DISCIPLINARUM SCIENTIA *Disciplinarum Scientia.* **Série: Naturais e Tecnológicas**
Santa Maria, v. 25, n. 3, p. 53-67, 2024
Recebido em: 20.12.2024. Aprovado em: 20.12.2024.
ISSN 2176-462X | DOI: doi.org/10.37779/nt.v25i3.5210

AVALIAÇÃO DE FERRAMENTAS DE ANÁLISE DE VULNERABILIDADES EM IMAGENS DOCKER: UMA ABORDAGEM AHP[1]

ASSESSMENT OF DOCKER IMAGES VULNERABILITY ANALYSIS TOOLS: AN AHP APPROACH

Ali Iddar[2] e Rogério C. Turchetti[3]

RESUMO

Este estudo avalia o desempenho de ferramentas de análise de vulnerabilidades em imagens Docker, utilizando o Processo de Análise Hierárquica (AHP) como metodologia de tomada de decisão multicritério. Diante do crescente uso de contêineres Docker e dos riscos associados às vulnerabilidades presentes nas imagens disponíveis no Docker Hub, a pesquisa visa comparar e classificar as principais ferramentas destinadas à identificação dessas fragilidades. O AHP foi aplicado em duas etapas: inicialmente, em cada estudo individualmente e, posteriormente, nos resultados consolidados de todos os estudos. Os resultados indicaram que a escolha da ferramenta deve levar em conta não apenas a eficácia geral, mas também fatores contextuais e a possibilidade de utilizar múltiplas ferramentas para uma detecção mais precisa. A pesquisa destaca ainda a necessidade de avaliações contínuas, especialmente em relação às abordagens de análise dinâmica e suas comparações com ferramentas de análise estática.

Palavras-chave: análise hierárquica; contêineres; segurança.

ABSTRACT

In this study, we evaluated the performance of vulnerability analysis tools for Docker images. We used the Analytic Hierarchy Process (AHP) as a multi-criteria decision-making methodology. Given the growing use of Docker containers and the risks associated with vulnerabilities present in their images, which are available on Docker Hub, this research aims to compare and rank the main tools for identifying these weaknesses. We applied the AHP methodology in two stages: initially, in each study individually and, later, in the results of all the studies. The results indicated that the choice of tool should take into account not only overall effectiveness but also contextual factors and the possibility of using multiple tools for more accurate detection. This work also highlights the need for continuous evaluations, especially regarding dynamic analysis approaches and their comparison with static analysis tools.

Keywords: *hierarchical analysis; container; security.*

1 Revisão Bibliográfica Sistemática / Artigo de pesquisa aplicada.

2 Discente do curso Superior de Tecnologia em Redes de Computadores, CTISM - Universidade Federal de Santa Maria (UFSM). E-mail: ali-iddar@redes.ufsm.br. ORCID: https://orcid.org/0000-0003-3229-0519

3 Docente do curso Superior de Tecnologia em Redes de Computadores e do Mestrado Acadêmico PPGEPT - CTISM - Universidade Federal de Santa Maria (UFSM). E-mail: turchetti@redes.ufsm.br.

INTRODUÇÃO

O Docker emergiu como uma tecnologia revolucionária no campo da virtualização, oferecendo uma abordagem eficiente e flexível para o desenvolvimento, implantação e gerenciamento de aplicações. A plataforma Docker utiliza contêineres que compartilham o kernel do sistema operacional hospedeiro, resultando em uma solução mais leve e eficiente do que máquinas virtuais (MEADUSANI, 2018). A popularidade do Docker cresceu rapidamente com o Docker Hub, repositório de imagens mantidas e gerenciadas por fornecedores certificados e pela comunidade (Shu, 2017). Essa facilidade de compartilhamento, embora benéfica para a comunidade de desenvolvedores, também introduz riscos significativos à segurança.

O estudo de Shu (2017) analisou 356.218 imagens no Docker Hub, revelando que tanto as imagens oficiais quanto as da comunidade contêm, em média, mais de 180 vulnerabilidades quando consideradas todas as versões. Alarmantemente, mais de 80% das imagens, independentemente de sua origem, apresentam pelo menos uma vulnerabilidade de alta gravidade (SHU, 2017). Esses dados destacam a necessidade crítica de uma abordagem mais rigorosa em relação à segurança das imagens usadas nos contêineres.

A segurança no contexto do Docker envolve múltiplos aspectos. É essencial garantir que as imagens estejam livres de vulnerabilidades e/ou configurações incorretas antes de sua distribuição. Isso inclui a varredura contínua em busca de *malware*, utilizando conjuntos de assinaturas e métodos de detecção heurística comportamental (SOUPPAYA, 2017). Além disso, práticas como o armazenamento seguro de segredos fora das imagens e a manutenção de um conjunto confiável de imagens e registros são cruciais para mitigar riscos de segurança.

O ecossistema Docker apresenta desafios únicos de segurança devido à sua natureza. Os contêineres se comunicam diretamente com o kernel da máquina hospedeira, o que, embora eficiente, potencialmente aumenta a superfície de ataque (ALYAS, 2022). O Docker utiliza namespaces do Linux para isolamento, mas certos sistemas de arquivos do kernel permanecem não isolados, representando riscos adicionais. A complexidade do ambiente Docker é evidenciada pela diversidade de vulnerabilidades encontradas. Um estudo revelou que 36% das imagens oficiais no Docker Hub contêm vulnerabilidades *Common Vulnerabilities and Exposures* (*CVE*) de alta prioridade (MARTIN, 2018). Além disso, o movimento DevOps (Desenvolvimento e Operações), facilitado pelo Docker, pode inadvertidamente levar à inclusão de ferramentas de desenvolvimento ou versões de pacotes desatualizadas nas imagens, aumentando ainda mais os riscos de segurança.

