# SSVC Copilot: um Assistente de Priorização de Vulnerabilidades Baseado no Arcabouço SSVC

**Jan Filho[1], Arthur Albuquerque[1], Eduardo Falcão[2], Andrey Brito[1]**

[1]Unidade Acadêmica de Sistemas e Computação – UFCG

[2]Departamento de Engenharia de Computação e Automação – UFRN

jan.filho@lsd.ufcg.edu.br, arthur.albuquerque@lsd.ufcg.edu.br

eduardo@dca.ufrn.edu.br, andrey@computacao.ufcg.edu.br

***Resumo.*** *O elevado número de vulnerabilidades identificadas em sistemas torna a priorização de correções um problema relevante. Embora métricas como CVSS e EPSS sejam amplamente utilizadas, elas não consideram o contexto operacional e organizacional. Neste trabalho, apresentamos o SSVC Copilot, um assistente de priorização de vulnerabilidades baseado no arcabouço Stakeholder-Specific Vulnerability Categorization (SSVC). A solução utiliza agentes especializados para inferir pontos de decisão do SSVC a partir de múltiplas fontes de Cyber Threat Intelligence, combinando automação e interação humano–IA. A avaliação com 144.163 CVEs publicados entre 2016 e 2026 demonstrou que o SSVC Copilot alcançou taxas de concordância de até 97,18% com o CISA Vulnrichment. Por fim, os resultados indicam que o contexto fornecido pelo humano permite à IA incorporar sua avaliação de risco e sugerir decisões futuras alinhadas.*

***Abstract.*** *The large number of vulnerabilities identified in systems makes patch prioritization a relevant challenge. Although metrics such as CVSS and EPSS are widely used, they do not consider operational and organizational context. In this work, we present SSVC Copilot, a vulnerability prioritization assistant based on the Stakeholder-Specific Vulnerability Categorization (SSVC) framework. The solution employs specialized agents to infer SSVC decision points from multiple Cyber Threat Intelligence sources, combining automation with human–AI interaction. The evaluation conducted on 144,163 CVEs published between 2016 and 2026 demonstrated that SSVC Copilot achieved agreement rates of up to 97.18% with CISA Vulnrichment. Finally, the results indicate that the context provided by humans enables the AI to incorporate their risk assessment and suggest future decisions accordingly.*

## 1. Introdução

Uma das principais preocupações durante o desenvolvimento de aplicações é a segurança. Mesmo quando desenvolvedores projetam cuidadosamente a arquitetura do sistema e adotam processos rigorosos de implantação, vulnerabilidades permanecem inevitáveis. Além de inevitáveis, as vulnerabilidades tendem a surgir em grande número. Isso acontece porque o amplo conjunto de bibliotecas e ferramentas normalmente incorporadas às aplicações modernas amplia a superfície de ataque. Para lidar com isso, ferramentas de monitoramento da cadeia de suprimentos de *software* permitem o rastreamento automatizado de

dependências e vulnerabilidades associadas a elas. No entanto, um desafio fundamental persiste: dado o elevado número de vulnerabilidades identificadas e os recursos humanos limitados disponíveis para corrigi-las, **quais vulnerabilidades devem ser priorizadas?**

Abordagens tradicionais para priorização de vulnerabilidades baseiam-se em sistemas de pontuação consolidados, como o *Common Vulnerability Scoring System* (CVSS) [FIRST 2024] e o *Exploit Prediction Scoring System* (EPSS) [Jacobs et al. 2021], que atribuem a cada vulnerabilidade um nível de severidade ou de explorabilidade. Na prática, essas abordagens são frequentemente operacionalizadas por meio da definição de valores de corte para filtrar vulnerabilidades – por exemplo, priorizando apenas aquelas acima de um determinado limiar. No entanto, tais métricas capturam dimensões distintas e incompletas do risco: enquanto o CVSS reflete predominantemente o impacto potencial de uma vulnerabilidade, desconsiderando sua exploração no mundo real, o EPSS estima a probabilidade de exploração observada, mas não incorpora o impacto ou o contexto específico do ativo afetado. Como resultado, vulnerabilidades de alta severidade podem nunca ser exploradas, enquanto vulnerabilidades de alta probabilidade de exploração podem ter impacto limitado no ambiente em questão.

A aplicação de limiares ajuda a reduzir o volume de vulnerabilidades a serem tratadas, mas não fornece uma visão holística de risco que considere simultaneamente impacto, explorabilidade e contexto operacional. Para lidar com essa limitação, o arcabouço *Stakeholder-Specific Vulnerability Categorization* (SSVC) [CISA 2023] foi proposto como uma abordagem estruturada e orientada à decisão, permitindo mapear sinais técnicos e contextuais em recomendações de priorização acionáveis. O SSVC modela a priorização de vulnerabilidades como um processo estruturado e sensível ao contexto, e não como uma abordagem puramente quantitativa baseada em pontuação. Em vez de atribuir um único valor numérico, o SSVC avalia vulnerabilidades com base em um conjunto de critérios de decisão, como o status de exploração e automatização da vulnerabilidade, o nível de exposição do sistema, e o impacto humano que ela causa.

Nesse contexto, podemos mencionar o *Vulnrichment Project* [CISA 2026] e a abordagem de automatização de "SSVC de Pior Caso" [Tsuji et al. 2026]. O *Vulnrichment* é um projeto mantido pela *Cybersecurity and Infrastructure Security Agency* (CISA) com o objetivo de enriquecer os registros de vulnerabilidades públicas com os pontos de decisão do SSVC. No entanto, o projeto não detalha como os pontos de decisão do SSVC são atribuídos às vulnerabilidades e ignora algumas diretrizes do arcabouço SSVC, o que dificulta a análise da consistência e da acurácia das decisões produzidas. A abordagem de "SSVC de Pior Caso" tem como contribuição a automação da decisão de impacto humano, um ponto de decisão não coberto pelo *Vulnrichment*. Por outro lado, automatizar essa decisão não permite incorporar plenamente as nuances de uma avaliação de risco realizada por um humano.

Neste trabalho, propomos o **SSVC Copilot**, um assistente de suporte à decisão para a priorização de vulnerabilidades que operacionaliza o SSVC por meio de agentes especializados. Esses agentes agregam informações heterogêneas provenientes de fontes externas para apoiar a atribuição de valores aos critérios do SSVC. O SSVC Copilot eleva a cobertura de vulnerabilidades analisadas em comparação ao *Vulnrichment*, além de aprimorar a acurácia dos pontos de decisão ao implementar mais diretrizes do arcabouço. Além disso, o SSVC Copilot se inspira na abordagem de "SSVC de Pior Caso"

para acelerar a avaliação de impacto humano, incorporando também o histórico de avaliações realizadas pelo operador para sugerir recomendações futuras alinhadas ao contexto e aos padrões previamente observados.

