APCO’s Key Attributes

• Some apps are not safe or effective
  • Misleading claims such as “Better than 9-1-1”
  • Interference with emergency operations

• APCO created a list of Key Attributes of Effective Apps for Public Safety and Emergency Response to guide the selection of apps on AppComm. Examples:
  • **Operability** (efficient use of data, minimal battery strain)
  • **Security**
  • **Communication with 9-1-1, sending data to PSAPs and Comm Centers, and interfacing with PSAPS**

• Sought input from orgs with technical expertise
App Workshops

• Partnering with orgs like FirstNet, NIST, and DHS
• Gathering public safety professionals, app developers, and industry experts
• Discussing app-related issues such as:
  • Security Requirements
  • Public Safety Data Classifications
  • Interoperability
App Security Testing

• Partnership with DHS OIC
• Explored whether app testing platforms can detect security vulnerabilities such as:
  • Excessive Permissions
  • Sensitive Data Exposure
• Adapted a program used by federal agencies for public safety and considered broader issues such as:
  • Data interoperability between agencies
  • Meeting security requirements
  • Compliance with federal, state, & local laws, regulations, & policies
Testing Workflow

User Registration via AppComm

APCO Assesses Functionality & Makes Determination

Automated Mobile App Testing by Kryptowire

Fail

APCO & SME (NIST, DHS, Kryptowire)

(Optional) User/Developer Inquiry

(Optional) Analyst Recommendations

Notify User with Results

Resubmits App

Accept & Notify

Reject & Notify

Pass

User Remediates
Mobile App Analysis

Android and iOS application analysis
- No source code needed
- Submitted to Kryptowire directly from AppComm
- Automated analysis
- Combination of Dynamic and Static Code Analysis
- Generation of developer feedback reports
- Confidential results
Security Evaluation Categories

Security
• The Kryptowire platform scans for various security issues that can be present in applications causing them to be vulnerable to exploitation. These types of issues can lead to sensitive data that is handled by the application to be compromised by malicious parties. Each of these issues should be considered carefully by the reviewing party and developers to ensure that applications are as secure as possible before they are distributed to users.

Privacy & Information Access
• Mobile applications can potentially have access to a wealth of sensitive information about the user and or device. These findings can range from applications integration with ad networks to proper handling of a user’s password. Each piece of information that is accessed by the application should be able to be justified by core functionality which is needed for operation.

Device Access
• The platform performs analyses to identify which sensitive functionality of the device that the application might access. The analysis will be able to identify the usage of these functionalities as well as the context to better understand the nature of the access. As with the privacy and information access, it should be evaluated whether the application requires these functionalities for its intended task.
Sample Report

**Required Developer Action**

Uses hard coded credentials for secure operations

Scans were performed on the application’s byte code and any packaged SDKs to search for hard-coded credentials used in cryptographic functions. These codes are declared as constant values within the application’s code.

The application contains a hard coded credential to perform secure operations such as encryption or web authentication. This allows anyone who has access to the application to retrieve the credentials and perform the same operations compromising security and privacy. Proper functionality provided by the native platform should be used to proper create and store credentials. More information can be found [here](#) and [here](#) (Android iOS)

**Required Developer Explanation**

Integrates with an ad network

This application integrates with one or more ad networks which can potentially expose sensitive user information to 3rd parties.

We identify through Dynamic Analysis if the application integrates with an ad network during execution. Please explain the ad networks your application integrates with and what data is given to them.

Developer explanation:
Sample Finding – sensitive data exposure

- User and device identifier sent to 3rd party
- Context provided to know what data is sent and where it is sent to

**Required Developer Action**

Exposes sensitive information

We scan all network traffic generated during Dynamic Analysis for any sensitive information. This includes both plain text HTTP and encrypted HTTPS.

The application exposes personally identifiable information (PII) in a communication to an external location. This causes a high risk to a user’s privacy. It should be evaluated if the data being exposed is being sent to authorized 3rd parties and if the app can operate as normal with the removal of the identified data exposure.

<table>
<thead>
<tr>
<th>Type</th>
<th>Data Value</th>
<th>Data Sent To</th>
<th>Possible Values</th>
</tr>
</thead>
<tbody>
<tr>
<td>User_Identifier</td>
<td>{&quot;channel&quot;:{&quot;background&quot;:false,&quot;tags&quot;:[]},&quot;opt_in&quot;:false,&quot;set_tags&quot;:true,&quot;device_type&quot;:&quot;ios&quot;},&quot;identity_hints&quot;:{&quot;user_id&quot;:&quot;&quot;,&quot;device_id&quot;:&quot;&quot;}</td>
<td>https://</td>
<td></td>
</tr>
</tbody>
</table>
Participants and Scan findings

- 20 app developers
- 10 dropped out during the pilot
- 33 apps tested total (15 Android & 18 iOS)
- Findings in two categories: red and orange
  - Red findings are more critical and should be corrected by developers unless sufficient evidence is provided to justify the behavior
  - Orange findings correspond to behavior that is often necessary for operation, but requires that the developer explain the need for the functionality. These findings can often identify unnecessary code that is included in the application which can be safely removed
Android apps

