Implementação de sistema SIEM *open-source* em conformidade com o RGPD

Mestrado em Cibersegurança e Informática Forense

Ana Paula Henriques Vazão

Leiria, novembro de 2020

Implementação de sistema SIEM *open-source* em conformidade com o RGPD

Mestrado em Cibersegurança e Informática Forense

Ana Paula Henriques Vazão

Trabalho de Projeto realizado sob a orientação do Professor Doutor Carlos Rabadão, Professor Coordenador da Escola Superior de Tecnologia e Gestão do Instituto Politécnico de Leiria e co-orientação do Professor Doutor Leonel Santos, Professor Adjunto da Escola Superior de Tecnologia e Gestão do Instituto Politécnico de Leiria.

Leiria, novembro de 2020

Originalidade e Direitos de Autor

O presente relatório de projeto é original, elaborado unicamente para este fim, tendo sido devidamente citados todos os autores cujos estudos e publicações contribuíram para o elaborar.

Reproduções parciais deste documento serão autorizadas na condição de que seja mencionada a Autora e feita referência ao ciclo de estudos no âmbito do qual o mesmo foi realizado, a saber, Curso de Mestrado em Mestrado em Cibersegurança e Informática Forense, no ano letivo 2019/2020, da Escola Superior de Tecnologia e Gestão do Instituto Politécnico de Leiria, Portugal, e, bem assim, à data das provas públicas que visaram a avaliação destes trabalhos.

Dedicatória

Dedico este trabalho à minha família (Pais e Mana), aos meus amigos e a todos que me apoiaram durante esta longa difícil caminhada.

Agradecimentos

Muitas foram as pessoas que contribuíram para que fosse possível concretizar este trabalho, que contribuíram com apoio, ajuda e esforço.

Em primeiro lugar, quero agradecer ao Professor Doutor Carlos Manuel da Silva Rabadão e ao Professor Doutor Leonel Filipe Simões Santos, por me terem acompanhado e encorajado de forma incondicional e sacrificado o seu tempo de descanso para que pudesse concluir este trabalho.

Quero agradecer ao colega de mestrado e amigo Tiago Silva, também meu colega de trabalho, pela motivação e apoio diário pois este foi não foi um percurso fácil. Agradeço também a todos os meus colegas de mestrado e a todos os professores pelo apoio e ensinamentos.

Ao terminar esta etapa particularmente importante da minha vida, não poderia deixar de expressar o mais profundo agradecimento à Instituição onde trabalho, o Santuário de Fátima, pelos incentivos e pelo apoio disponibilizado, particularmente P. Carlos Cabecinhas, Reitor do Santuário. Gostaria de deixar uma palavra de particular apreço ao P. Vítor Coutinho, Vice-Reitor e meu antigo Diretor, ao Fausto Ferreira, meu atual Diretor, e ao David Mendes e ao Mário Ervilha, meus colegas do Gabinete de Sistemas de Informação do Santuário de Fátima, que me apoiaram e me ajudaram a trilhar este longo caminho e que contribuíram para a realização deste trabalho.

Deixo também um reconhecido agradecimento aos restantes colegas de trabalho, que de forma direta ou indireta, contribuíram ou auxiliaram na elaboração desta dissertação, pela paciência, atenção e força que prestaram em momentos menos fáceis. Destes, destaco de especial forma o contributo de Aline Venâncio e de Clara Costa na elaboração das traduções.

Quero deixar o meu profundo agradecimento ao meu Pai, à minha Mãe e à minha Irmã por me terem incentivado e colaborado em todas as etapas do projeto.

Para terminar, quero deixar, o meu profundo obrigado a todos que contribuíram para a realização deste trabalho e que não foram referidos anteriormente.

A todos, sem exceção, expresso os meus mais sinceros agradecimentos.

Nota Prévia

A componente prática deste projeto foi realizada numa empresa real, contudo a pedido da mesma foi alterado o nome e os seus dados foram anonimizados/mascarados.

Resumo

É inegável que um dos grandes desafios das Organizações é o de garantir a segurança dos seus dados, sendo que o Regulamento Geral de Proteção de Dados (RGPD) veio aumentar o nível de complexidade da segurança de um sistema informático, pois impõe que sejam aplicados níveis de proteção acrescidos.

Os sistemas *Security Information and Event Management* (SIEM) podem ajudar a ultrapassar os desafios criados pela obrigatoriedade do cumprimento do RGPD, pois permitem a definição de medidas técnicas para a proteção e controlo dos dados pessoais. Com o crescente aumento e a complexidade dos ataques informáticos, a implementação de um SIEM vai permitir mitigar os riscos e contribuir para proteger a informação, muitas vezes sigilosa, que as Organizações possuem, como por exemplo os dados pessoais.

No âmbito do presente trabalho foi implementado um sistema SIEM *open-source*, aplicando medidas técnicas para a proteção e controlo dos dados pessoais, por forma a assegurar a conformidade com o RGPD. Além disso, foi efetuada uma pesquisa documental sobre vários SIEM *open source* e, tendo em conta os resultados da mesma, selecionaram-se para o estudo comparativo os seguintes SIEM *open-source:* Graylog e o Alienvault OSSIM. Considerámos importante também incluir no estudo o SIEM Splunk, porque este disponibiliza uma versão freeware e é uma solução líder do mercado, e a solução Elastic Stack, pois é uma solução que é muito utilizada para a gestão de *logs* e, com recurso a outras ferramentas *open-source*, pode tornar-se num SIEM.

Depois de selecionadas as soluções a analisar, foi realizado um estudo comparativo entre as quatro: Graylog, Alienvault OSSIM, Splunk e Elastic Stack. Além disso, estas soluções foram testadas para aferir a facilidade de utilização e de administração, foram monitorizadas em tempo real para avaliar o seu comportamento relativamente a um ataque de força bruta e foram comparadas em relação a algumas das suas funcionalidades.

Como se considerou que seria importante basear a arquitetura do protótipo na rede de uma entidade real, foi efetuado o levantamento dos requisitos da rede informática da empresa XLog, definida a arquitetura e implementado o protótipo tendo como base a solução Elastic Stack, à qual foram adicionados outros componentes*:* ElastAlert, Slack e o ReadonlyRest. Além disso, foram realizados vários testes ao protótipo, a título de exemplo, simulou-se um

ataque com a ferramenta Pupy a uma máquina Microsoft Windows e, em simultâneo, identificaram-se os *logs* criados no processo. Durante a realização dos testes também se recolheram métricas de forma a mesurar o custo da pseudonimização dos dados sensíveis.

Em suma, através da implementação do protótipo SIEM *open-source* pretendeu-se criar uma ferramenta útil para a análise e deteção de ameaças em tempo real, mas que, em simultâneo, garantisse uma atuação em conformidade com cumprimento do RGPD.

Palavras-chave: Regulamento Geral de Proteção de Dados, *Security Information and Event Management*, Gestor de *Logs*, *Logs* de segurança, *Malware*

Abstract

It is undeniable that one of the great challenges Organizations have to face is to assure the security of their data, and the General Data Protection Regulation (GDPR) has come to increase the level of complexity in the security of a computer system, as it requires the application of added levels of protection.

Security Information and Event Management (SIEM) systems can help overcome the challenges created by mandatory compliance of GDPR, as they allow the definition of technical measures of protection and control of personal data. With the increasing rise and complexity of computer attacks, the implementation of a SIEM will allow to moderate the risks and it will contribute to the Organizations information protection, often confidential, such as personal data.

In this study, an open-source Security Information and Event Management system was implemented, applying technical measures for the protection and control of personal data, in order to ensure compliance with the General Data Protection Regulation. In addition, a documental research was carried out on several open source SIEM and, considering it´s results, the following open-source SIEM were selected for the comparative study: Graylog and Alienvault OSSIM. We also considered important to include the SIEM Splunk in the study, once it provides a freeware version and is a market-leading solution, and the Elastic Stack solution, since is a solution widely used in log management and, using other open-source tools, can become a SIEM.

After selecting the solutions for analysis, a comparative study was carried out among the four: Graylog, Alienvault OSSIM, Splunk and Elastic Stack. In addition, these solutions have been tested to evaluate their easiness of use and administration, monitored in real time to assess their behavior towards a brute force attack and compared regarding some of their features.

Because it was considered important to base the prototype architecture on the network of a real entity, it was conducted a survey of the XLog company's computer network requirements, the architecture was defined and the prototype was implemented based on the Elastic Stack solution, to which other components were added: ElastAlert, Slack and ReadonlyRest. In addition, several tests were made on the prototype, as an example, it was

simulated an attack with the Pupy tool on a Microsoft Windows machine and, simultaneously, the logs created in the process were identified. During the tests, metrics were also collected in order to measure the cost of pseudonymisation of sensitive data.

In summary, it was intended, through the implementation of the open-source SIEM prototype, to create a useful tool for analysis and detection of threats in real time that, at the same time, would guarantee a performance according with GDPR compliance.

Keywords: General Data Protection Regulation, Security Information and Event Management, Log management, Security logs, Malware

Índice

Lista de Figuras

Lista de tabelas

Lista de siglas e acrónimos

AD	Active Directory
ACK	Acknowledge
ACL	Access Control List
API	Application Programming Interface
APM	Application Performance Monitoring
ATT&CK	Adversarial Tactics, Techniques, and Common Knowledge
CA	Certificate authority
CERT	*Computer Emergency Readiness Team*
CPU	Central Processing Unit
CSV	Comma Separated Values
DPO	Data Protection Officer
DSL	Domain specific Language
ECE	Elastic Cloud Enterprise
ECS	Elastic Common Schema
EPS	Events per Second
ESM	Enterprise Security Manager
GELF	Graylog Extended Log Format
HDD	Hard Disk Drive
HIDS	Host Intrusion Detection Systems
HIPAA	Health Insurance Portability and Accountability Act

HTTP	Hypertext Transfer Protocol
HTML	HyperText Markup Language
HTTPS	Hyper Text Transfer Protocol Secure
ICMP	Internet Control Message Protocol
IDS	Intrusion Detection System
IIS	Internet Information Services
IoCs	Indicators of Compromisse
IP	Internet Protocol
IPS	Intrusion Prevention System
ISP	Internet Service Provider
JSON	Javascript Object Notation
LDAP	Lightweight Directory Access Protocol
LTS	Long Term Support
MAC address	Media Access Control Address
NIDS	Network Intrusion Detection Systems
NIST	National Institute of Standards and Technology
NoSQL	Not Only SQL
NSM	Network Security Monitoring
OSSIM	Open Source Security Information Management
OTX	Open Threat Exchange
PKI	Public Key Infrastructure
RAID	Redundant Array of Independent Disks

RAM	Random Access Memory
RAT	Remote Access Trojans
RBAC	Role-Based Access Control
RFC	Request for Comments
RGPD	Regulamento Geral de Proteção de Dados
ROCK	Response Operation Collection Kit
SaaS	Software as a Service
SEM	Security Event Management
SIEM	Security Information and Event Management
SIM	Security Information Management
SIRP	Incident Management and Response Platform
SPADE	Statistical Packet Anomaly Detection Engine
SPL	Search Processing Language
SQL	Structured Query Language
SSD	Solid-State Drive
SSH	Secure Shell
SSL	Secure Socket Layer
SYN	Synchronization
TCP	Transmission Control Protocol
TLS	Transport Layer Security
UDP	User Datagram Protocol
URL	Uniform Resource Locator

USB	Universal Serial Bus
USM	Unified Security Management
VPN	Virtual Private Network
W3C	World Wide Web Consortium
XML	Extensible Markup Language

1. Introdução

A proliferação de *malware* tem vindo a aumentar, quer em número de ocorrências, quer na complexidade da sua atuação. Segundo o relatório da empresa AV-TEST, até maio de 2019 foram reportados 903.14 milhões de programas maliciosos em 2019 e 865.63 milhões em 2018 (AV-TEST, 2019). Por outro lado, a empresa Malwarebytes refere no seu relatório que no ano de 2019 houve um aumento de 1% de *malware* relativamente ao ano de 2018, contudo, a nível empresarial, existiu um aumento de 13% de *malware* (Malwarebytes, 2020). Relativamente a este tipo de situações, é importante referir que os riscos para a informação não advêm apenas de fatores externos e que a ameaça também pode ser motivada por elementos internos às Organizações, como é o caso de um funcionário descontente. Tendo em conta estas possibilidades, é fundamental que as Organizações definam normas de conduta na resposta a incidentes de segurança e que se automatize a execução de algumas das tarefas com o objetivo de conseguir uma atuação mais eficaz e rápida na resolução dos incidentes de segurança. Ferramentas como *Intrusion Detection System/Intrusion Prevention System* (para a prevenção e deteção de intrusões), *firewalls* (para controlo de tráfego de e para a Internet) e antivírus (para a verificação de vírus) são importantes na prevenção e resolução dos incidentes de segurança. Complementarmente, o *Security Information and Event Management* (SIEM) pode agregar as informações das ferramentas anteriores e, além disso, ainda pode detetar ataques e identificar comportamentos anormais, podendo reconhecer ataques do tipo *Zero Day*, o que vai possibilitar uma rápida resposta aos incidentes de segurança que sejam detetados (Orvalho, 2015; L. Sousa, 2016).

Em maio de 2018, com a entrada em vigor do Regulamento Geral de Proteção de Dados (European Commission, 2020; União Europeia, 2016), operou-se uma mudança drástica na abordagem do que é necessário fazer às questões que se prendem com a segurança e a privacidade dos dados pessoais, uma vez que o não cumprimento do regulamento é passível de sanções muito gravosas para a Organização. Devido ao RGPD, tornou-se necessária a implementação de um conjunto de procedimentos por parte de cada Entidade, por forma a garantir a salvaguarda dos direitos dos cidadãos que estão definidos no regulamento. A implementação de um SIEM numa Organização aumenta o desempenho da mesma em termos de segurança e permite a identificação mais precoce de ataques, assim como de eventuais

vulnerabilidades de violação de dados. Em suma, pelas suas potencialidades, um SIEM pode ser uma ferramenta muito útil para que uma Entidade possa atuar em conformidade com o RGPD.

Contudo, é inevitável que exista uma mudança de paradigma na implementação e administração de um SIEM devido ao RGPD, uma vez que os *logs* de segurança podem conter dados pessoais, tais como o utilizador, o email, o nome do computador ou até os *cookies* (informação de navegação) caso estes elementos identifiquem uma pessoa. O RGPD obriga a que se apliquem várias medidas para proteger os dados pessoais, como por exemplo a cifragem, a pseudonimização ou a anonimização, o que afeta a implementação das soluções SIEM, pois estas são obrigadas a implementar várias medidas técnicas para estarem em conformidade legal.

Em suma, é necessário repensar a forma como as Organizações tratam, recolhem e armazenam os dados pessoais e também quais as medidas técnicas e organizativas que permitem adequar o risco e a probabilidade de ocorrência de uma violação de dados pessoais com o nível de segurança pretendida.

1.1. Caraterização do Problema de Investigação

Com a entrada em vigor do RGPD é primordial que as Organizações possam garantir a monitorização dos acessos aos dados pessoais e vários níveis de segurança nas suas infraestruturas. Para tal é necessário implementar soluções que sejam escaláveis e financeiramente ajustadas à sua realidade.

A implementação de um sistema de monitorização da infraestrutura informática (no presente contexto estamo-nos a referir a um SIEM) possui vários desafios pois é complexa e, para além disso, em algumas situações, a superfície de ataque é vasta e pode existir uma grande quantidade de dados gerados, entre eles os registos de *logs*. Como os *logs* de segurança podem conter dados pessoais, é necessário garantir a segurança adequada aos mesmos, o que implica um aumento na complexidade da implementação e administração de um SIEM.

Perante o cenário anteriormente descrito, para a implementação de uma solução SIEM *é* necessário garantir que se apliquem várias medidas técnicas para garantir a segurança dos dados pessoais. O principal objetivo deste trabalho é a definição e implementação de uma

arquitetura de um sistema SIEM baseado em soluções *open-source*, incorporando medidas técnicas para a proteção e controlo dos dados pessoais assegurando a conformidade com o RGPD.

A deteção de ameaças e das violações de segurança são duas funcionalidades que o SIEM deve possuir, pois as mesmas podem comprometer a disponibilidade, integridade ou a confiabilidade dos dados da Organização.

Assim, é de referir que os principais objetivos definidos para o presente trabalho estão relacionados com a escolha do SIEM *open-source*, a conformidade com o RGPD, o levantamento de funcionalidades, a definição das linhas orientadoras, a criação de um protótipo e a sua avaliação.

Relativamente às medidas técnicas e organizativas descritas no Regulamento, vamos focar-nos nas medidas técnicas que devem ser integradas no SIEM *open-source*.

Em suma os objetivos deste trabalho são:

- Implementar um sistema SIEM *open-source*, incorporando medidas técnicas para a proteção e controlo dos dados pessoais assegurando a conformidade com o RGPD;
- Identificar as técnicas e os procedimentos mais adequados para a proteção dos dados pessoais;
- Realizar o estudo comparativo entre as soluções SIEM, identificando os pontos fortes e fracos de cada solução estudada;
- Definir e implementar uma arquitetura de um sistema SIEM baseado em soluções *open-source* e que permita incorporar as medidas técnicas para a proteção e controlo de dados pessoais, assegurando a conformidade com o RGPD;
- Criar um protótipo que implemente a arquitetura definida de um SIEM *open-source*;
- Documentar a implementação do protótipo;
- Testar e Avaliar o desempenho do protótipo.

Devido à complexidade dos SIEM o trabalho vai focar-se nas medidas técnicas sugeridas pelo RGPD. A arquitetura escolhida será obrigatoriamente *on-premise* pois esse foi um dos requisitos para poder realizar o estudo na empresa XLog.

1.2. Metodologia

Devido a contingências familiares, profissionais e também à sua complexidade, o presente trabalho teve uma duração de dois anos. Durante esta linha de tempo as soluções evoluíram, a perspetiva da investigadora relativamente ao assunto também mudou, o que obrigou à reformulação de várias partes do trabalho que já estavam dadas como terminadas. Todavia, tentou manter-se a estrutura base, porque foi devido à informação recolhida em determinados períodos temporais que se delineou a perspetiva a desenvolver.

É de referir que o presente estudo foi assente em três etapas: pesquisa documental, estudo comparativo e desenvolvimento de um protótipo.

Uma vez identificado o objetivo principal deste estudo, procedeu-se à recolha do material bibliográfico que permitisse a compreensão das temáticas trabalhadas e desenvolvidas no presente estudo. Os materiais consultados possibilitaram uma compreensão mais aprofundada do RGPD, dos SIEM e de todos os conceitos relacionados com os mesmos.

Foi realizado um estudo comparativo entre quatro soluções SIEM: Splunk[1], Elastic Stack[2], Graylog[3] e OSSIM[4]. Neste âmbito, foi efetuado um levantamento das funcionalidades, da arquitetura, e a conformidade com o RGPD. Também foi criado um cenário de testes para cada uma das soluções com o principal propósito de testar a usabilidade dos mesmos, a facilidade de instalação e de utilização.

Após o levantamento dos requisitos, que se baseou nas especificidades da empresa XLog e também no mapeamento das funcionalidades de um SIEM descritas no ponto 2.6.1, foi possível definir a arquitetura da solução proposta, assim como a implementação de um protótipo para um cenário de testes.

[1] https://www.splunk.com/
[2] https://www.elastic.co/elastic-stack
[3] https://www.graylog.org/
[4] https://cybersecurity.att.com/products/ossim

1.3. Motivações pessoais

O crescente número de ataques sofisticados e a obrigatoriedade legal de cumprimento do RGPD implica que quem desempenha funções relacionadas com as tecnologias informáticas tenha que ter uma preocupação acrescida com a segurança dos sistemas informáticos. A principal motivação para a realização deste trabalho académico foi a necessidade de ter formação específica na área.

A escolha da temática prende-se com o facto de ter sentido uma afinidade muito grande com as temáticas relacionadas com os *logs* de segurança e a sua gestão, neste caso específico os SIEM, aliada à necessidade que senti de ter formação especializada para desempenhar com mais eficiência as minhas funções profissionais.

1.4.Estrutura da Dissertação

Na introdução, foi caraterizado o problema de investigação, definidos os objetivos para o trabalho, descrita a metodologia escolhida

No segundo capítulo serão analisadas as temáticas relacionadas com a segurança da infraestrutura informática e a conformidade com o RGPD. Neste ponto, serão também analisados alguns trabalhos académicos no âmbito dos quais foram implementados SIEM ou gestores de *logs*.

O estudo comparativo das quatros soluções SIEM selecionadas é efetuado no terceiro capítulo, no qual também se descreve a implementação de um cenário de testes para as referidas soluções para aferir a facilidade de instalação e de utilização.

A descrição da implementação do protótipo é realizada no quarto capítulo, parte do trabalho no qual se analisa o cenário com e sem as medidas técnicas que permitem a conformidade com o RGPD.

Por último, no capítulo das conclusões, é efetuado um elenco das principais ideias abordadas neste trabalho, são feitas algumas considerações sobre os resultados obtidos. Também são identificadas algumas das limitações enfrentadas no decorrer do trabalho e identificados os contributos que o presente trabalho pode dar e são por último são avançados enumeradas algumas perspetivas de futura investigação.

2. Background e Estado da Arte

A segurança é um dos grandes desafios que as Organizações enfrentam, sendo necessário analisar e monitorizar os eventos de segurança e, através dos mesmos, detetar as inúmeras ameaças que podem comprometer a informação dos seus ativos.

A análise centralizada dos *logs* de segurança das aplicações, dos servidores, dos clientes, dos equipamentos de rede é fundamental para a identificação de eventuais anomalias, de vulnerabilidades e de ataques, pois permite correlacionar, analisar e armazenar dados das diferentes origens. Normalmente, quando se pretende centralizar *logs* de diferentes origens recorre-se aos gestores de *logs* ou aos sistemas SIEM (*Security Information and Event Management*).

A centralização e análise dos *logs* de segurança podem também contribuir para auxiliar as organizações na garantia de conformidade com o RGPD, porque é possível demonstrar e testar as medidas técnicas que garantem a segurança no tratamento da informação. Além disso, quando se implementa um gestor de *logs* ou um SIEM é necessário salvaguardar os dados pessoais que os mesmos armazenam com recurso a técnicas de anonimização ou pseudonimização.

Neste capítulo vão ser debatidos vários conceitos chave, os quais enumeramos de seguida: o Regulamento Geral de Proteção de Dados, segurança nos sistemas de informação, os ataques, os *logs*, os SIEM e a gestão de eventos de segurança de informação.

2.1. Regulamento Geral de Proteção de Dados

O Regulamento Geral de Proteção de Dados foi aprovado pelo Parlamento Europeu em 27 de abril de 2016 e entrou em vigor em 25 de maio de 2018, tendo sido previsto um período de dois anos para a sua implementação nos sectores público e privado. É de salientar que um regulamento, ao contrário de uma diretiva, pode ter aplicação direta no sistema jurídico dos diferentes Estados-Membros, sem que seja necessário transpô-lo para o direito interno de cada país (Saldanha, 2018).

Para as Entidades que tratam dados pessoais, o RGPD difere da legislação anterior num ponto muito importante que é o valor das coimas. Este valor teve um aumento exponencial, que pode chegar aos 20 milhões de euros ou a 4% do valor de faturação anual da Empresa

(Saldanha, 2018). Além disso, o RGPD possui um alcance bastante abrangente, pois inclui não apenas as Organizações da União Europeia, mas todas as Organizações que processam dados pessoais de cidadãos da União Europeia (Dezeure, 2018; Microsoft, 2020).

No Regulamento são definidos novos requisitos legais que determinam a forma como as Organizações devem processar, organizar e proteger os dados pessoais, estando previstas, como já foi referido anteriormente, sanções financeiras muito duras em caso de incumprimento (Zerlang, 2017). O objetivo deste conjunto de regras é o de devolver aos cidadãos o controle sobre os seus dados pessoais e regulamentar o ambiente dos negócios (Zarzosa, 2017). Nos termos do Número 1, Artigo 4º, deste Regulamento são considerados dados pessoais os dados que identificam o Titular ou dados que, embora não identifiquem diretamente a pessoa, permitem a fácil identificação do seu Titular (EUR-Lex, 2016).

Ou seja, o conceito de dados pessoais é bastante amplo, pois não se limita apenas aos dados constantes no documento de identificação legal, abrangendo toda informação em qualquer formato que possibilite a identificação do Titular (Dezeure, 2018; Magalhães & Pereira, 2018). Sendo assim, enumeramos alguns exemplos de dados pessoais (Dezeure, 2018; Magalhães & Pereira, 2018; Varanda, 2019): nome e apelido, morada, endereço IP (*Internet Protocol*), número de telefone, número do cartão de identificação, número de contribuinte, localização geográfica, *cookies* (informação de navegação), endereço *MAC* (*Media Access Control Address*); endereço de e-mail, número de utente dos serviços de saúde, entre outros.

É, também importante que se tenha especial atenção em relação aos "dados especiais" especificados no Artigo 9 do RGPD, porque o seu tratamento está proibido, com exceção de algumas situações muito específicas (Saldanha, 2018). São considerados "dados especiais", aqueles que revelem as seguintes informações de um indivíduo: a origem racial ou a origem ética, as opiniões políticas, as convicções religiosas ou filosóficas, a filiação sindical, os dados relativos à saúde, à vida sexual ou à orientação sexual e o tratamento de dados genéticos e biométricos que identifiquem inequivocamente uma pessoa (Saldanha, 2018).

O Regulamento sugere que sejam tomadas as medidas adequadas para a proteção dos dados pessoais, sendo que a cifragem (codificação das mensagens para que apenas pessoas autorizadas as possam ler), a minimização (limitar a recolha de dados pessoais ao estritamente necessário) e a pseudonimização (substituição dos dados pessoais por identificadores artificiais impedindo a identificação do Titular) são três das medidas que este

documento nomeia (Comissão Europeia, 2019; EUR-Lex, 2016; Magalhães & Pereira, 2018). O conhecimento de quem, de quando e da finalidade para a qual a informação é acedida, utilizada e armazenada são considerados mais importantes no RGPD do que a própria informação (Ricardo, 2018).

É muito importante ter em conta que estão definidos no RGPD três direitos fundamentais relativamente aos dados pessoais: o direito ao esquecimento, o direito à restrição de processamento e o direito à portabilidade dos dados (Paquette, 2018).

Pseudonimização e Anonimização

Anonimização é o processo de remoção ou alteração da informação pessoal existente nos dados, com o objetivo de impedir a identificação unívoca dos indivíduos (Pinho, 2017). A diferença entre a pseudonimização e a anonimização prende-se com o facto de na pseudonimização os dados poderem voltar a ser associados a um Titular recorrendo a técnicas computacionais bastante complexas, por outro lado, o estado original dos dados anonimizados não podem ser recuperados (Mourby et al., 2018; Varanda, 2019).

A pseudonimização exige que as informações suplementares (informações que permitem identificar o Titular) sejam mantidas separadas e sujeitas a medidas técnicas e organizativas, por forma a garantir que os dados pessoais não sejam atribuídos ao Titular (Magalhães & Pereira, 2018; Saldanha, 2018). Na pseudonimização, o processo de cifragem é fundamental, pois, com recurso a uma chave específica, permite que o texto original seja transformado em texto cifrado, a operação inversa é a decifragem, que transforma o texto cifrado no texto original (Varanda, 2019).

A pseudonimização dos *logs* pode ser implementada nas seguintes fases (Varanda, 2019): geração, análise, armazenamento e apresentação. Caso a pseudonimização seja implementada na geração de *logs* é necessário implementar uma solução específica para cada fonte ou método de criação de *logs*. É de referir que o proprietário da informação deve garantir a privacidade dos dados pessoais armazenados, sendo libertado o gestor dos sistemas informáticos desta responsabilidade (Varanda, 2019).

A pseudonimização na fase de ingestão dos registos pode ser efetuada entre o *host* e o servidor de gestão de *logs*, à saída da informação dos *host* ou à entrada desta no servidor. Esta operação obriga ao desenvolvimento de código específico para processar os diferentes

tipos de entrada, contudo, possui uma grande flexibilidade na sua implementação (Varanda, 2019).

A pseudonimização de registos concretizada através da duplicação de índices consome mais recursos porque parte da informação vai ser duplicada, mas é um método fácil de utilizar e de implementar e as pesquisas nos índices pseudonimizados ou nos originais são igualmente rápidas e eficientes (Varanda, 2019).

Finalmente, a pseudonimização na fase de apresentação dos registos não necessita que estes sejam pseudonimizados antes de serem arquivados, o que permite que o processo de ingestão seja muito rápido. Para os utilizadores comuns, o processo de pesquisa é mais lento, pois a pseudonimização de dados é feita na interface de monitorização (Varanda, 2019).

2.2. Segurança nos sistemas informáticos

O conceito de segurança nos sistemas informáticos refere-se aos cuidados, mecanismos e ferramentas que podem ser implementados para proteger a informação que as empresas possuem, principalmente a que é considerada sigilosa (Guimarães, Lins, & Oliveira, 2006), sendo que esta informação pode estar armazenada num sistema informático ou estar em circulação pela rede (Granjal, 2017). A norma ABNT NBR ISO/IEC 27002:2005 define a segurança da informação como "a proteção da informação de vários tipos de ameaças para garantir a continuidade do negócio, minimizar o risco ao negócio, maximizar o retorno sobre os investimentos e as oportunidades de negócio" (ISO/IEC 27002, 2005, p. 10).

Tendo em conta a relevância que a informação possui para as Organizações é essencial que estas implementem mecanismos e ferramentas que ajudem a garantir as propriedades consideradas fundamentais no âmbito da Segurança da Informação, tais como: a confidencialidade, a integridade, a disponibilidade, a autenticidade, o não-repúdio, a resiliência e o controlo de acessos (Granjal, 2017).

Considera-se que um incidente de segurança ocorre quando um evento adverso tem a possibilidade de comprometer a segurança dos sistemas informáticos de uma organização, ou seja, é considerado um incidente deste tipo sempre que um ativo de informação da organização está exposto a riscos (ISO/IEC 27005, 2011). Na eventualidade de não serem geridos de forma correta, os incidentes de segurança podem provocar um impacto negativo significativo numa instituição, sendo que a severidade do impacto de um incidente é medido

de acordo com as consequências que podem advir para a atividade da organização (ISO/IEC 27005, 2011).

2.3. Ferramentas de proteção dos sistemas informáticos

A segurança dos sistemas informáticos é um problema de ordem técnica e social. A nível técnico é um grande desafio, uma vez que é necessário gerir uma multiplicidade de arquiteturas de hardware, software e políticas de segurança. Por outro lado, torna-se também um problema social, uma vez que os utilizadores domésticos ou as pequenas empresas desconhecem os problemas de segurança que têm de enfrentar e não possuem mecanismos para minimizar os eventuais danos que possam sofrer (Zúquete, 2018).

O *malware* pode ser detetado em várias situações, por exemplo, pelos antivírus nos computadores dos utilizadores; através dos mecanismos de segurança que limitam a rede (*firewalls*); com recurso aos IDS (IDS - *Intrusion Detection System*) ou através dos mecanismos distribuídos que recolhem dados das ferramentas anteriores através de sensores, permitindo uma visão mais abrangente da atividade do *malware* (Brown & Stallings, 2017).

Uma ferramenta essencial em qualquer sistema operativo, sem exceção, são os antivírus, uma vez que todos os sistemas operativos possuem vulnerabilidades que permitem que o *malware* possa infetar os computadores (Lowe, 2017). Além disso, os antivírus possibilitam a verificação das assinaturas digitais de vírus conhecidos (Lowe, 2017) e eliminam o software malicioso (Tavares, 2015).

Para que fosse possível controlar os riscos inerentes das ligações dos equipamentos informáticos das Organizações a outras redes (não controladas pelas Entidade), como é exemplo a Internet, surgiram as *Firewalls* (Zúquete, 2018). A *firewall* atua entre a rede da Organização e a Internet, o que possibilita a criação de um muro ou perímetro de segurança com a finalidade de proteger a rede da Entidade contra ataques provenientes da Internet (Brown & Stallings, 2017). Por outro lado, a existência de um único ponto de saída de dados, permite, entre outras funcionalidades, a definição dos tipos de serviços que podem ser acedidos, a seleção de endereços IP, dos protocolos ou dos portos (Brown & Stallings, 2017).

O IDS é o conjunto de componentes constituídos por hardware ou software que têm como finalidade detetar, identificar e atuar em caso da ocorrência de atividades anormais da rede de uma Organização é (Tavares, 2015; Zúquete, 2018). Os IDS podem ser classificados em

dois tipos (Brown & Stallings, 2017; Tavares, 2015; Zúquete, 2018) relativamente ao seu âmbito:

- **IDS baseado em Host (HIDS - *Host Intrusion Detection Systems*)**: procura detetar atividades suspeitas nos eventos e identificar as caraterísticas da máquina na qual está instalado, e, caso se verifique a ocorrência de situações anómalas, emite alertas.
- **IDS baseado em rede (NIDS - *Network Intrusion Detection Systems*)**: analisa os protocolos de rede, de transporte e de aplicação que circulam na rede com o objetivo de identificar atividades suspeitas.

O IPS - *Intrusion Prevention System* tem uma atuação diferente do IDS, pois este apenas regista os eventos e gera alertas para o utilizador/administrador, em contrapartida, o IPS intervém diretamente na rede e bloqueia os eventos que considera maliciosos (Gordon, 2010; Peixinho, Fonseca, & Lima, 2013).

Um sistema SIEM também contribui para a segurança dos dados de uma entidade, pois agrega dados referentes aos fluxos de tráfego e aos *logs* de vários componentes de segurança e também outros elementos importantes para o funcionamento da Organização, tais como: servidores; sistemas operativos e aplicações, possibilitando que o analista obtenha uma visão global do sistema informático (Elbaz, 2016; Gartner, 2020). Através de um SIEM pode ser detetada a saída anormal de um grande volume de dados ou a ligação de um utilizador a uma quantidade incomum de servidores. Além disso, o desempenho de um SIEM pode melhorar significativamente através da utilização de algoritmos de *machine-learning* para a deteção de ameaças (Elbaz, 2016; Exabeam, 2020a).

Em síntese, é fundamental implementar várias ferramentas de segurança para a proteção dos dados de uma Organização, pois essa opção permite a existência de várias camadas de segurança e possibilita a identificação de incidentes de segurança conhecidos e os ainda não foram reconhecidas (*Zero Day*).

2.4. Ataques

Um ataque informático pode ser caracterizado como um conjunto de ações que têm a finalidade de explorar vulnerabilidades com o objetivo de viabilizar uma ação ilícita (Zúquete, 2018). O agente que efetua o ataque é o atacante, podendo ser definidos dois tipos de ataques: passivo e ativo (Brown & Stallings, 2017). Num ataque passivo, o atacante tenta

descobrir ou utilizar informações do sistema, embora não efetue alterações nos ativos da Organização, em contrapartida, num ataque ativo existe a tentativa de alterar os ativos e de afetar o seu funcionamento (Brown & Stallings, 2017).

Se os ataques forem classificados tendo como base a sua origem, são definidos como internos e externos. Um ataque interno é iniciado por um elemento que está no perímetro interno da Organização, mas que utiliza as credenciais para fins que não foram autorizados. Os ataques que são executados fora do perímetro da Organização, podem ser classificados de externos e, normalmente, os atacantes não possuem credenciais de acesso e, por isso, as suas ações são mais intrusivas (Brown & Stallings, 2017).

2.4.1. Metodologias

Um teste de penetração ou *pentest* é um conjunto de testes metodológicos que podem ser efetuados a redes de computadores, a sistemas operativos, a websites, a base de dados, a aplicações e a redes sem fios, entre outros, com o objetivo de descobrir, de mapear e de identificar todas as vulnerabilidades a que estão expostos. Através de um *pentest* é possível descobrir falhas e, posteriormente, criar mecanismos de defesa para as vulnerabilidades identificadas (Moreno, 2015).

As fases de um teste de penetração segundo Vacca (2012) são três e reproduzem as fases que um atacante recorreria para efetuar um ataque real (Vacca, 2012). Como se pode observar na imagem seguinte as fases são: de pré-ataque, de ataque e de pós-ataque.

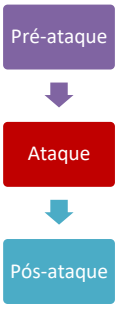


Figura 2-1 Fases de um *pentest*, adaptado de Vacca (Vacca, 2012, p. 531)

A primeira fase é a de pré-ataque, na qual o atacante ou a equipa de testes de penetração vai investigar ou explorar o potencial alvo, recolhendo o maior número de informações sobre a Organização que se pretende atacar. O reconhecimento em ordem a um ataque pode ser passivo e/ou ativo, sendo que o reconhecimento ativo pode ser detetado pelas ferramentas de proteção da Entidade. Esta fase permite a recolha de informações que podem ser utilizadas

para a criação de um mapa da infraestrutura que se pretende atacar, facilitando o planeamento e definição da estratégia de ataque (Vacca, 2012).

Na fase de ataque pretende-se que o alvo seja comprometido, pelo que o hacker ou a equipe de *pentest* podem explorar uma vulnerabilidade física ou lógica da Organização que foi identificada na fase de pré-ataque ou podem aproveitar a existência de uma política de segurança fraca para obter o acesso ao sistema. É de referir que o *hacker* só necessita de encontrar uma vulnerabilidade que lhe permita obter o acesso ao sistema, em contrapartida, a equipa de testes de penetração está empenhada em encontrar todas as vulnerabilidades existentes, pois ninguém pode garantir qual é a vulnerabilidade que vai ser utilizada para os acessos indevidos. Na eventualidade de o atacante conseguir obter o acesso ao alvo existem várias operações que pode executar: pode tentar escalar privilégios; instalar aplicações para garantir que mantém o acesso; tentar alargar o controlo a outros sistemas da Organização. Depois de atingirem o objetivo os atacantes podem tentar eliminar o seu rasto na rede, num processo que alguns designam de "cobrir as suas pegadas" (Vacca, 2012).

A última fase de ataque é exclusiva da equipa de testes de penetração, é denominada de fase de pós-ataque e consiste em colocar os sistemas informáticos no estado em que estavam antes do ataque. No *pentest*, para cada vulnerabilidade encontrada, é importante que sejam recomendadas estratégias de atenuação apropriadas aos riscos que as mesmas representam para a organização (Vacca, 2012).

Para que um *pentest* seja bem-sucedido, é fundamental que se adote uma metodologia de atuação, para garantir que todos os aspetos de segurança informática de uma atividade são testados (Vacca, 2012). Existem várias metodologias que podem ser utilizadas nos testes de penetração, pelo que, de seguida, são listadas algumas dessas metodologias (Beggs, 2014; Chebbi, 2018; Kevin Orrey, 2014; Moreno, 2015; OWASP, 2018; OWTF, 2019; Pentest-Standard, 2014; Pete Herzog, 2017):

- **OSSTMM (*Source Security Testing Methodology Manual*)** – tem como propósito avaliar a segurança digital, tendo em conta o objetivo de negócio (Pete Herzog, 2017);
- **ISSAF (*Information Systems Security Assessment Framework*)** – trata-se de uma Framework que imita as etapas de *hacking*, com algumas fases adicionais (Chebbi, 2018);

- **OWASP (*Open Web Application Security Project*)** – possui metodologia específica para servidores web e para aplicações web (OWASP, 2018);
- **PTF (*Penetration Test Framework*)** – apresenta um conjunto de etapas sequenciais e fornece hiperligações para ferramentas e comandos (Kevin Orrey, 2014);
- **PTES (*Penetration Testing Execution Standard*)** – trata-se de uma metodologia que reflete com eficiência as atividades levadas a cabo por uma pessoa maliciosa (Pentest-Standard, 2014);
- **OWTF (*Offensive Web Testing Framework*)** – é uma Framework que engloba várias ferramentas que permitem tornar o teste de penetração mais eficiente (OWTF, 2019);
- ***Backtrack*** – é uma metodologia que pode ser ajustada ao que é definido pelo projeto (Moreno, 2015).

As metodologias são importantes, mas possuem limitações e, por isso, devem ser enquadradas numa estrutura que analise a rede na perspetiva do invasor. Em 2009, do CERT (*Computer Emergency Readiness Team*), introduziu um conceito que é conhecido como "*Cyber kill chain*" e que contempla os passos dados por um atacante quando este ataca uma rede. Os ataques efetuados por um atacante nem sempre são lineares, pois algumas etapas podem ocorrer em paralelo, ou este pode efetuar múltiplos ataques ao mesmo alvo (Martin, 2020; Velu, 2017). O *Cyber kill chain* é formado por sete fases, como de seguida se enumeram (Branquinho, Seidl, Moraes, Branquinho, & Azevedo, 2014; Martin, 2020):

- ***Reconnaissance*** - conhecido no mundo da segurança da informação como *Information Gathering*, fase na qual é efetuado o levantamento de toda a informação disponível sobre o alvo;
- ***Weaponization*** - criação do *payload* que atacará/infetará o alvo de forma automática;
- ***Delivery*** - entrega do *payload* no ambiente informático do alvo, para que tal se concretize, podem recorrer-se a múltiplos mecanismos, tais como os anexos de emails, páginas com *malware* ou equipamentos USB (Universal Serial Bus) removíveis;
- ***Exploitation*** – após a entrega do *payload* é necessário que o código malicioso seja executado, para tal é imprescindível que sejam exploradas as vulnerabilidades do

sistema informático, por exemplo, recorrendo a funcionalidades de *autoexecução* ou até ludibriando os utilizadores para que executem o *payload*;

- ***Installation*** – manter o acesso permanente ao ambiente informático do alvo;
- ***Command & Control (C2)*** – garantir a comunicação entre o software malicioso e o atacante, para que este último possa recolher os dados do alvo ou enviar novas instruções maliciosas;
- ***Actions and Objetives*** – nesta fase o atacante pode ter atingido o seu objetivo ou planear novas ações.

Os modelos lineares possuem alguns pontos fracos pois não levam em consideração os comportamentos dos atacantes, o modelo "*diamond model*" procura identificar as motivações do invasor, a vitima e a tecnologia utilizada para realizar um ataque (Strager, 2020). O modelo "*diamond model*" de Betz (Caltagirone, Pendergast, & Betz, 2013; Roberts & Brown, 2017) difere em muitos aspetos das fases de um teste de penetração, uma vez que procura compreender a interação entre os vários atores (adversário e vítima) e também as ferramentas do atacante (infraestrutura e capacidades), no entanto, os dois podem complementar-se, pois oferecem duas perspetivas diferentes sobre o mesmo problema (Caltagirone, Pendergast, & Betz, 2013; Roberts & Brown, 2017).

A Framework MITRE ATT&CK (Adversarial Tactics, Techniques & Common Knowledge)[5] organiza e categoriza os diferentes tipos de ataques, as ameaças e os procedimentos que foram realizados por atacantes, o que vai possibilitar a identificação de vulnerabilidades nos sistemas informáticos. Estas informações são recolhidas em tempo real e estão divididas em diversas matrizes: PRE-ATT&CK, ATT&CK Enterprise e a Mobile (Lubeck, 2019; MITRE, 2020b).

As matrizes PRE-ATT&CK e ATT&CK Enterprise apresentam um conjunto de táticas que se alinham com o *Cyber kill chain*, sendo que o PRE-ATT&CK está alinhado com as três primeiras fases do *Cybe kill chain* e a matriz ATT&CK Enterprise está alinhada com as quatro últimas fases deste (Anomali, 2020). O modelo *Diamond Model* e a Framework MITRE ATT&CK também são complementares, pois o primeiro permite agrupar as invasões e a segunda documenta detalhadamente o comportamento do atacante (MITRE, 2020a).

[5] https://attack.mitre.org/

Contudo, a Framework MITRE ATT&CK também possui diferenças com o *Cyber kill chain* e o *Diamond Model*, pois o primeiro não se foca na perspetiva do invasor e o segundo não apresenta profundidade técnica. Com o MITRE ATT&CK existe um grande foco no atacante, permitindo que se possa avaliar os recursos de tecnologia de segurança de uma Organização (Teiss, 2020).

Devido à sua complexidade e também ao seu grau de abstração, muitas vezes procura-se simplificar os modelos de deteção de intrusões, resumindo-os em ciclos ou em modelos, no entanto, através deles, é possível compreender a complexa interação entre o defensor e o atacante e fornecem a base para o planeamento de resposta a incidentes de segurança (Roberts & Brown, 2017).

2.4.2. Tipos de ataques

A existência de políticas de segurança pouco eficazes, as vulnerabilidades do software e do hardware possibilitam múltiplos e numerosos tipos de ataques que exploram as fragilidades dos sistemas informáticos de uma Organização. Alguns dos ataques acontecem devido à utilização de ferramentas que se encontram na Internet por parte do utilizador, o que vai aumentar a eventualidade de ocorrência de um ataque (Santos, Bessa, & Pimentel, 2008).

Session Hijacking é uma técnica que consiste na usurpação da entidade de um utilizador, com o objetivo de obter informação relevante (Santos et al., 2008). O atacante consegue obter a chave de sessão válida de um utilizador (como é o caso de um cookie exclusivo), o que vai permitir o acesso não autorizado a serviços ou sistemas (Lallan, 2015) privados ou institucionais.

Outro tipo de ataque que pode ser levado cabo é o que recorre ao *SQL Injection*, sendo que este ataque pode acontecer quando um scanner de porta encontra uma vulnerabilidade na base de dados, que permita a execução de instruções de SQL não autorizadas, o que pode possibilitar que o atacante possa aceder às passwords da base de dados (Lallan, 2015).

Por outro lado, os atacantes também podem utilizar ferramentas que ajudam o utilizador a recuperar uma password esquecida para conseguirem aceder de forma não autorizada ao sistema, este tipo de ataque é denominado de *Password Cracking*. De uma forma geral, estas ferramentas tentam adivinhar de forma sistemática a senha (Lallan, 2015).

Existe uma grande diversidade de ataques que um atacante pode levar a cabo, no Anexo A – Exemplos de *Malware* descrevem-se alguns dos mais frequentes: negação de serviço (*Denial of servisse - DoS*), Código Malicioso ou *Malware* e a Engenharia social.

Para se proteger a informação devem adotar-se medidas preventivas que impeçam o atacante de ser bem-sucedido, pois os ataques descritos anteriormente podem causar sérios danos nos sistemas informáticos, assim como perdas financeiras muito graves para as Organizações, uma vez que permitem o acesso do atacante ao sistema e, em alguns casos, até o seu controlo.

2.5.*Logs*

Os *logs* são os registros dos eventos que ocorrem nos sistemas operativos, nas aplicações e nos equipamentos de uma Organização (Kent & Souppaya, 2006), possibilitando também o acesso à hora, à data e a outras informações relacionadas com o evento que o gerou (Rouse, 2012). Originalmente os *logs* eram utilizados para fazer a correção de problemas, sendo que atualmente estes podem ser utilizados em diversas funções através da correlação da sua informação (Kent & Souppaya, 2006). Damos o exemplo da função correlação que permite relacionar *logs* (normalmente de vários componentes distintos) num intervalo de tempo, possibilitando a identificação de situações anómalas que não seriam evidentes quando se analisa um único *log* (Kent & Souppaya, 2006; Lima, 2014).

Uma mensagem de *log* possui, de uma forma geral, três elementos fundamentais (Chuvakin, Schmidt, & Phillips, 2012; Escola Superior de Redes, 2015):

- ***Data/hora (Timestamp)*** – registo da data e da hora em que a mensagem de *log* foi gerada;
- **Origem** – sistema que gerou a mensagem de *log*, pode, por exemplo, ser um endereço IP ou o nome da máquina;
- **Dados** – informação sobre o evento gerado.

O conjunto de informações contidas num ficheiro de *log* depende do tipo de sistema que está a gerar os eventos. Por exemplo, um router possui informações como a "carga, *status* das *interfaces* e taxa de utilização das *interfaces*", enquanto que um servidor *web* contém informações específicas dos próprios serviços (Escola Superior de Redes, 2015).

Os *logs* podem possuir várias informações (Chuvakin et al., 2012): informações específicas sobre o que aconteceu, quando aconteceu e, caso seja, relevante quando o evento terminou, onde o mesmo decorreu (identificar o *host*, sistema de ficheiros, interface de rede, entre outros), quem esteve envolvido e a origem do evento.

2.5.1. Tipos de *logs*

Os *logs* podem ser categorizados em quatro tipos (Chuvakin et al., 2012; Q. Li & Clark, 2015): de segurança, operacionais, de *debugging* e de conformidade. Os *logs* de segurança contêm informações relacionadas com a segurança, têm como principal objetivo a deteção e resposta a ataques de *malware*, a tentativas de roubo de dados, assim como outros problemas relacionados com a segurança dos dados (Kent & Souppaya, 2006; Q. Li & Clark, 2015).

Como o termo indica, os *logs* de *debugging* são definidos pelos programadores de aplicações ou de sistemas para recolher informação sobre eventuais bugs. Geralmente os *logs* em produção estão desativados, pois podem ser gerados em grande número e, devido a isso, prejudicar o desempenho das aplicações/sistemas (Chuvakin et al., 2012; Q. Li & Clark, 2015).

Os Logs operacionais fornecem em tempo real informações sobre as atividades dos sistemas operativos ou que sejam relativas ao estado das aplicações. Finalmente, os *logs* de conformidade fornecem métricas para que possa ser avaliada a conformidade com os regulamentos legais (Chuvakin et al., 2012; Q. Li & Clark, 2015).

Em suma, os *logs* podem ter múltiplos propósitos, é de referir, contudo, que no presente trabalho se pretende recolher e gerir *logs* que possuam relevância para segurança dos sistemas informáticos. Os autores Kent e Souppaya (2006) identificam o software de segurança, os sistemas operativos e as aplicações como as principais fontes geradoras de *logs* (Kent & Souppaya, 2006).

É de salientar que os softwares de segurança geram uma parte significativa dos *logs* de segurança, listamos como exemplo os seguintes (Kent & Souppaya, 2006; Varanda, 2019): software *anti-malware* (antivírus), software de acesso remoto (VPN - Virtual Private Network), *web proxies*, software de gestão de vulnerabilidades, servidores de autenticação (Active Directory), routers (implementação de políticas para bloquear ou permitir determinado tipo de tráfego), *firewalls*, software de controlo de acesso à rede, entre outros.

Os sistemas operativos englobam servidores, máquinas clientes e dispositivos de rede (routers e *switches*). Os eventos de sistema (funções realizadas por componentes do sistema operativo) e os registros de auditoria (informações relacionadas com eventos de segurança) são os tipos mais comuns de dados de segurança (Kent & Souppaya, 2006; Varanda, 2019).

As aplicações podem gerar os seus próprios registos ou utilizar as funcionalidades do sistema operativo no qual estão instaladas. Os dados de segurança mais comuns nas aplicações são (Kent & Souppaya, 2006; Varanda, 2019): pedidos de cliente e respostas de servidor (podem ser registados os pedidos de URL - *Uniform Resource Locator* e a resposta dada pelo servidor), informação de conta (registo das alterações das configurações da conta ou tentativas de autenticação), informação de utilização (registo de métricas de utilização), atividades de operações significativas (registo das alterações de configurações ou falhas na aplicação).

Com o aumento do número, do volume e da diversidade de *logs* de segurança surge a necessidade de os gerir de forma eficaz, para que seja possível realizar operações de recolha, de remoção, de armazenamento e de análise (Kent & Souppaya, 2006). Esta temática vai ser desenvolvida com mais detalhe na secção 2.5.3 Gestão centralizada de *logs*.

2.5.2. Formato dos *logs*

A sintaxe e o formato dos *logs* definem como as suas mensagens são formatadas, transportadas, armazenadas e monitorizadas (Chuvakin et al., 2012). Existe uma enorme diversidade de formatos para a representação dos *logs* e também diversos protocolos de transmissão (Lima, 2014). Os formatos de *logs* são diferentes para cada solução e, para além dos formatos proprietários, é possível encontrar formatos de texto binários, como são os seguintes exemplos: *Comma Separated Values* (CSV), *Extensible Markup Language* (XML), *Syslog* e *JavaScript Object Notation* (JSON) (Lima, 2014).

Os *logs* podem ser criados por: sistemas operativos (*event logs* e *syslogs*), servidores web (*Microsoft IIS - Internet Information Services*, Apache, Nginx), servidores SSH - *Secure Shell*, *firewalls* (Windows e Linux), entre outros (Vacca, 2012; Varanda, 2019). Descreve-se no Anexo B – Formato dos *logs*, alguns dos formatos mais comuns de *logs*.

2.5.3. Gestão centralizada de *logs*

A análise de mensagens de *logs* ou a análise de *logs* deve ser centralizada e possibilitar recolha de informação (Chuvakin et al., 2012), para que depois seja possível obter dados relevantes para a gestão do sistema informático.

Através da gestão centralizada de *logs* é possível recolher e armazenar num único local todos os *logs* dos diversos componentes, o que permite o acesso centralizado aos mesmos (Constantine, 2019), permitindo que se defina quando e onde esses *logs* são armazenados e qual o seu período de retenção (Cybershark, 2019). Um *software* de gestão centralizado de *logs* pode ser caraterizado pelas seguintes funcionalidades (Cybershark, 2019): recolha de *logs* de todos os equipamentos e das aplicações de uma rede; retenção; pesquisa e indexação de *logs*, o que possibilita que as pesquisas sejam mais rápidas e, finalmente, elaboração de relatórios.

Para que se possa compreender o ciclo de vida de um *log* no processo de gestão centralizada de *logs*, Daubner (2018) propõe o esquema que apresentamos na Figura 2-2 (Daubner, 2018).

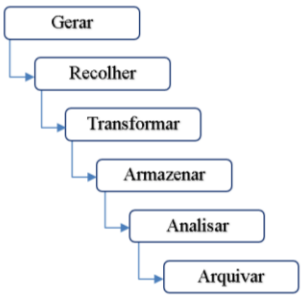


Figura 2-2 Tarefas da gestão centralizada de *Logs*, adaptado de Daubner (Daubner, 2018)

Como se pode constatar no esquema anterior, as tarefas do ciclo de vida de um *log* na gestão de *logs* estão dependentes da tarefa que a precede, ou seja, uma tarefa de um *log* só é executada quando a anterior estiver terminada.

Assim, a Geração de *logs* é a primeira tarefa do ciclo de vida no processo de gestão centralizada de *logs.* Esta tarefa pode ser complexa, pois existe uma grande diversidade de formatos resultantes de uma grande variedade de origens (Daubner, 2018).

Na tarefa de Recolha de *logs*, agregam-se os *logs* das diferentes origens para que possam ser processados e, na sua maioria, enviados para o servidor. De uma forma geral, os *logs* são transportados por software especializado denominado de *collector* (Daubner, 2018).

Os *logs* recolhidos, na generalidade dos casos, são transportados para um servidor. Existem várias formas de transporte, sendo que a mais simples é a que recorre ao protocolo UDP, no entanto, este não assegura um transporte confiável. Como a perda de um *log* de segurança não pode ser aceitável, é mais comum recorrer-se ao protocolo TCP, sendo que o protocolo UPD pode ser utilizado para recolha de métricas (Daubner, 2018).

A normalização de *logs* é uma das primeiras tarefas da Transformação de *logs*, no entanto, também se pode adicionar/remover dados. Contudo, em algumas situações a preservação dos *logs* pode ser importante. Por exemplo, em contextos de investigação criminal, pode ser muito relevante preservar a mensagem original, pois pode ser utilizada como prova (Daubner, 2018).

O Armazenamento centralizado é um procedimento fundamental na gestão de *logs* centralizada, mas que pode motivar a necessidade de armazenamento de um grande volume de dados. Para ultrapassar o eventual problema da gestão de um grande volume de dados, pode recorrer-se a uma base de dados relacional, como é exemplo o TimescaleDB[6]. Também se pode recorrer ao NoSQL, pois este permite a escalabilidade horizontal, com recurso a softwares como o MongoDB[7] ou Elasticsearch[8]. Outro método de armazenamento de *logs* é o *data lake*, este é centralizado e escalável para qualquer tipo de tamanho e de velocidade, podendo ser utilizado o software Apache Hadoop[9] (Daubner, 2018).

Uma das tarefas primordiais na gestão de *logs* é a Análise de *logs*, uma vez que a análise de *logs* está direcionada para a pesquisa e identificação de anomalias e de indícios de incidentes de segurança. No âmbito desta tarefa é necessário definir as metas: quais os *logs* a serem tratados, como deverão ser tratados, qual o limite de retenção dos *logs* armazenados, qual a frequência da análise e o que se pretende fazer com os resultados obtidos (Daubner, 2018).

No decorrer da tarefa Arquivar os *logs* que atingiram o período de retenção são arquivados, sendo de referir que, obviamente, nem todos os *logs* necessitam de ser arquivados. Por essa razão, devem ser definidas políticas que regulamentam quais os *logs* a armazenar e o período em que estes devem permanecer no arquivo definitivo. Neste contexto

[6] https://www.timescale.com/
[7] https://www.mongodb.com/nosql-explained
[8] https://www.elastic.co/elasticsearch/
[9] https://hadoop.apache.org/

é de referir que estes *logs* podem ser armazenados *offine* e comprimidos, o que permite reduzir os custos de manutenção do seu arquivamento.

Em suma, a gestão de *logs* pode ser definida por várias tarefas, funções ou fases, contudo, é de referir que em todas as perspetivas os *logs* necessitam de ser encaminhados das máquinas dos clientes para o servidor, têm de ser analisados para que se possa extrair a informação dos dados e devem ser armazenados num sistema que possibilite a sua visualização e a realização de pesquisas sobre as informações que os integram (Norrby, 2018).

2.5.4. Análise de *logs*

Utiliza-se o conceito de análise de *logs* quando se pretende descrever um incidente através de um conjunto de mensagens *log* (Chuvakin et al., 2012). A informação contida nos *logs* pode ser utilizada em diferentes etapas de gestão dos sistemas informáticos (Escola Superior de Redes, 2015): auditoria; estatísticas; reserva de recursos; identificação de falhas de hardware; investigação de incidentes de segurança; deteção de anomalias e possíveis ataques; e deteção de padrões de comportamento.

A análise de *logs*, como já foi referido, é muito importante para um auditor ou para um administrador de rede porque, através dos mesmos, podem ser detetados acessos indevidos ao sistema (Meyer, 2001). A título de exemplo, os acessos não autorizados ao sistema ou um *login* bem sucedido a horas suspeitas podem indicar que ocorreu um acesso indevido a esse sistema informático (Meyer, 2001). A análise de *logs* pode possibilitar a identificação de (Escola Superior de Redes, 2015): padrões de tráfego não comum; de comandos executados e processos instanciados; de alterações em sistemas e ficheiros; e de erros e exceções relacionadas com a execução de programas.

Devido à sua diversidade e volume, a análise de *logs* é uma tarefa complexa na qual é necessário enfrentar vários desafios (Escola Superior de Redes, 2015; Kent & Souppaya, 2006), tais como: grande volume de dados; ausência de informações; falsos alarmes; dados duplicados; diversidade de informações; dados inconsistentes; e dificuldade na obtenção de dados.

Em suma, a análise de *logs* é complexa e consome tempo e recursos, pois os sistemas que necessitam de ser monitorizados tendem a crescer. Mas, por outro lado, esta análise pode fornecer informação relevante sobre a segurança e a eficiência do sistema informático.

2.6. Security Information and Event Management

A monitorização das ocorrências de um sistema informático tem vindo gradualmente a aumentar de importância, pois têm surgido novas técnicas e métodos distintos de efetuar ataques informáticos (Tavares, 2015). Um dos principais objetivos de um SIEM é o de recolher e de concentrar a informação dos vários componentes num único local (L. Sousa, 2016), fazendo a monitorização dos privilégios de utilizadores, dos serviços e também das configurações dos sistemas (Johnson, 2015).

O termo SIEM foi utilizado pela primeira vez num artigo publicado pela empresa Gartner, que foi escrito por Mark Nicolett e por Amrit Williams, no qual foram descritas as funcionalidades do produto (Johnson, 2015; Kostrecová & Bınová, 2015; Whitman & Mattord, 2017; Zuech, Khoshgoftaar, & Wald, 2015). Atualmente, os SIEM podem ser disponibilizados em vários formatos: software, *appliances* ou em serviços online (Detken, Scheuermann, & Hellmann, 2015; Gordon, 2010; Johnson, 2015). O termo SIEM resulta da junção dos termos *Security Information Management* (SIM) e *Security Event Management* (SEM) (Catescu, 2018; Detken et al., 2015). O SIM realiza a monitorização dos *logs* em tempo real, a correlação dos eventos e as respetivas notificações, por outro lado, o SEM faz o armazenamento de dados a longo prazo, realiza análises e produz os relatórios (Catescu, 2018; Detken et al., 2015). Em síntese, podemos definir um SIEM como "um termo global usado para descrever um *software* que combina dois serviços: o SIM e o SEM" (Fernandes, 2015, p. 8).

O SIEM é uma ferramenta que recolhe e relaciona os eventos ocorridos na rede e que permite a criação de regras e de alertas que possibilitam a deteção de situações anómalas. Além disso, permite que sejam organizados os dados recebidos (evento, *log* ou notificação) de diversos componentes (servidor, firewall, IDS, router, etc.) de uma forma centralizada. Na Figura 2-3 é apresentada a estrutura de um sistema SIEM.

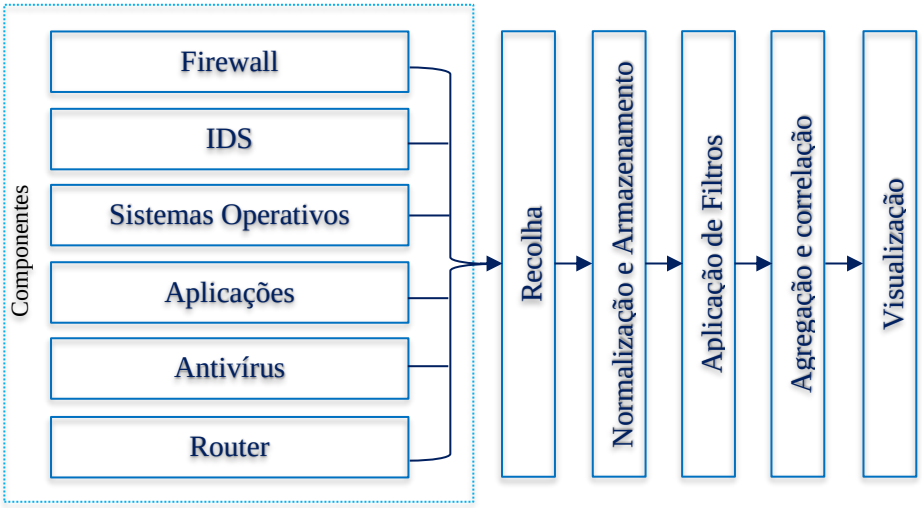


Figura 2-3 Estrutura de um sistema SIEM, adaptado de Ferreira (Ferreira, 2017, p. 11)

Como se pode constatar pela visualização da figura anterior, na fase de recolha são reunidos os dados dos diferentes componentes através de conectores. Estes componentes podem ter como base *hardware* ou *software* que interagem com o SIEM fornecendo-lhe os dados dos conectores (Ferreira, 2017).

Os componentes são diversificados, razão pela qual é indispensável normalizar os dados e guardá-los numa base de dados própria. Para eliminar a informação que não é relevante, é necessário recorrer a filtros e à aplicação de regras, só depois é que esta é visualizada sob a forma de alertas, de gráficos ou de relatórios (Ferreira, 2017).

Ou seja, os sistemas SIEM agregam toda a informação proveniente dos diversos componentes, fazem o seu tratamento e apresentam-na na consola de gestão (Alves, 2017). Através dos SIEM é possível saber os estados de segurança da infraestrutura informática, pese embora, estes sistemas não se limitem a recolher os dados dos diversos componentes e apresentá-los na consola, também possibilitam que se procurem padrões ao longo de vários meses, identificando outros tipos de ameaças (Pfleeger, Pfleeger, & Margulies, 2015). As equipas de resposta a incidentes de segurança dependem da grande capacidade de resposta e da eficácia dos alarmes, uma vez que estes podem ser o indício de possíveis ameaças (Alves, 2017). É possível efetuar várias pesquisas nos dados que os SIEM agregam, por exemplo, os analistas podem pesquisar todos os logins que foram efetuados entre a 1h e as 4h ou pesquisar os logins com endereços de IP de países que não estão contemplados nas políticas da Entidade (Pfleeger et al., 2015).

Na sua essência, o SIEM fornece uma visão ampla e detalhada da segurança de uma Entidade, possibilitando que os analistas realizem análises de segurança em tempo real (Watts, 2018).

2.6.1. Principais funcionalidades um SIEM

Tendo como base as informações contidas na bibliografia consultada, listamos um conjunto de funcionalidades que devem ser disponibilizadas por um SIEM (Arass & Souissi, 2019; Catescu, 2018; Cybershark, 2019; Exabeam, 2020b; Fortinet, 2019; Gartner, 2020; Graylog, 2018d; Logsingn, 2020; Mokalled et al., 2019; Petters, 2020; Vacca, 2012):

- recolher dados de diferentes origens;
- normalizar os *logs* de todos os componentes;
- Garantir a escalabilidade e a alta disponibilidade;
- analisar e correlacionar as informações dos equipamentos de segurança (IDS/IPS, *firewalls*, servidores) em tempo real;
- automatizar várias funções, reduzindo o tempo gasto na sua manutenção;
- realizar análises forenses aos *logs*, em tempo real, ou aos *logs* armazenados pelo sistema;
- efetuar análises complexas recorrendo ao *machine learning;*
- apresentar vários *dashboards* de segurança e permita a personalização de outros*;*
- criar relatórios personalizados e alertas;
- filtrar e destacar os eventos pela sua criticidade;
- detetar ameaças, incidentes de segurança e vulnerabilidades;
- emitir alertas nos casos em que existem atividades suspeitas na rede;
- disponibilizar *workflows* de resposta a incidentes;
- fazer a atualização em tempo real de ameaças;
- facilitar a *compliance*, como é exemplo o RGPD, ou seja, uma análise forense que salvaguarde o armazenamento dos dados em segurança;
- restringir os acessos e possibilitar vários níveis de permissões;
- fornecer vários mecanismos de segurança.

No âmbito do presente estudo, entendeu-se que seria pertinente resumir as funcionalidades essenciais de um SIEM, por esta razão optou-se por trabalhar o esquema

que se encontra na página da empresa Exabeam[10] (Exabeam, 2020b). Esta empresa elaborou um guia para explicar o que é SIEM e apresenta um esquema que resume as principais funcionalidades deste tipo de sistema. A figura seguinte é uma adaptação do esquema proposto pela Exabeam (2020) e que ilustra, de forma sucinta, as funcionalidades de um SIEM (Exabeam, 2020b). Segundo a Gartner, a empresa Exabeam foi classificada como sendo uma das empresas líder no Quadrante Mágico, que, no que diz respeito à avaliação de soluções SIEM (Ngo-Lam, 2020), avalia e classifica as soluções em quatro tipos: Líderes, Visionários, Nichos de Mercado e Desafiadores. A Gartner é uma empresa credível e credenciada que disponibiliza um grande número de análises e de estatísticas sobre as mais diversificadas tecnologias (LogRhythm, 2017).

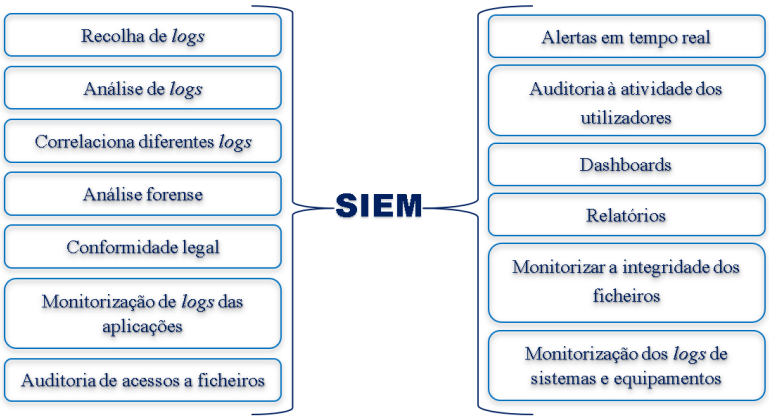


Figura 2-4 Esquema com as principais funcionalidades de um SIEM, adaptado de Exabeam (Exabeam, 2020b)

2.6.2. Vantagens e desvantagens

Uma das grandes vantagens dos SIEM é o facto de criarem *dashboards*, através dos quais é possível visualizar em tempo real o estado da infraestrutura de rede que está a ser monitorizada (Marques, 2018). Os SIEM também permitem organizar e centralizar um grande volume de *logs* e responder de forma mais rápida aos ataques identificados (Detken et al., 2015). Como um SIEM funciona de forma interrupta, consegue identificar incidentes de segurança em qualquer momento do dia (Walther, 2018). Em suma, permite aferir de forma fiável o grau de segurança da infraestrutura informática de uma Organização (Detken et al., 2015).

Por outro lado, os SIEM são caros e difíceis de administrar (Detken et al., 2015), pois são necessários técnicos especializados para a sua gestão, assim como equipamentos específicos e constante formação dos técnicos intervenientes (Walther, 2018). Nos casos em que é

10

necessário gerir um grande volume de dados, pode ocorrer um número significativo de falsos positivos, o que pode condicionar uma resposta rápida a uma situação de ataque (Zeinali, 2016). Por outro lado, os analistas confiam na eficiência dos SIEM, o que pode induzir a uma falsa sensação de conforto, levando a que estes técnicos se concentrem na gestão desta plataforma e que descurem outros métodos e mecanismos de auditoria de segurança informática (P. Rodrigues, 2013).

2.7. Gestão centralizada de *logs* versus Security Information and Event Management

Através da gestão centralizada de *logs* é possível compreender de forma mais detalhada um incidente e, além disso, por vezes, estes sistemas disponibilizam relatórios de conformidade que são muito úteis para a compreensão dos incidentes de segurança. Por outro lado, não sendo possível automatizar tarefas, vai exigir um maior número de especialistas na equipa de segurança. Outro problema que pode surgir está relacionado com o facto de os *logs* não estarem normalizados, o que vai tornar mais difícil a criação de relatórios (Cybershark, 2019).

Os SIEM e a gestão centralizada de *logs* estão profundamente interligados, todavia, diferem nos seus objetivos, na automatização e na análise em tempo real dos incidentes de segurança. O processo da gestão centralizada de *logs* tem como principal propósito a recolha e armazenamento dos dados, deixando para segundo plano a segurança, já um SIEM, pese embora também assegure a gestão centralizada dos *logs*, tem como um dos seus objetivos principais garantir a segurança dos dados (Catescu, 2018; Cybershark, 2019). Além disso, os SIEM possuem um grande número funções automatizadas, na correlação, na agregação, nos alertas e na criação de relatórios, minimizando, assim, a quantidade de trabalho do analista. Por outro lado, um processo de gestão centralizada de *logs* não possibilita um grande número de funções já automatizadas/pré-configuradas nos alertas e nas correlações, o que vai exigir um esforço maior por parte da Organização para conseguir obter informação significativa sobre os dados. Por último, um SIEM permite a análise em tempo real dos dados, pois disponibiliza um número significativo de tarefas automatizadas, o que não é possível através da gestão centralizada de *logs* (Cybershark, 2019).

Em suma, a segurança, a automatização de tarefas que permitem a análise em tempo real são fatores a ter em conta quando uma Organização opta por um SIEM ou por uma solução de gestão centralizada de *logs*.

2.8. Desafios na implementação de um *Security Information and Event Management*

O RGPD apresenta vários desafios para as Organizações no que diz respeito ao tratamento dos dados pessoais, pois estas passaram a ser obrigadas a aumentar amplamente a segurança e privacidade dos seus dados, dos sistemas informáticos e dos processos, sendo que as tecnologias como o SIEM podem ser uma ferramenta importante na superação desses desafios (Zerlang, 2017). A identificação dos fluxos de dados pessoais é fundamental para as Entidades que são obrigadas a estar em conformidade com RGPD e, tal como no caso anterior, os SIEM podem ter um papel fundamental neste processo (Zerlang, 2017). Em contrapartida, os dados recolhidos pelo SIEM podem conter informações pessoais, o que obriga à implementação de técnicas e de medidas adicionais para que as Organizações possam cumprir o regulamento (Boucas, 2018). É de referir que esta mudança de paradigma também constitui um desafio para a implementação de um SIEM.

Os SIEM podem recolher dados de centenas ou de milhares de fontes (base de dados, sistemas operativos, *firewalls*, dispositivos de rede, entre outros) o que vai implicar que milhões de *logs* sejam recolhidos diariamente por estes sistemas (Kotenko, Fedorchenko, Saenko, & Kushnerevich, 2018). Por outro lado, a grande diversidade e volume de *logs* recolhidos e analisados pelos SIEM podem significar um desafio de *Big Data*, pois o volume de dados armazenados cresce a uma grande velocidade (Jaeger, Cheng, & Meinel, 2018; Zuech et al., 2015) e estes devem ser processados com a maior brevidade, em ordem a poderem ser detetadas eventuais ameaças.

Devido ao grande volume de dados que os SIEM agregam e na eventualidade de se adotarem tecnologias de *Big Data*, é fundamental garantir a segurança dos *logs* e também fornecer os recursos informáticos adequados para o processamento e armazenamento dos mesmos (Kotenko et al., 2018). Uma das soluções possíveis pode passar pela adoção de uma solução distribuída (Spark, Storm, Trident ou Heron), através da qual é possível processar milhões de registos por dia utilizando um número reduzido de recursos (hardware) para o seu tratamento (Jaeger et al., 2018).

Por causa do aumento exponencial do volume dos dados a analisar, quando se pretende implementar um SIEM devem ser analisadas algumas das suas funcionalidades antes de ser escolhida uma solução, tais como: a relação entre o preço do *software* e o volume de dados que processa; as regras estáticas de correlação; a eficiência do rastreamento quando um

atacante está a efetuar um "movimento lateral" para escalar privilégios e o desempenho na automatização de tarefas. Caso se implemente um SIEM cujas mensalidades dependam do volume de dados enviados, podem ocorrer alguns constrangimentos, tais como: os custos com o SIEM podem aumentar consideravelmente ou, em oposição, só são enviados alguns *logs*, limitando a investigação de incidentes e a capacidade de deteção de ameaças. Ou seja, se o SIEM tiver restrições na sua atuação isso pode dificultar o trabalho do analista, uma vez que será necessário fazer um grande esforço manual para gerir as regras de correlação estáticas e ajustá-las às novas ameaças. Por outro lado, se o SIEM não conseguir enumerar todas as atividades realizadas pelos utilizadores, os atacantes podem mover-se pela rede sem serem detetados e, por último, se este sistema não oferecer a automação adequada, será necessário que a Organização tenha vários analistas para a gestão do SIEM, o que significa que a empresa enfrentará riscos maiores e períodos mais longos à exposição a ataques (Exabeam, 2019).

Para que o *machine learning* possa ser eficiente é necessário que os *logs* estejam normalizados, por outro lado, a velocidade na qual a normalização é executada também é relevante, pois, na eventualidade de serem tratados vários *logs* com formatos diferentes, é necessário ter um algoritmo que os trate de forma eficaz (Jaeger et al., 2018). Um sistema em produção está em constante mudança, o que significa que os algoritmos de *machine learning* aplicados na gestão dos *logs* enfrentam o desafio de criar uma solução de deteção de anomalias, que seja rápida, escalável, precisa e de baixo custo, para que se torne acessível a pequenas e a médias empresas (Collier & Azarmi, 2019).

Mesmo que uma Entidade adquira a solução de segurança mais cara do mercado, não é garantido que o produto funcione bem recorrendo apenas às configurações base, ou seja, é necessário que sejam criados vários cenários de teste que reflitam a realidade da Organização para que a solução seja eficaz (Anastasov & Davcev, 2014). A implementação de um SIEM pode ser um processo demorado, pois trata-se uma tarefa complexa, que pode implicar a formação dos quadros técnicos, a alteração das políticas de auditoria, a configuração de agentes para envio de *logs* e também a resolução de problemas que possam surgir devido às alterações de configuração (Pfleeger et al., 2015). Para ser eficiente, um SIEM deve adaptar-se à realidade da Entidade onde está implementado e também ter uma margem de evolução que garanta a deteção de novas ameaças. Para que se possa tirar partido da eficiência e eficácia dos SIEM é necessário que exista, pelo menos, um responsável que faça a manutenção e a personalização de novas funcionalidades (Pfleeger et al., 2015).

A agregação de todos os *logs* num único sistema (SIEM) também pode ter os seus riscos, pois a entidade está a depositar toda a sua confiança nos administradores e utilizadores deste sistema, em contrapartida, esta opção permite que sejam prontamente mitigados os riscos internos que possam advir da atuação de um funcionário descontente (Pfleeger et al., 2015).

Antes de implementar um SIEM é necessário conhecer bem a ferramenta, sendo fundamental perceber se o sistema escolhido oferece apenas as funções básicas e o administrador tem que implementar as restantes ou, por outro lado, se esta já vem pré-configurada, facilitando assim a sua implementação (Pfleeger et al., 2015).

2.9. Constrangimentos impostos pelo Regulamento Geral de Proteção de Dados

Existem alguns conceitos que devem ser tidos em consideração quando se implementa um SIEM, um deles é a proteção de dados por design e padrão, o que significa que um novo serviço ou negócio que utilize dados pessoais deve, na sua criação ou implementação (design), ter em consideração a privacidade dos dados que recolhe, adotando mecanismos de proteção para os defender. Por definição, significa que o responsável pelo tratamento dos dados só deve processar os dados que são absolutamente necessários, mantendo as configurações de privacidade para o utilizador com o máximo de proteção possível (Costa, 2017).

Um SIEM pode ser uma ferramenta importante para que as Entidades estejam em conformidade com as condições de proteção impostas pelo RGPD (Boucas, 2018; Malik, 2017) nos seguintes pontos (Boucas, 2018): mantém um registo das suas atividades de processamento; documenta o tipo de dados que são processados; define os objetivos do processamento; documenta quais os dados que foram tratados e com quem estes foram partilhados; define limites de retenção para os dados e garante que são tomadas as medidas de segurança adequadas para proteger os dados. Em caso de violação das regras estabelecidas, o SIEM também pode ajudar a fornecer algumas das informações que são requeridas em caso de incumprimento, por exemplo: quais são os dados pessoais que foram acedidos e afetados e quais são os riscos para os titulares desses dados (Boucas, 2018).