## 2. Fundamentação Teórica

### 2.1. Cadeia de Suprimento de Software e Descoberta de Vulnerabilidades

A adoção comum de bibliotecas de terceiros e de componentes reutilizáveis tem ampliado a superfície de ataque de aplicações modernas, tornando a **cadeia de suprimentos de *software*** um vetor crítico de vulnerabilidades. Nesse contexto, a identificação de dependências e de suas vulnerabilidades associadas tornou-se essencial para a gestão de risco. O primeiro passo para identificar as vulnerabilidades associadas a um *software* é criar uma lista dos componentes do *software* (SBOM, abreviação do inglês *Software Bill of Materials*). Ferramentas como Trivy [Aqua Security 2024] e Snyk [Snyk Ltd. 2024] automatizam esse processo ao analisar artefatos (e.g., imagens de contêiner, código-fonte) e listar suas vulnerabilidades, utilizando bases de dados oficiais.

### 2.2. Priorização de Vulnerabilidades por Sistemas de Pontuação

O ***Common Vulnerability Scoring System* (CVSS)** [FIRST 2024] é amplamente utilizado para quantificar a severidade de vulnerabilidades por meio de um modelo baseado em métricas que capturam diferentes dimensões do risco técnico. O escore final (*base score*), no intervalo [0, 10], é derivado de métricas que representam a viabilidade de exploração e o impacto técnico potencial da vulnerabilidade.

Como exemplo, a vulnerabilidade *CVE-2021-44228* (Log4Shell) [NIST 2021] possui pontuação máxima no CVSS, com *Base Score* 10.0 (*Critical*) e vetor *CVSS:3.1/AV:N/AC:L/PR:N/UI:N/S:C/C:H/I:H/A:H*. Esse vetor resume uma exploração remota, de baixa complexidade, sem privilégios ou interação do usuário, com alto impacto sobre confidencialidade, integridade e disponibilidade.

O ***Exploit Prediction Scoring System* (EPSS)** [Jacobs et al. 2021] é um modelo estatístico que estima a probabilidade de uma vulnerabilidade ser explorada ativamente no mundo real em um horizonte temporal definido (tipicamente 30 dias). Diferentemente do CVSS, que foca no impacto potencial, o EPSS baseia-se em dados empíricos e técnicas de aprendizado de máquina, considerando fatores como as características da vulnerabilidade, a disponibilidade de *exploits* e padrões históricos de exploração. O resultado é um escore probabilístico contínuo no intervalo [0, 1], que complementa o CVSS ao fornecer uma estimativa orientada à explorabilidade.

### 2.3. Fontes de Informação de CTI

Uma vulnerabilidade específica é identificada por um ***Common Vulnerabilities and Exposures* (CVE)**, que representa uma instância única e publicamente catalogada de uma falha de segurança em um sistema. Além disso, vulnerabilidades frequentemente são associadas a categorias de fraquezas descritas no ***Common Weakness Enumeration* (CWE)** [MITRE Corporation 2026], que classifica os tipos de erros de projeto ou de implementação que levam à sua ocorrência. Enquanto o CVE representa a vulnerabilidade específica, o CWE representa a causa raiz dessa vulnerabilidade.

O ***National Vulnerability Database*** **(NVD)** [NIST 2024] consolida vulnerabilidades catalogadas como CVEs, com metadados estruturados, descrições técnicas e métricas como o CVSS, além da classificação CWE quando disponível. Complementarmente, o catálogo ***Known Exploited Vulnerabilities*** **(KEV)** [CISA) 2024] lista vulnerabilidades com exploração confirmada em cenários reais, indicando risco ativo. A ***Exploit Database*** **(ExploitDB)** [Offensive Security 2024] reúne provas de conceito públicas, enquanto o ***Metasploit Framework*** [Rapid7 2024] disponibiliza módulos reutilizáveis para testes e validação. Em conjunto, essas fontes fornecem descrições, severidade e evidências práticas de exploração, enriquecendo a análise e a priorização de vulnerabilidades.

### 2.4. Decisão: o Arcabouço SSVC

O *Stakeholder-Specific Vulnerability Categorization* (SSVC) é um arcabouço de priorização que busca transformar sinais relevantes de severidade, explorabilidade e risco ativo em uma decisão estruturada e acionável. O SSVC é orientado ao contexto do agente responsável pela decisão, prevendo modelos distintos para diferentes *stakeholders*, como fornecedor, coordenador e implantador (*deployer*). Neste trabalho, adota-se o modelo do implantador, voltado à priorização de correções e mitigações em sistemas em operação. Esse modelo inclui os seguintes pontos de decisão:

- **Exploração:** estado atual da exploração da vulnerabilidade;
- **Exposição do sistema:** nível de exposição do sistema afetado;
- **Automatização:** potencial de exploração em larga escala;
- **Impacto humano:** impacto organizacional e na segurança de indivíduos.

A combinação desses fatores é organizada por uma lógica estruturada de decisão, representada como uma árvore[1], que conduz a categorias de prioridade, como **adiar, agendar, corrigir fora do ciclo regular ou corrigir imediatamente** a vulnerabilidade. Dessa forma, o SSVC atua como um mecanismo de integração de múltiplos sinais, permitindo a tomada de decisões mais consistentes e alinhadas ao contexto operacional.

## 3. Trabalhos Relacionados

Com base em uma análise da literatura recente (2020 – 2026), identificamos quatro linhas de pesquisa em sistemas de CTI alinhadas ao presente trabalho.

1. **Priorização de ameaças e vulnerabilidades:** aplicação de heurísticas e técnicas baseadas em IA para identificar, avaliar e priorizar os riscos mais críticos;
2. **Colaboração humano–IA em CTI:** integração entre especialistas humanos e sistemas de CTI, com o objetivo de aumentar a acurácia, a interpretabilidade e a consciência situacional na análise de ameaças;
3. **Representação, modelagem e enriquecimento de conhecimento em CTI:** transformação de dados brutos de ameaças em inteligência estruturada, semanticamente rica e interoperável, por meio de ontologias e modelagem de entidades;
4. **Sistemas de CTI autônomos e agentivos:** desenvolvimento de sistemas de CTI baseados em agentes autônomos, nos quais grandes modelos de linguagens (LLMs – *Large Language Models*) evoluem de ferramentas passivas a entidades capazes de raciocinar, orquestrar e executar ações.

A Tabela 1 apresenta os principais trabalhos relacionados classificados nessas categorias.

[1]A árvore de decisão SSVC pode ser visualizada em `https://certcc.github.io/SSVC/howto/deployer_tree/#deployer-decision-points`.