- All apps had at least one issue that was found
- 5 apps had red flags
  - 6 red flags total
  - Example findings:
    - Uses hard coded credentials for secure operations
    - Contains code to accept all SSL certificates
- 15 apps had orange flags
- 52 orange flags total
- Example findings:
  - Accesses device’s microphone
  - Accesses device’s camera
iOS apps

• 17 apps had at least one issue, 1 app had no issues
• 13 apps had red flags
  • 14 red flags total
  • Example findings:
    • iOS Application Transport Security feature disabled
    • Exposes sensitive user/device information
• 17 apps had orange flags
  • 57 orange flags total
  • Example findings:
    • Accesses the device’s contacts
    • Accesses the device’s microphone
Red & Orange Flags for 33 Apps

- Android Red Flags: 6
- Android Orange Flags: 52
- iOS Red Flags: 14
- iOS Orange Flags: 57
Specific Findings Overview

- **Android Findings**
  - Accepts all SSL certs – 3 apps
  - Hard coded credentials – 2 apps
  - Uses ad network – 2 apps
  - Can send SMS message – 1 app
  - Exposes sensitive information – 2 apps

- **iOS Findings**
  - Sensitive data exposure – 3 apps
  - Uses ad network – 2 apps
  - Can send SMS messages – 3 apps
  - Disables iOS Transport Layer Security – 8 apps
Addressing the Red & Orange Flags

- 20 red flag issues that were found:
  - 5 addressed/corrected by developers
  - 2 explained to be necessary for application’s operation
  - 1 flag determined to be a non-issue
  - 12 flags unaddressed

- 109 orange flag issues that were found:
  - 39 flags explained as necessary for the application’s operation
  - 4 flags determined to be non-issues
  - 2 flags addressed/corrected by developers
  - 64 flags unaddressed
Feedback from Developers

- Generally positive in terms of the value of app vetting, feedback from testing, and level of work required.
  - Universal agreement on value of app vetting for public safety
  - 0 – 8+ hours spent reviewing report feedback and implementing changes (mostly ~ 1 hour)
  - Satisfaction with perceived depth of testing varied
- Shared concern for protecting IP (not an issue for this type of testing)
- Process and communications
  - Desire for faster process, more detailed information about how testing works
  - Clarity needed on next steps
- Desire for testing/certification to be useful for marketing purposes
Pilot Testing Takeaways

• More context whenever possible
  • For all analysis findings where possible, more context for the developer is critical. This includes line numbers in code, the literal constant that was declared, or the specific server that was contacted.

• Static findings weren’t always relevant
  • Occasionally there was a finding which was found statically in code, but never used in the execution of the application. Typically we report this as a risk, but for this effort it became apparent that these findings were of little value. This is due mainly to the fact that each version of the application will be scanned and any previously unused code that is now invoked will be identified.

• Direct contact between Kryptowire and app developer was beneficial
  • In two cases, calls were setup between Kryptowire and the application developer for further discussion about a finding. In both of these cases, the issue was quickly resolved and the vetting process could continue.
Mobile Apps for 9-1-1
Mobile Apps & 9-1-1

• In 2015, APCO published a Fact Sheet and White Paper on Mobile Apps and 9-1-1
• Intended to educate the public and stakeholders in the app development community on the state of the 9-1-1 system and the role that mobile apps can play currently and in the future
• Shared early lessons learned from app efficiency and security testing pilots
APCO’s Vision for a 9-1-1 App

- Comply with industry standards
- Work anywhere in the country
- Be as reliable as the existing 9-1-1 system
- Preserve the familiar simplicity of 9-1-1
- Connect users to the appropriate PSAP
- Comply with FCC rules, state/local regs, and industry best practices
- Work without delay, regardless of device or app software updates
- Be free to the public and not impose unreasonable costs on PSAPs
- Be device and OS agnostic
- Meet public safety’s cybersecurity needs
- Efficiently use/monitor device battery and throttle-down if necessary
Comments to the FCC

• Dec. 2016, PSHSB sought comment on a request to initiate a proceeding to address concerns related to 9-1-1 apps

• APCO reiterated points from its white paper:
  • Mobile apps are not ready to replace a 9-1-1 call
  • Critical issues must be addressed, even before using apps to supplement a 9-1-1 call
    ➢ Ensuring reliability, universality, and security
    ➢ Ensuring sustainability, continued functionality of apps
    ➢ Ending misleading app descriptions that confuse the public and do a disservice to the efforts of committed public safety professionals
Upgrade required
Because you recently upgraded to iOS9, you must update this app.

Update App

Application crash
Your last session closed unexpectedly. Send a crash report?
No, thanks  Send report

Location Services
To continue you must enable location services
Go to Settings

Not Available In App Store
This app is not currently available in the store. Contact the app developer for more information.
OK
Claim: “The app can find you, but 9-1-1 can’t.”
Hypothetical 1
AppComm:  www.appcomm.org

DHS Report on “Securing Mobile Apps for First Responders”:  
https://www.dhs.gov/publication/csd-mobile-app-security-study-first-responders

APCO’s Fact Sheet and White Paper on Mobile Apps and 9-1-1:  

APCO’s Comments to the FCC on Apps for 9-1-1:  