# CiberVIZ - Ferramenta de Visualização e Deteção de Intrusões no Ciberespaço

**P. Marques (1) (2), L. Dias (1) (2) e M. Correia (2)**

(1) CINAMIL, Academia Militar – Lisboa, Portugal, marques.pmb@exercito.pt, dias.lfxcm@exercito.pt

(2) INESC-ID, Instituto Superior Técnico – Lisboa, Portugal, miguel.p.correia@tecnico.ulisboa.pt

**Resumo:** Este artigo apresenta a Ferramenta de Visualização e Deteção de Intrusões (CiberVIZ), uma ferramenta que fornece algoritmos de deteção de intrusão com um GUI eficaz. A ferramenta tem dois objetivos. Primeiro, visa ajudar o analista humano a observar as ameaças em curso, permitindo responder a estes ataques. Em segundo lugar, destina-se a ajudar o analista a estudar o desempenho dos algoritmos com parâmetros diferentes, a fim de decidir qual a combinação que melhor se adequa aos seus objetivos. A ferramenta integra um conjunto de algoritmos de deteção de intrusão recentemente propostos com base na aprendizagem automática não supervisionada. A CiberVIZ adota uma arquitetura de sistema cliente-servidor e implementa uma API REST que medeia a comunicação entre o servidor e o GUI no lado do cliente. No GUI, foi escolhido um aspeto visual claro e minimalista, contendo funcionalidades que permitem analisar os dados da rede de uma forma simples e clara.

**Palavras-chave:** Ciber-segurança, Aprendizagem Automática, Sistema de Deteção de Intrusões, Análise de Segurança, Ferramenta de Visualização.

## 1. INTRODUÇÃO

A quantidade significativa de informação digital gerada no mundo continua a crescer, ainda mais neste momento em que o trabalho e a educação à distância se tornaram parte da realidade de muitas pessoas (Hakak *et al.,* 2020). Este crescimento no campo da Tecnologia da Informação (TI) combinado com um ambiente de riscos e ameaças em constante mudança, torna a segurança cibernética essencial para todos nós. Portanto, para proteger indivíduos e organizações de forma mais eficaz, é necessário identificar as ameaças (Bowen *et al.*, 2007).

Com a evolução do campo da Aprendizagem Automática (ML do inglês *Machine Learning*), têm crescido o número de Sistemas de Detecção de Intrusão (IDSs do inglês *Intrusion Detection Systems*) que utilizam algoritmos de deteção de anomalias, ou de modo genérico ML não supervisionado (Buczak e Guven, 2015, Bhuyan *et al.,* 2014, Nascimento *et al.,* 2011). Os IDSs que utilizam tais algoritmos não precisam de ter as assinaturas dos ataques para os detetar, o que é uma vantagem na capacidade de detetar novos ataques em comparação com IDSs baseados no conhecimento (ou baseados em assinaturas).

Os IDSs podem ser melhorados com ferramentas de visualização para uma verificação mais rápida e eficaz dos resultados obtidos pelos algoritmos que utilizam (Staheli *et al.,* 2014, Musman e Temin, 2015). As ferramentas de visualização podem tornar-se uma componente chave do IDSs, permitindo ao analista visualizar graficamente a informação extraída do enorme volume de dados de monitorização que estão a ser recolhidos e produzidos.

A CiberVIZ é uma ferramenta de código fonte aberto (*open source*), desenvolvida pelos autores. Pretende, em primeiro lugar, ajudar o analista humano a observar as ameaças em curso, num primeiro passo para responder a estes ataques. Visa ser uma ferramenta de visualização gráfica para a deteção automática de ameaças no ciberespaço, intuitiva e fácil de usar. Em segundo lugar, visa ajudar o analista a estudar o desempenho dos algoritmos de deteção com diferentes parâmetros, a fim de decidir qual a combinação que melhor se adequa aos seus objetivos. Para este fim, fornece ao utilizador uma abstração dos processos de análise de eventos, permitindo a otimização da utilização dos algoritmos aplicados para a deteção de intrusão.