**Tabela 1. Classificação dos trabalhos relacionados.**

| Ano | Trabalho | Priorização | Humano-IA | Enriquecimento de Dados | Agentes Autônomos |
|---|---|---|---|---|---|
| 2020 | [Sadlek 2020] | | | ✓ | |
| 2021 | [Mavroeidis et al. 2021] | | | ✓ | |
| 2025 | [Janloy et al. 2025] | | ✓ | ✓ | ✓ |
| | [Kshetri and Voas 2025] | | ✓ | | ✓ |
| | [Spyros et al. 2025] | | | ✓ | |
| | [Liu et al. 2025] | ✓ | | ✓ | |
| | [Cohen et al. 2025] | | ✓ | ✓ | |
| | [Saccone et al. 2025] | | ✓ | ✓ | ✓ |
| | [Gustavsson et al. 2025] | ✓ | ✓ | ✓ | |
| 2026 | [Foundjem et al. 2026] | ✓ | | ✓ | ✓ |
| | *CISA Vulnrichment* [CISA 2026] | ✓ | | ✓ | |
| | SSVC de Pior Caso [Tsuji et al. 2026] | ✓ | | | |
| | SSVC Copilot | ✓ | ✓ | ✓ | ✓ |

No que concerne à **priorização de ameaças e vulnerabilidades**, o CyLens [Liu et al. 2025] propõe uma abordagem agentiva baseada em LLM em que um único modelo atua como agente central de raciocínio, realizando tarefas como correlação, contextualização e priorização de vulnerabilidades a partir de padrões como CVE, CVSS e EPSS. Os resultados mostram que modelos especializados em um domínio apresentam melhor desempenho em tarefas complexas. Já o CDDOM [Gustavsson et al. 2025] utiliza ferramentas de varredura de vulnerabilidades, combinando SBOMs e VEX (*Vulnerability Exploitability eXchange*, usado para indicar se uma vulnerabilidade afeta um produto) para ajudar na priorização, além de integrá-las a informações de severidade técnica, impacto de negócio e superfície de ataque. [Foundjem et al. 2026] propõe um arcabouço multiagente que realiza priorização por meio de redes neurais, combinando métricas como o CVSS com fatores contextuais, como a explorabilidade e a taxa de sucesso de ataques. Por fim, os projetos CISA *Vulnrichment* [CISA 2026] e SSVC de Pior Caso [Tsuji et al. 2026] contribuem para a priorização de vulnerabilidades ao automatizar os resultados de diferentes critérios de decisão do arcabouço SSVC.

No contexto de **colaboração humano–IA em CTI**, [Janloy et al. 2025] propõem uma plataforma agentiva centrada no humano, na qual agentes de IA validam, enriquecem e pontuam a inteligência enquanto garantem conformidade com requisitos de privacidade, mantendo o humano no processo de revisão e tomada de decisão. De forma semelhante, [Cohen et al. 2025] apresentam uma abordagem baseada em aprendizado recíproco humano–máquina, na qual modelos de classificação são continuamente refinados por especialistas, que também contribuem para a construção de representações conceituais do cenário de ameaças. O CDDOM [Gustavsson et al. 2025] também incorpora o humano ao processo, especialmente em etapas que exigem validação ou evidência para auditoria.

Com relação à **representação, modelagem e enriquecimento de conhecimento em CTI**, [Sadlek 2020] investiga diversos padrões e fontes de dados, como CVE, CVSS e STIX (*Structured Threat Information Expression*), para identificar relações entre eles e enriquecer informações sobre ameaças e vulnerabilidades, propondo uma ferramenta para a gestão de ativos e a identificação proativa de vulnerabilidades. [Mavroeidis et al. 2021] propõem uma abordagem baseada em ontologias para modelar o conhecimento sobre atores de ameaça, permitindo inferência automática por meio de raciocínio lógico. Já [Spyros et al. 2025] apresentam o arcabouço ThreatWise AI, que automatiza o ciclo de vida de CTI e enriquece os dados por meio de taxonomias, correlação e classificação, utilizando padrões como o STIX para o compartilhamento de informações. O trabalho de [Saccone et al. 2025] também realiza enriquecimento, ainda que de forma implícita, por meio de recuperação e contextualização de fontes heterogêneas. Por fim, o CISA *Vulnrichment* [CISA 2026] enriquece os registros de vulnerabilidades com informações sobre o SSVC, incluindo os níveis de automação e de explorabilidade da vulnerabilidade.

Na categoria de **sistemas de CTI autônomos e agentivos**, [Kshetri and Voas 2025] discutem o potencial transformador de sistemas agentivos distribuídos, nos quais múltiplos agentes especializados colaboram para realizar tarefas de detecção e resposta, enfatizando também a importância do humano no processo para governança e validação. [Saccone et al. 2025] propõem um assistente multiagente baseado em LLMs e na geração aumentada por recuperação (RAG – *Retrieval-Augmented Generation*), no qual agentes especializados cooperam para coletar, processar e agregar inteligência. Já [Foundjem et al. 2026] apresentam um sistema multiagente que utiliza pipelines agentivos com RAG para construir grafos de conhecimento sobre ameaças, enquanto [Liu et al. 2025] explora uma abordagem agentiva centralizada baseada em um único LLM com capacidade de raciocínio e de execução de tarefas.

O SSVC Copilot endereça todos os problemas mencionados: (i) ele usa **agentes autônomos** para (ii) implementar a **priorização de vulnerabilidades** SSVC, (iii) utilizando fontes diversas para **enriquecimento de dados** (EPSS, CVSS, KEV, ExploitDB), e (iv) permite uma **interação humano-IA** na validação de decisões tomadas pelos agentes.

## 4. Modelo de Ameaça

O modelo de ameaça adotado assume que as ferramentas de varredura operam corretamente, estão livres de falhas e identificam apenas vulnerabilidades previamente catalogadas como CVEs pelo MITRE/NVD, sem abranger vulnerabilidades de dia zero. Ainda assim, admite-se a possibilidade de falsos positivos, isto é, vulnerabilidades reportadas que não impactam efetivamente a aplicação analisada. A integridade das informações obtidas dessas bases é preservada por protocolos seguros, como o HTTPS, o que torna as adulterações detectáveis. O modelo ainda pressupõe que os agentes estão livres de vulnerabilidades capazes de comprometer seu comportamento ou seu processo decisório. No caso do agente de impacto humano, considera-se que os LLMs utilizados são íntegros e não sofrem comprometimento por ataques externos.

## 5. Arquitetura do SSVC Copilot

A Figura 3 apresenta a arquitetura do SSVC Copilot e o fluxo de processamento adotado. O processo inicia-se com a identificação de vulnerabilidades em artefatos, como imagens de contêineres ou código-fonte, por meio de ferramentas de varredura que retornam

CVEs, a partir das quais os agentes iniciam suas tarefas. O núcleo do sistema concentra-se no enriquecimento dessas vulnerabilidades e na avaliação dos pontos de decisão do SSVC, que são posteriormente consolidados em uma decisão final de priorização.

