to use the file on the device with the application installed. This includes reading and writing to the SD card and USB storage. The app can also mount and unmount external storage as well as format external storage. This permission deals with reading external storage on newer devices
- Contacts – find accounts on the device, see and modify the owner’s contact card and add or remove contacts from the device
- Video/Audio - capturing and encoding a variety of common audio/ video formats integrated into your applications

Next
• **Navigation and Access:** Show captured behavior in terms of
  - State
  - Plugins accessed
  - Screen shot of the app
  - Native permissions needed to call the plugins

---

CORDOVACONFIG Tool

**Add more plugins**

**Confirm Add more States/Plugin**
**Navigation and Access:** Show captured behavior on terms of
- External URLs accessed
- External apps called
- URL Navigation
Outline

• Introduction

• Problem Statement and Hypothesis

• The tool: CordovaConfig

• User study and Evaluation

• Concluding Remarks
User Study Flow

- **Step-1:** Participants are given a partially done app (code and configuration) and they are asked to update the app and complete the missing features, which also involved configuring the app settings.

- **Step-2:** Test the app and make sure it satisfies all the requirements.

- **Step-3:** Answer a Pre-Survey

- **Step-4:** Participants given the same app (linked to our tool) and asked to test it according to given test scenarios and use CordovaConfig to monitor and confirm the app configurations.

- **Step-5:** Answer Post-Survey

- **Step-6:** Interview participants (Open ended questions)
The app used in the user study is a single-page app which provides an employee directory service.

- Uses three plugins (camera, geolocation, contacts)
- Calls three built-in app

![User Study Flow Diagram]
H1: Using CORDOVACONFIG changes developers’ mental model in regard to understanding the boundaries of the required privileges of an app.

H2: Using CORDOVACONFIG improves developers’ understanding of a specific app’s configurations.
Recruited 22 participants, from the university students who were familiar with hybrid frameworks.
User Study — Results: Apps Security Awareness

- Question: Common Coding Practice?

- Mainly Functional Requirements
- Implement & enforce security policy
- Implement Least Privilege
- Sanitize Data Input

- Question: Describe your security awareness level

- Novice: 86%
- Intermediate: 14%
User Study — Knowledge of the App Configuration

- H1: Using CORDOVACONFIG changes developers’ mental model in regard to understanding the boundaries of the required privileges of an app.

- Participants answered a set of questions before and after using the tool, which focused on measuring developers understanding of configuring the app to satisfy specific requirements.

- We find significant difference between the pre and post survey scores thus supporting the hypothesis that developers are more likely to understand the purpose and the meaning of the app configuration items after using the tool.
User Study — Changing Developer Mental Model

• H2: Using CORDOVACONFIG improves developers’ understanding of a specific app’s configurations.

• Participants answered a set of questions before and after using the tool, which focused on measuring the developer’s understanding of the permissions, configurations, and native call requirements.

• We find significant difference between the pre and post survey scores thus supporting the hypothesis that developers are more likely to understand the boundary of the app and to recognize the needed permission/configurations needed using the tool.
User Study — Perceived Benefits of CORDOVACONFIG

Question: What is the value of using this tool?

- "It helped understand the app flow."
- "It restricts access to URLs, installed apps, and plugins."
- "Because it automates the configuration according to app behavior, so that your app is secure."
- "It seems like it is a good tool to ensure that the app does what is supposed to do."
- "I appreciate all descriptions and explanations."
- "It enhances the readability of Cordova configurations."
- "It saved time and effort.
- "Avoiding misconfigurations."
- "Easiness."
The participants also answer the Software Usability Score (SUS) survey.

**Chart Description:**
- The chart shows the frequency of SUS scores.
- The mean SUS score is 86.25.
- The standard deviation is 8.046.
- The sample size is 22.

**Grades:**
- **Not Acceptable:** 0 - 60
- **Marginal:** 61 - 70
- **Low:** 71 - 80
- **High:** 81 - 90
- **Acceptable:** 91 - 100

**SUS Score Legend:**
- WORST IMAGINABLE: 0
- POOR: 10
- OK: 20
- GOOD: 30
- EXCELLENT: 40
- BEST IMAGINABLE: 50
Outline

- Introduction
- Problem Statement and Hypothesis
- The tool: CordovaConfig
- User study and Evaluation
- Concluding Remarks
Conclusion

- Hybrid Applications are expected to grow fast the market
- Secure Coding and Developer awareness are key to secure apps
- Providing secure tooling support is essential to facilitate the production of secure apps
- Configurations are the first line of defense for hybrid apps. Several attacks can be muted by simply using proper configurations
- Our tool help in enhancing the developer’s security understanding and security of the apps they create.
Thank you!

abeer@sfsu.edu

abeeraljarrah.info