Neste contexto, as ferramentas de análise de vulnerabilidades tornam-se fundamentais. Elas desempenham um papel vital na identificação e mitigação de riscos de segurança em imagens Docker. Essas ferramentas são capazes de realizar varreduras detalhadas, detectando vulnerabilidades conhecidas, configurações incorretas e até mesmo código malicioso potencial. A importância

dessas ferramentas é amplificada pelo fato de que muitos usuários não estão cientes dos riscos associados às imagens que utilizam. Um estudo mostrou que 97% dos usuários se preocupam apenas com a execução bem-sucedida da imagem, ignorando parâmetros sensíveis nos comandos de execução (LIU, 2020). Essa falta de conscientização ressalta a necessidade de ferramentas automatizadas e eficazes para análise de segurança. Além disso, o tempo médio para corrigir uma vulnerabilidade em imagens Docker é significativamente maior do que em software comum - 422 dias em comparação com 181 dias (LIU, 2020). Esse atraso na correção de vulnerabilidades aumenta a janela de oportunidade para potenciais ataques, tornando ainda mais crítico o uso de ferramentas de análise de vulnerabilidades como parte integrante do ciclo de vida de desenvolvimento e implantação de contêineres.

Por outro lado, a análise dinâmica observa o comportamento do contêiner durante sua execução (BRADY, 2020). Essa abordagem permite a detecção de vulnerabilidades que só se manifestam em tempo de execução, como comportamentos maliciosos ou anomalias no uso de recursos. Técnicas de análise dinâmica em um dos estudos analisados utilizaram algoritmos de aprendizado de máquina não supervisionado para identificar padrões anômalos nas chamadas de sistema, no uso de recursos ou no tráfego de rede (TUNDE-ONADELE, 2019)

A principal diferença entre as duas abordagens reside na natureza e no momento da análise. Enquanto a análise estática oferece uma visão abrangente das vulnerabilidades potenciais antes da execução do contêiner, a análise dinâmica proporciona insights sobre o comportamento real do contêiner em tempo de execução (PINNAMANENI, 2022). A análise estática é geralmente mais rápida e menos intensiva em recursos, mas pode não detectar vulnerabilidades que só se manifestam durante a execução. Já a análise dinâmica, embora mais intensiva em recursos e potencialmente mais lenta (BRADY, 2020) pode identificar ameaças que escapam à análise estática (TUNDE-ONADELE, 2019).

Ao longo dos últimos anos, foram desenvolvidas diversas ferramentas de análise de vulnerabilidades em imagens Docker, e diversos estudos foram realizados avaliando a eficácia dessas ferramentas. Este estudo visa realizar uma avaliação comparativa das ferramentas de análise de vulnerabilidades em imagens Docker, avaliando os resultados desses estudos e comparando-os através do uso da metodologia de Processo de Análise Hierárquica. Os estudos foram selecionados por meio de uma busca sistemática nas seguintes bases acadêmicas: *IEEE Xplore*, *EI Compendex*, *Web of Science*, *ACM Digital Library*, *Scopus* e Google Acadêmico. Foram encontradas 62 publicações, submetidas a um processo de seleção com base em critérios pré-definidos e estabelecidos na metodologia. Essa seleção resultou em 10 estudos que avaliaram as ferramentas em termos de eficácia de detecção de vulnerabilidades, como é mostrado na Tabela 1.

Tabela 1 - Os estudos selecionados na revisão sistemática.

ID	Estudo	Ano
Pub-1	A Study on Container Vulnerability Exploit Detection	2019
Pub-2	*Container Vulnerability Scanners: An Analysis*	2020
Pub-3	*An Analysis of Security Vulnerabilities in Container Images for Scientific Data Analysis*	2021
Pub-4	*An Evaluation of Container Security Vulnerability Detection Tools*	2021
Pub-5	Segurança em Imagens Docker: Um Estudo de Ferramentas de Análise Estática	2021
Pub-6	*Concerns About Available Container Image Scanning Tools and Image Security*	2022
Pub-7	Investigating the Inner Workings of Container Image Vulnerability Scanners	2022
Pub-8	*Continuous Docker Image Analysis and Intrusion Detection Based on Open-Source Tools*	2022
Pub-9	*Vulnerability Analysis of Docker Hub Official Images and Verified Images*	2023
Pub-10	*Detecting Container Vulnerabilities Leveraging the CI/CD Pipeline*	2023

Fonte: Construção do Autor.

MATERIAL E MÉTODOS

Para avaliar a eficácia das ferramentas de detecção de vulnerabilidades em imagens Docker de forma mais eficaz e objetiva, e eliminar as dificuldades impostas pelos múltiplos critérios de avaliação usados nos estudos analisados, foi adotada a abordagem de Tomada de Decisão Multicritério (*MCDM*, do inglês *Multi-Criteria Decision Making*) [4]. Entre os diversos métodos de tomada de decisão multicritério (MCDM), neste estudo utilizou-se o método Processo de Hierarquia Analítica (*AHP*, do inglês *Analytic Hierarchy Process*).