O RGPD alterou a definição do que constitui informação pessoal. Assim, os *logs* de acesso, de erros e de segurança contém informações pessoais, o que implica que as Organizações são obrigadas a implementar medidas de segurança para as proteger (Black,

2017). É importante que se tenha presente que dados pessoais como o utilizador, o pronome, o sobrenome, o e-mail, os cookies ou endereços IP podem integrar os *logs* que a solução SIEM armazena (Boucas, 2018). Para atenuar o incumprimento RGPD, é necessário recorrer a técnicas, como a criptografia, a anonimização e a pseudonimização, para reduzir o risco de identificação de um Titular através dados pessoais recolhidos ou, no mínimo, que se assegure a sua segurança quando armazenados (Boucas, 2018). Devem ser *pseudonimizados* e separados os dados pessoais dos registos, para que estes só sejam acedidos e utilizados em caso de necessidade de monitorizar qualquer tentativa de acesso local ou devido a uma tentativa de mover os dados para fora da rede (Boucas, 2018) Para a implementação de um SIEM é muito importante ter em consideração os seguintes artigos do RGPD: o Artigo 5 (Princípios relativos ao tratamento de dados pessoais); Artigo 15 (Direito de acesso); Artigo 16 (Direito de retificação); Artigo 20 (Direito de portabilidade); Artigo 17 (Direito ao esquecimento); o Artigo 32 (Segurança do Tratamento) e também que devem ser implementadas medidas para que a atuação da Organização esteja em conformidade com os mesmos. Existem outros artigos que, em caso de notificação de uma violação, são importantes e que o SIEM pode ajudar a responder, a saber: o Artigo 33 (Notificação de uma violação de dados pessoais à autoridade de controlo); o Artigo 34 (Comunicação de uma violação de dados pessoais ao Titular dos dados) e o Artigo 58 (Auditoria à segurança dos dados pessoais).

Quando se pretende proteger só uma parte específica dos dados, não se deve aplicar anonimização, pois embora esta técnica esteja em conformidade com o Artigo 17, não está em conformidade com o Artigo 32, porque vai alterar a integridade dos dados e comprometer os direitos salvaguardados nos artigos: 15, 16 e 20 (Lourenço, 2018).

Para que esteja em conformidade com o RGPD, uma Organização deve implementar várias medidas e diversos mecanismos de controlo, tais como (Malik, 2017): nomear um DPO (*Data Protection Officer*); notificar uma violação até 72 horas depois da sua ocorrência; ter um inventário de todos os dados pessoais processados; ter a proteção de dados organizada por design e por padrão; fazer a avaliação do impacto da privacidade dos dados na atividade da Organização; implementar vários controlos, políticas e procedimentos de salvaguarda da proteção dos dados pessoais; ter um plano de resposta a incidentes devidamente documentado e testado; criar um plano de contingência para notificar as partes relevantes, em caso de ocorrência de uma violação do RGPD; monitorizar a rede e o

comportamento dos utilizadores, para que possam ser identificados incidentes de segurança rapidamente.

Em resumo, o RGPD acrescenta complexidade à gestão dos sistemas de informação, pois é necessário redefinir as normas e os processos para que a Organização esteja em conformidade com esse Regulamento.

Medidas técnicas para a conformidade com o Regulamento Geral de Proteção de Dados

Existem várias medidas técnicas que a implementação de um SIEM deve ter em consideração para este possa estar em conformidade com o RGPD. De seguida, enumeramos as funcionalidades que estão referenciadas na bibliografia consultada (Black, 2017; Elastic, 2018; EUR-Lex, 2016; Malik, 2017; Petters, 2020; Saldanha, 2018; Simko, 2018; Splunk, 2019c; Varanda, 2019):

- Permitir a anonimização dos dados pessoais;
- Possibilitar a pseudonimização dos dados pessoais;
- Limitar a retenção e permitir definir tempos de retenção dos dados pessoais;
- Garantir a segurança dos dados dos pessoais;
- Efetuar notificações de violações de dados;
- Restringir o acesso aos dados pessoais;
- Auditar e monitorizar os acessos a dados pessoais;
- Garantir resiliência;
- Permitir a recuperação a desastres;
- Garantir a proteção de dados por design e por padrão;
- Possibilitar a criação de Relatórios de Conformidade.

2.10. Projetos académicos que implementam *Security Information and Event Management* ou Gestores de *Logs*

Como se pretendia implementar um sistema SIEM *open-source*, incorporando medidas técnicas para a proteção e controlo dos dados pessoais de forma a assegurar a conformidade com o Regulamento Geral de Proteção de Dados (RGPD), foi efetuada uma pesquisa documental sobre estas temáticas. Para a construção da Tabela 2.1 foram analisados sobretudo dissertações de mestrado, teses de doutoramento e relatórios técnicos, porque se pretendia realizar um trabalho académico. Para que fosse possível preencher a Tabela 2.1,

foram realizadas pesquisas no Google, no Google Scholar, no IEEE Xplore Digital Library e no B-on.

Nesta fase, é fundamental perceber como se deve implementar protótipo SIEM e o que deve ser feito para que este esteja em conformidade com o RGPD. Na Tabela 2.1 são apresentados os trabalhos que foram considerados relevantes para o presente estudo, tendo o ano de 2015 sido considerado como a data limite mínima, pelo que os trabalhos elaborados antes de 2015 não foram tidos em conta na pesquisa efetuada. Tomou-se esta opção, porque se entendeu que esta temática tem sido alvo de uma acelerada atualização, pelo que se optou por investir na consulta de trabalhos mais recentes, pois seriam os mais adequados à realidade atual.

Para que seja possível escolher um SIEM, considerou-se importante comparar várias soluções *open-source*. Assim, foram analisados os trabalhos de Tavares (2015) e Zarzosa (2017), que comparam o desempenho de soluções SIEM, pese embora o âmbito dos dois trabalhos seja diferente, pois utilizam soluções comerciais e *open-source*, são contributos importantes para fazer um levantamento das funcionalidades a comparar (Tavares, 2015; Zarzosa, 2017). Daubner (2018), no trabalho que desenvolve, compara várias soluções pagas e *open-source*, com a finalidade de implementar uma solução de gestão de *logs*, e desenha uma solução de gestão de *logs* que tem como base o Elastic Stack (Daubner, 2018). Similarmente, Vainio (2018) também compara várias ferramentas *open-source* com o objetivo de implementar um gestor de *logs*, sendo que este autor também implementa a solução ELK Stack (Vainio, 2018). Por seu lado, Bělousov (2019) compara duas soluções SIEM *open-source*, neste caso, o Graylog e o ELK Stack, e implementa, em produção, a solução ELK Stack na empresa Master Internet Inc.. Marquina (2018) faz a comparação entre vários SIEM e constrói um cenário no qual implementa um SIEM com várias ferramentas *open-source*, tais como: Elastic Stack, Wazuh, Search Guard e Sentinl[11] (Marquina, 2018).

Como já se referiu anteriormente, no presente trabalho pretende criar-se um protótipo SIEM, no qual se irá procurar aferir as facilidades de implementação e de utilização de algumas soluções *open-source*, pelo que os trabalhos desenvolvidos por Devender e Adike (2019), de Daubner (2018), de Delgado (2018), de Marquina (2018), de Tavares (2015), de Simko (2018) e de Zeinali (2016) irão ser muito relevantes na definição do protótipo SIEM

[11] Sentinl

(Daubner, 2018; Delgado, 2018; Devender & Adike, 2019; Marquina, 2018; Simko, 2018; Tavares, 2015; Zeinali, 2016).

Higbee (2015), Lourenço (2018) e Ventrella (2018) implementaram protótipos no âmbito dos seus estudos, pelo que, para a concretização dos objetivos deste trabalho, é fundamental que sejam testadas as funcionalidades num ambiente controlado. Por isso foi considerado relevante a implementação de um protótipo com a solução escolhida, pelo que os trabalhos elencados podem ser um importante auxílio na sua definição (Higbee, 2015; Lourenço, 2018; Ventrella, 2018).

Como se pode visualizar na tabela seguinte, existem vários trabalhos que implementam soluções *open-source* (Bělousov, 2019; Daubner, 2018; Delgado, 2018; Devender & Adike, 2019; Higbee, 2015; Jain, 2018; Marquina, 2018; Norrby, 2018; B. Rodrigues, 2015; Simko, 2018; Tavares, 2015; Vainio, 2018; Ventrella, 2018; Zarzosa, 2017), que testam ataques a um SIEM (Delgado, 2018; Tavares, 2015) pelo que foram considerados significativos para a pesquisa, pois fornecem elementos chave para o desenvolvimento deste estudo.

No que diz respeito ao RGPD, salientamos o trabalho de Lourenço (2018), pois este realiza um estudo sobre várias formas de encriptação num cenário de testes em que a atuação do SIEM esteja em conformidade com o referido Regulamento (Lourenço, 2018). Devido à escassez de estudos nesta área, os resultados obtidos no âmbito dos trabalhos que foram analisados no decorrer da pesquisa levada a cabo são relevantes para a implementação do protótipo.

Tabela 2.1 - Trabalhos relevantes para o estudo

Ano	Tipo de Trabalho	Universidade/ Instituição	Autor	Título	Descrição
2015	Dissertação de Mestrado	Universidade do Minho (Escola de Engenharia)	Luís Tavares	Análise de eventos de segurança: baseado no OSSIM	Realiza a comparação entre várias soluções SIEM: IBM Security, HP/ArcSight, Splunk, OSSIM (Alian Vault). Implementa um cenário de testes com a solução *open source* OSSIM onde são efetuados ataques.
2015	Dissertação de Mestrado	Universidade de Lisboa (Faculdade de Ciências Departamento de Informática)	Bernardo Rodrigues	Open-Source Intelligence em Sistemas SIEM	Implementa uma framework de segurança que integra na sua arquitetura o SIEM ArcSight.
2015	Dissertação de Mestrado	Brigham Young University	Mathew Higbee	Deriving System Vulnerabilities Using Log Analytics	No âmbito desta dissertação, foi implementado um protótipo com a solução *open source* ELK Stack.
2016	Dissertação de Mestrado	Tallinn University of Technology (Faculty of Information Technology Department of Computer Engineering)	Seyed Zeinali	Analysis of Security Information and Event Management (SIEM) Evasion and Detection Methods	Nesta dissertação recorrendo a um cenário de testes foram avaliadas as soluções comerciais Splunk e AlienVault USM (Appliance Security Management).
2017	Relatório	DiSIEM	Susana Zarzosa (Editora)	D2.1 In-depth analysis of SIEMs extensibility	Realiza a comparação de várias soluções SIEM: HP ArcSight; IBM QRadar; Intel McAfee Enterprise Security Manager; Alienvault OSSIM and USM; XL-SIEM; Splunk e o Elastic Stack.
2018	Dissertação de Mestrado	Universidade de Lisboa (Faculdade de Ciências - Departamento de Informática)	André Lourenço	Operational Intelligence with GDPR	Com recurso à solução Splunk, foram criados vários protótipos para as várias formas de encriptação de dados e analisada a conformidade com o RGPD.
2018	Dissertação de Mestrado	University of Houston (Faculty of the Department of Information and Logistics Technology)	Utkarsh Jain	Lateral movement detection using ELK Stack	É apresentada uma solução de recolha e análise de *logs* recorrendo ao ELK Stack.
2018	Dissertação de Mestrado	Politecnico Di Torino	Carlo Ventrella	Data Science for Information Security with Open Source technologies	Foi testada a implementação de um protótipo com recurso ao ELK Stack para que fosse possível recolher e analisar *logs*.
2018	Dissertação de Mestrado	Uppsala Universitet	Elias Norrby	Investigation and Implementation of a Log Management and Analysis Framework for the Treatment Planning System RayStation	O investigador recorreu à ferramenta *open source* Elastic Stack, assegurando a conformidade com os regulamentos e leis em vigor para analisar os bugs encontrados no sistema RayStation de várias clínicas.
2018	Dissertação de Mestrado	University of Houston (The Faculty of the Department of College of Information and Logistics Technology)	Pablo Delgado	Developing an Adaptive Threat Hunting Solution: The Elasticsearch Stack	Com recurso ao ELK Stack implementa-se um cenário de testes para identificar anomalias e ataques.
2018	Dissertação de Mestrado	Masaryk University (Faculty of Informatics)	Lukáš Daubner	Effective computer infrastructure monitoring	Nesta dissertação foram comparadas e descritas várias soluções: Operations Management Suite; Elastic Stack; Graylog e Nagios. Foi implementada recorrendo a um cenário de testes uma solução de gestão de *logs* recorrendo ao Elastic Stack.
2018	Dissertação de Mestrado	Masaryk University (Faculty of Informatics)	Severin Simko	Implementation of Systems for Intrusion Detection and Log Managemento	O trabalho analisa as soluções OSSEC e Graylog e implementa vários cenários de testes com o Graylog.
2018	Dissertação de Mestrado	Aalto University (School of Science)	Antti Vainio	Implementation of Centralized Logging and Log Analysis in Cloud Transition	O principal objetivo do trabalho foi o de analisar as soluções existentes para a gestão de *logs* (ELK Stack e Splunk). No trabalho foi implementada a solução ELK Stack.
2018	Dissertação de Mestrado	Universitat Oberta de Catalunya	Luis Miguel Jaso Marquina	Ventajas e Implementación de un sistema SIEM	Na parte teórica, o investigador fez um estudo comparativo entre várias ferramentas SIEM: QRadar; Splunk; LogRhythm; ESM (Enterprise Security Manager) da McAfee; USM da

Ano	Tipo de Trabalho	Universidade/ Instituição	Autor	Título	Descrição
					AlientVault; ArcSight; RSA NetWitness Suite da Dell e o Elastic Stack. Na parte prática, foi criado um cenário de testes no qual foi implementado um SIEM com recurso a componentes *open-source:* Elastic Stack; Wazuh (HIDS); Search Guard e o Sentinl (GitHub).
2019	Dissertação de Mestrado	Faculty of Computing (Blekinge Institute of Technology)	Vamshi Devender, Sneha Adike,	Design and Performance of an Event Handling and Analysis Platform for vSGSN-MME event using the ELK stack	Com recurso à solução Elastic Stack implementam-se vários cenários de teste.
2019	Dissertação de Mestrado	Brno University of Technology (Faculty of Business and Management - Institute of Informatics)	Petr Bělousov	Security Enhancement Deploying Siem in a Small ISP Environment	Foi efetuada a comparação de desempenho entre as soluções ELK Stack e o Graylog. Implementou-se a solução ELK Stack na empresa Master Internet Inc.

Tendo em conta a documentação analisada, cuja informação recolhida serviu de base para a elaboração da Tabela 2.1, é de referir que as soluções SIEM *open-source* que foram implementadas em trabalhos académicos foram o Graylog e o Alienvault OSSIM. O Splunk não é um SIEM *open source*, contudo disponibiliza uma versão *freeware*, o Splunk Free (Educba, 2020) e, devido a este facto, também foi considerado pertinente incluí-lo no presente estudo. A solução *open-source* ELK Stack (renomeada para Elastic Stack) não é considerada uma solução SIEM, todavia, também foi considerada relevante devido às suas funcionalidades (rapidez, flexibilidade e escalabilidade). É de referir que, mesmo não sendo um SIEM, como se pode constatar na Tabela 2.1, existem exemplos de estudos que implementam SIEM com recurso ao ELK Stack/Elastic Stack, tais como Bělousov (2019), Marquina (2018) e até o relatório de Zarzosa (2017), que compara diversas soluções, entre as quais o ELK Stack/Elastic Stack.

Através da pesquisa documental, verificou-se que foram realizados vários trabalhos académicos sobre os SIEM e alguns desses trabalhos testaram o desempenho de soluções *open-source*. Contudo, não foi possível localizar um estudo em que seja testada a implementação de uma solução *open-source* recorrendo a um protótipo e que, em simultâneo, também sejam implementados mecanismos de cifragem, de minimização e de pseudonimização para garantir a conformidade com o RGPD.

2.11. Síntese

Devido ao RGPD é necessário que as Entidades repensem a forma como utilizam, gerem, armazenam e protegem os dados pessoais. Neste capítulo, enquadrou-se o Regulamento Geral de Proteção de Dados com as políticas de gestão da infraestrutura informática e apresentaram-se algumas das técnicas de salvaguarda de dados sugeridas nesse regulamento, tais como a pseudonimização e a anonimização. Resumidamente, o RGPD veio alterar a forma como é gerida uma infraestrutura informática de uma Organização, pois existe a obrigatoriedade de implementar procedimentos e medidas de segurança adicionais para garantir a conformidade legal de atuação dos sistemas de segurança.

Os *logs* de segurança permitem detetar tentativas de ataques e de exploração de vulnerabilidades, possibilitam rastrear e auditar as ações realizadas pelos utilizadores, pois contêm informações sobre quem realizou determinada tarefa, em que momento foi realizada, quem esteve envolvido e qual a sua origem. Contudo, os *logs* de segurança só são úteis quando se faz a gestão centralizada dos mesmos, pois dessa forma é possível obter informações relevantes para a gestão da infraestrutura informática.

Os SIEM e os gestores centralizados de *logs* recolhem e armazenam *logs*, contudo diferem entre si, pois os SIEM têm como objetivo garantir a segurança e oferecem um grande número de funcionalidades automatizadas. Em contrapartida, a implementação e manutenção de um SIEM é dispendiosa, complexa e necessita de constantes ajustes para responder às eventuais alterações ocorridas na Organização e, também, à constante evolução e complexidade dos ataques informáticos.

Os *logs* de segurança podem conter dados pessoais, como por exemplo os *cookies* ou os endereços IP, por essa razão foi necessário mudar o paradigma de implementação de um SIEM, para que seja possível assegurar a conformidade com o RGPD. Assim, na implementação de um SIEM é fundamental garantir a privacidade dos dados pessoais que se recolhe, adotando mecanismos de proteção para os defender, como é o caso a pseudonimização dos dados pessoais e do controlo de acessos.

Com recurso à análise documental, tendo como um dos objetivos o cumprimento dos requisitos exigidos no Regulamento Geral de Proteção de Dados, foi efetuado um levantamento das funcionalidades que um SIEM deve apresentar.

Quando se procedeu à análise da informação recolhida sobre os temas anteriormente abordados, verificou-se, no caso particular dos SIEM, que era muito reduzido o número de estudos disponíveis nos quais foi testada a implementação de um SIEM que garantisse a conformidade com o RGPG. Aliás, mais especificamente, em nenhum dos trabalhos analisados foi descrita/testada a implementação de um SIEM que, em simultâneo, também assegurasse a pseudonimização dos dados, de forma a garantir que a sua atuação está em conformidade com o RGPD. É de referir que este facto foi considerado importante para a estruturação deste estudo científico, uma vez que pode vir a ser um dos contributos para este tema.

Recorrendo à análise dos trabalhos académicos que implementam SIEM ou gestores de *logs* foi possível identificar quatro soluções: Graylog, o Alienvault OSSIM, o Elastic Stack e o Splunk Free.

Podemos considerar que não existe uma fórmula universal para combater os problemas de segurança e que, em simultâneo, salvaguarde a conformidade com o RGPD, todavia, a implementação de um SIEM ajustado à realidade de uma Organização pode contribuir para reduzir os incidentes de segurança informática e também o impacto que estes possam ter.

3. Soluções *Security Information and Event Management*

Tendo como base a análise documental do capítulo anterior foi possível identificar duas soluções SIEM *open-source* o Graylog e o Alienvault OSSIM (AT&T Cybersecurity). Além destes, foi também foi incluído o Elastic Stack, pois, pese embora não seja considerado um SIEM nativo, permite a implementação de um SIEM *open-source* com recurso a outras ferramentas, como foi exemplificado nos trabalhos académicos de Bělousov (2019), Marquina (2018). Também foi selecionado o Splunk Free, uma vez que, mesmo não sendo *open-source*, disponibiliza uma licença freeware.

No entanto, com o propósito de fundamentar a seleção dos SIEM a comparar no presente estudo, foi também efetuada uma pesquisa tendo em conta as funcionalidades e a respetiva posição no mercado. Na Figura 3-1 são apresentadas as soluções SIEM disponíveis no mercado, com recurso à utilização do Magic Quandrant, que classifica as soluções em quatro tipos: Líderes, Visionários, Nichos de Mercado e Desafiadores.

Segundo o relatório da Gartner de dezembro de 2017, as soluções líderes de mercado são as seguintes: Splunk, IBM[12], LogRhythm[13] e a McAfee[14]. Como soluções desafiadoras são apresentadas a Micro Focus[15] (ArcSight) e a Dell Technologies[16] (RSA). O mesmo relatório apresenta como soluções visionárias as seguintes: Rapid7[17], Securonix[18] e a Exabeam[19]. Finalmente, o referido documento identifica as soluções que encontraram um nicho de mercado: ManageEngine[20], BlackStratus[21], SolarWinds[22], Trustwave[23], EventTracker[24],

[12] https://www.ibm.com/pt-en
[13] https://logrhythm.com/
[14] https://www.mcafee.com/en-us/index.html
[15] https://www.microfocus.com/en-us/products/siem-security-information-event-management/overview
[16] https://www.rsa.com/en-us/customers/dell-technologies
[17] https://www.rapid7.com/
[18] https://www.securonix.com/
[19] https://www.exabeam.com/
[20] https://www.manageengine.com/
[21] https://www.blackstratus.com/
[22] https://www.solarwinds.com/pt/
[23] https://www.trustwave.com/en-us/
[24] https://www.netsurion.com/eventtracker-support

Fortinet[25], Venustech[26], FireEye[27], Micro Focus[28] (NetIQ) e a solução da AT&T Cybersecurity[29] (antigamente denominada de AlienVault). De todas as soluções analisadas, só o Splunk e a AT&T Cybersecurity disponibilizam opções de licenciamento *open-source ou* freeware e, como estas soluções já tinham sido referenciadas no capítulo anterior, foram selecionadas para serem testadas no âmbito do presente estudo.

Figura 3-1 Quadrante mágico aplicado à avaliação de soluções SIEM (LogRhythm, 2017)

Para além do que já foi referido sobre o Elastic Stack (antigamente denominado de *ELK Stack)* e segundo o Google Trends (Google Trends, 2018), em novembro de 2018, o Elastic Stack tem vindo a aumentar de popularidade, tendo-se tornado mais popular que o Splunk. No capítulo anterior também foi referenciado o Graylog, tendo-se optando pela sua inclusão no presente estudo, pois este é uma alternativa ao Elastic Stack, além de utilizar o Elasticsearch na sua arquitetura, que é um componente do Elastic Stack.

Reforçando o que anteriormente foi mencionado e utilizando a informação recolhida e analisada na reflexão que foi realizada no 2.10 deste trabalho sobre este tema, no qual foram descritas várias soluções SIEM, é possível listar como tendo licenças *open-source* ou

[25] https://www.fortinet.com/
[26] https://www.venusense.com/
[27] https://www.fireeye.com/
[28] https://www.microfocus.com/en-us/products/netiq/overview
[29] https://cybersecurity.att.com/

freeware as seguintes soluções: o Splunk Free, o OSSIM da AT&T Cybersecurity, o Elastic Stack e o Graylog. De seguida, vão ser tratados os resultados da pesquisa documental sobre as principais funcionalidades das soluções selecionadas, das quais elencamos: arquitetura, recolha de dados, resiliência, pesquisas e relatórios e alertas. Também serão mapeadas, para cada solução, as medidas técnicas identificadas no capítulo anterior relativamente ao RGPD, serão também testadas a facilidade de instalação e de utilização das soluções e monitorizado, em tempo real, um ataque de força bruta e, por fim, fundamentada a seleção da solução a utilizar no protótipo.

3.1. Splunk

O Splunk é uma plataforma que recolhe e armazena centralmente todos os *logs* do sistema informático de uma Organização (Baxter, 2018). O seu principal produto é o Splunk Enterprise, que é direcionado para uma grande variedade de profissões, tais como analistas, programadores ou gerentes (Contreras, Koelpin, Delgado, & Sigman, 2018). Contudo, o Splunk Enterprise foi construído tendo como base as operações realizadas no âmbito das Tecnologias da Informação e, por essa razão, de uma forma geral, é utilizado em operações de segurança, sendo que abrange também os dispositivos móveis e os IoTs (Contreras et al., 2018).

É de referir que a solução Splunk possui quatro soluções de licenciamento (Baxter, 2018; Splunk, 2019a): o Splunk Cloud, o Splunk Enterprise, o Splunk Light e o Splunk Free, sendo que as duas primeiras possuem, de uma forma geral, as mesmas funcionalidades, já o Splunk Light e o Splunk Free apresentam várias limitações.

O Splunk recolhe e indexa os *logs* em tempo real e, através da linguagem SPL (*Search Processing Language*), permite a criação de gráficos, relatórios, alertas e *dashboards*. Um dos principais objetivos desta ferramenta é o de auxiliar as atividades de uma Organização, através do diagnóstico de problemas, de métricas, de padrões ou utilizando *machine learning* para as atividades comerciais (Irace, 2018).

3.1.1. Arquitetura

Pode optar-se por implementar a solução Splunk num único servidor, no qual são processadas a indexação, a pesquisa de dados e todas as outras funcionalidades. É de referir que no ambiente distribuído existe a separação lógica entre a indexação e os cabeçalhos de

pesquisa, sendo que estas funções estão distribuídas, pelo menos, por duas máquinas servidoras. Na implementação mais simples de um ambiente distribuído existe um servidor indexador que recebe e indexa os dados e uma máquina separada do indexador, na qual estão os cabeçalhos pesquisa, sendo que estes comunicam entre si e realizam diferentes tarefas (Baxter, 2018).

Num ambiente cluster são combinados vários indexadores e/ou cabeçalhos de pesquisa num cluster principal de indexação e/ou de pesquisa de alta disponibilidade (caso o servidor fique inativo), tendo também redundância de dados (armazena mais que uma cópia de dados no cluster de indexação). Caso se pretenda uma recuperação a desastres ainda mais eficiente, pode-se criar um cluster *multiwebsite* no qual existem dois cabeçalhos de clusters indexados e/ou de pesquisa em diferentes locais físicos ou websites (Baxter, 2018).

Em suma, um ambiente distribuído não implica que exista necessariamente um cluster, mas a implementação de um cluster implica um ambiente distribuído, pois existem vários indexadores e/ou cabeçalhos de pesquisa que executam funções separadas (Baxter, 2018).

Como se pode visualizar na Figura 3-2, na implementação de um cluster é necessário um servidor de licenças para a gestão das licenças dos diversos componentes. Para a implementação de um cluster de indexação e/ou um cabeçalho de pesquisa vai ser necessário um servidor mestre para cada um destes elementos, permitindo que se possa distribuir os ficheiros de configuração e outros componentes necessários ao funcionamento dos mesmos (Baxter, 2018).

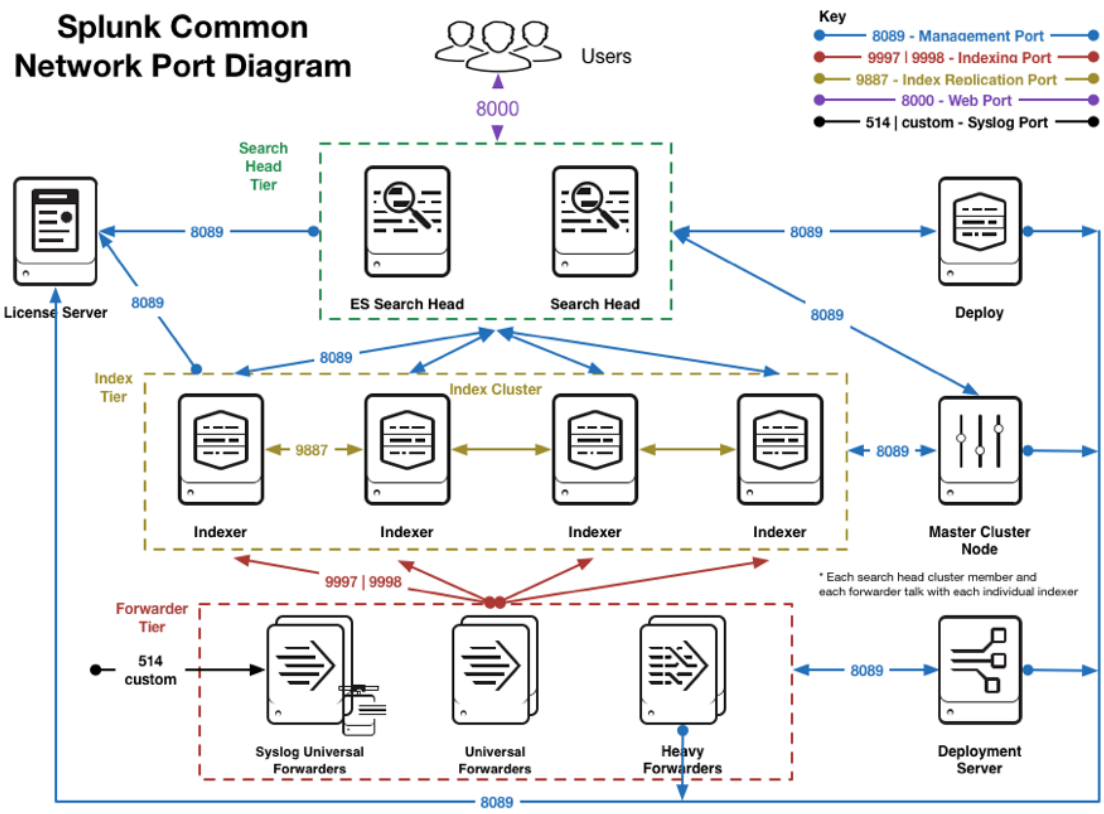


Figura 3-2 Arquitetura do Splunk (Splunk, 2016b)

No caso específico da versão Splunk Free, esta solução só permite a implementação de um servidor, apenas suporta um único utilizador e não permite a configuração de alertas ou a definição de perfis de utilizador. Contudo, o Splunk Free utiliza um indexador e um motor de pesquisa próprios (Varanda, 2019).

Como requisitos mínimos de hardware/software para um servidor, a solução recomenda que no mínimo se tenha 12 CPU cores a 2Ghz, 12 GB de RAM, um disco SSD (*Solid-State Drive*) com 1 TB, uma placa de rede a 1 GB e como sistemas operativos recomenda o Linux ou o Microsoft Windows a 64 bits (Splunk, 2020b).

Em síntese, caso se pretenda implementar a solução Splunk numa pequena Organização pode-se recorrer a um único servidor, todavia, na maior parte dos casos, é necessário implementar um cluster distribuído que forneça confiabilidade e escalabilidade.

3.1.2. Funcionalidades

O SIEM Splunk permite resolver problemas relacionados com a infraestrutura de rede, pesquisar e monitorizar o desempenho dos servidores, criar alertas e detetar ameaças de segurança. Através deste software, com recurso ao *machine learning*, é também possível detetar anomalias, analisar os comportamentos do utilizador e prever potenciais problemas (Baxter, 2018). A solução Splunk também permite pesquisar, analisar e visualizar um grande

volume de dados (Splunk, 2018b). De entre o elevado número de funcionalidades que o Splunk oferece, listamos as seguintes (Splunk, 2018b):

- **Recolher e Indexar** – Recolher os dados de qualquer origem, formato ou local, em tempo real;
- **Esquema Instantâneo** – Pode ser efetuado qualquer filtro aos dados que estão na base de dados da plataforma;
- **Cronologia de eventos baseada no tempo** – É determinada automaticamente a data e hora de qualquer evento;
- **Splunk *Search Processing Language* (SPL)** – Esta linguagem avançada de consulta permite a realização de pesquisas avançadas sobre os dados. Esta também permite cinco tipos de correlação, a saber: tempo, transações, subpesquisas, consultas e uniões;
- **Resultados interativos** – Podem ser formatados gráficos e tabelas em tempo real;
- **Amostragem de dados** – Permite que seja possível otimizar os dados em tempo real;
- **Machine Learning** – Através do recurso à análise integrada do Splunk ou com modelos próprios, é possível otimizar os recursos da Organização. É também possível serem criados os modelos com recurso à linguagem *Python*;
- **Monitorizar e Alertar** – É possível criar alertas para sinalizar situações críticas que, por sua vez, podem desencadear automaticamente várias operações;
- **Integridade dos dados** – O Splunk permite criar *hash* dos dados indexados para que seja possível garantir a integridade ao longo do tempo;
- **Escalável e alta disponibilidade** – Esta opção tem como base uma arquitetura distribuída com ajuste horizontal, para que se possam processar grandes volumes de dados;
- **Apps** – Existem várias apps certificadas para auxiliar nas tarefas de monitorização e de criação de relatórios e de alertas.

O Splunk permite monitorizar, analisar, pesquisar os dados das diferentes origens para que, posteriormente se consiga obter uma perspetiva do funcionamento do sistema informático (Splunk, 2018a).

3.1.3. Recolha de dados

O Através do Splunk é possível recolher e indexar dados de várias fontes: *logs* de servidores Web, aplicações, dispositivos de rede ou saídas de *scripts*. Consegue ainda

recolher e indexar dados complexos em tempo real (Diakun, Johnson, & Mock, 2014) ou previamente guardados. Para que seja possível indexar e relacionar a informação dos *logs* é necessário proceder à sua recolha, sendo que existem vários métodos para o fazer, como de seguida se elenca (Diakun et al., 2014; Sigman et al., 2017): indexando ficheiros e diretorias; obtendo os dados através das portas de rede; *Scripts*; *Modular inputs*; utilizando o *Universal Forwarder* para recolher os dados; fazendo o *upload* do ficheiro com as mensagens de *logs*; definindo os tipos de eventos e *tags*; e definindo filtros na recolha.

Na sua generalidade, os *forwarders* possuem as seguintes funcionalidades: segurança SSL, permitem a utilização de qualquer porta de rede disponível, fazem compressão de dados, permitem configurar o buffer e fazem a marcação (*tagging*) dos metadados (Zarzosa, 2017). O *Universal Forwarder* é uma versão gratuita do Splunk Enterprise que recolhe *logs* e os encaminha para o Splunk para serem indexados (Baxter, 2018).

Esta aplicação não exige configurações complexas para a recolha de *logs*, tornando a sua centralização simples e rápida. Para cada um dos tipos de recolha, são disponibilizados pela solução manuais exemplificativos das operações a efetuar.

3.1.4. Resiliência

Tendo em conta os resultados obtidos pelas pesquisas realizadas, a solução só faz referência à resiliência na descrição da solução Enterprise (Splunk, 2016c, 2017). Nos produtos Splunk Enterprise e Splunk Cloud são disponibilizadas as seguintes funcionalidades (Splunk, 2019d): *clustering*, pesquisa distribuída, alta disponibilidade e recuperação a desastres.

3.1.5. Pesquisa e relatórios

O Splunk permite que os utilizadores realizem pesquisas semelhantes às que são efetuadas no Google ou na pesquisa de ficheiros, mas também fornece mecanismos para efetuar pesquisas de outros tipos (Marlette, 2016). Aliás, a capacidade de pesquisa é uma das principais funções da solução Splunk e a que vai alimentar a maior parte das outras funcionalidades (Diakun, Johnson, & Mock, 2018). É de referir que esta solução possui uma linguagem de pesquisa própria, a SPL que é constituída por algumas centenas de comandos para a realização de pesquisas (Diakun et al., 2018).

Tendo em conta a enorme capacidade que o Splunk tem para receber *logs* de diversas fontes, torna-se necessário filtrar a informação que se pretende analisar. A possibilidade de filtrar os conteúdos é bastante importante, pois além de permitir que o gestor ou

administrador mais facilmente analise a informação, torna o sistema mais rápido e permite ainda a criação de alertas para determinados eventos.

As pesquisas na solução Splunk normalmente começam com uma pesquisa simples, que pode ser apurada através de vários comandos separados pelo caractere *pipe* (|). É de referir que a pesquisa é sequencial, ou seja, o resultado do comando à esquerda é utilizado como entrada para o próximo comando, que está colocado à direita do caractere *pipe* (Diakun et al., 2018).

Depois de ser criada uma pesquisa e verificada a utilidade do resultado, existem várias opções para guardar as pesquisas, tais como (Baxter, 2018): relatórios, alertas, tipos de eventos e *dashboards*. É possível ainda criar relatórios por intermédio das pesquisas ou através da ferramenta *Pivot* (Zarzosa, 2017).

Todas as soluções possuem as funcionalidades de pesquisa, *dashboard* e *reporting* e monitorização, contudo os alertas só estão disponíveis nas soluções pagas. O Splunk Light possui algumas restrições, tais como (Splunk, 2019d): tabelas, *data models* e os pivots.

3.1.6. Alertas

Os alertas podem ser utilizados para monitorizar e responder a eventos específicos, e são criados com recurso aos resultados de uma pesquisa que foi guardada e nos quais são procurados padrões nos *logs* em tempo real ou através de programação. Este recurso é acionado quando os resultados da pesquisa coincidem com as condições definidas, para responder aos alertas podem utilizar-se ações de alerta (Zarzosa, 2017).

A solução Splunk Free não possui as funcionalidades de monitorização e de alerta, sendo que estas só estão disponíveis nas seguintes soluções: Splunk Light, Splunk Enterprise e Splunk Cloud.

3.1.7. Pontos fortes e fracos

A instalação e configuração do Splunk é simples e fácil, sendo que o seu recurso mais importante é a monitorização de *logs* (Farmer, Crow, & Rodgers, 2019). Esta solução possui uma arquitetura modular que suporta diferentes APPs e é uma ferramenta útil para a análise forense, bem como para a gestão de *logs* (Gartner, 2018).

O Splunk disponibiliza funcionalidades importantes para a proteção dos dados sensíveis, o que vai contribuir para que possa atuar em conformidade com o RGPD, uma vez que suporta a ofuscação e pseudonimização ao nível do campo (Web3us, 2018).

É de referir que o suporte OT/IoT depende de terceiros (Web3us, 2018), o que pode aumentar os custos inerentes à implementação da solução. É um produto dispendioso (Gartner, 2018; Marquina, 2018) e, pese embora as pesquisas sejam eficientes, estas são complexas de realizar e exigem que o utilizador tenha conhecimentos avançados de base de dados (Gartner, 2018).

3.2. Elastic Stack

A solução Elastic Stack não é um SIEM (Marquina, 2018; Zarzosa, 2017), pelo menos não pode considerar-se que tem as mesmas características que as restantes soluções, todavia esta solução pode melhorar a segurança dos SIEM ou até, com recurso à articulação com outras soluções, tornar-se num (Zarzosa, 2017).

Os principais componentes da solução Elastic Stack são os seguintes (Elasticsearch, 2019h, 2019y; Settle, Paquette, Goldstein, & Kroh, 2019; Shukla, Kumar, Chhajed, & Ochoa, 2017): o Kibana, o Logstash, o Elasticsearch, os Beats, entre outros (Security, Alerting, Monitoring, Reporting, Graph, Machine Learning, Elasticsearch SQL, Canvas e o Elastic SIEM). Os componentes pagos, que anteriormente formavam o X-Pack, são os seguintes (Elasticsearch, 2019i): *Security*, *Alerting*, *Monitoring*, *Reporting*, *Machine Learning*, *Graph*, e o Elasticsearch SQL. É de referir que a partir do Elastic Stack 6.8 e 7.1, estão disponíveis na licença Basic os recursos de segurança, tais como a comunicação encriptada por TLS, a criação e gestão de utilizadores, o RBAC (*Role-Based Access Control*) que protege o acesso ao nível do índice e do cluster e também o *Spaces*, para proteger o Kibana (Kearnsh, 2019; Kumar, 2019). Com estas funcionalidades é possível implementar um cluster seguro, sendo que no Kibana foram disponibilizadas as funcionalidades *Spaces* e o *multi-tenancy* (Kearnsh, 2019). Todavia, ainda existem funcionalidades de segurança que só estão disponíveis através da assinatura Gold, como é o caso da autenticação *Active Directory*/LDAP ou da segurança ao nível do campo e do documento (Kearnsh, 2019).

O termo ELK Stack resulta do acrónimo dos três principais produtos *open-source*: o Elasticsearch, o Logstash e o Kibana, tendo sido posteriormente foram adicionados os Beats e a solução foi renomeada para Elastic Stack. Os componentes que constituem a solução

Elastic Stack estão descritos com mais detalhe no Anexo C - Componentes do Elastic Stack. É de salientar que esta solução é amplamente utilizada como uma plataforma de gestão de *logs* (Daubner, 2018).

Na versão 7.2 do Elastic Stack foi lançada a App Elastic SIEM, que tem como objetivo a construção da visão da empresa Elasticsearch sobre um SIEM. É possível, com o Elastic SIEM, a realização de pesquisas de alerta ou a deteção interativa de ameaças, com recurso à análise dos eventos de segurança relacionados com os postos de trabalho e com os equipamentos de rede (inclui suporte para as *firewalls*: Cisco ASA e Palo Alto). O ECS (*Elastic Common Schema*) é outra funcionalidade importante da versão 7.2 do Elastic Stack, pois vai facilitar a normalização de dados de origens distintas, permitindo também a correlação, pesquisa e análise entre as várias fontes (Paquette, 2019).

O Elastic Stack tem tido uma grande adesão e, devido a esse facto, possui inúmeros plugins e extensões para diversos produtos (Daubner, 2018). Existem várias categorias de licenças para esta solução, que de seguida são enumeradas (Daubner, 2018; Elasticsearch, 2019i, 2019r): *Open-source* (solução de código aberto), Basic, Gold, Platinum e Enterprise. A solução Basic que é *open-source* e que disponibiliza algumas das funcionalidades básicas do X-Pack (Elasticsearch, 2019r), foi, entretanto, descontinuado e separado pelas funcionalidades descritas anteriormente. Existem outras modalidades de produtos que a solução disponibiliza, mas destacamos o serviço Cloud como uma solução SaaS - Software as a Service (Daubner, 2018; Elasticsearch, 2019j): Elasticsearch Service, Elastic App Search Service e Elastic Site Search Service.

O Elastic Stack oferece diversos recursos de processamento de dados, é flexível e suporta uma ampla gama de formatos, por exemplo, caso se pretenda enviar dados para o Elasticsearch, a saída deve estar no formato JSON (*Javascript Object Notation*), uma vez que este utiliza o JSON para armazenar os documentos (Zarzosa, 2017).

3.2.1. Arquitetura

Como se pode observar na Figura 3-3, a arquitetura genérica do Elastic Stack é constituída pelos Beats, Logstash, Elasticsearch e o Kibana. Os Beats são responsáveis pelo envio de dados para o Elasticsearch e para o Logstash, sendo que, posteriormente, se podem visualizar os dados do Elasticsearch no Kibana.

Como já foi referido anteriormente, o Elasticsearch permite a realização da análise de dados em tempo real, tendo como ponto de partida as diversas origens dos dados, é escalável e possibilita a realização de pesquisas de "texto completo", por sua vez o Kibana tem como principal função a de permitir a visualização dos dados do Elasticsearch (Srivastava, 2019). Através do Kibana também é possível efetuar diversos tipos de pesquisas e visualizar os dados em vários formatos (Srivastava, 2019). É de referir que o Logstash recolhe, analisa, aprimora e transforma os dados e depois envia-os para o Elasticsearch (Zarzosa, 2017).

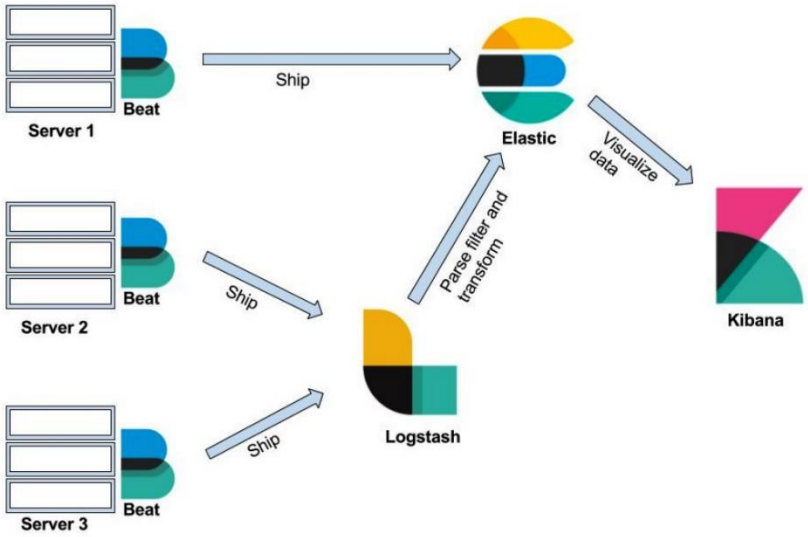


Figura 3-3 Arquitetura da solução Elastic Stack (Srivastava, 2019, p. 9)

Esta arquitetura é simples e fácil de integrar com outros mecanismos de recolha de dados e de integrar no Elastic Stack (Elasticsearch, 2019g).

Para implementar o Elastic Stack o hardware deve ter como requisitos mínimos os seguintes: 8 CPU cores para cada nó, 16 a 32 GB de RAM e um disco SSD, cujo tamanho depende do volume de *logs* a tratar (Sebastian Herzberg, 2015; Shukla & Kumar, 2019; Shukla et al., 2017). Como software, a solução recomenda o sistema operativo Linux, pese embora também seja compatível com os sistemas operativos Microsoft Windows (Shukla & Kumar, 2019).

O Elastic Stack é utilizado em vários contextos, desde a gestão de *logs*, a análise de métricas, as pesquisas analíticas e de aplicação (Elasticsearch, 2019g), por essa razão, cada Organização deve adaptar o esquema anterior à sua realidade.

3.2.2. Funcionalidades

As funcionalidades do Elastic Stack que são descritas nesta secção refletem as funcionalidades elencadas nos pontos anteriores para cada componente, das quais destacamos as seguintes (Collier & Azarmi, 2019; E-Cogni Treinamentos, 2017; Senanayaka, 2018): apresenta um interface intuitivo e personalizável; disponibiliza bibliotecas para múltiplas linguagens de programação; permite a compatibilidade com várias linguagens clientes, entre as quais o Ruby; o Python, o PHP, o Perl, o .NET, o Java e o JavaScript; possibilita a análise de dados em tempo real; proporciona a escalabilidade com alta disponibilidade; permite a realização de pesquisas de texto completo; fornece funcionalidades de Monitorização, de Segurança, de Alertas, de Relatórios, *Canvas*, ECS, Elastic SIEM, e de *Machine Learning*.

É de salientar que o Elastic Stack é um produto de código aberto altamente personalizável e, como já foi referido anteriormente, um dos mais populares para a gestão e monitorização de *logs* (Daubner, 2018).

3.2.3. Recolha de dados

A recolha de dados no Elastic Stack é efetuada pelos Beats, e, como já foi descrito anteriormente, estes conseguem recolher dados de diversas fontes (Dixit, 2017). Os Beats são ferramentas que não necessitam de muitos recursos para serem eficientes e que podem enviar dados de centenas ou milhares de máquinas para o Logstash ou para o Elasticsearch (Dixit, 2017). O Elasticsearch utiliza o formato JSON, deste modo a saída dos dados dos Beats deve ser sempre em JSON (Zarzosa, 2017).

Os Beats são disponibilizados em diferentes aplicações (o Filebeat, o Metricbeat, o Winlogbeat, o Auditbeat, o Heartbeat, o Packetbeat e o Functionbeat) que permitem a recolha de diversificados tipos de dados. Para além dos Beats fornecidos pela empresa, a comunidade Beats disponibiliza inúmeros Beats para diversas funções, sendo que estes também são de código aberto (Elasticsearch, 2019f). Caso os Beats disponibilizados não possuam as funções pretendidas, é possível desenvolver o próprio Beats e contribuir para a comunidade (Dixit, 2017).

3.2.4. Resiliência

Para a solução *open-source*, são disponibilizadas um conjunto de funcionalidades que permitem a escalabilidade e a resiliência, tais como (Elasticsearch, 2019i): *clustering*, alta

disponibilidade e balanceamento automático dos dados; e a pesquisa entre clusters. A replicação entre clusters só está disponível nas versões pagas (Platinum e Enterprise).

Caso se utilize o Filebeat ou o Winlogbeat para a recolha dos *logs*, a entrega dos dados é efetuada pelo menos uma vez, pois tem de ser confirmada a receção dos. O fluxo de dados entre estes dois Beats e o Logstash ou entre o Logstash e o Elasticsearch são síncronos e aceitam confirmações (Elasticsearch, 2019g).

As filas persistentes do Logstash proporcionam a proteção entre as falhas dos nós. Para que se garanta a resiliência a nível do disco local, deve ser configurado o RAID (*Redundant Array of Independent Disks*) no mesmo (Elasticsearch, 2019g).

O Elasticsearch consegue detetar falhas no funcionamento e, através da replicação entre clusters, pode-se recorrer a um cluster secundário como backup ativo, possibilitando que os seus dados estejam seguros e disponíveis (Elasticsearch, 2019k). Este componente permite que sejam efetuadas cópias dos *shard*s dos índices, permitindo a alta disponibilidade caso um nó falhe, pois estes nunca são guardados no mesmo local do que o original (Elasticsearch, 2019d).

Na versão 7.0 do Elasticsearch, a Elastic recriou completamente a camada de coordenação de cluster (*Zen Discovery*) com o intuito de a tornar mais rápida, mais segura e mais fácil de utilizar. Além disso, foram implementadas várias alterações que reduzem a probabilidade de erro humano e que fornecem opções mais claras para recuperação das falhas. Outra melhoria apresentada na versão 7.0 do Elastic Stack é a introdução de um disjuntor de memória real, que deteta com muito mais precisão as solicitações realizadas num nó e que impede que as mesmas tornem o nó instável (Wiltshire, 2019).

3.2.5. Pesquisa e relatórios

O Elasticsearch executa pesquisas quase em tempo real, onde a linguagem da consulta é baseada na estrutura JSON e é muito flexível. Quando se executa uma consulta no Elasticsearch, cada documento recebe uma pontuação que indica a sua relevância de acordo com a pesquisa realizada (Andhavarapu, 2017).

As consultas no Elasticsearch podem ser estruturadas ou de "*texto completo*", nas pesquisas estruturadas pesquisam-se números, datas, nomes e no segundo tipo de pesquisa

o Elasticsearch localiza todos os documentos que correspondem à pesquisa e de seguida classifica-os com base na relevância (Andhavarapu, 2017).

Existe um grande número de APIs disponíveis para gerir o Elasticsearch, estas ajudam a gerir os clusters, índices e pesquisas (Gupta & Gupta, 2017), são o exemplo (Elasticsearch, 2019l): *Document APIs*, *Search APIs*, *Indices APIs*, *cat APIs*, e *Cluster APIs*.

O Query DSL (*Domain Specific Language*) é umas das funcionalidades mais poderosas do Elasticsearch e permite que uma consulta utilize filtros, consultas, grupos e que ordene os resultados (Andhavarapu, 2017).

A *framework aggregations* é uma parte importante do Elasticsearch, sendo que esta *framework* permite criar conjuntos e gerar informações analíticas sobre o resultado de uma pesquisa (Gupta & Gupta, 2017). As agregações, outras das funcionalidades do Elasticsearch, permitem obter uma melhor perceção dos dados, por exemplo, num índice de uma biblioteca, é possível saber quantos livros existem para um determinado ano ou a média de livros por ano (Gupta & Gupta, 2017). As agregações podem ser divididas em quatro famílias, a saber (Elasticsearch, 2019a): *Bucketing* (cria blocos de documentos com base na pesquisa), *Metric* (calcula as métricas num conjunto de documentos), *Matrix* (o resultado da agregação é uma matriz) e Pipeline (agrega o resultado de outras agregações e também as métricas associadas).

Os relatórios são uma funcionalidade paga do Kibana, sendo que é possível criar relatórios a partir de uma pesquisa efetuada no Kibana ou num *dashboard*, podendo ser partilhados automaticamente com outras pessoas (seja online ou offline). Os relatórios estão otimizados para impressão, são personalizáveis (permite que seja adicionado o logótipo da empresa, entre outras formatações), permitem a formatação em PDF e podem ser exportados para CSV (*Comma Separated Values*). Os relatórios podem ser combinados com os alertas ou podem ser programados para envio do resumo mensal, semanal ou quinzenal. Podem ainda ser acionados quando uma condição for alcançada, como é o caso da ocorrência de um determinado número de erros num período de tempo específico (Elasticsearch, 2019w).

3.2.6. Alertas

A funcionalidade *Alert* para o Elasticsearch e para o Kibana é paga (Elasticsearch, 2019r), todavia é possível, através desta funcionalidade, saber se a utilização do CPU está a crescer inesperadamente, se o tempo de resposta de uma aplicação está a aumentar, se estão a surgir

muitos erros 503, ou se a taxa de indexação do Elasticsearch decresceu (Elasticsearch, 2019b).

É disponibilizada uma escolha variada de opções de alertas, sendo que estas podem ser integradas por email, PagerDuty[30] ou Slack[31] com a infraestrutura de monitoramento ou com qualquer sistema terceiro. É possível gerir todos os alertas a partir de uma única interface, podendo tomar conhecimento dos alertas que estão a ser executados e quais as ações que foram tomadas (Elasticsearch, 2019b).

Os alertas podem ser combinados com o *machine learning* para que possa ser possível aferir comportamentos incomuns de utilizadores e de aplicações. Outra funcionalidade importante é a de permitir a consulta do histórico de alertas (todas as execuções dos alertas podem ser indexadas no Elasticsearch), podendo ser visualizadas as ações realizadas, quais os alertas que estão a ser executados ou quantas vezes as condições definidas estão a ser atingidas (Elasticsearch, 2019b).

O componente *Wacher* do Kibana permite que sejam definidas as regras para que sejam criados alertas caso determinadas condições sejam atingidas, podem-se criar alertas com base nas palavras-chave das pesquisas de um utilizador ou monitorar a infraestrutura através da pesquisa de vários parâmetros, como é o caso a utilização da memória ou do disco (Gupta & Gupta, 2017).

3.2.7. Pontos fortes e fracos

O Elastic Stack é uma solução de código aberto que tem associado uma forte comunidade de programadores e de utilizadores (Daubner, 2018), além disso, é fácil de instalar, é escalável e disponibiliza bibliotecas para várias linguagens de programação (Senanayaka, 2018). O processamento é independente da origem, formato ou esquema dos dados (Senanayaka, 2018).

Devido à sua flexibilidade, esta solução permite um grande número de melhorias, alterações e modificações ao produto, tornando possível dar resposta a uma Organização em expansão (Corcoran, 2017).

[30] https://www.pagerduty.com/
[31] https://slack.com/intl/en-pt/

Atendendo ao facto que algumas ameaças podem ser incubadas durante meses sem serem detetadas, a retenção dos dados a longo prazo é muito importante. Neste âmbito, esta solução permite a retenção dos dados pelo tempo que for considerado necessário, o que vai possibilitar que o histórico de *logs* seja rápido e simples de gerir (Elasticsearch, 2019y). O Elastic Stack pode tornar-se parte integrante do fluxo de trabalho da cibersegurança, pois possui recursos que contribuem para a mesma, dos quais destacamos (Corcoran, 2017):

- **Controle do conteúdo** –O Elastic Stack proporciona uma solução flexível que pode acompanhar a evolução da Organização, permitindo que se possam adicionar novos atributos de dados aos documentos e, mesmo assim, garantindo as hiperligações para o conteúdo antigo.
- **Visualização** – possibilita a análise dos dados em séries temporais, o que é particularmente importante para a análise de segurança, pois revelam como as métricas mudam ao longo do tempo;
- **Comunidade** – o Elastic Stack possui uma forte comunidade que disponibiliza inúmeras ferramentas, tutoriais e cursos, por exemplo o SANS Institute[32] leciona um curso de análise de segurança recorrendo ao Elastic Stack.

Devido aos seus atributos, esta solução e os seus produtos são utilizados por enumeras empresas, tais como (Elasticsearch, 2019z): eBay[33], Facebook[34], Cisco[35], Microsoft[36], Slack, entre muitas outras.

Os dados de uma Organização necessitam de ser recolhidos, processados, normalizados, aprimorados e armazenados, sendo que estas etapas são definidas através do conceito "gestão de *logs"*, e este é um componente essencial do SIEM. O Elastic Stack é uma das soluções mais populares de gestão de *logs* e é parte integrante dos SIEM de código aberto (Wazuh e Apache Metron[37]) e também de SIEM pagos, como é o caso do SIEMonster[38] (Berman, 2018).

Relativamente aos pontos fracos, é de referir que o Elastic Stack não fornece, por defeito, a análise de riscos (Zarzosa, 2017), não fornece regras para a correlação de eventos e também

[32] https://www.sans.org/soc
[33] https://www.ebay.com/
[34] https://pt-pt.facebook.com/
[35] https://www.netacad.com/
[36] https://www.microsoft.com/pt-pt
[37] https://metron.apache.org/
[38] https://siemonster.com/

não possibilita a gestão de incidentes (Berman, 2018). Devido a este facto, o Elastic Stack não pode ser utilizado como um SIEM sem que se recorra a outras plataformas e serviços, o que pode aumentar os custos da sua implementação (Berman, 2018). Todavia, como já foi referido anteriormente, com a versão 7.2 do Elastic Stack foi lançado o Elastic SIEM, que possui funcionalidades que permitem que os analistas de segurança protejam a sua Organização. É verosímil pensar que esta funcionalidade será melhorada em versões futuras permitindo a implementação de regras de deteção e também integração de inteligência de ameaças (Paquette, 2019; Settle et al., 2019). É de referir que a versão 7.2 também disponibiliza novos recursos para os Beats, facilitando a recolha dos *logs* de segurança (Settle et al., 2019).

Caso existam grandes volumes de dados e seja necessário garantir que não existe perda de dados, pode ser indispensável o recurso a outras ferramentas (Berman, 2018). O Kafka[39], o Redis[40] ou o RabbitMQ[41] normalmente são implementados entre os Beats e o Logstash para garantir que não existem perdas de dados (Berman, 2018). O Elastic Stack pode integrar inúmeros componentes distintos, o que pode implicar uma árdua gestão da solução (Senanayaka, 2018).

Outro ponto que pode ser considerado uma desvantagem é o facto de que muitos dos recursos importantes são pagos, como é o caso dos alertas, dos relatórios, das funcionalidades avançadas de segurança e também de algumas funcionalidades de monitorização (Daubner, 2018). Por outro lado, devido ao forte suporte da comunidade existem inúmeras extensões de código aberto que podem resolver o problema, todavia esta possibilidade também tem desvantagens (Daubner, 2018): vai aumentar a complexidade e a as extensões podem não oferecer atualizações ao mesmo tempo que a solução disponibiliza uma nova versão.

Além disso, não é possível relacionar os dados com o seu contexto de uma forma nativa, sendo que o contexto é importante na deteção efetiva de ameaças, por exemplo, um funcionário da equipe de vendas que aceda ao servidor da contabilidade vai provocar um alerta caso este não tenha permissões para aceder ao mesmo (Berman, 2018).

[39] https://kafka.apache.org/
[40] https://redis.io/
[41] https://www.rabbitmq.com/

Para se obter escalabilidade, esta solução necessita que seja feito uma otimização das suas configurações, o que por vezes pode ser um processo de tentativa erro (Borkar, 2018). Por último, sublinhamos que curva de aprendizagem da solução é árdua e requer uma gestão intensa, mas depois de ser ultrapassada a curva de aprendizagem consegue-se implementar uma solução robusta para a gestão de *logs* (Tal, 2018).

3.3. Graylog

O Graylog é um SIEM de código livre que tem como principais potencialidades a facilidade de utilização e a escalabilidade, o que possibilita a gestão de grandes quantidades de *logs*. É uma ferramenta poderosa de análise de *logs*, através da qual se podem obter dados importantes, e, na interface web da solução, podem-se pesquisar e visualizar os dados armazenados pelo Graylog (Daubner, 2018).

Esta solução foi escrita na linguagem de programação Java e utiliza algumas tecnologias de código aberto, tais como o Elasticsearch e o MongoDB (Daubner, 2018). Disponibiliza três tipos de licenças, que se seguida se elencam: *Open-source*, a Free Enterprise (com as funcionalidades Enterprise, mas limitada a 5GB por dia) e a Enterprise (Graylog, 2018f).

O processamento de *logs* é efetuado através de *Streams*, sendo que estes podem ser agrupados em grupos virtuais de *logs*, cuja categorização pode ser efetuadas de acordo com regras específicas. A título exemplificativo, podem-se agrupar *logs* pelo nível de gravidade, pelo endereço IP ou pela origem. As *streams* suportam dois tipos de regras o And lógico ou OU lógico, no qual o componente *Message Filter Chain* é um sistema de pipelines responsável pela análise dos *logs*, por definir quais são os campos estáticos e por atribuir os *logs* às *streams* apropriadas. O *Extractor* é um elemento do *Message Filter Chain* que é utilizado para a análise e extração dos campos estáticos de um *log*, sendo que para cada formato de *log* existe um *Extractor* diferente (Simko, 2018).

Os *Index Sets* controlam a forma como os *logs* são armazenados no servidor Graylog, definem também as políticas de rotação e de retenção de dados e configuram a forma de armazenar os dados no Elasticsearch (Simko, 2018). Como os *Index Sets* são independentes entre si, estes podem ser configurados de maneira diferenciada (Simko, 2018) e é espectável que um *Index Set* que agrupe os dados pessoais possa ter uma política de retenção muito baixa.

O Graylog permite agregar dados de várias origens, efetuar pesquisas utilizando vários parâmetros, analisar, visualizar e produzir vários relatórios de forma centralizada, sem ser necessário formação especializada dos técnicos. Esta solução SIEM possibilita a realização de pesquisas sem que esteja definido um objetivo rígido, potenciando assim a deteção de ameaças, porque os resultados das pesquisas podem posteriormente ser expandidos ou destacados, caso se pretenda perceber a extensão ou amplitude do incidente/problema (Graylog, 2018c).

3.3.1. Arquitetura

A solução Graylog é eficiente na gestão de *logs* porque a sua arquitetura foi concebida tendo como objetivo principal o de realizar a sua gestão. Para que uma solução seja eficiente como Gestor de *Logs*, é fundamental que permita a implementação de uma arquitetura específica, pois a gestão de *logs* não é apenas um mecanismo de pesquisa de texto completo, uma vez que as pesquisas/métricas estão veiculadas a um período de tempo (Graylog, 2018e).

Através da sua arquitetura, o Graylog permite a criação de um nível de abstração, o que vai facilitar o acesso aos dados, pois não é necessário a criação de índices ou a aplicação de escrever filtros (Graylog, 2018e). Como se pode visualizar na Figura 3-4, a arquitetura da solução Graylog é composta por três componentes principais: Graylog, MongoDB e Elasticsearch. A solução pode recolher os *logs* através de HTTP(S), do *Syslog*, do GELF (*Graylog Extended Log Format*) e dos Beats (Graylog, 2018e).

Os Beats permitem uma comunicação segura no Graylog, uma vez que são autenticados por certificado, o que significa só possível o envio de mensagens de origens autenticadas para o Graylog (Graylog, 2018e). O *Syslog* é muito utilizado para registar os dados dos equipamentos de rede e das máquinas, todavia, caso seja necessário que um *log* possua informação mais detalhada, será mais adequada a utilização do GELF (Graylog, 2018e). O GELF não apresenta algumas das lacunas do *Syslog*, como exemplo, não tem a limitação máxima a 1024 bytes, não tem inúmeros RFCs (*Request for Comments*) e possui um tipo de dados estruturado (Graylog, 2018e). É utilizado para fazer o login em aplicações e, devido à sua estrutura, não é necessário ter uma preocupação com os tempos de espera, com problemas de ligação ou com outro tipo de interrupções, pois o GELF pode ser enviado por UDP (Graylog, 2018e). Quando existem múltiplos servidores Graylog, o componente

LoadBalancer é essencial, uma vez que assegura o roteamento das mensagens, o que possibilita uma configuração altamente fiável (Graylog, 2018e).

O Graylog disponibiliza uma interface web (Graylog UI) para os seus utilizadores (Graylog, 2018e), na qual é possível visualizar, pesquisar e analisar os dados (Simko, 2018). O Graylog UI obtém os dados através da *API REST Graylog* e dos protocolos HTTP(S), sendo que a API é utilizada como o principal canal de comunicação entre a interface disponibilizada para o utilizador e o servidor Graylog. Através da *API REST Graylog* é possível desenhar um *frontend* de acordo com os objetivos pretendidos (Simko, 2018). A principal função do Server é a de comunicar com todos os outros componentes, contudo também recebe os dados dos clientes (Simko, 2018).

Nesta solução, todas as mensagens são armazenadas no Elasticsearch, já o componente Mongo DB assegura o armazenamento das configurações e dos metadados (Graylog, 2018a). Nesta solução é utilizado um cluster dedicado do Elasticsearch que pode ser constituído por vários nós (Simko, 2018). É de referir que o MongoDB é uma base de dados que armazena os dados NoSQL (Not Only SQL) numa estrutura flexível semelhante à dos documentos JSON (Simko, 2018). No Graylog, como já foi referido, o MongoDB só arquiva metadados e configurações, como por exemplo, os utilizadores, as permissões e os índices (Simko, 2018).

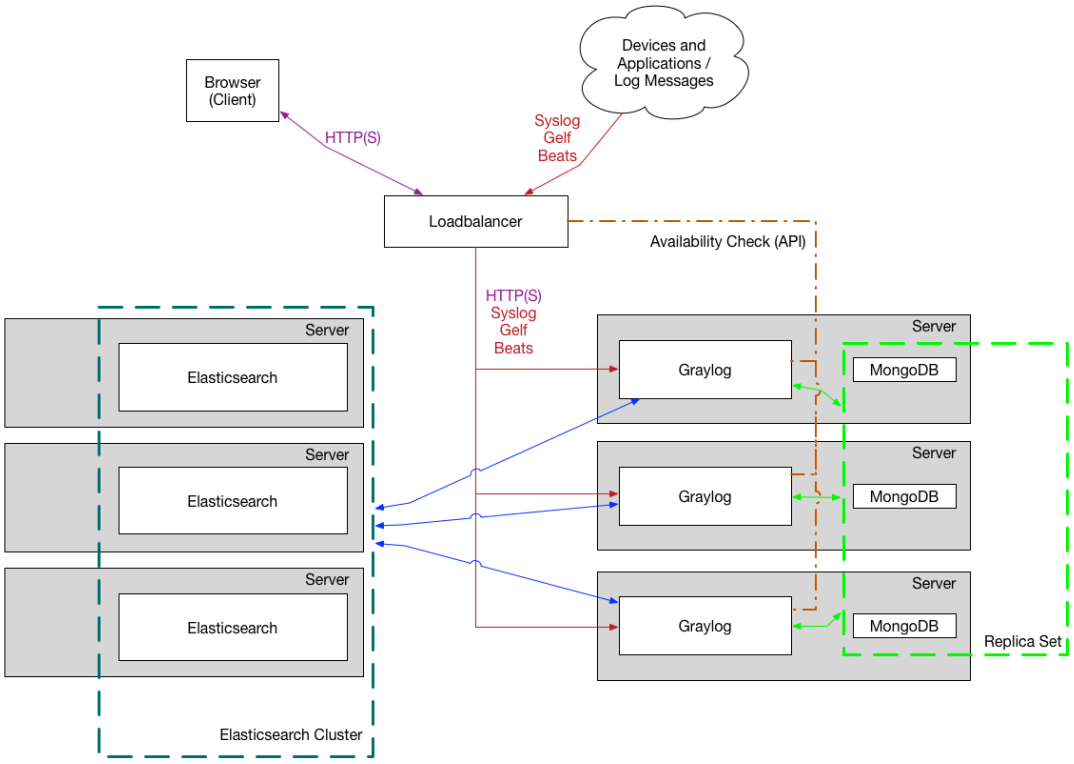


Figura 3-4 Arquitetura do Graylog (Graylog, 2018a)

Para a solução Graylog, na avaliação do produto, é recomendado que se instalem todos os componentes no mesmo servidor, no entanto, para a implementação em produção, aconselha-se que os componentes sejam implementados em servidores diferentes, para melhorar o desempenho (Graylog, 2018e). As vantagens de se optar por uma configuração menos complicada no Graylog é a simplicidade e rapidez da instalação, no entanto, esta implementação não é escalável e não é confiável em caso de falha, uma vez que todos os componentes estão num servidor, assim, a confiabilidade aumenta quando a solução possui vários nós, no entanto, a manutenção e a implementação tornam-se mais complicadas (Daubner, 2018).

Como requisitos mínimos de hardware, são recomendados pela solução os seguintes: 4 CPU cores, 8 GB de RAM e discos SSD (Graylog, 2020b); recomenda as distribuições Linux como sistema operativo (Graylog, 2020c), não sendo possível instalar o Graylog nos sistemas operativos Microsoft Windows (Graylog, 2020d).

3.3.2. Funcionalidades

Como Nas funcionalidades será de destacar no Graylog a capacidade de gestão um grande volume de dados com rapidez e flexibilidade, facilidade de utilização, resiliência, formato próprio de *logs* (GELF) oferece um grande número de plugins, além dessas, elencamos outras funcionalidades que também consideramos importantes (Graylog, 2018c, 2019; Graylog Marketplace, 2019; Simko, 2018):

- ***Multithread* e pesquisa distribuída** – permite a realização de várias pesquisas distribuídas em simultâneo;
- **Eficiência na análise de dados** – através de *dashboards*, disponibiliza uma grande variedade de métricas e de tendências num único local;
- ***Content Packs*** – são coleções que partilham configurações dos dispositivos mais populares, simplificando a sua implementação;
- ***Logs* de auditoria de utilizador** – possibilita o rastreamento das atividades dos utilizadores;
- ***Offline Log Archival*** – permite armazenar os dados mais antigos num dispositivo de armazenamento externo e, caso sejam necessários, o Graylog pode importá-los e possibilita que fiquem disponíveis para pesquisa.

É de referir que o design, a escalabilidade e a flexibilidade do Graylog garantem que, com uma infraestrutura mais económica, se possam efetuar análise de dados de uma forma célere e num ambiente fácil de gerir (Graylog, 2018c).

3.3.3. Recolha de dados

Como já foi referido anteriormente, um *log* é recolhido pelo servidor Graylog e depois é processado pelo *Message Filter Chain*, este é um processador de mensagens responsável por analisar, filtrar e configurar campos estáticos de um determinado *log.* Os *logs* são filtrados de acordo com as regras predefinidas e roteados de acordo com determinadas categorias (*Streams*), ou seja, para cada *Stream* é possível definir regras individuais e estruturar um conjunto de índices. Através do *Index Set* é possível especificar como as mensagens são armazenadas no Elasticsearch, definir o número de *shards* e a política de retenção dos dados (Simko, 2018).

O Graylog suporta três tipos de fontes de dados de entrada (Simko, 2018):

- **Protocolos e formatos padrão** – o *Syslog* é o protocolo mais utilizado para o registo de diferentes tipos de logs e é suportado por uma ampla variedade de dispositivos. Os *logs* também podem ser gerados pelo *Rsyslog* ou pelo *Syslog-ng*. O Graylog também suporta *logs* em texto simples ou em formato JSON, os protocolos UDP e TCP e as filas de transporte do Apache Kafka, RabbitMQ.
- **Conectores de terceiros** – o Graylog suporta um sistema chamado *Graylog Collector Sidecar*, que é um serviço/daemon para os sistemas operativos Microsoft Windows e Linux que é utilizado como conector de *logs*. O conector pode utilizar como agentes o NXLog[42], Filebeat ou o Winlogbeat e encaminhar os *logs* recolhidos para o endereço IP/porta do servidor Graylog.
- **GELF –** é o formato dos *logs* do Graylog, que consiste numa *string* JSON que é utilizada especialmente para encaminhar e processar os *logs* das aplicações. Pode suportar várias linguagens de programação e é capaz de registar todas as exceções criadas por uma determinada aplicação, fornece ainda uma estrutura otimizada e que possibilita a compressão de dados**.**

[42] https://nxlog.co/

Em suma, o Graylog possui o seu próprio formato de *logs*, o GELF, mas pode recolher *logs* com vários tipos de formatos, como é o caso do *syslog* ou o *rsyslog*.

3.3.4. Resiliência

O Graylog foi criado especificamente para a recolha, gestão e análise de *logs*, logo é escalável e pode processar grandes volumes de dados sem a ocorrência de problemas. O seu mecanismo de armazenamento permite guardar dados encriptados, para que seja possível movê-los de forma segura para outros destinos também seguros (Graylog, 2018f).

Na licença Enterprise, a funcionalidade *archiving* permite restaurar os *logs* ou, caso não se tenha um licenciamento Enterprise, podem ser utilizadas as funcionalidades do Elasticsearch para restaurar os nós (*Snapshot* e *Restore*). As configurações do Graylog são fáceis de realizar, basta fazer um backup das configurações que estão armazenadas no MongoDB e dos ficheiros de configuração do sistema operativo (Graylog, 2018e).

A funcionalidade *Journal* evita a perda de dados na eventualidade de ocorrer uma interrupção da rede ou uma falha nos *Index Sets*. É possível replicar os dados sem que seja necessário recorrer a componentes adicionais, garantido que não existe perda de dados nos períodos em que a infraestrutura atinge picos incomuns de processamento de dados (Graylog, 2018c).

3.3.5. Pesquisa e relatórios

As pesquisas no Graylog são muito semelhantes à sintaxe do *Lucene*, ou seja, por norma todos os campos da mensagem são incluídos na pesquisa, caso não se especifique o campo ou campos a procurar. A solução permite filtrar os dados através de um ou de vários campos, por um ou por mais termos ou, até, por uma expressão exatamente igual à que foi inserida pelo utilizador. Os *Wildcards* são um tipo de filtro através do qual é possível substituir nos termos de pesquisa vários, um ou nenhum caractere (este recurso requere muita memória, por isso é necessário ativá-lo no ficheiro de configuração). Também é possível realizar pesquisas por período temporal, ou seja, podemos selecionar um período de tempo absoluto, relativo ou por palavras-chave (Graylog, 2018e).

As pesquisas efetuadas pelos utilizadores podem ser guardadas para posteriores utilizações, sendo que os resultados da pesquisa são apresentados na forma de um histograma, no qual é possível visualizar o número de *logs* agrupados num determinado período de tempo (Graylog, 2018e).

Existem várias ferramentas que permitem a análise dos resultados das pesquisas, por exemplo, é possível guardar os resultados das pesquisas em *dashboards* para que possam ser analisadas ao longo do tempo. É de referir que as informações estatísticas podem apresentar as seguintes operações: o total, a média, o mínimo, o máximo, o desvio padrão, a variância, a soma e a cardinalidade. É ainda possível criar gráficos para qualquer campo numérico, nos quais se pode visualizar o número de mensagem por um período de tempo (Graylog, 2018e).

A solução também permite que se possam exportar os resultados das pesquisas para o formato CSV e, caso seja configurado, permite sublinhar a amarelo os termos do *log* que correspondem à pesquisa introduzida.

Os relatórios são uma funcionalidade disponibilizada pela licença Enterprise, através da qual é possível combinar vários *widgets* do *Dashboard* e criar um documento no qual podem ser disponibilizadas determinadas informações. Na seção *Scheduling* o corpo do e-mail e o layout do relatório é configurável pelo utilizador, assim como a frequência de envio para o e-mail (Graylog, 2018e).

3.3.6. Alertas

Alertas (*Alerts*) são sempre baseados em *streams*, sendo que é possível através da definição de condições, acionar alertas automáticos (Graylog, 2018e). Por exemplo, é emitido um alerta sempre que o evento de segurança com ID 1102 (ocorre quando os *logs* são limpos) for recebido (Graylog, 2018b).

Esta funcionalidade está disponível para todas as licenças do Graylog (Graylog, 2020e). É de referir que os Alertas são pesquisas periódicas que podem acionar notificações quando uma condição definida é satisfeita (Graylog, 2018b).

3.3.7. Pontos fortes e fracos

Como já foi referido, o Graylog pode ser utilizado como um SIEM ou como um gestor de *logs*, além de permitir que se tenha a visibilidade da infraestrutura e dos eventos de segurança. É uma arquitetura estável e que requer pouca manutenção, mesmo tendo como comparação produtos similares mais caros (Dienst, 2019). O Graylog possui alguns pontos fortes, tais como (Tal, 2018): é simples de configurar, a sua curva de aprendizagem é suave, a autenticação, os alertas, a análise, os *dashboards* e as permissões são funcionalidades da licença *open-source*.

Embora a solução possua vários plugins e de as suas funcionalidades serem úteis, é de referir que, caso se pretendam implementar funcionalidades além das que o constituem, é necessário recorrer a outras ferramentas. Além disso, a representação gráfica é básica, por isso pode ser necessário recorrer ao Grafana e/ou ao Kibana para otimizar o seu desempenho neste campo (Tal, 2018).

3.4. OSSIM

O SIEM *open-source* da empresa AlienVault, que tem o nome OSSIM (Open Source Security Information and Event Management), disponibiliza vários recursos importantes para a segurança da informação de uma empresa, ou seja, recolhe, normaliza e relaciona os *logs* (AlienVault, 2019).

Em janeiro de 2019 a empresa AlienVault foi comprada pela empresa AT&T Cybersecurity, o que implicou uma mudança de nome do SIEM (Meftah, 2019). A AT&T Cybersecurity oferece vários serviços relacionados com a segurança, de entre os quais dois SIEM, sendo um é *open-source*, o AlienVault OSSIM™, e um produto pago, o USM Anywhere™.

Na solução OSSIM, o armazenamento dos *logs* é feito na base de dados chamada de *SQL Storage*. O USM pode armazenar os dados tal como foram recolhidos ou já depois de normalizados, tendo como objetivo análises forenses e de conformidade, bem como a realização de pesquisas no arquivo (AT&T Cybersecurity, 2019d). Através da interface gráfica, o analista pode configurar quais são os limites para o backup e para o armazenamento, ou seja, é possível definir em relação aos *logs* o tempo máximo de retenção na base de dados, sendo que estes são automaticamente eliminados após o término do período definido (Zarzosa, 2017).

