Classificação de artefatos de vulnerabilidades de software usando dados públicos da Internet

Estevao Rabello Ussler, Daniel Sadoc Menasche

[1]Instituto de Computação – Universidade Federal do Rio de Janeiro (UFRJ)
Caixa Postal 68.530 – 21941-590 – Rio de Janeiro – RJ – Brazil

{estevaoru, sadoc}@dcc.ufrj.br

Abstract. *The increasing complexity and interconnection of software systems has led to a significant increase in security vulnerabilities, posing substantial challenges for the protection of sensitive systems and data. This work proposes a comprehensive approach to classify, filter and annotate artifacts related to software vulnerabilities. Artifacts cover not only patches and exploits, but also concept test code (PoCs), providing a comprehensive view of existing threats. Using several public databases, such as NomiSec, InTheWild and NVD (National Vulnerability Database), this study seeks to provide a more organized and in-depth view of software vulnerabilities. In addition to contributing to the understanding of these vulnerabilities, the research aims to facilitate the implementation of effective security and mitigation measures. Implementing this approach promises to positively impact the information security community by providing valuable insights for the protection of sensitive systems and data. Integrating techniques for classifying and filtering software vulnerability artifacts is critical to ensuring the effectiveness of cyber defense strategies in an increasingly challenging environment.*

Resumo. *A crescente complexidade e interconexão de sistemas de software têm levado a um aumento significativo nas vulnerabilidades de segurança, representando desafios substanciais para a proteção de sistemas e dados sensíveis. Este trabalho propõe uma abordagem abrangente para classificar, filtrar e anotar artefatos relacionados a vulnerabilidades de software. Os artefatos abrangem não apenas patches e exploits, mas também códigos de testes de conceito (PoCs), fornecendo uma visão abrangente das ameaças existentes. Utilizando diversas bases de dados públicas, como o NomiSec, InTheWild e NVD (National Vulnerability Database), este estudo busca fornecer uma visão mais organizada e aprofundada das vulnerabilidades de software. Além de contribuir para a compreensão dessas vulnerabilidades, a pesquisa visa facilitar a implementação de medidas eficazes de segurança e mitigação. A implementação dessa abordagem promete impactar positivamente a comunidade de segurança da informação, fornecendo insights valiosos para a proteção de sistemas e dados sensíveis. A integração de técnicas de classificação e filtragem de artefatos de vulnerabilidades de software é fundamental para garantir a eficácia das estratégias de defesa cibernética em um ambiente cada vez mais desafiador.*

Identificação:

- Período de vigência: 3/11/2022 até a presente data

- Tipo da bolsa: PIBIC
- Tipo do relatório: Parcial
- Título do projeto: Classificação de artefatos de vulnerabilidades de software
- Nome do bolsista: Estevão Rabello Ussler
- Seu curso: Ciência da computação
- Nome do orientador e seu departamento: Daniel Sadoc Menasché, Depart. de Computação, Inst. de Computação
- Instituto e centro: Inst de Computação/CCMN
- Período de recebimento da bolsa, pelo bolsista: de novembro de 2022 até presente data

1. Introdução

A crescente complexidade e interconexão de sistemas de software têm levado a um aumento significativo nas vulnerabilidades de segurança, representando desafios substanciais para a proteção de sistemas e dados sensíveis [Ponce et al. 2022, Miranda et al. 2021, Figueiredo et al. 2023, Yadmani et al. 2022, Rokon et al. 2020, Miranda et al. 2023]. Este trabalho propõe uma abordagem abrangente para classificar, filtrar e anotar artefatos relacionados a vulnerabilidades de software. Os artefatos abrangem não apenas patches e exploits, mas também códigos de testes de conceito (PoCs), fornecendo uma visão abrangente das ameaças existentes.