PROCESSO DE ANÁLISE HIERÁRQUICA (AHP)

O Processo de Análise Hierárquica (AHP) é uma metodologia de tomada de decisão desenvolvida por Thomas L. Saaty entre as décadas de 1970 e 1980 (BERNASCONI, 2009). Essa técnica se destaca por sua capacidade de abordar problemas complexos que envolvem múltiplos critérios de avaliação. A principal vantagem do AHP é sua habilidade de estruturar um problema de forma hierárquica, permitindo a avaliação sistemática de fatores quantitativos e qualitativos (BADRI, 1999). No contexto deste trabalho, o AHP se mostra particularmente útil devido à complexidade inerente ao problema, onde diversos aspectos técnicos e práticos precisam ser considerados simultaneamente. A capacidade do AHP de incorporar julgamentos pessoais de forma estruturada (VARGAS, 1990) é, especialmente, relevante neste cenário.

Para este estudo, o método AHP foi adotado para classificar as ferramentas de análise de vulnerabilidades em uma escala de 1 a 5 estrelas, considerando não apenas as métricas de desempenho,

4 *Multi-Criteria Decision Making*: é um dos ramos mais conhecidos da teoria de tomada de decisão. O *MCDM* lida com problemas de decisão que envolvem múltiplos critérios, que podem estar em conflito entre si e ter unidades de medidas diferentes. Os critérios recebem pesos de importância e o problema pode ser representado em um formato chamado de matriz de decisão (TRIANTAPHYLLOU, 2000).

mas também a quantidade de estudos realizados sobre cada ferramenta. Essa abordagem visa proporcionar uma classificação mais robusta e confiável.

O método AHP foi aplicado em duas etapas: (1) aplicação do AHP em cada estudo individualmente: nesta etapa, o AHP é utilizado para classificar as ferramentas conforme as métricas e resultados específicos de cada estudo; (2) aplicação do AHP nos resultados consolidados: após a análise individual, os resultados são consolidados e submetidos a uma nova aplicação do AHP para obter uma classificação geral das ferramentas. O processo de aplicação do AHP em cada etapa segue quatro passos fundamentais, apresentados na sequência deste trabalho.

MONTAR A MATRIZ DE JULGAMENTO

A montagem da matriz de julgamento é uma etapa crucial na aplicação do método AHP. Este processo envolve a comparação dos critérios utilizando a escala de Saaty (SAATY, 2006). Essa escala numérica, que varia de 1 a 9 com valores intermediários, expressa a importância relativa entre os elementos comparados. Na escala de Saaty, um critério pode ter importância moderada (valor 3) em relação a outro critério (valor 1/3), por exemplo. Essa abordagem permite uma avaliação mais precisa das relações entre os diferentes critérios considerados na análise.

Após a realização das comparações pareadas, os julgamentos são organizados em uma matriz quadrada. A interpretação dos valores na matriz, segundo Bhushan (2004) segue uma lógica específica:

- Os elementos na diagonal da matriz são sempre iguais a 1, indicando que um critério é igualmente importante a si.
- Se o valor do elemento na linha *i* e coluna *j* da matriz for maior que 1, isso indica que o critério na linha *i* é considerado mais importante do que o critério na coluna *j*.
- Se o valor do elemento (*i*,*j*) for menor que 1, isso significa que o critério na coluna *j* é considerado mais importante que o critério na linha *i*.

Um princípio fundamental da AHP é o da comparação inversa. De acordo com este princípio, o elemento na posição (*j*,*i*) da matriz, que representa a comparação inversa, deve ser recíproco do elemento (*i*,*j*). Em outras palavras, se o critério da linha *i* é x vezes mais importante que o critério da coluna *j*, então o critério da coluna *j* é 1/x vezes mais importante que o critério da linha *i* (SAATY, 2003).

CÁLCULO DOS PESOS

O processo de cálculo dos pesos dos critérios segue uma sequência de etapas bem definidas. Esse processo é fundamental para determinar a importância relativa de cada critério de avaliação

utilizado. O primeiro passo consiste em calcular a soma de cada coluna da matriz de julgamentos. Em seguida, realiza-se uma normalização dos elementos da matriz, dividindo cada elemento pela soma correspondente à sua coluna (ISHIZAKA, 2006). Esse procedimento resulta em uma nova matriz normalizada, na qual os elementos de cada coluna somam 1.

A normalização dos elementos da matriz tem dois objetivos principais: o primeiro é padronizar a escala dos julgamentos, garantindo que todos os valores estejam em uma mesma faixa (de 0 a 1). O segundo assegura que a soma dos pesos dos critérios seja igual a 1 (ou 100%), refletindo adequadamente a distribuição das prioridades.

Essa abordagem permite uma comparação direta dos pesos relativos de cada critério, já que todos os valores estão em uma mesma escala normalizada. O passo final envolve o cálculo da média aritmética de cada linha da matriz normalizada (ISHIZAKA, 2006). Essa média representa o peso final do critério correspondente àquela linha. É importante notar que a soma dos pesos de todos os critérios deve ser igual a 1 (ou 100%), garantindo uma distribuição proporcional da importância entre os critérios.