## 2. ALGORITMOS

A CiberVIZ integra 3 algoritmos de deteção de intrusões na rede que utilizam técnicas baseadas em aprendizagem não supervisionada. As entidades (utilizadores ou máquinas) que se constituem ameaças potenciais, são agrupadas de forma destacada das restantes. O comportamento anómalo tendencialmente fica em agrupamentos isolados facilmente identificados. Os algoritmos integrados são: i) *FlowHacker*; ii) *OutGene* e iii) *DynIDS.* Estes estão apresentados sucintamente nos parágrafos abaixo.

O *FlowHacker* utiliza 28 características divididas em dois grupos. Metade das características está relacionada com o computador de origem, e a outra metade é a mesma, mas relativa ao computador de destino. Oito das 14 características consideram o fluxo bidirecional (de e para os portos) de 4 protocolos da camada de aplicação, que são: HTTP (porto 80), IRC (porto 194), SMTP (porto 25), SSH (porto 22), e IRC (porto 6667). As restantes

características são: número de ligações efetuadas, número de portos utilizados pela fonte, número de portos contactados pela fonte, a soma dos bytes enviados pela fonte, e a soma dos pacotes enviados pela fonte (Sacramento *et al.*, 2018).

O *OutGene* tem várias semelhanças com *FlowHacker* mas introduz dois novos conceitos: i) *zoom genético*, utilizando um algoritmo genético que identifica o melhor subconjunto de características que conduzem à formação do mesmo resultado de agrupamento que todas as características gerariam; e ii) *distensão temporal*, para detetar ataques furtivos que utilizam um baixo ritmo de execução, analisando o fluxo de eventos em diferentes janelas de tempo em diferentes escalas de tempo. O *OutGene* remove as 2 características associadas à porta 6667 (IRC), uma vez que é menos utilizada hoje em dia (Dias *et al.*, 2019).

O *DynIDS* tem como principal inovação, a seleção de parte das características com base no fluxo de tráfego. Este algoritmo utiliza 12 características estáticas, metade relativas à fonte e metade relativas ao destino. Contudo, também utiliza características baseadas em portas obtidas dinamicamente, de acordo com os dados analisados em cada janela de tempo. O *DynIDS* utiliza 4 características para cada porta (número de pacotes enviados e recebidos numa porta, fonte e hospedeiro de destino). As características dinamicamente definidas são identificadas com base no algoritmo *DYN3_X* que filtra à razão x/3 as portas que aparecem em mais fluxos, as portas que aparecem em menos fluxos e as portas utilizadas por menos máquinas. Esta abordagem não limita a capacidade do sistema de detetar ataques relacionados com portas específicas, porque as características estão correlacionadas com o fluxo de tráfego na rede. Em relação ao agrupamento, aplica 3 algoritmos de agrupamento para obter um melhor desempenho na identificação de outliers[1], o que é feito através da interceção dos resultados dos três algoritmos (Dias *et al.*, 2020).

## 3. CiberVIZ

A CiberVIZ é baseada numa arquitetura cliente-servidor utilizando um Interface de Programação de Aplicação (API do inglês *Application Programming Interface*) do tipo Transferência de Estado Representacional (REST do inglês *REpresentational State Transfer*). Escolhemos esta abordagem para remover o armazenamento de dados e o processamento complexo do lado do utilizador. Os algoritmos e a base de dados estão do lado do servidor (*back-end*) com mais poder computacional para fazer o processamento e com mais capacidade de armazenamento. Utilizamos REST porque é um mecanismo simples de pedido/resposta. As suas chamadas são baseadas em mensagens e seguem o padrão HTTP. Integramos os algoritmos de deteção de intrusão de rede no *back-end* do nosso sistema e as comunicações são feitas através de pedidos HTTP(S). Os resultados obtidos são enviados para o cliente e são exibidos no Interface Gráfico do Utilizador (GUI do inglês *Graphical User Interface)*, que é local.

O GUI oferece uma camada de abstração que permite a qualquer tipo de utilizador utilizar a CiberVIZ. O utilizador pode visualizar e analisar os dados mais eficazmente à medida que estes são apresentados graficamente. É claramente mais simples do que a abordagem comum de executar scripts através do terminal de comando. Foram acrescentadas características ao sistema para obter dados relevantes para o utilizador e para proporcionar flexibilidade na análise dos dados.