A fonte de artefatos escolhida para este trabalho, o NomiSec, consiste num repositório cujo propósito central é a coleta automatizada de Provas de Conceito (PoCs) presentes no Github. Essa coleta é realizada por meio da busca em todos os repositórios do Github que mencionam Common Vulnerabilities and Exposures (CVEs). Os dados coletados são então organizados em arquivos JSON, onde, no caso de múltiplos repositórios referenciando uma mesma CVE, todos os encontrados são armazenados no mesmo arquivo JSON. Esses arquivos não apenas contêm os links para os repositórios relevantes, mas também descrições detalhadas, datas de criação, atualização e inclusão no NomiSec, além de informações sobre os usuários responsáveis pela postagem. Esses arquivos também englobam dados relacionados à visibilidade dos repositórios, discussões associadas e outras métricas pertinentes. Essa abordagem sistemática e abrangente do NomiSec oferece uma valiosa fonte de informações para a comunidade de segurança da informação, possibilitando análises mais profundas e informadas sobre vulnerabilidades de software e suas potenciais implicações.

Neste contexto, nossas principais contribuições são:

- avaliação da abrangência do NomiSec: identificamos que o NomiSec abrange uma parcela signifiativa dos dados sobre vulnerabilidades presentes no Github, mas que a abrangência não é total, e envolve atrasos de publicação
- identificação da qualidade dos dados: percebemos que alguns dados divulgados no NomiSec referem-se a artefatos que não são uma ameaça, como simples verificadores (*checkers*)
- representação e visualização dos dados: apontamos direções iniciais para representar e visualizar os dados presentes em repositórios de dados sobre vulnerabilidades.

O restante deste artigo está organizado da seguinte forma. A Seção 2 discute a relação entre fontes, indicando que algumas são complementares e outras substitutíveis.

Em seguida, a Seção 3 reporta observações sobre o fato de que alguns dos artefatos apresentados como provas-de-conceito (PoC) são na realidade apenas verificadores de vulnerabilidades (*checkers*). Motivados por esta descoberta, consideramos análises adicionais dos artefatos na Seção 4 e a Seção 5 conclui.

2. Relação entre fontes: complementares ou substituíveis?

2.1. NomiSec versus InTheWild

Durante um estágio posterior de nossa pesquisa, outros colegas, que haviam investigado mais a fundo o repositório do InTheWild, compartilharam conosco a observação de que também havia uma quantidade significativa de links do Github neste repositório. Essa informação despertou nossa curiosidade, levando-nos a iniciar uma investigação mais detalhada e subsequente análise.

A fonte Inthewild.io é uma plataforma notável que se destaca como uma fonte confiável para artefatos relacionados a vulnerabilidades de software. A proposta central do Inthewild.io é catalogar e disponibilizar informações sobre incidentes de segurança reais que ocorreram ”em campo”, ou seja, em ambientes de produção. Isso é alcançado por meio da coleta de dados de várias fontes, incluindo feeds de notícias, fóruns de segurança e relatórios de empresas de segurança cibernética. Esses dados são meticulosamente organizados e apresentados em uma interface acessível, permitindo que os pesquisadores e profissionais de segurança da informação obtenham insights valiosos sobre as ameaças atuais e emergentes.

Com base nessa análise preliminar dos dados do InTheWild, constatou-se que o mesmo abriga 10 fontes de PoCs, sendo o Github a única fonte presente no NomiSec. Foram identificados 5489 links provenientes do Github presentes no InTheWild, em comparação com os 11627 links do Github encontrados no NomiSec. Essa discrepância suscitou uma investigação mais aprofundada para determinar a interseção desses conjuntos de links entre o InTheWild e o NomiSec.

Observou-se que todos os 5489 links do Github presentes no InTheWild também estavam no NomiSec. Entretanto, chamou atenção o fato de o NomiSec possuir 6138 links adicionais que não estavam presentes no InTheWild. Essa constatação levou à conclusão de que os dois repositórios são complementares: o InTheWild é mais abrangente devido à inclusão de outras fontes, enquanto o NomiSec oferece um conjunto exclusivo de links.

É importante ressaltar que nem todos esses links, que estão presentes exclusivamente no NomiSec, são necessariamente PoCs. Portanto, uma investigação mais aprofundada se faz necessária para avaliar sua relevância.

2.2. Abrangência do NomiSec com relação à totalidade do Github: relevância e atraso

Durante nossa pesquisa, ao examinar os arquivos JSON associados às CVEs (Vulnerabilidades e Exposições Comuns) no GitHub, observamos uma peculiaridade: alguns desses arquivos continham mais de um link para a mesma CVE. Essa descoberta nos levou a uma investigação mais aprofundada sobre a presença de múltiplos links para uma única CVE.