As soluções OSSIM/USM suportam três tipos de correlação (Zarzosa, 2017):

- **Correlação lógica** – faz a relação entre os *logs* e as regras especificadas pelo analista;
- **Correlação cruzada** – verifica se o IP de destino possui alguma vulnerabilidade que já foi identificada e armazenada na base de dados. Para que se possa realizar

essa correlação, é necessário ativar algumas ferramentas que verificam as vulnerabilidades de um *host*, como o Nessus[43] ou o OpenVAS[44].

- **Correlação de inventário** – faz uma correlação entre os *logs* e as características de um determinado destino, permitindo assim reduzir o número de falsos positivos e consolidando a confiabilidade dos mesmos.

Com recurso ao *Open Threat Exchange* (OTX), as soluções OSSIM/USM permitem que possam ser partilhadas ameaças ou *hosts* maliciosos em tempo real, sendo que os dados são partilhados através do AlienVault OTX Pulse, que fornece um resumo da ameaça e também os IoCs (*Indicators of Compromisse*). O servidor USM vai comparar os endereços IP que foram assinalados como maliciosos pelo OTX com os endereços IP de cada *log* da sua infraestrutura, caso encontre na rede da Organização algum dos endereço IP identificados como malicioso, poderá ser emitido um alerta (H. Sousa, 2019; Zarzosa, 2017).

3.4.1. Arquitetura

A USM Appliance junta várias tecnologias indispensáveis para a segurança de uma Entidade na plataforma a Web UI (AT&T Cybersecurity, 2019a, 2020b). A solução paga, a USM pode ser implementada numa única *appliance* ou distribuída por vários servidores (AT&T Cybersecurity, 2019a, 2020b), em contrapartida, a solução OSSIM não é escalável e permite uma única *appliance*, tendo também um volume de *logs* limitado a cinco dias de retenção ou ao armazenamento de quatro milhões de *logs* (Kcoe, 2019).

Como se pode visualizar na Figura 3-5, os componentes da arquitetura da solução USM Anywhere são os seguintes: Web UI, USM Server, USM Logger e USM Sensor. A arquitetura da solução OSSIM é semelhante, mas não inclui o USM Logger (AT&T Cybersecurity, 2019a), todavia, para que se possa perceber melhor o seu funcionamento como um todo, serão explicados de seguida todos os componentes da figura seguinte.

Assim, os sensores são instalados na infraestrutura de rede que se pretende supervisionar e os dados recolhidos nos diferentes dispositivos são normalizados e enviados para o servidor para serem processados. O USM Appliance Server agrega e correlaciona os *logs* recolhidos pelos USM Appliance Sensors e inclui também a interface Web (Web UI), para que seja possível administrar a infraestrutura de rede, criar relatórios e gerir os eventos de segurança

[43] https://www.tenable.com/products/nessus
[44] https://www.openvas.org/

(Zarzosa, 2017). O USM Appliance Logger só está disponível na versão USM e permite guardar os *logs* recolhidos pelos sensores, o mesmo acontecendo para a pesquisa forense e para a criação de relatórios de conformidade mais detalhados (AT&T Cybersecurity, 2019a, 2020b).

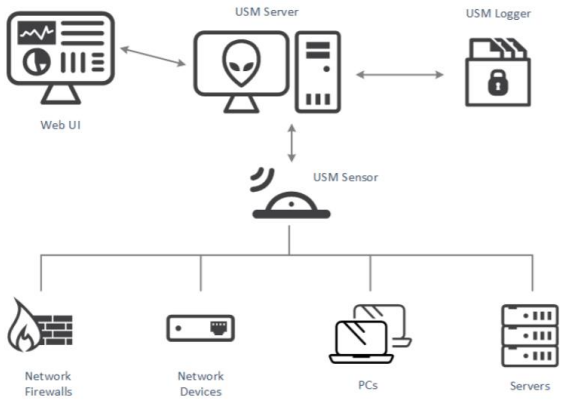


Figura 3-5 Arquitetura da solução USM Anywhere (AT&T Cybersecurity, 2020b, p. 17)

O fluxo de trabalho da USM Appliance não é flexível, portanto, o primeiro passo no fluxo é o USM Appliance Sensors recolher passivamente os *logs* e investigar ativamente os elementos da rede para obter informações sobre as suas atividades e o seu estado. O USM Appliance Sensor transforma os dados em *streams* de *logs*, e agrupa-os tendo em conta um conjunto de campos de dados, posteriormente, os dados são remetidos para o USM Appliance Server. Após este procedimento, o USM Appliance Server correlaciona os *logs* e avalia os riscos que estes podem conter, posteriormente envia-os para o USM Appliance Logger, que regista digitalmente e armazena os *logs*, para que possam ser realizadas análises forenses, garantindo também a conformidade destes com vários regulamentos, entre as quais a ISO27001 (AT&T Cybersecurity, 2019a, 2020b).

Como requisitos mínimos de Hardware, a solução OSSIM recomenda 2 CPU cores, 4 a 8GB de RAM, 250GB HDD (*Hard Disk Drive*) e uma placa de rede a gigabit (AT&T Cybersecurity, 2020a).

3.4.2. Funcionalidades

A solução OSSIM e a solução USM Anywhere partilham várias funcionalidades, das quais enumeramos as seguintes (Alien Vault, 2018; Garth, 2018; H. Sousa, 2019):

- **Descoberta e inventário de ativos –** através da descoberta e inventário de ativos é possível realizar um *scan* à rede, detetando alterações nos ativos e sinalizando os

ativos desconhecidos na rede. A solução OSSIM armazena informações de cada ativo, tais como: IP interno, sistema operativo, modelo e o endereço MAC;

- **Identificação de vulnerabilidades –** o levantamento das vulnerabilidades pode ser efetuado com/sem a autenticação do administrador da rede. A identificação das vulnerabilidades e da conformidade é efetuada através da comparação entre o software que está instalado nos ativos com o que consta numa base de dados de vulnerabilidades. Finalmente, o levantamento de vulnerabilidades pode ser definido para que possa ser periodicamente executado;
- **Deteção de intrusões –** esta funcionalidade realiza a monitorização do tráfego da rede, com o propósito de identificar atividades suspeitas. A deteção de atividades suspeitas na solução OSSIM é conseguida com recurso a um HIDS e a um NIDS.
- **Monitorização de comportamento –** os dados utilizados para a monotorização da rede são recolhidos pelos dispositivos de rede, através da monitorização da disponibilidade dos ativos e através de *port mirroring*. Os dados recolhidos sobre os padrões de tráfego e os fluxos de rede permitem detetar anomalias que podem indicar violações nas políticas de segurança.
- **Correlação de eventos SIEM –** as soluções permitem relacionar os *logs* do tráfego de rede, a atividade dos *hosts* e outras métricas para identificar atividades suspeitas;
- **Suporte da solução através de blogs –** as soluções possuem uma comunidade de suporte para todos os produtos, no qual está incluída a solução OSSIM;
- **Open Threat Exchange –** é uma plataforma que permite partilhar ameaças entre os utilizadores das soluções a nível mundial.

A solução USM possui outras funcionalidades importantes, tais como (AlienVault, 2019): a gestão de *logs;* o suporte dedicado (e-mail e telefone); a procura contínua de ameaças; a visualização complexa de dados e de painéis analíticos; e a documentação online.

3.4.3. Recolha de dados

Como já foi referido, o OSSIM utiliza os USM Appliance Sensors para recolher os dados, que posteriormente são enviados para o USM Appliance Server. Os agentes incluem *plugins* que definem a forma de analisar e de obter as informações geradas pelos diferentes dispositivos e de as transformar em eventos de segurança com formato normalizado (Zarzosa, 2017). Os *plugins* são baseados em expressões regulares e utilizam dois valores

diferentes para identificar a origem dos dados, o tipo de dados e as métricas que suportam, que são seguintes (Zarzosa, 2017):

- **Plugin ID** – identifica o tipo de origem de um *log*, como é o caso do Cisco firewall ou do Snare Windows Agente;
- **Plugin SID** – identifica tipos específicos de eventos para um determinado tipo de origem (ID do plugin).

As soluções OSSIM / USM possuem um conjunto de 11 *plugins* pré-configurados que podem ser ajustados às necessidades do utilizador, permitem ainda que se adicionem novos *plugins* (Zarzosa, 2017).

Os *Monitor plugins* funcionam de forma ativa e são utilizados pelo servidor SIEM para enviar pedidos de consulta para uma ferramenta ou para um destino específico e receber a resposta (Zarzosa, 2017). Alguns dos exemplos das ferramentas utilizadas pelas soluções OSSIM / USM são as seguintes (H. Sousa, 2019; Zarzosa, 2017): Arpwatch, p0f, Nessus, Snort[45], SPADE (*Statistical Packet Anomaly Detection Engine*)[46], ntop[47], Nagios[48], OSSEC[49], OCS Inventory NG (*Open Computer and Software Inventory Next Generation*)[50], Snare[51], nmap, ping, Whois, ou tcptrack.

3.4.4. Resiliência

A resiliência só está disponível na solução USM, sendo que é possível implementar uma solução de alta disponibilidade com um conjunto de nós redundantes que se espelham mutuamente. Ou seja, se a instância primária falhar, a instância secundária fica ativa automaticamente, substituindo o nó em falta (AT&T Cybersecurity, 2020b).

No entanto, existem algumas restrições à implementação de alta disponibilidade, por exemplo (AT&T Cybersecurity, 2020b; Zarzosa, 2017): os dois nós têm de estar na mesma sub-rede, devem estar ligados por um cabo de rede dedicado, os nós devem ter instalada a mesma versão da solução, cada nó deve utilizar interfaces de rede isoladas (exemplo: eth1)

[45] https://www.snort.org/
[46] https://github.com/infosecdr/spade
[47] https://www.ntop.org/
[48] https://www.nagios.org/
[49] https://www.ossec.net/
[50] https://ocsinventory-ng.org/?lang=en
[51] https://www.snaresolutions.com/products/snare-agents/

e, finalmente, o utilizador *root* não pode ter na password carateres especiais (exemplo: ? * [ ]).

3.4.5. Pesquisa e relatórios

A funcionalidade pesquisa nas duas soluções disponibiliza vários filtros estruturados, tais como (AT&T Cybersecurity, 2019b): período temporal, nome do evento, endereço IP.

A solução OSSIM fornece um conjunto de versões simplificadas dos relatórios da USM Appliance, pese embora não possuem a flexibilidade, a personalização ou a riqueza do tratamento das informações que a USM proporciona. Nas duas soluções, os relatórios podem ser visualizados diretamente na interface do utilizador ou podem ser exportados para PDF (AT&T Cybersecurity, 2019c). Estas soluções disponibilizam vários tipos de relatórios – sendo que na solução OSSIM, alguns destes relatórios possuem algumas limitações quando comparando com os que são gerados pela solução USM – dos quais enunciamos os seguintes (AT&T Cybersecurity, 2019c):

- **Relatório de alarmes** – cria um relatório com os principais alarmes com base nos invasores, nos *hosts* atacados e nas portas e qual o risco associado a cada evento;
- **Detalhes dos ativos** – não cria um relatório, todavia, permite observar informações relevantes sobre um ativo ou a rede;
- **Relatório de disponibilidade** – não cria um relatório, contudo enumera a disponibilidade dos ativos e dos serviços;
- **Relatórios de conformidade** – permite criar um relatório com as informações exigidas para as diferentes regulamentações de conformidade, tais como o PCI DSS ou a ISO27001;
- **Relatório por localização geográfica** – distribui o número de alarmes pela sua localização geográfica;
- **Eventos SIEM** – cria um relatório sobre os principais eventos tendo como base os principais invasores, os *hosts* atacados, as portas e o risco de um *log*;
- **Base de dados sobre as ameaças e vulnerabilidades** – não cria um relatório, mas lista as ameaças e as vulnerabilidades encontradas;
- **Estado dos tickets** – disponibiliza várias métricas relacionadas com os tickets, como é exemplo o número de tickets por cada estado;

- **Relatório dos tickets** – lista os tickets criados com base nos alarmes, nos eventos, nas métricas, nas anomalias e nas vulnerabilidades;
- **Relatório da atividade do utilizador** – não cria um relatório, todavia permite visualizar as ações realizadas por utilizador;
- **Relatório de vulnerabilidades** – lista as vulnerabilidades detetadas para cada ativo.

Em suma, os relatórios gerados por estas duas soluções são diversificados e identificam, riscos, ataques e vulnerabilidades que podem comprometer os dados da Organização.

3.4.6. Alertas

Os Alertas disponibilizados pelas soluções OSSIM / USM são desencadeados como consequência de uma ação definida pelo utilizador (Zarzosa, 2017). Quando é recebido um *log* que corresponde às condições definidas nas políticas pelo analista, é acionada uma ação/alerta, sendo que existem três tipos de alertas (Zarzosa, 2017):

- **Envio de e-mail** – pode ser configurado um ou mais endereços de e-mail para o envio dos alertas, a configuração da mensagem a enviar, caso um alerta seja acionando;
- **Execução de um script** – pode ser definido um script para ser executado caso um alarme seja acionado (o script é executado na máquina na qual o servidor SIEM está instalado);
- **Abrir um ticket** – as soluções OSSIM / USM possuem um sistema de tickets e cada utilizador pode ter um ou mais tickets registados.

No USM Appliance é possível identificar os ativos críticos, para os quais se podem estabelecer políticas de geração de alertas quando esses ativos possam estar sujeitos as vulnerabilidades ou a ataques (AT&T Cybersecurity, 2020b).

3.4.7. Pontos fortes e fracos

O OSSIM possibilita a gestão de forma eficiente dos recursos que fazem parte da infraestrutura de rede e permite a atualização em tempo real de ameaças através do *Open Threat Exchange*. Além disso, permite também a identificação de ativos, faz a correlação de *logs*, assegura a definição de regras bastante complexas para a identificação de ações

suspeitas, é fácil de instalar (disponibiliza uma *Appliance*) e possui uma interface Web. Em suma, é uma solução gratuita que disponibiliza serviços profissionais (Trustradius, 2019).

Um dos maiores problemas da solução OSSIM é a escalabilidade, pois está limitada a uma única *appliance* e também possui limites de retenção de *logs*, não permitindo a gestão eficaz dos *logs*. Além disso, não possui suporte e pode ocorrer a perda de dados ou, até, uma falha no servidor devido ao *malware* ou a outra ameaça, uma vez que se está limitado a um servidor (Trustradius, 2019).

3.5. Conformidade com o Regulamento Geral de Proteção de Dados

O RGPD obriga a que as Organizações façam uma grande reforma na sua estrutura em termos operacionais (Varanda, 2019). Por esta razão, quando se implementa um SIEM, também é necessário verificar se este apresenta um nível de segurança adequado para o tratamento dos dados pessoais. O objetivo desta secção é o de fazer um mapeamento entre as medidas técnicas identificadas no Capítulo 2 relativamente ao tratamento de dados sensíveis sugerido pelo RGPD e cada uma das soluções escolhidas.

Como se pretende implementar um SIEM com recurso a ferramentas *open source*, serão destacadas as funcionalidades que são disponibilizadas nas versões *open-source* de cada solução. Todavia, considerámos importante que fosse elaborado um resumo das funcionalidades que cada solução apresenta na sua versão comercial, cuja análise está disponível no Anexo D – Conformidade com o Regulamento Geral de Proteção de Dados.

3.5.1. Splunk

Na licença Splunk Free, com recurso às funções de transformação definidas num ficheiro de configuração próprio, é possível anonimizar ou pseudonimizar os dados (Varanda, 2019).

Esta solução permite indexar 500 MB de dados por dia e os dados não expiram, pelo que podem ficar armazenados na plataforma vitaliciamente (DNSstuff, 2019), no entanto, suporta a definição de limites temporais na retenção dos dados (Splunk, 2016a).

O Splunk Free efetua o encaminhamento dos dados em tempo real e em segurança para destinos remotos (Splunk, 2020a). Na plataforma é garantida a segurança dos dados e é possível detetar se os dados indexados foram comprometidos (Splunk, 2015).

Contudo, como o Splunk Free fornece um acesso muito limitado às funcionalidades do Splunk Enterprise (Splunk, 2020c), não é possível a criação de clusters e de alertas, não é efetuada a monitorização dos dados, só permite um utilizador e não assegura o controlo das ações dos utilizadores (Splunk, 2015; Varanda, 2019).

3.5.2. Elastic Stack

O Logstash, através das pipelines na fase de ingestão, permite a pseudonimização e anonimização dos dados. Para realizar estas operações, recorre ao filtro *fingerprint* que integra funções de *hash* (SHA1, SHA256, SHA384, SHA512, MD5). Quando se utiliza uma chave no filtro *fingerprint*, recorre-se à função HMAC associada a cada função de *hash* (Varanda, 2019).

Como já foi referido anteriormente, a solução *open souce* do Elastic Stack disponibiliza um conjunto de funcionalidades que permitem a escalabilidade e a resiliência, das quais destacamos as seguintes (Elasticsearch, 2019i): *clustering* e alta disponibilidade; balanceamento automático dos dados; e a pesquisa entre clusters.

O Elastic Stack permite gerir o tempo de vida dos dados que se encontram nos índices, sendo que este pode assumir quatro fases (*Hot*, *Warm*, *Cold* e *Delete*) e que é possível especificar o tempo de retenção de cada uma das fases (Elastic, 2020b).

Na licença Basic (a partir da versão 6.8), o Elastic Stack disponibiliza a comunicação encriptada por TLS, permite a criação e gestão de utilizadores, possibilita a proteção nos acessos ao nível do índice e do cluster (Kearnsh, 2019; Kumar, 2019). Em contrapartida, não assegura a autenticação *Active Directory*/LDAP, não garante a segurança ao nível do campo ou do documento (Kearnsh, 2019), não permite a criação de alertas e, finalmente, não audita as ações dos utilizadores (Elastic, 2020a).

3.5.3. Graylog

A solução Graylog permite a pseudonimização e a anonimização dos dados através da funcionalidade "*processing pipeline*" (Graylog, 2018f; Varanda, 2019).

O tempo de retenção dos dados é configurado no *Index Set* e permite que seja definido qualquer tempo de retenção (Black, 2017). É ainda possível controlar quem pode aceder aos dados guardados na solução, além de permitir que sejam definidos vários níveis de permissões (Black, 2017). Possibilita também a autenticação LDAP/*Active Directory*

(Graylog, 2020a), contudo, não permite que sejam auditados os acessos realizados aos dados sensíveis pelos utilizadores que têm autorização para o fazer (Black, 2017).

Através dos clusters do Graylog e do Elasticsearch, é possível implementar redundância e resiliência (Black, 2017). Além disso, permite também a criação de Alertas e de *triggers*, disponibilizando múltiplas opções para a definição dos mesmos (Graylog, 2020e), mas não disponibiliza uma ferramenta específica para a notificação da violação dos dados pessoais.

3.5.4. OSSIM

A solução OSSIM disponibiliza múltiplas funcionalidades diferenciadas, como, por exemplo, a criação de relatórios de conformidade (PCI DSS e ISO27001) e, através do *Open Threat Exchange*, permite partilhar ameaças maliciosas em tempo real, mas, por outro lado, não possibilita uma gestão adequada dos *logs* (AT&T Cybersecurity, 2020b; UnifiedThreatWorks, 2020).

É de referir que o tempo máximo de retenção dos dados na solução OSSIM é de cinco dias (Kcoe, 2019), no entanto, o OSSIM disponibiliza a funcionalidade de criação de Alertas, permite a realização da análise comportamental do utilizador, possibilita a identificação de vulnerabilidades e a deteção de atividades maliciosas (AT&T Cybersecurity, 2020b).

Relativamente à pseudonimização e anonimização, só será possível implementar esta funcionalidade com recurso à integração de outras, além disso, não audita o acesso aos *logs* (AT&T Cybersecurity, 2020b; UnifiedThreatWorks, 2020). Como só permite uma única appliance, o OSSIM não permite redundância ou resiliência.

3.5.5. Resumo comparativo

Tabela 3.1 estão sistematizados os resultados da análise individualizada das funcionalidades de cada solução, assim como o mapeamento das medidas técnicas para proteção e controlo dos dados pessoais identificados no Capítulo 2. É de salvaguardar que as informações constantes nesta tabela foram as recolhidas no decorrer da pesquisa documental. No espaço próprio, no ponto 3.6.5, as quatros soluções serão alvo de uma comparação tendo como base os cenários de testes.

É de referir que Graylog e o Elastic Stack permitem que sejam definidos quais são os utilizadores que podem aceder a um determinado índice, no entanto, para o caso específico

dos dados sensíveis, não possibilitam a realização de uma auditoria às consultas efetuadas pelos utilizadores que possuem permissão para aceder aos dados.

Devido ao tipo de licenças disponibilizadas pelo OSSIM e pelo Splunk Free, só o Graylog e o Elastic Stack garantem a redundância e a resiliência.

Todas as quatro soluções garantem um nível de segurança básico aos dados, assegurando a segurança dos dados em trânsito e restringindo o acesso aos dados que estão na solução. No entanto, é de referir que só as soluções comerciais garantem a proteção dos dados por design e por padrão, uma vez que as soluções *open-source* disponibilizam um menor número de funcionalidades. Além disso, nenhuma das soluções *open-source* disponibiliza a funcionalidade de notificação de violação de dados.

Relativamente aos tempos de retenção e à anonimização/pseudonimização dos dados, só o OSSIM não disponibiliza as referidas funcionalidades, no entanto, permite a criação de relatórios de conformidade. Esta solução, à semelhança do Graylog e do Elastic Stack, também permite definir vários níveis de permissões.

Tabela 3.1 – Mapeamento das medidas técnicas sugeridas pelo RGPD com as soluções *open-source*

Funcionalidades	**Mapeamento das medidas técnicas sugeridas pelo RGPD com as soluções *open-source***			
	OSSIM	***Elastic Stack***	***Splunk Free***	***Graylog***
Anonimização	Não	Sim	Sim	Sim
Pseudonimização	Não	Sim	Sim	Sim
Permitir definir tempo de retenção para os dados	Não	Sim	Sim	Sim
Garantir a segurança dos dados dos utilizadores	Sim	Sim	Sim	Sim
Efetuar notificações de violação de dados	Não	Não	Não	Não
Restringir o acesso aos dados pessoais	Sim	Sim	Não	Sim
Auditar e monitorizar os acessos a dados pessoais	Não	Não	Não	Não
Garantir resiliência	Não	Sim	Não	Sim
Recuperação a desastres	Não	Sim	Não	Sim
Proteção de dados por design e por padrão	Não	Não	Não	Não
Criação de Relatórios de Conformidade	Sim	Não	Não	Não

De uma forma resumida, ao analisar a Tabela 3.1, é possível concluir que as soluções Elastic Stack e Graylog disponibilizam um maior número de funcionalidades para garantir a proteção e o controlo dos dados pessoais.

3.6. Cenários de Testes

Na pesquisa documental efetuada sobre as quatro soluções aferiu-se que as soluções Elastic Stack e Graylog são escaláveis e que disponibilizam várias funcionalidades adequadas para a proteção e controlo de dados. Como nesta fase se estava a considerar implementar duas soluções em conjunto como solução possível para a construção do protótipo, entendeu-se relevante que fosse criar um cenário de testes para testar a usabilidade de cada uma das soluções. Será de salientar que o Elastic Stack só disponibilizou a funcionalidade de segurança a partir da versão 6.8 e que o cenário de testes foi realizado na versão 6.5.

Um dos objetivos principais para a criação dos cenários de testes foi o de testar a usabilidade das quatro soluções. No entanto, é de referir que a facilidade de utilização por si só não é suficiente, pois também é importante a solução identifique ameaças e ataques. No presente contexto, entendeu-se que era relevante conseguir monitorizar em tempo real um ataque (neste cenário recorreu-se ao ataque de força bruta). Outro objetivo fundamental foi a avaliação do desempenho das várias soluções relativamente às seguintes funcionalidades: facilidade de instalação, facilidade de instalação de agentes, facilidade de configuração, facilidade de pesquisa, criação de alertas, criação de *Dashboards*, compatibilidade com os sistemas operativos Linux e Microsoft Windows e se permitia ou não a autenticação. Para mensurar os resultados obtidos, foi elaborada uma tabela resumo para que depois, mais facilmente se pudessem tirar conclusões sobre os resultados obtidos.

O computador no qual foram realizados os testes (sistema anfitrião) possuía as seguintes caraterísticas técnicas: ASUSPRO B8430U (i7-6500U), um disco SSD de 250GB, um disco HDD de 1TB e 16GB de memória RAM. O sistema anfitrião é a máquina que possui o sistema de suporte de máquinas ou de equipamentos virtuais, e, neste primeiro caso, o software Oracle VirtualBox versão 5.2.X.

Foram criados quatro cenários de teste, sendo que para as máquinas cliente foram escolhidos os sistemas operativos Linux (servidor web e controlo de equipamentos) e Microsoft Windows (postos de trabalho), uma vez que a empresa na qual vai ser implementado o SIEM utiliza maioritariamente estes dois sistemas operativos. O cenário de testes implementado foi baseado na topologia de rede de Tavares (Tavares, 2015), mas, uma vez que nesta fase só se pretendia avaliar funcionalidades e calcular o grau de complexidade da implementação do SIEM, suprimiu-se o router e o servidor Web, como se pode constatar na Figura 3-6.

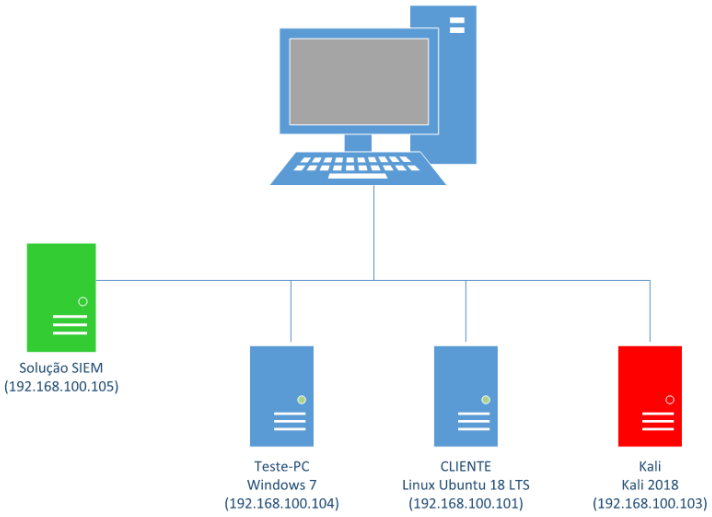


Figura 3-6 Cenário de testes das soluções SIEM

A infraestrutura definida para o cenário de testes e para a análise das funcionalidades das soluções escolhidas, como se pode visualizar no esquema anterior, é constituída por uma máquina virtual Teste-PC com o sistema operativo Microsoft Windows 7 (IP: 192.168.100.104), uma máquina virtual CLIENTE com o sistema operativo Linux Ubuntu 18 LTS (IP: 192.168.100.101) e a uma terceira máquina virtual KALI com o sistema operativo Kali 2018 instalado (IP: 192.168.100.103). É de referir que as caraterísticas das máquinas virtuais, bem como o seu IP, foram semelhantes em todos os testes.

Divergiu, no entanto, o número de máquinas testado em cada uma das soluções selecionadas: no OSSIM foi utilizado a *appliance* fornecida pela empresa AT&T Cybersecurity; utilizou-se uma máquina Linux Ubuntu 18 LTS para o Splunk e para o Graylog (IP: 192.168.100.105) e três máquinas Linux Ubuntu 18 LTS para o Elastic Stack (Elasticsearch com o IP: 192.168.100.105, o Logstash com o IP: 192.168.100.106 e o Kibana com o IP: 192.168.100.107).

Na máquina CLIENTE (IP: 192.168.100.101) também foi instalado o serviço de SSH com uma password simples para que, posteriormente, fosse possível efetuar um ataque ao serviço SSH com sucesso. Foram também efetuadas as configurações necessárias para que os eventos de *logs* fossem encaminhados automaticamente para as soluções instaladas. Desta forma, foi possível monitorizar em tempo real os eventos da máquina cliente na máquina servidor. Para implementar o cenário de testes fez-se um levantamento de boas práticas sugeridas pelas solução que se encontra no Anexo E – Boas práticas das soluções o que permitiu agilizar algumas configurações.

Através da máquina KALI, com o IP 192.168.100.103, foram realizados os ataques à máquina CLIENTE (IP: 192.168.100.101). Para a realização do ataque de força bruta foi utilizada a ferramenta de linha de comando Hydra[52] e a wordlist *rokyou.txt*, como se pode visualizar na Figura 3-7.

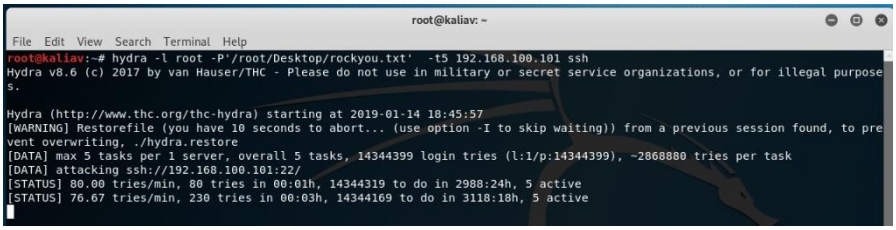


Figura 3-7 Exemplo de ataque efetuado pelo Kali a uma máquina Microsoft Windows

Uma vez que o servidor recolheu os dados dos *logs* em tempo real, foi possível acompanhar a evolução do ataque. Para todos os cenários, procedeu-se a um ataque à máquina CLIENTE (IP: 192.168.100.101) com recurso à máquina KALI (IP 192.168.100.103) para efetuar o ataque através do serviço SSH. Já na máquina Teste-PC (IP: 192.168.100.104) foram efetuadas várias tentativas falhadas de *login*. Em todas as soluções, foram executadas várias pesquisas, foram criados alertas que enviaram automaticamente uma mensagem de e-mail, foram instalados os agentes e criados *Dashboards*.

Como os testes foram semelhantes em todos os cenários, foi possível aferir o desempenho das soluções, o seu grau de complexidade, assim como a sua facilidade de utilização. O cenário implementado foi limitado a poucos ativos e só foi efetuado um tipo de ataque que é facilmente identificável, pelo que, caso as soluções fossem implementadas em ambiente profissional a análise das mesmas poderia ser diferente. Devido às razões enumeradas é importante referir que estes resultados não podem ser extrapoláveis, uma vez que os testes foram efetuados unicamente pela investigadora.

3.6.1. Cenário de testes da solução Splunk

Nesta solução optou-se por testar a totalidade das suas funcionalidades e não apenas as facultadas pela solução Splunk Free, esta decisão teve como finalidade beneficiar das funcionalidades que são disponibilizadas por uma solução paga. Escolheu-se o Splunk, pois segundo a *Gartner*, é um dos líderes de mercado a nível mundial e fornece uma versão *freeware* (Splunk Free).

[52] https://tools.kali.org/password-attacks/hydra

Tendo como base o esquema da Figura 3-6, foi instalado o software Splunk® Enterprise 7.0.3 na máquina com o sistema Operativo Linux Ubuntu 18 LTS e o IP: 192.168.100.105. Nas máquinas com os sistemas operativos Microsoft Windows e Linux (IP: 192.168.100.104 e IP: 192.168.100.101) foi instalado o *Splunk Universal Forwarder 7.0.3*. Depois de a solução ser instalada, foi possível aceder ao Splunk GUI com as credenciais utilizador: *admin* e password: *changeme*.

Além disso, foram efetuadas as configurações necessárias para que os eventos de *logs* fossem enviados automaticamente para o servidor (máquina servidor com IP: 192.168.100.105). Assim, foi possível monitorizar na máquina servidor e em tempo real os eventos da máquina cliente. Depois de ser configurada a recolha dos dados, na opção *Data Summary*, foram listadas as máquinas clientes que estavam a enviar os *logs* para o servidor, bastando clicar no nome do *host* para se pudessem efetuar pesquisas nos eventos do mesmo. Em relação à máquina Microsoft Windows, foi possível encontrar rapidamente as várias tentativas falhadas de autenticação. Na imagem seguinte pode visualizar-se o ataque em tempo real que realizado através da máquina Kali à máquina que possui o serviço de SSH.

Figura 3-8 *Logs* resultantes do ataque efetuado pela máquina Kali na solução Splunk

As pesquisas efetuadas foram eficazes, contudo, foi necessário estudar as regras da linguagem SPL, para que fosse possível criar um alerta em tempo real através de uma pesquisa, utilizando a frase: "*maximum authentication attempts exceeded*". Na Figura 3-9 foram apresentados alguns dos parâmetros que podem ser definidos num *Alert* da solução Splunk.

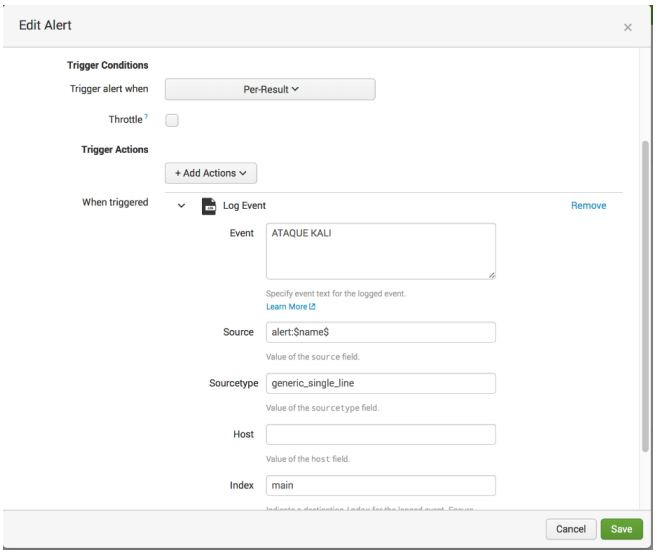


Figura 3-9 Configuração de um Alerta no Splunk

É de referir que as pesquisas também podem ser guardadas como *Report*, *Dashboard Panel* e *Event Type*. Através de comandos da pesquisa é possível criar tabelas e gráficos ou exportar os *logs* no formato CSV.

A documentação da solução está bem fundamentada para a solução paga e, sempre que necessário, recorreu-se à mesma para a resolução de problemas.

3.6.2. Cenário de testes da solução Elastic Stack

Como foi referido anteriormente, este cenário diferiu em número de máquinas, pois optou-se por instalar os componentes: Elasticsearch (IP: 192.168.100.105), Logstash (IP: 192.168.100.106) e o Kibana (IP: 192.168.100.107) em máquinas diferentes. A decisão de separar os componentes no Elastic Stack prende-se com o facto de esta solução poder integrar outras ferramentas, como são o Wazuh, o Kafka, o Redis ou de alguns dos seus componentes poderem ser integrados numa outra solução, como é o caso do Graylog com o Elasticsearch ou do Cyphon[53] que pode integrar o Logstash e o Elasticsearch.

A arquitetura definida para este cenário é mais complexa, por isso, o grau de complexidade na implementação também aumentou e a linha de aprendizagem foi mais demorada, uma vez que a maior parte das configurações é efetuada por linha de comandos ou com recurso a ficheiros de configuração com diversos parâmetros.

Nas máquinas foram instalados a seguintes versões das soluções: elasticsearch-6.5.0, logstash-6.5.0, kibana-6.5.0, foi também instalado no Linux o Beat filebeat.6.5.4 e no

[53] https://www.cyphon.io/

Microsoft Windows foram instalados os seguintes Beats: o winlogbeat-5.1.1-windows-x86_64 e o metricbeat-5.1.1-windows-x86_64.

No Kibana foi criado um *Index Pattern* para cada índice do Elasticsearch e, só depois, é que os *logs* ficaram disponíveis para integrarem as várias tarefas que é possível efetuar neste interface. Após a configuração dos "*Index Patterns*", já foi possível realizar pesquisas no separador *Discover*. Através das pesquisas, foi possível identificar as várias tentativas falhadas de autenticação na máquina Microsoft Windows e a visualização do ataque da máquina Kali ao serviço SSH da máquina Linux, como se pode observar na Figura 3-10.

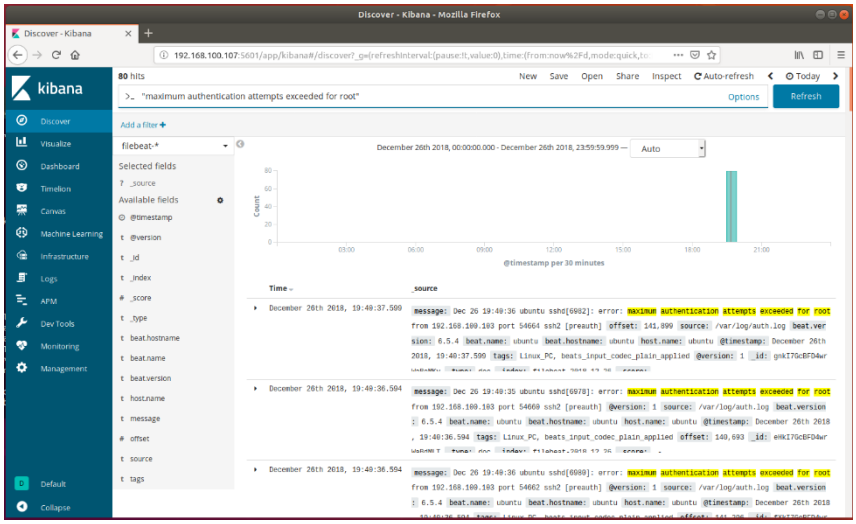


Figura 3-10 *Logs* resultantes do ataque efetuado pela máquina Kali na solução Elastic Stack

O Kibana permite que sejam construídos diversos gráficos e tabelas, contudo, na versão *open-source*, não são disponibilizados os relatórios, Alertas ou o *machine learning*. Os alertas no Elastic Stack são pagos, no entanto, optou-se por testar os mesmos. É de referir que os alertas (*Watcher)* são difíceis de implementar, pois estes são configurados através de comandos (ver Anexo F – Configuração do Watcher no Kibana) como se pode observar na imagem seguinte.

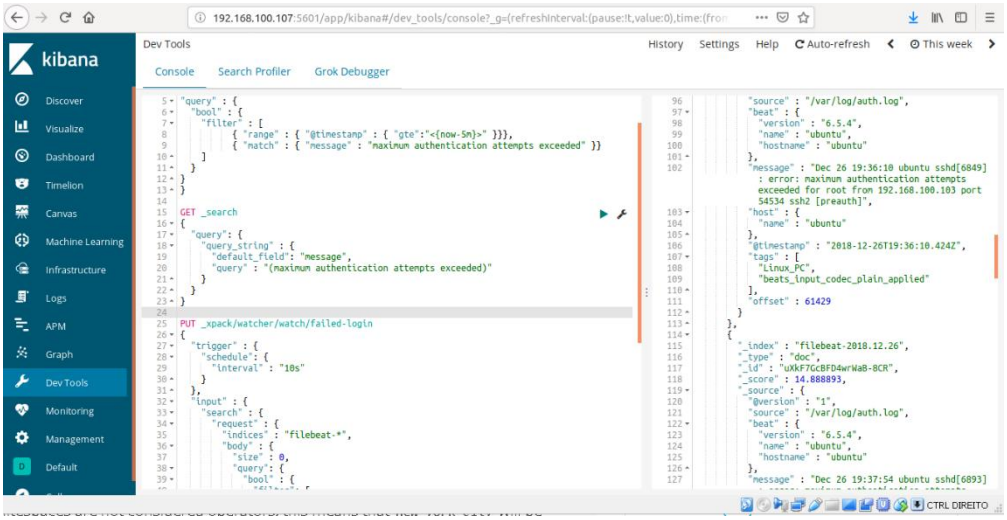


Figura 3-11 Configuração de um Alerta (*Watcher*) no Kibana

Depois de ser configurado o e-mail do remetente e o alerta de forma manual (*Watcher*) foi possível receber um e-mail com a notificação da existência de um grande número de tentativas falhadas para uma determinada máquina. É de salientar que a documentação da solução foi importante para a implementação do cenário, mas, como a solução está em constante evolução, foi necessário ter atenção a esse aspeto e procurar consultar a documentação relativa à versão que foi instalada no cenário.

3.6.3. Cenários de testes da solução Graylog

já foi referido no ponto 3.5 acimae ilustrado na Figura 3-6, foi instalada na máquina com o IP: 192.168.100.105 a solução Graylog. Foram também instaladas as seguintes versões no servidor: versão graylog-2.5, mongodb-org-3.6.list e a verão elastic-6.x.list. Na máquina cliente com o IP: 192.168.100.101 recorreu-se ao *rsyslog* para enviar os *logs* para a solução. Em ambiente Microsoft Windows, na máquina cliente com o IP: 192.168.100.104 recorreu-se ao NXLog, configurando o formato de *logs* como GELF.

Em relação ao servidor foi necessário efetuar várias configurações, para além das iniciais. No servidor foram configurados os Inputs das máquinas das quais foram recolhidos os *logs* (IP: 192.168.100.101 e o IP: 192.168.100.104). Foi criado um *Dashboard* para apresentar as tentativas falhadas na autenticação SSH, sendo que, quando se cria um *Dashboard*, o mesmo está sempre vazio, pelo que é necessário adicionar um *Widget*. Os *Widget* são resultados de pesquisa, sendo que podem ser adicionados vários tipos de pesquisa: gráficos de histogramas; estatísticas; valores de resultados em tempo real; entre outros. Também foi criado um outro utilizador para que fosse atribuída a permissão de visualização dos resultados de pesquisa, mas não permitindo que o *Dashboard* fosse alterado. Esta

funcionalidade é importante para o RGPD, pois os utilizadores só visualizam o resumo dos dados, o que garante que não têm acesso a dados sensíveis.

Nesta solução foram efetuadas várias pesquisas, sendo que a sintaxe é semelhante ao *Lucene*, ou seja, por norma todos os campos são incluídos na pesquisa, caso não se especifique qual o campo a pesquisar. As pesquisas podem ser exportadas para o formato CSV, possibilitando que sejam analisadas por outras ferramentas. Neste ponto, considerou-se que estas eram eficazes, uma vez que foi possível encontrar rapidamente o que se estava a investigar, sejam as várias tentativas falhadas de *login* na máquina Microsoft Windows ou as múltiplas tentativas falhadas na autenticação SSH na máquina Linux. A imagem seguinte ilustra o resultado da pesquisa que foi efetuada e se detetaram as tentativas falhadas na autenticação SSH na máquina Linux.

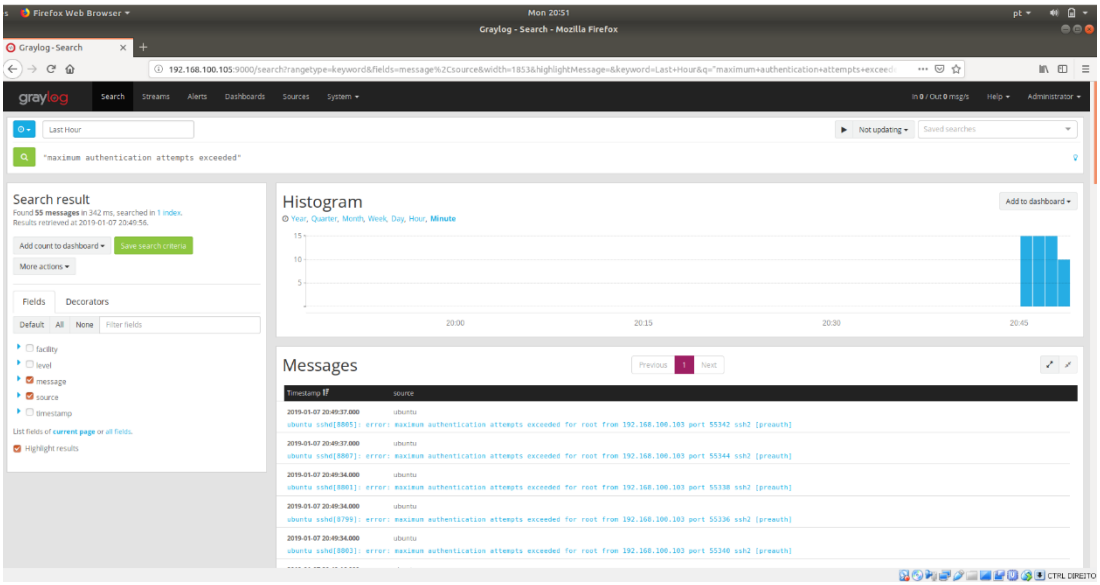


Figura 3-12 *Logs* resultantes do ataque efetuado pela máquina Kali na solução Graylog

Para que se pudesse receber um e-mail de alerta quando a máquina fosse atacada, foi necessário configurar o remetente e as *Streams* (mecanismo para encaminhar notificações em tempo real) de forma manual. As *Streams* possuem múltiplas configurações e podem ser efetuadas com as seguintes opções: Manage Rules (regras de *Streams*), Manage Outputs (encaminhamento de mensagens de *Streams* para outros locais) e *Manage Alerts* (configuração de alertas de notificações de acordo com os limites pré-estabelecidos). Na Figura 3-13, pode visualizar-se uma das configurações que foi efetuada na solução Graylog.

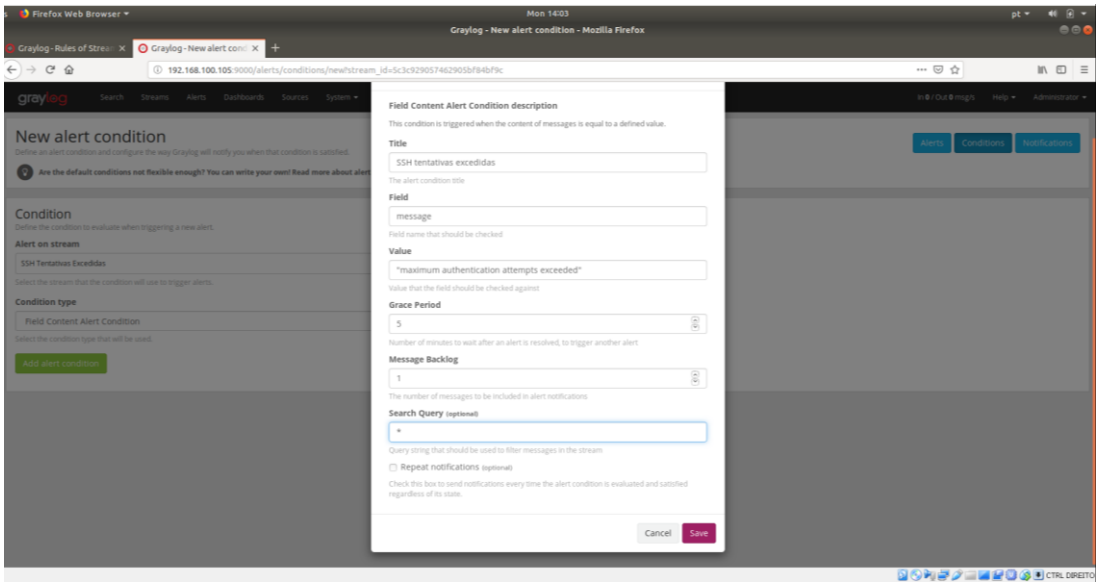


Figura 3-13 Configuração de um Alerta no Graylog

Depois de ser realizado o ataque com a máquina que tem instalado o Sistema Operativo Kali com o IP: 192.168.100.103 à máquina com o Sistema Operativo Linux, foi possível receber automaticamente um alerta em tempo real no e-mail.

Para que se possa utilizar a solução Graylog é necessário configurar o utilizador, pois este exige que se introduzam as credenciais de utilizador (utilizador/palavra passe) para aceder às suas funcionalidades. Esta solução também possibilita que sejam criados vários utilizadores e que sejam atribuídos diferentes níveis de permissões. Sempre que foi necessário, recorreu-se à documentação que a solução disponibiliza.

3.6.4. Cenários de Testes da solução OSSIM

Para a solução OSSIM foi utilizado a *appliance* fornecida pela empresa AT&T Cybersecurity. O processo de instalação é de fácil compreensão e de rápida execução (no mesmo podemos definir o endereço IP da solução e as redes a analisar). Depois de configurada a solução, já é possível aceder à interface gráfica e efetuar as restantes configurações na mesma. O OSSIM, tal como o Graylog, requere autenticação, pelo que é necessário introduzir as credenciais que foram definidas na instalação.

É de referir que existem duas formas de adicionar ativos: automaticamente e manualmente. Neste ponto, optou-se por configurar os ativos manualmente (*Add Assets* e de seguida *Add Host*), pois existem situações em que não é possível instalar o agente automaticamente. Como esta solução integra outras ferramentas, recorreu-se ao OSSEC para instalar o agente na máquina Linux (IP: 192.168.100.101). Para a máquina Microsoft

Windows foi possível fazer o download do agente OSSEC, pré-configurado através do OSSIM, o que simplificou as configurações na máquina Microsoft Windows (IP: 192.168.100.104).

As pesquisas, quando comparadas com as outras soluções, são mais rígidas, no entanto, foi possível encontrar indícios do ataque da máquina Kali ao serviço SSH da máquina Linux, como se pode observar na Figura 3-14. Também foi possível listar o número de tentativas de login falhadas na máquina Microsoft Windows.

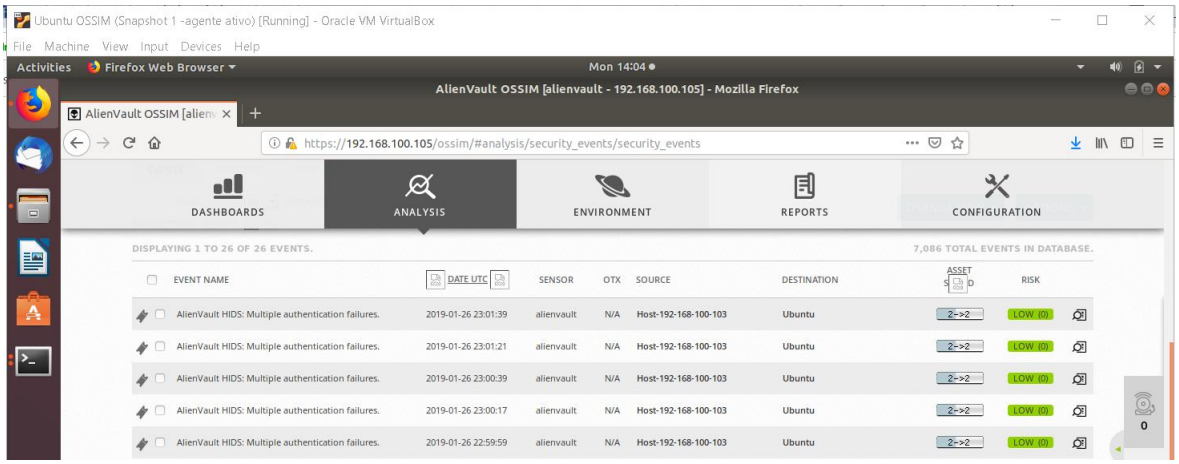


Figura 3-14 *Logs* resultantes do ataque efetuado pela máquina Kali na solução OSSIM

Para que fosse possível receber um alerta no e-mail com a comunicação que estava a ocorrer um grande número de tentativas falhadas numa determinada máquina, foi necessário configurar o remetente no ficheiro de configuração e, de seguida, na interface gráfica da solução.

Depois de concluído o passo anterior, para que fosse possível receber a mensagem de alerta no e-mail, foi necessário criar uma *Action*, na sua *Policy*, atribuir várias políticas (como por exemplo, uma política que foi selecionada "*Multiple failed logins from same source ip*") para que fosse possível criar um Ticket. Os Ticket podem ser configurados para serem enviados para o e-mail, como se pode visualizar na figura seguinte.

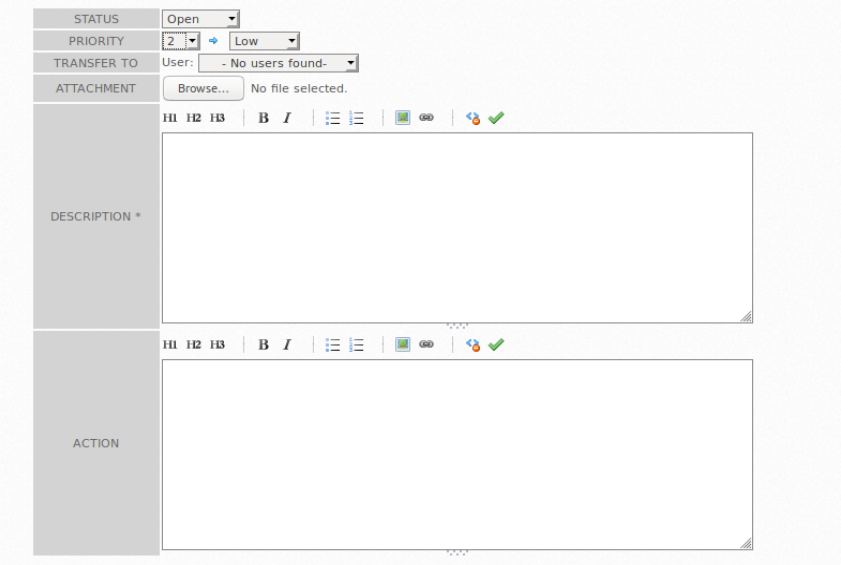


Figura 3-15 Configuração de um Alerta no OSSIM

A solução OSSIM fornece várias vistas e possibilita a elaboração de vários relatórios relacionados com a segurança, além de disponibilizar vários *dashboards* nos quais se podem visualizar as situações mais críticas da rede que está a ser monitorizar e identificar também as vulnerabilidades.

3.6.5. Resumo comparativo

Assim, devido à implementação das soluções no cenário de testes, surgiram algumas dúvidas que obrigaram à consulta da documentação disponibilizada pelas mesmas. É de referir que todas as soluções dão prioridade às suas soluções pagas, mas, mesmo assim, é relativamente fácil descobrir as informações para ultrapassar as dificuldades, contudo, na solução OSSIM foi bastante difícil encontrar uma solução para os problemas encontrados.

Os ambientes gráficos das soluções são semelhantes, pois todos utilizam interfaces Web para a interação com o utilizador e permitem a criação de *dashboards*. Relativamente à facilidade de utilização, considerou-se que a solução Graylog é a mais intuitiva.

Comparando os agentes para a recolha dos dados, é de referir que o Graylog necessita da ferramenta NXLog para o Sistema Operativo Microsoft Windows; a solução OSSIM integra a ferramenta OSSEC, permitindo, no caso do Microsoft Windows, que se possa efetuar o download da ferramenta já pré-configurada; o Splunk e o Elastic Stack disponibilizam os seus próprios agentes. Comparando todos os agentes, entendeu-se que o Splunk Universal Forwarder 7.0.3 foi aquele que requereu menos configurações, o que facilitou a sua instalação.

Como se pode visualizar na Tabela 3.3, de todas as soluções, aquela que é mais rígida em relação às pesquisas é a solução OSSIM, porque não permite que se pesquise por todos os campos do *log* e seu conteúdo, ao contrário das outras soluções, não inclui na pesquisa, por padrão, todos os campos da mensagem. No entanto, em todas as soluções foi possível encontrar as tentativas falhadas de login e os indícios do ataque da máquina Kali. Para que os alertas pudessem ser enviados para um e-mail, foi necessário configurar o serviço de e-mail. Neste aspeto, foram identificadas algumas dificuldades na configuração do e-mail do OSSIM, pois a documentação da solução não especifica como esta deve ser efetuada, sendo necessário recorrer a outras fontes para resolver o problema, como é o caso do Stack Overflow. Devido a esse facto, foi atribuída um nível médio de dificuldade na configuração de e-mail no OSSIM, como está elencado na Tabela 3.3.

Por seu lado, o Elastic Stack não exige a criação de uma conta para autenticação de utilizador, ao contrário de todas as outras soluções, que obrigam a que existam credenciais para aceder à interface web da plataforma. O Graylog, para além do que foi referido anteriormente, permite a criação de várias contas de utilizador, assim como de vários níveis de permissões. Deixamos a nota que, depois da implementação do cenário, o Elastic Stack disponibilizou na versão 6.8, para a licença Basic, os recursos de segurança básicos, nos quais já é possível a autenticação.

Como já foi referido, em todas as soluções analisadas foram criados *dashboards*, contudo, tendo em conta que um dos objetivos é que todas as ações levadas a cabo numa solução estejam em conformidade com o RGPD, é de referir que o Graylog possui uma funcionalidade muito importante neste âmbito. Esta solução possibilita que sejam atribuídos níveis de permissões, por exemplo, no caso de um gráfico no qual é apresentado o somatório da utilização das redes sociais em tempo real, é possível atribuir a utilizadores a permissão para o visualizar, mas impedir que estes tenham acesso à identificação dos utilizadores das referidas redes. Será de referir, mais uma vez, que o Elastic Stack disponibilizou, na versão 6.8 para a licença Basic, a funcionalidade de criar vários utilizadores e de atribuir vários níveis de permissões.

Relativamente aos Alertas, foi possível criar alertas em todas as soluções, pese embora na solução Splunk Free e no Elastic Stack esta funcionalidade não seja disponibilizada nas versões freeware. Como foi descrito nos cenários anteriores, existem várias formas de criar e de gerir alertas, contudo, consideramos que os alertas no Elastic Stack foram os mais

complicados de implementar, uma vez que foram implementados através de comandos (Watcher).

Fazendo uma comparação geral entre todas as soluções, consideramos que Elastic Stack foi a arquitetura mais complexa que se implementou, uma vez que o cenário tinha três máquinas. Relativamente à curva de aprendizagem, também mais foi demorada no Elastic Stack, uma vez que existiram funcionalidades que, para funcionaram, necessitaram da execução de vários comandos, como é o caso do *Watcher*.

Tabela 3.2 – Dados resultantes da implementação dos cenários

Funcionalidades	**Soluções implementadas nos cenários**			
	OSSIM	***ELK Stack***	***Splunk***	***Graylog***
Pesquisa (abrangência)	Estruturada	Abrangente	Abrangente	Abrangente
Configuração e-mail	Médio	Fácil	Fácil	Fácil
Alertas	Médio	Scripts	Médio	Médio
Dashboards	Sim	Sim	Sim	Sim
Autenticação	Sim	Não [54]	Sim	Sim
Instalação dos agentes	Médio/Difícil	Médio/Difícil	Fácil	Médio/Difícil
Grau de complexidade na implementação	Médio	Complexo	Simples	Simples
Facilidade de utilização	Médio	Médio	Médio	Simples
Compatibilidade com o Microsoft Windows	Sim	Sim	Sim	Sim
Compatibilidade com o Linux	Sim	Sim	Sim	Sim

É de referir que, no processo de implementação do cenário de testes e no decorrer dos testes, existiram algumas limitações, pois só foi possível efetuar as experiências num computador recorrendo a máquinas virtuais, o que significa que existia um número reduzido de ativos, o que vai implicar que as conclusões não possam ser extrapoladas.

3.7.Resumo comparativo entre soluções

Depois de ser realizada a análise documental das quatro soluções e de terem sido efetuados os testes em cenários próprios, foi possível fazer resumo comparativo entre as quatro soluções. Para que fosse possível realizar a análise comparativa entre estas soluções, utilizaram-se os requisitos que estão enumerados na Tabela 3.3. Na seleção dos critérios,

[54] A autenticação só foi disponibilizada a partir da versão 6.8

tiveram-se em conta as funcionalidades de um SIEM que estão enumeradas na Figura 2-4 do Capítulo 2, e que são as sugeridas pela empresa Exabeam (2020) em conjunto com algumas funcionalidades identificadas na análise documental sobre cada solução. Enumeramos de seguida as funcionalidades elencadas na Figura 2-4 do Capítulo 2: Recolha de *logs*, Análise de *logs*, Correlação de diferentes *logs*, Análise forense, Conformidade legal, Monitorização de *logs* das aplicações, Auditoria de acessos a ficheiros, Alertas em tempo real, Auditoria à atividade dos utilizadores, *Dashboards*, Relatórios, Monitorização da integridade dos ficheiros e Monitorização dos *logs* de sistemas e equipamentos.

No processo de seleção dos critérios, que devem servir de base para a realização de uma análise comparativa entre as soluções, também foram considerados relevantes os requisitos que possam contribuir para a segurança dos dados de uma forma mais eficaz. Além disso, também se selecionaram os requisitos relacionados com a arquitetura e escalabilidade das soluções. É de explicitar que não foram repetidos requisitos elencados na Tabela 3.1 e Tabela 3.2, pois já tinham sido utilizados para efetuar a comparação das soluções dum ponto de vista mais específico. Tendo em conta as funcionalidades disponibilizadas por cada uma das soluções, foi considerado importante que fosse efetuada uma comparação dos seguintes requisitos entre cada SIEM escolhido: Tipo de licença; Arquitetura; Requisitos mínimos de hardware/software; Identificação de vulnerabilidades dos equipamentos; Gestão de *logs*; Escalabilidade; Integração com o *Active Diretory* ou com o LDAP; Descoberta e inventário de ativos; *Machine learning*; Implementar *workflows* de resposta a incidentes; Escalabilidade e a Partilha de ameaças em tempo real.

Na Tabela 3.3 foi efetuado um estudo comparativo entre as quatro soluções escolhidas, tendo como base os requisitos indicados anteriormente. Será importante fazer a ressalva de que as funcionalidades indicadas como indisponíveis na versão freeware podem estar disponíveis nas versões comerciais.

Relativamente ao tipo de licença, caso se opte por um serviço na *Cloud*, este está limitado a um determinado volume de *logs* por dia (Splunk e Graylog) ou possui um período experimental (USM e Elastic Stack). Caso se decida implementar uma solução *On-Premise* (implementação local), a solução OSSIM possui limites reduzidos de retenção, o Splunk free não disponibiliza esta opção, e, por último o Elastic Stack e o Graylog não possuem limites.

Como já foi referido anteriormente, o Splunk Free não é um *open-source* e o Elastic Stack não é considerado um SIEM. Também são apresentados na Tabela 3.3 os requisitos mínimos

de hardware e de software, no caso específico do hardware serve apenas como identificador, pois este está sempre dependente do volume de dados a tratar. Relativamente aos sistemas operativos, o Elastic Stack e o Splunk Free são compatíveis com os sistemas operativos Linux e Microsoft Windows, o Graylog só é compatível com o sistema operativo Linux e o OSSIM fornece a sua própria *Appliance.*

Como se pode visualizar na Tabela 3.3, todas as soluções permitem a monitorização das aplicações, dos equipamentos ou dos sistemas e permitem a realização de análises forenses aos *logs*. Em contrapartida, nenhuma solução disponibiliza a funcionalidade de *machine learning,*

Como se pode visualizar na Tabela 3.3, nenhuma das soluções disponibiliza todas as funcionalidades enumeradas na tabela, todavia neste ponto destacamos duas funcionalidades que são indispensáveis quando se implementa um SIEM a escalabilidade e a gestão de *logs*.

Tabela 3.3 – Tabela comparativa das soluções: OSSIM, Elastic Stack, Splunk Free e graylog

Funcionalidades	**Comparação entre OSSIM, Elastic Stack, Splunk e Graylog**			
	OSSIM	***Elastic Stack***	***Splunk Free***	***Graylog***
Licença	*Open-source*	*Open-source*	*freeware*	*Open-source*
Arquitetura	*Appliance*	*On-premise / Cloud*	*On-premise / Cloud*	*On-premise / Cloud*
Requisitos mínimos de hardware/software	*2 CPU cores * 4 a 8GB de RAM* 250GB HDD*	*8 CPU cores * 16 a 32 GB de RAM * Disco SSD * Linux/* Microsoft *Windows*	*12 CPU cores a 2Ghz * 12 Gb de RAM Disco SSD* Linux/* Microsoft *Windows*	*4 CPU cores * 8 GB de RAM * Disco SSD * Linux*
Identificação de vulnerabilidades	Sim	Não	Não	Não
Descoberta e inventário de ativos	Sim	Não	Não	Não
Monitorização de aplicações/sistemas/equipamentos	Sim	Sim	Sim	Sim
Correlação de diferentes *logs*	Sim	Sim	Sim	Sim
Gestão de *logs* (recolha e análise de *logs*)	Não	Sim	Sim	Sim
Relatórios	Sim	Não	Sim	Não
Alertas	Sim	Não	Não	Sim
Escalabilidade	Não	Sim	Não	Sim
Integração LDAP ou Active Directory	Sim	Não	Não	Sim
Auditoria de acessos a ficheiros	Não	Não	Não	Não
Auditoria à atividade dos utilizadores	Não	Não	Não	Não
Análise Forense	Sim	Sim	Sim	Sim
Monitorizar a integridade dos ficheiros	Sim	Sim	Sim	Não
Machine learning	Não	Não	Não	Não
Implementar *workflows* de resposta a incidentes	Não	Não	Não	Não
Partilha de ameaças em tempo real	Sim	Não	Não	Não

De seguida, e tendo como base as tabelas Tabela 3.1, Tabela 3.2 e Tabela 3.3, é possível fazer um resumo dos pontos fortes e fracos de cada solução SIEM. A solução OSSIM destaca-se por oferecer várias funcionalidades que não estão disponíveis nas outras soluções, como são os seguintes casos: a identificação de vulnerabilidades; da monitorização de comportamento; dos relatórios de conformidade (é possível obter relatórios de regulamentações de conformidade); da descoberta e do inventário de ativos; e da funcionalidade *Open Threat Exchange* (atualização de ameaças em tempo real). Por outro lado, esta solução não é escalável, não permite uma gestão eficaz de *logs* e possui limites de retenção reduzidos.

A solução Splunk Free possibilita a gestão de *logs* e permite a construção de relatórios, deteta anomalias, efetua o enriquecimento dos dados. Contudo, esta solução não é *open-source* e na sua versão freeware não é escalável, não permite a geração de alertas e só permite um utilizador.

O Elastic Stack é escalável, possibilita a gestão dos *logs*, correlaciona os eventos e permite vários níveis de autenticação, contudo, não permite a criação de relatórios ou de alertas. Como foi referido anteriormente, a solução Elastic Stack ainda não pode ser considerada um SIEM, contudo, nas últimas atualizações já foram disponibilizadas várias funcionalidades de um SIEM.

Por último, o Graylog é uma solução escalável e que correlaciona eventos, possibilita a gestão dos *logs*, permite a definição de vários níveis de permissões aos utilizadores, a criação de vários utilizadores, por outro, não faz a monitorização de comportamento, não identifica vulnerabilidades e não permite a analise de riscos.

Todas as soluções elencadas na tabela anterior garantem a conformidade com o RGPD, caso se obtenha uma licença paga. Para a versão *open-source*, é necessário confirmar se as funcionalidades disponibilizadas garantem a conformidade com o RGPD. A pseudonimização pode ser feita pelo Elastic Stack (Wintergerst, Paquette, & McDiarmid, 2018), pelo Graylog (Graylog, 2018f) e pelo Splunk (Varanda, 2019). Para a solução OSSIM, tendo em conta a pesquisa efetuada, não foram encontradas referências a uma eventual forma de implementar essa medida.

As soluções Graylog, Elastic Stack e Splunk Free permitem que sejam definidos os tempos de retenção. Relativamente à solução OSSIM, devido ao seu limite em relação aos períodos de retenção reduzidos, esta questão não se aplica.

As soluções SIEM analisadas não fazem a gestão de incidentes, mas o Elastic SIEM já permite que se adicionem notas a eventos suspeitos (Settle et al., 2019). As restantes soluções, nas versões *freeware*, não fazem a gestão de incidentes, contudo, em todas elas é possível detetar incidentes de segurança.

A escolha de um SIEM *open-source* é uma tarefa exigente, logo é importante ponderar e aferir as funcionalidades que se pretendem implementar na Organização. Considerou-se importante ter uma perspetiva mais prática dos SIEM selecionados, portanto implementaram-se quatro cenários para avaliar as soluções SIEM selecionadas. A solução Splunk difere das outras soluções, uma vez que se testou a versão Enterprise do Splunk e não o Splunk Free.

Nas tabelas anteriores foi efetuado uma sistematização das funcionalidades disponibilizas por cada sistema SIEM que foi alvo de análise, tendo-se verificado que não existe nenhuma ferramenta que disponibilize todas as funcionalidades enumeradas. Tais resultados vão dificultar a seleção de uma solução para a criação do protótipo SIEM, pois todas as funcionalidades listadas nas tabelas foram consideradas importantes para a concretização dos objetivos definidos para este trabalho. Tendo em conta as informações que constam nas tabelas, foi equacionado, como solução possível para a construção do protótipo, optar pela implementação de mais do que uma solução SIEM em simultâneo, para fosse possível tirar partido de todas as funcionalidades requeridas ou por recorrer a outras ferramentas *open-source*, o que pode aumentar a complexidade na implementação.

Relativamente à perspetiva prática, considerou-se o Graylog como a solução mais intuitiva; o Elastic Stack exigiu uma curva de aprendizagem mais demorada; o OSSIM disponibiliza funcionalidades diferenciadas; e o Splunk requere uma implementação simplificada para a realização dos testes.

Escolha da solução

Devido às caraterísticas enumeradas e também pelo enorme dinamismo demonstrado ao disponibilizar melhorias e novas funcionalidades depois da realização dos cenários de testes, optou-se por escolher a solução Elastic Stack para a implementação do protótipo.

No que diz respeito ao RGPD, o Elastic Stack (a partir da versão 6.8) permite restringir o acesso dos utilizadores a determinadas informações. Esta solução permite ainda que sejam definidos vários níveis de permissão, por exemplo só os administradores têm permissão para aceder a todos os dados recolhidos/produzidos pela solução em tempo real, já os restantes utilizadores acedem à informação que é considerada relevante para o seu trabalho. Por outro lado, e ainda neste contexto, o Elastic Stack já possui uma funcionalidade (filtro *fingerprint*) que permite a pseudonimização dos dados, que é uma das medidas referenciadas no RGPD.

Como já foi referido, a solução Elastic Stack tem sofrido várias atualizações e melhorias no decorrer do desenvolvimento deste trabalho, por exemplo, a partir das versões Elastic Stack 6.8 e 7.1 estão disponíveis, na licença Basic, os recursos de segurança mais básicos, oferecendo serviços semelhantes ao Search Guard na sua versão *open-source*. O produto ReadonlyRest disponibiliza um maior número de funcionalidades de segurança para o Elasticsearch, contudo, na sua versão *open-source*, são disponibilizadas um número reduzido funcionalidades de segurança para o Kibana. A partir da versão 6.8 do Elastic Stack já é possível implementar procedimentos de segurança entre os vários elementos da solução. Outra funcionalidade muito importante que foi lançada na versão 7.2 do Elastic Stack foi a App Elastic SIEM, que pretende refletir a visão da Empresa Elasticsearch sobre um SIEM. Com a versão 7.2 do Elastic Stack foi disponibilizado o ECS (normaliza os diferentes tipos de *logs*), o que vai possibilitar a correlação de diferentes tipos de eventos. Para além disso, esta solução é escalável e flexível, permite a gestão de *logs*, contudo não faz a análise de riscos ou fornece regras para a correlação de eventos.

Já foi referido que a solução Elastic Stack não é um SIEM nativo, todavia, com recurso a outras ferramentas *open-source*, pode-se tornar num. Na Tabela 2.1 do Capítulo 2 estão listadas duas implementações com recurso a esta solução. Por exemplo, o SANS Institute possui dois cursos que implementam um SIEM *open-source* recorrendo ao Elastic Stack (SANS, 2020). Na tabela seguinte, que foi adaptada do artigo de Berman (2018), é efetuado um mapeamento das funcionalidades da solução Elastic Stack com as funcionalidades de um SIEM (Berman, 2018). Tendo em conta os dados constantes na tabela, é possível verificar que existem várias funcionalidades em que esta solução tem de recorrer a outras ferramentas *open-source* para se tornar num SIEM. É de referir que a solução, desde 2018 até 2020, já sofreu várias melhorias/alterações, desta forma, a tabela foi elaborada tendo como base a estrutura de Berman (2018), tendo sido, no entanto, foram identificadas outras funcionalidades disponibilizadas.

Tabela 3.4 – Mapeamento das funcionalidades do Elastic Stack *open-source* com um SIEM, adaptado de Berman (Berman, 2018)

Funcionalidades	**Mapeamento das funcionalidades do Elastic Stack *open-source* com um SIEM**	
	Sim/Não	Observações
Recolha de *logs*	Sim	É uma das funcionalidades estruturais da solução. Contudo para grandes volumes de dados pode ser necessário recorrer a outro componente como por exemplo o Kafka.
Processamento dos *logs*	Sim	Para que seja possível processar vários tipos de *logs* é necessário ter vários tipos de ficheiros de configuração. Configurações de filtros complexas podem afetar o desempenho do Logstash por isso é necessário monitorizar as suas pipelines.
Armazenamento	Sim	Os dados recolhidos dos diferentes componentes são armazenados no Elasticsearch que é escalável e tolerante a falhas. Todavia é necessário encontrar uma solução para arquivar os dados mais antigos, mas que possam ser acedidos rapidamente em caso de necessidade.
Pesquisas	Sim	Dependem de uma análise precisa, o que implica que é necessária experiência para obter dados de forma eficiente.
Correlação	Sim/Não	Faz a normalização entre os diversos componentes utilizando o ECS, permitindo a correlação entre os mais diversificados *logs* de segurança. Por outro lado, o Elastic Stack não fornece regras para a correlação exigindo que este trabalho seja efetuado de forma manual.
Dashboards	Sim	São extremamente poderosos, mas exigem experiência, contudo já são disponibilizados pela solução vários *Dashboards* de segurança.
Alertas	Não	É necessário recorrer a componentes *open-source* como é exemplo o ElastAlert ou até soluções comerciais.
Gestão de incidentes	Sim/Não	Na versão atual (7.9) a funcionalidade *Timelines* disponibiliza um espaço de trabalho para caça a ameaças (onde é possível efetuar filtros e adicionar notas a eventos suspeitos).

O Elastic Stack, na sua forma *open-source*, não disponibiliza a funcionalidade de *machine learning* para a investigação de anomalias, contudo, referenciamos o projeto o HELK, uma vez que já implementa uma arquitetura vocacionada para essa funcionalidade. No Anexo G – Ferramentas complementares / alternativas do Elastic Stack, são descritas ferramentas e projetos *open-source* ou freeware que podem complementar a solução Elastic Stack.

3.8. Síntese

Este capítulo teve como finalidade efetuar um estudo comparativo entre quatro soluções: Splunk Free, Elastic Stack, Graylog e OSSIM. Numa primeira fase, para que fosse possível identificar os pontos fortes e fracos de cada ferramenta, foi efetuada uma análise documental das soluções selecionadas e, numa segunda fase, foram implementados cenários de testes, para testar a facilidade de instalação, de administração e as funcionalidades disponibilizadas por cada solução. Para todos os cenários, procedeu-se a um ataque de uma máquina e, de seguida, procuraram-se indícios do ataque e foi criado um alerta em tempo real para o e-mail selecionado.

Além disso, foram enumerados os recursos, as capacidades, a arquitetura, os pontos fortes e fracos de cada solução. Foi também realizado o mapeamento, para as quatro soluções, com

as medidas técnicas sugeridas pelo RGPD relativamente ao tratamento dos dados sensíveis que foram identificadas no Capítulo 2.

Por último, fez-se um resumo comparativo das soluções tendo como base os resultados da pesquisa documental e as funcionalidades enumeradas Figura 2-4 do Capítulo 2, sugeridas pela empresa Exabeam. Na análise comparativa foram considerados pertinentes os requisitos relacionados com a arquitetura, com a escalabilidade e que promovem a segurança dos dados de uma forma mais eficaz.

É de referir que soluções Graylog e Elastic Stack possibilitam a pseudonimização, contudo, tendo como base as pesquisas que foram efetuadas, a conformidade de uma solução com o RGPD só é garantida através das suas versões pagas. Caso se pretenda implementar uma solução totalmente *open-source*, é necessário recorrer a outras ferramentas de código livre.

Através do estudo comparativo e da implementação dos cenários para as quatro soluções selecionadas, foi possível escolher a solução que vai ser utilizada para implementar o protótipo: o Elastic Stack. Apesar desta solução ter uma curva de aprendizagem demorada, compensa com a escalabilidade, a flexibilidade, pela constante evolução, na integração dos seus componentes em arquiteturas *open-source*, como são exemplo os projetos HELK e o ROCK.

Uma vez que não foram avaliadas todas as soluções SIEM *open-source* do mercado, o estudo não é representativo, no entanto, entendemos que pode ser é um contributo para esta temática, porque analisa quatro soluções *open-source* ou freeware, as suas funcionalidades e a sua conformidade com o RGPD. Outra razão para que os resultados não possam ser extrapolados prende-se com o facto de ter sido utilizado um número reduzido de ativos, ter sido apenas um elemento a avaliar as soluções e de os testes terem sido realizados nos cenários, pelo que não foram aprofundadas as funcionalidades de cada solução.

4. Implementação do protótipo

A implementação de um SIEM é bastante complexa, pois integra múltiplas funcionalidades, além disso, neste trabalho vamos focar-nos na implementação de um SIEM que atue em conformidade com o RGPD, com particular atenção à pseudonimização dos dados sensíveis e na implementação de medidas técnicas para proteção e controlo dos dados pessoais.

A implementação do protótipo possui duas abordagens: numa primeira abordagem, não serão tidas em consideração as medidas que garantem a conformidade com o RGPD e, numa segunda abordagem, serão implementadas no protótipo as referidas medias. Cada abordagem representa um modelo de implementação de um SIEM, sendo que se optou por esta divisão para fosse possível destacar as diferenças entre as duas arquiteturas.

Pretende-se que o protótipo garanta a segurança dos *logs* dos sistemas e das aplicações, que permita a realização de pesquisas que permitam a identificação de possíveis ameaças e que pseudonimize os dados sensíveis.

O protótipo vai recolher *logs* de segurança de equipamentos e de aplicações, no entanto será dada a prioridade aos *logs* gerados pelos sistemas operativos e à gestão destes. Pretende-se com este protótipo aferir o esforço que a pseudonimização requere a nível de hardware e, como prova de conceito, será testada a usabilidade do mesmo.

4.1. Caracterização da rede da entidade

Um dos objetivos deste trabalho é a implementação de um protótipo SIEM *open-source* que esteja em conformidade com o RGPD e considerou-se que seria importante basear a arquitetura do mesmo na rede de uma entidade real. Para tal, solicitámos autorização à empresa XLog com o objetivo de desenvolver parte do trabalho na sua infraestrutura. A XLog é uma empresa de consultoria na área da formação, que tem como missão a formação de alunos em tecnologias informáticas. Deixamos o apontamento que, embora o nome da empresa seja fictício, o esquema de rede é real, contudo foi desenhado de uma forma genérica por razões de segurança. Para que fosse possível desenvolver o projeto na empresa XLog, ficou acordado que todos os dados recolhidos ficariam na sua infraestrutura, devido a este facto o protótipo teve de adotar obrigatoriamente uma arquitetura *on-premise*.