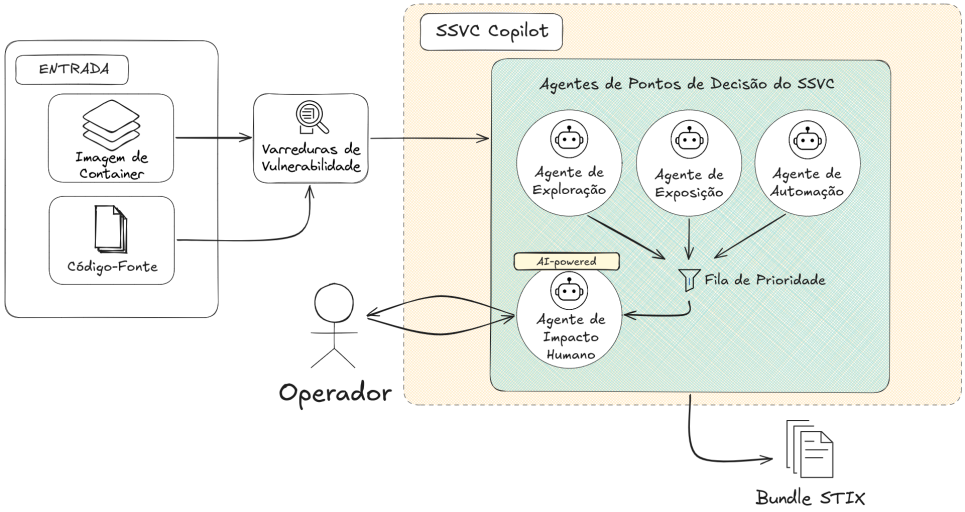


**Figura 1. Arquitetura do SSVC Copilot.**

Com a lista de vulnerabilidades, os agentes responsáveis pelos pontos técnicos do SSVC (exploração, exposição de sistema e automatabilidade) tomam decisões com base nas evidências coletadas na etapa prévia de enriquecimento. Definidos esses três pontos, o resultado parcial pode revelar que algumas decisões SSVC (adiar, agendar, corrigir fora do ciclo regular ou corrigir imediatamente) não são alcançáveis, e essa informação é utilizada para priorizar as vulnerabilidades a serem analisadas pelo agente de impacto humano. Por fim, a interação humano–IA define o último ponto do SSVC: o potencial de impacto humano decorrente da vulnerabilidade, que depende do contexto organizacional. Após isso, a decisão SSVC final é estruturada em STIX, favorecendo a integração com outros sistemas. O processamento pode ser ampliado através da replicação dos agentes.

### 5.1. Ponto de Decisão: Exploração

O ponto de decisão de exploração estima o grau de evidência de exploração associado a uma vulnerabilidade. Neste trabalho, são considerados três possíveis valores: *exploração ativa*, quando há indícios concretos de exploração em ambientes reais; *prova de conceito (PoC)*, quando há artefatos públicos que demonstram a possibilidade de exploração; e *ausência de exploração conhecida*, quando não são encontradas evidências relevantes.

A inferência desse ponto é realizada a partir da combinação de diferentes fontes de evidência. Vulnerabilidades presentes na lista KEV, mantida pela CISA, são diretamente classificadas como exploração ativa, uma vez que essa base representa CVEs com histórico conhecido de exploração real. Para a identificação de vulnerabilidades classificadas como *PoC*, o agente busca *exploits* públicos em fontes amplamente utilizadas pela comunidade de segurança, como ExploitDB [Offensive Security 2024] e Metasploit Framework [Rapid7 2024]. Além disso, referências disponibilizadas pelo NVD também são analisadas, já que frequentemente apontam para repositórios públicos e implementações relacionadas à exploração do CVE.

O agente também incorpora o EPSS como mecanismo complementar de inferência, adotando os limiares apresentados como referência na documentação do SSVC [CERT Coordination Center 2025]. Vulnerabilidades com EPSS superior a 90% são classificadas como exploração ativa, enquanto valores acima de 75% permitem classificá-las como *PoC*, mesmo sem evidências externas explícitas. Além disso, valores acima de 55% são utilizados como fator incremental na decisão previamente obtida, elevando o nível de exploração em uma categoria.

Outra evidência utilizada é a relação entre vulnerabilidades e fraquezas conhecidas (CWE). A documentação do SSVC descreve conjuntos de CWEs associados a caminhos de exploração amplamente conhecidos [CERT Coordination Center 2024a], permitindo assumir a existência de *PoC* para vulnerabilidades diretamente relacionadas a essas fraquezas. Por fim, vulnerabilidades que não satisfazem nenhum dos critérios descritos são classificadas como ausência de exploração conhecida.

### 5.2. Ponto de Decisão: Exposição de Sistema

O ponto de decisão de exposição de sistema busca estimar o grau de exposição do serviço afetado pela vulnerabilidade, considerando os níveis *pequeno*, *controlado* e *aberto*. Diferentemente dos demais pontos técnicos do SSVC, esse fator está mais relacionado às características e configurações do serviço analisado do que à vulnerabilidade em si.

A automatização desse ponto apresenta desafios adicionais, uma vez que determinar a exposição real de um sistema exige conhecimento contextual sobre a rede, a arquitetura e a configuração operacional. Dessa forma, o SSVC Copilot utiliza o vetor de ataque do CVSS como evidência auxiliar para inferir esse ponto. Essa abordagem adota uma estratégia conservadora baseada no pior caso, priorizando cenários em que a vulnerabilidade seja classificada como crítica em vez de subestimar seu potencial de risco.

Como consequência, vulnerabilidades exploráveis remotamente tendem a elevar a exposição inferida, mesmo em situações nas quais o serviço possua exposição prática limitada. Por outro lado, vulnerabilidades que exigem acesso local mantêm classificações mais moderadas mesmo quando associadas a serviços expostos à Internet, já que a própria vulnerabilidade impõe restrições adicionais de exploração. Embora essa abordagem simplifique a automatização do ponto de decisão, o sistema utiliza inicialmente o nível de exposição inferido a partir das informações da vulnerabilidade. Durante a avaliação, esse valor é apresentado ao operador, que pode revisá-lo e ajustá-lo conforme o contexto operacional do serviço. Além disso, a configuração definida pode ser reutilizada em avaliações futuras relacionadas ao mesmo serviço.

### 5.3. Ponto de Decisão: Automação

O ponto de decisão de automação determina se a exploração de uma vulnerabilidade pode ser realizada de forma automatizada (*sim ou não*). Para essa inferência, o SSVC Copilot combina evidências diretas de automação com métricas do CVSS.

A inferência de automatização positiva ocorre quando o CVE apresenta simultaneamente vetor de ataque em rede (*AV: Network*), ausência de interação do usuário (*UI: None*), baixa complexidade de ataque (*AC: Low*) e ausência de privilégios necessários (*PR: None*), condições que indicam viabilidade e facilidade de automação da exploração.

