



主役登場

可視化データによる透明化時代
到来、そのときあなたは

鐘本 楊 Yo Kanemoto

NTT 社会情報研究所
主任研究員■ソフトウェアサプライチェーンリスクの
具体例・経験談

CI/CD (Continuous Integration/Continuous Delivery) 技術、コンテナ技術の進化により、ソフトウェアの開発・利用はとても便利になりました。容易に試せるゆえに悪意あるコードに気付かずに利用してしまう可能性が高くなったと感じます。例えば、私はPython言語を活用してソフトウェア開発をしますが、パッケージ管理システムを利用してソフトウェアをインストールする際に、打ち間違いによって、間違ったパッケージをインストールしそうなことがありました。途中で気付いて中断し、セキュリティ検査を行いました。本日のウイルスと違って、コンピュータウイルスの感染は一瞬ですので、「やってしまった」とヒヤッとしました。今の事例は人のミスを誘う攻撃で、誰にでも起こり得ることと思います。よく利用されるソフトウェアに悪いコードを混入させる手口もあります。攻撃者はアカウントを乗っ取ったり、開発者の振りをしていたりして悪意あるコードを混入してきます。

私はソフトウェアの説明だけ見て、便利そうだから利用するという行為はリスクを伴うと感じています。そのため、ソフトウェアの中身が分からないという不透明さを抜本的に解決したいという思いで研究を始めました。

■可視化データ活用の現状と課題

SBOM (Software Bill of Materials) を含め、可視化データを活用したセキュリティ確保が機能するには、まず可視化データの品質の確保が一番重要と考えています。つまり、データが誤ってはいはその後

く分析はすべて信頼できないものとなります。そこで、可視化データの品質に関する課題をいくつか紹介します。

1番目は、可視化データの構成要素が足りていないことです。現状、ソフトウェアの名称や識別子に関する記載は9割程度と比較的記載するケースが多いですが、SBOMの作成者を記載するケースは3割程度と少ない状況です。将来的に可視化データの提供品質・提供者によって、信頼するに値するかを判断する場面がやってくると思いますので、作成者を含め、これらの情報が今後明確に記載されることに期待しています。

2番目は、可視化データの構成要素の表現が不適切なことや表記揺れがあることです。例えば、ソフトウェアのベンダ名を記載すべき個所にpipやmavenのようなソフトウェアのエコシステム名を記載しているというケースがありました。表記揺れも機械的な処理が難しくなりますので、脆弱性検査などの検知漏れにつながってしまいます。

3番目は網羅性です。利用するすべてのソフトウェアが可視化データに網羅されることが一番ですが、現実には可視化できていない隠れたソフトウェアも存在します。私たちはこれらの隠れた依存関係を「暗黙的依存関係」と呼んでいます。暗黙的依存関係にあるソフトウェアは脆弱性検査等で検査されないおそれがあるので、パッケージ情報だけでなく、コードベース関係性を検知・把握する技術を利用することが必要と考えます。

■SBOM活用の未来

上記の3つの課題も含めて、私は、可視

化データを通じてソフトウェアの透明性を評価し、確かな透明性を持つソフトウェアが明らかになる仕組みが必要ではないかと考えております。そのため、ソフトウェアに関する「情報の豊富さ」と「情報の正確さ」を評価する観点で研究しています⁽¹⁾。

「情報の豊富さ」とはソフトウェアの目的・動作等を十分に説明しているかという観点です。説明が十分でない場合、利用者は正しく仕組みを理解せずに利用することになり、セキュリティリスクにさらされます。

「情報の正確さ」とは提供される可視化データの情報が正確なものであるかという観点です。コンポーネント情報の意図的な消去や依存関係の深さが十分可視化されているかを利用者側で把握することが大事です。

可視化データを提供するのは人間です。良いソフトウェア・悪いソフトウェアがあるように、良い可視化データ・悪い可視化データがあると思っています。スーパーで色・つやを見てから果物を買ったり、幼児用の食品は特に食品成分表を見てから購入したりすることと同じように、将来的には、利用者が可視化データの品質を見比べ、そのうえでソフトウェアやシステムを利用するか判断する世の中になっていくと考えます。そうした透明性ある世の中でも、問題がないようにソフトウェア開発者やサービス提供者に今から働きかけていきたいと思っています。

■参考文献

(1) <http://id.nii.ac.jp/1001/00228550/>