Software Bill of Materials (SBOM) standardization

Robert A. Martin
Sr. Secure Software & Technology Prin. Eng.
Trust & Assurance Cyber Technologies Dept.
Cyber Solutions Technical Center

OMG Technical Meeting | Long Beach, CA
All types of Enterprises are Incorporating SW & SW-Enabled Things...

Medical

Buildings

Aeronautics

Manufacturing

Vehicles

Energy

Shipping
These Changes Go Well beyond Traditional Information Technology…

Water Treatment

Status & Health Monitoring

Oil & Gas

Hydro Power & Dam Mngt

Smart Munitions

Remote Management
Need Assurance of More Than Security — Need Assured Trustworthy Systems
Pervasiveness of connected SW & SW-enabled capabilities requires
supply chain security skills / new awareness of SW risks

<table>
<thead>
<tr>
<th>IT Risk</th>
<th>Operational Risk</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Loss of data or capability</strong></td>
<td><strong>Loss of property or lives</strong></td>
</tr>
<tr>
<td><strong>Loss of safety or reliability</strong></td>
<td></td>
</tr>
</tbody>
</table>

**Scratch Built Software**

- Majority of products built with no 3rd Party dependencies

**Assembled Software**

- Use of open source and 3rd party libraries, modules, frameworks, and services
- Multi-party software updating/patching

**Traditional Computers**

- Servers, databases
- Desktops, office apps
- Laptops, e-mail
- Tablets, browsers
- Switches, Routers

**Software Enabled Everything**

- Healthcare
- Aeronautics
- Smart Energy
- Oil & Gas
- Microgrids
- Implantable Medical
- Smart Manufacturing
- Water Treatment
- Hydro Power
- Smart Cities
- Smart Munitions
- Intelligent Vehicles
- Intelligent Shipping
- Dam Management
- Building Management
- Autonomous Systems
For Software-Enabled IIoT Version Control is Crucial

- SW & HW Part numbers/names
- SW & HW versions
- Libraries & Frameworks Used
- Tool Chain Used/Flags/Options
- Languages & versions used
The Supply Chain for Software-Enabled Capabilities is Opaque

Customer

Integrating Manufacturer/Supplier

Tier 2 Manufacturer/Supplier

Supplier

COTS

Software

Tier 3 Manufacturer/Supplier

US

Off-shore

Foreign

Foreign Developers

Tier 4 Manufacturer/Supplier

Contractor

Tier 4 Manufacturer/Supplier

US

Global

Foreign Location

© 2019 The MITRE Corporation. All rights reserved. Approved for Public Release; Distribution Unlimited. Case No: 19-01876-25
Market Transparency through “Software Bill of Materials”

- Third party components are a known systemic risk.
  - Transparency can drive tools and behavior to document risk, support mitigations, and drive better SW development practices.

- NTIA at Commerce launched an open, community-driven, cross-sector “multistakeholder process” to promote software component transparency.
  - Understand the problem and define basics of SBOM
  - Develop use cases across sectors on how such data can be used, today and in the future.
  - Guidance on how to use existing standards to implement SBOM
    - Software ID tags (SWID)
    - Software Package Data Exchange (SPDX)

- First phase deliverable mid-November 2019
- More info or to join: afriedman@ntia.doc.gov

© 2019 The MITRE Corporation. All rights reserved. Approved for Public Release; Distribution Unlimited. Case No: 19-01876-25
Roles and Benefits for SBOM Across the Supply Chain

Introduction
The Software Supply Chain
About this document: Goals and Methodology

Perspective: Produce Software
Reduce unplanned, unscheduled work
Reduce code bloat
Adequately understand dependencies within broader complex projects
Know and comply with the license obligations
Monitor components for vulnerabilities
End-of-life (EOL)
Make code easier to review
A blacklist of banned components
Provide an SBOM to a customer

Perspective: Choose Software
Identify potentially vulnerable components
A more targeted security analysis
Verify the sourcing
Compliance with policies
Aware of end-of-life components
Verify some claims
Understand the software’s integration
Pre-purchase and pre-installation planning
Market signal

Perspective: Operate Software
Organization can quickly evaluate whether it is using the component
Drive independent mitigations
Make more informed risk-based decisions
Alerts about potential end-of-life
Better support compliance and reporting requirements
Reduce costs through a more streamlined and efficient administration

Ecosystem, Network Effects, and Public Health Benefits of SBOM
Accelerated Vulnerability Management
Lowering Adoption Hurdles for SBOMs

End Users in Industry, Government, and Commerce

- Agriculture and Food
- Energy
- Transportation
- Chemical Industry
- Postal and Shipping

Sectors
- Water
- Public Health
- Telecommunications
- Banking and Finance
- Key Assets