VERIFICAÇÃO DA CONSISTÊNCIA

A etapa seguinte consiste em avaliar a consistência da matriz de julgamento. Esse processo é realizado por meio do cálculo da Razão de Consistência (RC), que permite quantificar o grau de inconsistência presente na matriz de julgamento. O processo de verificação da consistência, segundo Bhushan (2004) segue uma sequência de passos bem definidos:

1. Obtenção do vetor de pesos dos critérios: este vetor é calculado multiplicando cada coluna da matriz de julgamento pelo peso correspondente do critério, previamente calculado. O resultado é um vetor onde cada elemento representa o somatório dos produtos entre os elementos da coluna e seus respectivos pesos.
2. Cálculo do autovalor máximo (λmax): para cada elemento do vetor de pesos, divide-se pelo peso correspondente do critério. A soma desses resultados é então dividida pelo número de critérios (*n*), obtendo-se assim o autovalor máximo (λmax) da matriz de julgamento.
3. Cálculo do Índice de Consistência (IC): O IC é calculado utilizando a fórmula a seguir: (1) $IC = (\lambda max - n)(n - 1)$ onde *n* é o número total de critérios na matriz de julgamento. Este índice é um passo intermediário essencial para o cálculo final da Razão de Consistência.
4. Cálculo da Razão de Consistência (RC): A RC é obtida através da fórmula a seguir: (2) $RC = IC/IR$ onde IC é o Índice de Consistência calculado anteriormente, e IR é o Índice de Consistência Randômico, um valor tabelado que varia conforme o tamanho da matriz (valor de *n*).

A verificação da consistência é um passo crucial na aplicação do método AHP, pois permite avaliar a coerência dos julgamentos realizados. Uma baixa Razão de Consistência indica que os julgamentos são suficientemente consistentes, conferindo maior confiabilidade aos resultados obtidos na avaliação das ferramentas de detecção de vulnerabilidades em imagens Docker.

CLASSIFICAÇÃO DAS FERRAMENTAS

O último passo do processo AHP é a classificação das ferramentas de análise de vulnerabilidades em imagens Docker em uma escala de 1 a 5 estrelas, com base em sua eficácia de detecção. O processo de classificação segue as seguintes etapas:

1. Cálculo da pontuação final: para cada ferramenta, multiplica-se os resultados obtidos em cada critério pelo respectivo peso calculado anteriormente. A soma desses resultados ponderados resulta na pontuação final.
2. Definição dos intervalos de classificação: A pontuação máxima obtida é dividida em 5 intervalos iguais, representando as classificações de 1 a 5 estrelas. O intervalo mais alto corresponde à melhor classificação (5 estrelas), enquanto o mais baixo corresponde à classificação inferior (1 estrela).
3. Atribuição da classificação em estrelas: cada ferramenta é associada ao seu respectivo intervalo de classificação, atribuindo-lhe a classificação em estrelas correspondente

Essa abordagem de classificação em estrelas oferece uma comparação direta e intuitiva do desempenho relativo de cada ferramenta de análise de vulnerabilidades em imagens Docker.

A classificação em estrelas proporciona uma representação clara e compreensível da eficácia de cada ferramenta, facilitando a interpretação dos resultados por parte de profissionais e pesquisadores da área de segurança de contêineres. Essa metodologia de classificação permite uma rápida identificação das ferramentas mais eficazes, auxiliando na tomada de decisões informadas sobre a escolha e implementação de soluções de análise de vulnerabilidades em ambientes Docker

AUTOMATIZAÇÃO DO AHP

Para facilitar e agilizar a aplicação do método AHP na avaliação das ferramentas de análise de vulnerabilidades em imagens Docker, foram desenvolvidas planilhas automatizadas utilizando o Microsoft Excel[5]. Essas planilhas foram projetadas para realizar os cálculos complexos do AHP de

5 As planilhas foram disponibilizadas publicamente para consulta e reutilização. Elas podem ser acessadas através do seguinte repositório no GitHub: https://github.com/ali-id/AHP-Calc-Worksheets.

