Assurance Considerations For Software Supply Chains and 3rd Parties

Jeffrey Voas, PhD, FIEEE, FAAAS, FIET

j.voas@ieee.org

(Talk funding from IEEE New Fellows Selection Committee and IEEE Reliability Society. Thanks to IEEE Technology Management Council Hong Kong Chapter for organizing)
Three 3\textsuperscript{rd} Parties

- Software Supplier
- Software Assessor
- Assessors of Assessors, i.e., validating independent labs who sell, as a service, “product” vetting.
Premise for 3rd Party Software Assessments

Third party software should be tagged with some guarantee of quality/trustworthiness.

• Situation: Software Of Unknown Pedigree (SOUP)

• Goal: Software of Known Pedigree (SOKP)

Problems:

1. What is “good” software?
2. Or is the question “good enough”?
3. Or is the question “better”?
4. Who creates “a” vetting criteria of quality?
5. What is the criteria?
Huge Assumption

3rd Party Assessments Imply Independence
Why Is Software Hard?

• Software is non-Physical
• Physical sciences have measures
• Software does have measures, often direct, but with no correlation to “how good?, better?”
• Question: Are there physics for non-physical systems?
• A Timely, Repeatable Vetting Process that Ensures Fairness
Ah, But a Solution?

- We know how to build very reliable software in accordance to very reliable specifications
- Who does this? Regulated sectors.
- $1K/LOC?
- Will anyone “non-regulated” pay such?
Three Key Messages That 3rd Party Assessments Can Convey

• Compliance with Standards vs.

• Compliance with Requirements vs.

• Fitness for Purpose!
“A consumer [patient] may not be able to assess accurately whether a particular drug is safe, but [they] can be reasonably confident that drugs obtained from approved sources have the endorsement of the U.S. Food and Drug Administration (FDA) which confers important safety information. Computer system trustworthiness has nothing comparable to the FDA. The problem is both the absence of standard metrics and a generally accepted organization that could conduct such assessments. There is no Consumer Reports for Trustworthiness.”

Three Schools of “Thought”

3rd Party Assessments generally employ one or more of these perspectives.
And So For Supply Chain, Shouldn’t Product Assessments be a Function of What is Measureable?
Linking Assurance Assessment to Policy

System\{Reliability, Security, Performance, Availability, Safety, ..\}
Policy is $f(\text{System\{Reliability, Security, Performance, Availability, Safety, ..\}})$

OR

System\{Reliability, Security, Performance, Availability, Safety, ..\} is $f(\text{Policy})$
• Assurance is Multivariate
• Assurance not Well-Bounded or Well-Defined, however it can be Scoped
• Assurance incurs Financial trade-offs
• Assurance is a Risk-tolerance Activity
• Assurance is Temporal
Assurance is Temporal: Time Changes Everything
Fundamental Statement

Assurance is a function of:

1. The degree to which the *functional* system requirements are met, and

2. The degree to which the *inherited component behaviors* interoperate, i.e., cooperate or do not (interference).
Two Components

Two puzzle pieces labeled with Greek letters ξ and ψ.
Inherited Component Behaviors

Component $\xi$ has inherited behaviors:

$$(aR, bP, cF, dSa, eSe, fA, gT, hM)$$

Component $\psi$ has inherited behaviors:

$$(iR, jP, kF, lSa, mSe, nA, oT, pM)$$
Composing Pre-Inherited Properties

- $F(\xi \circ \psi)$ will inherit the Component’s Inherited Behaviors
- Similar to genetics
- A priori
Concerned originally about apps

Vetted 3rd party apps go into an app store.

If feedback is good, an app becomes app store accepted and money is deposited
Case Study

Take a COTS smart-phone off the shelf (Phone A), and re-engineer it through customizations (Phone B) such that a user:

1. Notices little difference between A and B,
2. Gains new security and data privacy, and
3. Retains expected performance
Risks in Mobile Security Supply Chain

Devices

Multi-Level Mobile Phone Security Architecture

Applications

Services

User Space

Android is a custom JVM (DALVK) running on modified Linux

Encryption Layer for I/O

Policy Manager (Profiling & Policing)

I/O Devices
(USB, WiFi, GPRS, Bluetooth, Internal Flash, SD card)

Enforce SELinux Policies on Data Flows for Android Apps
Prevent unauthorized Hardware Flows (USB, Network, FS)

Secure Verify Test Deploy

Hardware & App Providers

Enterprise Security

Dell

Motorola

DARPA

NIST

National Institute of Standards and Technology
Third Party Effects

- Threat Space
- Environment
- Software
- Physical System

3rd party
Worries

- Lost Phone - Forensics
- Password Length
- Counterfeit phone parts or apps
- Personnel Localization
- Camera Localization
- Identity Management
- Power Consumption
- Upgrade Policy
Approach

- Quarantine “any” functionality deemed “risky.”
- Secure/Vet the software apps
- Build an app vetting process and app-store
- Secure the hardware
- Rewrite encryption
- Sign our vetted apps
- Build a policy procedure for updates
- Secure operating system
- Rewrite app permissions when warranted
Our Result

• Thousands of modified smart-phones in use today.
• Reduces the need for soldiers to carry heavy radio packs, and offer most of the core features of smart-phones, e.g., cameras, apps, localization, velocity, etc. that radio packs don’t.
• Language translation is an app that has saved lives.
Component Architecture

