Sistema de monitorização inteligente

Inteligent monitoring system

Luís António Pinto de Barros
Escola Superior de Tecnologia e Gestão
Politécnico da Guarda
Guarda, Portugal
Luisantoniio1998@gmail.com

António Mário Ribeiro Martins
Escola Superior de Tecnologia e Gestão
Politécnico da Guarda
Guarda, Portugal
amrmartins@ipg.pt

Resumo — **Durante o meu estágio na empresa Securnet, desenvolveu-se um projeto que envolveu a criação de um laboratório de rede com uma DMZ, onde se alojou servidores SFTP, de e-mail e web. O objetivo principal foi estabelecer um ambiente controlado para a monitorização e otimização dos recursos da rede.**

Inicialmente, configurou-se o laboratório de rede, criando uma rede com DMZ e procedendo à instalação e configuração dos servidores necessários.

Em seguida, implementei o sistema de monitorização Zabbix, que possibilitou a supervisão constante dos recursos da rede, incluindo a deteção de eventos e problemas. Além disso, configurei os *hosts* para garantir a monitorização eficaz de todas as máquinas do laboratório.

Realizou-se a integração entre o Zabbix e o sistema de gestão de ativos GLPi, proporcionando uma visão mais abrangente e organizada das informações relacionadas com os dispositivos monitorizados.

Estabeleceu-se também a integração entre o Zabbix e o Grafana, o que permitiu a criação de painéis de monitorização personalizados e a análise visual dos dados recolhidos. Esta integração melhorou significativamente a apresentação dos resultados da monitorização.

Para facilitar o acesso às ferramentas de monitorização, desenvolvi uma aplicação móvel e web em React Native, centralizando assim a sua utilização e simplificando o acesso remoto às informações da rede.

Por fim, implementaram-se técnicas de machine learning nos dados recolhidos pelo Zabbix, com o propósito de reduzir a ocorrência de eventos irrelevantes.

Palavras Chave – Engenharia de rede; Zabbix; GLPi; Grafana; React Native.; Machine Learning

Abstract — **During my internship at Securnet, I developed a project that involved creating a network laboratory with a DMZ, where I hosted SFTP, email, and web servers. The primary objective was to establish a controlled environment for monitoring and optimizing network resources.**

Initially, I configured the network laboratory by creating a network with a DMZ and proceeding with the installation and setup of the necessary servers.

Next, I implemented the Zabbix monitoring system, which allowed continuous supervision of network resources, including event and issue detection. Additionally, I configured the hosts to ensure effective monitoring of all machines in the laboratory.

I integrated Zabbix with the GLPi asset management system, providing a more comprehensive and organized view of information related to monitored devices.

I also established integration between Zabbix and Grafana, enabling the creation of customized monitoring dashboards and visual analysis of collected data. This integration significantly improved the presentation of monitoring results.

To simplify access to monitoring tools, I developed a mobile and web application using React Native, centralizing their use, and simplifying remote access to network information.

Finally, I implemented machine learning techniques on the data collected by Zabbix with the aim of reducing the occurrence of irrelevant events.

Keywords - Network engineering; Zabbix; GLPi; Grafana; React Native; Machine Learning

I. MOTIVAÇÃO

A motivação essencial deste projeto derivou da necessidade crescente de garantir uma monitorização de rede, sobretudo num cenário tecnológico em que as ameaças cibernéticas estão em constante evolução. A criação de um laboratório controlado com DMZ foi fundamental para simular situações reais, permitindo avaliar a resposta e eficácia das ferramentas em uso. A integração do Zabbix com outras ferramentas, como o GLPi e o Grafana visa não só a eficiência operacional, mas também a clareza na análise de apresentação de dados. Além disso, a inclusão de técnicas de machine learning reflete a busca pela vanguarda na filtragem de eventos, reduzindo ruídos e maximizando a relevância da informação captada. Em suma, a motivação foi criar uma infraestrutura robusta que, além de garantir a segurança, permitisse inovações no campo da cibersegurança.