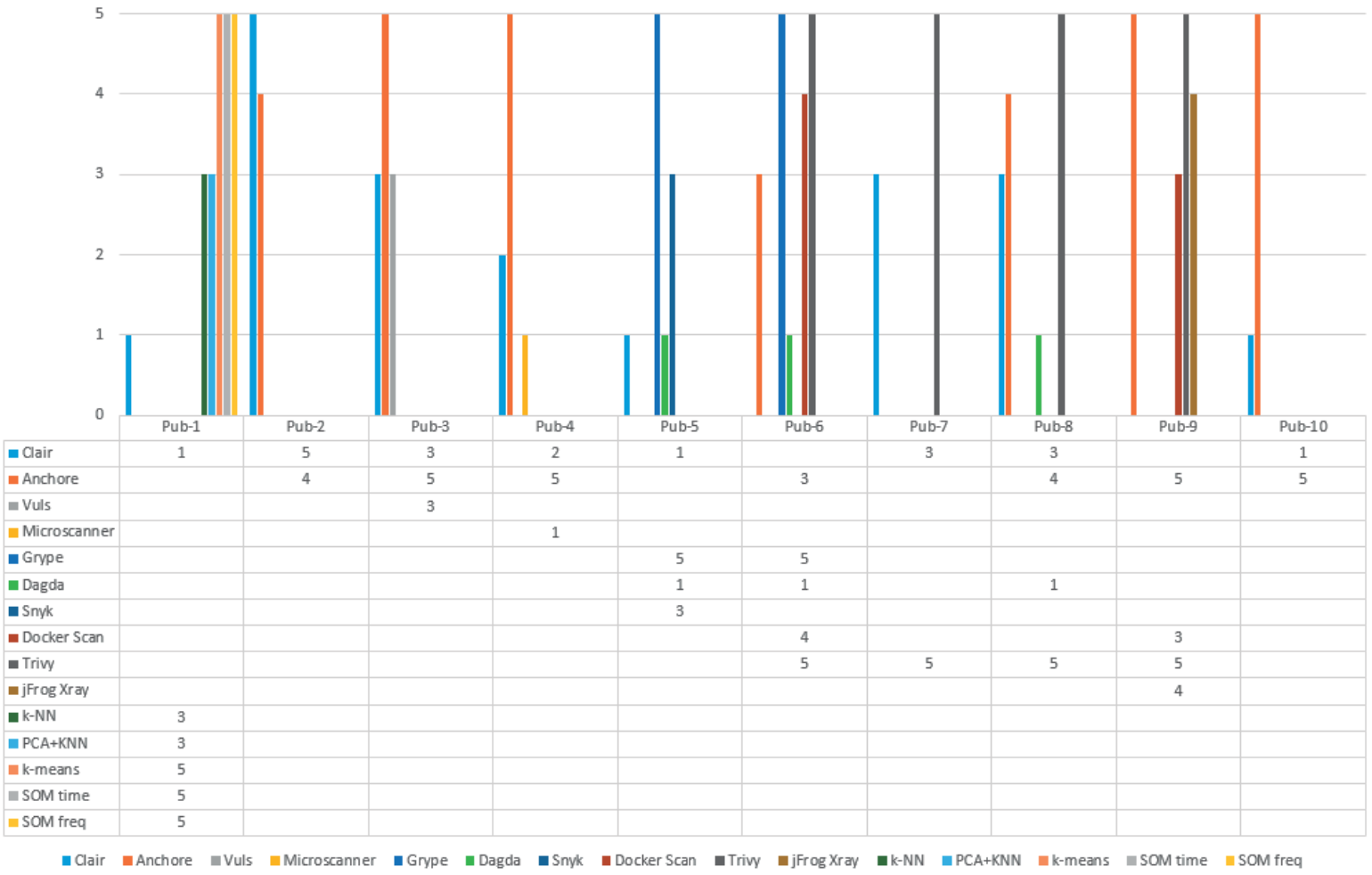
forma automática, reduzindo significativamente o tempo e o esforço necessários para a análise. As planilhas automatizadas abrangem as principais etapas do método AHP, incluindo: construção da matriz de julgamentos, cálculo dos pesos dos critérios, verificação da consistência dos julgamentos e a classificação final das alternativas (ferramentas).

Para utilizar essas planilhas, o usuário precisa apenas fornecer as seguintes informações: os critérios de avaliação, as ferramentas a serem classificadas, as comparações pareadas entre os critérios e os resultados dos experimentos realizados com cada ferramenta. Uma vez inseridos esses dados, as planilhas realizam automaticamente todos os cálculos necessários, gerando a classificação final das ferramentas em uma escala de 1 a 5 estrelas. A automatização do processo AHP através dessas planilhas oferece várias vantagens, incluindo a redução de erros de cálculo, economia de tempo na análise e padronização do processo de avaliação.

RESULTADOS E DISCUSSÃO

A avaliação e classificação das ferramentas de análise de vulnerabilidades em imagens Docker foi realizada através da aplicação do Processo de Análise Hierárquica (AHP) em duas etapas distintas. Essa abordagem permitiu uma análise abrangente e objetiva da eficácia de detecção de cada ferramenta. Na primeira etapa, o método AHP foi aplicado individualmente para cada publicação analisada, considerando as métricas específicas utilizadas em cada estudo. A Figura 1 apresenta um resumo visual dos resultados dessa aplicação inicial do AHP, mostrando a pontuação obtida por cada ferramenta em cada estudo analisado. Na segunda etapa, os resultados consolidados de todos os estudos foram submetidos a uma nova aplicação do AHP. Esse processo resultou em uma classificação geral das ferramentas em uma escala de 1 a 5 estrelas, conforme ilustrado na Figura 2.

Figura 1 - Resultados da aplicação do método AHP por ferramenta e por estudo.



	Pub-1	Pub-2	Pub-3	Pub-4	Pub-5	Pub-6	Pub-7	Pub-8	Pub-9	Pub-10
Clair	1	5	3	2	1		3	3		1
Anchore		4	5	5		3		4	5	5
Vuls			3							
Microscanner				1						
Grype					5	5				
Dagda					1	1		1		
Snyk					3					
Docker Scan						4			3	
Trivy						5	5	5	5	
jFrog Xray									4	
k-NN	3									
PCA+KNN	3									
k-means	5									
SOM time	5									
SOM freq	5									

Fonte: Construção do Autor.

A análise dos resultados revela que a ferramenta Anchore se destacou significativamente, obtendo a pontuação máxima (5 estrelas) na classificação geral. Esse desempenho notável é confirmado pela consistência das avaliações elevadas que a Anchore recebeu em diversos estudos, como evidenciado na Figura 1. Especificamente, a Anchore foi objeto de análise em 7 estudos diferentes, demonstrando um desempenho relevante, isto é: obteve a nota máxima (5 estrelas) em 4 estudos, recebeu 4 estrelas em dois estudos e alcançou 3 estrelas em um dos estudos. Essa consistência nas avaliações, juntamente com a ampla cobertura em diferentes estudos, contribuiu significativamente para a classificação de topo da Anchore na avaliação geral.