Como evidência complementar, o agente considera a existência de *scripts* e implementações públicas de *exploits* como indicativo de potencial de automatização. Vulnerabilidades que não satisfazem esses requisitos são classificadas como não automatizáveis.

### 5.4. Fila de Prioridade

Como sistemas normalmente apresentam um grande volume de vulnerabilidades e o SSVC inclui um ponto de decisão dependente de avaliação humana, os três primeiros critérios do arcabouço são utilizados para calcular um índice de urgência de avaliação.

Esse índice é derivado dos conjuntos de decisões finais possíveis na árvore de decisão, considerando os cenários de impacto humano: baixo, médio, alto e muito alto. Por exemplo, o estado de um CVE pode ser resumido como **"O/O/I/I"**. Isso indica que o CVE exige correção fora do ciclo habitual (***O**ut-of-cycle*) quando o impacto humano é *baixo* ou *médio*, e correção imediata (***I**mmediate*) quando o impacto é *alto* ou *muito alto*. Assim, um CVE **"O/O/I/I"** deve ser priorizado em relação a um CVE **"O/O/O/O"**, pois o primeiro pode resultar em tratamento *imediato*, enquanto o segundo permanece restrito ao tratamento *fora do ciclo regular*.

### 5.5. Ponto de Decisão: Impacto Humano

O impacto humano depende de contexto organizacional e operacional, tornando sua automatização mais complexa. Assim, o SSVC Copilot adota uma abordagem *human-in-the-loop*, na qual LLMs e mecanismos automatizados apoiam o operador sem substituí-lo.

A etapa inicial do agente de impacto humano utiliza uma estratégia baseada em pior caso, inspirada em [Tsuji et al. 2026]. Nela, o operador informa previamente os possíveis níveis máximos de impacto sobre confidencialidade, integridade e disponibilidade dos sistemas avaliados. Esses dados são combinados com métricas do CVSS para gerar uma sugestão inicial de impacto humano. Durante o processo, a ferramenta disponibiliza um *chatbot* com conhecimento sobre SSVC, sobre a abordagem de pior caso e sobre o domínio analisado, apoiando a compreensão dos critérios avaliados.

Em seguida, para cada CVE, a ferramenta apresenta a sugestão derivada do cenário de pior caso e uma sugestão produzida pela IA. A página de avaliação contém um formulário com perguntas de múltipla escolha baseadas nas definições de *Impacto em Segurança* e *Impacto em Missão* do SSVC [CERT Coordination Center 2024b]. As respostas são analisadas por uma LLM, que compara o contexto informado às definições oficiais do SSVC para estimar o valor de impacto humano, acompanhado de uma justificativa textual. O operador também pode definir diretamente o impacto humano a qualquer momento. O fluxo geral dessa etapa é ilustrado na Figura **??**.

Durante a avaliação dos CVEs, um *chatbot* contextualizado com informações sobre SSVC, o serviço analisado, o CVE e os critérios de avaliação permanece disponível para apoiar a decisão. As informações fornecidas durante as avaliações, incluindo respostas aos formulários, interações com o *chatbot*, justificativas e decisões registradas, são armazenadas e reutilizadas em análises futuras. A cada decisão registrada, os demais CVEs são reavaliados, fazendo com que evidências associadas a outros CVEs no mesmo serviço contribuam para recomendações progressivamente mais informadas. O operador, contudo, mantém o controle da decisão, podendo aceitar, ajustar ou substituir as sugestões fornecidas pelo sistema conforme o contexto disponível.

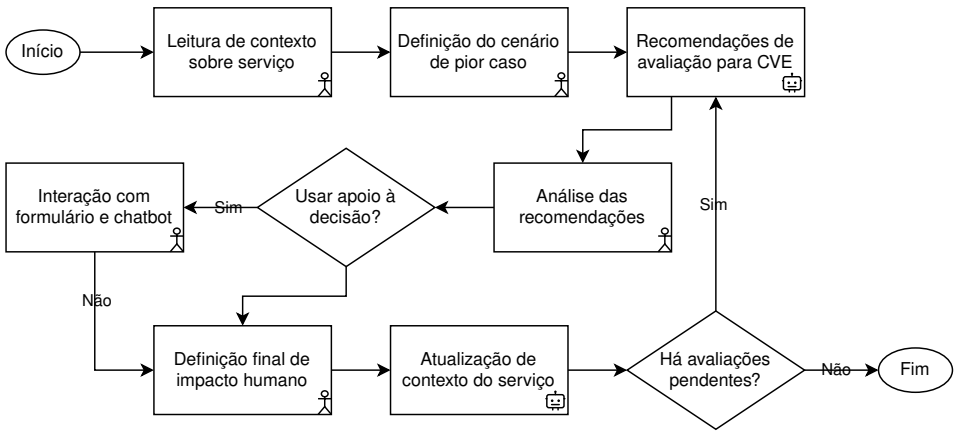


**Figura 2. Fluxo de avaliação do impacto humano no SSVC Copilot.**

## 6. Avaliação

Para priorizar vulnerabilidades de forma eficiente, os agentes devem fornecer respostas acuradas. No entanto, não há um gabarito definitivo para cada ponto de decisão dos CVEs. Assim, embora as informações do projeto *Vulnrichment* não sejam uma verdade absoluta, elas são adotadas como referência para medir a **taxa de concordância** entre os agentes de exploração e automação e a avaliação publicada pelo CISA para vulnerabilidades entre 2016 e 2026. Além disso, o CISA *Vulnrichment* não cobre todas as vulnerabilidades reportadas no NVD, conforme observado na Figura 3.

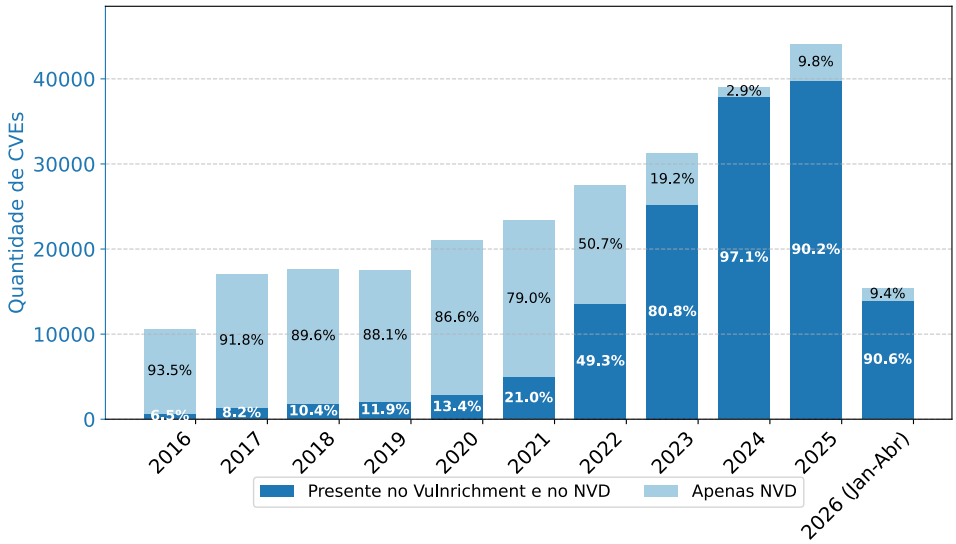


**Figura 3. Cobertura dos CVEs do CISA *Vulnrichment* em relação ao NVD por ano.**

Também avaliamos as recomendações de decisão do agente de impacto humano, comparando com as sugestões automatizadas, mas com menos contexto proveniente do "SSVC de Pior Caso", e com as decisões finais do humano.