II. OBJETIVOS

Os objetivos deste projeto estão centrados na criação de um ambiente de monitorização robusto e eficaz, com ênfase na integração de múltiplas ferramentas e técnicas inovadoras. Para atingir tal meta, foram estabelecidos os seguintes pontos:

- Configurar e testar o laboratório de rede com DMZ e servidores associados;
- Implementar e otimizar o sistema de monitorização Zabbix para supervisão contínua da rede;
- Integrar o Zabbix com o sistema de gestão de ativos GLPi para uma visão consolidada dos dispositivos;

- Estabelecer conexão entre o Zabbix e o Grafana para desenvolvimento de painéis de monitorização personalizados e análise visual avançada;
- Desenvolver uma aplicação móvel e web em React Native para centralizar e simplificar o acesso á monitorização de rede;
- Implementar técnicas de *machine learning* nos dados do Zabbix para filtrar e minimizar eventos não pertinentes;
- Testar o sistema na sua totalidade para garantir a eficácia e a eficiência das implementações;
- Documentar todas as etapas e configurações para permitir futuras expansões ou modificações do sistema

III. ESTADO DE ARTE

A monitorização e otimização de redes são imperativos na era digital, onde a segurança e eficiência são cruciais. Compreender a paisagem tecnológica é fundamental para selecionar as melhores ferramentas para atender às demandas do projeto.

Inicialmente foram analisadas as soluções de monitorização de redes existentes, conforme a Tabela 1.

Plataforma / **Caracteristicas**	Zabbix	Nagios	PRTG	SolarWinds	WireShark	Splunk	Graylog
Monitorização em tempo real	✓		✓	✓	✓		
Integração	✓	✓	✓			✓	✓
Alertas configuráveis	✓	✓					
Web interface	✓	✓	✓	✓		✓	✓
Monitorização de aplicações	✓			✓			
Análise de pacotes					✓		
Centralização de logs				✓		✓	✓

Tabela 1- Comparação de funcionalidades por software

A. Soluções de monitorização existentes

Das várias opções Zabbix, Nagios e PRTG foram escolhidos para uma análise mais aprofundada. O Zabbix, por ser o foco principal neste projeto, enquanto Nagios e PRTG pela sua ampla adoção no mercado.

B. Análise Crítica

O Zabbix é versátil e extensível, uma solução *end-to-end* para monitorização de redes. O Nagios, embora robusto pode necessitar de mais configurações manuais. PRTG por sua vez destaca-se pela interface amigável e eficiência em cenários variados.
Já o Zabbix oferece integrações interessantes com outras ferramentas, como o GLPi e Grafana, embora estas sejam de menor ênfase no contexto deste projeto.
A seguir, será analisado as soluções de *machine learning* que podem ser aplicadas aos dados provenientes do Zabbix, conforme a Tabela 2.

Bibliotecas / **Caractertísticas**	TensorFlow	Scikit-learn	Keras	Pytorch
Open source	✓	✓	✓	✓
Suporte para redes neuronais	✓		✓	✓
Flexibilidade e modularidade	✓	✓	✓	
Algoritmos predefinidos		✓		
Computação auto diferenciável	✓			✓

Tabela 2 - Comparação de características por bibliotecas Python

C. Bibliotecas de machine learning existentes

TensorFlow e Scikit-learn foram as bibliotecas em foco, graças á sua popularidade e eficácia na análise de padrões.

D. Análise Crítica

O TensorFlow com a sua biblioteca abrangente, é uma solução robusta para análises complexas, embora possa ser mais desafiador. Scikit-learn, é ideal para tarefas mais diretas para *machine learning.*
Ao selecionar tecnologias, a adaptabilidade, integração e eficácia em cenários práticos foram os principais critérios considerados.