Como se pode visualizar no esquema seguinte, a empresa XLog é constituída por três edifícios ligados entre si por fibra ótica. A empresa possui 100 postos de trabalho maioritariamente, tem instalado o sistema operativo Microsoft Windows. Além disso, também é constituída por um *datacenter* (faz a gestão centralizada de utilizadores e de recursos), um antivírus, uma firewall e vários servidores que gerem os mais diversos serviços. Além da rede física, existem duas redes sem fios Wi-Fi disponíveis na Organização: a "XLog" e a "Log", cuja gestão é efetuada pela controladora que está situada no *datacenter*. A controladora fornece os endereços IP's aos postos de trabalho, que são dinâmicos, não sendo possível identificar um posto de trabalho pelo seu endereço IP. Para que os serviços prestados pela empresa não sejam colocados em causa, esta possui acesso à Internet redundante.

A nível aplicacional, destacamos que a página institucional da empresa XLog disponibiliza várias informações sobre os seus serviços e na qual são publicados as notícias e campanhas. A página da empresa está alojada num servidor Linux e foi implementado em PHP e MySQL.

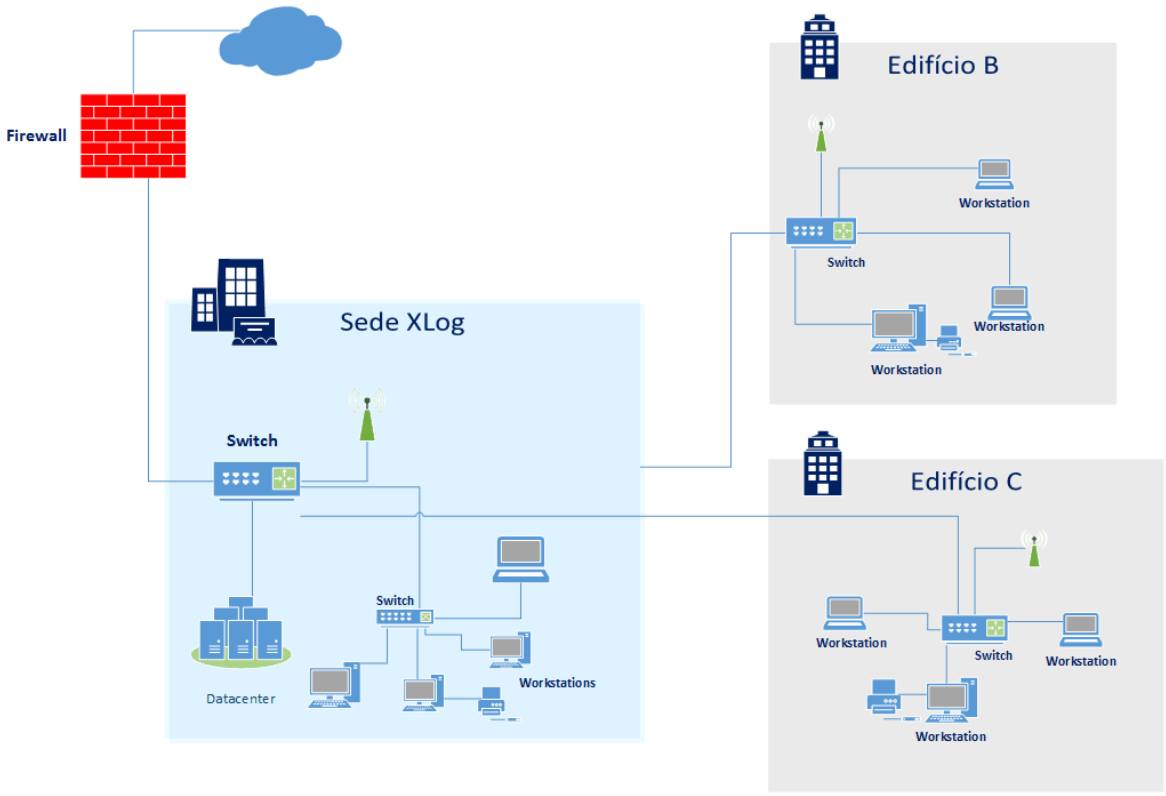


Figura 4-1 Esquema de rede da empresa XLog

4.2. Identificação dos pré-requisitos

Para que fosse possível definir os pré-requisitos a utilizar no cenário de testes foi necessário analisar os trabalhos académicos que implementam um SIEM utilizando a solução Elastic Stack. Tendo como referência a Tabela 2.1, damos o exemplo de Marquina (2018) que criou um cenário de testes no qual foi implementado um SIEM com recurso a componentes *open-source*: Elastic Stack; Wazuh (HIDS); Search Guard e o Sentinl. Bělousov (2019), também referenciado na tabela Tabela 2.1, implementa um SIEM recorrendo aos seguintes componentes *open-source* do Elastic Stack: Beats, Logstash, Kibana e Elasticsearch.

O trabalho de Marquina (2018) acrescenta um HIDS (Wazuh) à arquitetura do SIEM, contudo, no presente trabalho, o nosso foco está em garantir uma atuação em conformidade com o RGPD e, por essa razão, vamos assumir que o HIDS já está implementado (tal como acontece com a empresa XLog). O Sentinl permite a criação de alertas, contudo só é compatível com a versão 7.6.1 do Kibana (Lmangani, 2020). O Search Guard é uma ferramenta que faculta funcionalidades importantes para a arquitetura do protótipo, pelo que é um forte candidato a integrar a arquitetura do protótipo.

O levantamento dos requisitos incidiu sobre as funcionalidades que um SIEM deve possuir e na implementação das medidas técnicas para o controlo e segurança dos dados pessoais, como é exemplo a pseudonimização dos dados.

Fazendo um mapeamento entre as funcionalidades listadas no ponto 2.6.1, o estudo comparativo realizado no capítulo anterior, os objetivos definidos para este trabalho e a caraterização da rede da empresa XLog consideram-se fundamentais os seguintes requisitos:

- Recolher dados de diferentes origens;
- Garantir a compatibilidade com diferentes sistemas operativos;
- Normalizar os *logs* de todos os componentes;
- Analisar e correlacionar as informações dos equipamentos em tempo real;
- Apresentar vários *dashboards* de segurança e permitir a personalização de outros;
- Criar relatórios personalizados e alertas;
- Filtrar e destacar os eventos pela sua criticidade;
- Detetar ameaças, incidentes de segurança e vulnerabilidades;
- Emitir alertas nos casos em que existem atividades suspeitas na rede;

- Restringir os acessos e possibilitar vários níveis de permissões;
- Fornecer vários mecanismos de segurança da Organização;
- Possibilitar a pseudonimização dos dados pessoais;
- Limitar a retenção e permitir definir tempos de retenção dos dados pessoais;
- Auditar os acessos aos dados pessoais;
- Restringir o acesso aos dados pessoais;
- Garantir a segurança dos dados pessoais;
- Garantir a integridade dos dados;
- Garantir a proteção de dados por design e por padrão;

Existem dois requisitos que nesta fase da implementação não foram considerados fundamentais, no entanto, seria importante que o protótipo os disponibilizasse numa segunda fase. Os mesmos são: garantir que é efetuada uma notificação da violação de dados e que é possível serem criados relatórios de conformidade.

Enumeramos, de seguida, as medidas técnicas relevantes para a proteção e controlo dos dados pessoais, mas que não vão ser implementadas na abordagem do protótipo SIEM que não necessita estar em conformidade com o RGPD: que restrinja os acessos e possibilite vários níveis de permissões; que forneça vários mecanismos de segurança; que permita a pseudonimização dos dados; que permita definir tempos de retenção para os dados; que audite os acessos aos dados pessoais; que restrinja o acesso aos dados pessoais; que garanta a segurança dos dados pessoais; que garanta a integridade dos dados; que garanta a proteção de dados por design e por padrão.

Devido à complexidade de implementação de um SIEM, não foram implementadas todas as funcionalidades enumeradas no ponto 2.6.1 do Capítulo 2, remetemos a sua implementação para o trabalho futuro. Todavia, consideramos importante referenciar as mesmas:

- garantir a escalabilidade e a tolerância a falhas;
- garantir a resiliência e a recuperação a desastres;
- realizar análises complexas recorrendo ao *machine learning*;
- assegurar a gestão de incidentes;
- garantir a atualização em tempo real de ameaças;

- realizar análises forenses aos *logs*, em tempo real, ou aos *logs* armazenados pelo sistema;
- automatizar várias funções, reduzindo o tempo gasto na sua manutenção;
- efetuar a notificação da violação de dados e a criação de relatórios de conformidade.

Depois de ter sido efetuado o levantamento dos requisitos que o protótipo deve apresentar, passamos de seguida para descrição da implementação do protótipo.

4.3. Protótipo e cenário de testes

O ambiente de testes do protótipo é exigente a nível de Hardware e, devido a este facto, foi implementado na infraestrutura de rede da empresa XLog, contudo foi segregado para uma VLAN de testes.

Neste ponto, são apresentadas duas abordagens, sendo que cada uma representa um modelo de implementação do protótipo SIEM, neste caso, numa das abordagens é apresentada a implementação sem as medidas técnicas para a proteção e controlo dos dados pessoais. A outra abordagem tem em consideração as medidas técnicas anteriormente referidas. Nos cenários de testes foram recolhidos *logs* de segurança dos sistemas operativos Microsoft Windows e Linux.

De uma forma resumida, pretende-se comparar as duas abordagens de implementação do protótipo SIEM e a confirmação da usabilidade dos pré-requisitos definidos no ponto 4.2.

4.3.1. Arquitetura

Tendo como base o esquema de rede da empresa XLog (Figura 4-1), adaptou-se o cenário de testes do Capítulo 3 (Figura 3-6), ao qual foi adicionado o servidor web e foi alterada a versão do sistema operativo do Microsoft Windows 7 para Microsoft Windows 10 (Figura 4-2). Para o controlo de segurança, a empresa XLog possui uma firewall, um antivírus e várias políticas de segurança implementadas nos servidores, todavia, para que fosse exequível implementar o protótipo, optou-se por implementar uma arquitetura *On-Premise* simplificada, mas que represente os dois sistemas operativos utilizados na empresa: o Linux e o Microsoft Windows.

É que referir que na arquitetura da abordagem de implementação na qual os dados pessoais nos *logs* vão ser pseudonimizados foi acrescentado um segundo servidor, que vai

conter o índice com os dados pessoais (valor do campo original e o valor de *hash*). No esquema também está representado o servidor que recolhe as métricas das máquinas do protótipo. Com a exceção da máquina que possui o sistema operativo Microsoft Windows, recorreu-se à distribuição Ubuntu para instalar os diversos componentes da solução.

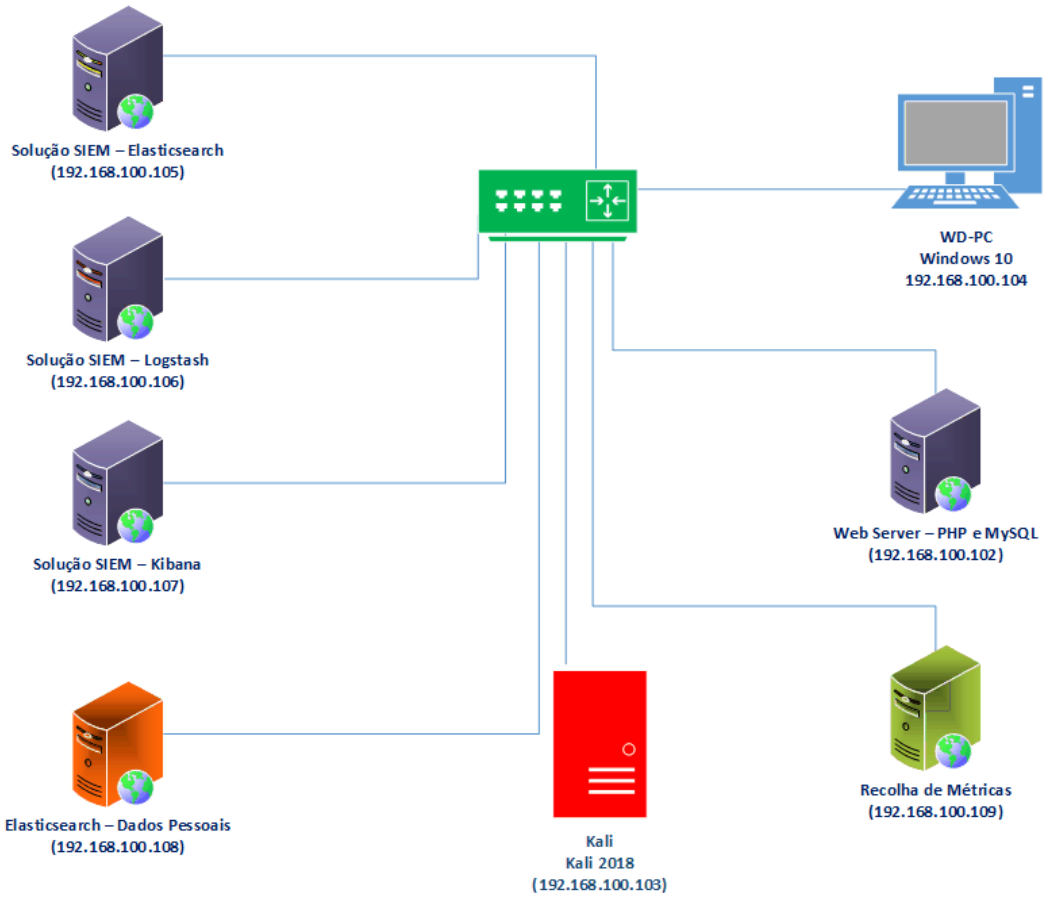


Figura 4-2 Arquitetura proposta para o protótipo

O protótipo foi implementado na rede da empresa XLog, numa VLAN de testes com máquinas virtuais que foram especificamente configuradas para a realização do presente trabalho.

4.3.2. Recolha de eventos de segurança

Relativamente à recolha de eventos, foi necessário tomar algumas decisões, porque não é possível ativar todos os eventos em simultâneo, devido ao facto de que seria muito difícil efetuar a sua gestão, no entanto, devem ser ativados os eventos que permitam identificar um incidente de segurança. As referências consultadas (artigos de opinião, congressos e tutorais) não apresentam uma fórmula específica para este processo, contudo, existem várias políticas que devem ser ativadas para que seja possível fazer o rastreamento de um incidente.

No presente trabalho, optou-se por ativar no sistema operativo Microsoft Windows 10 a maior parte dos eventos sugeridos pela ferramenta LOG-MD Free Edition (Imfsecurity, 2020), cujo objetivo é o de ajudar os utilizadores a decidir quais os *logs* que devem ser recolhidos. Nas Figura 4-3 e Figura 4-4 pode-se visualizar o inventário realizado às políticas da máquina Microsoft Windows utilizada no cenário de testes do protótipo (Figura 4-2) pela ferramenta LOG-MD Free Edition. As imagens seguintes mostram parte dos resultados da auditoria antes e depois da ativação de várias das políticas sugeridas pela ferramenta LOG-MD.

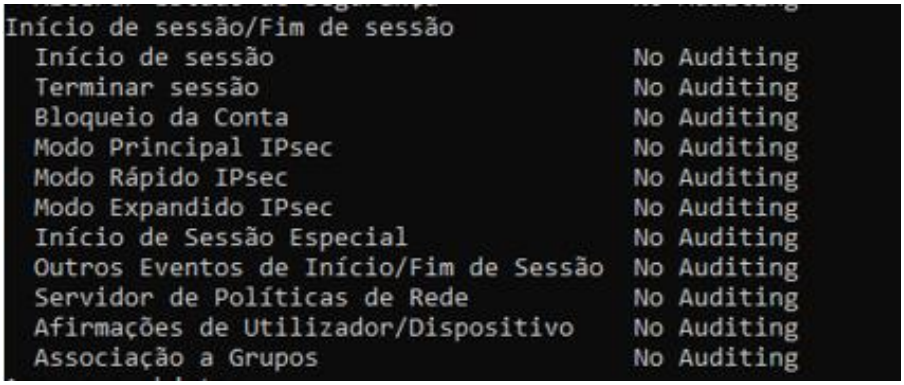


Figura 4-3 Auditoria feita pelo LOG-MD Free Edition às políticas da máquina Microsoft Windows antes de serem ativadas as políticas

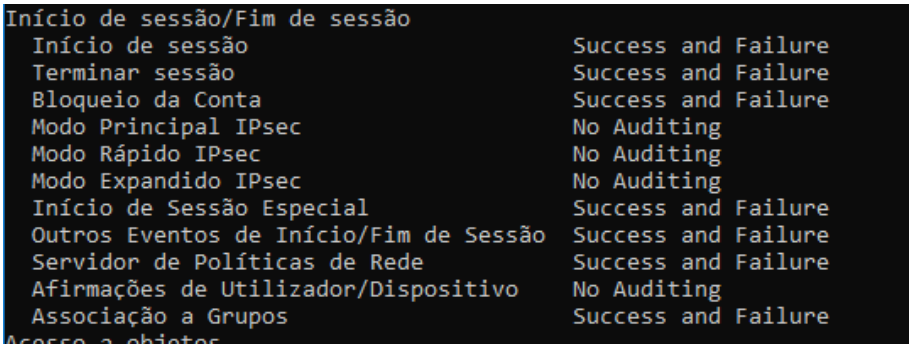


Figura 4-4 Auditoria feita pelo LOG-MD Free Edition às políticas da máquina Microsoft Windows depois de serem ativadas as políticas

Como foi referido no anexo do ponto 2.5.2 do Capítulo 2, para serem detetados ataques mais complexos recorre-se ao Sysmon. Por essa razão, o Sysmon também foi instalado na máquina com o sistema operativo Microsoft Windows do cenário de testes do protótipo, esta ferramenta foi configurada com recurso ao ficheiro de configuração criado pela SwiftOnSecurity, que permite rastrear comportamentos suspeitos. A solução Elastic Stack já contempla a utilização da ferramenta Sysmon e os *logs* podem ser recolhidos pelos Beats.

Relativamente aos *logs* dos servidores com o sistema operativo Linux, optou-se por recolher os *logs* do sistema através do agente Filebeat, em detrimento de outros agentes, pois o primeiro já faz parte da solução Elastic Stack e pode-se tirar partido da funcionalidade ECS, que normaliza os *logs* dos diferentes componentes e que permite que seja efetuada a correlação entre eles.

Contudo, após terem sido realizados alguns testes no cenário implementado no ponto 3.6.2, considerou-se necessário auditar a linha de comando dos utilizadores do servidor Web. Assim, foi instalado no servidor web a ferramenta Snoopy Logger[55], cuja principal funcionalidade é a de recolher todos os comandos executados pelos utilizadores na linha de comando.

4.3.3. Medidas de segurança

Na implementação do protótipo, vamos assumir que estão acauteladas as medidas de segurança recomendadas para os servidores Linux, por exemplo, a encriptação do disco, uma password forte, a definição de vários níveis de permissões, as atualizações, a desativação do login *root* ou a desativação dos serviços que não a ser utilizados, uma vez que estas medidas já eram asseguradas pela empresa.

O Elastic Stack, na licença Basic, disponibiliza vários recursos de segurança para todos os seus componentes: Beats, Logstash, Elasticsearch e o Kibana. Quando se instala o Elastic Stack, a encriptação não fica ativa, ou seja, os dados e as passwords são enviados em "*texto simples*" e podem ser visualizados por utilizadores não autorizados (toda a informação pode ser visualizada, como ilustra a Figura 4-5).

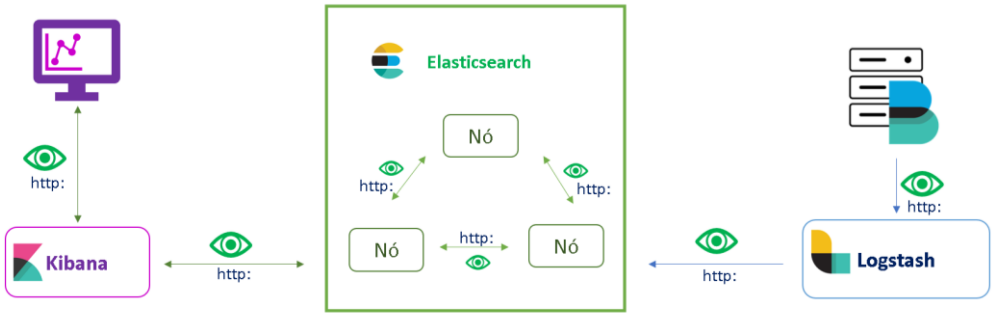


Figura 4-5 – Esquema das comunicações do Elastic Stack sem encriptação, esquema adaptado de (Elastic, 2019)

[55] https://github.com/a2o/snoopy

O Elastic Stack permite que a segurança seja implementada por fases: pode-se bloquear os acessos não autorizados aos dados, aplicar a encriptação às comunicações entre os nós de um cluster e aplicar a encriptação fora do cluster. Na Figura 4-6 é apresentada a implementação da encriptação dos dados em todas as comunicações realizadas pela solução Elastic Stack. Garantindo, assim, que só os utilizadores autorizados acedem aos dados contidos nas várias comunicações.

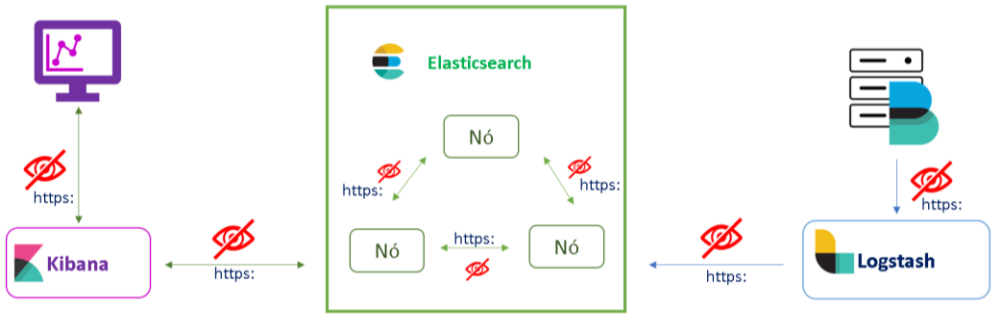


Figura 4-6 – Esquema das comunicações do Elastic Stack com encriptação, esquema adaptado de (Elastic, 2019)

A encriptação foi ativada nos diversos componentes, porém, esta operação exigiu que fossem feitas várias configurações, as quais estão detalhadas no Anexo H – Elastic Stack encriptação. É de referir que a realização desta tarefa exigiu a dedicação de vários dias, pois foi necessário criar um certificado e, como este não era fiável, a solução considerava que a ligação não era segura e não permitia a autenticação. O "*debug*" ao problema foi demorado, tendo sido sendo necessário consultar o blog do Elastic, do Stack Overflow, o Youtube, entre muitas outras fontes, para o problema ter sido resolvido.

Após inúmeras tentativas infrutíferas, percebeu-se que era necessário desativar a verificação do certificado, como se pode visualizar na Figura 4-7, na qual se apresenta o resultado do teste à configuração do Metricbeat.

```
informatica@kibana:/etc/metricbeat$ sudo metricbeat test output
elasticsearch: https://11.0.50.22:9201...
  parse url... OK
  connection...
    parse host... OK
    dns lookup... OK
    addresses: 11.0.50.22
    dial up... OK
  TLS...
    security... WARN server's certificate chain verification is disabled
    handshake... OK
    TLS version: TLSv1.3
    dial up... OK
  talk to server... OK
  version: 7.9.0
```

Figura 4-7 – Resultado do teste à configuração do Metricbeat

4.3.4. Medidas de segurança para incrementar a conformidade com o RGPD

Para a abordagem na qual se pretende implementar as medidas técnicas de proteção e controlo dos dados pessoais foi fundamental encontrar uma solução *open-source* que permitisse efetuar auditoria às consultas efetuadas aos dados pessoais. Era necessário também que fosse possível identificar o utilizador que realizou a pesquisa e a data do acesso aos dados pessoais.

Na pesquisa documental efetuada no Capítulo 3, foram identificadas duas soluções que disponibilizam, de forma gratuita, estas medidas de segurança: o Search Guard e o plugin Elasticsearch da ReadonyRest[56]. Após ter sido efetuada uma pesquisa mais detalhada a estas duas soluções, verificou-se a que empresa Floragunn GmbH, que disponibiliza a solução Search Guard, está a ser processada pela Elastic por violação de propriedade intelectual (Banon, 2019a, 2019b), pelo que se optou por selecionar o plugin Elasticsearch, da ReadonlyRest.

O ReadonlyRest disponibiliza dois produtos deste tipo: o plugin Elasticsearch e o plugin Kibana. O plugin Elasticsearch adiciona vários níveis de segurança ao Elastic Stack e encripta os dados que estão em transporte entre os diferentes componentes: Beats, Logstash, Elasticsearch e o Kibana. Este plugin disponibiliza várias funcionalidades, entre as quais destacamos as seguintes: a criação de grupos e de utilizadores, a autenticação LDAP e a segregação da informação (ReadonlyREST, 2018b). São também disponibilizadas três tipos de licenças, a saber (ReadonlyREST, 2018a): Free (Elasticsearch), Pro (Elasticsearch e Kibana) e Enterprise (Elasticsearch e Kibana).

4.3.5. Descrição da implementação do sistema SIEM *open-source*

Todas as máquinas utilizadas no cenário do protótipo são máquinas virtuais, como se pode visualizar na Tabela 4.1 existem cinco máquinas com o sistema operativo anfitrião Microsoft Windows 10 e uma com o VMware ESXi 6.7. Para os servidores, utilizou-se o sistema operativo Ubuntu (várias versões) e para a máquina cliente o sistema operativo Microsoft Windows 10.

[56] https://readonlyrest.com/

Ainda como se pode observar na Tabela 4.1, as máquinas que têm mais memória são as máquinas virtuais nas quais foi instalado o Elasticsearch e o Kibana, porque são as que mais recursos requerem a esse nível.

Tabela 4.1 – Especificações técnicas da implementação do protótipo

<table>
<tr><th rowspan="2">Especificação do Hardware</th><th colspan="4">Especificação do Software</th></tr>
<tr><th>Sistema operativo Anfitrião</th><th>Software de virtualização</th><th>Caraterísticas das máquinas virtuais</th><th>Software instalado</th></tr>
<tr><td rowspan="2">Desktop Intel(R) Core(TM) i3-8100 CPU @ 3.60GHz
RAM16,0GB
SSD 240 GB
HDD 1 TB
HDD 4 TB</td><td colspan="2" rowspan="2">VMware ESXi™ 6.7</td><td>8,0 GB RAM
500 GB HDD
Ubuntu 18.04 LTS</td><td>Elasticsearch
Elastalert
Netdata

Auditbeat
Filebeat
Metricbeat
Packetbeat</td></tr>
<tr><td>5,0 GB RAM
200 GB HDD
Ubuntu 18.04 LTS</td><td>Logstash
Netdata

Auditbeat
Filebeat
Heartbeat
Metricbeat
Packetbeat</td></tr>
<tr><td>Desktop Intel(R) Core (TM) i7-3770 CPU @ 3.40GHz
RAM 12,0 GB
SSD 240 GB
HDD 500 GB</td><td>Microsoft Windows 10 Pro</td><td>Oracle VM VirtualBox Manager 6.1</td><td>8,0 GB RAM
200 GB HDD
Ubuntu 18.04 LTS</td><td>Kibana
Netdata

Auditbeat
Filebeat
Metricbeat
Packetbeat</td></tr>
<tr><td>Desktop Intel(R) Core (TM) i5CPU 760@ 2.80GHz
RAM 6,0 GB
HDD 1 TB</td><td>Microsoft Windows 10 Pro</td><td>Oracle VM VirtualBox Manager 5.2</td><td>3,0 GB RAM
50 GB HDD</td><td>Microsoft Windows 10

Auditbeat
Metricbeat
Packetbeat
Winlogbeat
Sysmon</td></tr>
<tr><td>Desktop Intel(R) Core (TM) i5CPU 760@ 2.80GHz
RAM 6,0 GB
HDD 2 TB</td><td>Microsoft Windows 10 Pro</td><td>Oracle VM VirtualBox Manager 5.2</td><td>4,0 GB RAM
120 GB HDD
Ubuntu 18.04 LTS</td><td>Servidor Web
Netdata

Auditbeat
Filebeat
Heartbeat
Metricbeat
Packetbeat
Snoopy</td></tr>
<tr><td>Portátil Asus – BU403UA
Intel(R) Core (TM) i7-6500U CPU @ 2.50GHz
RAM 12,0 GB
SSD 240 GB
HDD 1 TB</td><td>Microsoft Windows 10 Pro</td><td>Oracle VM VirtualBox Manager 5.1</td><td>3,0 GB RAM
130 GB HDD
Kali 2020.3</td><td>Pupy</td></tr>
<tr><td rowspan="2">Desktop Intel(R) Core (TM) i7-3770 CPU @ 3.40GHz
RAM 16,0 GB
SSD 500 GB</td><td rowspan="2">Microsoft Windows 10 Pro</td><td rowspan="2">Oracle VM VirtualBox Manager 6.1</td><td>6,0 GB
200 GB SSD
Ubuntu 20.04 LTS</td><td>Prometheus
Grafana</td></tr>
<tr><td>7,0 GB
200 GB
Ubuntu 20.04 LTS</td><td>Elasticsearch
Kibana
ReadonlyRest</td></tr>
</table>

No resumo comparativo realizado na Tabela 3.3, do capítulo 3 deste trabalho, verificamos que a solução Elastic Stack não disponibiliza as funcionalidades relativas à elaboração de relatórios ou à criação de alertas. Como as duas funcionalidades fazem parte dos pré-requisitos necessários, foi efetuada uma pesquisa documental com o objetivo de fazer um levantamento das ferramentas *open-source ou freeware* que ofereciam as funcionalidades pretendidas. Assim, tendo em conta os resultados dessa pesquisa, optou-se por adicionar à arquitetura a ferramenta *open-source* Elastalert[57] para a criação dos alertas, como se pode visualizar a Figura 4-9. Para a criação de relatórios, não se encontrou uma ferramenta compatível com a versão instalada do Kibana a versão 7.8.

O Elastalert faz a integração com várias ferramentas, entre as quais o Slack. Escolheu-se o Slack porque esta plataforma disponibiliza de forma gratuita um conjunto de serviços, tendo-se criando um canal para listar e tratar os alertas recebidos.

Para que se pudesse ter uma visão geral da rede informática, foi necessário recolher os *logs* dos seus diversos elementos constituintes, neste caso específico, foram instalados nas máquinas (cliente/servidor) seis tipos de Beats: o Auditbeat, o Filebeat, o Heartbeat, o Metricbeat, o Packetbeat e o Winlogbeat.

Na máquina com o sistema operativo Microsoft Windows, o Auditbeat foi utilizado para monitorizar os acessos aos processos e aos ficheiros. No sistema operativo Linux, o Auditbeat recolheu os mesmos dados que o *auditd*, que é utilizado para monitorizar a atividade dos utilizadores.

O Winlogbeat permitiu monitorizar os acessos efetuados pelos utilizadores, assim como todos os eventos gerados pela máquina. Como já foi referido, para que se possa detetar ataques informáticos mais complexos, procedeu-se a instalação do Sysmon.

Os *logs* do sistema operativo Linux foram recolhidos pelo Filebeat, sendo que este Beat envia os dados contidos nos ficheiros de *logs* para o Elasticsearch. Como os *logs* do sistema operativo Linux são ficheiros de texto, a solução recomenda a utilização do Filebeat. Além disso, foi também instalado o Snoopy que guarda os comandos realizados na linha de comandos.

[57] https://github.com/Yelp/elastalert

Nos servidores Linux, nos quais se encontra o Logstash e o servidor Web, também foi instalado o Heartbeat, o que permitiu examinar a disponibilidade do Elasticsearch ou dos serviços do servidor Web.

Para aferir os vários tipos de métricas das aplicações ou dos sistemas operativos (CPU ou RAM das máquinas), foi utilizado o Metricbeat nos dois sistemas operativos Linux e Microsoft Windows. O Packetbeat foi utilizado para monitorizar a rede dos dois sistemas operativos. Na Figura 4-8 são identificados a máquina e os Beats contidos na arquitetura.

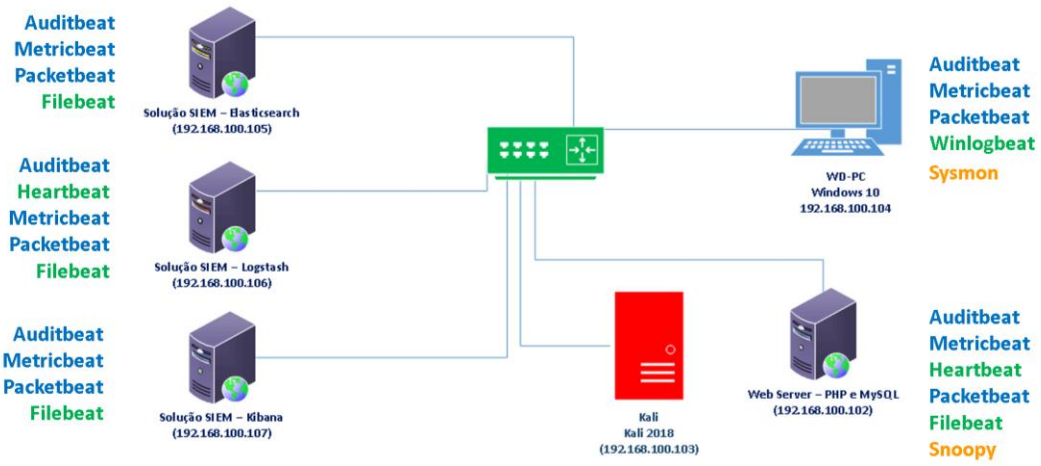


Figura 4-8 – Esquema que identifica na arquitetura os Beats instalados

O Logstash, o Elasticsearch e o Kibana são componentes do Elastic Stack, e, de uma forma muito sucinta, é possível referir que o Logstash recolhe os dados dos beats e, através de pipelines, transforma-os e envia-os para o Elasticsearch. O Elasticsearch é um mecanismo de análise, de pesquisa e de armazenamento que recebe os dados dos Beats e do Logstash. Finalmente, o Kibana é a interface Web da solução, através da qual se faz a gestão e a visualização dos dados guardados no Elasticsearch.

O Elastalert cria alertas tendo como base a informação que está no Elasticsearch, para tal, através deste componente são realizadas pesquisas periódicas ao Elasticsearch e, caso encontre uma correspondência com as regras definidas, é emitido um alerta. Finalmente, o Slack permite visualizar todos os alertas criados pelo Elastalert. Na Figura 4-9 é apresentado o diagrama dos componentes que integram o protótipo e o fluxo dos dados.

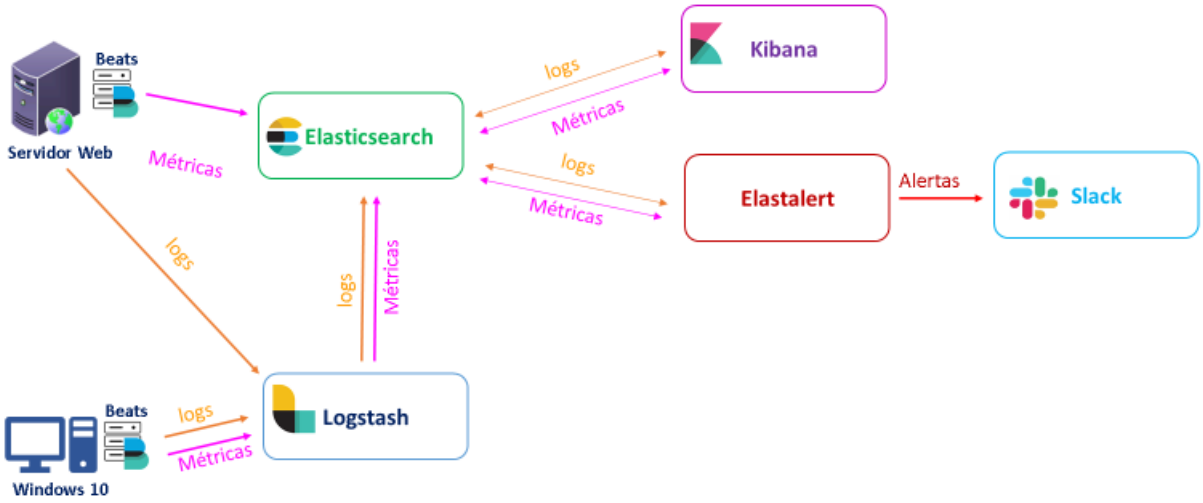


Figura 4-9 – Diagrama dos componentes do protótipo sem a pseudonimização

Como se pode observar no diagrama anterior, aos componentes da solução Elastic Stack, foram adicionados o Elastalert e o Slack para que fosse possível criar alertas e guardar o seu histórico. As restantes funcionalidades são asseguradas pelo Elastic Stack, como por exemplo, a autenticação e a encriptação. De seguida, será detalhado o funcionamento das pipelines do Logstash, porque a pseudonimização será efetuada através de uma pipeline do Logstash, além disso, também será descrito o processo de instalação do Elastic Stack e do Elastalert.

Logstash

O Logstash é uma pipeline de processamento de dados que recebe dados dos vários agentes, e, caso seja necessário, transforma-os e, depois, envia-os para Elasticsearch. O ficheiro de configuração é composto por três elementos distintos: *input*, *filter* e *output*. Contudo, só dois elementos são obrigatórios (entradas e saídas), pois os plugins filtro são opcionais (Elastic, 2020c). Na Figura 4-10 é ilustrada a ordem pela qual os elementos são executados: os plugins Entrada permitem definir a origem dos *logs*, os plugins Filtros especificam a forma de analisar e transformar os *logs* e os plugins de Saída definem o seu formato e destino.

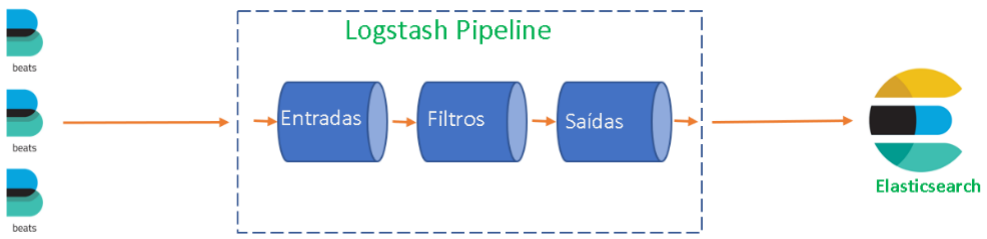


Figura 4-10 – Pipeline do Logstash, adaptado de (Elastic, 2020c)

Quando se configura o Logstash, é necessário testar o serviço, sendo que é possível testar o serviço através da linha de comando. A configuração efetuada na linha de comando é a mais elementar que uma pipeline pode assumir.

```
#comando para testar o serviço do logstash
bin/logastash -e 'input { stdin { } } output { stdout {} }
```

Figura 4-11 – Execução de uma pipeline através da linha de comados

Nas duas abordagens de implementação foi utilizado o plugin Beats como entrada e o plugin Elasticsearch como saída. A título de exemplo, apresentamos o conteúdo do ficheiro de configuração da implementação sem a pseudonimização:

```
input {
       beats {
              port => 5043
              id => beats
              add_field => { "[@metadata][typeso]" => "host" }
       }

       beats {
              port => 5042
              id => beats_server
              add_field => { "[@metadata][typeso]" => "server" }
       }
}
output {
              elasticsearch {
                     #IP do primeiro servidor
                     hosts => ["https://IP_Elastic:9201"]
                     user => "elastic"
                     password => "******"
                     index => "%{[@metadata][beat]}-%{[@metadata][version]}-
%{+yyyy.MM.dd}"
                     id => elasticsearch_saida_server
                     ssl => true
                     ssl_certificate_verification => false
                     cacert => "/etc/logstash/client-ca.cer"
       }
}
```

No ficheiro de configuração existem dois plugins de entrada com portos diferentes, o que vai permitir efetuar operações diferentes na pipeline. Através do porto é possível decidir quais os *logs* a pseudonimizar.

Instalação da solução Elastic Stack e manutenção

Na instalação da solução Elastic Stack requereu que fossem instalados os componentes Elastic Stack: Elasticsearch, Kibana, Logstash e os Beats, sendo que esta operação está descrita no Anexo I – Manual de instalação Elastic Stack.

Na solução foi necessário ativar as seguintes funcionalidades: a encriptação, a criação de *dashboards*, a monitorização na solução e a configuração dos Beats, para que fosse possível enviar dados para App Elastic SIEM que, entretanto, foi denominada de *Security*. Devido à complexidade de algumas operações, a curva de aprendizagem foi demorada.

Instalação do Elastalert e manutenção

Optou-se por instalar o Elastalert no servidor no qual já se encontrava o Elasticsearch, no entanto, a ferramenta permite que este possa ser instalado numa outra máquina. A instalação do Elastalert está descrita no Anexo J – Manual de instalação Elastalert, sendo que também foi necessário configurar a parte da segurança, uma vez que a encriptação e a autenticação já estavam configuradas no protótipo.

Para verificar se o Elastalert estava a funcionar de forma adequada foi necessário criar uma regra. Como se pode ver no fragmento do código da regra, foi criado um alerta, baseado na frequência, para o índice ou índices *winlogbeat*-*. A aplicação desta regra vai motivar que seja criada uma mensagem de alerta se, no intervalo temporal de uma hora, ocorrerem três tentativas de autenticação falhadas. Para uma melhor compreensão do que foi referido, as configurações estão destacadas a negrito.

```
# (Required)
# Index to search, wildcard supported
index: winlogbeat-*

# (Required, frequency specific)
# Alert when this many documents matching the query occur within a timeframe
num_events: 3

# (Required, frequency specific)
# num_events must occur within this amount of time to trigger an alert
timeframe:
  hours: 1

# (Required)
# A list of Elasticsearch filters used for find events
# These filters are joined with AND and nested in a filtered query
#                       For                       more                       info:
http://www.elasticsearch.org/guide/en/elasticsearch/reference/current/query-
dsl.html
filter:
  - term:
        winlog.logon.failure.status: "This is either due to a bad username or
authentication information"
```

Para testar a regra na máquina Microsoft Windows foi digitada propositadamente uma password errada três vezes, tendo-se verificado que foi criada automaticamente uma

mensagem de alerta que foi remetida pelo Elastalert para o Slack, como se ilustra na Figura 4-12.

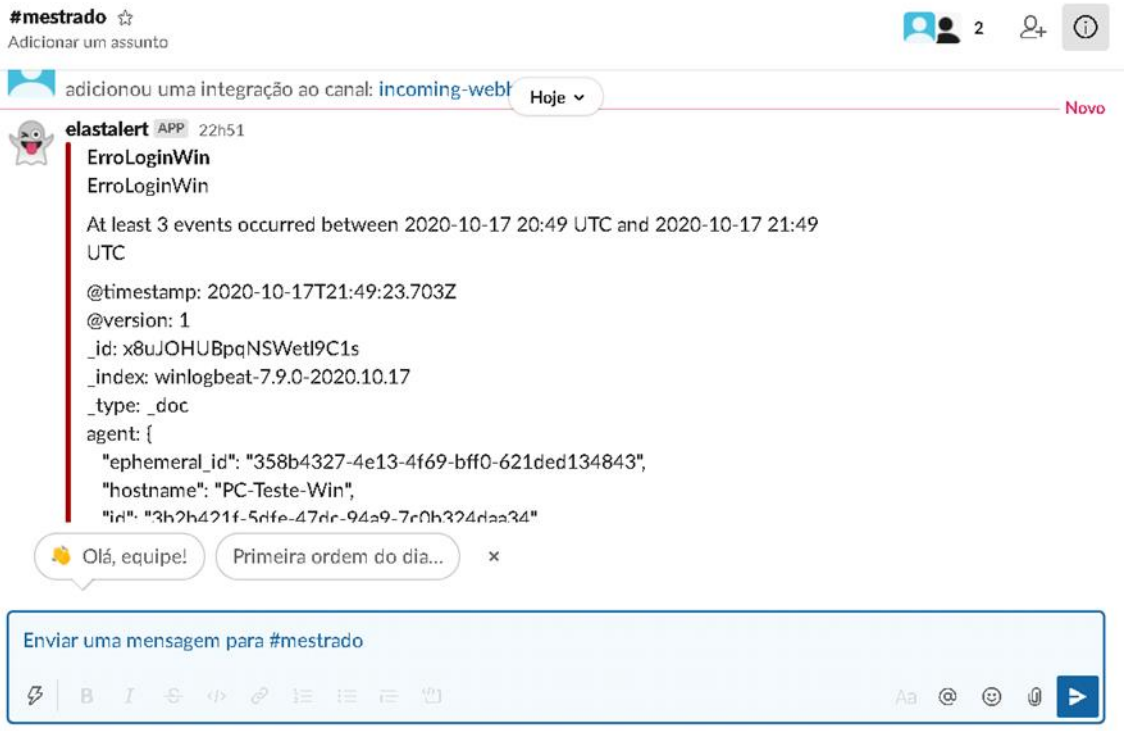


Figura 4-12 – Exemplo de um alerta na plataforma Slack

Posteriormente foi alterada a configuração do alerta para não fossem enviados os dados dos *logs* no campo da mensagem. Com recurso ao Slack, é possível ter um histórico dos alertas recebidos e serem analisadas as decisões tomadas.

4.3.6. Descrição do sistema SIEM *open-source* em conformidade com o RGPD

O RGPD recomenda que se faça a anonimização ou a pseudonimização dos dados, pelo que no presente estudo se considerou importante que, em caso de violações de segurança ou situações anómalas, fosse possível identificar o equipamento, o utilizador ou o endereço de IP de origem, por isso, optou-se por pseudonimizar os dados pessoais.

Como se pode observar na Figura 4-13, foi utilizado o digrama base apresentado na secção anterior, no entanto, considerou-se importante que fossem separados os dados que vão ser auditados dos dados que podem ser visualizados por pessoas autorizadas, para tal instalou-se um segundo Elasticsearch, um segundo Kibana e também o plugin Elasticsearch da ReadonyRest. A pseudonimização vai ser assegurada pelo Logstash, com recurso a uma pipeline, sendo que as mensagens com os dados pessoais pseudominizadas são enviadas para o primeiro servidor no qual podem ser pesquisadas e tratadas. As mensagens de recuperação

(possuem sempre o campo original e valor do *hashes*) são enviadas para o segundo servidor, no qual é possível auditar os acessos.

O plugin Elasticsearch da ReadonyRest permite adicionar uma nova camada de segurança ao índice no qual estão guardados os campos que permitem a recuperação, ou seja, através da informação contida nos campos é possível identificar o utilizador e a máquina responsável por determinada ocorrência.

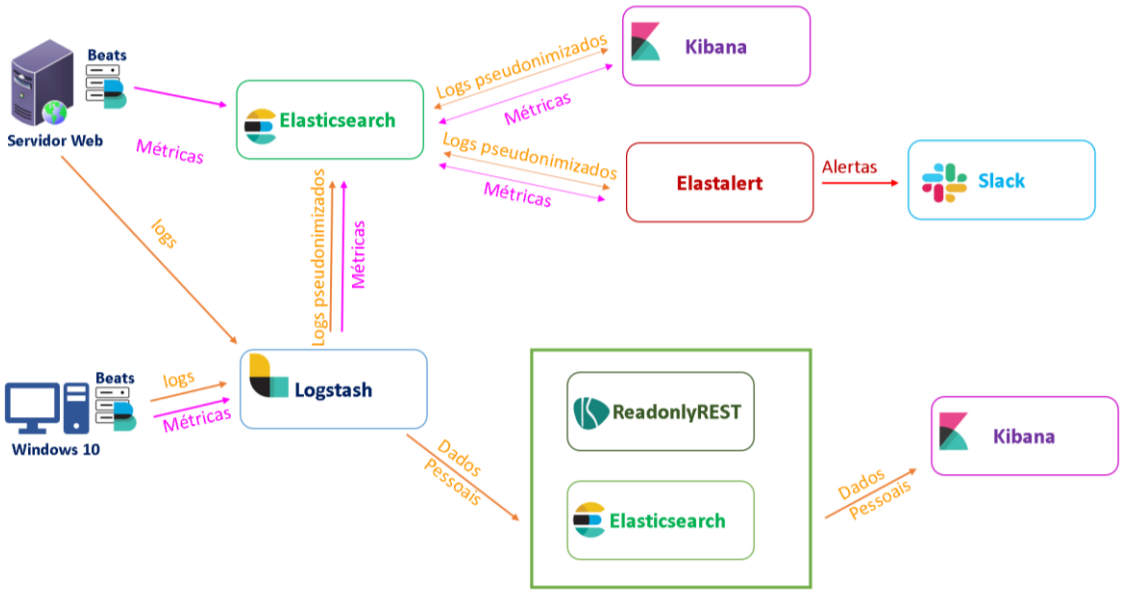


Figura 4-13 – Diagrama dos componentes do protótipo com a pseudonimização

Nos próximos pontos serão detalhados o processo de pseudonimização e também as configurações efetuadas para que fosse possível auditar as operações realizadas pelos utilizadores com acesso autorizado ao índice que contém os dados de recuperação.

Pseudonimização

Um dos requisitos do protótipo é a pseudonimização dos dados, tendo-se optado por realizar esta operação na fase de ingestão, procurando, assim, tirar proveito das funcionalidades que as pipelines do Logstash disponibilizam.

Tendo como campo de ação a recolha de *logs* de segurança dos sistemas operativos Microsoft Windows e Linux, optou-se por pseudonimizar o nome da máquina, o utilizador e o endereço IP origem. A escolha destes campos prende-se com o facto de os mesmos poderem permitir a identificação de um Titular e de estarem presentes nos *logs* recolhidos pelo protótipo.

Para que fosse possível criar o ficheiro de configuração da pipeline, que vai pseudonimizar os dados pessoais, foi necessário realizar um levantamento dos campos a pseudonimizar nos *logs* dos dois sistemas operativos. Na Tabela 4.2 é apresentado o mapeamento entre os diferentes índices dos Beats e os campos que podem conter o nome da máquina, do utilizador e do endereço IP origem. Embora só se faça a pseudonimização dos campos contidos no Winlogbeat, pois pretende-se obter uma prova de conceito, o levantamento foi feito para todos os Beats utilizados no protótipo. É de referir que na máquina com o sistema operativo Microsoft Windows não foi instalado o e-mail e outras aplicações que são utilizadas como ferramentas de trabalho, sendo assim, o número de campos a pseudonimizar pode variar e é necessário pesquisar nos seus *logs* e verificar se é necessário pseudonimizar outros campos. Esta situação ocorre porque o ECS que permite normalizar os dados dos *logs* das diferentes origens possuem inúmeros campos que podem não ter sido preenchidos nos *logs* da máquina Microsoft Windows do protótipo.

Tabela 4.2 – Listagens dos campos a pseudonimizar

Beat	Listagem dos campos que identificam a máquina, o utilizador e o IP origem		
	Máquina	**Utilizador**	**IP origem**
Auditbeat	message agent.hostname agent.name host.hostname host.name user.name	message user.audit.name user.effective.group.name user.effective.name user.filesystem.group.name user.filesystem.name user.group.name user.name user.saved.group.name user.saved.name	message host.ip
Filebeat	message agent.hostname agent.name host.hostname host.name message	message related.user user.name	message host.ip monitor.name
Heartbeat	message agent.hostname agent.name observer.hostname	message	message observer.ip monitor.ip
Metricbeat	message agent.hostname agent.name host.hostname host.name	message user.name	hessage host.ip
Packetbeat	message agent.hostname agent.name host.hostname host.name	message	message destination.ip host.ip path related.ip server.ip
Winlogbeat	message agent.hostname agent.name dns.question.name host.hostname host.name related.user source.domain user.name winlog.computer_name winlog.event_data.SubjectUserName winlog.event_data.TargetDomainName	message related.user user.name winlog.event_data.TargetUser Name winlog.event_data.SubjectUser Name	message host.ip

As funções de *hash* são um algoritmo matemático que transformam um conjunto de dados de comprimento variável num conjunto de dados de comprimento fixo, sendo que os valores dos dados devolvidos pela função podem ser chamados de *hashes* ou valor de *hash*.

O filtro *fingerprint* permite que sejam criados *hashes* do valor do campo que se pretende pseudonimizar, este plugin pode ser configurado para utilizar as seguintes funções de *hash*: SHA1, SHA256, SHA384, SHA512, ou MD5. Por norma, o parâmetro da configuração "método" está definido com o SHA1, contudo a documentação sobre este tema recomenda que este não seja utilizado, pois pode ser considerado inseguro. A solução aconselha a

utilização do SHA256, porque é aquela que oferece um bom equilíbrio entre a robustez e a performance, devido a esse facto, optou-se por utilizar a função de *hash* SHA256.

A utilização do "*salt*" no armazenamento das senhas é sempre recomendado, por isso é aconselhada a utilização de uma chave. Quando se inclui uma chave no filtro *fingerprint* a função HMAC é usada, é de referir que a função HMAC é um método robusto para criar *hashes*, pois o mesmo incorpora um segredo adicional com recurso a uma chave. No excerto do ficheiro de configuração da pipeline pode ser visualizado um exemplo da utilização do filtro *fingerprint*. Como se pode constatar, o método definido foi o "**SHA256**" e também foi definida uma chave, o significou que a função HMAC foi utilizada.

```
fingerprint {
       source => "[agent][name]"
       target => "[@metadata][fingerprints]"
       method => "SHA256"
       key => "HMAC-CHAVE"
       id => fingerprint
}
```

Na Figura 4-14 é ilustrado o processo de pseudonimização dos dados, sendo que a mensagem de *log* é recebida na pipeline, nesta são verificados quais os campos que são passiveis de ser pseudonimizados. No esquema é ilustrado que foram pseudonimizados os dados relativos ao nome, ao utilizador e ao IP, sendo que, para cada um, vão ser geradas *hashes* e que o valor de *hash* vai substituir o valor original do campo. Para cada campo que foi pseudonimizado, vai ser enviado o valor original e o valor da *hash* para um índice separado (*data_key*) que está alojado no servidor secundário. O *log*, já com os campos pseudonimizados, é enviado para o servidor principal no qual pode ser visualizado e tratado.

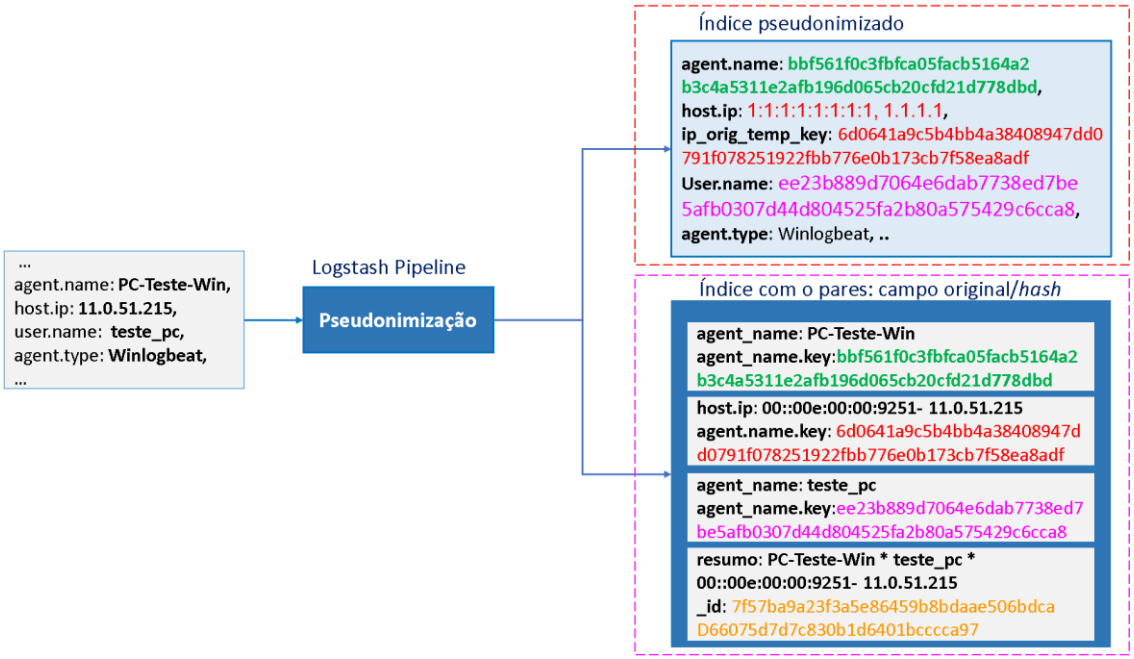


Figura 4-14 – Esquema do processo da pseudonimização

Não foi só o filtro *fingerprint* que foi utilizado na operação pseudonimização, de seguida serão explicados quais os filtros utilizados. Para que fosse possível realizar uma cópia do *log*, recorreu-se ao filtro *clone*. Como se pode visualizar, foi dado um nome à nova cópia do *log* e foi adicionada uma *tag* (etiqueta).

```
clone {
       clones => [ "clone_name_equip "]
       add_tag => [ "clone_name_equip"]
}
```

Para que fosse possível manter determinados campos do *log* copiado pelo filtro clone e descartar outros, recorreu-se ao filtro *prune* com a opção *whitelist*. No ficheiro de texto seguinte é possível verificar que são descartados todos os campos, exceto o campo *host.ip*, campo *tags* e o campo *@timestamp*, sendo que de seguida foram adicionados os campos contêm o valor original/valor *hash*. Foi sempre criado o campo original e campo do valor do *hashes*, além disso, foi também criado um campo resumo com todos os valores dos campos pseudonimizados. Nos *logs* enviados não podem existir os três campos (nome, utilizador e IP) em simultâneo, sendo que para os campos que não estão presentes foi-lhe atribuído o valor de "–".

```
prune {
       interpolate => true
       whitelist_names => [ 'host.ip','tags', '@timestamp' ]
       add_field => { "agent.name" => "%{[@metadata][name]}" }
       add_field => { "agent.name_key" => "%{[@metadata][fingerprints]}" }
       add_field => { "agent.user" => "%{[@metadata][user]}" }
       add_field => { "agent.user_key" => "%{[@metadata][fingerprintsus]}" }
       add_field => { "agent.ip" => "%{[@metadata][ip_orig]}" }
```

```
        add_field => { "agent.ip_key" => "%{[@metadata][fingerprintsip]}" }
        add_field => { "resumo" => "%{[@metadata][name]} * %{[@metadata][user]}
*%{[@metadata][ip_orig]}" }
}
```

Na configuração da pipeline, quando se tentou atribuir o valor de *hash* ao campo IP ocorreu de forma sistemática um erro, pois o tipo de dados não era compatível e a pipeline não funcionava. Como se tinha definido que se ia utilizar a função SHA256 e o HMAC, optou-se por alterar o campo IP para um valor fixo, neste caso, preencheu-se com o número "1" e adicionou-se ao *log* um novo campo com o valor *hash* do IP, sendo que ao novo campo foi dado o nome de *ip_orig_temp_key* (ver Figura 4-14). Através do campo *ip_orig_temp_key* é possível relacionar o IP que está no índice com o campo original e com o respetivo valor *hash*. De seguida é apresentado o excerto do código que efetua estas operações.

```
if [host][ip] {
        mutate {
                copy => { "[host][ip]" => "ip_orig_temp" }
        }

        mutate {
                split => [ "ip_orig_temp" , "," ]
                add_field => { "ip_testes" => "%{[ip_orig_temp][1]}" }
        }
        mutate {
                replace => { "[@metadata][ip_orig]" => "%{[ip_testes]}" }
        }

        fingerprint {
                source => "[host][ip]"
                target => "[@metadata][fingerprintsip]"
                method => "SHA256"
                key => "HMAC-SHA-256"
        }
        #vai-se substituir o valor IP pelo valor da hash caso não se esteja perante a cópia
        if "clone_name_equip" not in [tags] {
                mutate {
                        add_field  => { "new_host_ip" => [ "1:1:1:1:1:1:1:1" , "1.1.1.1" ] }
                }
                #criar o campo onde se vai guardar o valor de hash
                mutate {
                        add_field  => { "ip_orig_temp_key" =>  "%{[@metadata][fingerprintsip]}"}
                }

                mutate {
                        copy => {  "new_host_ip" =>   "[host][ip]"   }
                }
                mutate {
                        gsub => [ "%{"[@metadata][ip_orig]}" , " %{[@metadata][fingerprintsip]} " ]
                }
                mutate {
                        remove_field => [ "new_host_ip" , "ip_orig_temp" , "ip_testes" ]
                }
        }
}
```

No excerto de código anterior é possível observar que também é utilizado o filtro *mutate*, que realiza várias operações, entre as quais destacamos as seguintes: adicionar um campo

(*add_field*), copiar um valor (*copy*), remover um campo (*remove_field*) ou dividir (*split*) uma *string* em várias *substrings*.

Ao serem analisados os *logs*, verificou-se que o nome do computador e do utilizador também podem aparecer em maiúsculas/minúsculas. Para que fosse possível substituir o valor original pelo valor de *hash*, recorreu-se ao filtro *mutate* para colocar os dados do campo em maiúsculas (*uppercase/lowercase*), este valor vai ser utilizado para substituir o utilizador/nome do computador pelo seu valor de *hash* na mensagem, caso exista correspondência (*gsub*).

```
mutate {
       uppercase => [ "[@metadata][user_uppercase]" ]
       id => mutate_username_uppercase
}
mutate {
       uppercase => [ "[@metadata][user_lowercase]" ]
       id => mutate_username_uppercase
}
mutate {
       gsub    =>    [    "[message]",    "%{[@metadata][user_uppercase]}"    ,    "
%{[@metadata][fingerprintsus]} " ]
       id => mutate_subs_msguser2
}
```

Pode-se comprovar-se o resultado da pseudonimização na Figura 4-15, sendo que foi limitada a visualização ao comado *ping* efetuado pelo utilizador, porque existem situações em que o valor do campo *user.name* é igual ao nome da máquina e, logo o seu valor de *hash* é igual.

Figura 4-15 – Exemplo da pseudonimização de campos (nome da máquina, Utilizador, IP e mensagem)

Nos testes realizados, verificou-se que era criado no índice *data_key* um registo para cada *log* processado pela pipeline. Como resultado, o índice possuía milhares de registos iguais, tendo em conta este aspeto, optou-se por criar um *id* único através do *fingerprint* e do método

"MURMUR3" que garante que não existem valores duplicados no índice. O método "MURMUR3" é uma função de *hash* que recorre a operações de multiplicação e de rotação, para tal recorreu-se a um resumo dos campos que vão ser pseudonimizados.

De seguida apresenta-se o excerto do filtro *fingerprint*, no qual foi utilizado o método "MURMUR3" e também do plugin de saída, para facilitar a compreensão este foi colocado a negrito na criação do *id* e na sua aplicação.

```
      fingerprint {
            source => ["resumo"]
            target => "[@metadata][fingerprint_id]"
            method  => "MURMUR3"
      }
                  ***-----------------------------------------------***
      if "clone_name_equip" in [tags] {
            elasticsearch {
                  ssl => true
                  ssl_certificate_verification => false
                  hosts => ["https://IP_Segundo_Elasticserch:9201"]
                  index => "data_key"
                  document_id => "%{[@metadata][fingerprint_id]}"
                  user => "logstash"
                  password => "*******"
                  truststore => "/etc/logstash/keystore.jks"
                  truststore_password => "readonlyrest"
            }
      }
```

Foram detalhadas algumas das operações realizadas no ficheiro de configuração da pipeline do Logstash no Anexo K – Configuração da pipeline do Logstash.

Plugin Elasticsearch da ReadonyRest

Como já foi referido, optou-se por separar os dados que permitem a recuperação dos dados de um Titular, para tal, configurou-se um segundo servidor no qual foram instalados o Elasticsearch e o Kibana.

Como se pretendia auditar os acessos aos dados que estão guardados no Elasticsearch, instalou-se o plugin Elasticsearch, criado pela ReadonyRest. É de referir que no Anexo L – ReadonlyREST está descrito o processo de instalação do plugin.

Para além desta instalação, foi necessário criar utilizadores e definir quais as permissões dos mesmos. Assim, foram criados três utilizadores, um para que o Logstash pudesse comunicar com o Elasticsearch e dois no Kibana. O utilizador *elastic* tem permissões de administrador e pode aceder a todas as opções do Kibana, já o utilizador *userk* pode ler os índices do Elasticsearch e tem permissão de escrita no índice *audit**, pois pretendia-se, numa fase posterior, auditar as operações realizadas no índice *data_key*. É de referir ainda que o

utilizador *userk* não possui permissões para aceder ao menu de gestão do Kibana. De seguida, listamos a configuração dos utilizadores e as respetivas permissões de acesso.

```
    - name: "Logstash can write and create its own indices"
      auth_key: logstash:*****
      type: allow
      hosts: [127.0.0.1, IP_ELASTIC]
      actions: ["cluster:*",
"indices:data/read/*","indices:data/write/*","indices:admin/*"]
      indices: ["logstash*", "data*", "elas*","<no-index>"]

    - name: "Kibana Admin"
      auth_key: elastic:***
      kibana_access: admin
      indices: ["logstash*", "data*", "elas*", "audit*", ".kibana*", ".async*",
".apm*",".monitoring*"]

    - name: "Kibana User"
      auth_key: userk:***
      type: allow
      actions: ["cluster:*", "indices:data/read/*", "indices:data/write/audit*"]
```

Para que o Logstash possa comunicar com o Elasticsearch, foi necessário copiar o certificado que foi criado (a sua criação é explicada no Anexo L – ReadonlyREST), que se designa de *keystore.jks*, para a pasta na qual se encontra o Logstash. Posteriormente, foi necessário configurar na pipeline o plugin do Elasticsearch que vai comunicar com o Elasticsearch que possui o índice *data_key*. No ficheiro de código seguinte, a negrito, são destacadas as configurações necessárias para que a pipeline do Logstash possa comunicar com o Elasticsearch.

```
      if  "clone_name_equip" in [tags] {
            elasticsearch {
                  ssl => true
                  ssl_certificate_verification => false
                  hosts => ["https://IP_Segundo_Elasticserch:9201"]
                  index => "data_key"
                  document_id => "%{[@metadata][fingerprint_id]}"
                  user => "logstash"
                  password => "*******"
                  truststore => "/etc/logstash/keystore.jks"
                  truststore_password => "readonlyrest"
            }
      }
```

Para que possa ser possível auditar as operações realizadas pelos utilizadores ao índice *data_key* é necessário ativar a funcionalidade auditoria. Na configuração da auditoria, é possível definir quais os índices que se pretendem auditar, o nome do ficheiro da auditoria e se se pretende guardar a pesquisa. De seguida, são apresentadas as configurações efetuadas relativamente à auditoria ao índice *data_key*.

```
    enable: true
    audit_collector: true
    audit_include_query: ["data*"]
    audit_index_template: "'audit_logs'-yyyy-MM"
    audit_serializer: tech.beshu.ror.requestcontext.QueryAuditLogSerializer
```

Após terem sido efetuadas as configurações que permitem que sejam auditadas as operações realizadas no ficheiro de configurações *readonlyrest.yml*, o índice é automaticamente criado, como se pode constatar na imagem seguinte.

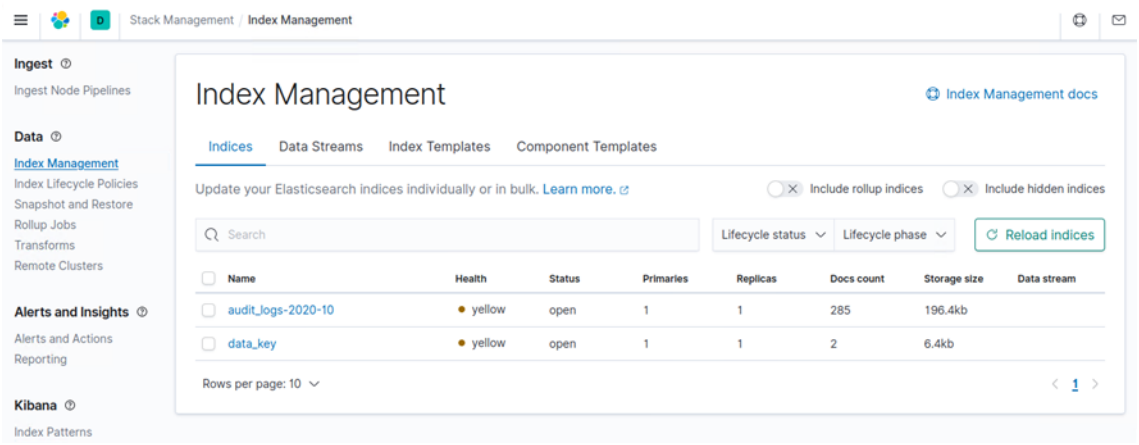


Figura 4-16 – Criação do índice que realiza a auditoria, gerido pelo plugin da ReadonlyRest

Caso o utilizador *userk* necessite de obter a identificação do nome do computador no qual ocorreu um incidente, acede ao índice *user_key* e pesquisa pelo valor de *hash*. Como se pode visualizar na Figura 4-17, a auditoria guarda a operação realizada, a identificação de quem efetuou a operação, o período temporal e o IP de origem.

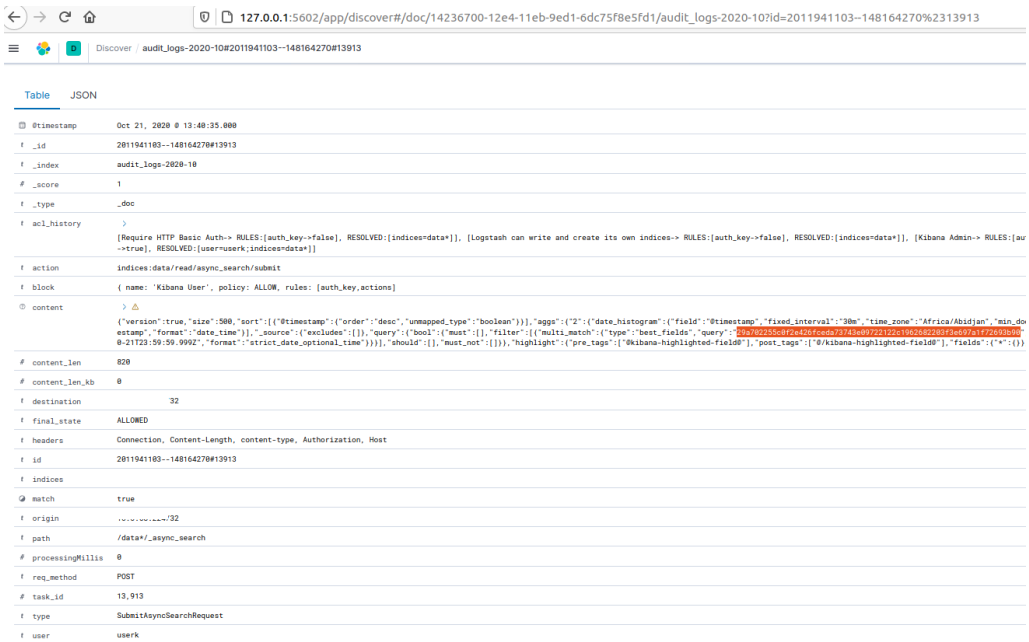


Figura 4-17 – Resultado da auditoria a uma operação realizada pelo utilizador *userk*

Tendo em consideração o cenário descrito, é possível afirmar que o plugin Elasticsearch da ReadonlyRest permite que seja efetuada uma auditoria às ações dos utilizadores de forma bastante pormenorizada.

4.4. Testes e resultados

No ponto 4.3 foi descrita a implementação do protótipo, justificou-se a escolha dos diversos componentes que integram a sua arquitetura e descreveram-se as configurações efetuadas.

O principal objetivo desta secção é o de apresentar os resultados dos testes de usabilidade do protótipo, tendo-se considerado importante demonstrar a aplicação das medidas técnicas para proteção e controlo dos dados pessoais, validar a capacidade de deteção de ameaças e de violações em tempo real e de demonstrar as restantes funcionalidades requeridas nos pré-requisitos. Também foram recolhidas as métricas do protótipo nas duas abordagens, para que se pudesse calcular o esforço da pseudonimização.

Em suma, o objetivo principal desta secção foi o de testar a usabilidade do protótipo, tendo em conta os pré-requisitos definidos. Para a avaliação dos resultados obtidos foi elaborada uma tabela resumo com recurso aos pré-requisitos definidos 4.2 para que depois se pudessem tirar conclusões.

4.4.1. Demonstração das medidas que garantem a conformidade com o RGPD

Na descrição da implementação do protótipo foram referidas muitas das funcionalidades que estão relacionadas com as medidas que garantem a conformidade com o RGPD, contudo, neste ponto, vamos enumerar várias das medidas implementadas.

Para validar a pseudonimização, foram listados os campos que se pretendem pseudonimizar (nome da máquina, IP origem e utilizador) na opção *Discovery* do Kibana.

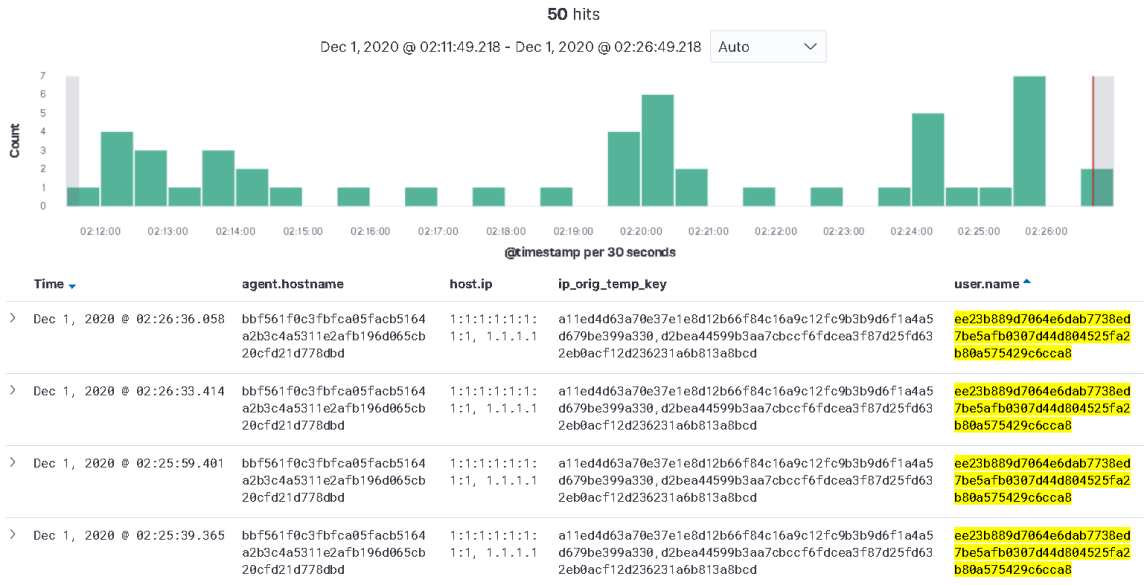


Figura 4-18 – Exemplo da pseudonimização dos campos: nome da máquina, endereço IP de origem e utilizadores

Para restringir os acessos ou atribuir vários níveis de permissões, recorreu-se ao plugin Elasticsearch da ReadonlyRest e ao Kibana. Na Figura 4-19 é possível visualizar o utilizador *anav*, que possui as permissões definidas pelo administrador que também possui o mesmo nome.

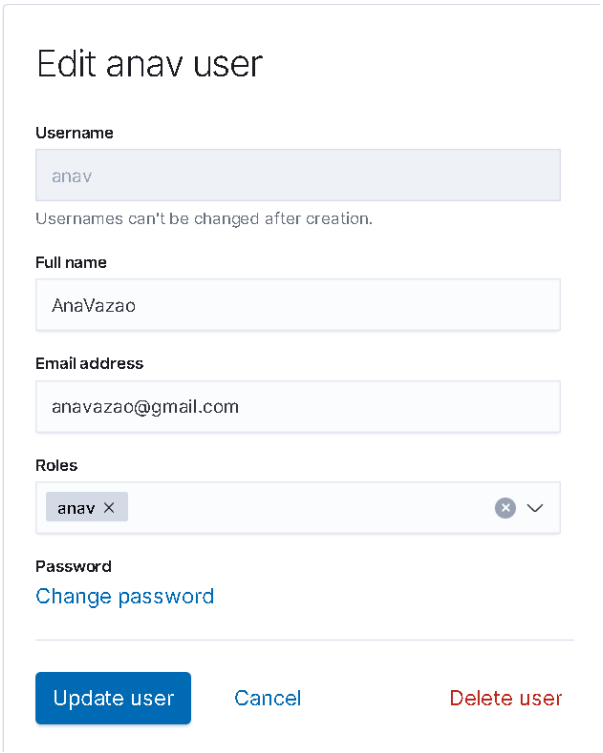


Figura 4-19 – Exemplo de um utilizador no Kibana ao qual se atribuíram permissões

No que diz respeito ao plugin Elasticsearch da ReadonlyRest, a configuração tem de ser feita na linha de comando, como está exemplificado no ponto 4.3.6. Quando o utilizador tenta aceder à gestão dos índices, tal operação não é permitida e a sessão é terminada.