### 6.1. Ponto de Decisão: Exploração

As Tabelas 2a e 2b apresentam matrizes de confusão entre o agente de exploração e o *CISA Vulnrichment* para 144.163 CVEs. A configuração final do agente (Tabela 2a) obteve uma

concordância de 65,31%, enquanto uma versão simplificada (Tabela 2b), sem regras baseadas em EPSS e CWE, e desconsiderando CVEs com evidências (*links*) indisponíveis, alcançou 97,18%. Esse resultado indica que a maior parte das divergências decorre de decisões metodológicas deliberadas, alinhadas às diretrizes do SSVC, conforme explicado a seguir.

**Tabela 2. Agente de exploração *vs CISA Vulnrichment*.**

| Agente \ CISA | Active | PoC | None |
|---|---|---|---|
| **Active** | 1389 | 563 | 480 |
| **PoC** | 0 | 29145 | 45176 |
| **None** | 2 | 3663 | 63378 |

**(a) Com regras de EPSS e CWE. Concordância: 65,31%.**

| Agente \ CISA | Active | PoC | None |
|---|---|---|---|
| **Active** | 1389 | 0 | 0 |
| **PoC** | 0 | 33042 | 0 |
| **None** | 2 | 3689 | 105674 |

**(b) Sem regras de EPSS e CWE, e sem CVEs com evidências indisponíveis. Concordância: 97,18%.**

A Tabela 2a mostra que a maior parte das divergências ocorre entre as classes *PoC* e *None*, principalmente nos casos em que o agente classifica a vulnerabilidade como *PoC* enquanto o *CISA Vulnrichment* a classifica como *None* – 45176 CVEs. Esse comportamento decorre de decisões metodológicas alinhadas às diretrizes do SSVC. O agente utiliza o EPSS como sinal complementar para ajustar o fator de exploração, permitindo que valores elevados elevem a classificação para *PoC* ou *Active*. Além disso, o agente considera como evidência de *PoC* não apenas bases tradicionais de *exploits*, mas também as CWEs, conforme recomendado pela documentação do SSVC [CERT Coordination Center 2024a].

Algumas divergências também são explicadas pela disponibilidade das evidências utilizadas no momento da análise. Em alguns casos, o *CISA Vulnrichment* considera referências externas para *exploits* que não estão mais disponíveis nos registros atualmente retornados pelo NVD. Assim, enquanto o agente realiza a classificação com base nas evidências acessíveis no momento da coleta, o *Vulnrichment* possivelmente reflete uma análise baseada em um histórico de referências. A Tabela 2b reflete o resultado do agente quando desconsidera i) informações do EPSS, ii) informações do CWE, e iii) CVEs com *links* indisponíveis para evidências, gerando uma taxa de concordância de 97,18%.

### 6.2. Ponto de Decisão: Automação

A Tabela 3 apresenta os resultados reportados pelo agente de automação e pelo CISA *Vulnrichment*. A taxa de concordância foi de 92,80%, correspondendo a 133.442 casos de concordância e 10.721 de divergência. Em termos agregados, o agente classificou 26,53% das amostras como *sim*, enquanto o *Vulnrichment* classificou 22,35% nessa mesma categoria. Essa proximidade indica alinhamento na distribuição global das classes, ainda que não capte, isoladamente, a concordância caso a caso.

**Tabela 3. Agente de automação *vs CISA Vulnrichment*. Concordância: 92,80%.**

| Agente \ CISA | Sim | Não |
|---|---|---|
| **Sim** | 30051 | 8191 |
| **Não** | 2163 | 103391 |

Observa-se que os erros estão relativamente equilibrados entre falsos positivos e falsos negativos, sem viés significativo para *sim* ou *não*. Esse comportamento sugere

que o agente não é sistematicamente mais permissivo nem mais conservador do que a referência, mantendo consistência com o critério adotado pelo CISA.

### 6.3. Ponto de Decisão: Impacto Humano

A avaliação do agente de impacto humano considera que a estimativa de impacto humano no SSVC envolve certo grau de subjetividade, pois depende do contexto do serviço, do impacto na missão organizacional, dos riscos à segurança de pessoas e das prioridades do operador. Assim, analisa-se em que medida as sugestões da IA se alinham às decisões dos operadores e se tornam mais aderentes ao seu perfil decisório ao longo do processo. Além disso, comparamos a decisão assistida pela IA à abordagem baseada no pior caso, verificando se há ganho real em introduzir um agente adaptativo nessa decisão.

#### 6.3.1. Metodologia da Avaliação

O experimento envolveu cinco participantes, que avaliaram 16 CVEs[2] relacionados a bombas de insulina. Esse domínio foi escolhido por sua relação direta com o exemplo da documentação do SSVC [CERT Coordination Center 2026], que apresenta uma análise envolvendo vulnerabilidades nesse tipo de dispositivo. Assim, o cenário experimental permite avaliar o agente de impacto humano em um contexto no qual a segurança do paciente, a continuidade do serviço e o impacto operacional são relevantes.

O fluxo de avaliação iniciou-se com a apresentação do contexto para que o participante compreendesse o serviço analisado. Em seguida, o participante preencheu o formulário inicial de pior caso, destinado à definição dos impactos máximos associados ao comprometimento de confidencialidade, integridade e disponibilidade da aplicação.

Após essa etapa inicial, os participantes seguiram para a triagem dos 16 CVEs. Para cada CVE, a ferramenta apresentava a sugestão de impacto humano obtida pela abordagem de pior caso e a sugestão produzida pela IA com base no contexto acumulado até aquele momento. O participante podia responder ao formulário de impacto humano, consultar o *chatbot*, revisar as justificativas e registrar sua decisão final de impacto humano.

A cada decisão registrada, o contexto disponível ao agente era atualizado e o CVE mais crítico pendente era reavaliado. Dessa forma, o experimento permitiu observar o alinhamento entre as sugestões da ferramenta e as decisões finais dos operadores, bem como a evolução desse alinhamento ao longo da avaliação.