Quando um ficheiro (com dados de tráfego) é carregado na aplicação (no servidor), várias análises padrão são realizadas imediatamente, a fim de acelerar os pedidos que o analista pode fazer. Nesta fase, o utilizador pode também definir novas análises a serem feitas imediatamente. A Fig. 1 mostra o ponto de vista neste momento. Está dividida em 5 partes correspondentes à sua função: 1) Mostra o nome do ficheiro selecionado e um gráfico com o número de eventos no ficheiro. O eixo-x é o registo temporal dos eventos. 2) Apresenta um menu e uma caixa de texto de entrada para escolher o intervalo de tempo para a análise. 3) Uma caixa de texto de entrada permite ao utilizador escolher a janela de tempo para as análises (ou várias, separadas por vírgula). Tem também botões de opção para selecionar o método que será aplicado para extrair as características. Os métodos atualmente suportados correspondem às características de cada um dos 3 algoritmos suportados, à combinação de características de *DynIDS* e *OutGene*, e a uma seleção livre (o utilizador escolhe a lista de portos a analisar). 4) Tem um botão de retorno para a vista inicial. 5) Apresenta três botões que permitem: iniciar a análise, visualizar o gráfico de eventos com mais detalhe, e visualizar os resultados obtidos.


*Fig. 1. Vista de análise do CiberVIZ.*

Relativamente à visualização dos resultados, a Fig. 2 a primeira imagem que o utilizador tem. No ponto 1

[1] Elemento que está isolado num agrupamento (ameaça).

está o nome do ficheiro. No ponto 2 está a caixa de entrada para escolher o número máximo de elementos no mesmo agrupamento. Para encontrar *outliers*, o valor por defeito tem de ser deixado como 1. O utilizador pode também selecionar a vista IPs numa rede mostrada no menu. Esta vista pode ser: interna (Int), apenas IPs pertencentes à rede; ou externa (Ext), apenas IPs fora da rede. O ponto 3 mostra uma lista dos parâmetros de processamento já efetuado. Com um duplo clique numa linha da lista, o GUI mostra a respetiva vista de resultados. Depois da análise do ficheiro, a ferramenta apresenta uma lista de resultados obtidos, na Fig. 3. Esta lista (ponto 4) inclui o registo temporal dos agrupamentos, o número do agrupamento, o número de máquinas desse agrupamento, e os IPs dessas máquinas. O Ponto 5 está associado a três funcionalidades: i) exibir o mapa de calor dos dados obtidos; ii) guardar em formato .csv o resultado dos agrupamentos; e iii) apresentar as métricas correspondentes ao desempenho da análise efetuada.

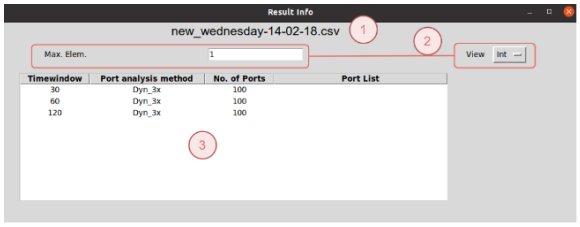


*Fig. 2. Primeira vista dos resultados do CiberVIZ.*

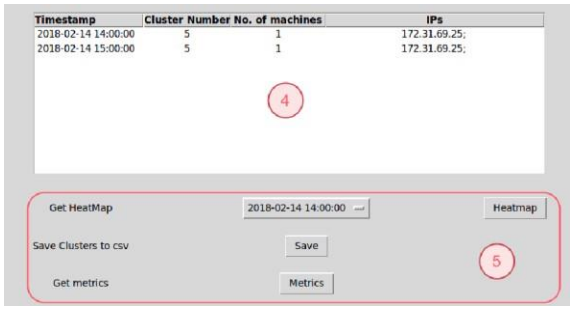


*Fig. 3. Segunda vista dos resultados do CiberVIZ.*

## 4. AVALIAÇÃO E RESULTADOS

Na primeira parte da avaliação usaram-se as métricas de avaliação de sistemas e software apresentadas na norma ISO/IEC 25010:2011. A segunda parte compara os resultados obtidos pela CiberVIZ com os resultados descritos nos artigos originais dos algoritmos implementados na ferramenta.