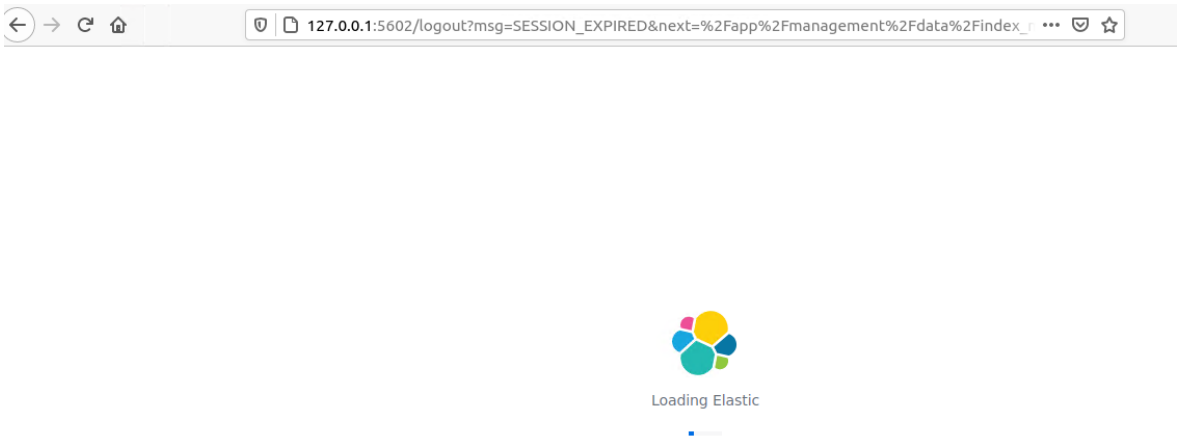


Figura 4-20 – Janela do Kibana com a tentativa de acesso pelo utilizador *userk* a uma opção para o qual não possui permissões

Os vários mecanismos de segurança foram descritos aquando da descrição da implementação do protótipo, sendo que, quando se acede à página web do Kibana, surge um aviso devido ao certificado que foi criado para este propósito, contudo, é possível visualizar a implementação do protocolo *http* com uma camada adicional de segurança, neste caso, o *https* que está implementado na solução.

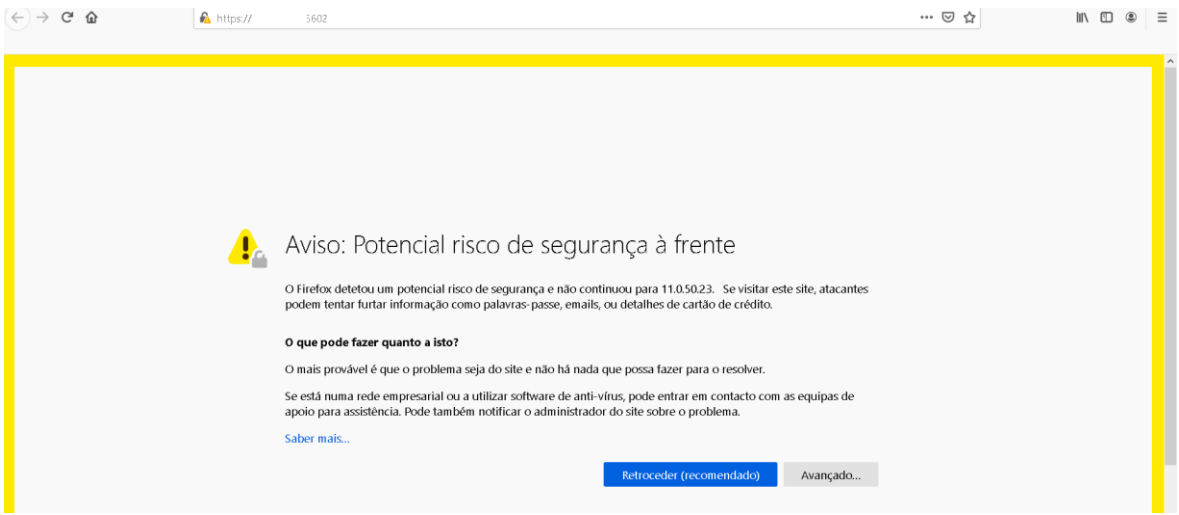


Figura 4-21 – Acesso à página do Kibana através de *https*

Outra funcionalidade importante que o Kibana disponibiliza no âmbito da segurança é a possibilidade de gerir os tempos de retenção para os dados que estão guardados nos índices. Os dados no Elasticsearch, como se observar na Figura 4-22, podem assumir quatro fases,

como de seguida se identificam: *Hot*, *Warm*, *Cold* e *Delete*; sendo que para cada uma das fases é possível especificar o tempo de retenção.

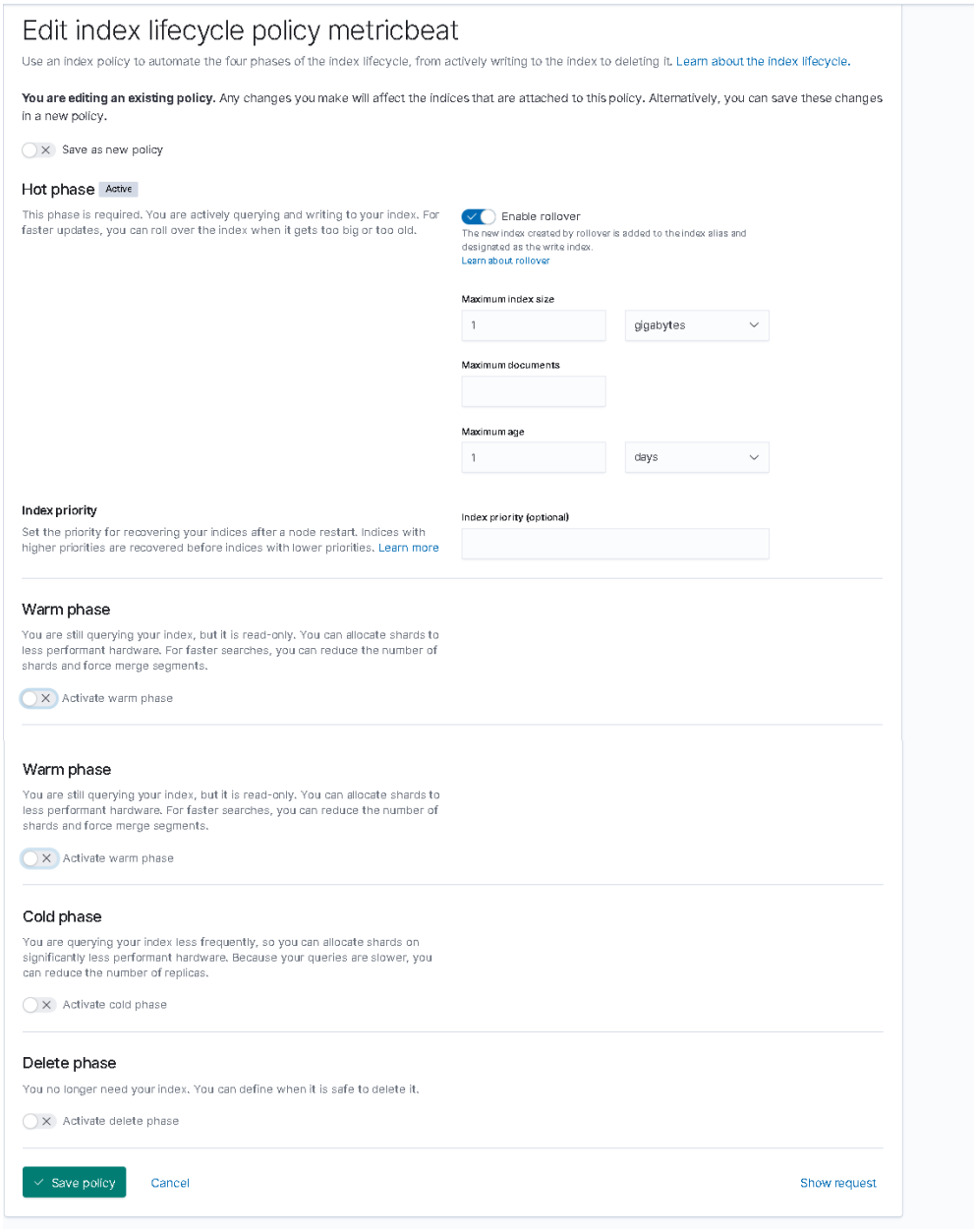


Figura 4-22 – Janela na qual é possível definir o período de retenção para cada índice

O plugin Elasticsearch, da ReadonlyRest, permite auditar as pesquisas realizadas pelos utilizadores. Na imagem seguinte são apresentados os detalhes de uma pesquisa que o utilizador *userk* realizou.

Figura 4-23 – Detalhe dos dados guardados no índice de auditoria quando se acede ao índice que contém dados pessoais

Quando são enviados pela primeira vez os *logs* para o Elasticsearch, este coloca o campo @version a “1”. Para qualquer operação de escrita realizada no *log*, o Elasticsearch incrementa numa unidade o número da versão. Esta funcionalidade é muito importante, porque garante a integridade do *log* no Elasticsearch. Na imagem seguinte apresenta-se um exemplo de um *log* com @version=”1”.

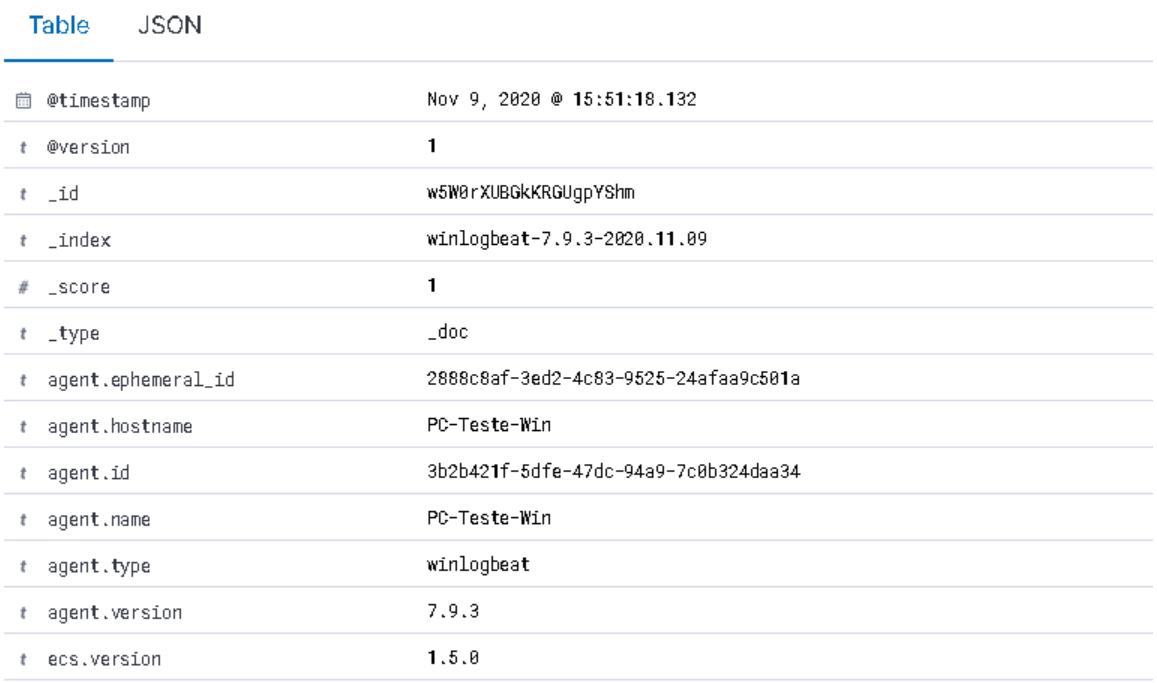


Figura 4-24 – Exemplo de um *log* com o número da versão igual a uma unidade

No ponto seguinte, criado especificamente, para esse fim, serão analisados a deteção de ameaças e de incidentes de segurança e a criação de alertas com o propósito de justificar a sua conformidade com o RGPD.

4.4.2. Simulação de ataques

Um dos pré-requisitos definidos para este trabalho foi a deteção de ameaças e de violações de segurança e também a criação de alertas, caso existam atividades suspeitas na rede. O Kibana disponibiliza uma interface Web, na qual os analistas informáticos podem identificar, em tempo real, possíveis violações ou indícios de que o sistema foi comprometido, com recurso a um conjunto de ações.

Como já foi referido anteriormente, os ataques à segurança podem ser internos ou externos, no caso de ataque interno as ações são aparentemente legítimas e autorizadas, uma vez que o atacante obtém de forma fraudulenta as credenciais, no caso de um ataque externo, os atacantes não possuem as credenciais e, por esta razão, normalmente as suas atividades são mais intrusivas.

Como prova de conceito, foi simulado um ataque interno e dois ataques externos, no ataque interno recorreu-se à ferramenta Pupy e nos ataques externos recorreu-se à ferramenta

Hydra e ao script *Simple-SYN-Flood*[58]. A escolha das ferramentas prende-se com o facto de as mesmas poderem realizar os ataques que foram identificados no ponto 2.4.2, também se tentou diversificar as ferramentas utlizadas, assim, foram selecionadas as seguintes: a ferramenta Pupy que está referenciada no Framework MITRE ATT&CK (MITRE ATT&CK, 2020), a ferramenta Hydra que é disponibilizada pela distribuição Kali (Kali, 2020) e o script *Simple-SYN-Flood* que é disponibilizado no GitHub.

Pupy

Na simulação do ataque informático interno, foi selecionada a ferramenta de acesso remoto Pupy. Entre 2016 e 2017, esta ferramenta foi utilizada para efetuar espionagem industrial, sobretudo na Arábia Saudita, o ataque foi identificado em 2017 pela Unit 42 da empresa Palo Alto Networks que se dedica a identificar ataques informáticos (Bryan & Falcone, 2017). A ferramenta Pupy[59] está identificada na Framework MITRE ATT&CK, na qual são enumeradas as múltiplas formas de ataque, como por exemplo a integração com o Mimikatz[60] ou a comunicação SSL (MITRE ATT&CK, 2020).

O Pupy é uma ferramenta multiplataforma (Microsoft Windows, Linux e Android) de acesso remoto, que permite a gestão de vários computadores em simultâneo ou que efetua ligações de SSH. É de referir que no Anexo M – Pupy é descrita a sua instalação, bem como algumas opções da ferramenta.

Para o cenário de ataque foi utilizado o Gmail e o Microsoft Windows 10, numa primeira fase, foi criado o *payload* cliente. Depois, este foi comprimido com uma password e, só depois, é que foi possível fazer o upload do ficheiro na Drive do Gmail (o ficheiro comprimido sem password é removido automaticamente pelo Gmail). Nesta fase é necessário convencer o utilizador que tem direitos de administrador que o ficheiro enviado contém algo que ele necessita e que é útil, para que execute o ficheiro e que ignore os alertas do antivírus Microsoft Defender. A partir do momento em que o ficheiro foi executado com sucesso, o "atacante" obteve acesso ao computador. Na Figura 4-25 está representado de forma gráfica o ataque efetuado à máquina Microsoft Windows.

[58] https://github.com/Leeon123/Simple-SYN-Flood
[59] https://github.com/n1nj4sec/pupy
[60] https://github.com/gentilkiwi/mimikatz

Figura 4-25 – Esquema do ataque efetuado recorrendo à ferramenta Pupy

Para ser criado o *payload* cliente para o sistema operativo Microsoft Windows, foi necessário escolher o sistema operativo, a arquitetura, o endereço IP da máquina na qual está o Pupy e o tipo de ligação. Na imagem seguinte, é possível visualizar a criação do *payload* cliente no sistema operativo Microsoft Windows com recurso à ferramenta Pupy (posteriormente o ficheiro criado foi renomeado para *cliente.exe).*

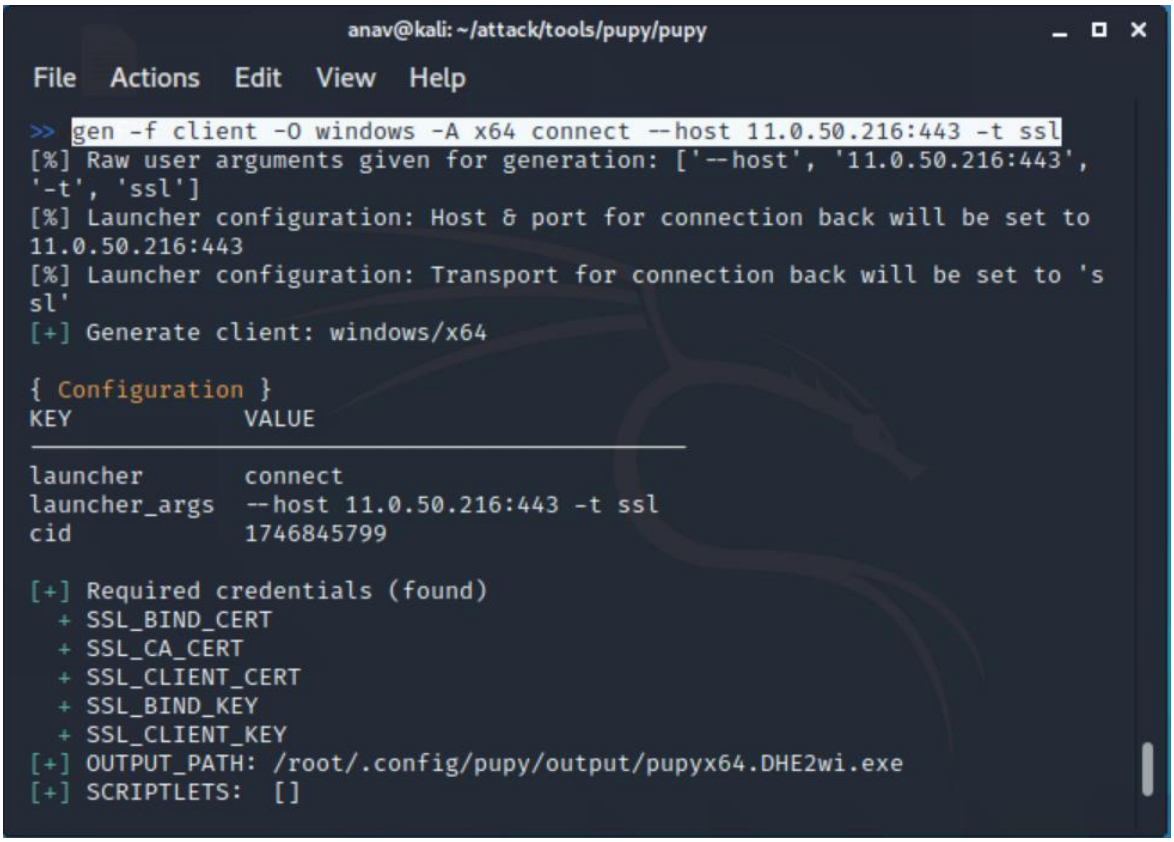


Figura 4-26 – Criação do *payload* cliente através do Pupy

Quando o utilizador tenta executar o ficheiro, a firewall do Microsoft Windows 10 emite um aviso e é necessário possuir direitos de administrador para conseguir desbloquear o ficheiro, como se pode visualizar na Figura 4-27.

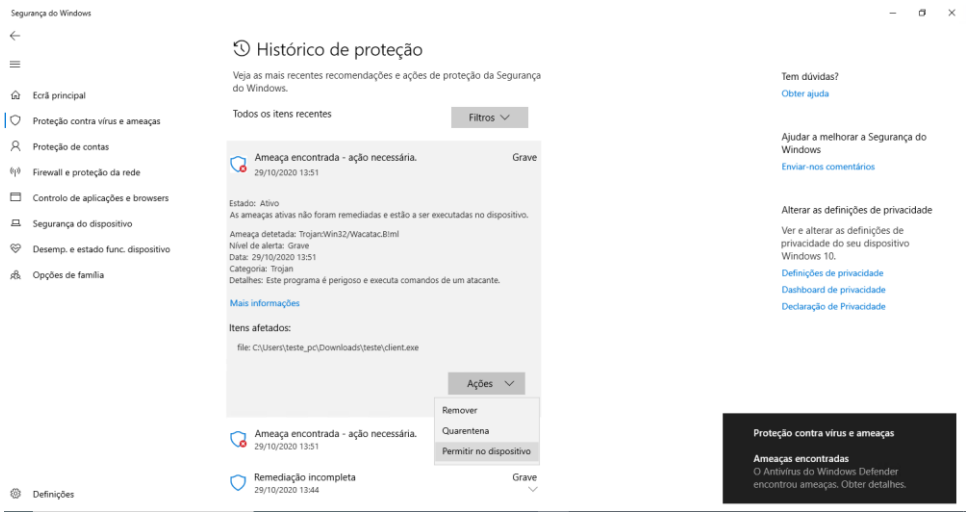


Figura 4-27 – Alerta emitido pelo antivírus Microsoft Defender quando se tenta executar o *payload* cliente.exe

A partir do momento que o utilizador executa o ficheiro *cliente.exe* com sucesso, o atacante consegue ter acesso à máquina Microsoft Windows, como está ilustrado na Figura 4-28.

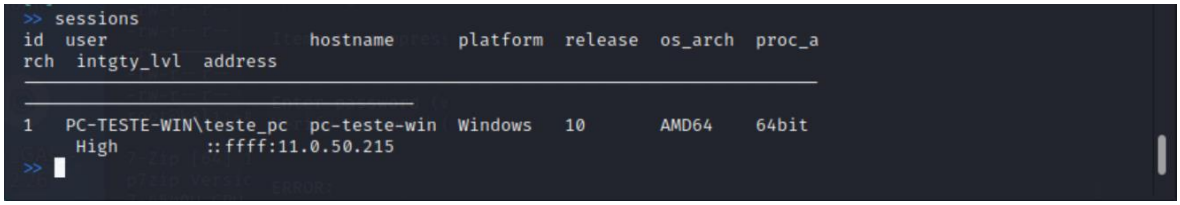


Figura 4-28 – Exemplo da ligação SSH ao computador da vítima

Recorrendo ao SIEM implementado no protótipo, é possível visualizar as comunicações entre as duas máquinas, para tal foi selecionado o processo *cliente.exe*. Na solução Elastic Stack, na opção *Security*, é disponibilizada a funcionalidade *timelines*, cujo sua principal função é a de analisar possíveis anomalias. Recorreu-se a esta funcionalidade para tentar rastrear os *logs* criados pelo ataque (ver imagem seguinte).

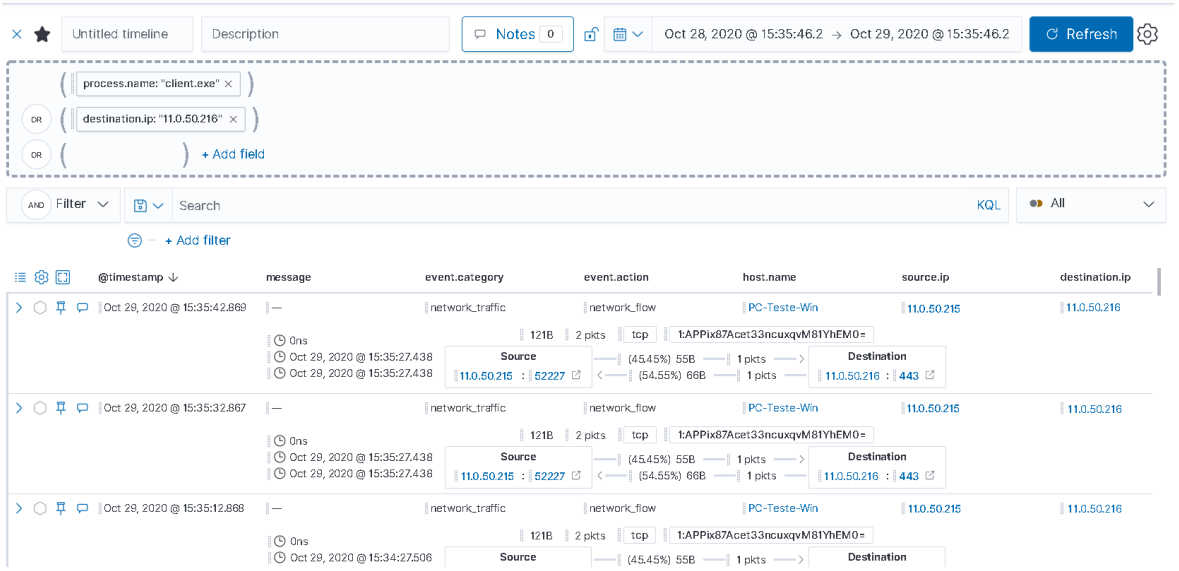


Figura 4-29 – Ligações efetuadas pelo processo cliente ao endereço IP de destinto 11.0.50.216

Em relação às passwords fracas é possível obter a password através das suas *hashes*, assim, como prova de conceito, foram descarregadas as *hashes* dos utilizadores, através do comando *creddump* (pode-se visualizar o resultado na figura seguinte).

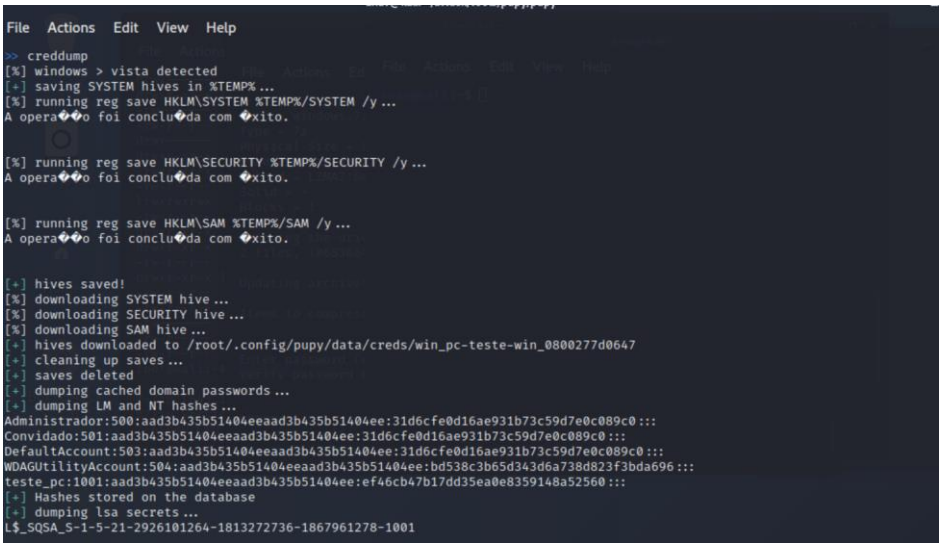


Figura 4-30 – Cópia das *hashes* dos utilizadores da máquina Microsoft Windows

É de referir que foi difícil encontrar indícios do comando *creddump*, tendo sido necessário analisar os *logs* na opção *Discover* do Kibana e, só depois, foi possível encontrar o rasto desta ação. Na Figura 4-31 podem ser observados os indícios desta ação. É de referir que o utilizador responsável por esta operação é o utilizador que executou o *payload cliente.exe*.

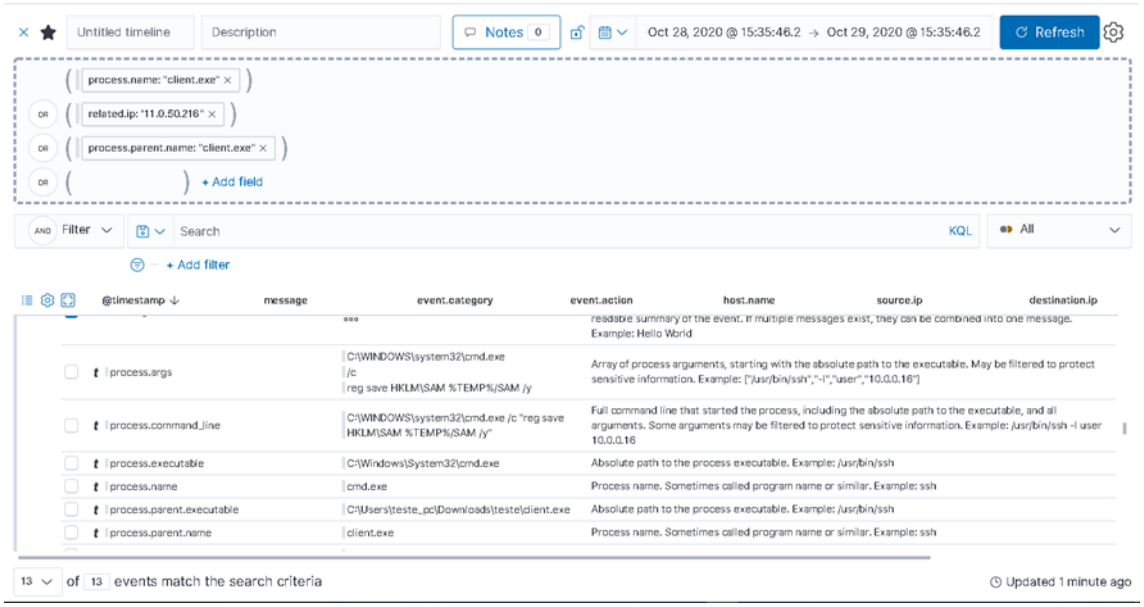


Figura 4-31 – Exemplo dos *logs* criados com o comando *creddump* do Pupy

O Pupy incorpora a ferramenta Mimikatz, que é bastante conhecida e utilizada para obter credenciais de acesso, por esta razão, também se executou a mesma. Na imagem abaixo pode ser visualizada a execução do comando.

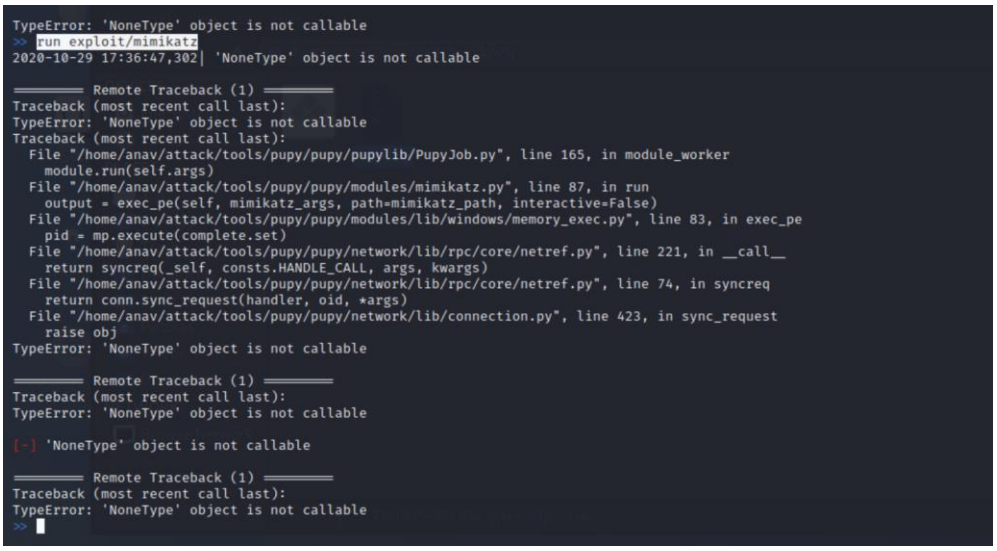


Figura 4-32 – Resultado da execução do Mimikatz através do Pupy no computador da vítima

No momento que se tenta executar a ferramenta a antivírus Microsoft Defender bloqueia a operação (ver Figura 4-33), sendo que, desta vez, não foi permitida a sua execução, e, por isso, o atacante não conseguiu obter as credenciais.

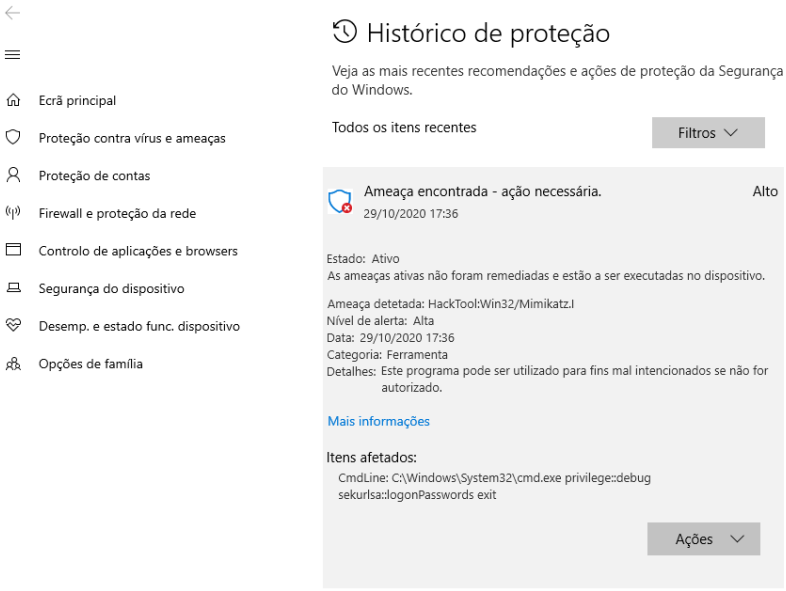


Figura 4-33 – Exemplo do Mimikatz bloqueado pelo antivírus Microsoft Defender

Contudo, foi possível obter indícios da tentativa de obtenção de credenciais. É de destacar facto de no *log* não constar o nome da ferramenta, mas de existirem indícios que ocorreu uma tentativa de escalar privilégios com o propósito de obter credenciais, como podemos observar na figura seguinte.

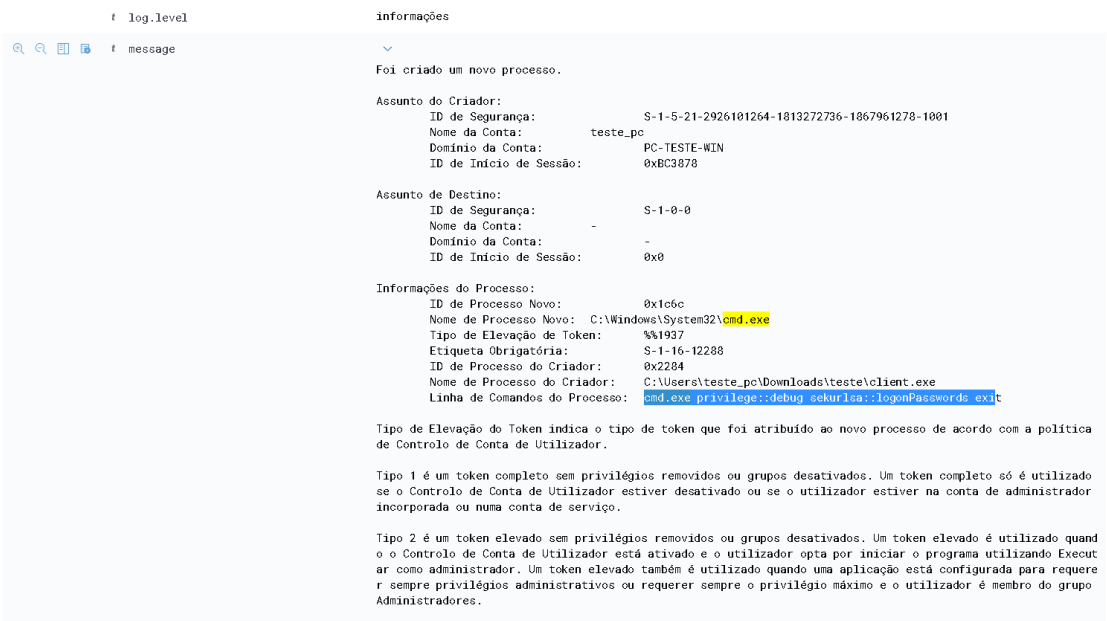


Figura 4-34 – *Log* que resultou da tentativa de executar o mimikatz no computador da vítima

Com o propósito de se ter um termo de comparação, foi executado o *payload cliente.exe* numa máquina com o sistema operativo Microsoft Windows Vista sem as atualizações. Verificou-se que não foi emitido nenhum alerta, assim sendo, através do Mimikatz o atacante

obteve a password do sistema em segundos (a password do sistema era uma password fraca), na imagem seguinte é apresentada a password do sistema operativo.

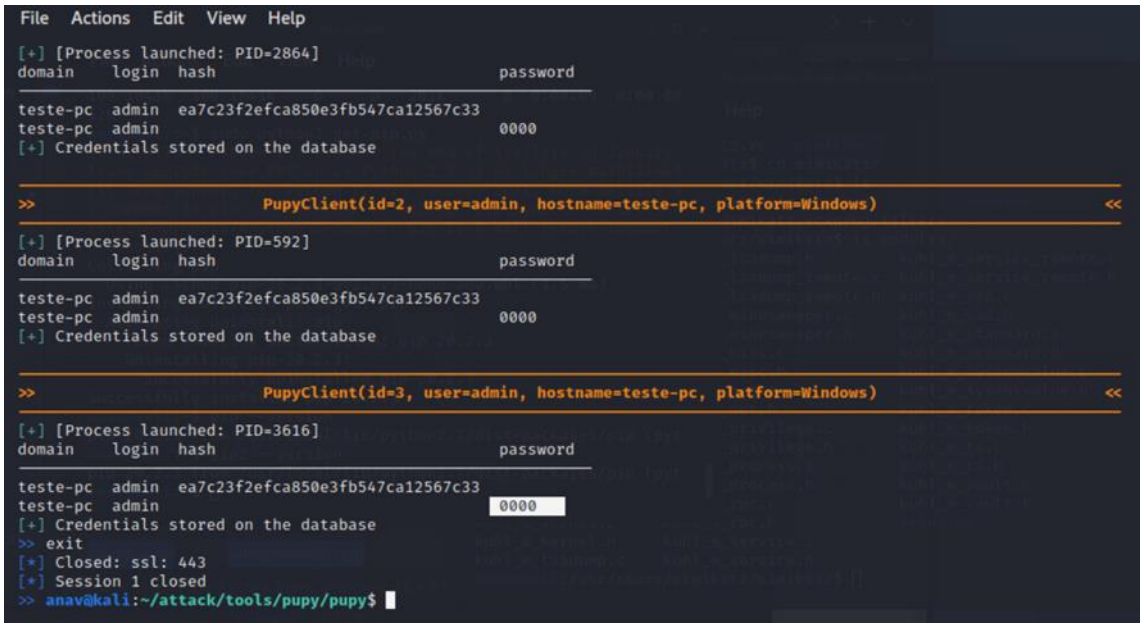


Figura 4-35 – Resultado do Mimikatz no sistema operativo Microsoft Windows Vista

O Pupy possui muitas funcionalidades e do leque de possibilidades oferecidas, foi selecionada a que permite aceder à linha de comando do computador da vítima. Na Figura 4-36 é ilustrada a execução do comando s*hell*.

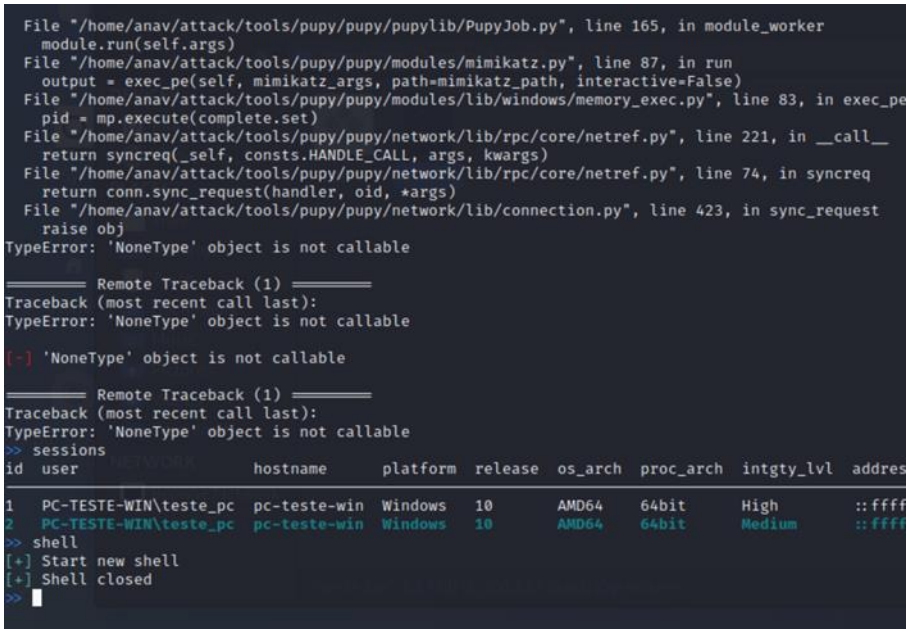


Figura 4-36 – Comando *Start new shell* do Pupy

Na próxima imagem é possível visualizar o resultado da ação do comando s*hell*, sendo que é através deste comando que se acede à linha de comandos do computador da vítima e que é possível visualizar todas as pastas e diretorias ou executar comandos.

Figura 4-37 – Linha de comandos na consola do Pupy

O rastreamento desta operação foi difícil de identificar, pois existiam muitos *logs* gerados pelo *cliente.exe*, todavia, com recurso aos *logs* criados pelo Sysmon, foi possível encontrar o comando ***cmd.exe*** utilizado pelo Pupy para criar uma nova linha de comando. É possível observar os *logs* criados pelos Sysmon na Figura 4-38.

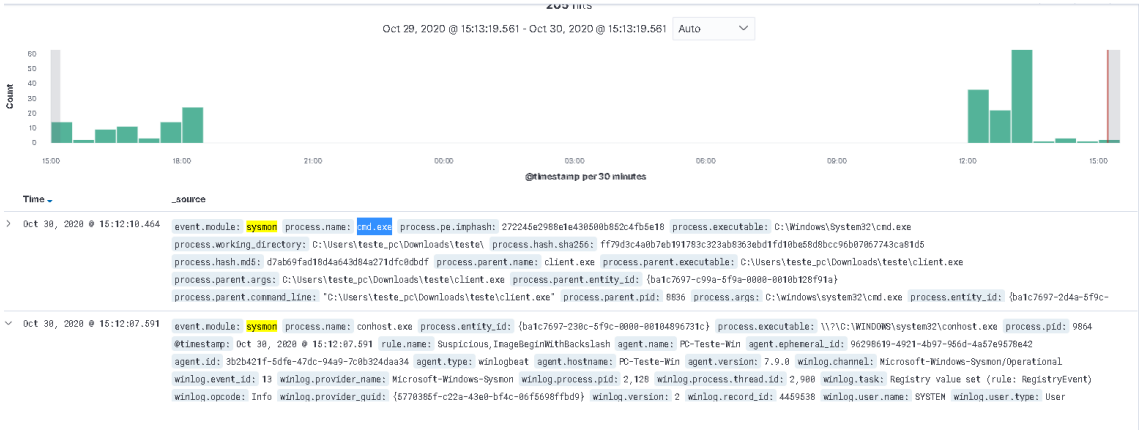


Figura 4-38 – Execução do cmd.exe pelo Pupy que criou uma nova linha de comandos

Na imagem seguinte são apresentados os detalhes do *log* no qual se pode visualizar a execução do comando *cmd.exe*, é também possível verificar que o utilizador que executou o *payload* cliente surge como responsável por estas ações, neste caso específico, o utilizador *teste_pc* na máquina *PC-TESTE-WIN*.

t	host.os.name	Windows 10 Pro
t	host.os.platform	windows
t	host.os.version	10.0
t	log.level	informações
t	message	Process Create: RuleName: UtcTime: 2020-10-30 12:50:14.364 ProcessGuid: {ba1c7697-0c06-5f9c-0000-0010b8240f1c} ProcessId: 2392 Image: C:\Windows\System32\cmd.exe FileVersion: 10.0.18362.449 (WinBuild.160101.0800) Description: Windows Command Processor Product: Microsoft® Windows® Operating System Company: Microsoft Corporation OriginalFileName: Cmd.Exe CommandLine: "C:\windows\system32\cmd.exe" CurrentDirectory: C:\Users\teste_pc\Downloads\teste\ User: PC-TESTE-WIN\teste_pc LogonGuid: {ba1c7697-ae83-5f91-0000-00207838bc00} LogonId: 0xBC3878 TerminalSessionId: 1 IntegrityLevel: High Hashes: MD5=D7AB69FAD18D4A643D84A271DFC0DBDF,SHA256=FF79D3C4A0B7EB1 ParentProcessGuid: {ba1c7697-c99a-5f9a-0000-0010b128f91a} ParentProcessId: 8836 ParentImage: C:\Users\teste_pc\Downloads\teste\client.exe ParentCommandLine: "C:\Users\teste_pc\Downloads\teste\client.exe"
t	process.args	C:\windows\system32\cmd.exe
t	process.command_line	"C:\windows\system32\cmd.exe"
t	process.entity_id	{ba1c7697-0c06-5f9c-0000-0010b8240f1c}
t	process.executable	C:\Windows\System32\cmd.exe
t	process.hash.md5	d7ab69fad18d4a643d84a271dfc0dbdf
t	process.hash.sha256	ff79d3c4a0b7eb191783c323ab8363ebd1fd10be58d8bcc96b07067743ca81d5
t	process.name	cmd.exe
t	process.parent.args	C:\Users\teste_pc\Downloads\teste\client.exe
t	process.parent.command_line	"C:\Users\teste_pc\Downloads\teste\client.exe"
t	process.parent.entity_id	{ba1c7697-c99a-5f9a-0000-0010b128f91a}

Figura 4-39 – Mensagem do *log* no qual se encontra o *cmd.exe* executado pelo Pupy

Hydra

Como prova de conceito, optou-se por repetir o teste realizado no ponto 0 do Capítulo 3, sendo que foi realizado um ataque de força bruta ao servidor Web através do serviço SSH e recorrendo à ferramenta Hydra (Figura 4-40).

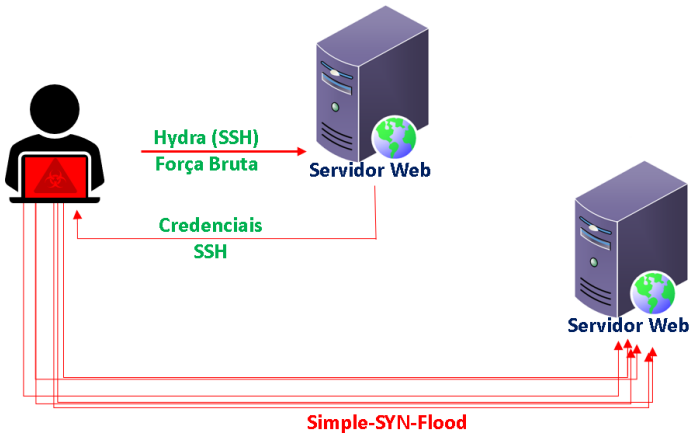


Figura 4-40 – Esquema dos ataques força bruta/negação de serviço

O servidor Web possuía uma password fraca e, por essa razão, foi possível obter as credenciais de acesso à máquina vítima, como se pode observar pelos dados obtidos e que estão disponíveis na Figura 4-41.

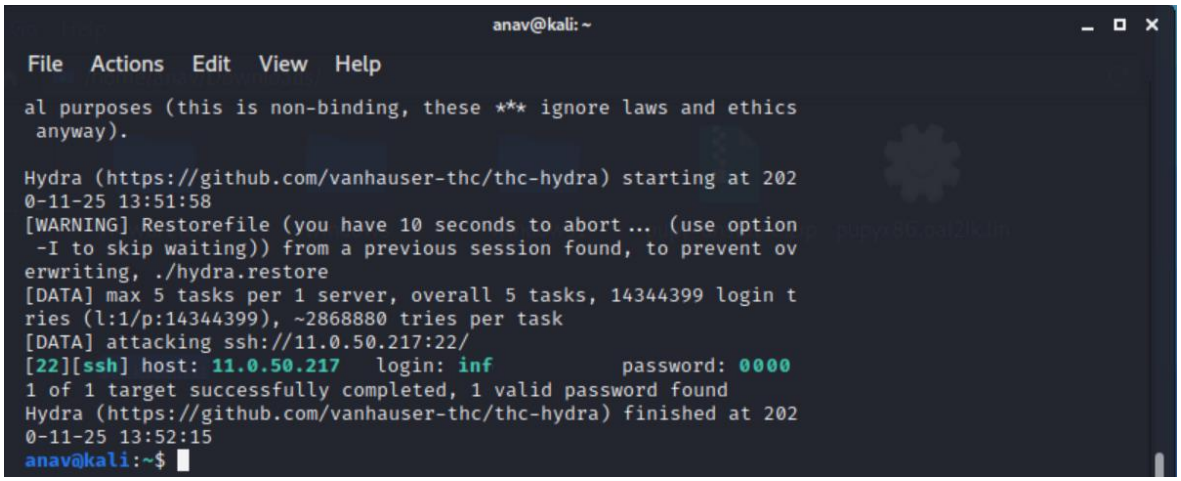


Figura 4-41 – Credenciais obtidas através da ferramenta Hydra

Assim, foi possível comprovar que um ataque de força bruta pode provocar muito "ruído", por exemplo, na Figura 4-42 é possível verificar a ocorrência de 2466 tentativas de autenticação ao serviço SSH.

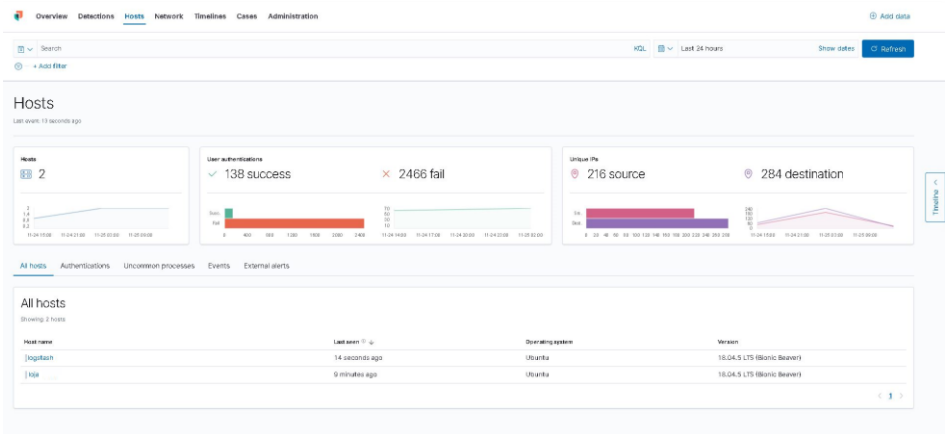


Figura 4-42 – *Dashboard* do Kibana que apresenta as 2466 tentativas de autenticação através do serviço SSH

Foi também possível validar a criação de alertas pela ferramenta Elastalert, assim, com esse propósito, foi criada uma regra para os Erros de login SSH, na qual, na eventualidade de serem encontradas correspondências com a regra criada, a ferramenta deveria criar o alerta e enviá-lo para o Slack. Na Figura 4-43 é possível visualizar as 141 novas mensagens (será de referir que o Elastalert foi desligado para que não fossem criadas mais mensagens).

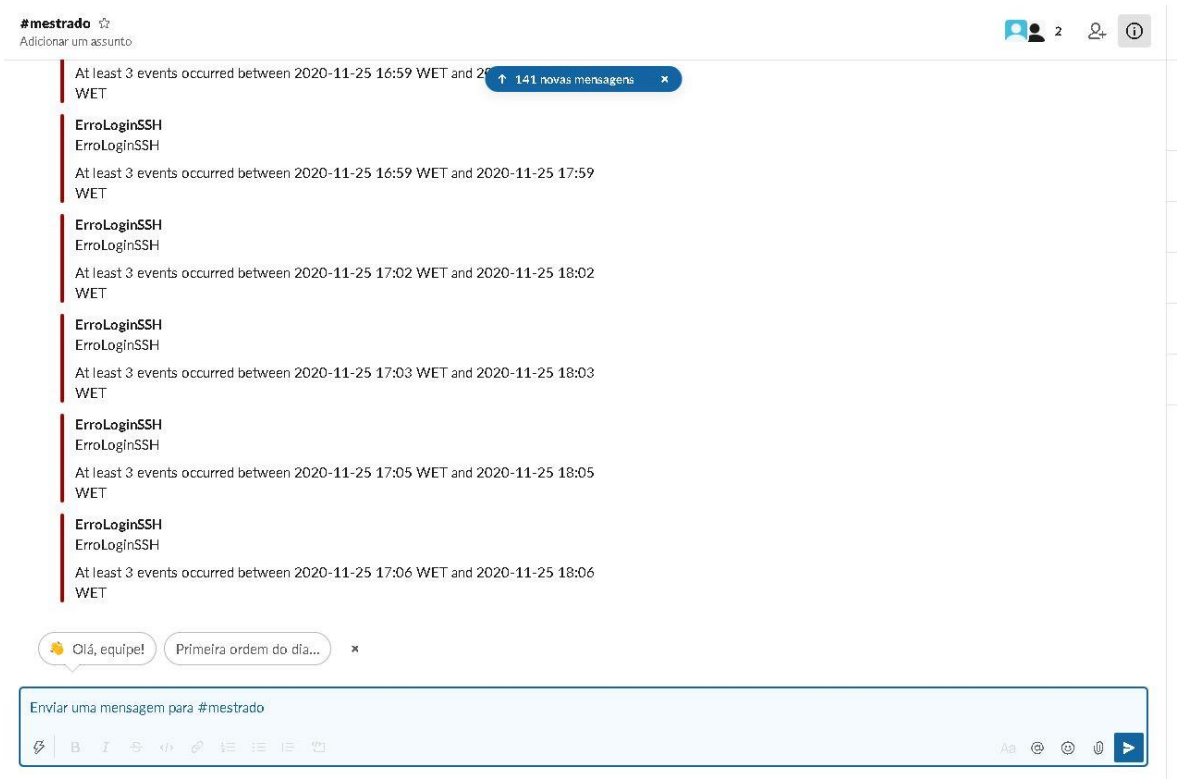


Figura 4-43 – Alertas criados no Slack, apresentando as múltiplas tentativas de erro de autenticação SSH

Simple-SYN-Flood

Como é possível visualizar no esquema da Figura 4-40, também foi realizado um ataque externo de negação de serviço com recurso ao script *Simple-SYN-Flood*. Através deste script pretendeu-se simular um ataque de negação de serviço SYN Flood, à semelhança do que foi descrito no ponto 2.4.2, com o propósito de esgotar ou de bloquear os serviços e de impedir o acesso de outros utilizadores.

Para simular o ataque externo de negação de serviço, acedeu-se remotamente ao servidor Web, com recurso ao SSH que estava a ser executado localmente (127.0.0.1:8000). Assim, foi descarregado e executado o ficheiro, não tendo sido necessário efetuar mais nenhuma operação para conseguir o que se pretendia. Na imagem seguinte é possível visualizar o comando utilizado.

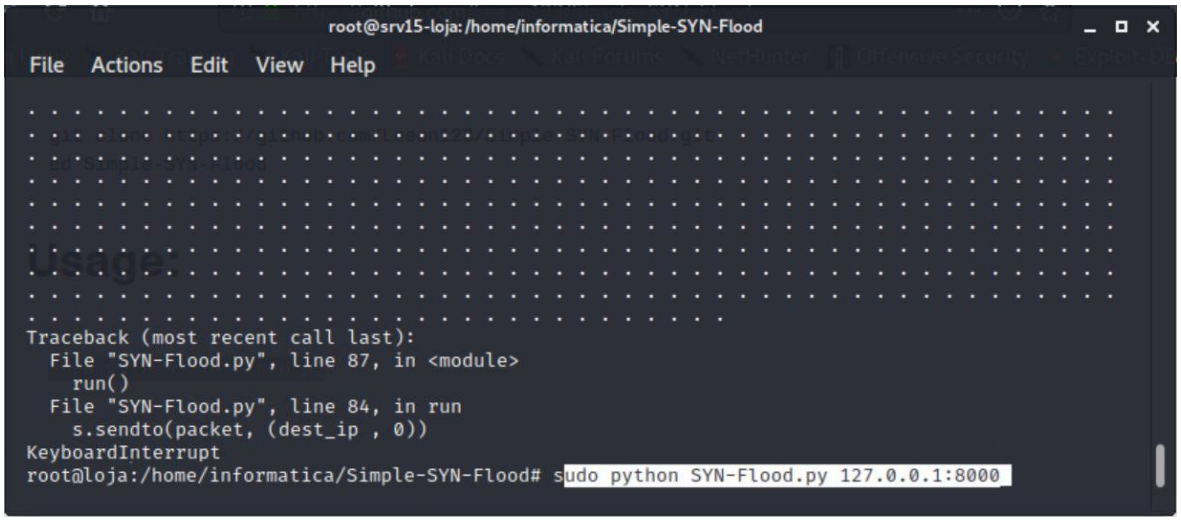


Figura 4-44 – Ataque de negação de serviço recorrendo ao script Simple-SYN-Flood

No decorrer do ataque o servidor não bloqueou, contudo, pode-se constatar pela visualização da Figura 4-45, a pressão a que foi submetido, pois na imagem é apresentado o *dashboard* do Kibana no qual é retratado o fluxo de dados do servidor Web.

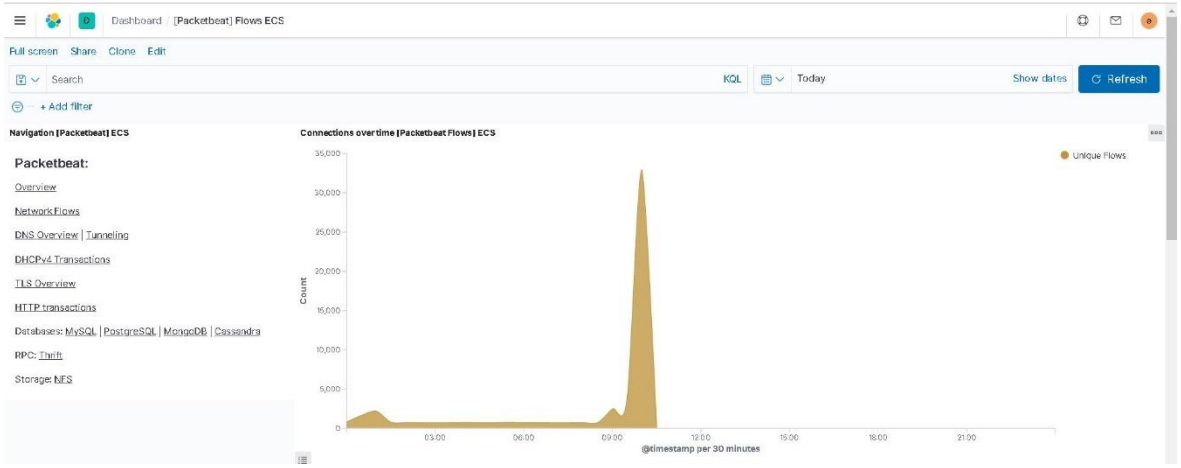


Figura 4-45 – *Dashboard* do Kibana no qual pode visualizar a evolução do ataque

Através dos cenários de ataque criados e dos resultados obtidos, foi possível identificar "as pegadas" dos ataques no cenário de testes do protótipo, para tal, recorreu-se às funcionalidades disponibilizadas pelo protótipo. Em algumas situações, a funcionalidade de recolha dos *logs* do Sysmon é uma mais-valia, pois identifica operações realizadas no computador que foram identificadas como tendo sido levadas a cabo no âmbito de ataques.

4.4.3. Métricas da pipeline do Logstash com e sem pseudonimização

Como já foi referido, optou-se por pseudonimizar e não anonimizar os dados pessoais dos *logs* de segurança dos sistemas operativos Microsoft Windows e Linux, pois isso iria permitir que se pudesse identificar a máquina e o utilizador em caso da ocorrência de um

incidente ou de uma anomalia. A pseudonimização vai ser efetuada através de uma pipeline do Logstash, que recorre à utilização de vários plugins. Os plugins vão aumentar a utilização do CPU no decorrer da realização das operações solicitadas, sendo que, devido a este facto foi considerado importante mensurar o esforço para realizar esta operação.

Com o objetivo de obter métricas mais rigorosas, optou-se por isolar a solução Elastic Stack, para tal desligaram-se os beats que comunicavam com o Logstash em todas as máquinas, com a exceção do Metricbeat que está instalado no Logstash. O beat Metricbeat é aquele que iria enviar os dados para o Elasticsearch, para que depois o Kibana pudesse disponibilizar as métricas. Além disso, também foi ativado o Netdata que iria recolher as métricas do servidor no qual estava instalado o Logstash, que posteriormente seriam tratadas pelo Prometheus e pelo Grafana.

Tendo como base a Tabela 4.2, que lista os campos a pseudonimizar para a máquina, utilizador e endereço IP, optou-se por selecionar um *log* continham o nome da máquina e do endereço IP. A esse *log*, na mensagem do mesmo, foi acrescentada informação. Na imagem seguinte é possível visualizar a mensagem do *log* e a informação que foi acrescentada.

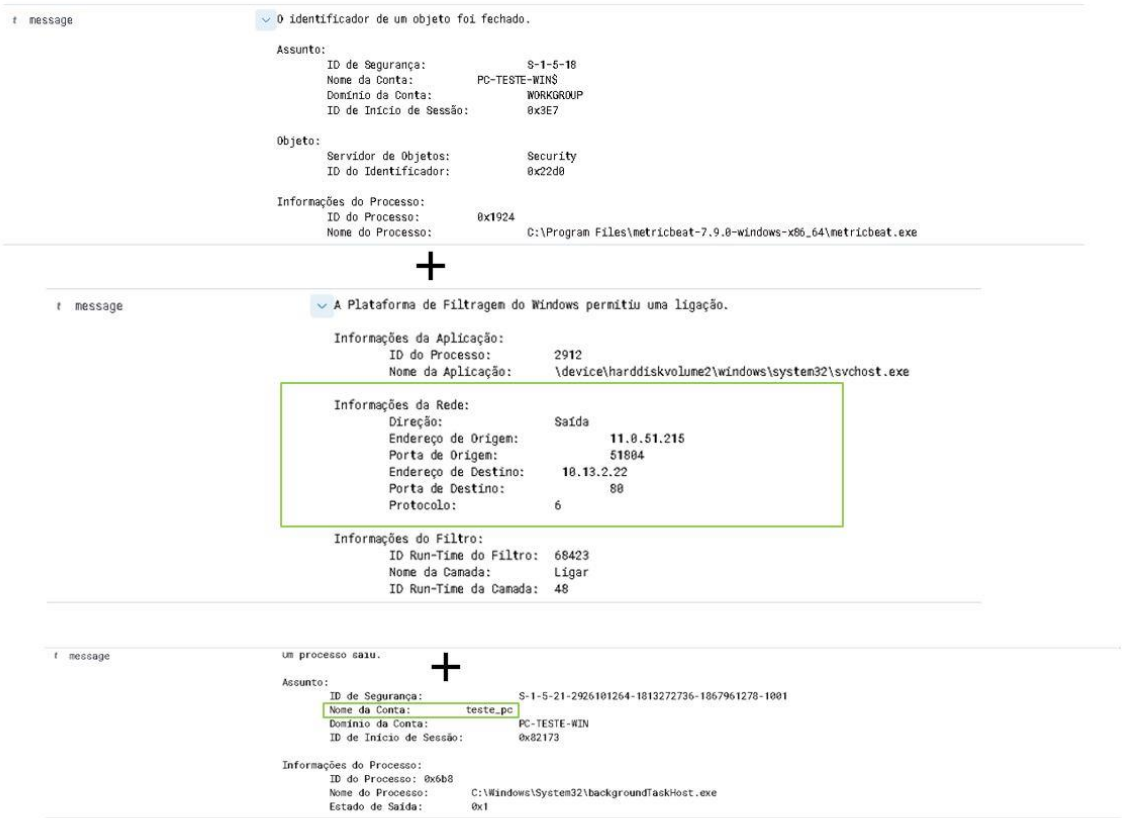


Figura 4-46 – Processo de criação do campo mensagem do *log*

Ao criar a mensagem de *log*, procurou-se que a mesma fosse exemplificativa da pipeline do protótipo, e, devido a esse facto, estão presentes no *logs* os três campos a pseudonimizar: nome da máquina, utilizador e o IP origem. Depois de ter sido selecionada a informação a colocar no ficheiro CSV, foi criado o *log* que seria utilizado nos testes. Para além da alteração da mensagem, foram adicionados dois campos no final da mesma, o *user_name* e o *winlog_event_data_SubjectUserName* com o valor *teste_pc*, como se pode observar na Figura 4-47.

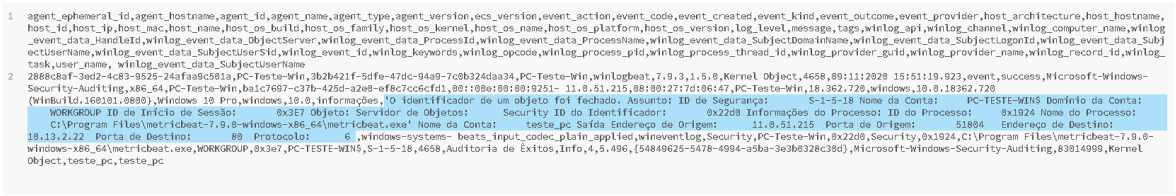


Figura 4-47 – Apresentação dos campos definidos para o *log* e respetivo conteúdo

Para que fosse possível mesurar o custo da pseudonimização, foram criados dois cenários, um no qual os *logs* não iriam ser pseudonimizados e outro em que se iria realizar a pseudonimização dos *logs*. Para que isso se concretizasse, foram recolhidas métricas nos dois cenários e utilizou-se o mesmo ponto de partida, ou seja, para cada um dos cenários foi utilizado o mesmo *Snapshot* em todas as máquinas virtuais.

Em cada cenário, foram utilizados quatro ficheiros CSV, sendo que o *log* selecionado foi copiado 100, 1000 e 10000 vezes (o primeiro ficheiro contém a linha criada manualmente). Os ficheiros foram enviados em quatro momentos temporais, tendo existindo o cuidado de ter sido escolhido um período temporal em que a pipeline não apresentasse outros dados, como se pode visualizar na imagem seguinte (existiu pelo menos 40 minutos de intervalo entre cada teste).

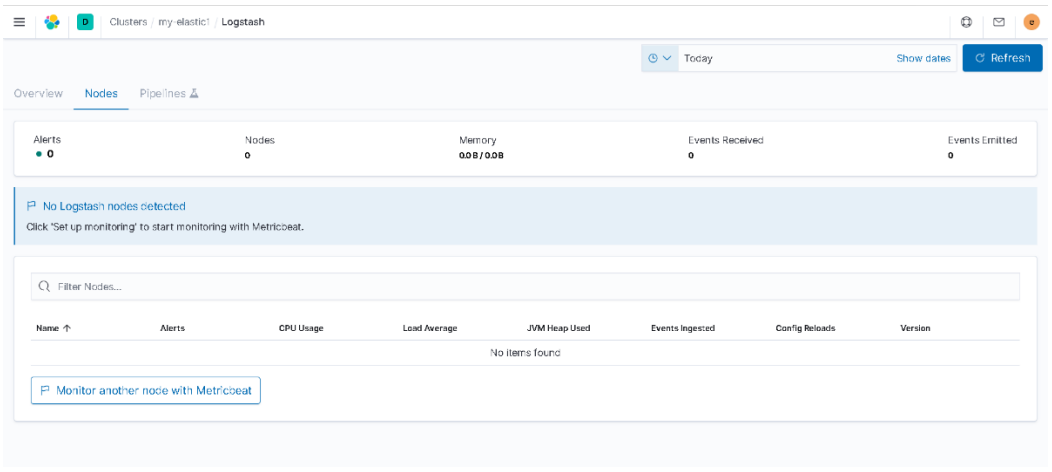


Figura 4-48 – Pipeline sem filtros para pseudonimizar os dados

Como já foi referido anteriormente, os dados do ficheiro foram enviados para a pipeline do Logstash, tendo de seguida sido efetuado um mapeamento dos campos, após o término do qual estes foram enviados para os dois Elasticsearch. Para a pseudonimização dos dados, foi utilizado o ficheiro de configuração que está descrito no Anexo K – Configuração da pipeline do Logstash, no entanto, foi necessário efetuar alguns ajustes, como foi o caso do plugin filtro CSV e do plugin *file*. Para o filtro CSV, como se pode visualizar na Figura 4-49, numa primeira fase foi enviado o ficheiro CSV para o Logstash, para ser lido pelo plugin *file*, e, depois, através dos filtros CSV foi realizado o mapeamento dos campos; após o término desta tarefa foi realizado o mapeamento dos *logs*, que posteriormente foram enviados para o Elasticsearch "Principal".

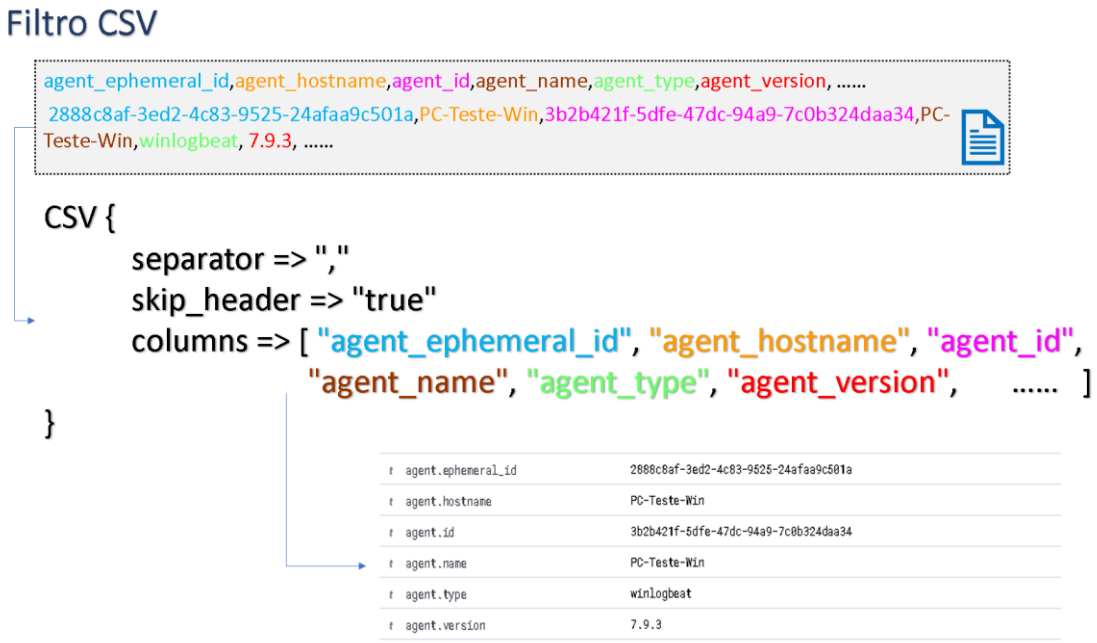


Figura 4-49 – Filtro CSV

No âmbito deste trabalho, optou-se por recolher dois tipos de métricas: as referentes ao Logstash, que são disponibilizadas no Kibana e recolhidas pelos Beats, e as métricas do servidor no qual está instalado o Logstash, que são recolhidas pelo Netdata e enviadas para o Prometheus e para o Grafana. Como se pode visualizar na Figura 4-50, as métricas são recolhidas pelos Beats e pelo Netdata.

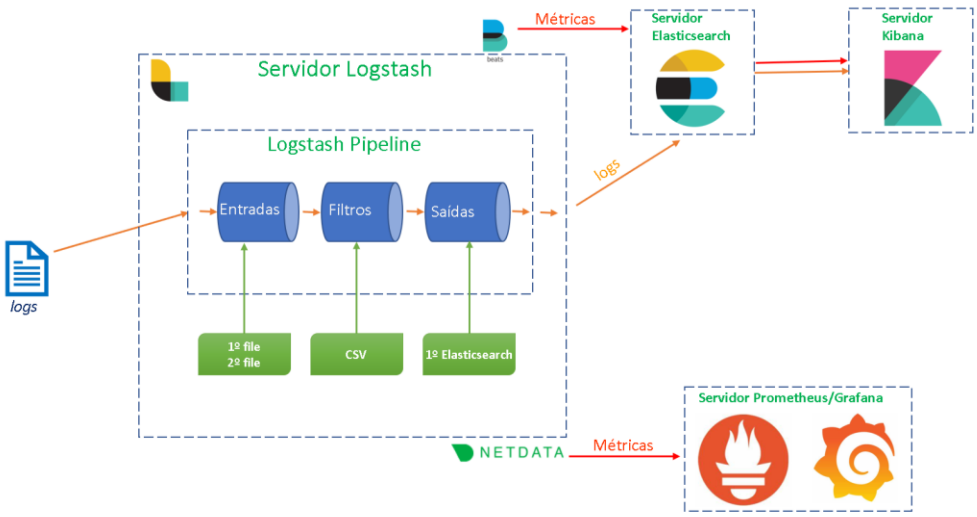


Figura 4-50 – Pipeline sem a pseudonimização

Os dois cenários possuem várias semelhanças entre si, como se pode constatar na Figura 4-50 e na Figura 4-51, contudo, no cenário no qual se procedeu à pseudonimização dos dados, foi necessário recorrer a um segundo servidor no qual estavam instalados o Elastic Stack e o Kibana.

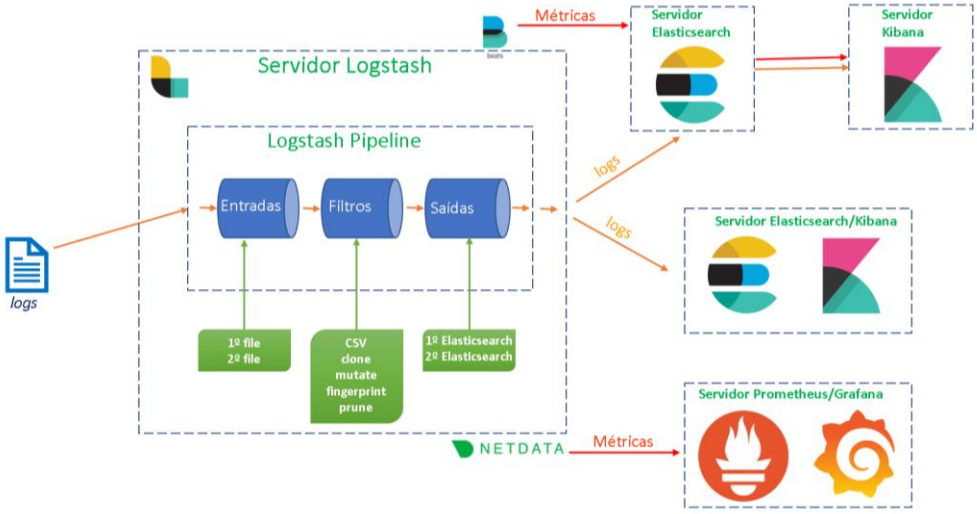


Figura 4-51 – Pipeline com os filtros para a pseudonimização dos dados

Além disso, foram selecionadas as métricas que estão elencadas na Tabela 4.3, a saber: número de eventos recebidos por segundo, JVM Heap (máquina virtual Java), número de eventos emitido por segundo pelo Logstash e a utilização do CPU.

Tabela 4.3 – Métricas da pipeline do Logstash com e sem pseudonimização

Métricas	**Hora do início do teste**	**Métricas da pipeline do Logstash disponibilizadas no Kibana**			
		Nº de Eventos recebidos (/s)	***JVM Heap (MB)***	***Nº de Eventos emitidos (/s)***	***Utilização do CPU (%)***
1 *log*	9:50	-	199.9/MB	-	2%
100 *logs*	10:50	-	184,6/MB	-	2%
1000 *logs*	11:30	75/s	203,7/Mb	100,1/s	91%
10000 *logs*	12:10	387,5/s	201,0/MB	387,5/s	94%
1 *log* pseudonimizado	14:50	-	201,5/MB	-	60%
100 *logs* pseudonimizados	15:40	-	223,9/MB	-	95%
1000 *logs* pseudonimizados	17:00	25/s	208,0/MB	33,37/s	93%
10000 *logs* pseudonimizados	18:00	220,83/s	286,4/MB	220,83/s	91%

Tendo em conta as informações constantes na Tabela 4.3, foi possível verificar que o aumento não é exponencial, ou seja, seria expetável que existisse um aumento considerável de utilização do CPU entre o envio de 100 *logs* e o envio de 10000 *logs*, mas no caso dos *logs* pseudonimizados acontece precisamente o inverso. Esta situação acontece porque o JVM Heap distribui os *logs* por um período temporal superior, como se pode observar na imagem seguinte. Ainda nessa imagem, é possível verificar que o período temporal necessário para processar os *logs* que foram pseudonimizados foi superior aos que não foram.

Figura 4-52 – Gráfico que apresenta a distribuição temporal da ingestão dos 10000 *logs* com e sem a pseudonimização

Relativamente às métricas do número de eventos recebidos/emitidos por segundo, estas apresentam um comportamento que pode ser considerado espectável, ou seja, para a ingestão dos 1000/1000 os valores são superiores no cenário onde é realizada a pseudonimização, apesar de a diferença não ser muito significativa.

Através das métricas recolhidas pelo Netdata e visualizadas no Grafana, é possível constatar que a memória RAM não teve um aumento muito significativo na sua utilização, contudo, é possível verificar que o processamento de 10000 *logs* teve um período temporal superior ao processamento ao de 1 *log*, como se pode observar na Figura 4-53 e na Figura 4-54.

Figura 4-53 – Gráfico que apresenta a distribuição temporal da memória quando se processa 1 e 10000 *logs* sem a pseudonimização dos dados

Figura 4-54 – Gráfico que apresenta a distribuição temporal da memória quando se processa 1 e 10000 *logs* com a pseudonimização dos dados

Comparando o gráfico do teste relativo ao cenário no qual foi não efetuada pseudonimização com o gráfico do teste relativo ao cenário no qual se realizou a

pseudonimização, é possível constatar que os resultados não apresentam diferenças muito significativas.

Assim, ao analisar o processamento do primeiro *log* com e sem a pseudonimização, verifica-se que, no que respeita às métricas do CPU, existiu em ambos os casos um aumento muito significativo, pois a diferença é superior a 50%, sendo um indício que a pseudonimização como era espectável, vai sobrecarregar o CPU.

Foram utilizadas duas pipelines para recolher a métricas, uma na qual foi efetuada a pseudonimização e outra na qual apenas foi efetuado o mapeamento dos campos. Tendo em conta o tipo de testes realizados, é de salientar, como já foi referido anteriormente, que as conclusões não podem ser extrapoladas, apesar disso, consideramos que é um contributo para esta temática.

Nesta secção descreveu-se o processo de recolha das métricas e dos resultados obtidos, contudo entendeu-se que era pertinente explicar de forma mais detalhadamente todo o processo. Assim, no Anexo O – Métricas da pipeline com e sem pseudonimização são documentadas as opções tomadas, os problemas encontrados e exibidos os gráficos com as métricas recolhidas.