IV. TOPOLOGIA DE REDE

Neste trabalho, configurou-se uma rede simulando a complexidade encontrada em infraestruturas empresariais reais, englobando não só a instalação de dispositivos, mas também a construção de um ecossistema digital análogo ao usado por organizações modernas.

A. Configuração da rede

O laboratório envolveu a implementação de servidores essenciais: e-mail, web e SFTP. Além disso, foi dada atenção à instalação de uma *firewall* e de um *switch*, componentes cruciais para a conexão e segurança na infraestrutura.

B. Implementação da DMZ

Dentro das medidas de segurança, enfatizou-se a construção de uma DMZ. Esta camada serve como uma barreira, colocando recursos como servidores web e e-mail em uma área intermediária entre a rede interna e externa, blindando assim ativos sensíveis de ameaças externas.

A arquitetura de rede implementada pode ser visualizada na Figura subsequente, Figura 1.

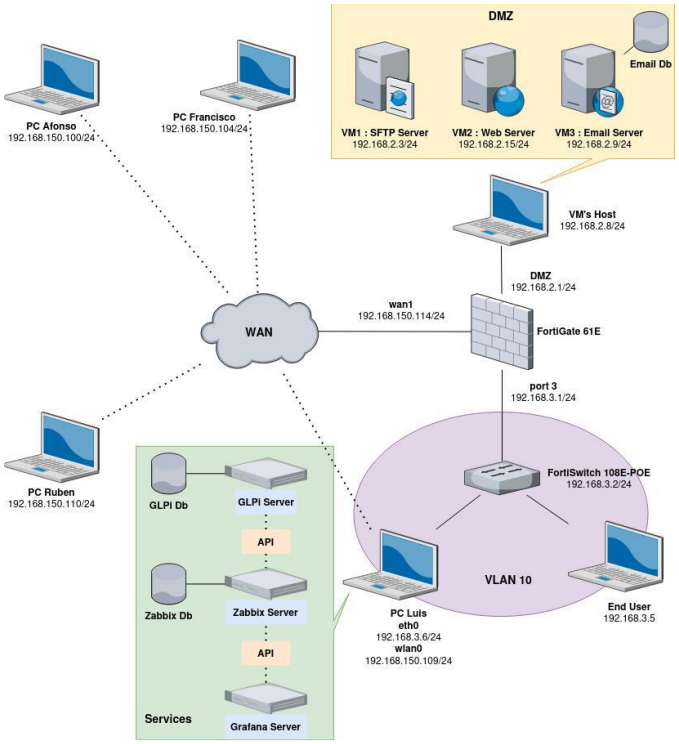

Figura 1 - Arquitetura de rede

Importante ressaltar que, na configuração da firewall, as regras foram estritamente definidas para permitir apenas os protocolos essenciais à comunicação interna da rede e ao acesso á internet. Esta abordagem restritiva assegurou que apenas os protocolos vitais para a operacionalidade dos servidores, comunicação dentro da rede e acesso seguro á internet estivessem autorizados, reduzindo assim possíveis vetores de ataque e garantindo a integridade da rede.

V. ZABBIX, GRAFANA E GLPI

No atual cenário digital, a monitorização e gestão eficaz da infraestrutura de rede é fundamental para garantir a continuidade dos negócios, a segurança e a otimização dos recursos. Neste contexto, a implementação do Zabbix, em conjunto com o Grafana e o GLPi, oferece uma solução integrada para a monitorização, visualização e gestão de incidentes na rede.

A. Implementação do Zabbix

O Zabbix foi adotado como a principal ferramenta de monitorização da infraestrutura de rede, dada a sua versatilidade e capacidade de monitorizar uma vasta gama de métricas. Desde a saúde do hardware, passando pelo tráfego da rede até o desempenho de aplicações, o Zabbix proporciona uma visão abrangente do estado operacional da infraestrutura. As alertas e triggers configuráveis garantem que as equipas sejam notificadas proactivamente sobre potenciais problemas, permitindo uma rápida intervenção.