Descobrimos que um total de 1.227 arquivos continham mais de um link, variando entre 2 e 396 links por arquivo. Impulsionados por essa constatação, decidimos conduzir

uma busca manual no GitHub para verificar as CVEs com um número significativo de links, com o propósito de garantir a integridade e a abrangência das informações contidas nos repositórios.

No entanto, durante essa análise, nos deparamos com uma discrepância que para nós foi surpreendente: muitos dos links encontrados no GitHub não estavam presentes no repositório NomiSec. Esta discrepância ressalta a importância de uma investigação minuciosa para avaliar a consistência e a completude dos dados entre diferentes fontes, mesmo quando uma das fontes (NomiSec) é subconjunto da outra (Github).

Para fornecer uma visão mais clara dessa discrepância, apresentamos no gráfico abaixo as CVEs com maior número de links. As barras claras indicam, o número de links para repositórios encontrados pelo NomiSec, enquanto as barras escuras indicam o número de links de repositórios presentes no GitHub.

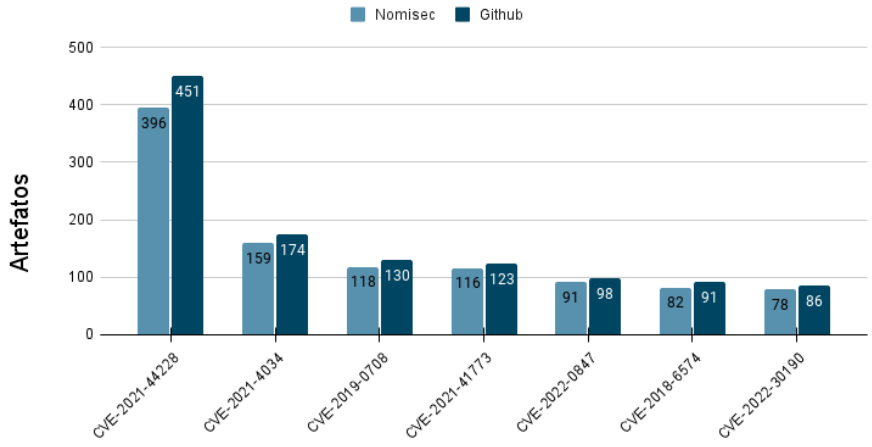


Figura 1. Abrangência do NomiSec: NomiSec abrange parcelas uma fração significativa dos artefatos do Github, mas não todos.

Com base nos dados apresentados anteriormente, é possível concluir que o NomiSec não abrange completamente o GitHub, pois há muitos links associados às CVEs presentes no GitHub que não estão refletidos no NomiSec.

Essa constatação levanta questionamentos sobre as razões pelas quais o NomiSec não atinge uma cobertura completa do GitHub, considerando que essa seria, em teoria, sua premissa fundamental. Para explorar essa questão, formulamos algumas hipóteses.

- A primeira hipótese sugere a existência de um filtro para determinar quais links serão incluídos ou não no repositório do NomiSec. Isso sugere a possibilidade de que apenas uma parte dos links seja considerada relevante o suficiente para ser incluída no repositório.
- A segunda hipótese considera a possibilidade de que os links discrepantes sejam adicionados ao NomiSec em intervalos de tempo específicos e não sejam atualizados em tempo real. Isso implica que pode haver uma defasagem na atualização dos dados entre o GitHub e o NomiSec.
- Por fim, a terceira hipótese levanta a possibilidade de que o NomiSec utilize a API do GitHub para obter os links e informações dos repositórios relacionados

às CVEs, e que as limitações dessa API possam dificultar a obtenção de todos os links relacionados às CVEs. Isso sugere que o NomiSec pode enfrentar restrições técnicas que afetam sua capacidade de obter informações de forma abrangente do GitHub.

Essas hipóteses fornecem um ponto de partida para investigações adicionais sobre as limitações e desafios enfrentados pelo NomiSec na coleta e atualização de dados do GitHub.