Product & Service Suppliers

- Medical Devices
- Merchandise
- Automobiles
- Trains
- Vessels/Boats
- Building Mgmt Sys
- Software

INTEGRATED DEVELOPMENT ENVIRONMENTS (IDEs)

CLOUD TOOLS

SOURCE CODE & PACKAGE REPOSITORIES

FRAMEWORKS

SOFTWARE COMPOSITION ANALYSIS

BUILD CHOREOGRAPHY

Tool-to-Tool SBOM Exchange Standard effort

© 2019 The MITRE Corporation. All rights reserved. Approved for Public Release; Distribution Unlimited. Case No: 19-01870-25
Ecosystem of SW Development, Integration, and Management Tools
SW Development, Integration, and Management Tools

**Source Code & Package Repositories**

Amazon ECR, Assembla, Azure Container Registry, Beanstalk, Bitbucket, Codebase, Docker, GitHub, GitLab, Glitch, Google Container Registry, JFrog Artifactory, JFrog Xray, inedo, Kubernetes, Launchpad, Maven, Nexus (Sonatype), Phabricator, ProjectLockr, Repository Hosting, Savannah, SourceForge, SourceRepo, Subversion, and Unfuddle

**Build & Build Choreography Capabilities**

Ansible, Autotab, Bamboo, Bitrise, Buildkite, Buildroot, CircleCI, CMake, CruiseControl, Final builder, GCC, Gitlab CI, GoCD, Integrity, Jenkins, Strider CD, TeamCity, Terraform, Travis CI, Urbancode, and Vagrant

**Developer Desktops (Embedded, Web, Cloud, Desktops/Servers)**


Frameworks: .NET, Angular, Ansible, Apache Spark, ASP.NET, Bootstrap, Chef, Cordova, CryEngine, Django, Drupal, Express, Flask, Flutter, Hadoop, HTML5 Builder, Laravel, Node.js, Pandas, Puppet, React Native, React.js, Ruby on Rails, Spring, TensorFlow, Torch/PyTorch, Unity D, Unreal Engine, Visual Online, Vue.js, and Xamarin

Cloud Tools: Azure, AWS CodeBuild, Cloud Foundry, Google Cloud Build, Kwatere, Pivotal, and Red Hat

**Software Composition Analysis:**

Black Duck Software Composition Analysis (Synopsys), CAST Highlight (CAST Software), Finate State, FlexNet Code Insite (Flexera), Ion Channel, Insignary, SourceClear, Sonatype, Snyk, and WhiteSource
## Usage Scenarios for Tool-to-Tool SBoM

<table>
<thead>
<tr>
<th>Scenario</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Refer, Transfer or Purchase</strong></td>
<td>(definition of what it is)</td>
</tr>
<tr>
<td><strong>Pedigree</strong></td>
<td>(history of how it was produced)</td>
</tr>
<tr>
<td><strong>Provenance</strong></td>
<td>(chain of custody of it)</td>
</tr>
<tr>
<td><strong>Integrity</strong></td>
<td>(cryptographic basis of unalteredness)</td>
</tr>
<tr>
<td><strong>Proper and Legal</strong></td>
<td>(conditions about its use)</td>
</tr>
<tr>
<td><strong>Known Sw Vulns</strong></td>
<td>(known fixes are applied to it)</td>
</tr>
<tr>
<td><strong>Assurance</strong></td>
<td>(safe-secure-resilient)</td>
</tr>
<tr>
<td><strong>SBoM of a SW Service</strong></td>
<td>(SBoM of sw delivering service)</td>
</tr>
<tr>
<td><strong>Supply Chain Sequence Integrity</strong></td>
<td></td>
</tr>
</tbody>
</table>

© 2019 The MITRE Corporation. All rights reserved. Approved for Public Release; Distribution Unlimited. Case No: 19-01876-25
Provenance and Pedigree

DEFINITIONS

- **Provenance** *
  1. The origin, or source of something
  2. The history of ownership of a valued object, or work of art, or literature

- **Pedigree** *
  1. A register recording a line of ancestors
  2. An ancestral line: lineage
     The origin and the history of something; broadly: background, history

CONFUSION

- Many use “Provenance” for both meanings.
The provenance of a piece of data is both the custodianship as well as the lineage of processing and/or derivation that led to the piece of data.

*Definitions (from Merriam-Webster.com)
Provenance (Chain of Custody) of A6 includes Company C and Company B.

Pedigree (Lineage) of A6 includes the processes P1 and P2 and other artifacts used to create A6.

