SecHack365

静的解析による仮想マシンの脆弱性検知ツール

Molysis

開発駆動コース仲山ゼミ
藤村 匡弘

Molysisとは？

仮想マシンイメージを**静的に解析して脆弱性を検知**するツールを作りました。検知できる脆弱性はOSにインストールされているライブラリが持つCVEの脆弱性を検知するツールです。早く簡単に実行できるをコンセプトに開発しています。

利用している仮想マシンイメージは本当に安全なのか

普段利用している仮想マシンイメージは本当に安全ですか？
ディストリビューションが配布している仮想マシンイメージも時間経過とともに脆弱性が増えます。

例えば、Vagrant Hubで最もダウンロードされているUbuntuのイメージは公開されてから**1年も経過**しています。

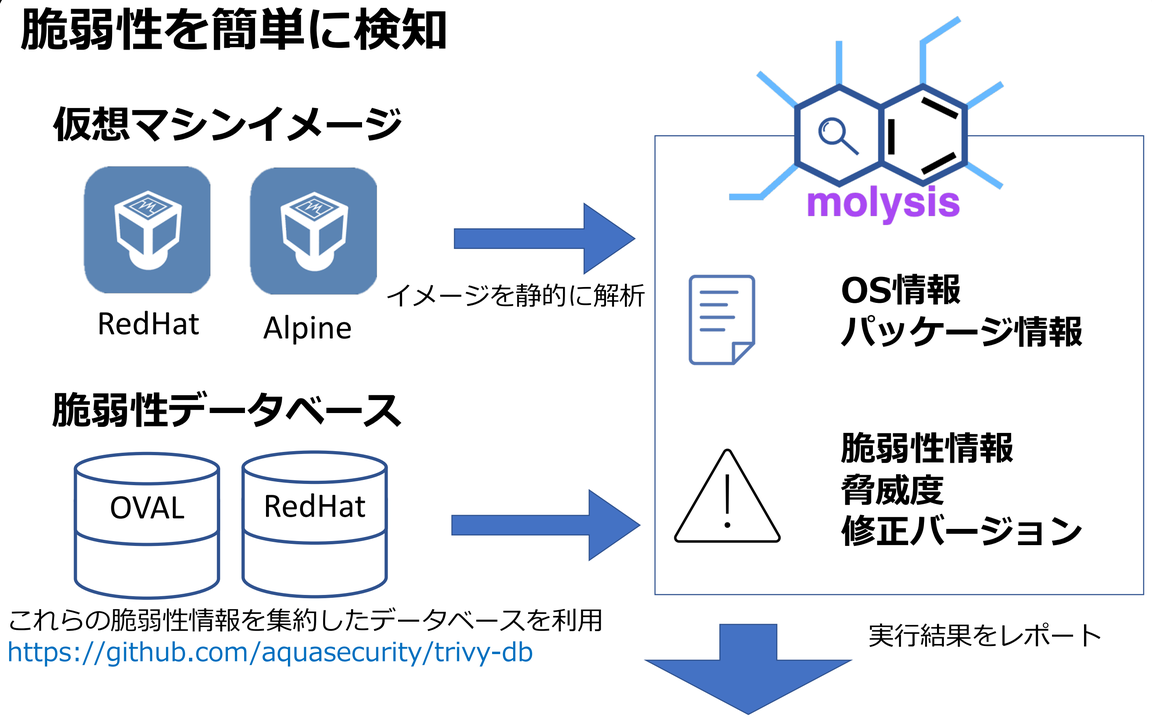
Molysisを利用すれば**仮想マシンを構築することなくリスクを検出**

コンテナセキュリティの世界では

コンテナの商用利用ではイメージを**デプロイする前に静的に脆弱性検査**するのが当たり前になっています。

trivy clair

脆弱性を簡単に検知



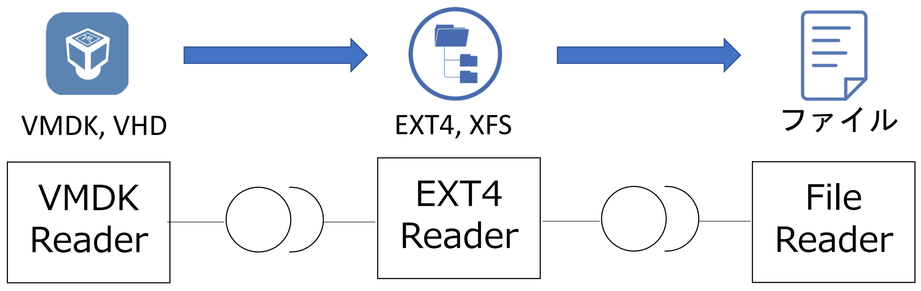

```
alpine: 3.8.2
=============
Total: 55 (UNKNOWN: 0, LOW: 0, MEDIUM: 26, HIGH: 18, CRITICAL: 11)

+---------------------+------------------+----------+-------------------+---------------+---------------------------------+
|       LIBRARY       | VULNERABILITY ID | SEVERITY | INSTALLED VERSION | FIXED VERSION |              TITLE              |
+---------------------+------------------+----------+-------------------+---------------+---------------------------------+
| musl                | CVE-2019-14697   | CRITICAL | 1.1.19-r10        | 1.1.19-r11    | musl libc through 1.1.23        |
|                     |                  |          |                   |               | has an x87 floating-point       |
|                     |                  |          |                   |               | stack adjustment imbalance,     |
|                     |                  |          |                   |               | related...                      |
+---------------------+------------------+----------+-------------------+---------------+---------------------------------+