3. Identificando verificadores (*checkers*)

O processo de validação e confiabilidade dos dados no repositório NomiSec revelou uma questão crítica relacionada à presença de checkers. O NomiSec, em sua premissa, tem como objetivo exclusivo coletar automaticamente apenas Provas de Conceito (PoCs), representando uma fonte confiável para a comunidade de segurança da informação. No entanto, uma parte substancial de nosso trabalho envolveu uma investigação minuciosa para avaliar se essa premissa era efetivamente cumprida.

Durante nossa pesquisa, conduzimos uma análise manual detalhada, empregando termos-chave como "checker" e outras palavras relacionadas. Essa busca revelou a presença de 55 artefatos suspeitos nos dados coletados pelo NomiSec. Cada artefato identificado como potencialmente relacionado a checkers foi submetido a uma análise cuidadosa, na qual examinamos os repositórios e os códigos associados para determinar sua natureza e função.

Os resultados dessa análise foram documentados em um relatório detalhado, publicado e disponibilizado publicamente no GitHub (https://github.com/leoambrus/Tagging-NomiSec) onde 50 artefatos foram declarados como positivos e 5 como negativos para checkers. Nesse relatório, apresentamos uma análise caso a caso dos repositórios identificados como checkers, indicando se eram de fato apenas checkers ou se também continham PoCs.

Tentando aprimorar a qualidade das fontes

Como parte do esforço para aprimorar a qualidade das fontes de dados em segurança cibernética, conduzimos uma análise minuciosa sobre a presença de checkers no repositório NomiSec. Após a conclusão desta análise, compartilhamos nossas descobertas com o criador do NomiSec, destacando a identificação dos checkers entre os artefatos coletados. No entanto, apesar de nossas comunicações, os checkers identificados permaneceram no repositório NomiSec até o momento, o que suscita dúvidas sobre a integridade e a precisão dos dados fornecidos pela plataforma.

Essa observação ressalta a importância crítica da validação contínua dos dados em repositórios de segurança. Além disso, destaca a necessidade urgente de transparência e ação corretiva por parte dos mantenedores desses repositórios para garantir a confiabilidade e utilidade das informações disponibilizadas. A persistência dos checkers no NomiSec evidencia a urgência de vigilância constante e adoção de medidas proativas para salvaguardar a qualidade das fontes de informação utilizadas pela comunidade de segurança cibernética. Destacamos que *checkers* em geral são inóquos, mas artefatos intencionalmente maléficos, e.g., que contenham a linha `rm -rf ~/`, podem ser apre-

sentados como provas de conceito para 1) confundir a comunidade de segurança, aumentando o risco associado a uma vulnerabilidade e 2) atrapalhar o trabalho dos profissionais de segurança.

4. Representando, organizando, visualizando e etiquetando artefatos

Durante nossa investigação, identificamos uma limitação na organização dos dados no repositório NomiSec, os quais eram estruturados exclusivamente com base nos anos de publicação das CVEs, como por exemplo, CVE-1999-identificador até CVE-2024-identificador. Esta abordagem de organização revelou-se menos favorável para pesquisadores ou estudantes que necessitavam localizar vulnerabilidades específicas ou estratégias de mitigação. Dada a amplitude de CVEs publicadas anualmente, que podem variar de uma única a milhares, torna-se imperativo adotar métodos mais eficientes de organização e classificação dos dados.

Com o intuito de contornar essa questão, exploramos a utilização da técnica de clusterização, juntamente com a visualização dos clusters por meio da técnica t-SNE (t-distributed Stochastic Neighbor Embedding).

4.1. Metodologia ingênua para representar, organizar e visualizar os dados: TF-IDF seguido de K-Means e t-SNE

O processo de clusterização dos textos extraídos dos arquivos JSON envolve várias etapas. Inicialmente, os arquivos JSON são percorridos e o texto relevante de cada documento é extraído. Isso é realizado através de uma função de extração que concatena os campos 'name', 'html_url' e 'description' de cada arquivo, gerando uma coleção de textos representativos dos documentos originais.

Em seguida, os textos extraídos são transformados em vetores numéricos usando a técnica de frequência de termos-inverso da frequência nos documentos (TF-IDF). Durante essa etapa, as stopwords em inglês são removidas para evitar que termos comuns prejudiquem a representação semântica dos textos.