A ferramenta Trivy demonstrou um desempenho notável, ocupando a segunda posição na classificação geral, como ilustrado na Figura 2. Essa posição é resultado de um desempenho consistente nos estudos em que foi avaliada. Especificamente, o Trivy foi objeto de análise em 4 estudos diferentes e, em cada um deles, alcançou a classificação máxima de 5 estrelas, como pode ser observado na Figura 1. Essa consistência na obtenção da pontuação máxima é um indicador significativo da eficácia e confiabilidade da ferramenta Trivy na detecção de vulnerabilidades em imagens Docker.

Um aspecto particularmente interessante dos resultados é a comparação direta entre Trivy e Anchore nos estudos em que ambas foram avaliadas conjuntamente. Especificamente, nos estudos identificados como Pub-6 e Pub-8, o Trivy superou o desempenho da Anchore. Esse resultado sugere que, em determinados contextos ou para certos tipos de análise, o Trivy pode oferecer vantagens sobre a Anchore. Essa observação é especialmente relevante considerando que a Anchore ocupou a primeira posição na classificação geral. O fato de o Trivy superar a Anchore em comparações diretas, mesmo ocupando a segunda posição geral, destaca a importância de considerar não apenas as classificações gerais, mas também os resultados de comparações específicas ao escolher uma ferramenta para um determinado ambiente ou caso de uso.

Figura 2 - Resultados da aplicação do método AHP nos resultados consolidados todos os estudos.

Classificação geral das ferramentas		
Anchore	5	Estrela(S)
Dagda	1	Estrela(S)
Docker Scan	1	Estrela(S)
Grype	2	Estrela(S)
Trivy	4	Estrela(S)
Clair	3	Estrela(S)
Vuls	1	Estrela(S)
Microscanner	1	Estrela(S)
Snyk	2	Estrela(S)
jFrog Xray	1	Estrela(S)
k-NN	1	Estrela(S)
PCA+KNN	1	Estrela(S)
k-means	1	Estrela(S)
SOM time	1	Estrela(S)
SOM freq	1	Estrela(S)

Fonte**:** Construção do Autor.

A ferramenta Clair ocupou a terceira posição na classificação geral das ferramentas de análise de vulnerabilidades em imagens Docker, conforme ilustrado na Figura 2. No entanto, uma análise mais detalhada de seu desempenho nos estudos individuais revela um padrão de resultados bastante variado e inconsistente. Ao examinar a Figura 1, que apresenta os resultados da aplicação do método AHP por ferramenta e por estudo, observa-se uma significativa flutuação no desempenho da Clair, isto é: em um dos estudos, a Clair alcançou a classificação máxima de 5 estrelas, demonstrando excelente eficácia na detecção de vulnerabilidades. Por outro lado, em três estudos diferentes, a Clair recebeu a classificação mínima de 1 estrela, indicando um desempenho consideravelmente abaixo das expectativas nesses casos. Nos demais estudos, o desempenho da Clair variou entre esses extremos. A inconsistência observada no desempenho da Clair é um aspecto importante a ser considerado por profissionais e pesquisadores ao avaliar essa ferramenta para uso em seus projetos. Enquanto a Clair demonstra potencial para excelente desempenho em alguns cenários, sua variabilidade sugere que pode não ser igualmente eficaz em todos os contextos.

A ferramenta Grype, desenvolvida pela empresa Anchore, demonstrou um desempenho notável nos estudos em que foi avaliada. Conforme ilustrado na Figura 1, a Grype alcançou a classificação máxima de 5 estrelas nos dois estudos que a analisaram. Esse resultado é particularmente relevante, considerando que a Grype é uma solução relativamente nova no mercado de ferramentas de análise de vulnerabilidades em imagens Docker. Apesar de sua classificação geral ter sido impactada pelo número limitado de estudos que a avaliaram, a Grype mostrou-se altamente competitiva. Em diversas métricas de avaliação, ela superou ferramentas mais estabelecidas como Anchore e Trivy. Esse desempenho superior em comparações diretas sugere que a Grype apresenta potencial para a detecção de vulnerabilidades em contêineres. É importante destacar uma informação relevante sobre o cenário atual das ferramentas de análise de vulnerabilidades: em janeiro de 2023, a Anchore, em sua versão de código aberto, foi oficialmente descontinuada (ANCHORE, 2023). Como substituta, a empresa desenvolvedora recomenda o uso das ferramentas Grype e Syft, ou a versão comercial Anchore Enterprise.

Essa transição da Anchore para a Grype como a principal ferramenta recomendada pela empresa é um fator importante a ser considerado. Ela sugere que a Grype pode incorporar melhorias e avanços baseados na experiência acumulada com o desenvolvimento e uso da Anchore e que o foco de desenvolvimento e suporte da empresa está agora direcionado para a Grype, o que pode resultar em atualizações e melhorias mais frequentes. O excelente desempenho da Grype nos estudos disponíveis, combinado com sua posição como sucessora recomendada da Anchore, sugere que essa ferramenta merece atenção especial de profissionais e pesquisadores da área de segurança de contêineres. Embora mais estudos sejam necessários para estabelecer uma classificação geral mais robusta, os resultados apresentados indicam que a Grype pode se tornar uma das principais ferramentas no campo de análise de vulnerabilidades em imagens Docker.