#### 6.3.2. Resultados e Discussões

O operador pode categorizar o impacto humano de uma CVE como *baixo, médio, alto e muito alto*, sendo associados os valores numéricos *1, 2, 3, e 4*, respectivamente. A Tabela 4 apresenta, para cada participante e cada CVE, a diferença entre a sugestão exibida pela ferramenta e a decisão final do operador. Portanto, o valor 0 indica alinhamento total, enquanto valores maiores indicam divergência. A comparação entre os blocos da

[2]CVE-2016-5085, CVE-2016-5084, CVE-2016-5086, CVE-2018-10634, CVE-2019-10964, CVE-2020-10627, CVE-2020-27256, CVE-2020-27258, CVE-2020-27264, CVE-2020-27266, CVE-2020-27268, CVE-2020-27269, CVE-2020-27270, CVE-2020-27272, CVE-2020-27276 e CVE-2022-32537.

figura mostra que a sugestão da IA apresenta maior alinhamento na segunda metade da avaliação, após a análise dos primeiros CVEs. Esse comportamento está relacionado ao funcionamento incremental do agente, que atualiza seu contexto a cada decisão registrada e passa a incorporar evidências sobre o serviço e sobre o perfil decisório do participante.

**Tabela 4. Diferença entre a decisão dos participantes e as sugestões da IA e da abordagem de pior caso.**

| # | Sugestão da IA | | | | | Sugestão de Pior Caso | | | | |
|---|---|---|---|---|---|---|---|---|---|---|
| | P1 | P2 | P3 | P4 | P5 | P1 | P2 | P3 | P4 | P5 |
| 1 | 1 | 0 | 0 | 0 | 0 | 0 | 2 | 0 | 0 | 0 |
| 2 | 1 | 0 | 0 | 0 | 1 | 1 | 0 | 2 | 0 | 1 |
| 3 | 0 | 0 | 0 | 0 | 1 | 1 | 2 | 0 | 0 | 1 |
| 4 | 0 | 0 | 2 | 0 | 0 | 1 | 0 | 0 | 0 | 1 |
| 5 | 0 | 0 | 0 | 0 | 1 | 1 | 0 | 0 | 0 | 0 |
| 6 | 0 | 1 | 0 | 0 | 0 | 1 | 1 | 0 | 0 | 0 |
| 7 | 1 | 0 | 0 | 0 | 1 | 0 | 1 | 0 | 0 | 0 |
| 8 | 2 | 2 | 0 | 1 | 0 | 1 | 0 | 0 | 1 | 1 |
| 9 | 0 | 0 | 0 | 0 | 0 | 1 | 1 | 0 | 0 | 1 |
| 10 | 0 | 0 | 0 | 2 | 0 | 1 | 0 | 0 | 1 | 0 |
| 11 | 0 | 0 | 0 | 1 | 0 | 1 | 0 | 0 | 2 | 0 |
| 12 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 2 | 1 |
| 13 | 0 | 0 | 1 | 3 | 1 | 1 | 0 | 1 | 1 | 1 |
| 14 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 2 | 1 |
| 15 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 1 | 1 |
| 16 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 1 |

Enquanto o agente incorpora os aprendizados das avaliações para se aproximar das prioridades do operador, a abordagem de pior caso isolada permanece essencialmente estática. Como depende de uma interação inicial e não evolui com decisões, justificativas e interações posteriores, tende a manter um padrão de alinhamento semelhante ao longo dos CVEs avaliados. Isso evidencia sua limitação como estratégia isolada e reforça o papel do agente como complemento adaptativo à avaliação de impacto humano.

Os resultados agregados indicam uma melhora consistente no alinhamento entre as sugestões do agente de impacto humano e as decisões finais dos operadores. Considerando todos os participantes, a solução proposta alcançou 78,75% de concordância, passando de 70% na primeira metade dos CVEs para 87,5% na segunda metade. Essa evolução também aparece no erro médio, que foi de aproximadamente 0,28 no experimento completo, reduzindo de 0,375 para 0,2 entre as duas metades da avaliação. Esse comportamento indica que o agente passou a produzir recomendações progressivamente mais aderentes ao perfil decisório dos avaliadores.

Em comparação, a abordagem de pior caso obteve 52,5% de concordância e erro médio de 0,55. Além disso, 71% do contexto fornecido pelos operadores concentrou-se nas nove primeiras decisões, sugerindo que a necessidade de interação tende a diminuir após uma etapa inicial de calibração. Dessa forma, os resultados indicam que o agente complementa a abordagem de pior caso ao incorporar contexto acumulado, reduzir divergências ao longo do processo e apoiar decisões mais alinhadas ao contexto operacional.

Também foi analisado o custo total do agente de impacto humano ao longo da avaliação dos 16 CVEs, incluindo, em cada chamada à IA, o processamento do contexto inicial e daquele progressivamente acumulado por meio das interações e decisões anteriores. Com o modelo *Gemini 3 Flash Preview* e os preços de maio de 2026, o consumo

observado variou de 319.889 *tokens* (R$ 2,54) a 660.156 tokens (R$ 4,24). Como mostra a Tabela 5, no caso mais custoso, a maior parte do consumo decorreu do *chatbot*, e no caso de menor custo, ele se distribuiu entre o *chatbot* e os processamentos internos da ferramenta.

**Tabela 5. Consumo total de *tokens* e custo estimado na avaliação dos 16 CVEs.**

| Etapa | Menor consumo | | Maior consumo | |
|---|---|---|---|---|
| | *Tokens* | Custo | *Tokens* | Custo |
| Chatbot | 154.771 | R$ 1,45 | 428.541 | R$ 2,82 |
| Recomendações de avaliação | 165.118 | R$ 1,08 | 231.615 | R$ 1,42 |
| Total | 319.889 | R$ 2,54 | 660.156 | R$ 4,24 |

## 7. Conclusão

Este trabalho apresentou o SSVC Copilot, um assistente de priorização de vulnerabilidades baseado no arcabouço SSVC. A solução combina agentes especializados, fontes heterogêneas de inteligência de ameaças e interação humano–IA para apoiar a definição dos pontos de decisão utilizados na priorização de CVEs. Diferentemente de abordagens baseadas apenas em pontuações ou limiares fixos, o SSVC Copilot busca incorporar evidências técnicas e contexto operacional ao processo decisório, preservando a supervisão humana nos pontos em que essa avaliação é necessária.