Os vetores TF-IDF resultantes são então submetidos ao algoritmo K-Means para realizar a clusterização dos documentos. O K-Means agrupa os documentos em k clusters, onde k é um número pré-definido. Neste estudo, utilizamos k=10 como número de clusters. O algoritmo atribui cada documento ao cluster mais próximo com base na similaridade dos vetores TF-IDF.

Para visualizar os resultados da clusterização em um espaço bidimensional, é utilizado o algoritmo t-SNE (t-distributed Stochastic Neighbor Embedding). O t-SNE mapeia os vetores TF-IDF em um espaço 2D, preservando as relações de proximidade entre os pontos e permitindo uma visualização intuitiva das estruturas de similaridade nos dados.

Essas etapas compõem o processo de clusterização dos textos dos arquivos JSON, permitindo a identificação de padrões e estruturas semânticas nos dados que podem ser úteis para análises posteriores.

Limitações. Ao aplicar de forma ingênua a clusterização aos dados do NomiSec e visualizá-los usando t-SNE, observamos que os clusters formados refletiam principalmente a estrutura já existente, baseada nos anos das CVEs, não contribuindo significativamente para a organização ou compreensão dos dados (Figura 4).

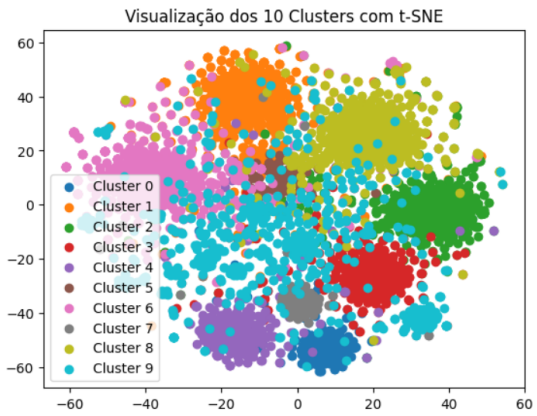


Figura 2. Visualização dos clusters considerando os anos de publicação

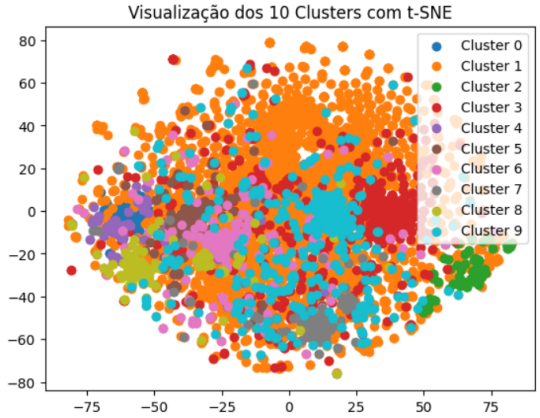


Figura 3. Visualização dos clusters sem considerar os anos e com remoção de *stop words*

4.2. Tentando contornar os desafios, removendo datas e termos pouco informativos

Diante desse obstáculo, tentamos refinar o processo de clusterização removendo os anos e possíveis palavras irrelevantes, na esperança de obter resultados mais claros e úteis. No entanto, essa abordagem não se mostrou eficaz, resultando em uma visualização confusa dos dados.

Limitações. Simplesmente remover datas e termos pouco informativos não é suficiente para gerar uma visualização adequada dos dados. Para tal, é necessário enriquecer os dados com informações advindas de outras fontes, fazendo o cruzamento de dados, por exemplo, entre o NVD e o NomiSec. Alternativamente, podemos também fazer o etiquetamento manual dos dados, conforme descrito a seguir.