```

Molysisの特徴

低リソースで動作

Molysisは仮想マシンイメージの解析において、**ファイルを展開することなく、透過的にストリームで処理する**ため、環境のスペックやリソースに依存することなく実行することが可能。



全て透過的に解析

各処理はio.Readerと呼ばれるストリーム処理の共通のインターフェースで実装されている。
Molysisでは、仮想マシンイメージや、ファイルシステムの読み取り処理をストリームで完結するためにパーサーを実装している。

https://github.com/masahiro331/go-ext4-filesystem
https://github.com/masahiro331/go-vmdk-parser

メモリ使用量について

Molysisの実行時に利用したメモリ量を計測した結果、
1MB程度のメモリ使用量で動作していることがわかる。

```
(pprof) top5
Showing nodes accounting for 1121.21kB, 96.78% of 1158.53kB total
Dropped 15 nodes (cum <= 5.79kB)
Showing top 5 nodes out of 32
      flat  flat%   sum%        cum   cum%
  867.80kB 74.91% 74.91%   899.93kB 77.68%  github.com/masahiro331/go-ext4-filesystem/pkg.(*Ext4Reader).Next
  196.16kB 16.93% 91.84%   196.16kB 16.93%  bufio.NewReaderSize
   28.12kB  2.43% 94.26%    28.12kB  2.43%  github.com/lunixbochs/struc.(*Field).Unpack
   20.13kB  1.74% 96.00%    20.13kB  1.74%  github.com/molysis/molysis/analyzer/pkg/apk.ApkPkgAnalyzer.uniquePkgs
    8.99kB  0.78% 96.78%    17.39kB  1.50%  github.com/aquasecurity/trivy-db/pkg/db.Config.GetAdvisories
(pprof)
```

マルチプラットフォーム対応

Molysisは実行環境を問わず利用することが可能
公開されているOSSの脆弱性検知ツールの多くはWindowsでの動作をサポートしておらず、Mac OSやLinux Distributionのみサポートしているが、MolysisはWindowsでも実行できる。

mac OS Windows CentOS ubuntu Amazon Linux

動作環境を選ばない

Molysisの開発を通じて

開発する中で利用しているOSSに必要な機能が足りなかったためPRを出すことで機能を取り入れてもらった。
経緯や改善の詳細については記事としてInternet上に公開している。

https://github.com/knqyf263/go-rpmdb
https://masahiro331.hatenablog.com/entry/2020/12/20/052234

今後の展望

VMDKやEXT4だけではなく、VHDやXFS、LVMといったファイル形式にも対応し、解析できる範囲を拡大する。

開発したツールはGitHubにて公開しています。
https://github.com/molysis/molysis

molysis

NICT 国立研究開発法人 情報通信研究機構