Assessing software integrity of virtual appliances through software whitelists:

Is it any good?

Jun Ho Huh, Mirko Montanari, Derek Dagit, Rakesh Bobba, Dongwook Kim, Yoonjoo Choi, Roy Campbell

Sponsored by the AFRL / AFOSR
A virtual appliance (VA) is a virtual machine + some combination of software installed, designed to serve a specific purpose.
How a virtual appliance store works

1. Instantiate
2. Configure VA
3. Publish
4. Instantiate
5. Use

Publishers

Consumers

Base-Image-Instance

Instance-Virtual-Appliance-1

Cloud Provider

Trusted Base Image

Virtual Appliance 1

Cloud VA Store
Consumers’ concerns

• Is a VA really configured with all the software that it promises to have?

• Are all the software packages installed properly/cleanly?

• Will the VA behave as expected?
Our contributions

• An empirical study on the software integrity of real-world virtual appliances
  - Assessed through a software whitelist-based framework
  - Conducted on 151 Amazon VAs

• Our study shows significant variation in software integrity

• We demonstrate
  - The need for a whitelist-based framework to verify VAs
  - Feasibility and scalability of using whitelists for integrity assessment
Software whitelist-based framework

• Verify every file in a VA against a whitelist of known-good hash values.
  - More manageable in appliance store settings as many packages get installed on multiple VAs.
  - E.g., 18% of rpm-based VAs have perl 5.8.8

• Based on the presence of unverified files and missing files, we rate software integrity
  - 3 “fully verified” and integrity protected
  - 2 “partially clean” or medium integrity
  - 1 “modified” or low integrity
# unverified/missing files vs # modified packages
Classification based on the outliers

- Investigated the outliers by forming natural clusters based on % of packages given scores 1 and 2.
- \( k \)-means clustering was used to identify two clusters, \( k=2 \)
### Characteristics of VA clusters (average values)

<table>
<thead>
<tr>
<th>Group</th>
<th># VAs</th>
<th># All files</th>
<th># software</th>
<th># 3</th>
<th># 2</th>
<th># 1</th>
</tr>
</thead>
<tbody>
<tr>
<td>Cluster 1</td>
<td>137</td>
<td>52,033</td>
<td>421</td>
<td>418 (99%)</td>
<td>2 (0.5%)</td>
<td>1 (0.3%)</td>
</tr>
<tr>
<td>Cluster 2</td>
<td>14</td>
<td>52,946</td>
<td>453</td>
<td>205 (43%)</td>
<td>77 (17%)</td>
<td>171 (40%)</td>
</tr>
</tbody>
</table>

About 9% of the VAs have a significant portion of modified packages installed.
Characteristics of the low-integrity VA cluster

• Taking a closer look at 14 potentially untrusted VAs,
  - Significant portion of unverified files is common system files like `/bin/cut` and `/bin/grep`
  - 14 VAs are from different publishers, were built to provide different functions

• Virus scanners flagged 7 of the 14 VAs as malicious
  - What about the other 7?
  - In total, 41 of 111,981 unverified files were infected

None of them mentioned anything about software customization efforts!
Thank you

Questions?

Full paper to be presented in ASIACCS 2013

Jun Ho Huh
jhhuh@illinois.edu