Provenance and Pedigree provide a basis on which to reason about the trustworthiness of an artifact or document.
The Path to Code Provenance at Uber

April 17, 2019

Uber

Code Provenance

Ensuring we have a **verifiable attestation** of the **origin of all code** running in production so that we can have a **root of trust** as we move forward to **defining** and **enforcing** a collection of **policies** throughout the different stages of the **software development process**.
Code Provenance

What do we get out of all this?
- "Chain of custody" for all code landing in production releases
- Enabling response in the event that anything goes away
- Flexible, enforced policies for what code is allowed to land in production releases

What are we protecting against?
1. Lazy / shortcutting insider
2. Malicious insider
3. Engineer laptop controlled by malicious outsider
4. Build / deploy infrastructure attacked by malicious outsider
Usage Scenarios and Tool-to-Tool SBOM candidate elements

**Usages**
1. Refer, Transfer or Purchase (definition of what it is)
2. Pedigree (history of how it was produced)
3. Provenance (chain of custody of it)
4. Integrity (cryptographic basis of unalteredness)
5. Intellectual Property Constraints
6. Known SW Vulns (known fixes are applied to it)
7. Assurance (secure-safe-resilient)
8. SBOM of a SW Service (SBOM of SW delivering service)
9. Supply Chain Sequence Integrity

**SBOM elements**
- Author of SBOM
- SBOM population method
- SBOM Time-Stamp
- Supplier
- Components (sources, executables, patches)
- Version
- Notes
- Licenses
- Created Using
- Created By
- Item Hash/Signature

**Correlated Info**
Usages

1. Refer, Transfer or Purchase
   (definition of what it is)

2. Pedigree
   (history of how it was produced)

3. Provenance
   (chain of custody of it)

4. Integrity
   (cryptographic basis of unalteredness)

5. Intellectual Property
   Constraints

6. Known SW Vulns
   (known fixes are applied to it)

7. Assurance
   (secure-safe-resilient)

8. SBoM of a SW Service
   (SBoM of SW delivering service)

9. Supply Chain Sequence
   Integrity

SBoM elements

- Author of SBoM
- SBoM population method
- SBoM Time-Stamp
- Supplier
- Components
   (sources, executables, patches)
- Version
- Notes
- Licenses
- Created Using
- Created By
- Item Hash/Signature

Correlated Info

None
Usages

1. Refer, Transfer or Purchase (definition of what it is)
2. Pedigree (history of how it was produced)
3. Provenance (chain of custody of it)
4. Integrity (cryptographic basis of unalteredness)
5. Intellectual Property Constraints
6. Known SW Vulns (known fixes are applied to it)
7. Assurance (secure-safe-resilient)
8. SBoM of a SW Service (SBoM of sw delivering service)
9. Supply Chain Sequence Integrity

SBoM elements

- Author of SBoM
- SBoM population method
- SBoM Time-Stamp
- Supplier
- Components (sources, executables, patches)
- Version
- Notes
- Licenses
- Created Using
- Created By
- Item Hash/Signature

Correlated Info

None
Usages

1. Refer, Transfer or Purchase (definition of what it is)
2. Pedigree (history of how it was produced)
3. Provenance (chain of custody of it)
4. Integrity (cryptographic basis of unalteredness)
5. Intellectual Property Constraints
6. Known SW Vulns (known fixes are applied to it)
7. Assurance (secure-safe-resilient)
8. SBoM of a SW Service (SBoM of SW delivering service)
9. Supply Chain Sequence Integrity

SBoM elements

- Author of SBoM
- SBoM population method
- SBoM Time-Stamp
- Supplier
- Components (sources, executables, patches)
- Version
- Notes
- Licenses
- Created Using
- Created By
- Item Hash/Signature

Correlated Info

None
The image is a diagram illustrating the elements of a Software Bill of Material (SBoM) and the usages of SBoM information. The diagram is divided into three sections:

**Usages**
- Refer, Transfer or Purchase (definition of what it is)
- Pedigree (history of how it was produced)
- Provenance (chain of custody of it)
- Integrity (cryptographic basis of unalteredness)
- Intellectual Property Constraints
- Known SW Vulns (known fixes are applied to it)
- Assurance (secure-safe-resilient)
- SBoM of a SW Service (SBoM of sw delivering service)
- Supply Chain Sequence Integrity

**SBoM elements**
- Author of SBoM
- SBoM population method
- SBoM Time-Stamp
- Supplier
- Components (sources, executables, patches)
- Version
- Notes
- Licenses
- Created Using
- Created By
- Item Hash/Signature

**Correlated Info**
None
Usages

1. Refer, Transfer or Purchase
   (definition of what it is)
2. Pedigree
   (history of how it was produced)
3. Provenance
   (chain of custody of it)
4. Integrity
   (cryptographic basis of unalteredness)
5. Intellectual Property Constraints
6. Known SW Vulns
   (known fixes are applied to it)
7. Assurance
   (secure-safe-resilient)
8. SBoM of a SW Service
   (SBoM of sw delivering service)
9. Supply Chain Sequence Integrity