Os resultados indicam que os agentes automatizados são capazes de inferir pontos de decisão do SSVC de forma escalável e com elevada proximidade em relação ao CISA *Vulnrichment*. Para automação, o agente alcançou 92,80% de concordância. No caso de exploração, a concordância poderia chegar a 97,18% ao desconsiderar divergências associadas à indisponibilidade de referências históricas de *exploits*. No entanto, a configuração final adotou deliberadamente regras adicionais baseadas em EPSS e CWE, pois elas estão alinhadas às diretrizes do SSVC e foram consideradas metodologicamente mais adequadas. Assim, parte das divergências observadas representa uma escolha de projeto voltada a produzir decisões mais completas e aderentes ao arcabouço adotado.

A avaliação do ponto de decisão de impacto humano reforça a importância de uma abordagem *human-in-the-loop*. Embora a estratégia baseada em pior caso seja útil como referência inicial, ela permanece estática após a calibração feita pelo operador. Por outro lado, o agente de impacto humano incorpora progressivamente o contexto das avaliações, tornando suas sugestões mais alinhadas aos critérios e prioridades do operador ao longo do processo. Esse comportamento evidencia a capacidade do SSVC Copilot de apoiar decisões futuras e reduzir a carga cognitiva associada à priorização de vulnerabilidades.

Ainda assim, a qualidade das recomendações depende das avaliações fornecidas pelo operador, de modo que a responsabilidade final permanece sob controle humano. Trabalhos futuros poderiam envolver a sofisticação via habilidades agênticas, ampliar o número de participantes na avaliação, explorar outros domínios de aplicação e medir de forma mais direta a redução do esforço humano obtida pelo uso contínuo do agente.

## 8. Uso de IA Generativa

Utilizamos as ferramentas Grammarly e ChatGPT na revisão textual, e Claude Code e OpenAI Codex no desenvolvimento dos agentes. As contribuições geradas foram revisadas e aprovadas pelos autores, que assumem total responsabilidade pelo conteúdo final.

## Referências

Aqua Security (2024). Trivy: Comprehensive vulnerability scanner for containers and other artifacts. https://github.com/aquasecurity/trivy. Accessed: 2026-04-13.

CERT Coordination Center (2024a). SSVC Decision Point: Exploitation. https://certcc.github.io/SSVC/reference/decision_points/exploitation. Accessed: 2026-05-08.

CERT Coordination Center (2024b). SSVC Decision Point: Human Impact. https://certcc.github.io/SSVC/reference/decision_points/human_impact/. Accessed: 2026-05-08.

CERT Coordination Center (2025). EPSS probability as input to exploitation. SSVC: Stakeholder-Specific Vulnerability Categorization. Acesso em: 31 jul. 2026.

CERT Coordination Center (2026). Worked Example. https://certcc.github.io/SSVC/topics/worked_example/. SSVC: Stakeholder-Specific Vulnerability Categorization. Acesso em: 21 maio 2026.

CISA (2023). Stakeholder-specific vulnerability categorization (ssvc). https://www.cisa.gov/ssvc. Accessed: 2026-04-13.

CISA) (2024). Known Exploited Vulnerabilities Catalog. Accessed: 2026-04-15.

CISA (2026). Vulnrichment project. https://github.com/cisagov/vulnrichment. Accessed: 2026-04-29.

Cohen, D., Te'eni, D., Yahav, I., Zagalsky, A., Schwartz, D., Silverman, G., Mann, Y., Elalouf, A., and Makowski, J. (2025). Human–ai enhancement of cyber threat intelligence. *International Journal of Information Security*, 24(2):99.

FIRST (2024). Forum of incident response and security teams. common vulnerability scoring system (cvss). https://www.first.org/cvss/. Accessed: 2026-04-13.

Foundjem, A., Nganyewou Tidjon, L., Da Silva, L., and Khomh, F. (2026). Multi-agent ai framework for threat mitigation and resilience in machine learning systems. *ACM Trans. Softw. Eng. Methodol.* Just Accepted.

Gustavsson, P. M., Kollberg, M., Nadjafi, M., Wiktorin, J., and Persson, V. (2025). The cyber due diligence object model (cddom): A structured approach to cybersecurity risk, interoperability, and regulatory alignment. *SSRN Electronic Journal*. Preprint, not peer reviewed.

Jacobs, J., Romanosky, S., Edwards, B., Adjerid, I., and Roytman, M. (2021). Exploit prediction scoring system (epss). *Digital Threats*, 2(3).

Janloy, K., Wuttidittachotti, P., Wannapiroon, P., and Kaewduangta, P. (2025). Agentic ai-driven and human-centered cyber immune platforms for cyber threat intelligence verification under pdpa regulations. In *2025 5th International Conference on Educational Communications and Technology (ICTAECT)*, pages 1–7.

Kshetri, N. and Voas, J. (2025). Agentic artificial intelligence for cyber threat management. *Computer*, 58(5):86–90.

Liu, X., Liang, J., Yan, Q., Jang, J., Mao, S., Ye, M., Jia, J., and Xi, Z. (2025). Cylens: Towards reinventing cyber threat intelligence in the paradigm of agentic large language models. *arXiv preprint arXiv:2502.20791*.

Mavroeidis, V., Hohimer, R., Casey, T., and Jesang, A. (2021). Threat actor type inference and characterization within cyber threat intelligence. In *2021 13th International Conference on Cyber Conflict (CyCon)*, pages 327–352.

MITRE Corporation (2026). Common weakness enumeration (cwe). `https://cwe.mitre.org/`. Accessed: 2026-04-29.

NIST (2021). CVE-2021-44228 Detail. `https://nvd.nist.gov/vuln/detail/CVE-2021-44228`. Accessed: 2026-04-15.

NIST (2024). National Institute of Standards and Technology. National Vulnerability Database (NVD). Accessed: 2026-04-15.

Offensive Security (2024). Exploit Database. Accessed: 2026-04-15.

Rapid7 (2024). Metasploit Framework. Accessed: 2026-04-15.

Saccone, F., Manzi, A., Di Sorbo, A., Costante, E., and Visaggio, C. A. (2025). Design and implementation of a multi-agent threat intelligence assistant based on generative ai. In *2025 IEEE International Conference on Big Data (BigData)*, pages 6883–6888.

Sadlek, L. (2020). Threat management based on information about vulnerabilities. Master's thesis, Faculty of Informatics - Masaryk University, Brno, Czech Republic.

Snyk Ltd. (2024). Snyk: Developer security platform. `https://snyk.io`. Accessed: 2026-04-13.

Spyros, A., Koritsas, I., Papoutsis, A., Panagiotou, P., Chatzakou, D., Kavallieros, D., Tsikrika, T., Vrochidis, S., and Kompatsiaris, I. (2025). Ai-based holistic framework for cyber threat intelligence management. *IEEE Access*, 13:20820–20846.

Tsuji, D., Ueki, Y., Kawaguchi, N., Kakuta, T., and Nakakoji, H. (2026). Automating ssvc: A worst-case scenario approach for evaluating human impact of vulnerabilities. *IEICE Communications Express*, pages 1–4.