As ferramentas Docker Scan e JFrog Xray apresentaram resultados intermediários em suas avaliações individuais, oferecendo insights interessantes sobre seu desempenho na análise de vulnerabilidades em imagens Docker. O JFrog Xray foi avaliado em apenas um estudo, no qual obteve uma classificação de 4 estrelas. Esse resultado indica um desempenho satisfatório, sugerindo que a ferramenta é capaz de realizar análises de vulnerabilidades com potencial. No entanto, a limitação de ter sido avaliado em apenas um estudo torna difícil fazer afirmações mais abrangentes sobre sua consistência ou eficácia em diferentes cenários.

Por outro lado, o Docker Scan apresentou resultados mais variados nos estudos em que foi analisado. Em um estudo, alcançou a classificação máxima de 5 estrelas, demonstrando excelente desempenho, e 3 estrelas em outro, indicando um desempenho moderado. Essa variação nos resultados do Docker Scan sugere que sua eficácia pode depender do contexto específico ou dos tipos de vulnerabilidades analisadas em cada estudo. Um ponto importante a ser destacado em relação ao Docker Scan é a observação feita no estudo Pub-6. Nesse estudo específico, a métrica de número de vulnerabilidades identificadas foi excluída da aplicação do método AHP para o Docker Scan.

O motivo dessa exclusão foi a constatação de que 77% das vulnerabilidades detectadas por essa ferramenta eram duplicadas. Essa informação indica uma potencial limitação do Docker Scan em distinguir entre vulnerabilidades únicas e duplicadas, e o número bruto de vulnerabilidades detectadas pode não ser um indicador confiável da eficácia da ferramenta. A exclusão dessa métrica no estudo Pub-6 representa uma abordagem metodológica importante, pois evita que o desempenho do Docker Scan seja artificialmente inflado por detecções duplicadas. Isso ressalta a necessidade de considerar não apenas a quantidade, mas também a qualidade e a relevância das vulnerabilidades detectadas ao avaliar a eficácia de uma ferramenta de análise.

A ferramenta Snyk, embora avaliada em apenas um estudo, demonstrou um desempenho mediano. Nesse único estudo em que foi analisada, a Snyk alcançou a segunda colocação, ficando atrás apenas da ferramenta Grype. Nesse estudo, a Snyk superou outras ferramentas bem estabelecidas como Clair e Dagda em termos de eficácia na detecção de vulnerabilidades. Esse desempenho da Snyk sugere que, apesar da limitada avaliação, a ferramenta possui potencial significativo para a análise de vulnerabilidades em imagens Docker. No entanto, é importante ressaltar que uma avaliação mais abrangente, envolvendo múltiplos estudos e diferentes cenários, seria necessária para confirmar a consistência desse desempenho. Em contraste, outras ferramentas de análise estática, incluindo Vuls, Dagda e MicroScanner, apresentaram resultados menos expressivos em comparação com as ferramentas líderes. Essas ferramentas obtiveram classificações inferiores tanto nos resultados individuais de cada estudo quanto na classificação geral consolidada. Vuls demonstrou um desempenho abaixo da média nos estudos em que foi avaliada, indicando possíveis limitações em sua capacidade de detecção de vulnerabilidades em comparação com outras ferramentas. A ferramenta Dagda, apesar de ser avaliada em múltiplos estudos, consistentemente recebeu classificações baixas, sugerindo que pode ter dificuldades em competir com ferramentas mais avançadas em termos de eficácia de detecção. Por fim, MicroScanner apresentou resultados menos expressivos, indicando que pode não oferecer o mesmo nível de detecção de vulnerabilidades que as ferramentas líderes do mercado.

Na análise das ferramentas de detecção dinâmica de vulnerabilidades em imagens Docker, os algoritmos *Self-Organizing Map* (*SOM*) e *K-means* demonstraram um desempenho excepcional em suas avaliações individuais, ambos alcançando a pontuação máxima de 5 estrelas. Entre esses dois, o algoritmo SOM apresentou uma ligeira vantagem sobre o K-means, indicando uma eficácia marginalmente superior na detecção de vulnerabilidades. Em contraste, os algoritmos *K-Nearest Neighbors* (*K-NN*) e *Principal Component Analysis + K-Nearest Neighbors* (*PCA+KNN*) apresentaram um desempenho moderado, recebendo uma classificação de 3 estrelas. Nesta comparação, o PCA+KNN demonstrou uma pequena vantagem, sugerindo que a adição da análise de componentes principais (PCA) oferece uma melhoria, ainda que modesta, na eficácia da detecção.

Esses resultados das avaliações individuais sugerem que os algoritmos SOM e K-means podem ser particularmente eficazes na análise dinâmica de vulnerabilidades em imagens Docker,

superando significativamente as abordagens baseadas em K-NN. No entanto, um aspecto importante a ser considerado é o resultado da classificação geral, que levou em conta todos os estudos analisados na revisão sistemática. Nessa classificação geral, todas as ferramentas de análise dinâmica obtiveram apenas 1 estrela. Esse resultado, aparentemente contraditório em relação às avaliações individuais, indica uma clara necessidade de conduzir mais estudos comparativos envolvendo essas ferramentas de análise dinâmica e outras ferramentas de análise de vulnerabilidades.