4.4.4. Demonstração das funcionalidades requeridas nos pré-requisitos

Para demonstrar que os Beats podem recolher os dados de várias origens através do Beat Metricbeat que está instalado no servidor Web, foi ativado o módulo MySQL para que fosse possível recolher as métricas. Na Figura 4-55 é possível visualizar o *dashboard* do MySQL que é fornecido pelo Kibana. Com recurso ao protótipo SIEM, foi possível monitorizar os *logs* dos sistemas operativos Microsoft Windows e Linux, assim como o número de operações realizadas na base de dados MySQL.

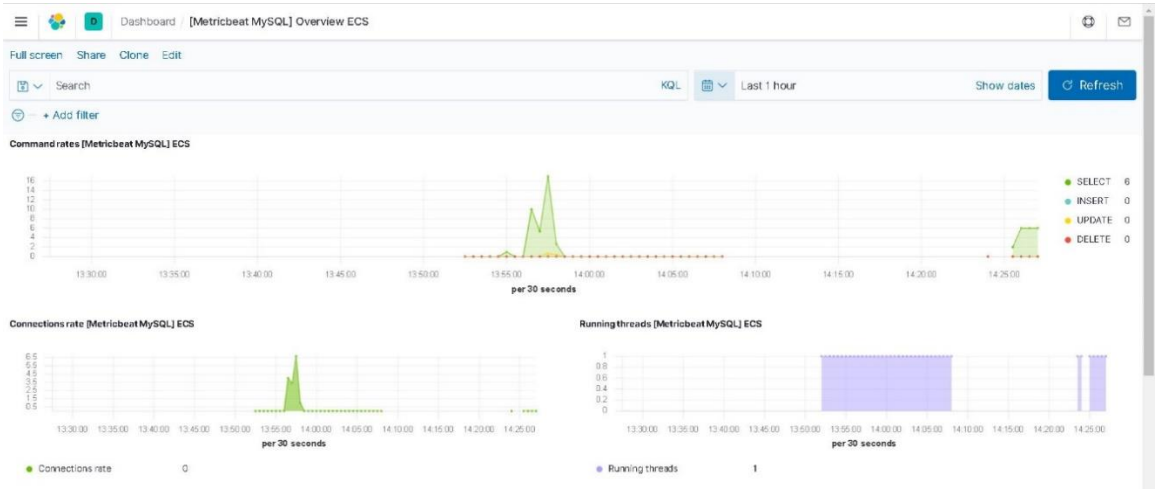


Figura 4-55 – *Dashboard* do Kibana no qual se podem visualizar as operações realizadas no MySQL

A compatibilidade com os diferentes sistemas operativos foi demonstrada na implementação do protótipo SIEM através dos Beats, pois estes foram instalados no sistema operativo Microsoft Windows e no Ubuntu. É de referir que não se procedeu à instalação do Kibana, do Logstash e do Elasticsearch no Microsoft Windows, pois definiu-se na arquitetura do protótipo que os servidores iriam estar instalados no sistema operativo Ubuntu.

A funcionalidade ECS permitiu normalizar os *logs* dos sistemas operativos utlizados na implementação do protótipo, para tal o Elasticsearch já disponibiliza *templates* para o efeito, como se pode visualizar na Figura 4-56.

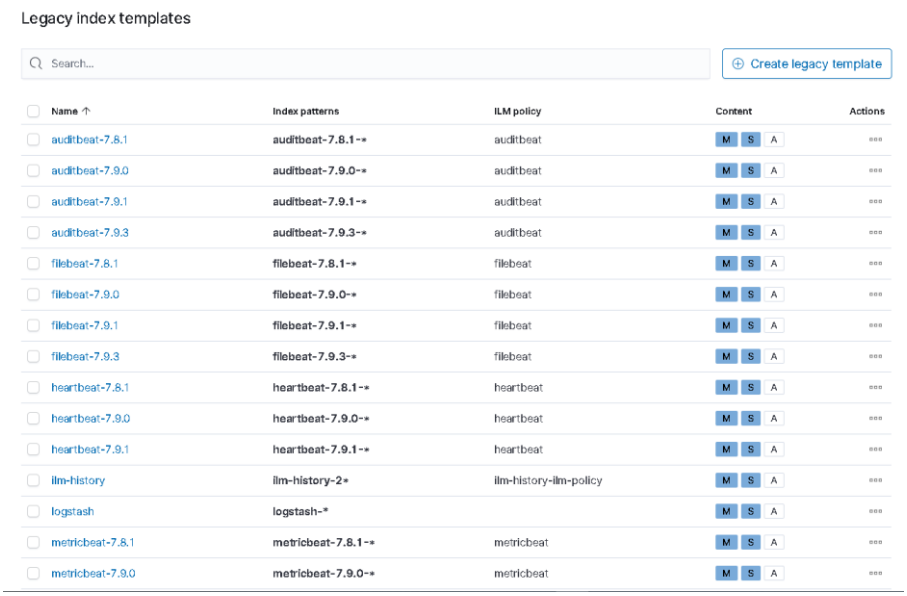


Figura 4-56 – *Templates* fornecidos pelo Elasticsearch que permitem normalizar os dados recorrendo ao ECS

Através da opção *Security*, no Kibana é possível que sejam analisadas as informações dos servidores em tempo real, apresentadas na forma de *Dashboards* que disponibilizam vários dados relacionados com a segurança. Na Figura 4-57 é apresentado um resumo das autenticações (sucessos/falhas), os números de endereços de IP distintos (origem/destino) de quatro clientes.

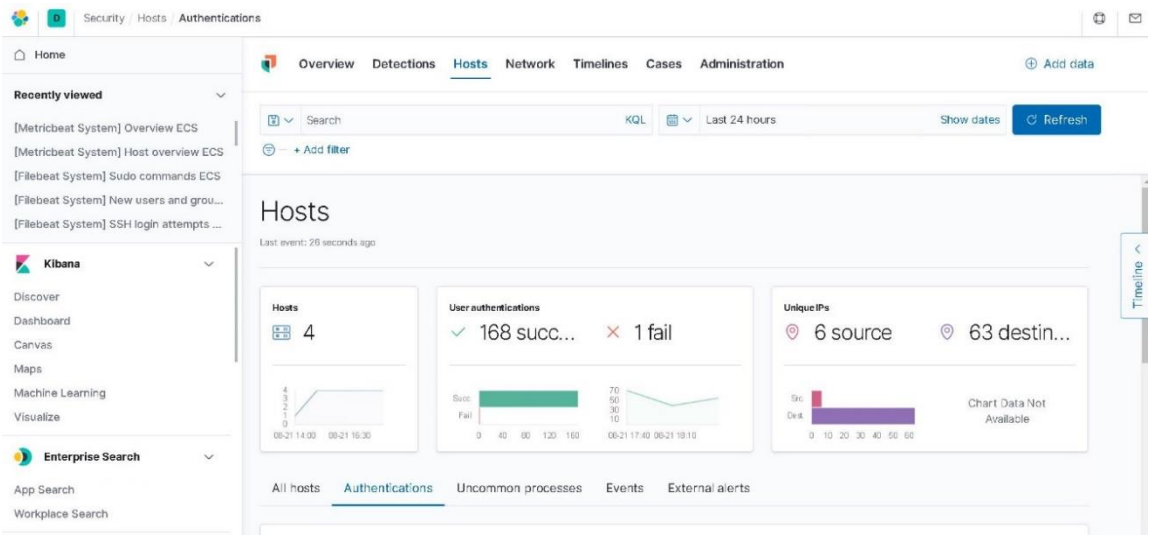


Figura 4-57 – *Dashboard* que apresenta um resumo do número de autenticações e o número de endereços IP distintos

Para validar a criação de *dashboards*, foram criados no protótipo dois *dashboards*, sendo que na imagem seguinte se encontra ilustrado o *dashboard* criado para o sistema operativo Linux, neste caso, são apresentados dois gráficos: um representa o número de *logs* criado pelo *Syslog* e o outro apresenta utilização da RAM.

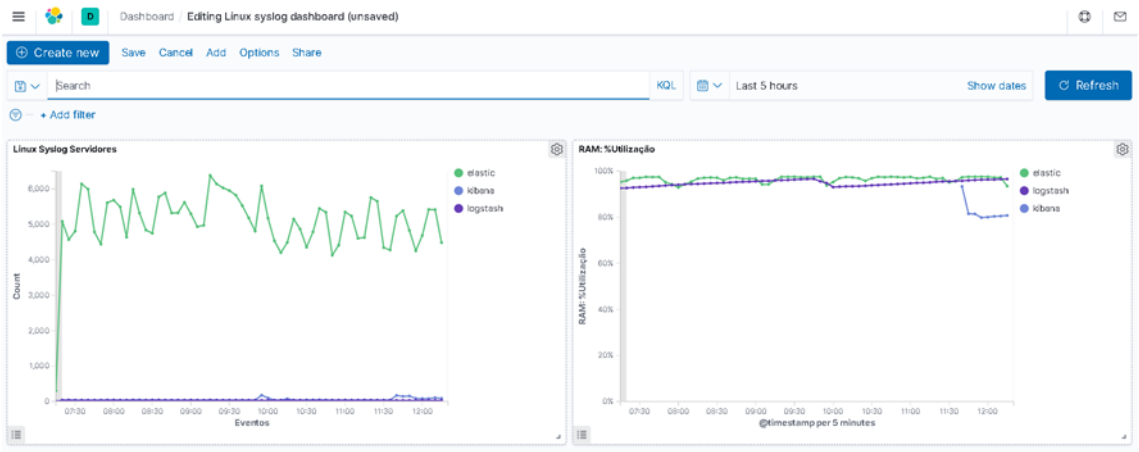


Figura 4-58 – *Dashboard* que apresenta o número de *logs* do *syslog* e a percentagem de utilização da RAM

Similarmente também foi criado um *dashboard* para o sistema operativo Microsoft Windows, neste é possível visualizar o número de *logs* produzido pelo sistema operativo, os

erros do sistema operativo Microsoft Windows e as tentativas de autenticação falhadas (ver Figura 4-59).

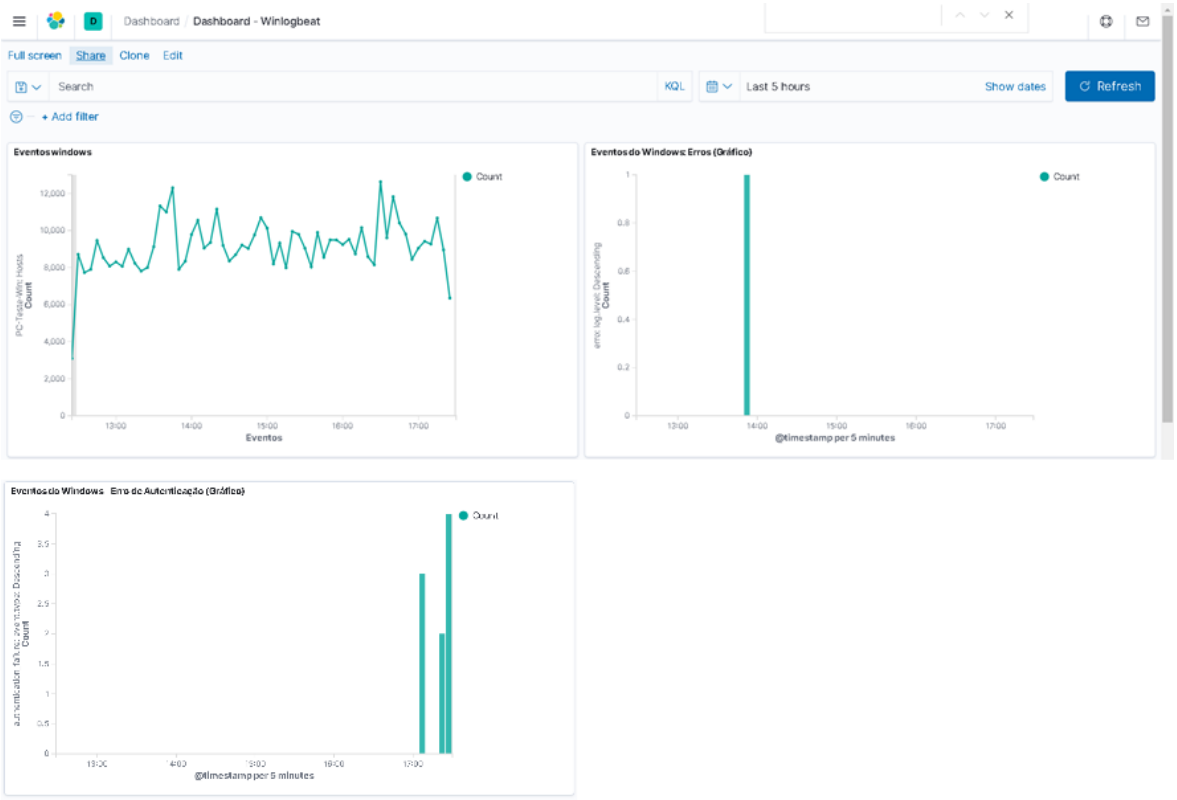


Figura 4-59 – *Dashboard* que apresenta o número de *logs* do Microsoft Windows, os seus erros e as tentativas de autenticações falhadas

No que diz respeito à criação de relatórios, não foi possível identificar uma ferramenta compatível com a versão do Kibana, contudo é de referir que o próprio Kibana permite a criação de relatórios sobre as pesquisas realizadas, apesar de serem bastante básicos. Com recurso à opção *Discover*, foi criada uma pesquisa e procedeu-se à gravação da mesma. De seguida, através da opção *Share* no *Discover*, foi escolhida a opção *Share* → *CSV Reports*. É de referir que os relatórios foram disponibilizados através da opção *Management* → *Stack Management* → *Reporting*. Na Figura 4-60 é ilustrado o processo de criação de um relatório no Kibana.

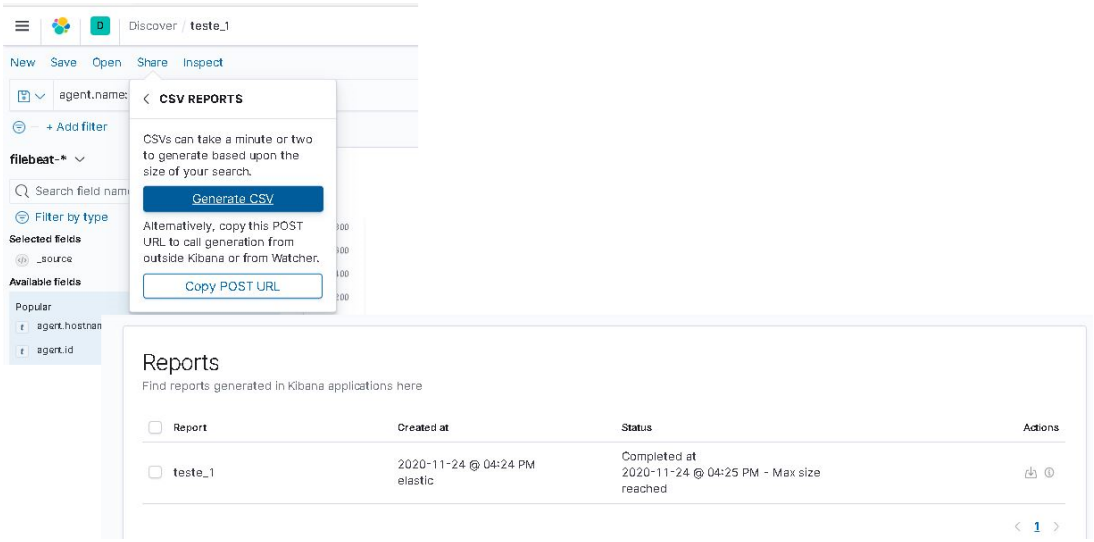


Figura 4-60 – Criação de um relatório no Kibana

Existem *dashboards* que destacam e que filtram os eventos pela sua criticidade, como se pode constatar pela visualização da próxima figura, na qual se podem ver os comandos que recorreram ao comando *sudo* para obter privilégios de administrador.

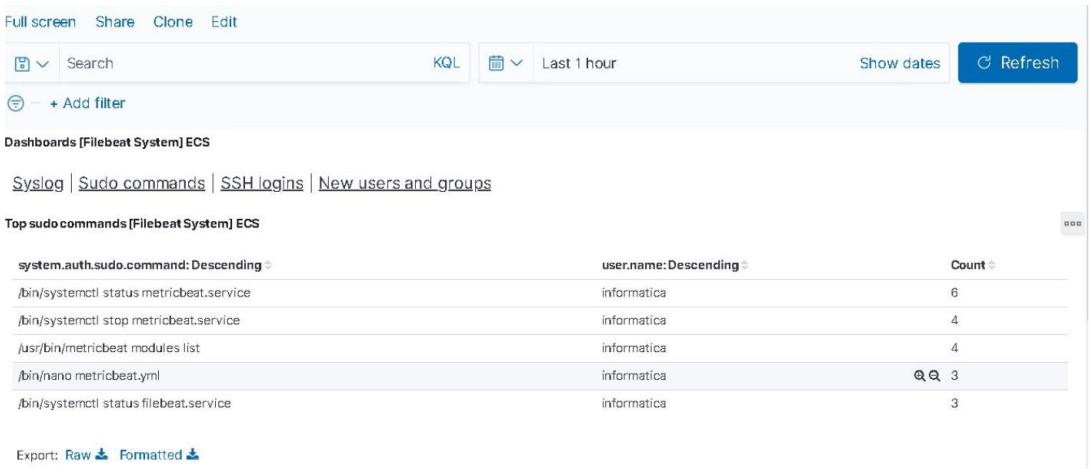


Figura 4-61 – *Dashboards* que lista os comandos recorreram ao comando *sudo*

4.4.5. Avaliação do protótipo

Nos pontos anteriores procurou-se demonstrar a usabilidade do protótipo, particularmente de várias das suas funcionalidades. Através da simulação de ataques, foi possível validar a deteção de ameaças ou de incidentes em tempo real. No cenário de ataque, também foi criada uma regra para que fosse emitido um alerta caso existissem várias tentativas de autenticação através do SSH.

Na Tabela 4.4 é apresentado o resumo das funcionalidades demonstradas ou testadas nos pontos anteriores. Como classificação, optou-se por aplicar Sim/Parcialmente, pois

considera-se que todos requisitos predefinidos estão representados no protótipo, mesmo que não o estejam na sua plenitude.

Tabela 4.4 – *Checklist* de validação dos pré-requisitos

Pré-requisitos	***Checklist* de validação dos pré-requisitos**	
	Sim/parcialmente	Observações
Recolher dados de diferentes origens	Sim	No protótipo recolheram-se *logs* dos sistemas operativos Microsoft Windows e Linux através dos Beats. Os Beats também são compatíveis com várias aplicações, como é exemplo o Zabbix, o MySQL ou o Oracle. Na Figura 4-55 são apresentadas as métricas do MySQL.
Fornecer compatibilidade com diferentes sistemas operativos	Sim	O Elasticsearch, o Kibana e o Logstash foram instalados no sistema operativo Linux, todavia estes componentes são compatíveis com o Microsoft Windows e com o MacOS. No entanto, os Beats, no protótipo, foram instalados no Ubuntu e no sistema operativo Microsoft Windows.
Normalizar *logs* de todos os componentes	Sim	Através da funcionalidade ECS, foi possível normalizar os *logs* de origens distintas, permitindo que se pudessem correlacionar os mesmos. Os logs recolhidos dos sistemas operativos foram normalizados através da funcionalidade ECS.
Analisar e correlacionar as informações dos equipamentos de tempo real	Sim	O Kibana fornece, na opção *Security*, a funcionalidade *Timelines*. Recorreu-se a esta funcionalidade para rastrear os *logs* criados pelo ataque, como, por exemplo, se pode constatar pela visualização da Figura 4-29.
Disponibilizar vários *dashboards* de segurança e permitir a personalização de outros	Sim	A solução escolhida já disponibiliza um grande número de *dashboards* de segurança, além desses, também foi possível criar outros. Como se pode observar na Figura 4-57 e na Figura 4-59.
Criar relatórios personalizados e alertas	Parcialmente	Não foi possível incorporar uma ferramenta para a criação de relatórios, contudo, o Kibana já disponibiliza múltiplos *dashboards* que podem ser utilizados para criar os referidos relatórios ou, então, é possível exportar o resultado das pesquisas para CSV (Figura 4-60).
Criar alertas	Sim	For incorporado na arquitetura do protótipo SEIM a ferramenta Elastalert, pois permite que sejam criados alertas. É possível visualizar a criação de um alerta na Figura 4-12.
Permitir filtrar e destacar os eventos	Sim	Para que fosse possível rastrear os *logs* do ataque, foi realizada uma pesquisa pelos eventos do Sysmon. Na Figura 4-38 é possível visualizar o resultado da pesquisa sublinhado a amarelo. Na Figura 4-61 é apresentado um *dashboard* que filtra os dados pela criticidade, enumerando a utilização através do comando *sudo*.
Detetar ameaças, incidentes de segurança e vulnerabilidades	Sim	Através das ferramentas disponibilizadas pelo Kibana foi possível rastrear as ações realizadas pela ferramenta Pupy, Hydra ou pelo script *Simple-SYN-Flood* .
Emita alertas nos casos em que existem atividades suspeitas na rede	Sim	O Elastalert permitiu a criação de múltiplas regras, sendo que na regra criada para este trabalho foi criado um alerta sempre que ocorressem três tentativas de autenticação falhadas no período temporal de uma hora (Figura 4-12).
Restringir os acessos e possibilite vários níveis de permissões	Sim	Através do Kibana, foi possível definir vários níveis de permissões (é possível visualizar alguns exemplos no Anexo H – Elastic Stack encriptação). Através do plugin Elasticsearch da ReadonlyRest foi possível restringir os acessos e atribuírem-se vários níveis de permissões, como se pode visualizar no ficheiro de configuração (do plugin Elasticsearch da ReadonlyRest) do ponto 4.3.6.
Fornecer vários mecanismos de segurança	Sim	Existem vários mecanismos de segurança, como por exemplo a autenticação, a possibilidade de serem atribuídos vários níveis de permissões, a encriptação e a auditoria aos acessos a um determinado índice.
Permitir a pseudonimização	Sim	A pseudonimização é realizada através da pipeline do Logstash (Figura 4-18).
Definir tempos de retenção para os dados	Sim	É possível definir tempos de retenção para dos dados (Figura 4-22).
Auditar os acessos a dados pessoais	Sim	Com o plugin Elasticsearch da ReadonlyRest, foi possível efetuar auditorias às operações realizadas pelos utilizadores autenticados no índice que contém os dados de recuperação. É possível visualizar o resultado da auditoria na Figura 4-17.
Restringir o acesso aos dados pessoais	Sim	Aos três utilizadores criados, foram atribuídos níveis de permissões diferentes, no ponto 4.3.6 é possível visualizar o ficheiro de configuração do plugin do ReadonlyRest.
Garantir a segurança dos dados pessoais	Sim	Na implementação do protótipo SIEM, foram efetuadas várias operações no sentido de limitar, de auditar e de garantir a segurança dos dados pessoais. São exemplos disso a pseudonimização, a encriptação e a auditoria às operações realizadas ao índice que possui as chaves de recuperação.
Garantir a integridade dos dados	Sim	A integridade dos dados é preservada através da encriptação e de vários níveis de permissões. O Elasticsearch faz a gestão da versão do *log*, sendo que a @version=”1” significa que o *logs* não foi aletrado (Figura 4-24).
Garantir a proteção de dados por design e por padrão	Parcialmente	O protótipo SIEM assegura a segurança dos dados em todo o seu ciclo de vida e foi aplicada a segurança que a solução fornece na sua versão *open-source*. Contudo, na versão comercial do produto as opções de segurança são mais robustas, sendo possível receber alertas sobre anomalias ou definir permissões ao nível do campo.

Tendo em conta o levantamento das funcionalidades elencadas na Tabela 4.4, constatamos que o cumprimento dos requisitos é quase total, pois apenas a criação de relatórios personalizados e a proteção de dados design e por padrão não tiveram um desempenho que permita assumir que cumprem na sua totalidade o pretendido. Neste contexto, não foi possível implementar uma ferramenta *open-source* para criar relatórios personalizados, contudo é possível exportar os dados para CSV e personalizar os mesmos.

Foi implementada a encriptação dos dados em todas as suas fases e existe a possibilidade de serem criados utilizadores com vários níveis de permissões, no entanto, a Solução Elastic Stack, na sua versão comercial, oferece uma solução mais robusta, pois permite a definição de permissões ao nível do campo.

Em suma, consideramos que esta arquitetura é ajustável à realidade de uma pequena ou média empresa, contudo será necessário validar os custos de implementação e de serem otimizadas algumas configurações para possa colocado em produção.

4.5. Síntese

Neste capítulo foi apresentada uma proposta para implementação de um protótipo de um SIEM *open-source*, tendo como base duas abordagens: uma que na qual foram aplicadas as medidas técnicas para estar em conformidade com o RGPD e outra em que estas não foram aplicadas. É de salientar que a abordagem que garante a conformidade com o RGPD é mais complexa de implementar, pois necessita de medidas adicionais de segurança, como por exemplo a auditoria ou a pseudonimização.

Para a implementação do protótipo SIEM *open-source* foram definidos os pré-requisitos, a arquitetura, quais os *logs* que seriam recolhidos e que medidas de segurança seriam implementadas. Posteriormente procedeu-se à criação das máquinas virtuais, à instalação e à configuração das diversas ferramentas.

Só depois de protótipo SIEM estar em funcionamento é que se implementou a camada de segurança disponibilizada pela solução Elastic Stack e também se procedeu à instalação do plugin Elasticsearch da ReadonlyRest, com o propósito de garantir uma atuação em conformidade com o RGPD.

É de referir que surgiram alguns problemas na instalação, na gestão e na configuração do protótipo SIEM que foi necessário ultrapassar, como por exemplo a incompatibilidade de bibliotecas ou dos problemas de configuração que foi necessário efetuar o *debug*.

É de referir que, para validar a deteção de ameaças no protótipo SIEM, simularam-se três ataques, um interno recorrendo à ferramenta Pupy e dois externos recorrendo à ferramenta Hydra e ao script *Simple-SYN-Flood*. Após os ataques, procedeu-se à investigação das "pegadas digitais" resultantes. Além disso, recolheram-se as métricas dos dois cenários e aferiu-se que, de uma forma geral, é necessário um esforço maior para efetuar a pseudonimização, particularmente no que diz respeito à utilização do CPU.

Tendo em conta os pré-requisitos definidos, efetuou-se a avaliação do protótipo através de uma *checklist* de validação. Verificou-se que a criação de relatórios e a proteção de dados por design e por padrão só foi parcialmente atingida. Além disso, não é possível criar relatórios da forma inicialmente pretendida, pese embora através do componente implementado, o Kibana, seja possível exportar os dados em CSV. Relativamente à proteção de dados por design e por padrão, também foram configurados vários níveis de segurança, contudo, na sua versão paga, através do Elastic Stack é possível definir níveis de permissão ao nível do campo.

Em sinopse, consideramos que o protótipo implementado, pelas suas funcionalidades, pode ser uma ferramenta útil para garantir a conformidade com o RGPD e, em simultâneo, permitir a análise e deteção de ameaças em tempo real, o que é muito importante para serem salvaguardados os dados de uma Organização.

5. Conclusões

A Segurança da Informação é um ponto fundamental em qualquer sistema ou Organização, a entrada em vigor do Regulamento Geral de Proteção de Dados e a obrigatoriedade do seu cumprimento veio reforçar a sua relevância. As Entidades que tratam dados pessoais têm de provar que implementam as medidas técnicas adequadas para a proteção e controlo dos dados pessoais.

A compreensão e utilização de um SIEM são importantes para melhorar a segurança da informação e promove a conformidade com o RGPD, pois possibilita que sejam identificados mais rapidamente ameaças ou incidentes de segurança.

Para que fosse possível concretizar os objetivos definidos inicialmente, este trabalho foi dividido em três etapas: pesquisa documental, estudo comparativo e desenvolvimento de um protótipo SIEM *open-source*.

No segundo capítulo, efetuou-se uma reflexão sobre vários conceitos chave: segurança da informação, *logs* de segurança, ataques, RGPD, gestores de *logs* e uma introdução aos SIEM. Com base nos resultados da pesquisa documental, foi possível enumerar as principais funcionalidades que um SIEM deve apresentar e também as medidas técnicas adequadas para a proteção e controlo de dados. Ainda no âmbito do pretendido neste capítulo foi efetuado um levantamento dos trabalhos académicos que abordam os gestores de *logs* e de SIEM. Os trabalhos académicos consultados foram uma base importante para a seleção de duas soluções SIEM *open-source*: o Graylog e o OSSIM. Pela sua flexibilidade e estabilidade, também foi escolhido o gestor de *logs* Elastic Stack, e, finalmente, o SIEM Splunk também foi selecionado porque disponibiliza uma versão *freeware*.

O principal objetivo do terceiro capítulo foi o de efetuar um estudo comparativo entre as quatro soluções identificadas no segundo capítulo e de escolher a ferramenta que seria utilizada na implementação do protótipo. Através do estudo comparativo realizado, foi possível identificar pontos fortes e fracos das soluções. É de referir que foi necessário fundamentar a escolha da ferramenta: o Elastic Stack, sobretudo porque este não é um SIEM, contudo, com a articulação com outras ferramentas, pode-se tornar num.

Tendo em conta os resultados da pesquisa documental realizada para a redação do segundo capítulo, foi possível identificar técnicas e procedimentos a implementar que garantem a proteção dos dados pessoais e, muito importante, a conformidade com o RGPD. No terceiro capítulo, com recurso aos resultados da pesquisa documental em articulação com uma competente prática, uma vez que foi criado um cenário de testes com o propósito de testar a usabilidade das soluções e a sua eficiência na deteção de um ataque de força bruta, foi possível realizar o estudo comparativo entre as soluções selecionadas e identificar os pontos fortes e fracos de cada uma. Os restantes objetivos elencados foram atingidos com a implementação, documentação e avaliação do protótipo.

No quarto capítulo, foi descrito o processo de implementação do protótipo, mapeando as funcionalidades de um SIEM enumeradas no segundo capítulo com os pré-requisitos que o protótipo tem de possuir para que o mesmo possa ser denominado de SIEM. Foi ainda documentada a implementação e avaliação do protótipo, tendo como ponto de referência os pré-requisitos definidos.

Consideramos que o principal objetivo do trabalho foi alcançado, uma vez que foi implementado um sistema SIEM *open-source*, com recurso à aplicação de medidas técnicas que garantem a proteção e controlo dos dados pessoais, assegurando assim uma atuação em conformidade com o RGPD. Uma das medidas referenciadas no regulamento é a pseudonimização dos dados pessoais, pelo que foi garantido que o protótipo psudonimiza os dados pessoais identificados nos *logs* de segurança (nome do computador, utilizador e IP origem) e, além disso, também permite auditar os acessos aos índices que têm a informação que permite a identificação do Titular, neste caso o utilizador do computador.

No protótipo optou-se por dividir os dados que estão pseudonimizados, o que implicou que fosse necessário um servidor secundário. Para a pseudonimização dos dados utilizou-se uma pipeline do Logstash, sendo de referir que a mesma recorre a filtros que naturalmente vão aumentar a utilização do CPU. Por esta razão recolheram-se métricas referentes à utilização do SIEM com a pseudonimização e sem a pseudonimização para tentar perceber o impacto desta operação.

Para os cenários onde foi efetuada a ingestão dos 10000 *logs*, na métrica do CPU a operação que não realiza a pseudonimização requere do CPU 94% e a que realiza a pseudonimização requere 91%, aparentemente estamos perante um contrassenso, contudo se analisarmos os gráficos da ingestão dos *logs* verificamos que o período temporal para a ingestão dos *logs* é

maior no cenário onde se realiza a pseudonimização. Se efetuarmos a comparação para o mesmo *log*, sem a pseudonimização é requerido 2% do CPU e com a pseudonimização é requerido 60%, podendo-se concluir que para processar o mesmo número de *logs* com pseudonimização é requerido um maior esforço do CPU.

Contudo, os resultados obtidos nas métricas não podem ser extrapoláveis, porque foi efetuado os testes com o mesmo *log* e o número de operações efetuadas na pipeline influenciam a utilização do CPU.

Durante a realização da parte prática foi necessário enfrentar vários desafios, as atualizações da solução Elastic Stack criaram alguns problemas na implementação, porque algumas das configurações ficaram obsoletas e foi necessário repensar tudo de novo. Dou exemplo do campo *type* que estava a ser utilizado na criação da pipeline e foi descontinuado pela solução. Também existiram problemas nos *Dashboards* criados, uma vez que uma das atualizações obrigou a voltar a criar os mesmos.

Existiram vários problemas de incompatibilidades de livrarias que foi necessário resolver, ou de configurações que foi necessário fazer um *debug* muito minucioso. O tempo de aprendizagem foi demorado pois o cenário implementado tornou-se complexo e foi necessário gerir várias ferramentas *open-source*.

A gestão do hardware também criou alguns problemas, porque não foi fácil gerir o problema do armazenamento ou os requisitos de CPU e de RAM. Inicialmente os servidores Kibana, Logstash e Elastic Stack estavam na mesma máquina, mas o seu desempenho era muito lento foi necessário arranjar alternativas.

Consideramos que a arquitetura modular do protótipo e pode ser uma mais valia para a gestão da segurança e conformidade com o RGPD, contudo ainda falta otimizar o mesmo para produção.

Em suma, entendemos que o presente trabalho concretiza os objetivos definidos, sendo, no entanto, de referir não foram implantados todos os requisitos identificados, como é exemplo o *machine learning* ou o da criação de relatórios de conformidade.

5.1. Contributo científico do estudo

O presente estudo relacionou a implementação de um SIEM *open-source* com as medidas técnicas necessárias para a proteção e controlo dos dados pessoais, com propósito de

assegurar a conformidade da atuação do SIEM com o RGPD. Enumeramos os principias contributos deste estudo:

- Implementar um sistema SIEM *open-source*, incorporando medidas técnicas para a proteção e controlo dos dados pessoais assegurando a conformidade com o RGPD;
- Identificar as técnicas e os procedimentos mais adequados para a proteção dos dados pessoais;
- Realizar o estudo comparativo entre as soluções SIEM, identificando os pontos fortes e fracos de cada solução estudada;
- Definir e implementar uma arquitetura de um sistema SIEM baseado em soluções *open-source* e que permita aplicar as medidas técnicas para a proteção e controlo de dados pessoais, assegurando a conformidade com o RGPD;
- Criar um protótipo que implemente a arquitetura definida de um SIEM *open-source*;
- Documentar a implementação do protótipo;
- Testar e Avaliar o desempenho do protótipo.

É de referir que quando se procedeu à pesquisa e análise documental, não se encontrou nenhum trabalho que relacionasse os dois conceitos anteriores, pelo que entendemos que o presente trabalho contribuiu para o avançar do conhecimento sobre as temáticas abordadas.

No decorrer da realização do trabalho, foi elaborado e publicado um artigo sobre o tema, com o título SIEM Open Source Solutions: A Comparative Study (Vazão, Santos, Piedade, & Rabadão, 2019), que partilhou algumas das conclusões com a comunidade académica, tornando-as acessíveis a um maior número de investigadores.

5.2. Tópicos para Trabalho Futuro

Ao longo da realização do trabalho, devido à natureza do que nos propusemos elaborar no âmbito da tese de mestrado, existiram vários requisitos que foram remetidos para trabalho futuro e dos quais listamos:

- garantir a escalabilidade e a tolerância a falhas;
- garantir a resiliência e a recuperação a desastres
- realizar análises complexas recorrendo ao *machine learning*;

- assegurar a gestão de incidentes;
- garantir a atualização em tempo real de ameaças;
- realizar análises forenses aos *logs*, em tempo real, ou aos *logs* armazenados pelo sistema;
- automatizar várias funções, reduzindo o tempo gasto na sua manutenção;
- efetuar a notificação da violação de dados e a criação de relatórios de conformidade.

Para além dos requisitos já elencados, consideramos que seria importante implementar o protótipo em produção, refinando algumas das configurações para que o mesmo possa ser uma ferramenta indispensável na deteção de ataques e de vulnerabilidades, com uma atuação em conformidade com o RGPD.

Bibliografia

Alien Vault. (2018). Compare OSSIM to USM | AlienVault. Retrieved May 27, 2018, from https://goo.gl/yo2uUx

AlienVault. (2019). Compare OSSIM to USM | AlienVault. Retrieved January 8, 2019, from https://goo.gl/yo2uUx

Alves, J. (2017). *Threat intelligence: using osint and security metrics to enhance siem capabilities*. Universidade de Lisboa. Retrieved from http://hdl.handle.net/10451/31162

Anastasov, I., & Davcev, D. (2014). SIEM implementation for global and distributed environments. In *2014 World Congress on Computer Applications and Information Systems (WCCAIS)* (pp. 1–6). IEEE.

Andhavarapu, A. (2017). *Learning Elasticsearch*. Packt Publishing. Retrieved from https://books.google.pt/books?id=2nc5DwAAQBAJ

Anomali. (2020). O que é MITRE ATT&CK™? Retrieved February 27, 2020, from https://www.anomali.com/pt/what-mitre-attck-is-and-how-it-is-useful

Arass, M., & Souissi, N. (2019). Smart SIEM: From Big Data Logs and Events To Smart Data Alerts. *International Journal of Innovative Technology and Exploring Engineering*, *Volume-8*(Issue-8), 7.

AT&T Cybersecurity. (2019a). About USM Appliance System Architecture and Components. Retrieved July 26, 2019, from https://www.alienvault.com/documentation/usm-appliance/system-overview/about-usm-architecture-components.htm

AT&T Cybersecurity. (2019b). Advanced Search Criteria for SIEM in AlienVault USM Appliance. Retrieved July 30, 2019, from https://www.alienvault.com/documentation/usm-appliance/events/advanced-search-criteria-for-security-events.htm

AT&T Cybersecurity. (2019c). AlienVault OSSIM Report Type. Retrieved July 30, 2019,

from https://www.alienvault.com/documentation/usm-appliance/reports/ossim-report-types.htm?Highlight=reports

AT&T Cybersecurity. (2019d). Event Storage Best Practices for AlienVault USM Appliance. Retrieved July 30, 2019, from https://www.alienvault.com/documentation/usm-appliance/events/event-storage-best-practices.htm

AT&T Cybersecurity. (2020a). AlienVault OSSIM® Installation Process. Retrieved October 24, 2020, from https://cybersecurity.att.com/documentation/usm-appliance/initial-setup/ossim-installation.htm

AT&T Cybersecurity. (2020b). USM Appliance™ - Deployment Guide. AT&T Cybersecurity. Retrieved from https://www.alienvault.com/documentation/resources/pdf/usm-appliance-deployment-guide.pdf

AV-TEST. (2019). *Security Report 2017/2018*. Germany. Retrieved from https://www.av-test.org/fileadmin/pdf/security_report/AV-TEST_Security_Report_2018-2019.pdf

Banon, S. (2019a). Dear Search Guard Users. Retrieved August 20, 2020, from https://www.elastic.co/pt/blog/dear-search-guard-users

Banon, S. (2019b). Dear Search Guard users #2, including Amazon Elasticsearch Service and Open Distro, and others. Retrieved August 20, 2020, from https://www.elastic.co/pt/blog/dear-search-guard-users-including-amazon-elasticsearch-service-open-distro-and-others

Bassett, S., & Paquette, M. (2018). Melhore a análise de segurança com o Elastic Stack, Wazuh e IDS | Elastic Blog. Retrieved June 6, 2019, from https://www.elastic.co/pt/blog/improve-security-analytics-with-the-elastic-stack-wazuh-and-ids

Baxter, J. (2018). *Splunk 7.x Quick Start Guide: Gain business data insights from operational intelligence*. Packt Publishing. Retrieved from https://books.google.pt/books?id=Ut18DwAAQBAJ

Beggs, R. (2014). *Mastering Kali Linux for Advanced Penetration Testing*. Packt

Publishing. Retrieved from https://books.google.pt/books?id=nSXiAwAAQBAJ

Bělousov, P. (2019). *Security Enhancement Deploying Siem in a Small Isp Environment*. Brno University of Technology.

Berman, D. (2018). Using the ELK Stack for SIEM | Logz.io. Retrieved May 20, 2019, from https://logz.io/blog/elk-siem/

Black, C. (2017). *Get One Step Closer To GDPR Compliance*. Retrieved from https://www.graylog.org/resources/get-one-step-closer-to-gdpr-compliance

Borkar, P. (2018). Comparing Security Data Lakes that Leverages ELK for Cybersecurity. Retrieved August 10, 2019, from https://www.exabeam.com/siem/data-lakes/

Boucas, E. (2018). Importance of Using SIEM for GDPR Compliance. Retrieved March 8, 2019, from https://www.cpomagazine.com/cyber-security/importance-of-using-siem-for-gdpr-compliance/

Branquinho, M., Seidl, J., Moraes, L., Branquinho, T., & Azevedo, J. (2014). *Segurança de Automação Industrial e SCADA*. Elsevier Editora Ltda. Retrieved from https://books.google.pt/books?id=FVkaBQAAQBAJ

Brown, L., & Stallings, W. (2017). *Segurança de Computadores: Princípios e Práticas*. (E. Brasil, Ed.). Elsevier Editora Ltda. Retrieved from https://books.google.pt/books?id=y2DcAwAAQBAJ

Bryan, L., & Falcone, R. (2017). Magic Hound Campaign Attacks Saudi Targets. Retrieved September 12, 2020, from https://unit42.paloaltonetworks.com/unit42-magic-hound-campaign-attacks-saudi-targets/

Caltagirone, S., Pendergast, A., & Betz, C. (2013). *The diamond model of intrusion analysis*. Center For Cyber Intelligence Analysis and Threat Research Hanover Md.

Catescu, G. (2018). *Detecting insider threats using Security Information and Event Management (SIEM)*. University of Applied Sciences Technikum Wien.

Chappell, G. (2019). Kibana 7.0.0 released | Elastic Blog. Retrieved July 2, 2019, from https://www.elastic.co/pt/blog/kibana-7-0-0-released

Chapple, M., & Seidl, D. (2017). *CompTIA CySA+ Study Guide: Exam CS0-001*. Wiley.

Retrieved from https://books.google.pt/books?id=QKilDgAAQBAJ

Chebbi, C. (2018). *Advanced Infrastructure Penetration Testing: Defend your systems from methodized and proficient attackers*. Packt Publishing. Retrieved from https://books.google.pt/books?id=BulODwAAQBAJ

Chhajed, S. (2015). *Learning ELK Stack*. Packt Publishing. Retrieved from https://books.google.pt/books?id=EfqoCwAAQBAJ

Chuvakin, A., Schmidt, K., & Phillips, C. (2012). *Logging and Log Management: The Authoritative Guide to Understanding the Concepts Surrounding Logging and Log Management*. (Newnes, Ed.). Elsevier Science. Retrieved from https://books.google.pt/books?id=Rf8M_X_YTUoC

Collier, R., & Azarmi, B. (2019). *Machine Learning with the Elastic Stack: Expert techniques to integrate machine learning with distributed search and analytics*. Packt Publishing. Retrieved from https://books.google.pt/books?id=jyOGDwAAQBAJ

Collins, M. (2017). *Network Security Through Data Analysis: From Data to Action*. O'Reilly Media. Retrieved from https://books.google.pt/books?id=0bM0DwAAQBAJ

Comissão Europeia. (2019). O que significa a proteção de dados «desde a conceção» e «por defeito»? Retrieved December 16, 2019, from https://ec.europa.eu/info/law/law-topic/data-protection/reform/rules-business-and-organisations/obligations/what-does-data-protection-design-and-default-mean_pt

Constantine, C. (2019). How does SIEM logging work? Retrieved October 21, 2019, from https://www.alienvault.com/blogs/security-essentials/everything-you-wanted-to-know-about-siem-and-log-management-but-were-afraid

Contreras, J., Koelpin, S., Delgado, E., & Sigman, E. (2018). *Splunk 7 Essentials, Third Edition: Demystify machine data by leveraging datasets, building reports, and sharing powerful insights, 3rd Edition*. Packt Publishing. Retrieved from https://books.google.pt/books?id=kLZTDwAAQBAJ

Corcoran, C. (2017). ELK Stack for Security Operations, Analytics and Reporting - OPAQ. Retrieved May 18, 2019, from https://opaq.com/newsroom/elk-stack-for-security-operations-analytics-and-reporting/

Costa, B. (2017). O novo regulamento geral de proteção de dados (GDPR) da UE e o impacto nos negócios de ecommerce | Tudo Sobre eCommerce. Retrieved March 9, 2019, from https://tsecommerce.com/blog/novo-regulamento-geral-protecao-dados-gdpr-da-ue-impacto-nos-negocios-ecommerce/

Cybershark. (2019). A Guide to SIEM and Log Management Solutions | BlackStratus. Retrieved October 21, 2019, from https://www.blackstratus.com/siem-log-management-solutions/

Daubner, L. (2018). *Effective computer infrastructure monitoring*. Masarykova univerzita. Retrieved from https://is.muni.cz/th/zo07q/

Delgado, P. (2018). *Developing an Adaptive Threat Hunting Solution: The Elasticsearch Stack*. University of Houston.

Detken, K., Scheuermann, D., & Hellmann, B. (2015). Using extensible metadata definitions to create a vendor-independent SIEM system. In *International Conference in Swarm Intelligence* (pp. 439–453). Springer.

Development, a2o F. (2020). Snoopy. Retrieved from https://github.com/a2o/snoopy

Devender, V., & Adike, S. (2019). *Design and Performance of an Event Handling and Analysis Platform for vSGSN-MME event using the ELK stack*. Faculty of Computing. Retrieved from http://www.diva-portal.org/smash/get/diva2:1303973/FULLTEXT02.pdf

Dezeure, F. (2018). A Layman's Guide on How to Operate Your SIEM Under the GDPR. Splunk. Retrieved from https://cdn2.hubspot.net/hubfs/484638/GDPR for Higher Ed/A Layman%27s Guide on How to Operate Your SIEM Under the GDPR.pdf?t=1526063621493

Diakun, J., Johnson, P., & Mock, D. (2014). *Splunk Operational Intelligence Cookbook*. Packt Publishing. Retrieved from https://books.google.pt/books?id=sNwkBQAAQBAJ

Diakun, J., Johnson, P., & Mock, D. (2018). *Splunk Operational Intelligence Cookbook: Over 80 recipes for transforming your data into business-critical insights using Splunk, 3rd Edition*. Packt Publishing. Retrieved from

https://books.google.pt/books?id=cHteDwAAQBAJ

Dienst, J. (2019). Graylog vs. LogRhythm NextGen SIEM Comparison - UPDATED 2019 | IT Central Station. Retrieved July 6, 2019, from https://www.itcentralstation.com/products/comparisons/graylog_vs_logrhythm-nextgen-siem

Dixit, B. (2017). *Mastering Elasticsearch 5.x*. Packt Publishing. Retrieved from https://books.google.pt/books?id=F1QoDwAAQBAJ

DNSstuff. (2019). 10 Best Free and Open-Source SIEM Tools in 2020. Retrieved July 15, 2020, from https://www.dnsstuff.com/free-siem-tools#splunk-free

E-Cogni Treinamentos. (2017). O uso da ELK stack para auxiliar na gestão de logs - E-cogni Treinamentos de TI. Retrieved May 17, 2019, from https://www.e-cogni.com.br/sysadmin/o-uso-da-elk-stack-para-auxiliar-na-gestao-de-logs/

EC-Council. (2016). *Computer Forensics: Investigating Network Intrusions and Cybercrime (CHFI)*. Cengage Learning. Retrieved from https://books.google.pt/books?id=5QFVDAAAQBAJ

Educba. (2020). Is Splunk Open Source? Retrieved August 20, 2020, from https://www.educba.com/is-splunk-open-source/

Elastic. (2018). GDPR Compliance & The Elastic Stack. Elastic. Retrieved from https://www.elastic.co/pdf/white-paper-elastic-gdpr-compliance-and-the-elastic-stack.pdf

Elastic. (2019). Fundamentals of securing elasticsearch. Elastic. Retrieved from https://www.elastic.co/training/fundamentals-of-securing-elasticsearch

Elastic. (2020a). Configure security. Retrieved September 20, 2020, from https://www.elastic.co/guide/en/kibana/current/xpack-security-audit-logging.html

Elastic. (2020b). Creaying Index Lifecycle Policies. Retrieved September 15, 2020, from https://www.elastic.co/guide/en/kibana/current/creating-index-lifecycle-policies.html

Elastic. (2020c). Stashing Your First Event. Retrieved August 25, 2020, from https://www.elastic.co/guide/en/logstash/7.x/first-event.html

ElasticHQ. (2018). ElasticHQ - Elasticsearch Management and Monitoring. Retrieved May 20, 2019, from https://www.elastichq.org/features.html

Elasticsearch. (2019a). Aggregations | Elasticsearch Reference [7.0] | Elastic. Retrieved May 3, 2019, from https://www.elastic.co/guide/en/elasticsearch/reference/current/search-aggregations.html

Elasticsearch. (2019b). Alerting: Alerts & Notifications for Elasticsearch | Elastic. Retrieved May 3, 2019, from https://www.elastic.co/pt/products/stack/alerting

Elasticsearch. (2019c). Auditbeat. Retrieved April 22, 2019, from https://www.elastic.co/pt/products/beats/auditbeat

Elasticsearch. (2019d). Basic Concepts | Elasticsearch Reference [7.0] | Elastic. Retrieved May 1, 2019, from https://www.elastic.co/guide/en/elasticsearch/reference/current/getting-started-concepts.html

Elasticsearch. (2019e). Beats: Data Shippers for Elasticsearch | Elastic. Retrieved April 22, 2019, from https://www.elastic.co/pt/products/beats

Elasticsearch. (2019f). Community Beats | Beats Platform Reference [7.0] | Elastic. Retrieved May 18, 2019, from https://www.elastic.co/guide/en/beats/libbeat/current/community-beats.html

Elasticsearch. (2019g). Deploying and Scaling Logstash | Logstash Reference [7.0] | Elastic. Retrieved April 27, 2019, from https://www.elastic.co/guide/en/logstash/current/deploying-and-scaling.html

Elasticsearch. (2019h). Elastic Stack Features (Formerly X-Pack): Extend Elasticsearch, Kibana, Beats & Logstash | Elastic. Retrieved April 19, 2019, from https://www.elastic.co/pt/products/stack

Elasticsearch. (2019i). Elastic Subscriptions. Retrieved February 9, 2019, from https://goo.gl/RRZj7h

Elasticsearch. (2019j). Elasticsearch-Powered SaaS Offerings. Retrieved April 22, 2019, from https://www.elastic.co/pt/cloud/

Elasticsearch. (2019k). Elasticsearch: RESTful, Distributed Search. Retrieved April 29, 2019, from https://www.elastic.co/pt/products/elasticsearch

Elasticsearch. (2019l). Elasticsearch Reference [7.0] | Elastic. Retrieved May 3, 2019, from https://www.elastic.co/guide/en/elasticsearch/reference/current/index.html

Elasticsearch. (2019m). Filebeat: Lightweight Log Analysis & Elasticsearch | Elastic. Retrieved April 22, 2019, from https://www.elastic.co/pt/products/beats/filebeat

Elasticsearch. (2019n). Functionbeat: Lightweight Shipper for Cloud Data | Elastic. Retrieved April 25, 2019, from https://www.elastic.co/pt/products/beats/functionbeat

Elasticsearch. (2019o). Getting started | Elasticsearch Reference [7.0] | Elastic. Retrieved April 29, 2019, from https://www.elastic.co/guide/en/elasticsearch/reference/current/getting-started.html

Elasticsearch. (2019p). Heartbeat: Monitor Services with Elasticsearch | Elastic. Retrieved April 25, 2019, from https://www.elastic.co/pt/products/beats/heartbeat

Elasticsearch. (2019q). Kibana User Guide [7.0] | Elastic. Retrieved May 3, 2019, from https://www.elastic.co/guide/en/kibana/current/index.html

Elasticsearch. (2019r). License settings | Elasticsearch Reference [7.0] | Elastic. Retrieved April 22, 2019, from https://www.elastic.co/guide/en/elasticsearch/reference/current/license-settings.html

Elasticsearch. (2019s). Logstash: Collect, Parse, Transform Logs | Elastic. Retrieved April 25, 2019, from https://www.elastic.co/pt/products/logstash

Elasticsearch. (2019t). Metricbeat: Lightweight Shipper for Metrics | Elastic. Retrieved April 22, 2019, from https://www.elastic.co/pt/products/beats/metricbeat

Elasticsearch. (2019u). Packetbeat: Network Analytics Using Elasticsearch | Elastic. Retrieved April 22, 2019, from https://www.elastic.co/pt/products/beats/packetbeat

Elasticsearch. (2019v). Query DSL | Elasticsearch: The Definitive Guide [master] | Elastic. Retrieved May 2, 2019, from https://www.elastic.co/guide/en/elasticsearch/guide/master/query-dsl-intro.html

Elasticsearch. (2019w). Reporting. Retrieved May 3, 2019, from

https://www.elastic.co/pt/products/stack/reporting

Elasticsearch. (2019x). SIEM Guide (Beta) [7.2] | Elastic. Retrieved July 24, 2019, from https://www.elastic.co/guide/en/siem/guide/current/index.html

Elasticsearch. (2019y). SIEM on the Elastic Stack. Retrieved July 25, 2019, from https://www.elastic.co/pt/products/siem

Elasticsearch. (2019z). Use Cases | Elastic Stack Success Stories | Elastic Customers. Retrieved May 18, 2019, from https://www.elastic.co/pt/use-cases/

Elasticsearch. (2019aa). Winlogbeat: Analyze Windows Event Logs | Elastic. Retrieved April 22, 2019, from https://www.elastic.co/pt/products/beats/winlogbeat

Elbaz, L. (2016). *Beginner's Guide to Information Security*. Peerlyst.

Escola Superior de Redes. (2015). *Tratamento de Incidentes de Segurança*. Rio de Janeiro: Escola Superior de Redes.

EUR-Lex. (2016). Regulamento (UE) 2016/679. *Jornal Oficial Da União Europeia*, (Legislação L119. 59º ano. 4 de maio). Retrieved from http://eur-lex.europa.eu/legalcontent/PT/TXT/PDF/?uri=OJ:L:2016:119:FULL&from=EN

European Commission. (2020). About the regulation and data protection. Retrieved from https://ec.europa.eu/info/law/law-topic/data-protection/eu-data-protection-rules_en

Exabeam. (2019). Ten Must-Have Features of a Modern SIEM. *Infosecurity Magazine*, p. 11. Retrieved from https://www.infosecurity-magazine.com/white-papers/features-of-a-modern-siem/

Exabeam. (2020a). Machine Learning for Cybersecurity: Next-Gen Protection Against Cyber Threats. Retrieved October 26, 2020, from https://www.exabeam.com/information-security/machine-learning-for-cybersecurity/

Exabeam. (2020b). The Essential Guide to SIEM. Retrieved July 26, 2020, from https://www.exabeam.com/siem-guide/

Farmer, G., Crow, J., & Rodgers, C. (2019). Graylog vs. Splunk Comparison | IT Central Station. Retrieved August 10, 2019, from https://www.itcentralstation.com/products/comparisons/graylog_vs_splunk

Fernandes, P. (2015). *Arquitetura de um SIEM tolerante a falhas para análise forense*. Universidade de Lisboa. Retrieved from http://hdl.handle.net/10451/20375

Ferreira, L. (2017). *A multi-level model for risk assessment in SIEM*. Universidade de Lisboa. Retrieved from http://hdl.handle.net/10451/31288

Floragunn GmbH. (2018). Licensing Model Overview. Search Guard. Retrieved from https://search-guard.com/wp-content/uploads/2018/03/SG_Licensing-model-overview.pdf

Fortinet. (2019). *What Really Matters When Selecting a Security Information and Event Management Solution*. Retrieved from https://www.fortinet.com/content/dam/fortinet/assets/ebook/eb-security-information-event-management.pdf

Garth, C. (2018). What is OSSIM. Retrieved June 13, 2020, from https://espprojects.co.uk/general/what-is-ossim/

Gartner. (2018). Do Use Splunk As A SIEM Or A Log Management Technology. Retrieved August 10, 2019, from https://www.gartner.com/reviews/review/view/580347

Gartner. (2020). Security Information And Event Management. Retrieved May 30, 2020, from https://www.gartner.com/en/information-technology/glossary/security-information-event-management

Google Trends. (2018). elasticsearch + logstash + kibana, splunk - Explore - Google Trends. Retrieved October 15, 2018, from https://trends.google.com/trends/explore?date=all&q=elasticsearch %2B logstash %2B kibana,splunk&hl=en-US

Gordon, S. (2010). *Operationalizing Information Security: Putting the Top 10 SIEM Best Practices to Work*. Gordon, Scott.

Granjal, J. (2017). *Segurança Prática em Sistemas e Redes com Linux*. Lisboa: FCA - Editora de Informatica, lda.

Graylog. (2018a). Architectural considerations — Graylog 3.0.0 documentation. Retrieved June 12, 2019, from https://docs.graylog.org/en/3.0/pages/architecture.html

Graylog. (2018b). Introduction to Alerting | Graylog. Retrieved June 25, 2019, from https://www.graylog.org/features/alerting

Graylog. (2018c). Resources | The Graylog Advantage | Graylog. Retrieved June 25, 2019, from https://www.graylog.org/resources/the-graylog-advantage

Graylog. (2018d). SIEM, Simplified | The Graylog Blog. Retrieved July 6, 2019, from https://www.graylog.org/post/siem-simplified

Graylog. (2018e). Welcome to the Graylog documentation — Graylog 3.0.0 documentation. Retrieved June 12, 2019, from https://docs.graylog.org/en/3.0/index.html

Graylog. (2018f). What GDPR Means for Log Management | Graylog. Retrieved June 15, 2019, from https://www.graylog.org/what-gdpr-means-for-log-management

Graylog. (2019). Introduction to Content Packs | Graylog. Retrieved July 8, 2019, from https://www.graylog.org/features/content-packs

Graylog. (2020a). External authentication. Retrieved October 24, 2020, from https://docs.graylog.org/en/3.3/pages/users_and_roles/external_auth.html

Graylog. (2020b). GRAYLOG ENTERPRISE EDITION. Retrieved August 20, 2020, from https://www.graylog.org/graylog-enterprise-edition

Graylog. (2020c). Installing Graylog. Retrieved October 24, 2020, from https://docs.graylog.org/en/3.3/pages/installation.html

Graylog. (2020d). Microsoft Windows. Retrieved October 24, 2020, from https://docs.graylog.org/en/3.3/pages/installation/windows.html?highlight=windows#microsoft-windows

Graylog. (2020e). Open Source VS. Enterprise. Retrieved February 8, 2020, from https://goo.gl/JjhZDG

Graylog Marketplace. (2019). Graylog Marketplace. Retrieved June 14, 2019, from https://marketplace.graylog.org/addons?kind=plugin

Gregory, L. (2018). Introducing Kibana Spaces for Organization and Security | Elastic Blog. Retrieved June 3, 2019, from https://www.elastic.co/pt/blog/introducing-

kibana-spaces-for-organization-and-security

Guimarães, A., Lins, R., & Oliveira, R. (2006). *Segurança em Redes Privadas Virtuais-VPNs*. Brasport.

Gupta, Y., & Gupta, R. (2017). *Mastering Elastic Stack*. Packt Publishing. Retrieved from https://books.google.pt/books?id=EVQoDwAAQBAJ

Hamilton, J., Gonzalez Berges, M., Tournier, J.-C., & Schofield, B. (2018). SCADA Statistics monitoring using the elastic stack (Elasticsearch, Logstash, Kibana). In *16th Int. Conf. on Accelerator and Large Experimental Control Systems* (pp. 451–455). Barcelona: JACoW Publishing. Retrieved from https://pdfs.semanticscholar.org/5db9/ca3742ac99dcd6d498b5a3553fb716d99ce1.pdf

Hershkovitch, D. (2019). Elastic Uptime Monitoring solution released | Elastic Blog. Retrieved April 2, 2019, from https://www.elastic.co/pt/blog/elastic-uptime-monitoring-solution-released

Higbee, M. (2015). *Deriving System Vulnerabilities Using Log Analytics*. Brigham Young University.

Imfsecurity. (2020). LOG-MD Free Edition. Retrieved from https://www.imfsecurity.com/free

Irace, M. B. M. (2018). Big Data: Analisando Dados com O SPLUNK Entreprise. Universidade Federal do Rio de Janeiro.

ISO/IEC 27002. (2005). ABNT NBR ISO/IEC 27002. INTERNATIONAL STANDARD.

ISO/IEC 27005. (2011). Information technology — Security techniques — Information security risk management. INTERNATIONAL STANDARD.

Jaeger, D., Cheng, F., & Meinel, C. (2018). Accelerating Event Processing for Security Analytics on a Distributed In-Memory Platform. In *2018 IEEE 16th Intl Conf on Dependable, Autonomic and Secure Computing, 16th Intl Conf on Pervasive Intelligence and Computing, 4th Intl Conf on Big Data Intelligence and Computing and Cyber Science and Technology Congress (DASC/PiCom/DataCom/CyberSciTech* (pp. 634–643). IEEE.

Jain, U. (2018). *Lateral movement detection using ELK stack*. University of Houston.

Johnson, L. (2015). *Security Controls Evaluation, Testing, and Assessment Handbook*. Syngress. Retrieved from https://books.google.pt/books?id=X7SYBAAAQBAJ

Joshi, R., & Gupta, B. (2019). *Security, Privacy, and Forensics Issues in Big Data*. IGI Global. Retrieved from https://books.google.pt/books?id=zK2rDwAAQBAJ

Kali. (2020). Hydra Package Description. Retrieved November 17, 2020, from https://tools.kali.org/password-attacks/hydra

Kcoe. (2019). Log retention in ossim | AT&T Cybersecurity. Retrieved July 27, 2019, from https://success.alienvault.com/s/question/0D50Z00008oGrCjSAK/log-retention-in-ossim

Kearnsh, S. (2019). Security for Elasticsearch is now free | Elastic Blog. Retrieved June 3, 2019, from https://www.elastic.co/pt/blog/security-for-elasticsearch-is-now-free

Keijser, J. (2017). *OpenVPN Cookbook*. Packt Publishing. Retrieved from https://books.google.pt/books?id=_1MoDwAAQBAJ

Kent, K., & Souppaya, M. (2006). Guide to computer security log management. *NIST Special Publication*, *92*.

Kevin Orrey. (2014). Penetration Test Framework. Retrieved March 17, 2019, from http://www.vulnerabilityassessment.co.uk/index.htm

Kostrecová, E., & Bınová, H. (2015). Research paper security information and event management. *Management*, *4*(2).

Kotenko, I., Fedorchenko, A., Saenko, I., & Kushnerevich, A. (2018). Parallelization of security event correlation based on accounting of event type links. In *2018 26th Euromicro International Conference on Parallel, Distributed and Network-based Processing (PDP)* (pp. 462–469). IEEE.

Kumar, S. (2019). Setting up RBAC in Elasticsearch with Kibana. Retrieved June 3, 2020, from https://medium.com/@Sushil_Kumar/setting-up-rbac-in-elasticsearch-with-kibana-195f07cd74a9

Lallan, E. (2015). *Ethical Hacking and Computer Securities for Beginners*. Lulu.com.

Retrieved from https://books.google.pt/books?id=LXJaCAAAQBAJ

Leeon123. (2018). Simple-SYN-Flood. Retrieved October 25, 2020, from https://github.com/Leeon123/Simple-SYN-Flood

Li, B., Shu, L., & Zeng, D. (2018). *Communications and Networking: 12th International Conference, ChinaCom 2017, Xi'an, China, October 10-12, 2017, Proceedings*. Springer International Publishing. Retrieved from https://books.google.pt/books?id=n1FTDwAAQBAJ

Li, Q., & Clark, G. (2015). *Security Intelligence: A Practitioner's Guide to Solving Enterprise Security Challenges*. Wiley. Retrieved from https://books.google.pt/books?id=3iGsBwAAQBAJ

Lima, F. (2014). *Formato de representação de eventos de segurança de informação*. Universidade do Minho.

Lmangani. (2020). Sentinl. Retrieved October 24, 2020, from https://github.com/lmangani/sentinl/releases

LogRhythm. (2017). Quadrante Mágico de SIEM Gartner de 2017 | LogRhythm. Retrieved April 29, 2018, from https://pt.logrhythm.com/2017-gartner-magic-quadrant-siem-report-a/

Logsingn. (2020). SIEM Key Features - Strong Cyber Security Defence. Retrieved May 30, 2020, from https://www.logsign.com/siem/siem-key-features

Lopez, R. (2018). 3 Splunk Best Practices I Learned the Hard Way - Aditum Partners. Retrieved August 7, 2019, from https://www.aditumpartners.com/3-splunk-best-practices-i-learned-the-hard-way/

Lourenço, A. (2018). *Operational Intelligence with GDPR*. Universidade de Lisboa.

Lowe, R. (2017). *Computação Segura é Como Sexo Seguro: Você tem que praticar para evitar infecções*. (Babelcube Inc., Ed.). Writing King. Retrieved from https://books.google.pt/books?id=TpmzDgAAQBAJ

Lubeck, L. (2019). Como usar MITRE ATT&CK: uma lista de técnicas e procedimentos de ataques e defesas. Retrieved February 27, 2020, from

https://www.welivesecurity.com/br/2019/06/07/como-usar-mitre-attck-uma-lista-de-tecnicas-e-procedimentos-de-ataques-e-defesas/

Magalhães, F., & Pereira, M. (2018). *Regulamento Geral de Proteção de Dados: Manual Prático 2ª Edição Revista e Ampliada*. (Vida Economica Editorial, Ed.).

Malik, J. (2017). The General Data Protection Regulation (GDPR) | AT&T Cybersecurity. Retrieved July 30, 2019, from https://www.alienvault.com/blogs/security-essentials/the-general-data-protection-regulation-gdpr

Malwarebytes. (2020). *2020 State of Malware Report*. Retrieved from https://resources.malwarebytes.com/files/2020/02/2020_State-of-Malware-Report.pdf

Marlette, T. (2016). *Splunk Best Practices*. Packt Publishing. Retrieved from https://tinyurl.com/y6wsfcqc

Marques, P. (2018). *Assessment on the effectiveness of design diversity for network security and monitoring*. Universidade de Lisboa. Retrieved from http://hdl.handle.net/10451/34890

Marquina, L. (2018). *Ventajas e Implementación de un sistema SIEM*. Universitat Oberta de Catalunya.

Martin, L. (2020). No the Cyber Kill Chain. Retrieved February 27, 2020, from https://www.lockheedmartin.com/en-us/capabilities/cyber/cyber-kill-chain.html

Martins, A. (2019). Utilizando o Sysmon para monitorar atividades do Windows. Retrieved October 26, 2020, from https://www.linkedin.com/pulse/utilizando-o-sysmon-para-monitorar-atividades-do-windows-martins

Meftah, B. (2019). AT&T Cybersecurity Is Born. Retrieved July 26, 2019, from https://www.alienvault.com/blogs/security-essentials/att-cybersecurity-is-born

Meyer, S. (2001). Event Logs: Defining Their Purpose in Today's Network Security Environment. SANS Institute. Retrieved from https://tinyurl.com/ycofxmrv

Microsoft. (2020). RGPD simplificado: um guia para sua pequena empresa. Retrieved June 13, 2020, from https://docs.microsoft.com/pt-br/microsoft-365/admin/security-and-

compliance/gdpr-compliance?view=o365-worldwide

MITRE. (2020a). Frequently Asked Questions. Retrieved March 10, 2020, from https://attack.mitre.org/resources/faq/

MITRE. (2020b). MITRE ATT&CK. Retrieved February 27, 2020, from https://attack.mitre.org/

MITRE ATT&CK. (2020). Pupy. Retrieved November 17, 2020, from https://attack.mitre.org/software/S0192/

Mokalled, H., Catelli, R., Casola, V., Debertol, D., Meda, E., & Zunino, R. (2019). The Applicability of a SIEM Solution: Requirements and Evaluation. In *2019 IEEE 28th International Conference on Enabling Technologies: Infrastructure for Collaborative Enterprises (WETICE)* (pp. 132–137). IEEE.

Monnappa, A. (2018). *Learning Malware Analysis: Explore the concepts, tools, and techniques to analyze and investigate Windows malwareMonnappa*. Packt Publishing. Retrieved from https://books.google.pt/books?id=QsNiDwAAQBAJ

Moreno, D. (2015). *Introdução ao Pentest*. Novatec Editora. Retrieved from https://books.google.pt/books?id=M4fDCAAAQBAJ

Mourby, M., Mackey, E., Elliot, M., Gowans, H., Wallace, S. E., Bell, J., … Kaye, J. (2018). Are 'pseudonymised'data always personal data? Implications of the GDPR for administrative data research in the UK. *Computer Law & Security Review*, *34*(2), 222–233.

Ngo-Lam. (2020). Exabeam Named a Leader in the 2020 Gartner Magic Quadrant for SIEM for 2nd Consecutive Year. Retrieved August 20, 2020, from https://www.exabeam.com/siem/exabeam-named-leader-2020-gartner-magic-quadrant-for-siem/

Norrby, E. (2018). *Investigation and Implementation of a Log Management and Analysis Framework for the Treatment Planning System RayStation*. Uppsala Universitet.

O'Leary, M. (2015). *Cyber Operations: Building, Defending, and Attacking Modern Computer Networks*. Apress. Retrieved from https://books.google.pt/books?id=ZfrNCgAAQBAJ

O'Leary, M. (2019). *Cyber Operations: Building, Defending, and Attacking Modern Computer Networks*. Apress. Retrieved from https://books.google.pt/books?id=dBKLDwAAQBAJ

Orvalho, J. (2015). Desenvolvimento de aplicações para dispositivos móveis: tipos e exemplo de aplicação na plataforma IOS. *II Workshop de Iniciação Científica Em Sistemas de Informação*.

OWASP. (2018). OWASP Security Knowledge Framework - OWASP. Retrieved March 17, 2019, from https://www.owasp.org/index.php/OWASP_Security_Knowledge_Framework#tab=Documentation

OWTF. (2019). Offensive Web Testing Framework. Retrieved from https://github.com/owtf/owtf

Ozkaya, E. (2018). *Learn Social Engineering: Learn the art of human hacking with an internationally renowned expert*. Packt Publishing. Retrieved from https://books.google.pt/books?id=e_RZDwAAQBAJ

Paquette, M. (2018). An Introduction to GDPR and Elasticsearch, the Elastic Stack | Elastic Blog. Retrieved June 3, 2019, from https://www.elastic.co/pt/blog/introduction-to-gdpr-with-elasticsearch

Paquette, M. (2019). Introducing Elastic SIEM | Elastic Blog. Retrieved July 24, 2019, from https://www.elastic.co/pt/blog/introducing-elastic-siem

Paro, A. (2019). *Elasticsearch 7.0 Cookbook: Over 100 recipes for fast, scalable, and reliable search for your enterprise, 4th Edition*. Packt Publishing. Retrieved from https://books.google.pt/books?id=qzCWDwAAQBAJ

Peixinho, I., Fonseca, F., & Lima, F. (2013). *Segurança de Redes e Sistemas*. Escola Superior de Redes.

Pentest-Standard. (2014). The Penetration Testing Execution Standard. Retrieved March 17, 2019, from http://www.pentest-standard.org/index.php/Main_Page

Pete Herzog. (2017). ISECOM - Open Source Security Testing Methodology Manual (OSSTMM). Retrieved March 17, 2019, from

http://www.isecom.org/research/osstmm.html

Petters, J. (2020). What is SIEM? A Beginner's Guide. Retrieved October 7, 2020, from https://www.varonis.com/blog/what-is-siem/

Pfleeger, C., Pfleeger, S., & Margulies, J. (2015). *Security in Computing*. (Prentice Hall, Ed.) (5th ed.). Pearson Education. Retrieved from https://books.google.pt/books?id=VjMqBgAAQBAJ

Pinho, F. (2017). *Anonimização de bases de dados empresariais de acordo com a nova Regulamentação Europeia de Proteção de Dados*. Universidade do Porto.

ReadonlyREST. (2018a). Compare all the features. Retrieved August 20, 2020, from https://readonlyrest.com/#compare

ReadonlyREST. (2018b). Security for Elasticsearch and Kibana. Retrieved November 5, 2019, from https://readonlyrest.com/#compare

Ricardo, C.-C. (2018). HS. Register–An Audit-Trail Tool to Respond to the General Data Protection Regulation (GDPR). *Building Continents of Knowledge in Oceans of Data: The Future of Co-Created EHealth*, *247*, 81.

Roberts, S., & Brown, R. (2017). *Intelligence-Driven Incident Response: Outwitting the Adversary*. O'Reilly Media. Retrieved from https://books.google.pt/books?id=wfwxDwAAQBAJ

RockNSM. (2019). ROCK NSM. Retrieved December 24, 2019, from http://rocknsm.io/

Rodrigues, B. (2015). *Open-source intelligence em sistemas SIEM*. Universidade de Lisboa.

Rodrigues, P. (2013). *Resilient event collection in SIEM systems*. Universidade de Lisboa.

Rodriguez, R. (2019). The Hunting ELK. Retrieved February 21, 2019, from https://github.com/Cyb3rWard0g/HELK

Rouse, M. (2012). What is Computer Security Incident Response Team (CSIRT) ? - Definition from WhatIs.com. Retrieved April 24, 2018, from https://whatis.techtarget.com/definition/Computer-Security-Incident-Response-Team-CSIRT

Russinovich, M., & Margosis, A. (2016). *Troubleshooting with the Windows Sysinternals Tools*. Pearson Education. Retrieved from https://books.google.pt/books?id=YQ86DQAAQBAJ

Saldanha, N. (2018). *Novo Regulamento Geral de Proteção de Dados* (FCA-Edito).

SANS. (2020). Cybersecurity Courses & Certifications. Retrieved October 7, 2020, from https://www.sans.org/cyber-security-courses/

Santos, P., Bessa, R., & Pimentel, C. (2008). *Cyberwar - O Fenómeno, as Tecnologias e os Actores*. FCA - Editora de Informatica, lda.

Saxena, S., & Gupta, S. (2017). *Practical Real-time Data Processing and Analytics: Distributed Computing and Event Processing using Apache Spark, Flink, Storm, and Kafka*. Packt Publishing. Retrieved from https://books.google.pt/books?id=9JlGDwAAQBAJ

Schifreen, R. (2015). Explore Sysmon and the Windows Event Viewer. Retrieved from https://www.techsupportalert.com/content/explore-sysmon-and-windows-event-viewer.htm

Search Guard. (2018). Product | Search Guard product overview for securing Elasticsearch cluster. Retrieved May 20, 2019, from https://search-guard.com/product/#feature-comparison

Sebastian Herzberg. (2015). How the Elastic Stack Keeps our Taxis Rolling. Retrieved August 20, 2020, from https://www.elastic.co/pt/blog/how-the-elastic-stack-keeps-our-taxis-rolling

Senanayaka, L. (2018). What is Elastic stack which everyone is talking about??? Retrieved April 26, 2019, from https://medium.com/@lakinisenanayaka/what-is-elastic-stack-which-everyone-is-talking-about-6c5d94d83983

Settle, M., Paquette, M., Goldstein, A., & Kroh, A. (2019). *Hands on with Elastic SIEM*. Elasticsearch. Retrieved from shorturl.at/BDLV9

Sharma, V. (2016). *Beginning Elastic Stack.* Apress. Retrieved from https://books.google.pt/books?id=vIOxDQAAQBAJ

Shukla, P., & Kumar, S. (2019). *Learning Elastic Stack 7.0 - Second Edition*. Packt Publishing Ltd.

Shukla, P., Kumar, S., Chhajed, S., & Ochoa, M. (2017). *Learning Elastic Stack 6.0: A beginner's guide to distributed search, analytics, and visualization using Elasticsearch, Logstash and Kibana*. Packt Publishing. Retrieved from https://books.google.pt/books?id=UvNFDwAAQBAJ

Sigman, B., Delgado, E., Diakun, J., Johnson, P., Mock, D., & Yadav, A. (2017). *Splunk: Enterprise Operational Intelligence Delivered*. Packt Publishing. Retrieved from https://tinyurl.com/y8aa9ytw

Sikorski, M., & Honig, A. (2012). *Practical Malware Analysis: A Hands-On Guide to Dissecting Malicious Software*. No Starch Press. Retrieved from https://goo.gl/qHgwUJ

Simko, S. (2018). *Implementation of Systems for Intrusion Detection and Log Managemento*. Masaryk University.

Siren. (2018). SENTINL 6. Retrieved May 20, 2019, from https://github.com/sirensolutions/sentinl

Soni, R. (2016). *Nginx: From Beginner to Pro*. Apress. Retrieved from https://books.google.pt/books?id=uQvpDAAAQBAJ

Sousa, H. (2019). *Sistema de gestão de eventos de segurança de informação em alta disponibilidade*. Instituto Politécnico do Porto. Retrieved from http://hdl.handle.net/10400.22/15500

Sousa, L. (2016). *Sonorização de eventos gerados por um SIEM*. Politécnico do Porto. Retrieved from https://tinyurl.com/ydxeqtls

Splunk. (2015). Splunk® Enterprise - The Platform for Operational Intelligence. Splunk.