B. Integração com o Grafana

Para melhorar a visualização das métricas recolhidas pelo Zabbix, foi implementada uma integração com o Grafana. Esta ferramenta de visualização *open source* amplia as capacidades do Zabbix, permitindo a criação de painéis de controlo interativos e visualmente atrativos. Assim, os administradores de rede têm à disposição uma interface intuitiva que destila as informações complexas da rede em representações gráficas claras e compreensíveis.

C. Integração com o GLPi

O GLPi, uma plataforma de gestão de serviços de TI, foi incorporado ao ecossistema para gerir incidentes e pedidos relacionados à infraestrutura de rede. Com a integração do Zabbix ao GLPi, os alertas e notificações podem ser automaticamente transformadas em tickets no sistema de gestão, facilitando a coordenação e a monitorização de incidentes. Esta integração assegura que os problemas identificados sejam devidamente registados, priorizados e resolvidos de forma eficaz e tempestiva.

VI. SCRIPT EM BASH

No âmbito de um estudo solicitado internamente, buscou-se elucidar atividades rotineiras com recurso ao Zabbix, utilizando um ficheiro CSV como fonte de dados. Para tal, configurou-se um item e um trigger no Zabbix. O critério do trigger foi estabelecido para disparar um alerta de severidade "Desastre" se o último dado recebido não fosse zero.
O script em bash, que serve como ponto central deste processo, verifica o ficheiro CSV e envia linhas com data idêntica à da máquina para o item do Zabbix:

```
#!/bin/bash
DATE_FORMAT="+%d-%m-%Y"
# Data da máquina
TODAY=$(date "$DATE_FORMAT")
tail -n +2
"/home/kali/Documents/scriptTasksZabbix/tasks.csv"
| while IFS= read -r line
do
if [[ $line = "$TODAY"* ]]
then
zabbix_sender -z 127.0.0.1 -s ZabbixServer -k
due.tasks -o "$line"
fi
done
```

Uma otimização sugerida para esta abordagem é a implementação de um *cron job*, que permitiria a execução automática do script em intervalos determinados, aumentando a eficiência e reduzindo intervenções manuais.

VII. ZABBIX, GRAFANA E GLPI

Face à complexidade crescente das infraestruturas tecnológicas, surge a imperativa necessidade de centralizar o acesso a ferramentas vitais como Zabbix, Grafana e GLPi. Esta centralização visa otimizar fluxos de trabalho e auxiliar na tomada de decisões bem fundamentadas.

A solução proposta é uma aplicação em React Native, tanto móvel quanto web. Esta aplicação unifica o acesso a estas ferramentas em diferentes plataformas, como web, Android e iOS, oferecendo uma interface coesa e intuitiva que agiliza a interação com cada sistema.

A. Características do React Native

- Desenvolvimento Multi-Plataforma: Permite criar aplicações para múltiplas plataformas a partir de um único código-fonte, otimizando o desenvolvimento e a manutenção.
- Interface Nativa e Alto Desempenho: Proporciona uma experiência de usuário comparável às aplicações desenvolvidas em linguagens nativas.
- Reutilização de Componentes: Favorece a consistência e diminui o tempo de desenvolvimento ao reutilizar componentes entre plataformas.

B. Sobre a aplicação

A aplicação foi concebida para ser intuitiva, com destaque para botões que direcionam rapidamente os utilizadores para funcionalidades específicas. Esta abordagem simplifica o acesso a serviços essenciais, eliminando a necessidade de memorização de endereços específicos de servidores.

C. Relevância em ambientes de produção

Em ambientes que integram múltiplos servidores como Zabbix, GLPi e Grafana, a centralização do acesso é crucial. Com a aplicação React Native, garante-se uma interface singular para todos esses serviços, contando que haja conexão com a rede corporativa, simplificando o processo e aumentando a eficiência operacional.
A aplicação em iOS e web pode é ilustrado nas figuras subsequentes, Figuras 2 e 3.