CONCLUSÃO

Com base nos resultados apresentados neste trabalho, conclui-se que as ferramentas Anchore e Trivy se destacaram consistentemente como as mais eficazes e precisas na detecção de vulnerabilidades em imagens Docker. Seu desempenho superior em múltiplos estudos as posiciona como opções confiáveis para profissionais e pesquisadores da área. Embora avaliada em apenas dois estudos, a ferramenta Grype demonstrou um alto desempenho, indicando seu potencial como uma solução eficaz. Seu desenvolvimento como sucessora da Anchore sugere que ela pode se tornar uma das principais ferramentas no futuro próximo. A ferramenta Clair apresentou resultados inconsistentes, com desempenho insatisfatório na maioria dos estudos. Essa variabilidade ressalta a importância de considerar o contexto específico de uso ao selecionar ferramentas de análise de vulnerabilidades. Para as demais ferramentas de análise estática, a revisão sistemática evidenciou uma clara necessidade de mais estudos avaliativos. Essa lacuna na pesquisa representa uma oportunidade importante para futuros trabalhos na área. Outro ponto importante a ser destacado é que a utilização combinada de diferentes ferramentas pode resultar em uma detecção mais abrangente e precisa de vulnerabilidades. Essa abordagem multi-ferramenta pode compensar as limitações individuais de cada solução. No campo da análise dinâmica, os algoritmos SOM e K-means apresentaram os melhores resultados. No entanto, a escassez de estudos comparativos entre essas abordagens dinâmicas e as ferramentas de análise estática indica outra importante área para pesquisas futuras.

REFERÊNCIAS

ALYAS, T. *et al.* **Container Performance and Vulnerability Management for Container Security Using Docker Engine**. Security and Communication Networks, v. 2022, 2022. Disponível em: https://onlinelibrary.wiley.com/doi/abs/10.1155/2022/6819002. Acesso em: ago. 2024.

ANCHORE. **Anchore Engine**. Anchore, Inc, 2023. Disponível em: https://github.com/anchore/anchore-engine. Acesso em: jun. 2024.

BADRI, M. **Combining the analytic hierarchy process and goal programming for global facility location-allocation problem**. International Journal of Production Economics, v. 62, 1999. Disponível em: https://www.sciencedirect.com/science/article/pii/S0925527398002497. Acesso em: jul. 2024.

BERNASCONI, M.; CHOIRAT, C.; SERI, R. **The Analytic Hierarchy Process and the Theory of Measurement**. University of Venice "Ca' Foscari", Department of Economics, Working Papers, v. 56, 2009. Disponível em: https://www.dse.univr.it/documenti/Seminario/documenti/documenti803241.pdf. Acesso em: ago. 2024.

BHUSHAN, N.; RAI, K.; CAHYONO, S. **Strategic Decision Making: Applying the Analytic Hierarchy Process (Decision Engineering)**. Springer, 2004.

BRADY, K. *et al.* **Docker Container Security in Cloud Computing**. In: 2020 10th Annual Computing and Communication Workshop and Conference (CCWC). 2020. p. 975-980. Disponível em: https://ieeexplore.ieee.org/document/9031195. Acesso em: 3 ago. 2024.

LIU, P. *et al.* **Understanding the Security Risks of Docker Hub**. In: COMPUTER Security - ESORICS 2020: 25th European Symposium on Research in Computer Security, ESORICS 2020, Guildford, UK, September 14-18, 2020, Proceedings, Part I. Springer-Verlag, 2020. Disponível em: https://dl.acm.org/doi/10.1007/978-3-030-58951-6_13. Acesso em: jun. 2024.

MARTIN, A. *et al.* **Docker ecosystem - Vulnerability Analysis.** Computer Communications, v. 122, 2018. Disponível em: https://www.sciencedirect.com/science/article/pii/S0140366417300956. Acesso em: ago. 2024.

MEADUSANI, S. R. **Virtualization using docker containers: For reproducible environments and containerized applications**. In: Culminating Projects in Information Assurance. 2018. Disponível em: https://repository.stcloudstate.edu/msia_etds/50. Acesso em: set. 2024.

SAATY, T.; VARGAS, L. **The analytic network process**. 2006. p. 1-26. Disponível em: https://www.researchgate.net/publication/226556079_The_Analytic_Network_Process. Acesso em: jul. 2024.

SAATY, T. L. **Decision-making with the AHP: Why is the principal eigenvector necessary?** European Journal of Operational Research, 2003. Disponível em: https://www.sciencedirect.com/science/article/pii/S0377221702002278. Acesso em: jul. 2024.

SHU, R.; GU, X.; ENCK, W. **A Study of Security Vulnerabilities on Docker Hub**. In: Proceedings of the Seventh ACM on Conference on Data and Application Security and Privacy. 2017. Disponível em: https://dl.acm.org/doi/10.1145/3029806.3029832. Acesso em: ago. 2024.

SOUPPAYA, M.; MORELLO, J.; SCARFONE, K. **Application Container Security Guide**. 2017. Disponível em: https://csrc.nist.gov/pubs/sp/800/190/final. Acesso em: jun. 2024.

TRIANTAPHYLLOU, E. **Multi-Criteria Decision Making Methods: A Comparative Study**. Springer, 2000. v. 44.

VARGAS, L. G. **An overview of the analytic hierarchy process and its applications**. European Journal of Operational Research, v. 48, 1990. Disponível em: https://www.sciencedirect.com/science/article/pii/037722179090056H. Acesso em: jul. 2024.

ISHIZAKA, A.; LUSTI, M. **How to derive priorities in AHP: A comparative study**. Central European Journal of Operations Research, v. 14, 2006. Disponível em: https://link.springer.com/article/10.1007/s10100-006-0012-9. Acesso em: maio 2024.