Splunk. (2016a). Deploy:SplunkBucketRetentionTimestampsAndYou Splunk Bucket Retention, Timestamps, and You. Retrieved July 15, 2020, from https://wiki.splunk.com/Deploy:SplunkBucketRetentionTimestampsAndYou

Splunk. (2016b). Does a forwarder need to connect through an indexer? - Question |

Splunk Answers. Retrieved April 18, 2019, from https://answers.splunk.com/answers/483577/does-a-forwarder-need-to-connect-through-an-indexe.html

Splunk. (2016c). The Splunk Guide to Operational Intelligence. Retrieved from https://www.splunk.com/pdfs/solution-guides/splunk-guide-to-operational-intelligence.pdf

Splunk. (2017). SPLUNK® For Big Data Analytics. Splunk. Retrieved from https://www.splunk.com/pdfs/solution-guides/splunk-for-big-data.pdf

Splunk. (2018a). Produtos Splunk | Splunk. Retrieved May 1, 2018, from https://www.splunk.com/pt_br/products.html

Splunk. (2018b). Recursos do Splunk Enterprise e Splunk Cloud. Retrieved April 16, 2019, from https://www.splunk.com/pt_br/products/splunk-enterprise/features.html

Splunk. (2019a). Free vs. Enterprise | Tabela de comparação da licença do Splunk | Splunk. Retrieved February 9, 2019, from https://goo.gl/yt5vMD

Splunk. (2019b). Pricing for Splunk Enterprise and Splunk Cloud. Retrieved March 11, 2019, from https://www.splunk.com/en_us/software/pricing.html

Splunk. (2019c). *Splunk GDPR Implementation Success*. Retrieved from www.splunk.com

Splunk. (2019d). Splunk Platform Comparison. Retrieved April 17, 2019, from https://www.splunk.com/en_us/software/features-comparison-chart.html

Splunk. (2020a). Free vs. Enterprise. Retrieved July 15, 2020, from https://www.splunk.com/pt_br/view/SP-CAAAE8W

Splunk. (2020b). Reference hardware. Retrieved August 25, 2020, from https://docs.splunk.com/Documentation/Splunk/7.3.1/Capacity/Referencehardware

Splunk. (2020c). Types of Splunk software licenses. Retrieved September 15, 2020, from https://docs.splunk.com/Documentation/Splunk/8.0.6/Admin/TypesofSplunklicenses

Srivastava, A. (2019). *Kibana 7 Quick Start Guide: Visualize your Elasticsearch data with ease*. Packt Publishing. Retrieved from https://books.google.pt/books?id=ZGSGDwAAQBAJ

Strager, J. (2020). Think Like a Hacker: 3 Cybersecurity Models Used to Investigate Intrusions. Retrieved October 26, 2020, from https://www.comptia.org/blog/think-like-a-hacker-3-cybersecurity-models-used-to-investigate-intrusions

Tal, L. (2018). Log Management Comparison: ELK vs Graylog. Retrieved July 6, 2019, from https://coralogix.com/log-analytics-blog/log-management-comparison-elk-vs-graylog/

Tavares, L. (2015). *Análise de eventos de segurança: baseado no OSSIM*. Universidade do Minho.

Teiss. (2020). Why MITRE ATT&CK™ is the cybersecurity framework of 2020. Retrieved May 15, 2020, from https://www.teiss.co.uk/why-mitre-attck-is-the-cybersecurity-framework-of-2020/

Tevault, D. (2020). *Mastering Linux Security and Hardening: Protect your Linux systems from intruders, malware attacks, and other cyber threats, 2nd Edition*. Packt Publishing. Retrieved from https://books.google.pt/books?id=tpbSDwAAQBAJ

Tidrow, R., Boyce, J., & Shapiro, J. (2015). *Windows 10 Bible*. Wiley. Retrieved from https://books.google.pt/books?id=ihOACgAAQBAJ

Trustradius. (2019). AlienVault OSSIM vs Elasticsearch | TrustRadius. Retrieved July 30, 2019, from https://www.trustradius.com/compare-products/alienvault-ossim-vs-elasticsearch

União Europeia. (2016). REGULAMENTO (UE) 2016/679 DO PARLAMENTO EUROPEU E DO CONSELHO. *Jornal Oficial Da União Europeia*. Retrieved from https://eur-lex.europa.eu/legal-content/PT/TXT/HTML/?uri=CELEX:32016R0679&qid=1606851672454&from=EN

UnifiedThreatWorks. (2020). AlienVault IT Compliance Management. Retrieved October 24, 2020, from https://www.unifiedthreatworks.com/Compliance.asp

Vacca, J. (2012). *Computer and Information Security Handbook* (2nd ed.). CRC Press.

Vainio, A. (2018). *Implementation of Centralized Logging and Log Analysis in Cloud Transition*. Aalto University.

Varanda, A. (2019). *O Regulamento Geral de Proteção de Dados e a Pseudonimização de Logs*. Instituto Politécnico de Leiria. Retrieved from http://hdl.handle.net/10400.8/4362

Vazão, A., Santos, L., Piedade, M. B., & Rabadão, C. (2019). SIEM Open Source Solutions: A Comparative Study. In *2019 14th Iberian Conference on Information Systems and Technologies (CISTI)* (pp. 1–5). IEEE.

Velu, V. (2017). *Mastering Kali Linux for Advanced Penetration Testing*. Packt Publishing. Retrieved from https://books.google.pt/books?id=JHg5DwAAQBAJ

Ventrella, C. (2018). *Information, Data Science for Source, Security with Open Technologies*. Politecnico Di Torino.

Walther, T. (2018). Digital transformation of the global cement industry. In *Cement Industry Conference (IAS/PCA), 2018 IEEE-IAS/PCA* (pp. 1–8). IEEE.

Watts, S. (2018). IEM vs Log Management: What's the difference? Retrieved March 10, 2020, from https://www.bmc.com/blogs/siem-vs-log-management-whats-the-difference/

Web3us. (2018). Security Information and Event Management (SIEM) | Web3us LLC. Retrieved September 10, 2019, from https://www.web3us.com/cyber-security/security-information-and-event

Whitman, M., & Mattord, H. (2017). *Principles of Information Security*. Cengage Learning. Retrieved from https://books.google.pt/books?id=59dUDgAAQBAJ

Wiltshire. (2019). Elastic Stack 7.0 Now Available - MarketWatch. Retrieved May 18, 2019, from https://www.marketwatch.com/press-release/elastic-stack-70-now-available-2019-04-10

Wintergerst, L., Paquette, M., & McDiarmid, D. (2018). Protecting GDPR Personal Data with Pseudonymization | Elastic Blog. Retrieved August 16, 2019, from https://www.elastic.co/pt/blog/gdpr-personal-data-pseudonymization-part-1

Yelp.com. (2019). ElastAlert. Retrieved May 20, 2019, from https://github.com/Yelp/elastalert

Yigal, A. (2016). Kafka vs. Redis: Log Aggregation Capabilities and Performance | Logz.io. Retrieved June 5, 2019, from https://logz.io/blog/kafka-vs-redis/

Zarzosa, S. (2017). In-depth analysis of SIEMs extensibility. DiSIEM.

Zeinali, S. (2016). *Analysis of security information and event management (siem) evasion and detection methods*. Tallinn University of Technology.

Zerlang, J. (2017). GDPR: a milestone in convergence for cyber-security and compliance. *Network Security*, *2017*(6), 8–11.

Zuech, R., Khoshgoftaar, T., & Wald, R. (2015). Intrusion detection and big heterogeneous data: a survey. *Journal of Big Data*, *2*(1), 3.

Zúquete, A. (2018). *Segurança em Redes Informaticas*. Lisboa: FCA - Editora de Informatica, lda.

Anexo A – Exemplos de *Malware*

Para o presente trabalho achou-se pertinente aprofundar alguns tipos de *malware* tais como a negação de serviço (*Denial of servisse - DoS*), Código Malicioso ou *Malware* e a Engenharia social.

A negação de serviço é uma tentativa de tornar um recurso ou serviço indisponível, ou seja, no decorrer de um ataque *DoS* vão ser enviados pacotes de dados para o alvo até este bloqueie ou esgote os seus recursos (Lallan, 2015; Santos et al., 2008). São exemplos de negação de serviço: o *Flooding*, o *Broadcast Storm* e o *E-mail bombing* (Santos et al., 2008).

O ataque *SYN Flood* é consiste numa tentativa de negação de serviço (*Denial of servisse - DoS*). Para fazer este tipo de ataque o atacante envia um pacote SYN (Synchronization), mas não responde ao ACK (Acknowledge) enviado pela vítima, o que implica que vai gerar um timer na *spool* TCP/IP com determinada validade, ficando à espera de receber uma resposta do atacante. Se o número de pacotes gerado pelo atacante ou atacantes for em grande número, se possuírem uma origem inválida e um timer válido, a máquina da vítima entra em colapso (Santos et al., 2008).

Um ataque *Flooding* acontece quando a "*máquina-alvo*" é sobrecarregada com pedidos para o estabelecimento de uma ligação, o que vai obrigar a máquina a reservar recursos e a ocupar o buffer à espera de ligações (Santos et al., 2008). O buffer é uma parte da memória de um equipamento que é alocada para conter dados (carateres ou matriz), na eventualidade de um buffer receber mais dados do que aqueles que pode manipular, ocorre um *buffer overflow*. Quando ocorre este tipo de ataque/erro, os dados nocivos vão ocupar o espaço de memória adjacente provocando uma falha no sistema (Lallan, 2015).

Quando são enviados para a rede um enorme número de pacotes redundantes e estes circulam por várias máquinas, conseguem esgotar os recursos da rede prejudicando, dessa forma, a sua utilização, este tipo de ataque é denominado *Broadcast Storm* (B. Li, Shu, & Zeng, 2018; Santos et al., 2008). Podemos explicar o *E-mail bombing* como um ato intencional de envio de várias cópias de conteúdo idêntico para um mesmo destinatário de e-mail. O principal objetivo desta ação é o de sobrecarregar o servidor de e-mail, perturbando e degradando o seu serviço (EC-Council, 2016).

Malware é um bloco de código que executa ações maliciosas e que pode atuar através de um executável, de um script, de código ou de qualquer outro software. Geralmente, penetra no sistema da Organização através de USB, de E-Mail ou de uma ligação à Internet e é utilizado com propósito de roubar informação confidencial, de perturbar o funcionamento da máquina, de provocar ataques de negação, de enviar e-mails de spam, de encriptar os ficheiros da vítima para depois pedir um resgate (*Ransomware*), de espionar ou de assumir o controlo da máquina infetada (Monnappa, 2018). O termo *Malware* é abrangente e é utilizado para descrever vários tipos de *malware* tais como (Monnappa, 2018): Vírus, *Worms*, *Trojans*, *Backdoors / Remote Access Trojans (RAT)*, *Adwares, Botnets, Information stealers, Ransomwares, Rootkits* e *Downloaders or droppers*.

Os vírus são programas com capacidade de autorreplicação, ou seja, no momento em que são executados pelo utilizador conseguem duplicar-se e infetar ficheiros e aplicações. Os *worms* não necessitam de ser executados pelo utilizador para se replicarem, estes possuem a capacidade de se duplicarem e de se propagarem pela rede explorando as vulnerabilidades dos sistemas (Escola Superior de Redes, 2015).

O *trojan* é um *malware* que tenta enganar o utilizador fornecendo as funcionalidades que o mesmo deseja, no entanto, para além dessas funcionalidades, o *trojan* contém funções maliciosas que são executadas quando o utilizador utiliza o software (Escola Superior de Redes, 2015). Um *backdoor* é um tipo de *trojan* que permite que o atacante aceda à máquina da vítima e que possa executar comados que vão comprometer o sistema informático (Monnappa, 2018; Sikorski & Honig, 2012). Já os *rootkits* permitem o acesso privilegiado ao atacante e, por outro lado, ocultam a sua presença, para que seja difícil a sua deteção (Monnappa, 2018).

Os *bootnets* são um conjunto de máquinas infetadas com o mesmo *malware* (*bots*) e que recebem instruções de um servidor que é controlado pelo atacante (Monnappa, 2018; Sikorski & Honig, 2012). Também existe um tipo de *malware* cujo objetivo é o de roubar dados confidenciais, como é caso das credenciais bancárias ou das teclas digitadas, sendo que este tipo de código malicioso é denominado de "*information stealers*" (Monnappa, 2018).

São designados de *adware* todos os anúncios indesejados que um computador disponibiliza, sendo que estes podem ser instalados na máquina através dos *downloads*

gratuitos. Finalmente, os *downloaders* ou *droppers* têm como propósito o de fazer o *download* ou de instalar componentes adicionais de *malware* (Monnappa, 2018).

É designada de engenharia social a tentativa de um atacante induzir um utilizador a fazer algo a seu pedido, ou seja, o atacante assume uma identidade falsa para conseguir passar ou para quebrar os procedimentos de segurança (Lowe, 2017).

Ao contrário dos sistemas informáticos, os utilizadores não podem ser protegidos por ferramentas como *firewalls* ou antivírus, pelo que, numa hora, um especialista de engenharia social pode recolher um volume de informação que, através de outro tipo de ataque, demoraria 100 horas a concretizar. Quando um sistema possui vários níveis de segurança por vezes é mais fácil "*hackear*" os seus utilizadores (Ozkaya, 2018).

As redes sociais como Facebook, Twitter, Instagram e Snapchat potenciaram os ataques de engenharia social, pois a maioria dos utilizadores publicam detalhes das suas vidas pessoais, tais como: o local de trabalho, a família ou as preferências sociais. Um atacante necessita apenas de passar pelas redes sociais para que possa assumir a identidade da maioria dos utilizadores (Ozkaya, 2018).

Muitos dos utilizadores dos sistemas informáticos são simpáticos, confiantes e estão dispostos a ajudar estranhos, o que pode levar a que revelem informações confidenciais. A imagem seguinte baseada no relatório da empresa Verizon's e ilustra o impacto que a engenharia social pode ter sobre uma Organização (Ozkaya, 2018).

Diagrama I - Exemplo ilustrativo do impacto de um ataque de engenharia social (Ozkaya, 2018, p. 11)

Para mitigar e prevenir ataques de engenharia social é importante saber identificá-los e também criar uma cultura de consciencialização de segurança da empresa (Ozkaya, 2018), sendo que a formação sobre este tema pode ser uma das medidas.

Anexo B – Formato dos *logs*

De seguida descrevem-se alguns dos formatos mais comuns de *logs*, começamos pelos logs do Sistema operativo Windows.

- ***Logs* dos Sistemas Operativos Microsoft Windows**

A Microsoft implementou o seu próprio mecanismo de *logs* para os sistemas operativos Microsoft Windows, denominado de *Windows Event Log* que é um dos exemplos mais comuns comuns do formato binário (Chuvakin et al., 2012). O *Event log* é utilizado para recolher dois tipos de *logs* (Chuvakin et al., 2012): os do Microsoft Windows e os dos serviços e aplicações. No sistema operativo Microsoft Windows são atribuídos IDs (códigos de identificação) aos *logs* tendo em conta as suas caraterísticas, por exemplo o ID 4625 identifica uma tentativa de acesso ao sistema (login) a uma conta bloqueada ou desativada (Schifreen, 2015).

Para que possam ser visualizados os eventos do sistema operativo Microsoft Windows, este disponibiliza a ferramenta *Event Viewer*. Esta ferramenta permite visualizar e pesquisar os *logs* do sistema operativo Microsoft Windows, possibilitando que possam ser rastreados possíveis problemas. Todavia, existem operações especificas que o *Event Viewer* não registra, não guarda os *hashes* dos processos ou vincula uma ligação de rede a um processo e, por essa razão, foram desenvolvidas outras ferramentas complementares, como é exemplo o Sysmon (Joshi & Gupta, 2019; Martins, 2019; Schifreen, 2015).

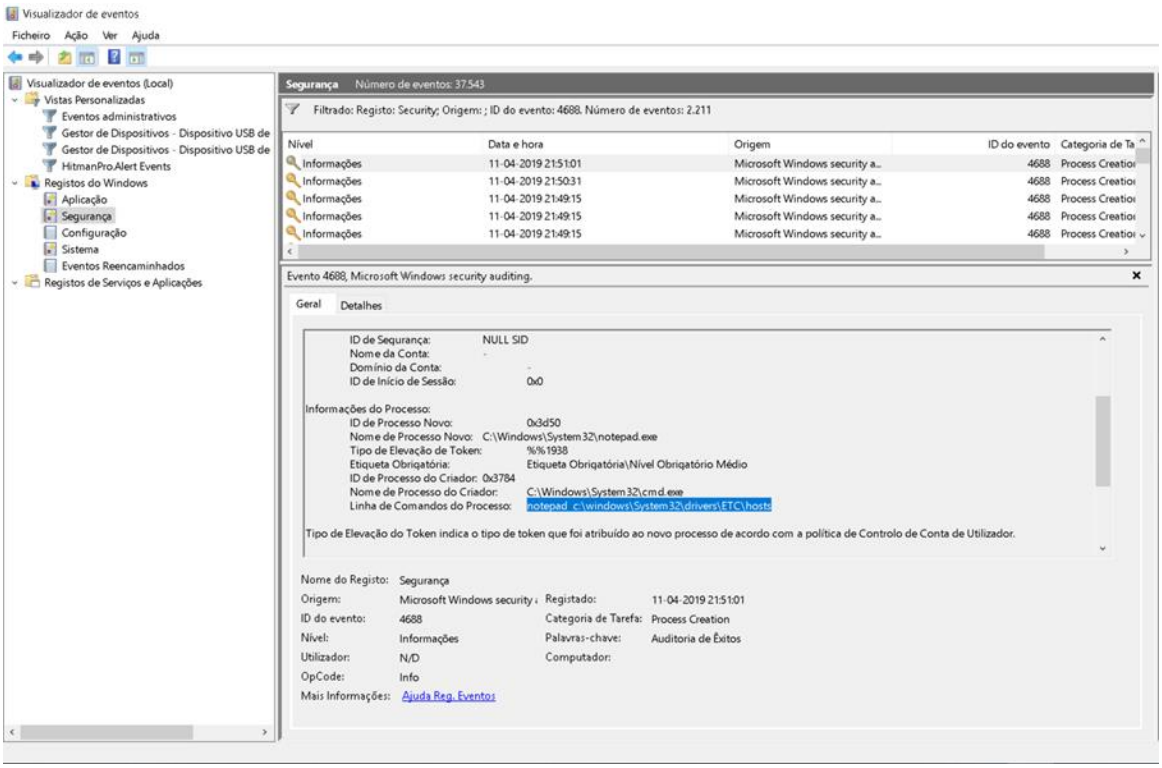


Diagrama II - Event Viewer

O Sysmon, que atualmente pertence à Microsoft, é uma ferramenta sofisticada, que foi desenhada para os sistemas operativos Microsoft Windows e que, na sua generalidade, os *logs* que gera possuem mais detalhe do que os criados pelo *Windows Event Log* (Joshi & Gupta, 2019). O Sysmon é instalado como um driver, sendo executado como um serviço do Microsoft Windows e que permite registar as atividades e os eventos do sistema (Joshi & Gupta, 2019). Esta ferramenta regista uma série de eventos que se sabe serem causados por hackers, vírus e *malware* (Russinovich & Margosis, 2016; Schifreen, 2015).

A configuração base do Sysmon está definida com o propósito de recolher todos os *logs* do sistema operativo Microsoft Windows, todavia é possível configurar quais os eventos que vão ser registados. O SwiftOnSecurity criou um ficheiro de configuração que permite rastrear comportamentos suspeitos, como por exemplo uma ligação de rede que é executada *c: \ Windows \ Temp* (O'Leary, 2019).

- ***Logs* dos Sistemas Operativos Linux**

A maioria dos sistemas operativos Linux utiliza o *Syslog*, contudo os sistemas operativos das distribuições CentOS, Mint e Ubuntu utilizam o *syslog daemon*. O OpenSuSE na versão 13.1 e versões inferiores também utiliza o *syslog daemon*, mas a partir da versão 13.2 não é um dos serviços que é instalado automaticamente (O'Leary, 2019).

O *Syslog* é um ficheiro de texto legível em ASCII e que é utilizado para armazenar os *logs* do *kernel* e de várias aplicações na pasta */var/logs* (Varanda, 2019). Quando o *Syslog* foi desenvolvido, é de referir que a segurança não era considerada um fator primordial e, por isso, na generalidade dos casos o *Syslog* utiliza o protocolo UDP - User Datagram Protocol (Kent & Souppaya, 2006).

Listamos, de seguida, três dos *daemons* mais comuns no Linux (O'Leary, 2015, 2019): o *syslogd*, o *syslog-ng*, e o *rsyslog*. A configuração do *syslogd* e do *rsyslog* partilham vários elementos (a seção de regras, a forma de definir o tipo de mensagens que são registadas e onde são guardadas é igual), já o *syslog-ng* possui uma abordagem diferente, porém todos permitem o envio dos *logs* para um destino remoto (O'Leary, 2015). O *rsyslog* e o *syslog-ng* permitem a utilização do protocolo TCP (Transmission Control Protocol), mas o *syslogd* utiliza o protocolo UDP. Finalmente, os formatos *syslog-ng* e *ryslog* oferecem outras funcionalidades, como é o caso dos filtros e do envio dos *logs* para uma base de dados (Chapple & Seidl, 2017; Vacca, 2012).

Os sistemas operativos Linux mais antigos utilizam o *SysVInit* ou o *Upstart* para gerir serviços e scripts, atualmente na maioria das distribuições foram substituídos pelo *systemd*. Na sua generalidade as distribuições que utilizam o *systemd* também utilizam o *systemd-journald* para armazenar os *logs* no sistema. Várias distribuições utilizam o *systemd-journald*, nomeadamente, CentOS 7.0, Mint 18, OpenSuSE 12.3 e o Ubuntu 15.04 e posteriores (O'Leary, 2019). A imagem seguinte ilustra um visualizador de *logs* no Linux.

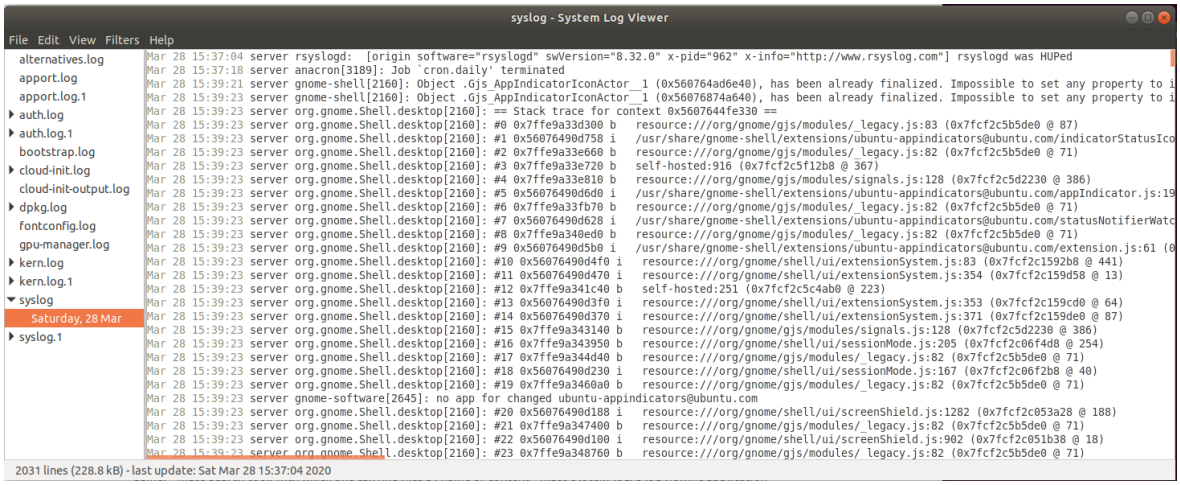


Diagrama III - System Log Viewer

O *systemd-journald* utiliza os mesmos valores para a classificação de mensagens que o *syslogs* e o *rsyslog*, porém, recorre um ficheiro binário para arquivar os logs, sendo necessário ferramentas especializadas para os interpretar (O'Leary, 2019).

- ***Logs* de servidores web**

Os *logs* criados pelos servidores *Web* normalmente contêm informação sobre quem visitou a página, a sua origem e a atividade realizada no servidor (Varanda, 2019).

A codificação utilizada pelo servidor *web* IIS pode ser em UTF-8 ou ANSI (sistemas operativos mais antigos), sendo de referir que a partir de 2018 os *logs* são codificados em UTF-8, utilizando o formato W3C (World Wide Web Consortium) estendido (O'Leary, 2019).

O servidor web da Apache Tem um sistema robusto de *logging* que permite descrever o estado do servidor e as ações solicitadas pelos clientes (O'Leary, 2019). Os *logs* criados no Apache podem ser de acesso ou de erro. Os *logs* de acesso contém a informação dos pedidos feita ao servidor, o número de páginas visitadas pelos clientes, o estado dos pedidos e os tempos de resposta (Varanda, 2019). Por outro lado, os *logs* de erro registam os erros encontrados pelo servidor quando processaram o pedido (Varanda, 2019).

O Nginx guarda os erros do servidor web num ficheiro denominado de *error.log*, e pode ser configurado para gravar determinados tipos de *logs* ou para definir quais os campos que vão ser guardados em cada entrada (Soni, 2016).

- ***Logs* das *firewalls***

As *firewalls* são uma ferramenta de segurança necessária para proteger o computador de pedidos indevidos de ligação. No sistema operativo Microsoft Windows, a firewall incluída no sistema fica ativa no momento da instalação e bloqueia pedidos de ligações que não foram solicitados pelo cliente (Tidrow, Boyce, & Shapiro, 2015).

A firewall do Microsoft Windows costuma estar localizada no caminho: C:\Windows\system32\logfiles\firewall\pfirewall.log e pode ser visualizada através de um editor de texto. O ficheiro de cada entrada apresenta o tipo de tráfego TCP/UDP, a data/hora, os portos e o endereço de origem e de destino (Varanda, 2019).

A firewall dos sistemas operativos Linux é a Netfilter, sendo que as *iptables* são a ferramenta que normalmente as gere (Tevault, 2020). Os sistemas operativos CentOS e

OpenSuSe disponibilizam uma ferramenta gráfica para gerir as *iptables*, o sistema Ubuntu fornece a ferramenta de linha de comando UFW (Uncomplicated Firewall) para gerir as *iptables* (O'Leary, 2019). Para que sejam gerados *logs*, é necessário que sejam previamente criadas regras de configuração na firewall (Varanda, 2019).

Atualmente, as distribuições Debian e Ubuntu disponibilizam uma nova framework de classificação de pacotes, a *nftables*, sendo de referir que nos sistemas Red Hat 8 e CentOS 8 a *nftables* é a tecnologia incluída no pacote de serviços. A *nftables* ainda está em desenvolvimento e isso pode ser um problema, contudo esta ferramenta disponibiliza um conjunto de funcionalidades que permitem a gestão da firewall de uma forma mais simples, como por exemplo, permite a criação de várias ações com uma única regra, ao contrário das *iptables* em que é necessário a criação de várias regras para cada ação (Tevault, 2020).

- **Outros formatos**

O secure Shell ou SSH é utilizado para garantir uma ligação segura entre um servidor e uma estação de trabalho (O'Leary, 2019; Varanda, 2019). A autenticação pode ser efetuada através de um utilizador e de uma password ou com recurso a uma autenticação baseada chaves (Varanda, 2019).

A imagem seguinte representa um exemplo do formato do ficheiro s*yslog*, neste caso particular esta mensagem descreve um evento relacionado com o serviço SSH (*Secure Shell*).

```
1. Jan 28 11:42:59 sshd[1184]: server Accepted password for teste
   from 10.10.10.10 port 6541 ssh2
```

Diagrama IV - *Syslog* (Escola Superior de Redes, 2015)

Os *logs* das VPNs são muito importantes, pois o seu tráfego está encriptado e através destes é possível saber qual o utilizador, contabilizar os *logons* e *logoffs*, o IP externo e interno (Collins, 2017). Por exemplo, o serviço OpenVPN utilizado nos sistemas operativos Linux pode enviar os *logs* para o syslog e nos sistemas operativos Microsoft Windows os *logs* podem ser visualizados, por exemplo, através do Notepad (Keijser, 2017).

Anexo C - Componentes do Elastic Stack

Beats

Para o tratamento e ingestão de diversas fontes de dados a solução Elastic Stack possui o componente Beats, o mesmo é não comercial, e tem como função complementar o Logstash (Shukla et al., 2017) no entanto também pode enviar os *logs* diretamente para o Elasticsearch (Daubner, 2018).

O Beats tem como núcleo a biblioteca *libbeat*, escrita na linguagem Go que fornece uma API (Application Programming Interface) para transportar os *logs* para Elasticsearch ou para o Logstash (Daubner, 2018). A biblioteca *libbeat* possibilita a criação de um Beat personalizado (Zarzosa, 2017). A família Beats é constituída pelos seguintes Beats (Elasticsearch, 2019e): o Filebeat, o Metricbeat, o Winlogbeat, o Auditbeat, o Heartbeat, o Packetbeat e o Functionbeat. Cada Beat possui uma função diferenciada, têm a sua própria configuração e é executado como um serviço/daemon independente (Daubner, 2018).

O Filebeat proporciona uma forma simples e rápida de encaminhar e centralizar os *logs* (Elasticsearch, 2019m), pode ser utilizado para enviar os *logs* do sistema operativo Linux pois na sua maioria estes são baseados em texto (Hamilton, Gonzalez Berges, Tournier, & Schofield, 2018; Sharma, 2016). É constituído por vários módulos internos: auditd, Apache, NGINX, System, MySQL entre outros, simplificando a recolha e a visualização dos formatos de *logs* mais comuns (Elasticsearch, 2019m).

Caso o Filebeat seja suspenso, ele sabe onde parou e quando fica ativo envia os *logs* a partir do ponto de paragem. Quando envia dados para o Logstash ou para o Elasticsearch o Filebeat utiliza um protocolo que lhe permite aferir se os servidores estão ocupados no processamento dos dados, caso esta situação se verifique reduz o volume de dados a enviar até o congestionamento no servidor estar resolvido (Elasticsearch, 2019m).

Para recolher as métricas dos sistemas e serviços, utiliza-se o Metricbeat, por intermédio deste é possível recolher as estatísticas do NGINX, Redis, CPU (Central Processing Unit), sistemas de ficheiros, entradas e saídas do disco e da rede, memória, entre outros. Os seus módulos internos recolhem métricas de serviços como o: Apache, Jolokia, NGINX, MongoDB, MySQL, PostgreSQL, Prometheus, entre muitos outros. A sua instalação é fácil, pois não contém dependências basta ativar os módulos pretendidos no ficheiro de

configuração. Caso exista alguma interrupção este conserva os dados e quando os serviços voltarem a estar ativos envia-os para Elasticsearch ou para o Logstash (Elasticsearch, 2019t).

Na monitorização do tráfico da rede utiliza-se o Packetbeat, este suporta vários protocolos de Aplicação, é compatível com as bases de dados e também com os protocolos HTTP (Hypertext Transfer Protocol) de baixo nível. Através do Packetbeat é possível aferir: a latência, os erros das aplicações, os tempos de resposta, as tendências de acesso dos utilizadores, entre muitos outros. Como os Beats anteriores este mantém os dados caso exista uma paragem para posterior envio (Elasticsearch, 2019u).

No Windows Event Log como o próprio no indica utiliza-se o Winlogbeat, este facilita o conhecimento da infraestrutura Microsoft Windows. Os eventos de segurança como: *logons* (4624), falhas de *logon* (4625), novo software instalado (11707) ou dispositivos de USB ligados (4663) podem ser configurados no Winlogbeat para serem enviados para o Logstash ou para o Elasticsearch. Do mesmo modo que os Beats anteriores é tolerante a interrupções, garantindo o envio dos dados no momento que o serviço é inicializado (Elasticsearch, 2019aa).

O Auditbeat consegue recolher os dados da framework "Linux audit framework" sem interferir no auditd e envia em tempo real os *logs* para o servidor de *logs* definido. O Auditbeat agrupa as mensagens relacionadas num único evento, normaliza e fornece dados estruturados ao Elasticsearch, permite também ter alguma tolerância a interrupções como todos os Beats descritos anteriormente (Elasticsearch, 2019c).

Para verificar se os serviços Web ou Websites estão a funcionar utiliza-se o Heartbeat, este recolhe o estado e o tempo de resposta, é tolerante a interrupções e também consegue efetuar *pings* via ICMP (Internet Control Message Protocol), TCP (Transmission Control Protocol) e HTTP (Elasticsearch, 2019p).

O Functionbeat permite recolher, enviar e monitorizar os *logs* da Cloud e enviá-los para o servidor de *logs* para posterior análise (Elasticsearch, 2019n).

Em suma os Beats são muito úteis pois não requerem muito tempo de instalação ou configuração, ocupam pouco espaço, possuem módulos para as soluções mais populares, e possibilitam a criação de um *Dashboard* de forma rápida.

Logstash

O Logstash é uma ferramenta *open-source*, que opera do lado do servidor e é um excelente mecanismo de fluxo de dados, que ajuda na criação de pipelines de dados em tempo real (Shukla et al., 2017). Um dos principais objetivos do Logstash é recolher dados de várias fontes em simultâneo, transformá-los e enviá-los para o "*stash*" selecionado (Elasticsearch, 2019s; Zarzosa, 2017). O Logstash pode receber entradas de protocolos TCP/UDP, de ficheiros, de dados do syslog-ng e do rsyslog, e de outras ferramentas que os administradores instalem para analisar os *logs* tais como: Puppet, CFEngine, Nagios, Graphite e Zabbix (Sharma, 2016). Devido à utilização de filas de memória internas, as filas podem ser persistidas no disco rígido para que não existam perdas caso exista uma falha (Daubner, 2018). A nível empresarial a segurança do transporte dos dados está disponível em toda a cadeia de entrega (Elasticsearch, 2019g), para a licença Basic só estão disponíveis algumas funcionalidades de segurança.

Esta solução é escalável horizontalmente, oferece alta disponibilidade e suporta inúmeros plugins (Zarzosa, 2017). O Logstash possui uma arquitetura extensível baseada em plugins, facultando suporte a quatro tipos de plugins: de entrada, de filtro, *codecs* e de saída (Shukla et al., 2017; Zarzosa, 2017). Devido à compatibilidade com mais de 200 plugins é possível combinar, misturar e adequar as diferentes entradas, filtros e saídas e trabalhar no pipeline em equilíbrio (Elasticsearch, 2019s). Os dados podem ser transformados através dos filtros do Logstash, que possui plugins para os mais diversificados tipos de dados, por exemplo caso o endereço IP (Internet Protocol) se encontre nos dados, o plug-in GeoIP pode adicionar a geolocalização ao mesmo, similarmente na saída podem ser obtidas outras informações que posteriormente podem ser utilizadas pelo Kibana para desenhar mapas (Srivastava, 2019).

O Logstash apresenta várias funcionalidades e recursos, em seguida listam-se algumas das mais relevantes (Chhajed, 2015; Elasticsearch, 2019g; Gupta & Gupta, 2017; Saxena & Gupta, 2017; Senanayaka, 2018):

- É *open-source*;
- Permite anonimizar dados sensíveis;
- Simplifica o processamento dos dados independentemente da origem, formato ou o esquema de dados.
- Integração perfeita com o Elasticsearch, os Beats e o Kibana;

- Processamento de dados centralizado;
- Escalável horizontalmente com alta disponibilidade e balanceamento da carga;
- Analisa, filtra e transforma os dados recorrendo a operações com baixa latência;
- Extensibilidade, fornece a possibilidade para criar e desenvolver formas de recolher os dados (entrada), filtrar e enviar os dados (saída);
- Interoperabilidade, permite utilizar outros componentes e ferramentas;
- Durabilidade da mensagem, existindo a garantia da entrega (pelo menos uma vez nos Beats: Filebeat ou o Winlogbeat);
- Variedade, sendo compatível com uma ampla diversidade entradas;
- Transporte seguro de ponta a ponta com autenticação e criptografia;
- API de monitoramento;
- Arquitetura de pipeline compatível com mais de 200 plugins, além disso permite também o desenvolvimento de plugins.

Em suma o Logstash é um pipeline de processamento de *logs open-source,* que pode recolher dados de várias fontes em simultâneo, transformá-los e de seguida enviá-los para armazenamento (Saxena & Gupta, 2017). Este certifica-se que os dados são transportados de forma escalável, estável e segura para o Elasticsearch (Elasticsearch, 2019g).

Elasticsearch

O Elasticsearch é um mecanismo de análise e pesquisa de "*texto completo*", altamente escalável e *open-source*, além disso permite que se armazene, pesquise e analise um grande volume de dados quase em tempo real (Elasticsearch, 2019o).

É possível efetuar diversos tipos de pesquisas, estas podem ser estruturadas, não estruturadas, métricas, geográficas ou definidas pelo utilizador através do Elasticsearch (Elasticsearch, 2019k), este também possui uma natureza distribuída o que permite gerir e dimensionar de uma forma fácil o crescimento do volume de dados (Andhavarapu, 2017).

A RESTful API no Elasticsearch oculta a complexidade do Apache Lucene, permitindo que as operações sejam mais simples e estejam disponíveis para qualquer linguagem de programação (Chhajed, 2015).

Na versão 5.x foi adicionada uma nova linguagem de script (*Painless*), esta é semelhante ao Groovy[61], simples de utilizar, eficiente no desempenho e não requer a instalação de nenhum plugin. O *Painless* foi desenhado para o Elasticsearch e permite que os scripts sejam executados em segurança (Dixit, 2017).

Existem vários conceitos importantes no Elasticsearch, tais como: Cluster, Node (Nó), Index (Índice), Type (Tipo), Document (Documento), Shards & Replicas (Elasticsearch, 2019d). Um cluster é conjunto de um ou mais nós (servidores), no qual todos os nós do cluster, preservam os dados e fornecem recursos para a indexação e pesquisa (Elasticsearch, 2019d). Por padrão o nome sugerido para o cluster é "*elasticsearch*", todavia cada cluster deve ter um nome exclusivo (Elasticsearch, 2019d). Um nó pode ser definido como um único servidor que faz parte do cluster, e que armazena os dados e participa dos recursos de indexação e de pesquisa do cluster. O Elasticsearch atribui um nome exclusivo a cada nó, mas o utilizador pode renomeá-lo para que seja mais fácil identificar o nós (Elasticsearch, 2019d). Dependendo das necessidades de implementação, podem-se adicionar ou remover nós momentaneamente (Andhavarapu, 2017). A imagem seguinte representa um cluster com três nós a trabalharem em conjunto, na imagem também está representado o *Shard*, que permite que um índice possa de ser dividido em várias partes (Andhavarapu, 2017; Elasticsearch, 2019d).

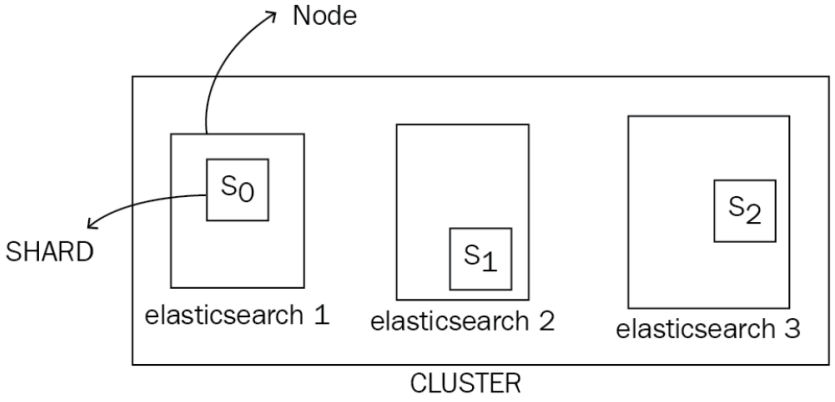


Diagrama V - Exemplo de um Cluster com três Nós (Andhavarapu, 2017, p. 11)

Um índice é uma coleção de documentos com caraterísticas semelhantes, pode-se ter um índice para os clientes, um para os produtos e outro para os pedidos estes são identificados por um nome diferenciado, devendo estar em minúsculas (Elasticsearch, 2019d). Num cluster pode-se definir o número de índices que se necessitar (Elasticsearch, 2019d). A

[61] Apache Groovy programming language

indexação de palavras significa inserir/atualizar os documentos num índice (Andhavarapu, 2017).

O *Type* foi descontinuado na versão 6.0.0 do Elasticsearch, é uma categoria/partição lógica do índice, que permitia armazenar diferentes tipos de documentos no mesmo índice (Elasticsearch, 2019d).

No Elasticsearch um documento é a unidade básica de informação que pode ser indexada, este documento está no formato JSON e pode-se armazenar o número de documentos que se desejar no índice (Elasticsearch, 2019d). O Elasticsearch consegue determinar automaticamente os tipos de dados dos campos no documento (Andhavarapu, 2017). A Tabela 1 relaciona os conceitos na linguagem SQL (Structured Query Language) com os conceitos utilizados no Elasticsearch (Andhavarapu, 2017).

Tabela 1 – Comparação entre os conceitos da linguagem SQL e o Elasticsearch

Database	Table	Row	Column
Index	Type (*Descontinuado na versão 6.0.0*)	Document	Field

Um grande volume de dados pode ser armazenado num índice, esta caraterística pode implicar que se exceda os limites do hardware de um nó ou lentidão nas pesquisas, para solucionar este problema o Elasticsearch permite que o índice seja dividido em várias partes (*shard*s). Quando se cria um índice é possível definir o número de *shards* desejados, onde cada *shards* é totalmente funcional e independente e pode ser hospedado em qualquer nó do cluster. Esta funcionalidade é importante pois permite que se divida e dimensione horizontalmente o conteúdo, aumentando o desempenho e rendimento. É importante fazer um levantamento dos requisitos da Entidade o que vai permitir definir o número correto de fragmentos, embora seja possível alterar o número de *shards* de um índice este é um processo complicado. Por norma, para cada índice do Elasticsearch é alocado em um *shard* primário e uma réplica (Elasticsearch, 2019d).

Como funcionalidades e recursos do Elasticsearch podem-se listar as seguintes (Andhavarapu, 2017; Daubner, 2018; Elasticsearch, 2019k; Gupta & Gupta, 2017; Senanayaka, 2018):

- Indexa qualquer tipo de dados;

- Disponibilidade e análise de dados em tempo real, assim que os dados são indexados, eles são disponibilizados para pesquisa e análise;
- Suporta mapeamento automático/dinâmico, processando os dados não estruturados automaticamente;
- Suporta agregações, que podem ser configuradas para criar resumos complexos de dados;
- As últimas consultas realizadas são guardadas em cache, quando os dados são modificados a cache é invalidada;
- Distribuído, pode-se configurar o número de nós que se desejar;
- Escalabilidade e alta disponibilidade;
- Segurança, as alterações são registadas em vários nós para que possa ser possível preservar os dados em casos de falha;
- API RESTful, com esta API e recorrendo ao JSON, a maioria das operações pode ser executada: pode-se adicionar um documento a índice, excluir ou atualizar uma entrada;
- Múltiplas bibliotecas de cliente em várias linguagens de programação (Java, Python, .NET, SQL e PHP);
- Resiliência, consegue detetar as falhas e gerir os *logs* para que estes fiquem seguros e disponíveis.

Em resumo o Elasticsearch é um mecanismo de pesquisa de "*texto completo*" quase em tempo real, que pode trabalhar com grandes volumes de dados, é escalável e tolerante a falhas.

Kibana

O Kibana é uma plataforma *open-source* desenhada para trabalhar em conjunto com o Elasticsearch (Elasticsearch, 2019q). A plataforma Kibana é utilizada para pesquisar, visualizar e interagir com os dados armazenados nos índices do Elasticsearch (Elasticsearch, 2019q). Esta plataforma vai facilitar a compreensão da informação de grandes volumes de dados, a sua interface é simples e permite criar rapidamente *dashboards* que mostram dados em tempo real (Elasticsearch, 2019q). A instalação do Kibana é fácil e não necessita de infraestrutura adicional (Elasticsearch, 2019q), e pode ser aplicada a diferentes casos de uso como a monitorização de sistemas ou de aplicações (Srivastava, 2019).

As funcionalidades que o Kibana na versão 7.0 disponibiliza estão organizadas por separadores tais como (Chappell, 2019; Elasticsearch, 2019q): Discover, Visualize, Dashboard, Canvas, Maps, Machine Learning, Infrastructure, Logs, Application Performance Monitoring (APM), Uptime, Dev Tools, Monitoring e Management.

O separador *Discover* fornece os recursos para analisar os *logs* armazenados, neste podemos executar vários filtros, definir quais as colunas a visualizar, exibir todos os campos de um registo ou converter a visualização tabular em visualização JSON. Este também apresenta um histograma, que organiza os dados levando em consideração o *timestamp* dos *logs* (Srivastava, 2019).

Existe uma grande diversidade de filtros disponibilizados no separador *Discover*, podemos filtrar os dados por tempo relativo onde se pode visualizar os dados da última semana ou dos últimos 15 minutos. Para facilitar as pesquisas é possível ativar a funcionalidade de preenchimento automático, assim quando se escreve um comando somos auxiliados por este recurso. Caso existam resultados que correspondem aos critérios inseridos estes são destacados a amarelo para facilitar a sua identificação (Srivastava, 2019).

A visualização é a essência do Kibana, no separador *Visualize* é possível visualizar todos os dados que estão armazenados no Elasticsearch recorrendo a uma variedade de gráficos e mapas (Gupta & Gupta, 2017). O Kibana utiliza as agregações do Elasticsearch para criar diferentes tipos de visualizações, que são as seguintes (Srivastava, 2019): Area Chart, Heat Map, Pie Chart, Data Charts, Data Table, Metric e Tag Cloud. Através da visualização dos dados é possível analisar tendências, identificar anomalias nos dados ou no padrão de utilização com bastante facilidade (Srivastava, 2019).

O separador *Visualize*, possui uma caraterística importante o *Timelion* que foi projetado para trabalhar em dados e respetivas séries temporais, através deste pode-se obter respostas a algumas perguntas difíceis de responder, ou seja, por intermédio deste sabemos a diferença entre os dados do último domingo e do anterior. O *Timelion* pode interligar diferentes fontes de dados e integrá-las numa única visualização, a sua linguagem é relativamente simples e na mesma visualização é possível visualizar os dados atuais e o histórico (Srivastava, 2019).

Através do painel *Dashboard* é possível reunir todas as pesquisas ou visualizações criadas num só local, estas podem ser organizadas de acordo com a necessidade do analista, pois este permite redimensionar, mover, editar ou remover todas as visualizações que foram

inseridas. Os dados visualizados no *Dashboard* podem ser atualizados em tempo real e também partilhados (Gupta & Gupta, 2017).

Para criar relatórios altamente personalizados com um conjunto de componentes também personalizáveis utiliza-se o *Canvas* que é uma funcionalidade sem custos. Esta função é distinta das restantes funcionalidades, e apresenta um espaço de trabalho onde se pode criar conjuntos de slides (semelhante ao conceito do Microsoft PowerPoint) com o nome de *workpad* (Collier & Azarmi, 2019).

No *Maps* é possível analisar os dados geográficos num mapa planisfério e onde se pode aumentar ou diminuir o nível de detalhe, ajudando a uma melhor compreensão dos dados que se está a visualizar (Elasticsearch, 2019q).

O Data Visualizer está disponível na licença Basic, neste é possível obter informação pertinente sobre os dados armazenados no Elasticsearch e selecionar os campos que vão ser utilizados no Machine Learning. Na licença Platinum ou de avaliação, é possível gerir as tarefas do *machine learning* e usufruir de todas as capacidades do Machine Learning. O *machine learning* no Kibana modela automaticamente o comportamento normal dos dados pelas séries temporais simplificando a análise dos dados, reduzindo os falsos positivos e identificando anomalias. Conforme os dados vão aumentando de tamanho e de volume, o esforço humano para analisar e identificar problemas de infraestrutura, ataques informáticos ou problemas no negócio torna-se impraticável sem recorrer ao *machine learning* (Elasticsearch, 2019q).

O separador *Infrastructure*, supervisiona a infraestrutura e identifica problemas em tempo real, pode-se analisar métricas e *logs* de servidores. Esta funcionalidade apresenta o estado da infraestrutura de uma forma genérica e posteriormente recorre-se à sua interface interativa para obter detalhes (Elasticsearch, 2019q).

Para analisar os *logs* armazenados o Kibana disponibiliza o separador Logs com uma interface compacta, semelhante a uma consola (Elasticsearch, 2019q).

A opção APM (Application Performance Monitoring) recolhe métricas detalhadas de desempenho e erros das aplicações em tempo real, esta é fornecida com a licença básica, contudo, para usufruir de todas as suas funcionalidades é necessário adquirir a licença Platinum (Elasticsearch, 2019q). Embora muitas operações possam ser realizadas no browser através das ferramentas disponibilizadas, o APM possui bastantes vantagens

adicionais, pois ele também recolhe exceções e erros e não tratados e também disponibiliza opções de pesquisa avançadas (Srivastava, 2019) .

A funcionalidade paga Uptime permite gerir o tempo de funcionamento das aplicações (Paro, 2019), esta é uma métrica importante na prestação de serviços (Hershkovitch, 2019).

O separador Dev Tools contém ferramentas de desenvolvimento que podem ser utilizadas para interagir com o Kibana (Elasticsearch, 2019q): Console, Search Profiler e Grok Debugger. Na funcionalidade Console executam-se as pesquisas do Elasticsearch, é constituída por dois painéis, no painel esquerdo executam-se as pesquisas e no painel do lado direito são disponibilizados os resultados. O Search Profiler permite criar perfis em qualquer consulta do Elasticsearch, ao executar uma pesquisa este fornece um relatório do perfil da consulta com os tempos de duração para cada componente o que permite otimizar as consultas. O *Grok* é uma sintaxe de correspondência de padrões que se pode utilizar para estruturar dados (Elasticsearch, 2019q), o Grok Debugger permite construir *grok patterns* antes de estes serem utilizados nas pipelines de processamento do Logstash (Srivastava, 2019).

O separador Monitoring possui dois objetivos, um é supervisionar o Kibana e o outro é supervisionar os dados do Elastic Stack. Este permite visualizar e supervisionar o desempenho e a integridade dos dados do Elasticsearch, Logstash e Beats em tempo real e também o seu histórico (Elasticsearch, 2019q).

Para configurar o Kibana utiliza-se o separador Management, neste realizam-se as configurações básicas e avançadas do Kibana, ajustam-se comportamentos e gerem-se as licenças (Elasticsearch, 2019q).

Para descobrir como os itens de um índice se relacionam, existe o recurso pago Graph que é uma rede de itens relacionados (Elasticsearch, 2019q). Esta ferramenta poderosa fornece um meio elegante e simples de descobrir as relações entre os termos indexados e a sua relevância (Srivastava, 2019). O *Spaces* está disponível a partir da versão 6.5 do Kibana, permite organizar os painéis de visualização e outros objetos que estejam guardados classificando-os em categorias relevantes (Gregory, 2018). O Elastic SIEM surge como versão beta no Elastic Stack 7.2, para análises de segurança e deteção de ameaças (Elasticsearch, 2019y).

O Kibana possui múltiplas funcionalidades e recursos, enumeramos de seguida algumas (Chhajed, 2015; Gupta & Gupta, 2017; Senanayaka, 2018; Settle et al., 2019; Srivastava, 2019):

- Pesquisa de dados com filtros (existem filtros de data, por campo, filtros interativos, entre outras opções);
- É possível gravar as pesquisas para que possam ser reutilizadas;
- Disponibiliza diferentes tipos de visualização de dados estruturados e não estruturados;
- Permite colocar *geodados* em qualquer mapa utilizando o Elastic;
- Capacidade de manipular e analisar grande volume de dados quase em tempo real;
- Fornece uma interface intuitiva, fácil de utilizar e altamente personalizável;
- Os Dashboards podem ser facilmente geridos e partilhados em diferentes sistemas;
- Os termos da pesquisa são destacados na lista de documentos a amarelo;
- Utiliza as agregações e subagregações do Elasticsearch;
- Possibilita a utilização de campos com script, permite a otimização das pesquisas e fornece ferramentas de desenvolvimento que podem interagir com o Kibana;
- Fornece o Machine Learning que modela automaticamente o comportamento normal dos dados tendo em conta o espaço temporal.
- Elastic SIEM, permite a análise dos *logs* de segurança.

O Kibana é responsável pela análise e gestão dos dados do Elasticsearch e de uma forma genérica podemos dizer que permite a execução de consultas e a visualização dos dados.

Elastic SIEM

O Elastic SIEM foi lançado como uma versão beta no Elastic Stack 7.2, e recorre ás funcionalidades do Elastic Stack, beneficiando da velocidade, da escalabilidade e do poder analítico da solução para realizar operações de segurança e também para procurar ameaças. No Elastic Stack 7.2 é mais fácil recolher os dados das mais diversificadas origens pois os Beats possuem novos recursos relacionados com os eventos de segurança, faz a normalização dos *logs* de segurança com o ECS permitindo a correlação entre diferentes equipamentos e a interação em segurança com os dados recorrendo ao SIEM app no Kibana (Elasticsearch, 2019y; Settle et al., 2019).

A Diagrama VI apresenta a estrutura do Elastic SIEM, este processa os eventos de segurança dos diversos equipamentos e apresenta informação relevante para a segurança da infraestrutura. Como se pode visualizar o Elastic SIEM é disponibilizado com o Kibana.

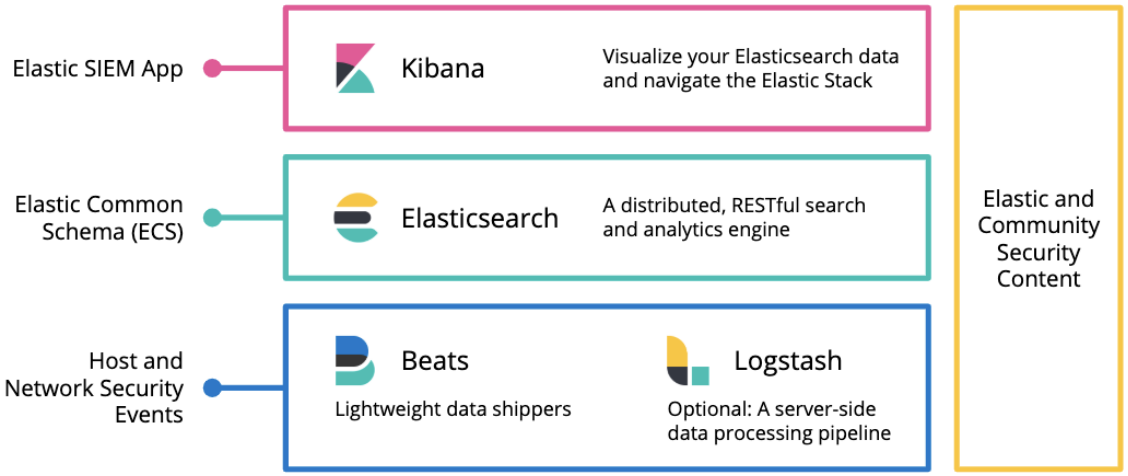


Diagrama VI - Estrutura da APP Elastic SIEM (Elasticsearch, 2019x)

As funcionalidades descritas na App Elastic SIEM são disponibilizadas em todas as licenças da solução Elastic SIEM, todavia o *machine learning* continua a ser uma funcionalidade paga (Settle et al., 2019).

Em suma o Elastic SIEM é um espaço de trabalho interativo onde os analistas podem fazer a triagem dos eventos e executar investigações iniciais. O histograma distribui os *logs* pelo período temporal (data da sua criação), onde é possível recolher, armazenar e partilhar evidencias de um ataque (Elasticsearch, 2019y; Settle et al., 2019).

Anexo D – Conformidade com o Regulamento Geral de Proteção de Dados

Neste ponto fazemos um resumo das funcionalidades que cada solução apresenta na sua versão comercial para garantir a conformidade com o RGPD.

Splunk

As opções de suporte Base (Splunk Light), Standard (Splunk Cloud e Splunk Enterprise) e Premium (está disponível para Splunk Cloud e Splunk Enterprise) estão em conformidade com o RGPD e caso o utilizador opte por estas soluções pagas, um consultor pode acompanhar todo o processo (Splunk, 2019c, 2019b). O Splunk através das funcionalidades que disponibiliza assegura a segurança dos dados e as notificações de violações (Artigos 32, 33 e 34), permite fazer auditorias de segurança aos dados pessoais (Artigo 58) e finalmente fornece relatórios e pesquisas aos dados pessoais (Artigos 15, 17, 18, 28) (Splunk, 2019c).

De seguida listam-se a título de exemplo algumas funcionalidades que potenciam a conformidade com o RGPD são (Dezeure, 2018; Splunk, 2020a):

- Monitorizar a utilização do Splunk para auditar se existem anomalias;
- Faz auditorias de segurança aos dados pessoais;
- Efetuar notificações de violação de dados;
- É garantido o transporte e a gestão dos dados em segurança, através de comunicações encriptadas;
- Permite vários tipos de permissões e autenticação ”*two-factor*”;
- Faz a monitorização dos dados e efetua alertas em tempo real;
- Oferece resiliência com as seguintes funcionalidades: *clustering*, pesquisa distribuída, alta disponibilidade e recuperação a desastres;
- Proteção de dados por design e por padrão.

Pelas funcionalidades elencadas, o Splunk fomenta a conformidade com o RGPD nas suas versões pagas.

Elastic Stack

Existem várias funcionalidades do Elastic Stack que podem contribuir para que a Organização esteja em conformidade com o RGPD. O diagrama seguinte relaciona as funcionalidades do Elastic Stack com as várias etapas que uma Entidade necessita de realizar para estar em conformidade com o RGPD. Na fase de preparação as Organizações devem identificar todos os fluxos de dados onde os dados pessoais são controlados ou processados. O Elastic Stack pode ser utilizado no mapeamento do fluxo de dados, planeamento e retenção de dados pessoais e na revisão dos contratos de fornecedores (Elastic, 2018).

No planeamento e retenção de dados pessoais a Organização decide quanto tempo os dados pessoais vão ser armazenados (período de retenção). O RGPD especifica que a retenção dos dados pessoais recolhidos deve ser limitada, tendo as Organizações a obrigatoriedade de excluir os dados quando não forem mais necessários ou quando o Sujeito dos dados retirar o consentimento. O Elasticsearch suporta índices baseados no tempo de criação, este podem ser excluídos após o período de retenção recorrendo a ferramentas como o Ansible, Chef, Puppet ou o Elastic Cloud Enterprise (ECE) onde é possível controlar a versão e a retenção de dados (Elastic, 2018).

O Elastic Stack pode ser utilizado em vários cenários de proteção: pode-se proteger o Elastic Stack quando ele é utilizado para armazenar os dados pessoais ou pode ser utilizado para contribuir com medidas de segurança, agindo como uma plataforma centralizada de registo e análise de segurança (Elastic, 2018). Através das funcionalidades que antigamente faziam parte do X-Pack o Elastic Stack pode responder aos seguintes requisitos (Elastic, 2018): Proteção de dados por design e padrão; Criptografia e pseudonimização*;* Controle de acesso; Gestão de *logs;* Monitorização e Deteção; Resiliência e Recuperação de Desastres .

O Elasticsearch pode auxiliar a gerir o processo de privacidade, quando um Titular dos dados pessoais exerce o seu direito ao esquecimento ou retirara o consentimento dado para a recolha de dados pessoais, neste ponto o maior desafio pode ser encontrar realmente todos dados pessoais do Titular. O Elasticsearch pode contribuir para a agilização do processo, pois efetua uma identificação rápida dos dados pessoais de um Titular em tabelas, consultas, relatórios ou aplicações. Recorrendo a APIs é exequível que se execute a ação apropriada para atender às solicitações do GDPR (Elastic, 2018).

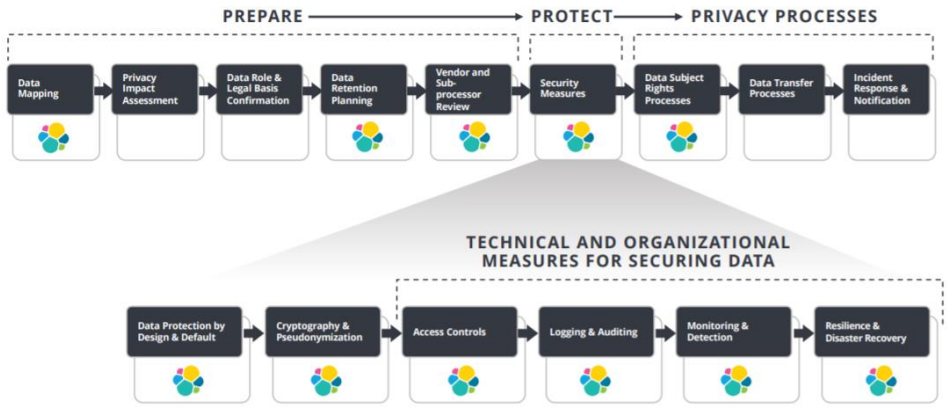


Diagrama VII - Funcionalidades dos Elastic Stack e o seu contributo para a conformidade com o RGPD (Elastic, 2018, p. 9)

O Elastic Stack é um meio facilitador para a conformidade do RGPD, o Elastic Cloud Enterprise permite que a Organização cumpra os requisitos de proteção por design e a minimização dos dados pessoais. As outras funcionalidades do Elastic Stack também contribuem para implementar uma abordagem eficaz para estar em conformidade com o RGPD, esta solução disponibiliza: segurança, controlo de acesso, resiliência, recuperação de desastres, criptografia, pseudonimização e monitorização (Elastic, 2018).

Graylog

A *Stream* é uma funcionalidade importante para a conformidade com o RGPD, através do sistema de pipelines é possível efetuar o encaminhamento do tráfego e também tratar os *logs* ou remover dados que já não são necessários (Black, 2017).

O *Index Set* é outra funcionalidade relevante para o RGDP, através das *Streams* é possível separar os dados, os dados que possuem informações pessoais são enviados para um *Index Set* onde se pode aplicar um período de retenção menor e os dados que não contem dados pessoais são enviados para outro *Index Set* com período de retenção maior. As funcionalidades anteriores vão ajudar a cumprir a política de retenção de dados, uma vez que é possível definir um processo para armazenar os dados, separar os dados pessoais dos não pessoais e caso seja solicitado proceder e evidenciar a sua remoção (Black, 2017).

Com a proteção de dados por design e padrão é necessário controlar quais os dados que são visualizados e quem os pode aceder, o Graylog permite que sejam atribuídos níveis de permissão através de "*Users e Roles"* às *Streams* e *Dashboards* configurados na solução (Black, 2017).

A auditoria e monitorização aos *logs* contribuem para a conformidade com o RGPD, o Graylog permite procurar problemas de segurança e violações, possibilitando que os administradores tenham uma visão global do estado da rede. Caso o Titular dos dados solicite o direito ao esquecimento o Graylog permite de uma forma rápida pesquisar informações sobre o Titular e caso seja solicitado apagar os seus dados (Black, 2017).

Uma funcionalidade importante para o RGPD são os *logs* de auditoria de utilizador, este recurso armazena e regista as ações/operações realizadas pelos utilizadores no servidor Graylog. Permitindo que exista um rastreamento de todas as atividades efetuadas pelos utilizadores e a sua segurança (Simko, 2018).

Caso exista uma perda de energia ou um pico no envio dos *logs* é importante ter alguma resiliência. O Graylog com a funcionalidade *Journal*, permite que os dados sejam armazenados no momento que chegam, assim se existir uma interrupção de rede ou de energia pode-se garantir que não existem perdas. Outra funcionalidade relevante são os Buffers, permitindo que se controle o envio dos dados para o Elasticsearch para que não exista sobrecarga do servidor(es). Através dos clusters do Graylog e do Elasticsearch é possível implementar redundância e resiliência (Black, 2017).

OSSIM

Uma das funcionalidades importante para o RGPD e que está disponível nas duas soluções é a possibilidade de criar relatórios de conformidade (PCI DSS e ISO27001). Também é possível como já foi referido anteriormente criar outros tipos de relatórios o que pode contribuir para a redução dos riscos na Entidade e promover a conformidade com o RGPD (Malik, 2017).

A versão comercial USM faz a auditoria às ações dos utilizadores e administradores, mas não guarda os acessos que cada utilizador realiza aos dados pessoais (UnifiedThreatWorks, 2020).

A solução USM, permite eliminar automaticamente os *logs*, basta definir o período de retenção para os mesmos. Existem outras funcionalidades disponibilizadas pela solução USM, algumas das quais são disponibilizadas na versão *open-source* (Malik, 2017):

- Detetar sistemas desconhecidos na rede;

- Avaliação de vulnerabilidades, o que permite identificar prováveis alvos para invasores;
- Deteção de atividade maliciosa (rede e *hosts*);
- Monitorização da integridade dos ficheiros, deteta alterações em ficheiros críticos ou atividades suspeitas do utilizador;
- Gestão de *logs*, faz a preservação de evidências para posteriores análises forenses;
- Alimentação integrada de inteligência para gestão de ameaças;
- Recursos de deteção de ameaças continuamente atualizados que permitem detetar automaticamente ameaças emergentes.

Em suma a solução comercial USM garante a conformidade com o RGPD.

Anexo E – Boas práticas das soluções

Splunk

Para que se possa administrar a solução Splunk de forma eficiente é importante normalizar os termos utilizados nos índices e nos componentes dos quais se vão recolher os *logs*. Uma vez que a solução altera as maiúsculas para minúsculas, devem-se nomear os índices com minúsculas, recomendam-se que os dados, para os vários tipos, apresentem, caso se aplique, o fornecedor, o tipo de dispositivo, a APP e o tipo de dados a indexar separados por um caractere, como por exemplo os dois pontos (Contreras et al., 2018).

Em relação ao *Cluster Master* deve-se definir qual o local em que vão estar os ficheiros "indexes.conf". Na implementação e gestão da solução, é importante documentar o que foi feito e explicar as decisões tomadas, além disso, também é importante comentar as linhas dos ficheiros .conf para que as configurações sejam utilizadas de forma apropriada no futuro (Contreras et al., 2018).

Caso seja possível, deve-se implementar a solução num ambiente de testes, para que se possa testar as novas aplicações, impedindo que os erros de instalação/configuração comprometam o ambiente de produção (Contreras et al., 2018).

Sempre que possível, devem ser utilizadas operações automáticas, permitindo assim que seja possível despender mais tempo em optimizar solução (Contreras et al., 2018). As APPs disponíveis no Splunkbase oferecem um grande número de funcionalidades extras, por isso recomenda-se que se recorra às APPs para gerir o ambiente da aplicação (Lopez, 2018).

Elastic Stack

Caso se pretenda utilizar o Elastic Stack como uma solução a longo prazo, é necessário realizar um estudo aprofundado da Organização para que se possa definir a infraestrutura necessária e estimar o volume de dados que vai ser tratado. Os clusters devem ser balanceados tendo em conta a necessidade de negócio, ou seja se existe um número gradual de pessoas a utilizar o sistema, pode ser necessário aumentar o número de nós para responder às solicitações (Corcoran, 2017).

A ativação das filas persistentes é aconselhável quando existe segurança ponto a ponto[62] (Elasticsearch, 2019g). Para que se possa ter alta disponibilidade recomenda-se que o Logstash tenha no mínimo dois nós e que cada Beat possua um nó no Logstash, pese embora também seja aceitável um nó por cada tipo de dados (Elasticsearch, 2019g).

Para se conseguir mais controlo sobre os dados, devem ser individualizados e personalizados os níveis de acesso através da definição dos níveis de permissões para cada utilizador (Corcoran, 2017).

A criptografia é recomendada para o transporte dos dados recolhidos pelos Beats para o Logstash e do Logstash para o Elasticsearch, sendo que existem várias opções de segurança que o Elastic Stack dispõe, a saber (Elasticsearch, 2019g): TLS, PKI (Public Key Infrastructure), LDAP (Lightweight Directory Access Protocol), AD (Active Directory).

Quando se partilha um *dashboard*, é muito importante que se tenha em consideração que o link deve ser *tokenizado*[63], para tal existem soluções pagas e gratuitas para realizar esta operação (Berman, 2018).

Graylog

Para que a solução Graylog possa ser eficiente, é necessário ter em consideração as seguintes regras básicas para conseguir a escalabilidade dos componentes (Graylog, 2018a): os nós Graylog devem ter disponíveis o máximo de CPU possível, os nós do Elasticsearch devem ter os discos mais rápidos que a Organização pode suportar, o mesmo se aplica para a RAM (Random Access Memory), finalmente o MongoDB não necessita de recursos significativos, pois só armazena os metadados e as configurações.

Para que não existam perdas de dados no Cluster do Elasticsearch, devem ser realizados backups dos índices (Graylog, 2018a). Além disso, o Graylog deve ser supervisionado, porque caso este fique sem espaço de armazenamento, será complicado restaurar o estado anterior, podendo mesmo implicar a instalação e a configuração de um novo servidor (Simko, 2018).

[62] Definir **queue.checkpoint.writes: 1** para ativar a escrita pelo menos uma vez
[63] Operação que consiste em substituir dados sensíveis por dados não sensíveis

OSSIM

Para as pequenas e médias empresas, a arquitetura All-in-One da USM Appliance é a indicada, pois não necessita de uma equipa de suporte grande para assegurar a sua gestão, além de integrar no mesmo servidor todos os componentes da solução. Caso a empresa necessite de mais recursos, esta solução é escalável (AT&T Cybersecurity, 2020b).

Para que a base de dados ou o disco não ocupem todo o espaço disponível, existem algumas boas práticas que permitem evitar que ocorram problemas de desempenho na solução USM, tais como (AT&T Cybersecurity, 2019d, 2020b): configurar limites para o backup e armazenamento; alterar a vida útil dos Alarmes de acordo com as necessidades da Entidade; limpar os *logs* do sistema e as caches regularmente; caso seja necessário, podem ser eliminados todos os dados da base de dados manualmente; por norma, os *logs* são armazenados sem tratamento na base de dados até serem excluídos, estes *logs* devem ser exportados para outro repositório de dados, removidos manualmente ou ter um tempo de validade, para que possam ser eliminados automaticamente; caso se tenha implementado uma arquitetura All-in-One USM Appliance, deve ser configurado o USM Appliance Logger em separado para reduzir a carga do servidor.

Anexo F – Configuração do Watcher no Kibana

O Watcher para que possa enviar e-mail recorreu aos seguintes componentes:

- **trigger** - no código construído o **trigger** é executado a cada 30 segundos;
- **input** – esta secção vai carregar os dados no Watcher, estes dados são denominados de **payload.** Os dados vão estar disponíveis nas fases seguintes o que vai permitir que possam ser criadas **conditions** ou gerar **actions**. Para avaliar o **payload** recorre-se à variável **ctx.payload**. No código vai-se avaliar o índice ou índices com o nome "**filebeat-*"**, de seguida este vai carregar no **payload** o número de documentos encontrados que contenham na mensagem a seguinte frase **" (maximum authentication attempts exceeded) "** nos últimos 5 minutos.
- **condition** - recorre-se à condição do tipo **compare** para determinar se o **payload** possui algum documento e se encontrar algum a **action** vai ser invocada;
- **actions** – Caso a condição seja avaliada como verdadeira a ação vai enviar um e-mail de alerta para anav@gmail.com com o assunto "Warning: Multiplas tentativas de login SSH".

```
PUT _xpack/watcher/watch/failed-login
{
  "trigger": {
    "schedule": {
      "interval": "30s"
    }
  },
  "input": {
    "search": {
      "request": {
        "indices": "filebeat-*",
        "body": {
          "size": 0,
          "query": {
            "bool": {
              "filter": [
                {
                  "range": {
                    "@timestamp": {
                      "from": "now-5m",
                      "to": "now"
                    }
                  }
                },
                {
                  "query_string": {
                    "default_field": "message",
                    "query": "(maximum authentication attempts exceeded)"
                  }
                }
              ]
            }
          },
          "aggs": {
            "types_count": {
              "value_count": {
                "field": "type"
              }
            }
          }

        }
      }
    }
  },
  "condition": {
    "compare": {
      "ctx.payload.hits.total": {
        "gt": 10
      }
    }
  },
  "actions": {
    "email_admin": {
      "email": {
        "to": "        __    .com",
        "subject": "Warning: Multiplas tentativas de login SSH",
        "body": ""
      }
    }
  }
}

```

Anexo G – Ferramentas complementares / alternativas do Elastic Stack

Como nem todas as funcionalidades da solução Elastic Stack são gratuitas vamos enumerar algumas alternativas de software livre e também ferramentas que possam complementar as funcionalidades que a solução disponibiliza de forma gratuita.

Começamos pelo Search Guard que é uma alternativa gratuita compatível com as últimas versões do Elasticsearch que pode garantir a segurança do Elastic Stack (Daubner, 2018; Marquina, 2018), todavia só a licença paga garante a total conformidade com o RGPD (Search Guard, 2018). Por exemplo tal como no Elastic Stack as funcionalidades (Floragunn GmbH, 2018): *Active Directory*, LDAP, auditoria de *logs* e a segurança ao nível do documento só estão disponíveis na versão paga. O produto ReadonlyREST (ReadonlyREST, 2018b) para o Elasticsearch oferece várias funcionalidades tais como (ReadonlyREST, 2018b): encriptação (exemplo: Fast HTTPS), autenticação (exemplo: LDAP), controlo de acessos (exemplo: ao nível do Elasticsearch e da camada transporte) e de autorização (exemplo: LDAP), porém quase todas as opções de segurança para o Kibana são pagas.

As soluções SENTINL 6 (Marquina, 2018; Siren, 2018) e a Elastalert (Yelp.com, 2019) são *open souce* e podem substituir o produto pago do Elastic Stack o Alerting, a primeira solução suporta a versão 6.x do Kibana e o ElastAlert suporta a versão 7 do Elasticsearch. As duas opções permitem definir alertas, a diferença entre elas é que o SENTINL 6 é um plugin do Kibana e também permite criar relatórios e o ElastAlert não é um plugin e é executado de forma independente (Daubner, 2018).

Para que se possa gerir e monitorizar os cluster do Elasticsearch pode-se recorrer à ferramenta ElasticHQ (ElasticHQ, 2018) que suporta as versões 2.x, 5.x, 6.x do Elasticsearch. O ElasticHQ gere e monitoriza todo o cluster do Elasticsearch e também é capaz de gerir múltiplos clusters de uma só vez (Daubner, 2018).

Uma das alternativas ao Kibana, é o Grafana que tem um plugin para o Elastic Stack, todavia a licença Enterprise fornece funcionalidades mais avançadas que a versão *open-source*.

Para garantir que não existem perdas de dados podem ser utilizadas ferramentas como o Redis ou o Kafka, para a agregação dos *logs* (Gupta & Gupta, 2017). O Kafka é orientado para gestão e armazenamento de uma grande quantidade de dados por um período prolongado de tempo o Redis é utilizado para a gestão de mensagens de curta duração e na qual a persistência não é necessária (Yigal, 2016).

É exequível complementar as tecnologias baseadas em anomalias (Suricata), em assinaturas (Wazuh) com o *machine learning* do Elastic Stack, facilitando a deteção de ameaças e a eficiência das investigações. O Wazuh é um HIDS *open-source*, permite analisar *logs*, verifica a integridade dos ficheiros, deteta *rootkits* e vulnerabilidades, faz avaliação da configuração e da resposta a incidentes. O Suricata é uma ferramenta de deteção de ameaças também gratuito com a capacidade de NIDS, NSM (Network Security Monitoring) e com processamento *pcap offline* (Bassett & Paquette, 2018).

Relativamente a projetos que podem ser implementados destacamos o HELK (Hunting ELK) que integra os principais componentes do Elastic Stack com outras soluções *open-source* como se pode visualizar no Diagrama VIII. A Principal funcionalidade do HELK é a investigação de ameaças (Threat Hunting) de dispositivos de rede ou ativos que apresentem anomalias. Embora este seja um projeto que está no estado *Alpha* pois ainda não foi testado em múltiplos senários que processem um grande volume de dados, contudo é possível ser implementado em produção pois é escalável para grandes volumes de dados (*Big Data*) tendo também a funcionalidade de *machine learning* através do Jupyter Notebook e do Apache Spark (Rodriguez, 2019).

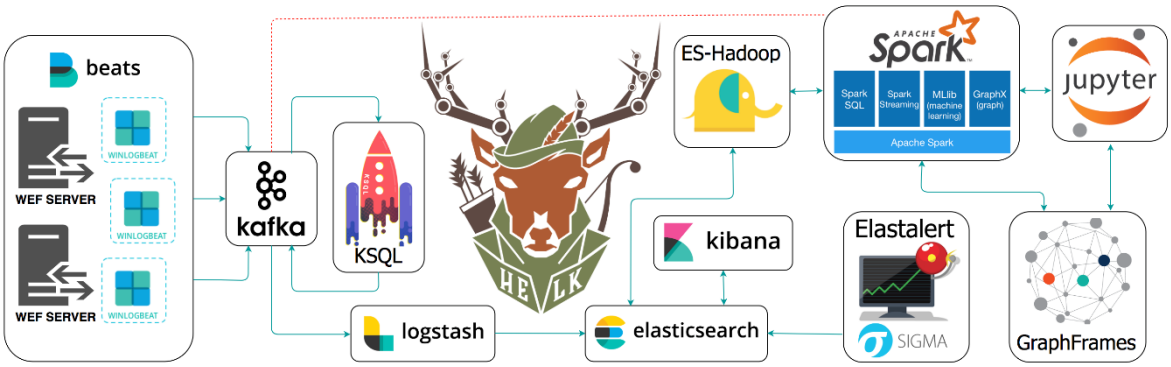


Diagrama VIII - Digrama da plataforma HELK (Rodriguez, 2019)

O ROCK (Response Operation Collection Kit) é uma plataforma de monitorização de segurança da rede NSM (Network Security Monitoring) *open-source* que inclui alguns componentes do Elastic Stack na sua Arquitetura como se pode visualizar na Diagrama IX.

Para além dos componentes do Elastic Stack recorre também a ferramentas como o Suricata, Bro, FSF (File Scanning Framework), Docket, Google Stenographe e o Kafka. Esta plataforma de monitorização de segurança de rede, agrega várias ferramentas *open-source* para facilitar a recolha de dados e a resposta a incidentes (RockNSM, 2019).

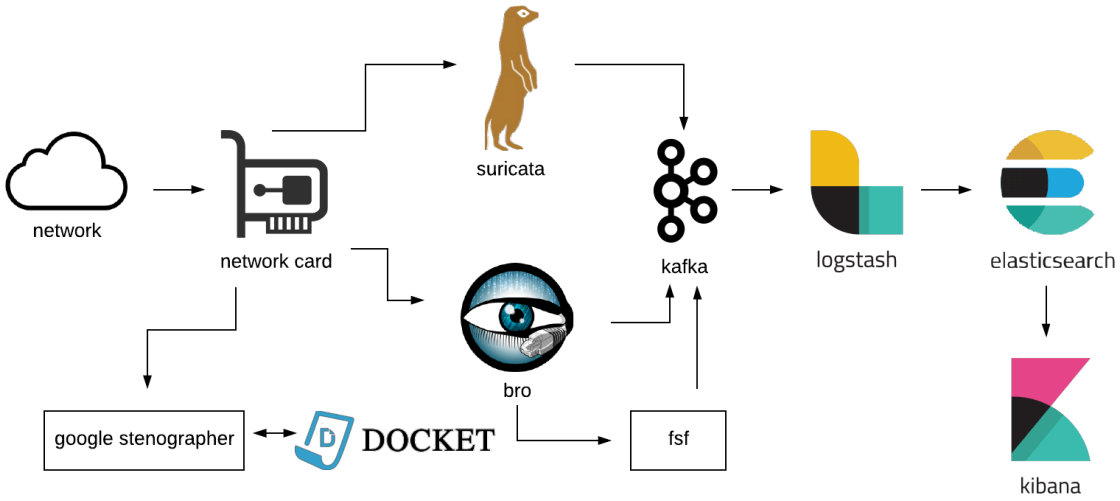


Diagrama IX - Digrama da plataforma ROCK (RockNSM, 2019)

Foram listadas várias alternativas *open-source*, todavia deve-se ter em consideração a compatibilidade com as últimas versões da solução e também se este tipo de versão cumpre com os requisitos pretendidos.