4.3. Aplicando novas formas de visualização

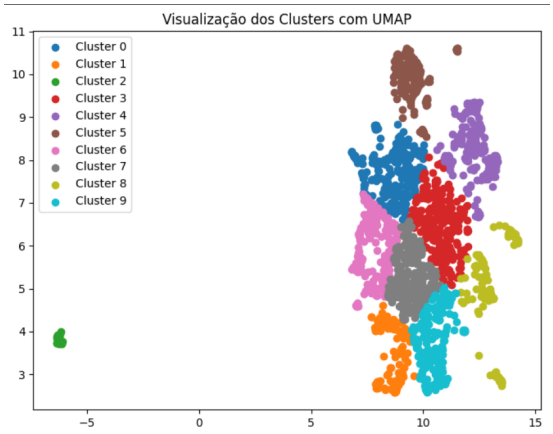


Figura 4. Visualização dos Clusters com UMAP

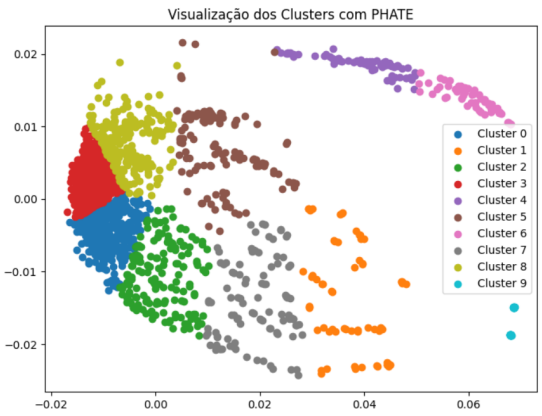


Figura 5. Visualização dos clusters com PHATE

4.4. Rumo ao etiquetamento manual para enriquecer os dados

Como resultado, optamos por uma abordagem alternativa e mais direta: a criação de um sistema de etiquetamento (taggeamento) manual. Para isso, realizamos uma análise mi-

nuciosa de uma amostragem representativa de CVEs de cada ano, abrangendo o período de 1999 a 2024. Durante essa análise, examinamos os códigos, o readme e as descrições associadas a cada CVE, a fim de criar etiquetas que possibilitassem uma identificação mais clara e eficiente dos artefatos. Algumas das etiquetas criadas incluem: #python, #scapylibrary, #DoS (Denial of Service), #port-scanning, #CCSInjection (Cross-Channel Scripting Injection), #java, #C, entre outras. Essas etiquetas foram cuidadosamente selecionadas para descrever detalhes importantes e pertinentes de cada CVE, abordando aspectos específicos como linguagens de programação utilizadas, técnicas de ataque, tipos de vulnerabilidades, entre outros. Acreditamos que essa abordagem de etiquetamento agregou valor significativo aos dados, facilitando a pesquisa e recuperação de artefatos específicos e contribuindo para uma melhor compreensão e utilização dos dados disponíveis no repositório NomiSec.

Limitações. O etiquetamento manual é a melhor solução para entender os dados. Entretanto, ele não escala adequadamente para grandes volumes de dados, então precisa ser aliado com outras abordagens, como *active learning*, para seleção adequada de amostras a serem etiquetadas, e cruzamento com dados de outras fontes já pré-etiquetadas por *experts*.

5. Conclusão e trabalhos futuros

Diante da necessidade de classificar, filtrar e anotar artefatos relacionados a vulnerabilidades de software, este estudo visou analisar a disponibilidade e a qualidade dos dados fornecidos pelo repositório NomiSec. A ênfase na análise do NomiSec se justificou pela sua relevância nesse contexto e pelo fato de ser uma fonte pública.

Ao avaliarmos a abrangência do NomiSec, investigamos a interseção entre os dados contidos no NomiSec e aqueles disponíveis no InTheWild, uma fonte amplamente reconhecida e utilizada na comunidade de segurança da informação. Buscamos também avaliar repositórios que aparentemente não continham Provas de Conceito (PoC) propriamente ditas, mas apenas ferramentas de verificação de vulnerabilidades, conhecidas como "checkers". Após achar alguns deles, colaboramos com o criador do repositório para que esses checkers fossem removidos. Por fim, iniciamos um processo de marcação manual, visando estruturar os dados provenientes do NomiSec. Essa etapa tornou-se essencial, uma vez que observamos que os dados não estavam categorizados ou estruturados de forma adequada. O objetivo dessa marcação manual foi organizar e padronizar os dados, permitindo uma análise mais sistemática e detalhada das vulnerabilidades de software registradas no repositório.

6. Relatório de Atividades

Durante a Semana da Iniciação Científica, apresentei um projeto intitulado ”Classificação de artefatos sobre vulnerabilidades de software”. Nosso objetivo principal foi classificar, filtrar e anotar artefatos relacionados a vulnerabilidades de software, como exploits, códigos de testes de conceito (PoC) e patches.