### *4.1. Avaliação - ISO/IEC 25010:2011*

*Capacidade funcional*: Realização dos objetivos através das funções da ferramenta. Avalia se os resultados apresentados estão coerentes comparando os 3 algoritmos integrados na ferramenta. A CiberVIZ cumpre o seu principal objetivo de detetar comportamentos anómalos e de classificar as máquinas como potenciais ameaças ou vítimas. A CiberVIZ permite ao utilizador escolher vários métodos de análise diferentes e oferece a capacidade de analisar a atividade da rede ao longo de vários períodos de tempo.

*Compatibilidade*: Capacidade da ferramenta de funcionar corretamente, independentemente do hardware ou software utilizado. Este aspeto também considera a interoperabilidade e o grau em que dois ou mais sistemas, produtos ou componentes podem trocar informações e utilizar a informação que foi trocada. O GUI da CiberVIZ é compatível tanto com ambientes Windows como Linux. É fornecido um contentor Docker, do *back-end*, com o código fonte, pacotes e bibliotecas, para que este funcione independentemente do sistema operativo utilizado. Quanto ao formato dos dados de entrada, estes estão limitados ao formato .csv e à estrutura das colunas com cabeçalhos fixos. No entanto, este formato é simples e fácil de introduzir os dados.

*Usabilidade*: Grau de eficácia, eficiência e satisfação obtido pelos utilizadores ao utilizar a ferramenta. A dificuldade de utilização, a capacidade de configurar e operar, a proteção contra erros do utilizador, e se os resultados são mostrados são importantes e devidamente organizados devem ser considerados. O GUI é simples e contém apenas as funções necessárias para uma análise eficiente dos dados. Os resultados obtidos são organizados e apresentados em tabelas e gráficos. O GUI tem um botão de ajuda que permite ao utilizador esclarecer dúvidas sobre a utilização da ferramenta, protegendo o sistema contra erros do utilizador.

*Segurança*: Como os dados (utilizados e armazenados) e todo o sistema são protegidos e a fiabilidade do sistema. A CiberVIZ está pronta a utilizar os pedidos HTTPS para obter mais segurança durante a comunicação. A segurança dos dados armazenados está diretamente relacionada com a segurança do servidor onde o *back-end* está a correr. Quanto à fiabilidade, numa pasta do GUI, existe um ficheiro de configurações que não pode ser corrompido. Caso contrário, o GUI deixará de funcionar.

### *4.2. Resultados*

Utilizamos o conjunto de dados do CSE-CIC-IDS2018, que foi criado para testar e avaliar IDSs (CIC, 2018). Este conjunto de dado representa a rede de uma empresa média, com seis sub-redes implantadas. Consideramos os cenários de ataque no conjunto de dados, que são: ataques de força bruta, ataques de negação de serviço, ataques da Web, ataques de infiltração e *scans* de IPs/portos (serviços).

Para analisar o desempenho dos algoritmos, consideramos as entidades detetadas como *outliers*. De acordo com os ataques e a sua duração apresentada no conjunto de dados, consideramos: Verdadeiros Positivos (VPs), as entidades que foram corretamente classificadas; Verdadeiros Negativos (VNs), as entidades que correctamente não foram classificadas como outliers; Falsos Negativos (FNs),

as entidades que deveriam ter sido classificadas como outliers mas não foram; e Falsos Positivos (FPs), as entidades que foram classificadas como outliers mas não são ameaças. As métricas que privilegiamos para a avaliação foram o F1-Score e o *Matthews correlation coefficient* (MCC), porque fornecem um resumo do desempenho dos algoritmos.

As figuras 4 e 5 mostram o $F_1$-Score e MCC dos algoritmos de deteção implementados no CiberVIZ. Estes foram obtidos através da análise do conjunto de dados do CSE-CIC-IDS2018 em várias janelas de tempo. Os valores apresentados nos gráficos foram calculados relativamente aos valores totais de VPs, VNs, FNs e FPs.