- App (Source/APK)
- NIST-Signed APK
- Report Manager
- App Status Manager
- Configuration Manager
- Request Manager
- Marketplace Adapter
- UI Manager
- Repository Manager
- Tool Adapter Manager
- App PreProcessor
- APK and File Manager
- Configuration files
- APK, reports
- Metadata, status
- Metadata, status, reports
### Apps

<table>
<thead>
<tr>
<th>ID</th>
<th>App</th>
<th>Status</th>
<th>User</th>
<th>Date/Time</th>
</tr>
</thead>
<tbody>
<tr>
<td>5381399</td>
<td>Barcodescanner</td>
<td>REVIEW PENDING</td>
<td>cni</td>
<td>2013-09-25 12:21:39.0</td>
</tr>
<tr>
<td>3082096</td>
<td>ballistic-info-1.1.0-SNAPSHOT</td>
<td>WARNING</td>
<td>cni</td>
<td>2013-09-25 12:19:33.0</td>
</tr>
<tr>
<td>8509237</td>
<td>ballistic-game</td>
<td>REVIEW PENDING</td>
<td>cni</td>
<td>2013-09-25 12:16:16.0</td>
</tr>
<tr>
<td>2180603</td>
<td>aegis-iraq-android-1.1.0-SNAPSHOT</td>
<td>REVIEW PENDING</td>
<td>cni</td>
<td>2013-09-25 12:12:02.0</td>
</tr>
<tr>
<td>7477626</td>
<td>UsageStats-release</td>
<td>REVIEW PENDING</td>
<td>cni</td>
<td>2013-09-25 12:00:25.0</td>
</tr>
<tr>
<td>1201446</td>
<td>Feedback-release</td>
<td>REVIEW PENDING</td>
<td>cni</td>
<td>2013-09-25 11:59:10.0</td>
</tr>
<tr>
<td>6205314</td>
<td>Agora-release</td>
<td>REVIEW PENDING</td>
<td>cni</td>
<td>2013-09-25 11:41:20.0</td>
</tr>
<tr>
<td>9822478</td>
<td>BlearingToolPlugin-1.1.0-SNAPSHOT</td>
<td>REVIEW PENDING</td>
<td>along</td>
<td>2013-09-24 15:17:39.0</td>
</tr>
<tr>
<td>6816906</td>
<td>sneaky-weather-3</td>
<td>REVIEW PENDING</td>
<td>glodek</td>
<td>2013-09-24 14:33:42.0</td>
</tr>
<tr>
<td>6177645</td>
<td>BlearingToolPlugin-1.1.0-SNAPSHOT</td>
<td>REVIEW PENDING</td>
<td>along</td>
<td>2013-09-24 14:07:54.0</td>
</tr>
<tr>
<td>7015334</td>
<td>BlearingToolPlugin-1.1.0-SNAPSHOT</td>
<td>REVIEW PENDING</td>
<td>along</td>
<td>2013-09-24 13:57:59.0</td>
</tr>
<tr>
<td>3420664</td>
<td>ballistic-mb-1.1.0-SNAPSHOT</td>
<td>REVIEW PENDING</td>
<td>along</td>
<td>2013-09-24 13:44:34.0</td>
</tr>
<tr>
<td>700935</td>
<td>UlavMapsPlugin-0.0.1-SNAPSHOT</td>
<td>REVIEW PENDING</td>
<td>cni</td>
<td>2013-09-21 19:46:18.0</td>
</tr>
<tr>
<td>2091125</td>
<td>ammocore-1.7.8.6</td>
<td>REVIEW PENDING</td>
<td>cni</td>
<td>2013-09-20 15:20:37.0</td>
</tr>
<tr>
<td>9546901</td>
<td>AfghanPhrases_7</td>
<td>ENDORSED</td>
<td>bschulte</td>
<td>2013-09-20 13:23:28.0</td>
</tr>
<tr>
<td>6089244</td>
<td>AfghanPhrases_7</td>
<td>ENDORSED</td>
<td>bschulte</td>
<td>2013-08-20 11:41:03.0</td>
</tr>
</tbody>
</table>

Last updated: 2013-08-24 12:43:12Z
General Conclusion for NIST’s App Vetting Workflow Process

Result: A 3rd party software vetting approach such as the one discussed can be employed for COTS, GOTS, open source, and a variety of software types.

The key is what tools, how good are they, and how accurate is the human assessor?

Vendor testimonials and other vendor supplied artifacts can also be used as partial evidence to support the argument for inclusion in an app store.
Closing Thoughts

1. Component-based systems employ pre-qualified components
2. Pre-qualified carry with them the “baggage” of “inherited behaviors” and potentially “malicious” or “faulty” behaviors
3. Counterfeit components are problematic
4. Malware is inevitable
5. NIST Special Publication (SP) 800-163: Guidelines for Testing and Vetting Mobile Applications (next slide) is one attempt to 3rd party assess 3rd party apps via a defined process that was highly transparent
6. Gartner: Top 10 Strategic Technology Trends For 2013
‘Cheap Trust’: Possible? Idea

- Number of Third party apps of similar functionality $N(1)$
- Number of Third party assessors $N(2)$
- Number of App stores: ‘assessors of assessors’ $N(3)$
- Number of end users (beta testers): $N(4)$
- ‘Cheap Trust’ is $\sim$ SUM($N(1), N(2), N(3), N(4))$
- What is this? – a diversity argument
- Diversity did not work as well in software (non-physical) as it does in hardware manufacturing. But this is different, and very cheap – might it work?