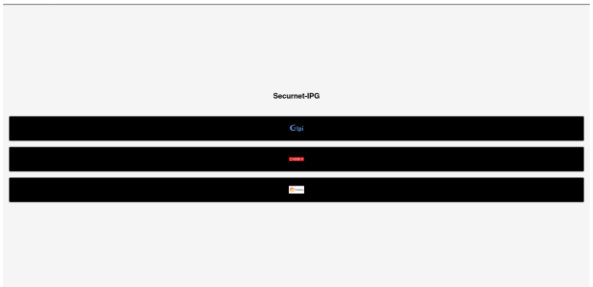

Figura 2 - App em web

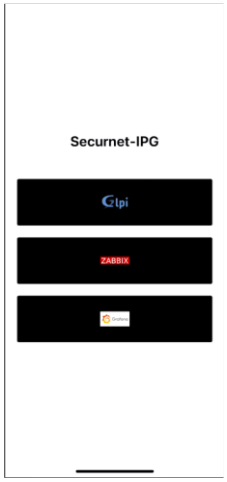

Figura 3- App em iOS

VIII. MACHINE LEARNING

À medida que as empresas modernas são inundadas por uma crescente onda de dados, o imperativo de extrair valor desse mar de informações torna-se mais urgente. O atual cenário digital, com sua constante corrente de dados, detém inúmeras oportunidades. Se interpretado corretamente, esse fluxo pode guiar decisões estratégicas, realçar a eficiência operacional e proporcionar uma vantagem competitiva.
Neste cenário, o *machine learning* destaca-se como uma força transformadora. Com a habilidade de decifrar padrões e irregularidades nos dados, otimizar processos e fundamentar tomadas de decisão, os algoritmos de *machine learning* apresentam-se como aliados valiosos. Considerando a avalanche de eventos que o Zabbix pode gerar, empregar técnicas de *machine learning* para filtrar e priorizar esses eventos torna-se quase obrigatório. Afinal, discernir entre eventos vitais e ruídos em sistemas complexos é um desafio.
A Figura 4 fornece um visual para este processo, ilustrando o fluxograma de ação quando um alarme é disparado no Zabbix.

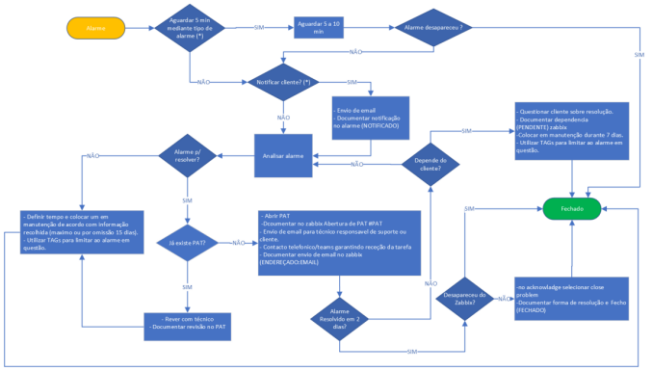

Figura 4 - Fluxograma de procedimento após alarme no Zabbix

Será aprofundada a exploração em torno da escolha e aplicação do algoritmo KMeans para segmentar eventos do Zabbix. Escolhido devido à sua eficácia em tratar dados numéricos contínuos e pela clareza na interpretação dos clusters formados, o KMeans também se destaca pela sua eficiência computacional, um atributo valioso ao lidar com grandes volumes de dados.

KMeans é um pilar no domínio do *clustering*, cuja missão é segmentar um conjunto de dados em K grupos predefinidos, identificando cada cluster pela sua média ou centroide. A operação deste algoritmo abrange a seleção inicial de K centroides, a atribuição de pontos do conjunto de dados ao centroide mais próximo e a calculação contínua dos centroides até a convergência.

Entretanto, como qualquer ferramenta, o KMeans tem as suas particularidades. A sua eficácia pode ser afetada pela seleção inicial de centroides, e seu design pressupõe que os clusters são esféricos e equilibrados. Além disso, é sensível a *outliers*, exigindo cuidados adicionais na preparação dos dados.