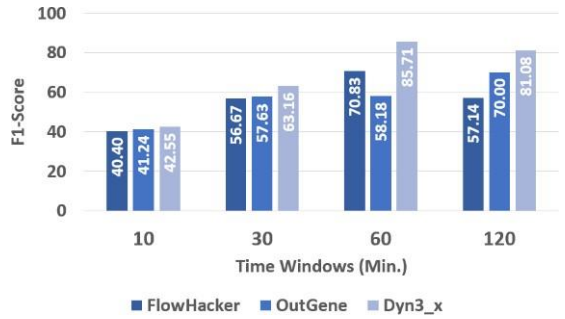


*Fig. 4. $F_1$-Score dos 3 algoritmos para o CSE-CIC-IDS2018.*

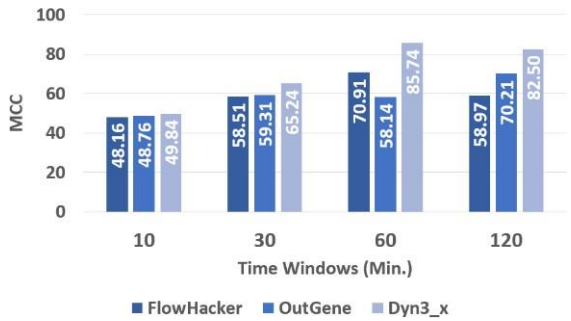


*Fig. 5. MCC dos 3 algoritmos para o CSE-CIC-IDS2018.*

## 5. CONCLUSÕES

A CiberVIZ é uma ferramenta de visualização para a deteção de intrusões, baseada numa arquitetura cliente-servidor e contendo um GUI interativo e simples. No *back-end* estão os algoritmos baseados em ML não supervisionado, responsáveis pela obtenção dos dados para o GUI e a REST API, que permitem a comunicação de dados entre o *back-end* e o *front-end* através de pedidos HTTP/HTTPS. O GUI tem botões, listas, gráficos e menus, que permitem a sua utilização por qualquer utilizador sem conhecimento prévio. O utilizador pode facilmente alterar os hiper-parametros dos algoritmos, obtendo resultados personalizados para diferentes tipos de análise.

## REFERÊNCIAS

**Artigos em revistas e conferências:**

M. H. Bhuyan, D. K. Bhattacharyya e J. K. Kalita (2014). Network Anomaly Detection: Methods, Systems and Tools. *IEEE Communications Surveys Tutorials*, 16(1): 303-336.

P. Bowen, J. Hash e M. Wilson (2007). Information security handbook: a guide for managers. *National Institute of Standards and Technology special publication*, 800-100.

A. L. Buczak e E. Guven (2015). A survey of data mining and machine learning methods for cyber security intrusion detection. *IEEE Communications Surveys & Tutorials*, 18(2):1153–1176.

L. Dias, H. Reia, R, Neves e M. Correia (2019). OutGene: Detecting Undefined Network Attacks with Time Stretching and Genetic Zooms. *International Conference on Network and System Security*, 199–220.

L. Dias, S. Valente e M. Correia (2020). Go With the Flow: Clustering Dynamically-Defined NetFlow Features for Network Intrusion Detection with DynIDS. *19th IEEE International Symposium on Network Computing and Applications.*

J. Estdale e E. Georgiadou (2018). Applying the ISO/IEC 25010 quality models to software product. *European Conference on Software Process Improvement*, 492- 503.

S. Hakak, W. Z. Khan e M. Imran, K.-K R. Choo e M. Shoaib (2020). Have you been a victim of COVID-19-related cyber incidents? Survey, taxonomy, and mitigation strategies. *IEEE Access*, 8: 124134–124144.

S. Musman e A. Temin (2015). A Cyber Mission Impact assessment tool. *IEEE International Symposium on Technologies for Homeland Security*, 1–7.

L. Sacramento, I. Medeiros, J. Bota, e M. Correia (2018). FlowHacker: Detecting Unknown Network Attacks in Big Traffic Data using Network Flows. *17th IEEE International Conference on Trust, Security and Privacy in Computing and Communications*, 567–572.

D. Staheli, T. Yu, R. J. Crouser, S. Damodaran, K. Nam, D, O'Gwynn, S. McKenna e L. Harrison (2014). Visualization evaluation for cyber security: Trends and future directions. *Proceedings of the Eleventh Workshop on Visualization for Cyber Security*, 49–56.

**Disponíveis online:**

Canadian Institute for Cybersecurity (2018). CSE-CIC-IDS2018 on AWS. Acesso em: 10-jul-2020. https://www.unb.ca/cic/datasets/ids-2018.html