Vuls 祭り #10 (2024/08/20)

VexLLM: LLMを用いたVEX自動生成ツール

https://github.com/AkihiroSuda/vexllm

須田 瑛大 (NTT ソフトウェアイノベーションセンタ)
akihiro.suda.cz@hco.ntt.co.jp

背景: 脆弱性が検出されすぎる

- Trivy などを使うと大量に脆弱性が検出されるが、全て対処しないといけないわけではない
  - イメージに含まれる全てのライブラリの全ての関数を使うわけではない
  - 全てのコマンドラインツールの全ての引数を使うわけではない
  - 実際に攻撃可能なものは3%程度 (先ほどの福田さんの発表参照)

背景: 脆弱性が検出されすぎる

- 例: python:3.12.4 イメージには CVE-2024-32002 (9.0 CRITICAL) を持つ git バイナリが含まれる
  - git の submodules と シンボリックリンクに関する脆弱性
  - git を実行しないことがわかっているなら気にする必要はない
  - git を実行するとしても実行対象リポジトリが定数になっているなら実際に攻撃を受ける可能性は必ずしも高くはない
  - CVSSv3 スコアが 9.0 だからといって焦る必要はない

VEXや.trivyignoreで脆弱性情報を抑制できる

- 気にしなくてよい脆弱性の番号をVEX や .trivyignore に書いておけば、Trivy の警告を抑制できる
  - .trivyignore: 無視する脆弱性番号しか含まない簡易VEX

- しかし、VEX や .trivyignore を自分で書くのは大変

VexLLM

- https://github.com/AkihiroSuda/vexllm
- OpenAI などの LLM に OpenVEX や .trivyignore を出力させる
- 用途に応じてヒントを指定する

```
vexllm generate python.json .trivyignore \
  --hint-not-server \
  --hint-compromise-on-availability \
  --hint-used-commands=python3 \
  --hint-unused-commands=git,wget,curl,apt,apt-get
```

VexLLM

- https://github.com/AkihiroSuda/vexllm
- OpenAI などの LLM に OpenVEX や .trivyignore を出力させる
- 用途に応じてヒントを指定する

入力（TrivyのJSON）　出力

```
vexllm generate python.json .trivyignore \
  --hint-not-server \
  --hint-compromise-on-availability \
  --hint-used-commands=python3 \
  --hint-unused-commands=git,wget,curl,apt,apt-get
```

サーバ用途ではない

情報漏洩や改竄は妥協できないが可用性については運用対処の余地がありうる

絶対使われるコマンド

絶対使われないコマンド

.trivyignore 出力例

コメント行はOpenVEXのJSON

```
# {"vulnerability":{"@id":"CVE-2024-32002","description":"Git is a revision cont
rol system. Prior to versions 2.45.1, 2.44.1, 2.43.4, 2.42.2, 2.41.1, 2.40.2, an
d 2.39.4, repositories with submodules can be crafted in a way that exploits a b
ug in Git whereby it can be fooled into writing files not into the submodule's w
orktree but into a `.git/` directory. This allows writing a hook that will be ex
ecuted while the clone operation is still running, giving the user no opportunit
y to inspect the code that is being executed. The problem has been patched in ve
rsions 2.45.1, 2.44.1, 2.43.4, 2.42.2, 2.41.1, 2.40.2, and 2.39.4. If symbolic l
ink support is disabled in Git (e.g. via `git config --global core.symlinks fals
e`), the described attack won't work. As always, it is best to avoid cloning rep
ositories from untrusted sources."},"products":[{"@id":"git-man@1:2.39.2-1.1"}],
"status":"not_affected","justification":"vulnerable_code_not_in_execute_path","i
mpact_statement":"{\"confidence\":0.6,\"reason\":\"This RCE vulnerability is spe
cific to recursive clones in Git, which is not a commonly used feature in the co
ntext of a Python container image.\"}"}
CVE-2024-32002
```

黄色い部分がLLMの出力

あくまでも参考程度

- 出力は毎回異なることが多い
- false positive も false negative も多い
- なるべく人間がレビューして手直しすべき

内部で生成しているプロンプト [1/2]

```
You are a security expert talented for triaging vulnerability reports.
You judge whether a vulnerability is likely negligible under the specified hints.

### Hints
* Artifact type: "container_image"
* Artifact name: "python:3.12.4"
* The artifact is a container image. So, kernel-related vulnerabilities are safely negligible.
* The artifact is not used as a network server program. So, server-specific vulnerabilities are safely
negligible.
* The following shell commands are known to be used: [python3]
* The following shell commands are known to be unused and their vulnerabilities are negligible,
although these commands might be still present in the artifact: [git wget curl apt apt-get]
* Put solid focus on Confidentiality and Integrity rather than Availability. e.g., denial-of-service
does not need to be considered as catastrophic as data leakage and modification.

### Input format:
The input is similar to [Trivy](https://github.com/aquasecurity/trivy)'s JSON, but not exactly same.
```

内部で生成しているプロンプト [2/2]

NTT

````
### Output format
If you find negligible vulnerabilities, print a JSON map formatted and indented as follows:
```json

{
        "CVE-2042-12345": {"confidence": 0.4, "reason": "This DDOS vulnerability is only exploitable in
public server programs."},
        "CVE-2042-23456": {"confidence": 0.8, "reason": "The vulnerable package \"foo\" is unlikely
used."}
}

```
* `confidence` (0.0-1.0): higher value if you are confident with the answer.
* `reason`: the reason why you think the vulnerability is negligible. Should be unique, descriptive,
and in 2 or 3 sentences.
Do not include non-negligible vulnerabilities in the result.
Only print a valid JSON.
````

今後

- false positive や false negative の定量評価
- Trivy のプラグインとして実行できるようにする
- Trivy の CLI ともっと密に連携させる
  - いちいちJSONや.trivyignoreを作る操作を要らなくする
- Trivy 以外、OpenVEX以外にも対応する
- 各種OSSコミュニティで採用してもらうことを目指す
  - LLMに与えるヒント(絶対使うコマンド、使わないコマンドなど)のファイルを作成・メンテナンスしてもらうようにしたい (VEX Hub上で?)