Uma ferramenta crucial no arsenal do KMeans é a distância euclidiana, uma métrica que quantifica a separação entre dois pontos em um espaço dimensional. Esta métrica, fundada nos ensinamentos do matemático Euclides, é fundamental para determinar a proximidade e atribuição de pontos aos centroides.

A. *Elbow method*

Primeiramente, foi determinado o número ideal de clusters para segmentar os eventos. Utilizando o *elbow method* com o algoritmo KMeans, constatou-se que o ponto de "cotovelo" indicava um K ótimo de 3 clusters.

B. *Modelagem e processamento de dados*

Os dados extraídos do Zabbix foram processados e preparados para serem inseridos no modelo KMeans. Após a aplicação do algoritmo, os eventos foram segmentados em 3 clusters distintos, com base nas suas características intrínsecas.

C. *Avaliação do modelo*

Foi avaliado o desempenho do agrupamento usando o coeficiente de silhueta. Este coeficiente oferece um equilíbrio entre a coesão dos membros de um cluster e a separação entre clusters diferentes. Com um coeficiente de 0.8491, os resultados sugerem uma segmentação significativa dos eventos, com clusters bem definidos.

Para representar a eficácia do agrupamento, geramos gráficos que ilustram o coeficiente de silhueta de cada amostra e a distribuição dos eventos em relação à contagem e duração média. Isso proporcionou uma visão clara da pertinência de cada evento aos clusters atribuídos. Estes gráficos podem ser visualizados nas figuras seguintes Figura 3 e 4.

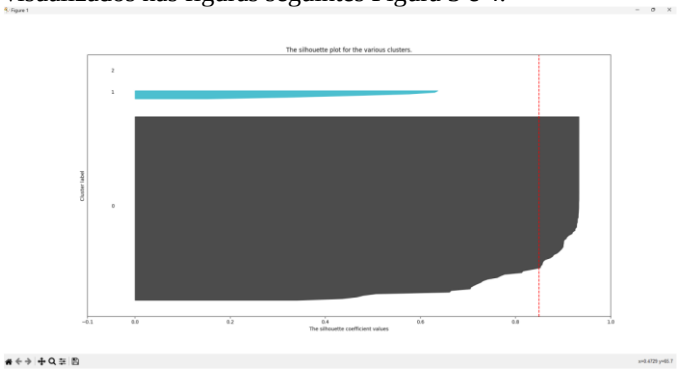

Figura 5 - Coeficiente de silhueta por cluster

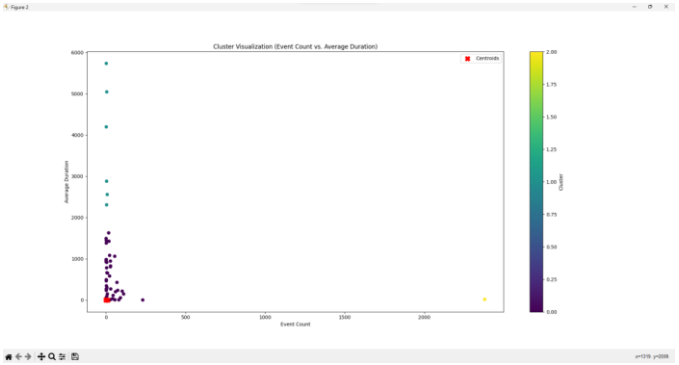

Figura 6 - Representação gráfica dos clusters

IX. CONCLUSÕES

Combinar monitorização de redes com *Machine Learning* oferece um caminho promissor para a gestão de eventos na era digital. Neste estudo, foi demonstrado um passo nessa direção, e a jornada de inovação continua. Futuros trabalhos podem explorar algoritmos de *clustering* alternativos ou a integração deste modelo diretamente na interface do Zabbix para insights em tempo real.