Utilizamos diversas bases de dados públicas, incluindo NomDesk, InTheWild e o NVD (National Vulnerability Database). A metodologia aplicada envolveu a extração de dados do GitHub, técnicas de web scraping, análise de dados e a aplicação de algoritmos

de clustering e expressões regulares para organizar e reduzir a redundância dos artefatos coletados.

Nossas contribuições gerais incluem a organização dos dados sobre artefatos de vulnerabilidades e a redução de informações duplicadas. Especificamente, destacamos a descoberta de padrões entre os dados e a visualização desses padrões, iniciando o processo de anotação manual.

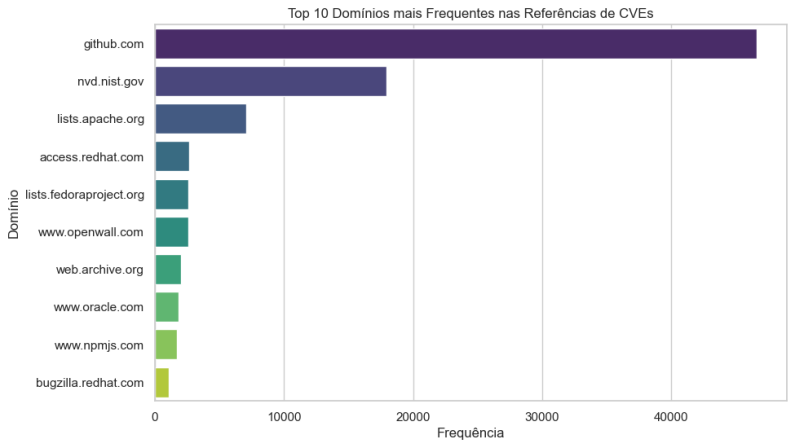
Esta experiência foi enriquecedora, permitindo-me colaborar com colegas e aprofundar meus conhecimentos em segurança da informação e técnicas de mineração de dados. Agradeço ao suporte das instituições envolvidas e à orientação dos nossos professores.

7. Análise e comparativos de novos repositórios

Durante nossas pesquisas para compreender melhor as fontes de dados utilizadas em nossos clusters, analisamos algumas descrições que estávamos usando, bem como seus remetentes. Nesse processo, nos deparamos com um novo repositório de dados no GitHub chamado CVE-Advisories, que chamou nossa atenção pela quantidade significativa de dados disponíveis. A partir disso, começamos a realizar análises e comparações, especialmente com os repositórios que já conhecíamos.

7.1. Validando as referências utilizadas pelo CVE-Advisories

Em nossa análise inicial, investigamos as fontes das referências desse repositório e identificamos os principais links fornecidos por eles, para constituir os seus dados que já foram revisados, somando aproximadamente 20.000 arquivos.



7.2. Dando início ao cruzamento de informações presentes nos repositórios Nomisec e CVE-Advisories

Após identificarmos ser uma fonte de dados, com referências confiáveis, começamos a tentar cruzar informações entre os repositórios como por exemplo as datas de publicação em cada um deles para gerarmos uma visualização do quão rápida é feita a postagem dos dados como pode ser visto na imagem a seguir.

