Artefato CONF'YEAR Apêndice: Artigo #7880

Douglas Rodrigues Fideles[1], Douglas Paim Lautert[1],

Diego Luis Kreutz[1], Silvio Ereno Quincozes[1]

[1]Universidade Federal do Pampa (UNIPAMPA)

{douglasfideles,douglaslautert}aluno.@unipampa.edu.br

{diegokreutz,silvioquincozes}@unipampa.edu.br

Resumo. *A construção e manutenção de datasets atualizados de vulnerabilidades enfrentam desafios como falta de padronização e necessidade de automação. Neste trabalho apresentamos a VulnSyncAI, uma ferramenta modular que utiliza PLN e LLMs para correlacionar informações de múltiplas fontes, garantindo datasets atualizados e relevantes. A VulnSyncAI melhora a eficácia de modelos de IA na detecção de ameaças, automatizando processos e aumentando a eficiência na criação de datasets representativos.*

O apêndice permite que os autores forneçam mais detalhes para os revisores sobre o artefatos. Utilize este documento para passar informações de acordo com os selos requisitados. Caso seja necessário acesso a recursos externos, o respectivo processo e credenciais devem estar descritos neste documento.

1. Selos Considerados

Este artefato está submetido para avaliação dos seguintes selos:

- Disponíveis: O código-fonte e os datasets estão disponíveis publicamente.
- Funcionais: A ferramenta é funcional e pode ser executada seguindo as instruções fornecidas (no README[2] principal).
- Sustentáveis: O código é modular, bem documentado (no README[2] principal) e projetado para ser extensível.
- Experimentos Reprodutíveis: O README[2] principal fornecem instruções detalhadas para reproduzir os experimentos descritos no artigo.

2. Informações básicas

Para utilizar a VulnSyncAI, para LLMs, é necessário, no mínimo, um computador com processador quad-core, 8GB de RAM e acesso à Internet. O sistema operacional pode ser Windows ou Linux. É essencial ter Python 3.10 ou superior instalado, além das bibliotecas conforme a documentação da ferramenta. Para as SLMs, servidores que suportem grandes cargas de dados na memória são indispensáveis, sendo fundamental o uso de hardware robusto para o funcionamento eficiente dessas soluções locais.

Para o trabalho foi utilizado a configuração SLM Llama3 e Gemini e Deepseek para API:

- LLMs do Gemini e DeepSeek: Computador local com Intel Core i7-7700HQ, 16GB RAM, Windows 10.

- SLM do Llama3 (DeepHermes-3-Llama-3-8B-Preview3): Máquina virtual na Google Cloud com 12 vCPUs, 40GB RAM, Ubuntu 20.04.

O código-fonte completo do VulnBuilderAI, juntamente com instruções detalhadas de instalação, configuração e uso, está disponível no seguinte repositório GitHub [1].

2.1. Dependências

A linguagem de programação utilizada foi Python na versão 3.10.8, juntamente com as bibliotecas requests (versão 2.30.0), google-generativeai (versão 0.8.3) e openai (versão 1.59.3).

Para executar o VulnSyncAI com todos os seus recursos, são necessárias chaves de API para as seguintes fontes de dados e modelos de linguagem (LLMs). As credenciais são:

- VULNERS_API_KEY: L884A3520PWGV71JAVXPLQ08AMUUTUVFLABVJQWD1QT5L5SU3481ZEXSHGJECOE5
- GEMINI_API_KEY: AIzaSyD5TM1e-ka1L3Cqv-YKRt0Iqo77kiFMC5Y
- DEEPSEEK_API_KEY: a1cd1757-5605-4e57-be56-ac55d8699ed9
- HUGGINGFACE_API_KEY: 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

Essas configurações devem ser adicionadas em: config.yaml, como descrito no README[2]. Onde fornecemos instruções detalhadas para reproduzir os experimentos descritos no artigo.

[1] https://github.com/datasets-community/VulnSyncAI
[2] https://github.com/datasets-community/VulnSyncAI