SBoM elements

- Author of SBoM
- SBoM population method
- SBoM Time-Stam
- Supplier
- Components
  (sources, executables, patches)
- Version
- Notes
- Licenses
- Created Using
- Created By
- Item Hash/Signature

Correlated Info

None
Usages

1. Refer, Transfer or Purchase (definition of what it is)
2. Pedigree (history of how it was produced)
3. Provenance (chain of custody of it)
4. Integrity (cryptographic basis of unalteredness)
5. Intellectual Property Constraints
6. Known SW Vulns (known fixes are applied to it)
7. Assurance (secure-safe-resilient)
8. SBoM of a SW Service (SBoM of sw delivering service)
9. Supply Chain Sequence Integrity

SBoM elements

- Author of SBoM
- SBoM population method
- SBoM Time-Stampe
- Supplier
- Components (sources, executables, patches)
- Version
- Notes
- Licenses
- Created Using
- Created By
- Item Hash/Signature

Correlated Info

Vulnerability Knowledge Bases
Vulnerability Management Systems
Usages

1. Refer, Transfer or Purchase (definition of what it is)
2. Pedigree (history of how it was produced)
3. Provenance (chain of custody of it)
4. Integrity (cryptographic basis of unalteredness)
5. Intellectual Property Constraints
6. Known SW Vulns (known fixes are applied to it)
7. Assurance (secure-safe-resilient)
8. SBoM of a SW Service (SBoM of sw delivering service)
9. Supply Chain Sequence Integrity

SBoM elements

- Author of SBoM
- SBoM population method
- SBoM Time-Stam
- Supplier
- Components (sources, executables, patches)
- Version
- Notes
- Licenses
- Created Using
- Created By
- Item Hash/Signature

Correlated Info

Notes on exploitability of vulns
Vulnerability Knowledge Bases
Weakness Knowledge Bases
Assessment Results
Design Review
Code Review
Attack Surface Analysis
Static Analysis
Dynamic Analysis
Fuzz Testing
Pen Testing
Blue Teaming
Red Teaming
Organized as an Assurance Case

© 2019 The MITRE Corporation. All rights reserved. Approved for Public Release; Distribution Unlimited. Case No: 19-01876-25
Usages

1. Refer, Transfer or Purchase (definition of what it is)
2. Pedigree (history of how it was produced)
3. Provenance (chain of custody of it)
4. Integrity (cryptographic basis of unalteredness)
5. Intellectual Property Constraints
6. Known SW Vulns (known fixes are applied to it)
7. Assurance (secure-safe-resilient)
8. SBoM of a SW Service (SBoM of sw delivering service)
9. Supply Chain Sequence Integrity

SBoM elements

- Author of SBoM
- SBoM population method
- SBoM Time-Stamp
- Supplier
- Components (sources, executables, patches)
- Version
- Notes
- Licenses
- Created Using
- Created By
- Item Hash/Signature

Correlated Info

Logging SBOMs of Services Used
Usages

1. Refer, Transfer or Purchase (definition of what it is)
2. Pedigree (history of how it was produced)
3. Provenance (chain of custody of it)
4. Integrity (cryptographic basis of unalteredness)
5. Intellectual Property Constraints
6. Known SW Vuln (known fixes are applied to it)
7. Assurance (secure-safe-resilient)
8. SBoM of a SW Service (SBoM of sw delivering service)
9. Supply Chain Sequence Integrity

SBoM elements

- Author of SBoM
- SBoM population method
- SBoM Time-Stam
- Supplier
- Components (sources, executables, patches)
- Version
- Notes
- Licenses
- Created Using
- Created By
- Item Hash/Signature

Correlated Info

Desired sequence of ordered software supply chain steps, and requirements for each step for a specific project of interest
Launched 24 Sep 2019

© 2019 The MITRE Corporation. All rights reserved. Approved for Public Release; Distribution Unlimited. Case No: 19-01878-25
Whitepapers → CISQ → OMG RFC → ISO Std

• Socialize at Mar19 OMG meeting
• Draft SBoM as a Whitepaper in 3-day CISQ SBoM working session at Sep OMG meeting
• Prototype draft format in tool ecosystem, revise and draft RFC based on prototype results
• Co-submit draft RFC w/CISQ to OMG at Dec19 or Mar20 meeting
• Mar20/Jun20 OMG meeting – charter FTF
• Jun20/Sep20 OMG meeting - approve as OMG Standard
• Sep20/Dec20 Fast Track to ISO
Questions?