	cve_number	Data de Publicação	pushed_at	Dif em dias	Dif em horas	Dif em minutos	Primeiro a publicar
0	CVE-2023-4863	2024-08-30 23:37:20+00:00	2023-12-18 04:25:00+00:00	256	6163.0	369792.0	NomiSec_dates
1	CVE-2023-4863	2024-08-30 23:37:20+00:00	2023-09-25 16:09:48+00:00	340	8167.0	490047.0	NomiSec_dates
2	CVE-2023-4863	2024-08-30 23:37:20+00:00	2023-09-25 22:13:12+00:00	340	8161.0	489684.0	NomiSec_dates
3	CVE-2023-4863	2024-08-30 23:37:20+00:00	2023-09-29 01:44:51+00:00	336	8085.0	485152.0	NomiSec_dates
4	CVE-2023-4863	2024-08-30 23:37:20+00:00	2023-10-01 00:48:15+00:00	334	8038.0	482329.0	NomiSec_dates
...	...	...	...	...	...	...	...
3693	CVE-2013-3827	2022-05-17 03:13:10+00:00	2023-08-05 08:29:14+00:00	-446	-10686.0	-641117.0	advisories_df_epss_unique
3694	CVE-2010-1622	2022-05-17 03:28:34+00:00	2023-04-18 14:15:42+00:00	-337	-8075.0	-484488.0	advisories_df_epss_unique
3695	CVE-2010-1622	2022-05-17 03:28:34+00:00	2022-03-31 14:26:47+00:00	46	1117.0	67021.0	NomiSec_dates
3696	CVE-2010-1622	2022-05-17 03:28:34+00:00	2022-12-05 02:36:35+00:00	-202	-4848.0	-290829.0	advisories_df_epss_unique
3697	CVE-2010-1622	2022-05-17 03:28:34+00:00	2023-02-20 08:30:38+00:00	-280	-6702.0	-402063.0	advisories_df_epss_unique

* Data de Publicação: Referente a data de publicação no repositório CVE-Advisories

* Pushed-at: Referente a data de publicação no repositório Nomisec

A imagem acima é resultado da união dos dados presentes no Nomisec com os do CVE-Advisories.Dos dados totais disponiveis para cada repositório, uma parcela de 3698 arquivos está simultaneamente nos dois repositórios, e após esta análise descobrimos que aproximadamente 2 terços desta parcela presente em ambos os repositórios foi publicada primeiramente no CVE-Advisories como observado na imagem a seguir.

```
Primeiro a publicar
advisories_df_epss_unique    2519
NomiSec_dates                1179
```

Após este estudo, ficamos ainda mais interessados neste novo repositório tendo como próximos objetivos cruzar cada vez mais informações para termos comparativos não so com o Nomisec e sim com outros repositórios para agregar em nossos futuros trabalhos.

8. Avaliação feita pelo próprio Bolsista

Participar da Iniciação Científica com este projeto foi uma experiência extremamente valiosa. Os aprendizados adquiridos, tanto nos aspectos técnicos quanto na dinâmica de trabalho em equipe, são fundamentais para meu desenvolvimento acadêmico e profissional. Identificar as áreas de melhoria me permite focar em habilidades específicas que necessitam de aperfeiçoamento, garantindo uma evolução contínua na minha trajetória científica. Agradeço a todos os envolvidos e estou ansioso para aplicar esses conhecimentos em futuros desafios e projetos.

Referências

Figueiredo, C., Lopes, J. G., Azevedo, R., Vieira, D., Miranda, L., Zaverucha, G., de Aguiar, L. P., and Menasché, D. S. (2023). A statistical relational learning approach towards products, software vulnerabilities and exploits. *IEEE Transactions on Network and Service Management*.

Miranda, L., Figueiredo, C., Menasché, D. S., and Kocheturov, A. (2023). Patch or exploit? nvd assisted classification of vulnerability-related github pages. pages 511–522.

Miranda, L., Vieira, D., de Aguiar, L. P., Menasché, D. S., Bicudo, M. A., Nogueira, M. S., Martins, M., Ventura, L., Senos, L., and Lovat, E. (2021). On the flow of

software security advisories. *IEEE Transactions on Network and Service Management*, 18(2):1305–1320.

Ponce, L. M. S., Gimpel, M., Fazzion, E., Cunha, Í., Hoepers, C., Steding-Jessen, K., Chaves, M. H., Guedes, D., and Meira Jr, W. (2022). Caracterização escalável de vulnerabilidades de segurança: um estudo de caso na internet brasileira. *Simpósio Brasileiro de Redes de Computadores e Sistemas Distribuídos*.

Rokon, M. O. F., Islam, R., Darki, A., Papalexakis, E. E., and Faloutsos, M. (2020). {SourceFinder}: Finding malware {Source-Code} from publicly available repositories in {GitHub}. pages 149–163.

Yadmani, S. E., The, R., and Gadyatskaya, O. (2022). Beyond the surface: Investigating malicious cve proof of concept exploits on github. *arXiv preprint arXiv:2210.08374*.