Anexo H – Elastic Stack encriptação

Alguns dos pré-requisitos elencados para o protótipo prende-se com a segurança dos dados, ou seja, a arquitetura proposta deve se garantir que os dados estão seguros em todas as suas fases. Por esta razão descreve-se o processo de encriptação para a solução Elastic Stack. Numa primeira fase encriptou-se as comunicações entre os nós do cluster, de seguida bloqueou-se o acesso não autorizado e finalmente aplicou-se a encriptação fora do cluster.

Encriptar as comunicações entre os nós

Neste ponto vamos ativar a segurança no Elasticsearch, configurar as passwords para os vários serviços. No Kibana vamos ativar autenticação e a criar vários perfis.

Para ativar a segurança no Elasticsearch desligamos o serviço do Elasticsearch editamos com o editor *vi* o ficheiro *elasticsearch.yml* e ativamos a segurança através do comando **xpack.security.enabled: true,** como se pode visualizar no exceto de código do ficheiro de configuração do Elasticsearch:

```
#desligar o serviço do elasticsearch
systemctl stop elasticsearch.service
```

```
# ---------------------------------- Various -----------------------------------
#
# Require explicit names when deleting indices:
#
#action.destructive_requires_name: true

xpack.security.enabled: true
```

Para guardar os certificados criou-se uma pasta especifica – **certs**

```
#criar a diretoria certs na pasta do elasticsearch
mkdir /etc/elastisearch/certs
```

O Elasticsearch fornece uma ferramenta que permite criar um certificado para os nós comunicarem entre si. Para criar o referido certificado devemos executar o seguinte comando:

```
#criação do certificado para permitir as comunicações encriptadas no cluster do Elasticsearch
cd /usr/share/elasticsearch
sudo bin/elasticsearch-certutil ca
```

```
informatica@elastic:/usr/share/elasticsearch$ sudo bin/elasticsearch-certutil ca
[sudo] password for informatica:
This tool assists you in the generation of X.509 certificates and certificate
signing requests for use with SSL/TLS in the Elastic stack.

The 'ca' mode generates a new 'certificate authority'
This will create a new X.509 certificate and private key that can be used
to sign certificate when running in 'cert' mode.

Use the 'ca-dn' option if you wish to configure the 'distinguished name'
of the certificate authority

By default the 'ca' mode produces a single PKCS#12 output file which holds:
    * The CA certificate
    * The CA's private key

If you elect to generate PEM format certificates (the -pem option), then the output will
be a zip file containing individual files for the CA certificate and private key

Please enter the desired output file [elastic-stack-ca.p12]:
Enter password for elastic-stack-ca.p12 :
informatica@elastic:/usr/share/elasticsearch$
```

Uma vez criado o certificado copiamos o mesmo para a diretoria **certs**.

```
#copiamos o certificado para a diretoria certs
cp elastic-certificates.p12 /etc/elasticsearch/certs
```

A documentação recomenda que se alterem as permissões para que o mesmo possa ser lido:

```
#alterar as permissões do certificado
cd /etc/elasticsearch/certs
chmod 640 elastic-certificates.p12
```

Uma vez realizadas anteriores voltamos a editar o ficheiro de configuração do Elasticsearch e adicionamos as seguintes linhas (identificadas a negrito):

```
# ---------------------------------- Various -----------------------------------
#
# Require explicit names when deleting indices:
#
#action.destructive_requires_name: true

xpack.security.enabled: true

xpack.security.transport.ssl.enabled: true
xpack.security.transport.ssl.verification_mode: certificate
xpack.security.transport.ssl.keystore.path: certs/elastic-certificates.p12
xpack.security.transport.ssl.truststore.path: certs/elastic-certificates.p12
```

De seguida voltou-se a ativar o serviço:

```
#ativar o serviço do elasticsearch
systemctl start elasticsearch.service
```

Bloquear os acessos não autorizados

Para bloquear os acessos não autorizados é necessário criar os utilizadores e as passwords do sistema para o Elastic Stack. Caso seja em produção deve-se utilizar passwords complicadas, no protótipo foi introduzida passwords fáceis de lembrar. O comando utilizado para criar as passwords foi:

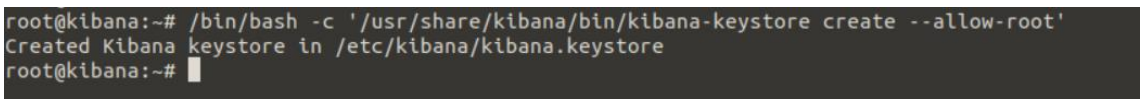
```
#para criar os utilizadores e as passwords do Elastic Stack executamos o seguinte comando
cd /usr/share/elasticsearch
bin/elasticsearch-setup-passwords interactive
```

Depois de se ativar a segurança foram necessárias realizar várias configurações para que o Kibana se possa ligar ao Elasticsearch. Primeiro desativamos o serviço do Kibana pois este deixou de conseguir comunicar com o Elasticsearch.

```
#desligar o serviço do kibana
systemctl stop kibana.service
```

Como as passwords dos serviços do Elastic Stack são sensíveis é recomendado que não se faça a gestão das mesmas através de níveis de permissões das pastas e ficheiros. Por este motivo o Kibana fornece uma ferramenta para gerir as credenciais do Elasticsearch: **kibana-keystore**. Exemplificamos a utilização do referido comando:

```
#criar um novo ficheiro keystore
su kibana -s /bin/bash -c '/usr/share/kibana/bin/kibana-keystore create'
```



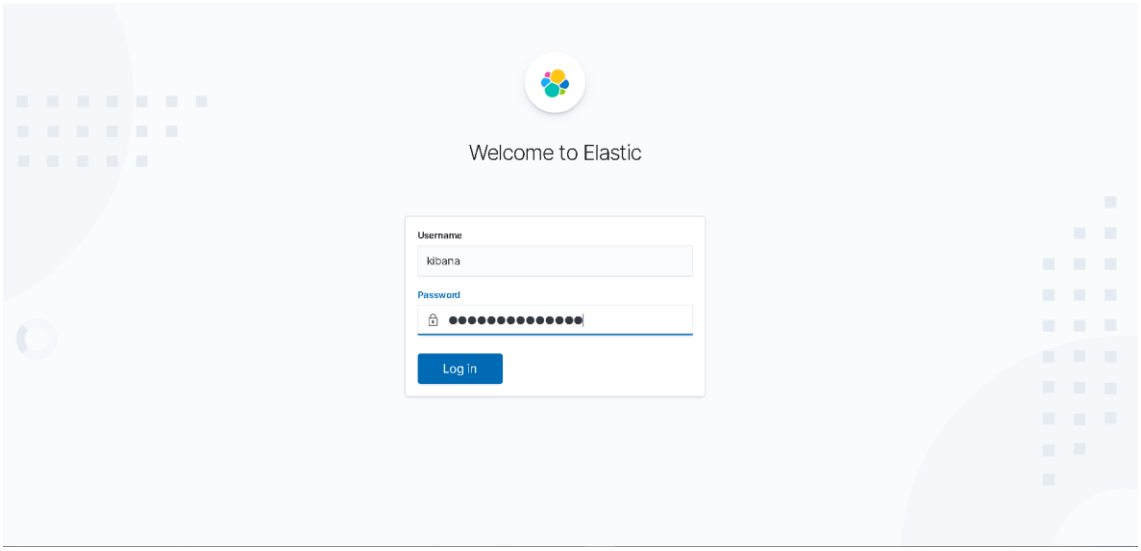

```
#criar o utilizador para o elasticsearch através do keystore
su kibana -s /bin/bash -c '/usr/share/kibana/bin/kibana-keystore add elasticsearch.username'

#criar a password para o elasticsearch através do keystore
su kibana -s /bin/bash -c '/usr/share/kibana/bin/kibana-keystore add elasticsearch.password'
```

```
root@kibana:~# /bin/bash -c '/usr/share/kibana/bin/kibana-keystore add elasticsearch.username --allow-root'
Enter value for elasticsearch.username: ******
root@kibana:~# /bin/bash -c '/usr/share/kibana/bin/kibana-keystore add elasticsearch.password --allow-root'
Enter value for elasticsearch.password: **************
root@kibana:~# /bin/bash -c '/usr/share/kibana/bin/kibana-keystore add elasticsearch.password --allow-root'
Setting elasticsearch.password already exists. Overwrite? [y/N] y
Enter value for elasticsearch.password: **************
root@kibana:~#
```

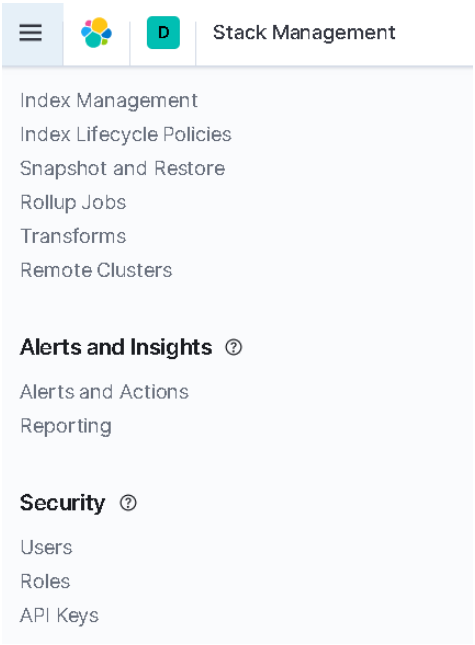
De seguida voltamos a ativar o serviço do Kibana.

```
1  #ativar o serviço do kibana
2  systemctl start kibana.service
```

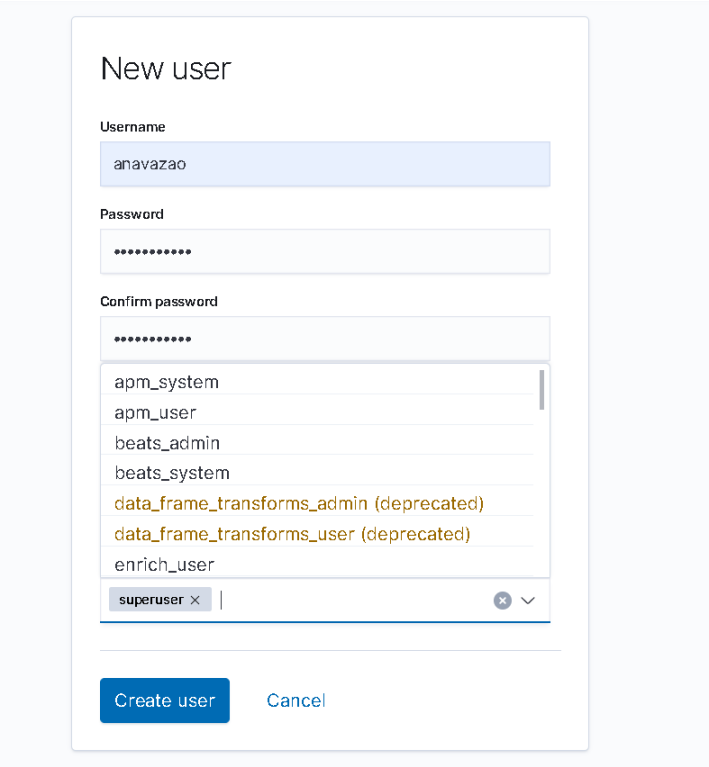
Ao ativar a segurança no Kibana, para aceder ao mesmo é necessário digitar a password que foi criada no ponto anterior, como se pode visualizar na imagem seguinte:

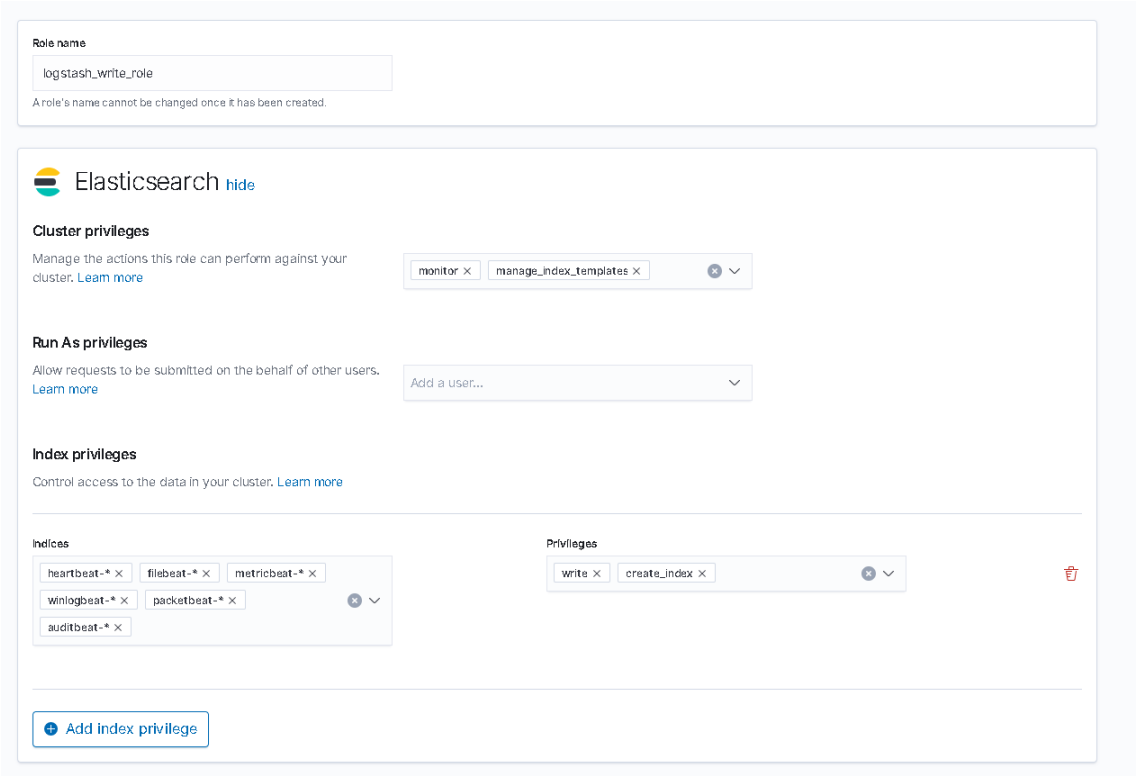


Depois de efetuadas as operações anteriores este já disponibiliza as opções segurança (***Security***), todavia é necessário aguardar alguns minutos, as mesmas não são imediatas.



A opção ***Security*** permite criar utilizadores com diferentes níveis de permissões. As imagens seguintes ilustram a criação de um utilizador e de uma nova regra.





Encriptação fora do cluster

Ativou-se a segurança nas comunicações HTTP para tal adicionou-se mais três linhas três linhas ao de configuração do Elasticsearch.

```
xpack.security.http.ssl.enabled: true
xpack.security.http.ssl.keystore.path: certs/elastic-certificates.p12
xpack.security.http.ssl.truststore.path: certs/elastic-certificates.p12
```

Para este ponto foi necessário criar um certificado para que os clientes se possam autenticarem no Elasticsearch.

```
1  #criar o certificado para os clientes se autenticarem no Elasticsearch
2  cd /usr/share/elasticsearch
3  sudo bin/elasticsearch-certutil ca --pem
```

```
root@elastic:/usr/share/elasticsearch# bin/elasticsearch-certutil ca --pem
This tool assists you in the generation of X.509 certificates and certificate
signing requests for use with SSL/TLS in the Elastic stack.

The 'ca' mode generates a new 'certificate authority'
This will create a new X.509 certificate and private key that can be used
to sign certificate when running in 'cert' mode.

Use the 'ca-dn' option if you wish to configure the 'distinguished name'
of the certificate authority

By default the 'ca' mode produces a single PKCS#12 output file which holds:
    * The CA certificate
    * The CA's private key

If you elect to generate PEM format certificates (the -pem option), then the output will
be a zip file containing individual files for the CA certificate and private key

Please enter the desired output file [elastic-stack-ca.zip]:
root@elastic:/usr/share/elasticsearch#
```

```
root@elastic:/usr/share/elasticsearch# ls -la
total 612
drwxr-xr-x   7 root root   4096 Oct  3 12:37 .
drwxr-xr-x 310 root root  12288 Sep 18 15:13 ..
drwxr-xr-x   2 root root   4096 Aug 20 15:38 bin
-rw-r--r--   1 root root   1401 Oct  2 21:58 client-ca.cer
-rw-r--r--   1 root root   1338 Oct  2 21:58 client.cer
-rw-r--r--   1 root root   1849 Oct  2 21:58 client.key
-rw-------   1 root root   3443 Oct  2 21:51 client.p12
-rw-------   1 root root   3443 Oct  2 20:58 elastic-certificates.p12
-rw-------   1 root root   2527 Oct  2 20:56 elastic-stack-ca.p12
-rw-------   1 root root   2510 Oct  3 12:37 elastic-stack-ca.zip
drwxr-xr-x   9 root root   4096 Aug 20 15:38 jdk
drwxr-xr-x   3 root root  12288 Aug 20 15:38 lib
drwxr-xr-x  52 root root   4096 Aug 20 15:38 modules
-rw-rw-r--   1 root root 544318 Aug 11 21:44 NOTICE.txt
drwxr-xr-x   2 root root   4096 Nov 26  2019 plugins
-rw-r--r--   1 root root   7007 Aug 11 21:43 README.asciidoc
root@elastic:/usr/share/elasticsearch#
```

O comando anterior cria um ficheiro comprimido som a extensão .zip e que contem no seu interior três ficheiros: chave privada, certificado público e o certificado assinado pela entidade certificadora (CA).

```
informatica@elastic:~/Downloads$ ls
client-ca.cer  client.cer  client.key  elastic-stack-ca.zip
informatica@elastic:~/Downloads$
```

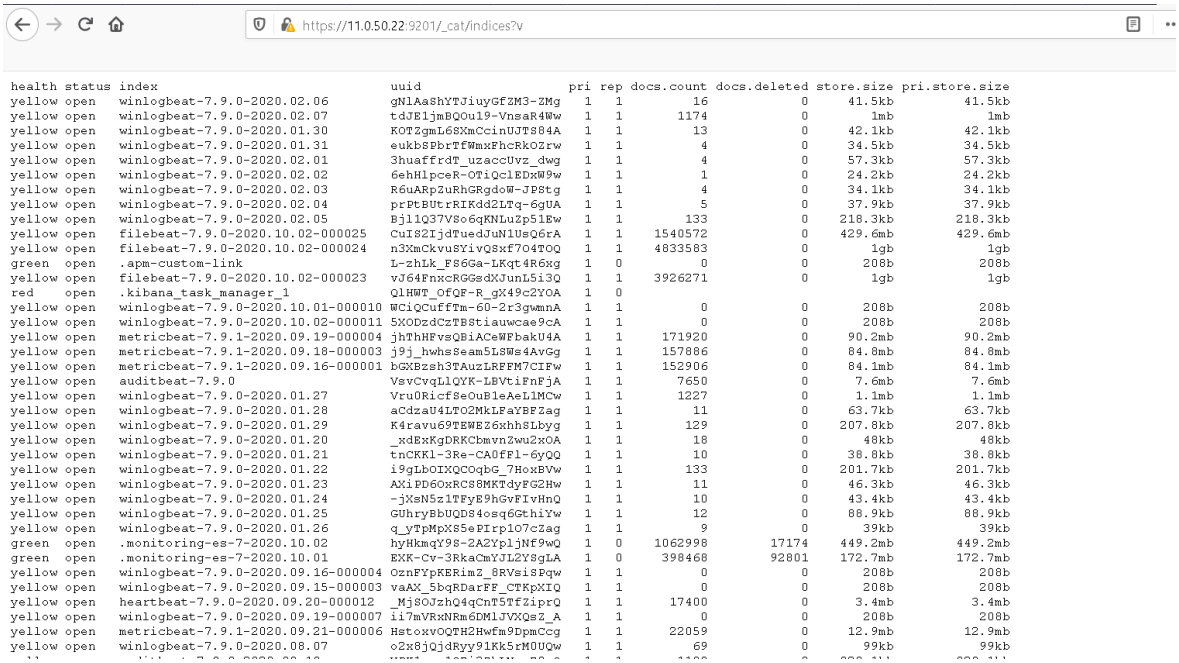
Após de se terem criado os ficheiros é necessário criar uma diretoria **certs** na pasta do Kibana e copiar os ficheiros criados para o interior da mesma.

```
1  #criar a diretoria certs na pasta do Kibana
2  cd /etc/kibana
3  mkdir certs
```

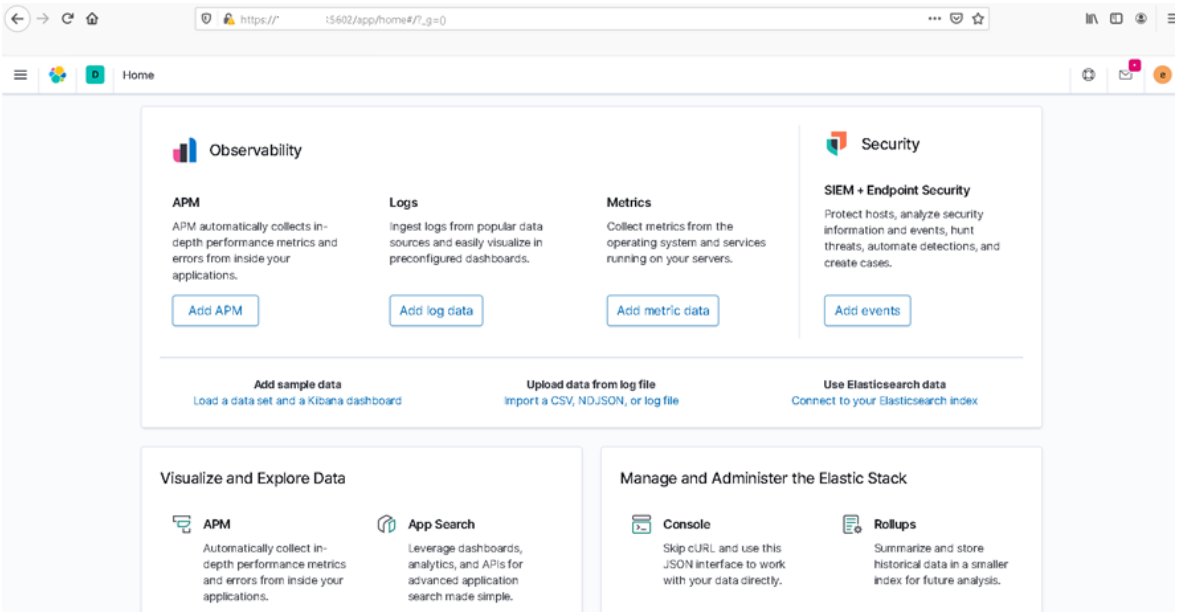
Também foi necessário adicionar algumas linhas ao ficheiro de configuração do Kibana (*kibana.yml*), as mesmas estão destacadas a negrito (foram acrescentadas no final das configurações):

```
……
# The URLs of the Elasticsearch instances to use for all your queries.
elasticsearch.hosts: ["https://IP_ELASTIC:9201"]
……
# Specifies locale to be used for all localizable strings, dates and number formats.
# Supported languages are the following: English - en , by default , Chinese - zh-CN .
#i18n.locale: "en"
xpack.security.enabled: true

elasticsearch.ssl.certificate: /etc/kibana/certs/client.cer
elasticsearch.ssl.key: /etc/kibana/certs/client.key
elasticsearch.ssl.certificateAuthorities: [ "/etc/kibana/certs/client-ca.cer" ]
elasticsearch.ssl.verificationMode: certificate

server.ssl.enabled: true
server.ssl.key: /etc/kibana/certs/client.key
server.ssl.certificate: /etc/kibana/certs/client.cer
```

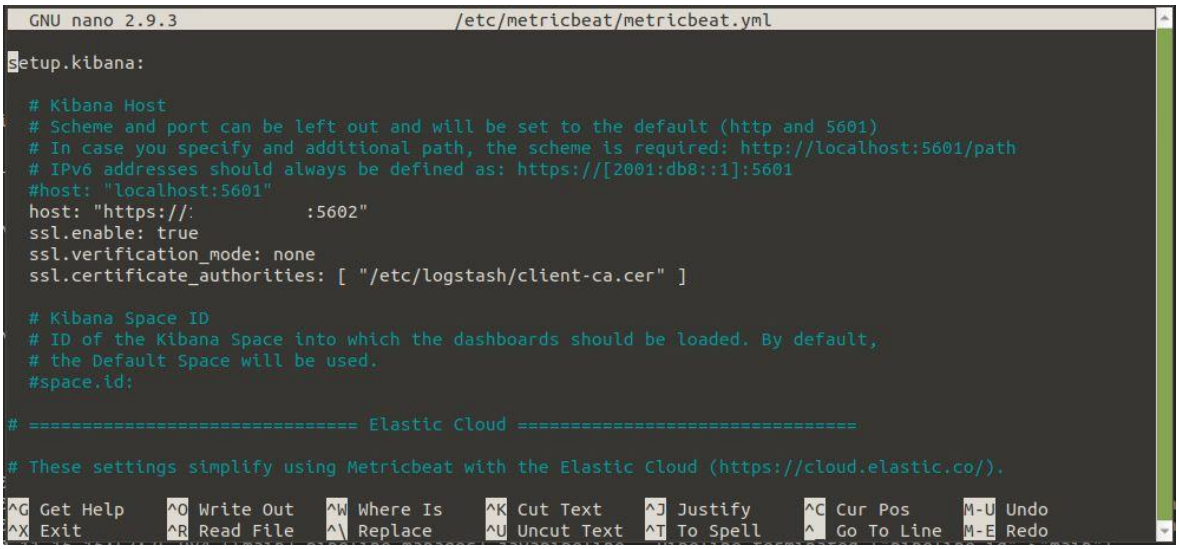
A partir deste momento para aceder aos índices necessário introduzir uma password e o protocolo utilizado é o https, como se pode visualizar na imagem seguinte.

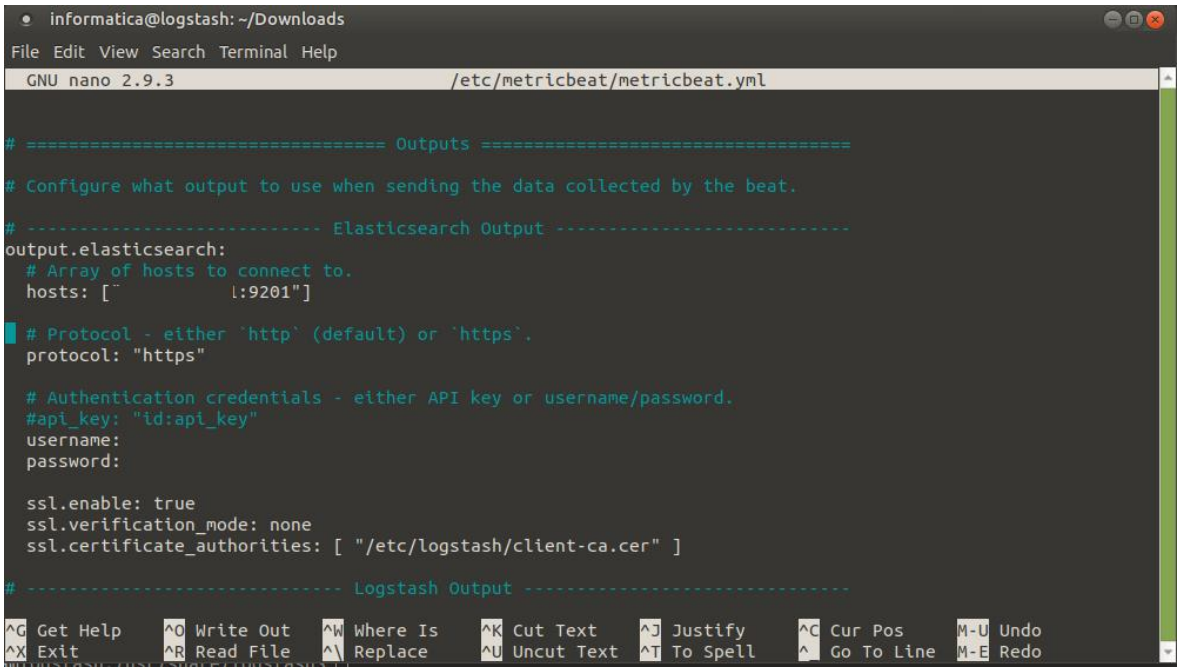


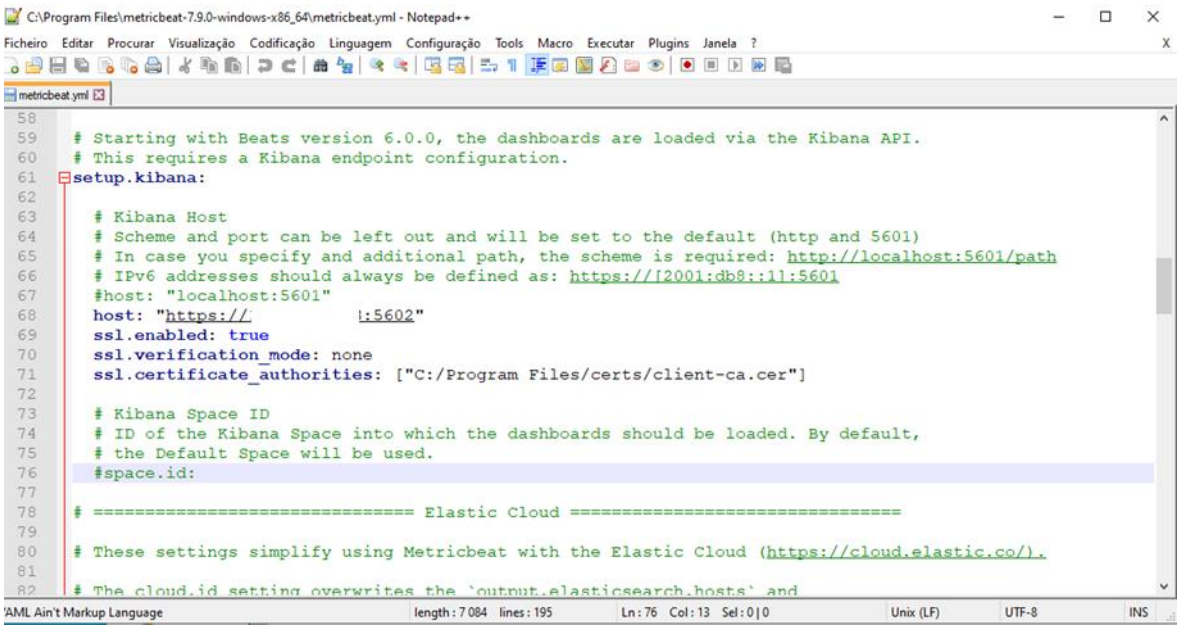
Da mesma forma o Kibana também já utiliza o protocolo https como está ilustrado na figura seguinte:



Os Beats e o Logstash para acederam ao Elasticsearch ou ao Kibana também se tem de autenticar. As configurações dos Beats são semelhantes no Microsoft Windows e no Linux. Tal como aconteceu com o Kibana tem de se criar uma diretoria e copiar os certificados gerados pelo Elasticsearch. Apresentamos as configurações no Linux e no Microsoft Windows.







No Logstash também se copiou o certificado, para as *piplenes* que vão enviar os dados para o Elasticsearch é necessário adicionar as linhas que estão a negrito ao filtro de saída.

```
elasticsearch {
        hosts => ["https://IP_ELASTIC:9201"]
        user => "elastic"
             password => "******"
        index => "%{[@metadata][beat]}-%{[@metadata][version]}-%{+yyyy.MM.dd}"
        ssl => true
        ssl_certificate_verification => false
        cacert => "/etc/logstash/client-ca.cer"
}
```

O processo de encriptar as comunicações foi bastante demorado, sendo necessário realizar as configurações várias vezes. Existe muita documentação para esta operação, contudo por vezes é difícil encontrar um fio condutor.

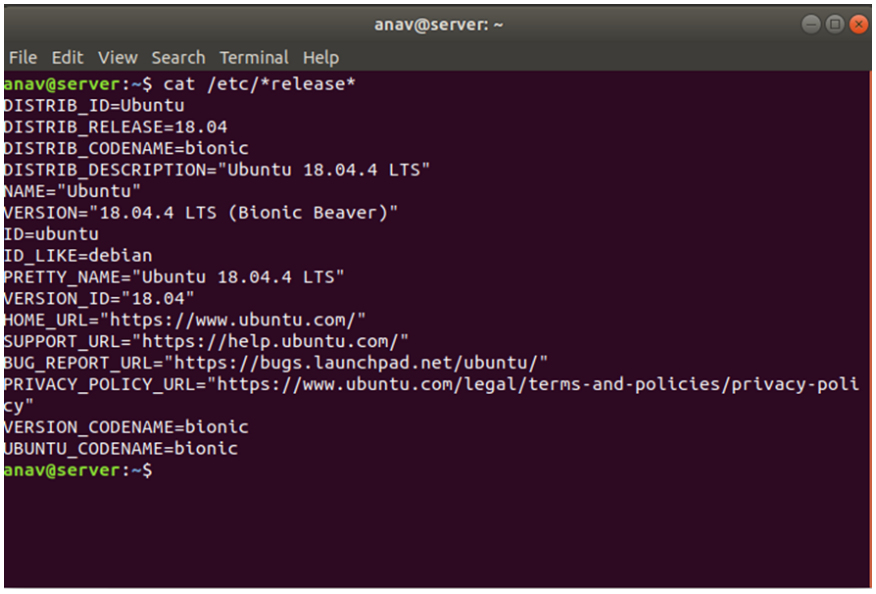
Anexo I – Manual de instalação Elastic Stack

Nas várias implementações da solução Elastic Stack utilizou-se sempre o sistema operativo Ubuntu Server Edition (várias versões) para os componentes Elasticsearch, Logstash e Kibana. Os Beats foram instalados nos sistemas operativos Ubuntu Server, Microsoft Windows 7 e Microsoft Windows 10.

Neste anexo pretende-se apresentar as configurações genéricas dos componentes Elasticsearch, Logstash e Kibana efetuadas em todas as implementações. Para essa finalidade utilizou-se uma Máquina Virtual com o sistema operativo Ubuntu 18.04.4 LTS Server, como se pode visualizar na imagem seguinte.

Os Beats foram instalados nos sistemas operativos Microsoft Windows e Linux e como as suas configurações são mais especificas optou-se por descrever a instalação nos dois sistemas operativos efetuadas no protótipo.

```
1  #Versão do sistema operativo
2  cat /etc/*release*
```



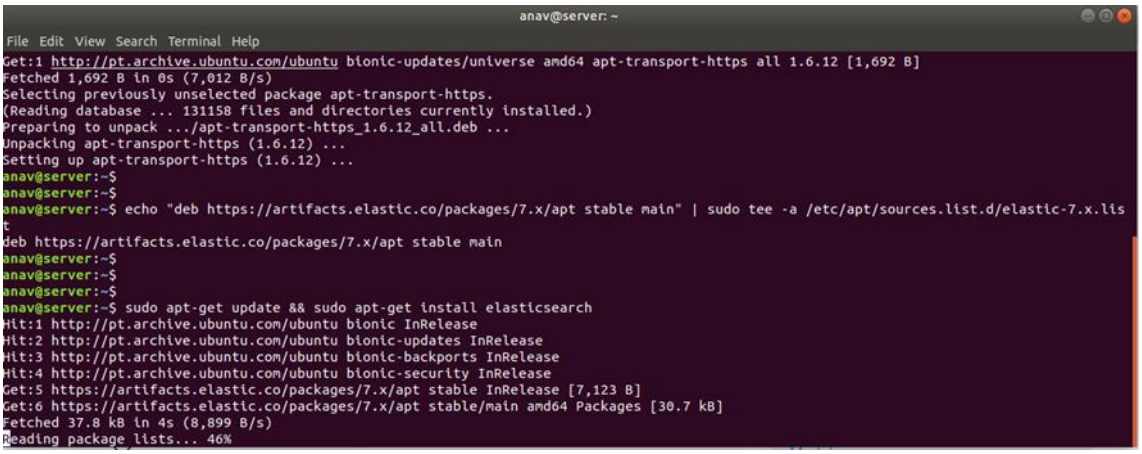
Devido às caraterísticas dos componentes normalmente a ordem de instalação é: Elasticsearch, Logstash, Kibana e por fim os Beats. Se os componentes Elasticsearch, Logstash e Kibana forem instalados na mesma máquina só é necessário executar uma vez os três próximos comandos. Se os componentes são instalados em máquinas diferentes, os mesmos devem ser executados antes da instalação do Elasticsearch, Logstash e Kibana.

```
# descarregar e instalar a chave pública
wget -qO - https://artifacts.elastic.co/GPG-KEY-elasticsearch | sudo apt-key add -

#instalar a package apt-transport-https
sudo apt-get install apt-transport-https

#guardar a definição do repositorio em: /etc/apt/sources.list.d/elastic-7.x.list:
echo "deb https://artifacts.elastic.co/packages/7.x/apt stable main" | sudo tee /etc/apt/sources.list.d/elastic-7.x.list
```

- **Elasticsearch instalação**

Para a instalação do Elasticsearch, atualizou-se o sistema operativo Linux e depois instalou-se o Elasticsearch. Salientamos que quando se instala o Elasticsearch é criado um cluster.

```
# instalação do elasticsearch
sudo apt-get update && sudo apt-get install Elasticsearch
```



Depois de instalado o Elasticsearch é necessário configurar o ficheiro *elasticsearch.yml*. O editor utilizado foi o ***vi***.

```
#Editar o ficheiro de configuração do elasticsearch
sudo vi /etc/elasticsearch/elasticsearch.yml
```

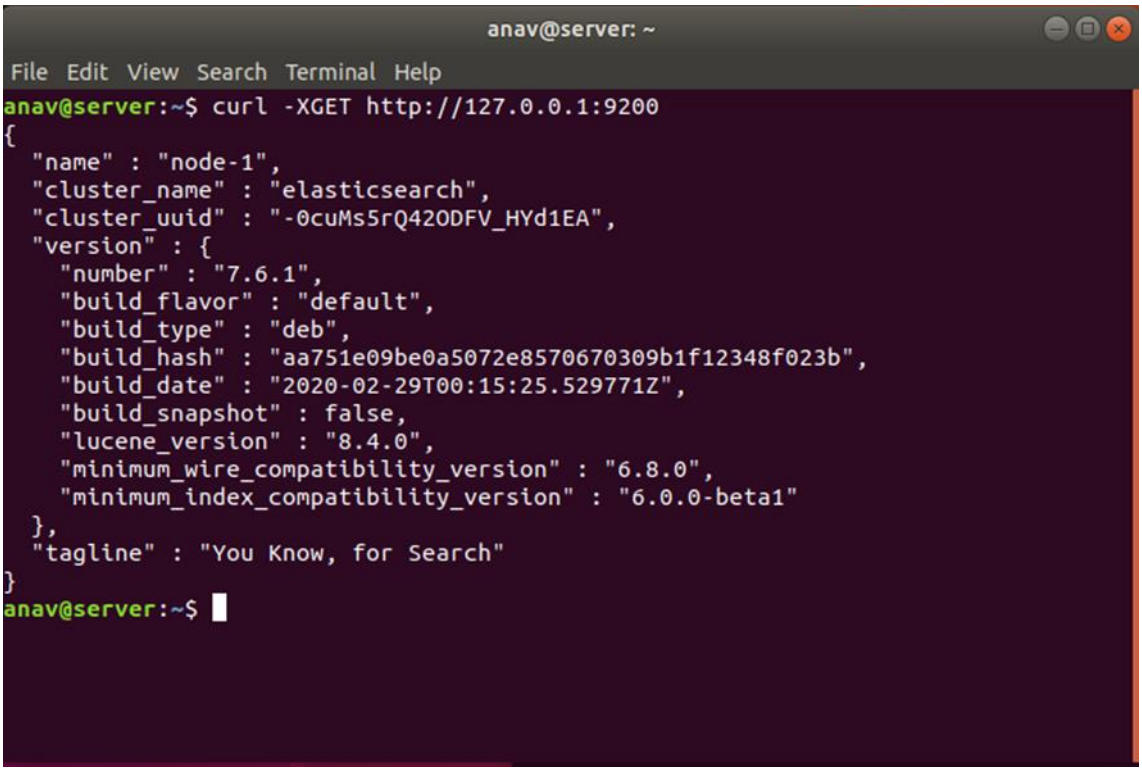
Existem várias configurações que podem ser realizadas, todavia as mais correntes são: alterar o IP (IP fixo máquina), alterar os caminhos do serviço, alterar o porto e atribuir um nome ao *cluster* e ao *node*. Nas implementações realizadas alterou-se o nome do cluster, do nó e colocou-se o IP fixo da máquina.

```
# ---------------------------------- Cluster -----------------------------------
#
# Use a descriptive name for your cluster:
#
cluster.name: my-application
#
# ------------------------------------ Node ------------------------------------
#
```

```
# Use a descriptive name for the node:
#
node.name: node-1
#
# Add custom attributes to the node:
#
#node.attr.rack: r1
#
……

# ---------------------------------- Network -----------------------------------
#
# Set the bind address to a specific IP (IPv4 or IPv6):
#
network.host: 127.0.0.1 (colocar IP da máquina)
#
# Set a custom port for HTTP:
#
http.port: 9200
#
# For more information, consult the network module documentation.
#
```

Depois de configurar o ficheiro *elasticsearch.yml* pode-se iniciar o serviço e verificar o estado do mesmo.

```
1  #comando para iniciar o serviço Elasticsearch
2  bin/elasticsearch
3
4  #comando para verificar o estado do Elasticsearch
5  curl -XGET http://127.0.0.1:9200
```



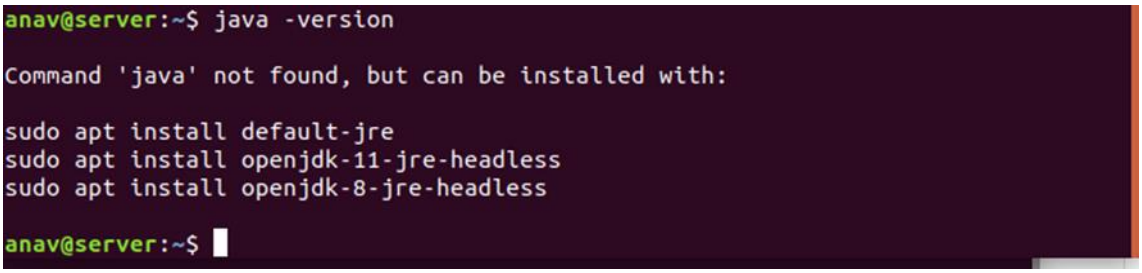
Através do *systemd* é possível configurar o Elasticsearch para iniciar o serviço automaticamente quando o sistema reiniciar/iniciar, para tal é necessário executar os comandos das linhas dois e três.

```
1  # iniciar o serviço automáticamente quando os sistema inicia
2  sudo /bin/systemctl daemon-reload
3  sudo /bin/systemctl enable elasticsearch.service
4
5  #iniciar o serviço
6  sudo systemctl start elasticsearch.service
7
8  #verificar o estado do serviço
9  sudo systemctl status elasticsearch.service
10
11 #parar o serviço
12 sudo systemctl stop elasticsearch.service
```

- **Logstash instalação**

O Logstash utiliza o java, por isso é necessário instalar uma das últimas versões do java, antes de se proceder à sua instalação. Na máquina virtual instalou-se a versão do java (*openjdk-11-jre-headless*) recomendada pela solução.

```
1  #comando que lista a versão do java instalado, caso não esteja instalado
2  #sugere que versões se deve instalar
3  Java -version
```




```
1  #instalou-se a seguinte versão do java
2  sudo apt install openjdk-11-jre-headless
```

Para a instalação do Logstash, atualizou-se o sistema operativo e depois instalou-se o Logstash.

```
#comando para atualizar o sistema operativo
sudo apt-get update

#comando para instalar o Logstash
sudo apt-get install logstash
```

```
anav@server:~$ sudo apt-get install logstash
Reading package lists... Done
Building dependency tree
Reading state information... Done
The following NEW packages will be installed:
  logstash
0 upgraded, 1 newly installed, 0 to remove and 3 not upgraded.
Need to get 174 MB of archives.
After this operation, 305 MB of additional disk space will be used.
Get:1 https://artifacts.elastic.co/packages/7.x/apt stable/main amd64 logstash all 1:7.6.1-1 [174 MB]
0% [1 logstash 1,072 kB/174 MB 1%]
```

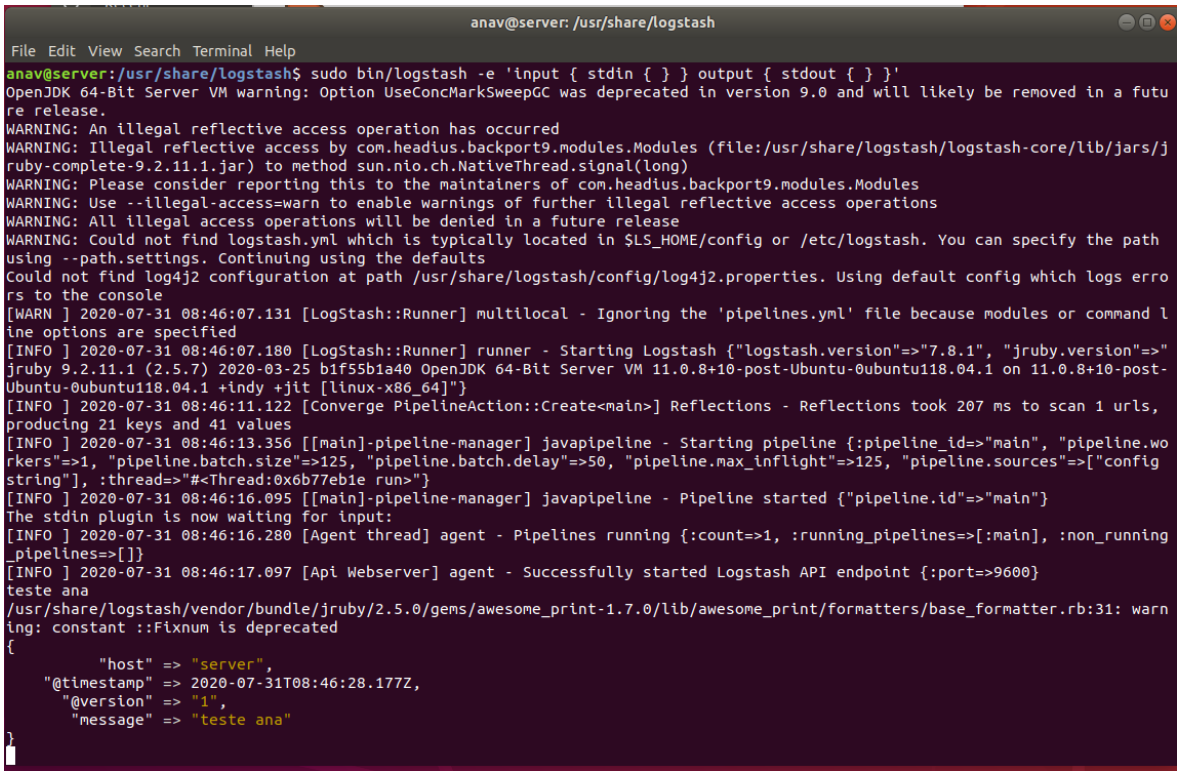
Depois de instalado o Logstash é necessário configurar o ficheiro *logstash.yml.* O editor utilizado foi o ***vi***. No ficheiro só foi alterado o nome do *node*.

```
#editar o fiheiro de configuração do logstash
sudo vi /etc/logstash/logstash.yml
```

```
# ------------  Node identity ------------
#
# Use a descriptive name for the node:
#
 node.name: nome_no
#
# If omitted the node name will default to the machine's host name
#
```

Depois de se instalar o Logstash pode-se testar o estado do serviço, através dos seguintes comandos.

```
#posicionar-se na diretoria onde se vai testar o seviço
cd /usr/share/logstash

#comando para testar o serviço do Logstash
bin/logstash -e 'input { stdin { } } output { stdout { } }'
```



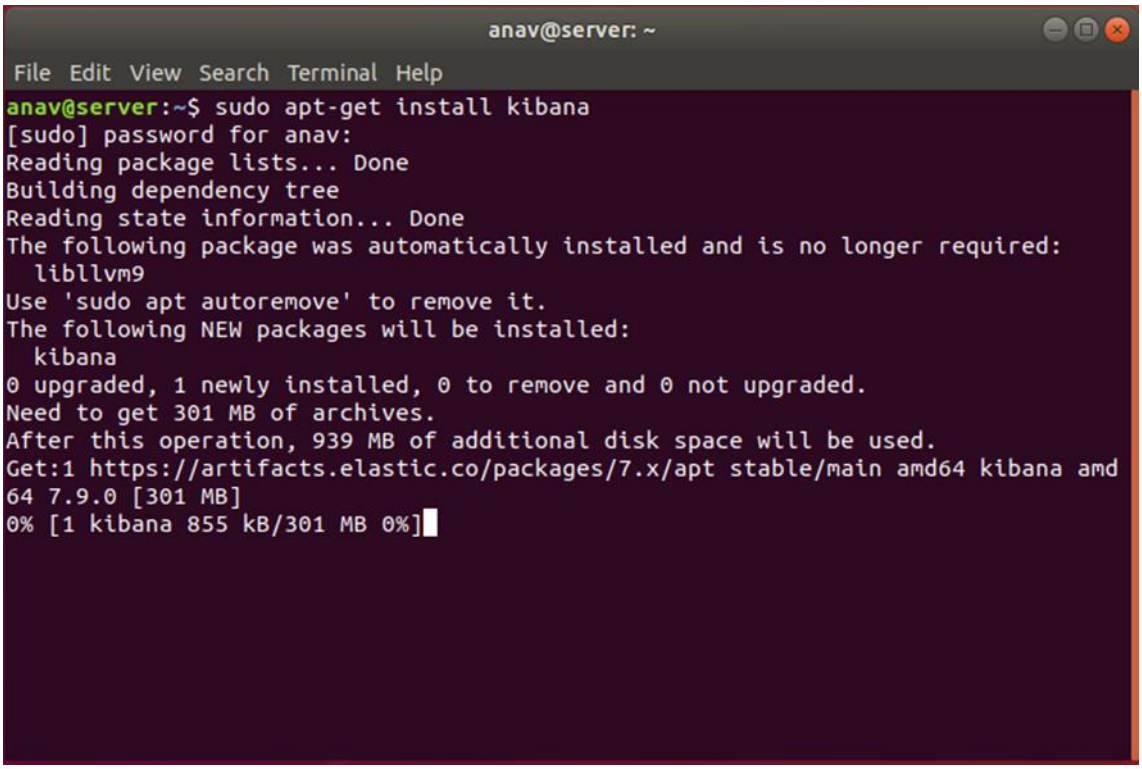
Também se configurou Logstash para iniciar o serviço automaticamente quando o sistema reiniciar/iniciar, para tal é necessário executar os seguintes comandos:

```
# iniciar o serviço automáticamente
sudo systemctl daemon-reload
sudo systemctl enable logstash.service
```

- **Kibana instalação**

Para a instalação do Kibana, atualizou-se o sistema operativo e depois instalou-se o Kibana.

```
# comando para atualizar o sistema operativo
sudo apt-get update

#comando para instalar o kibana
sudo apt-get install kibana
```



Depois de instalado o Kibana é necessário configurar o ficheiro *kibana.yml*. O editor utilizado foi o ***vi***. No ficheiro foram alterados os parâmetros a negrito.

```
1  #editar o fiheiro de configuração do kibana
2  sudo vi /etc/kibana/kibana.yml
```

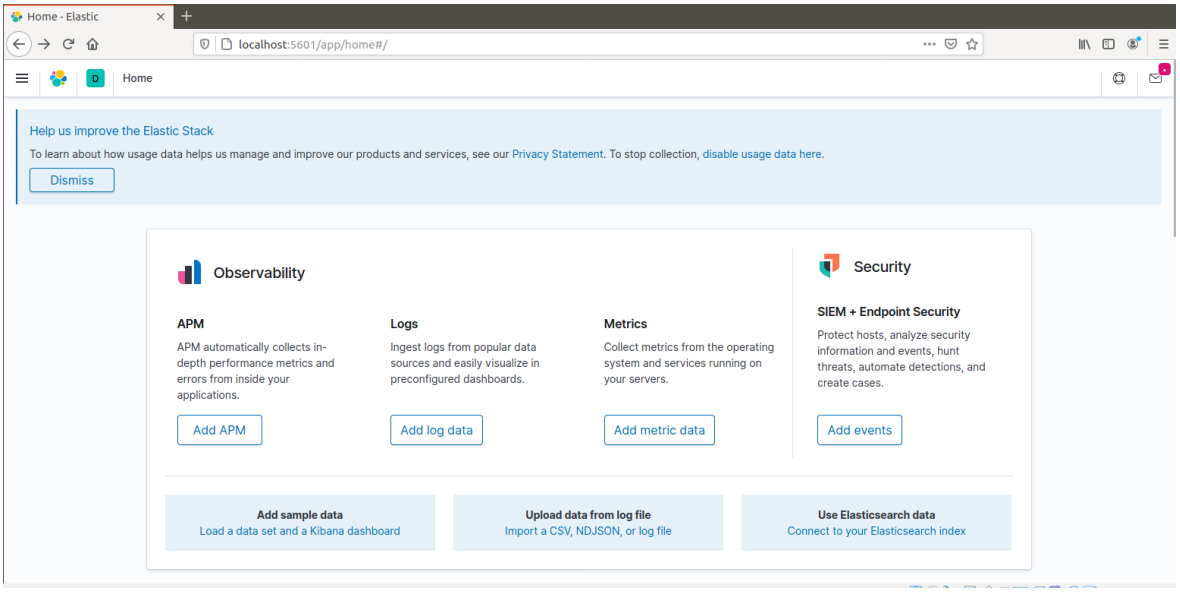
```
# Kibana is served by a back end server. This setting specifies the port to use.
server.port: 5601

# Specifies the address to which the Kibana server will bind. IP addresses and host names
are both valid values.
# The default is 'localhost', which usually means remote machines will not be able to connect.
# To allow connections from remote users, set this parameter to a non-loopback address.
server.host: "IP_MAQUINA_KIBANA"

# Enables you to specify a path to mount Kibana at if you are running behind a proxy.
# Use the `server.rewriteBasePath` setting to tell Kibana if it should remove the basePath
# from requests it receives, and to prevent a deprecation warning at startup.
# This setting cannot end in a slash.
#server.basePath: ""

# Specifies whether Kibana should rewrite requests that are prefixed with
# `server.basePath` or require that they are rewritten by your reverse proxy.
# This setting was effectively always `false` before Kibana 6.3 and will
# default to `true` starting in Kibana 7.0.
#server.rewriteBasePath: false


# The maximum payload size in bytes for incoming server requests.
#server.maxPayloadBytes: 1048576

# The Kibana server's name.  This is used for display purposes.
server.name: "your-hostname"

# The URLs of the Elasticsearch instances to use for all your queries.
elasticsearch.hosts: ["http://IP_MAQUINA_ELASTIC:9200"]
```

Também se configurou Kibana para iniciar o serviço automaticamente quando o sistema reiniciar/iniciar, para tal é necessário executar os seguintes comandos:

```
#iniciar o serviço automáticamente
sudo systemctl daemon-reload
sudo systemctl enable kibana.service
```

Para verificar se o serviço está a funcionar podemos verificar o seu estado, ou aceder ao endereço: **http://localhost:5601/** (nas implementações efetuadas o *localhost* foi substituído pelo IP fixo da máquina onde está instalado o Kibana)

```
anav@server:~$ sudo systemctl start kibana.service
anav@server:~$ sudo systemctl status kibana.service
● kibana.service - Kibana
   Loaded: loaded (/etc/systemd/system/kibana.service; enabled; vendor preset: e
   Active: active (running) since Mon 2020-08-31 20:53:33 UTC; 10s ago
 Main PID: 28902 (node)
    Tasks: 7 (limit: 4655)
   CGroup: /system.slice/kibana.service
           └─28902 /usr/share/kibana/bin/../node/bin/node /usr/share/kibana/bin/

Aug 31 20:53:33 server systemd[1]: Started Kibana.
lines 1-9/9 (END)
```



- **Beats instalação**

Tendo como base a Figura 4-2 neste ponto vamos descrever o processo de instalação dos Beats nos sistemas operativos Microsoft Windows e Linux (Ubuntu Server Edition). Devido à sua complexidade e especificidade recorreu-se ao cenário do protótipo para a criação do manual.

Para todos os sistemas operativos foi necessário efetuar o Download do Beat, configurar e iniciar o seu serviço. Na realização do presente trabalho foi necessário instalar várias versões dos Beats, todavia como o processo de instalação é o mesmo, optou-se por descrever a instalação dos beats na versão 7.5.

☐ Microsoft Windows

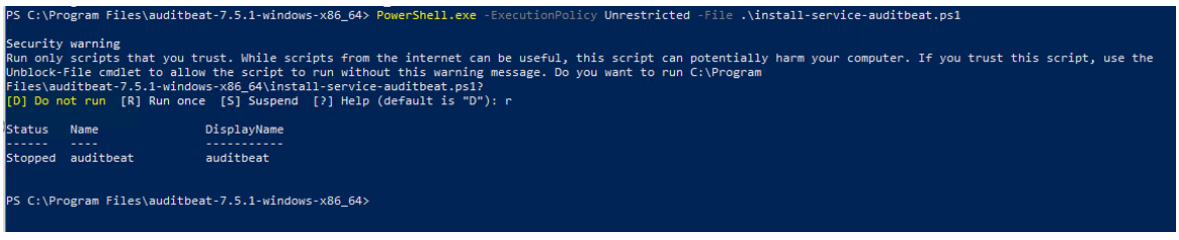
Neste ponto vamos descrever a instalação dos beats (Auditbeat, Metricbeat, Packetbeat e o Winlogbeat) no sistema operativo Microsoft Windows. Para todos os beats descarregou-se a última versão do ficheiro do beat que se pretendia instalar na página do Elastic. De seguida descomprimiu-se o ficheiro e copiou-se a pasta já descomprimida para ***C:\'Program Files'***. Para instalar os Beats no sistema operativo Microsoft Windows recorreu-se ao ***PowerShell***.

Nas configurações de cada beat deve-se sempre introduzir o IP do Kibana e o IP do Logstash ou do Elasticsearch, no caso específico das máquinas clientes foi introduzido o IP do Logstash. Também se configurou para todos os beats as *tags* identificativas do sistema operativo.

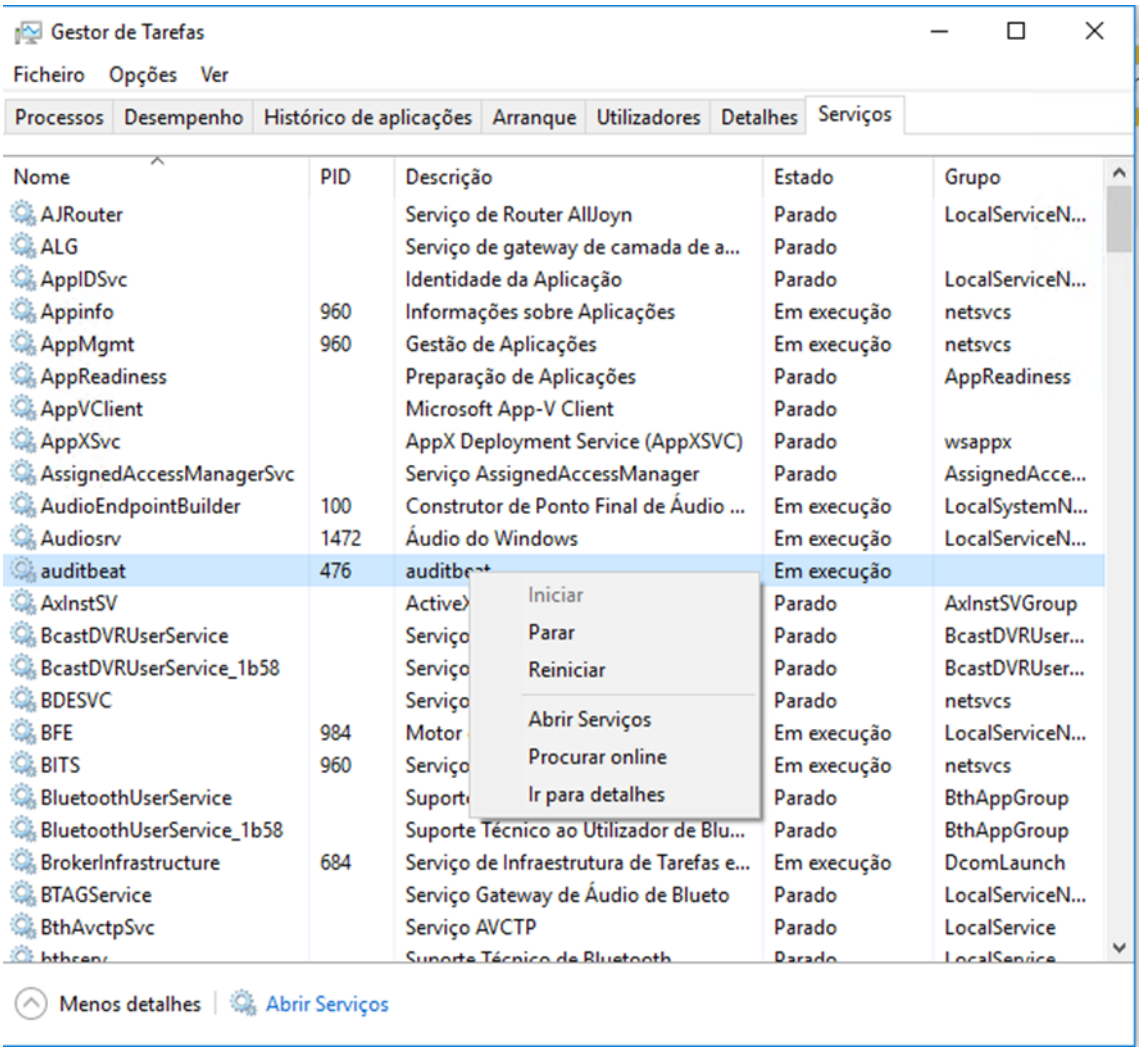
✓ Auditbeat

Após de se ter copiado a pasta descomprimida do Auditbeat para a pasta "***Program Files***", através do ***PowerShell*** executámos o seguinte comando para instalar o Auditbeat.

```
1  #instalar o serviço auditbeat
2  powershell.exe -executionpolicy bypass -file .\install-service-auditbeat.ps1
```



Depois de instalar o Auditbeat já se pode visualizar o serviço no Gestor de Tarefas.



Antes de se iniciar o serviço deve-se configurar o ficheiro ***auditbeat.yml***. O Auditbeat permite definir no mesmo as diretorias que se pretende verificar a integridade dos ficheiros, também vai recolher dados sobre a máquina e os seus processos.

É possível testar as configurações do Auditbeat executamos o seguinte comando:

```
#testar as configurações do auditbeat
auditbeat.exe test config
```

Pode-se iniciar o serviço no Gestor de Tarefas ou através da linha de comandos:

```
#iniciar o serviço auditbeat
Start-Service auditbeat
```

- ✓ **Metricbeat**

Para instalar o Metricbeat, também se copiou a pasta descomprimida para a pasta "***Program Files***" e instalou-se o serviço através do ***PowerShell***.

```
#instalar o serviço metricbeat
powershell.exe -executionpolicy bypass -file .\install-service-metricbeat.ps1
```

```
PS C:\Program Files\metricbeat-7.5.1-windows-x86_64> powershell.exe -executionpolicy bypass -file .\install-service-metricbeat.ps1

Status   Name               DisplayName
------   ----               -----------
Stopped  metricbeat         metricbeat


PS C:\Program Files\metricbeat-7.5.1-windows-x86_64> Start-Service metricbeat
```

O Metricbeat permite ativar vários módulos, para que fosse possível recolher as métricas do sistema operativo Microsoft Windows e do sistema ativou-se os módulos: sistema e *windows*. Para tal foi necessário executar os seguintes comandos:

```
#ativar o modulo windows no metricbeat
metricbeat.exe modules enable windows

#activar o módulo system no metricbeat
metricbeat.exe modules enable system
```

É possível verificar os módulos que estão ativos e quais os módulos a ativar, para tal executamos o comando:

```
#visualizar os módulos ativos e os que que são disponibilizados pelo metricbeat
metricbeat.exe modules list
```

```
C:\Program Files\metricbeat-7.5.1-windows-x86_64>metricbeat.exe modules list
Enabled:
system
windows

Disabled:
aerospike
apache
appsearch
aws
azure
beat
beat-xpack
ceph
cockroachdb
consul
coredns
couchbase
couchdb
docker
dropwizard
elasticsearch
elasticsearch-xpack
envoyproxy
etcd
golang
graphite
haproxy
http
jolokia
kafka
kibana
kibana-xpack
kubernetes
kvm
logstash
```

Posteriormente deve-se configurar os ficheiros ***metricbeat.yml***, ***windows.yml*** e ***system.yml***. No módulo *system* foram acrescentadas algumas configurações para conseguir métricas mais detalhadas, no mesmo também se ativou alguns módulos. Listam-se de seguidas as configurações do mesmo:

```
# Module: system
#Docs: https://www.elastic.co/guide/en/beats/metricbeat/7.5/metricbeat-module-system.html

- module: system
  period: 10s
  metricsets:
    - cpu
    #- load
    - memory
    - network
    - process
    - process_summary
    - socket_summary
    #- entropy
    #- core
    - diskio
    #- socket
  processes: ['.*']
  process.include_top_n:
    by_cpu: 5      # include top 5 processes by CPU    by_memory: 5
# include top 5 processes by memory
# Configure the metric types that are included by these metricsets.
  cpu.metrics:  ["percentages", "normalized_percentages", "ticks"]
# The other available options are normalized_percentages and ticks.
  core.metrics: ["percentages"]  # The other available option is ticks.

- module: system
  period: 1m
  metricsets:
    - filesystem
    - fsstat
  processors:
  - drop_event.when.regexp:
      system.filesystem.mount point: '^/(sys|cgroup|proc|dev|etc|host|lib)($|/)'

- module: system
  period: 15m
  metricsets:
    - uptime
```

Também é possível testar as configurações e iniciar o serviço com os seguintes comandos:

```
#testar as configurações do metricbeat
metricbeat.exe test config

#iniciar o serviço metricbeat
Start-Service metricbeat
```

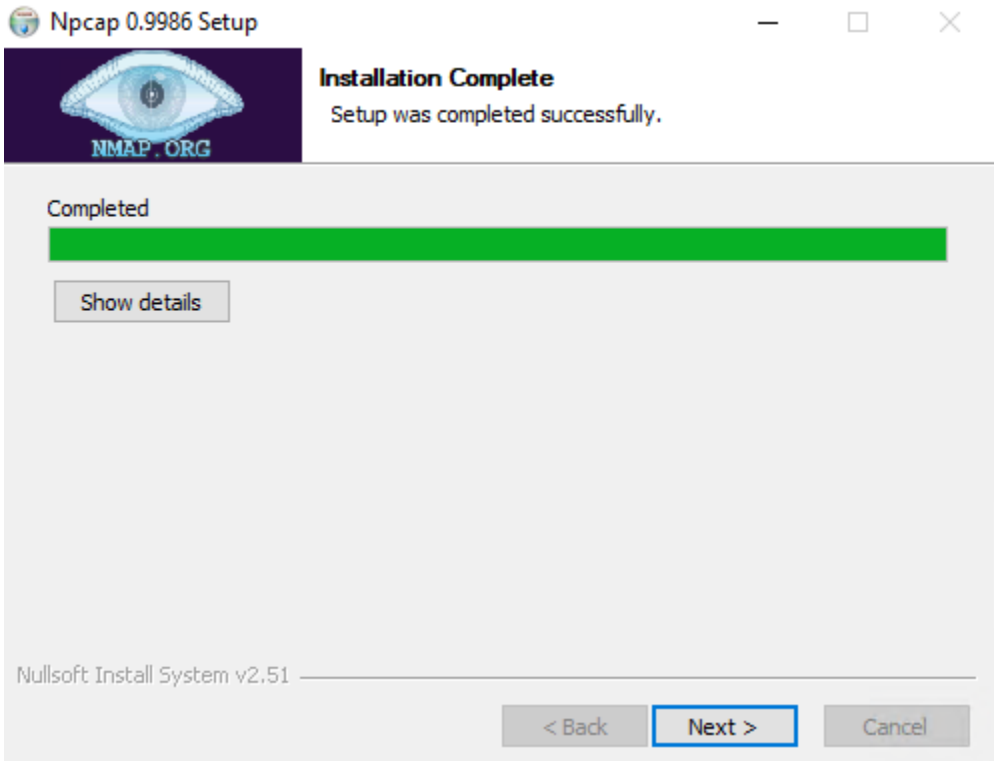
- ✓ **Packetbeat**

No Packetbeat identicamente também se fez o download, descomprimiu e copiou-se a pasta descomprimida para a pasta "***Program Files***" e instalou-se o serviço através do ***PowerShell***.

```
#instalar o serviço packetbeat
powershell.exe -executionpolicy bypass -file .\install-service-packetbeat.ps1
```

```
PS C:\Program Files\packetbeat-7.5.2-windows-x86_64> PowerShell.exe -ExecutionPolicy UnRestricted -File .\install-service-packetbeat.ps1

Security warning
Run only scripts that you trust. While scripts from the internet can be useful, this script can potentially harm your computer. If you trust this script,
you want to run C:\Program Files\packetbeat-7.5.2-windows-x86_64\install-service-packetbeat.ps1?
[D] Do not run  [R] Run once  [S] Suspend  [?] Help (default is "D"): r

Status   Name               DisplayName
------   ----               -----------
Stopped  packetbeat         packetbeat


PS C:\Program Files\packetbeat-7.5.2-windows-x86_64>
```

O beat Packetbeat para funcionar no Microsoft Windows necessita que se instale o Npcap[64], instalou-se na máquina de teste a versão ***npcap-0.9986***.



O Packetbeat fornece um comando para listar as interfaces de rede, uma vez que obriga a definir no ficheiro de configuração as interfaces que se pretende recolher dados. Para tal executamos o seguinte comando:

```
#listar as interfaces de rede
packetbeat.exe devices
```

[64] https://nmap.org/npcap/

```
C:\Program Files\packetbeat-7.5.2-windows-x86_64>packetbeat.exe devices
0: \Device\NPF_{                                } (NdisWan Adapter) (Not assigned ip address)
1: \Device\NPF_{                                } (NdisWan Adapter) (Not assigned ip address)
2: \Device\NPF_{                                } (NdisWan Adapter) (Not assigned ip address)
3: \Device\NPF_{                                } (Intel(R) PRO/1000 MT Desktop Adapter) (fe80::
        )
4: \Device\NPF_Loopback (Adapter for loopback traffic capture) (Not assigned ip address)

C:\Program Files\packetbeat-7.5.2-windows-x86_64>
```

A imagem anterior lista todas as interfaces, no presente trabalho selecionámos a interface 3. Pode-se visualizar de seguida a configuração da interface de rede.

```
#============== Network device ================================

# Select the network interface to sniff the data. On Linux, you can use the
# "any" keyword to sniff on all connected interfaces.
packetbeat.interfaces.device: 3

#==================== Flows ===================================

# Set `enabled: false` or comment out all options to disable flows reporting.
packetbeat.flows:
  # Set network flow timeout. Flow is killed if no packet is received before being
  # timed out.
  timeout: 30s

  # Configure reporting period. If set to -1, only killed flows will be reported
  period: 10s
```

Como todos os beats anteriores, é possível testar a configuração do ficheiro e iniciar o serviço através de comandos, como se pode visualizar na imagem seguinte.

```
1  #testar as configurações do packetbeat
2  metricbeat.exe test config
3
4  #iniciar o serviço packetbeat
5  Start-Service packetbeat
```

- ✓ **Winlogbeat**

Identicamente também se recorreu ao ***PowerShell*** para instalar o Winlogbeat, após se ter copiado a pasta descomprimida para a pasta "**Program Files**".

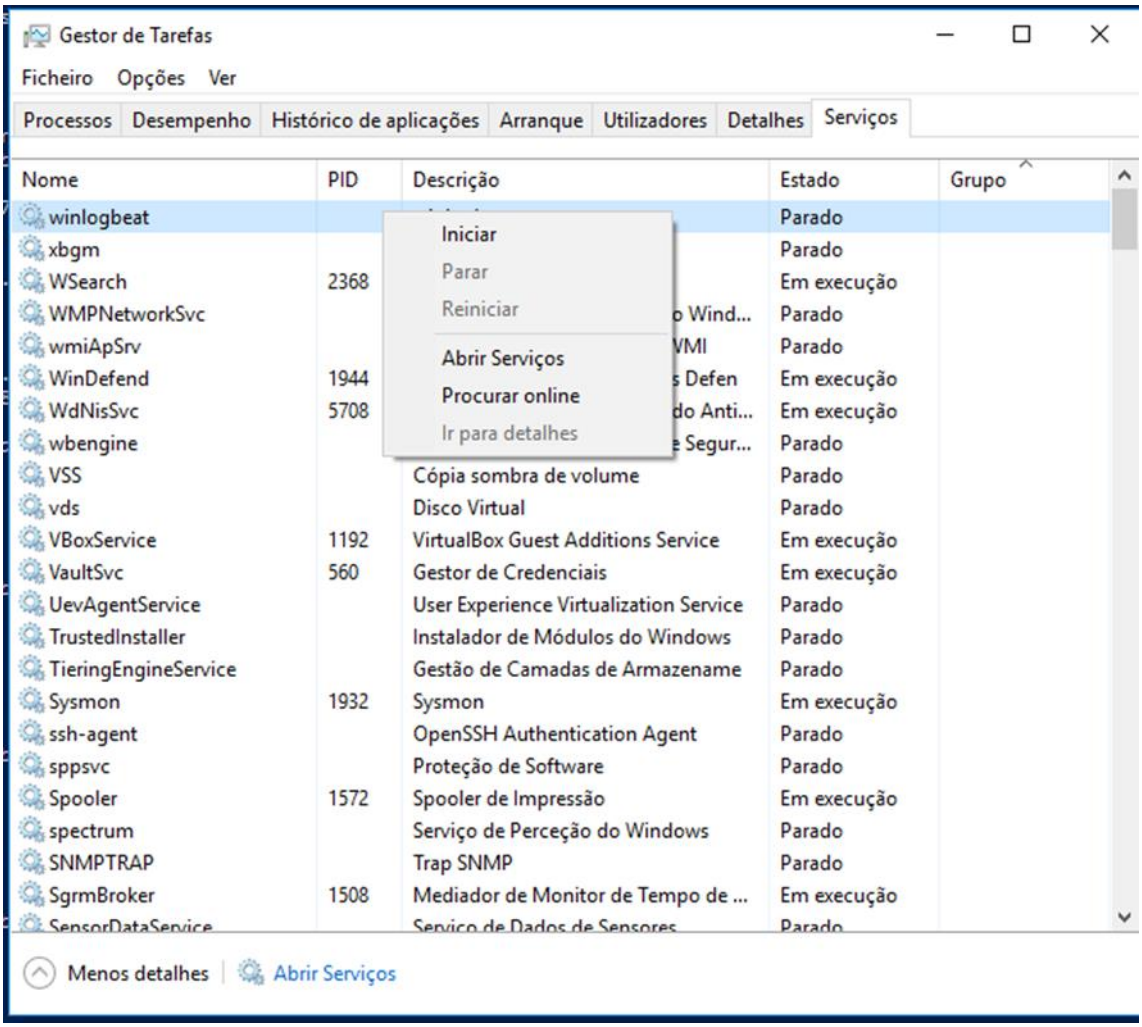
```
1  #instalar o serviço winlogbeat
2  powershell.exe -executionpolicy bypass -file .\install-service-winlogbeat.ps1
```

```
PS C:\Program Files\winlogbeat-7.5.1-windows-x86_64> powershell.exe -executionpolicy bypass -file .\install-service-winlogbeat.ps1

Status   Name               DisplayName
------   ----               -----------
Stopped  winlogbeat         winlogbeat


PS C:\Program Files\winlogbeat-7.5.1-windows-x86_64>
```

Depois de se instalar o serviço deve-se configurar o ficheiro ***winlogbeat.yml***. No mesmo definiu-se o tipo de *logs* que se pretende recolher. É possível testar as configurações do Winlogbeat executamos o seguinte comando:

```
#testar as configurações do winlogbeat
winlogbeat.exe test config
```

Depois de instalar o Winlogbeat já se pode visualizar o serviço no Gestor de Tarefas.



O serviço Winlogbeat pode ser iniciado através do Gestor de Tarefas ou através do seguinte comando:

```
#iniciar o serviço winlogbeat
Start-Service winlogbeat
```

☐ Linux (Ubuntu Server Edition)

Neste ponto vamos descrever a instalação dos beats (Filebeat, Heartbeat, Auditbeat, Metricbeat e o Packetbeat) no sistema operativo Linux (Ubuntu Server Edition) .

As máquinas que possuem o sistema operativo Linux instalado são servidores, logo não se envia os *logs* para o Logstash (neste momento não existem utilizadores a aceder às mesmas só os administradores), mas sim diretamente para o Elasticsearch. Nos beats

instalados no sistema operativo Microsoft Windows configurou-se o endereço IP do Logstash, aqui vamos configurar o IP do Elasticsearch.

Nas máquinas com o sistema operativo Linux só é necessário executar os próximos comandos uma vez em cada servidor.

```
# descarregar e instalar a chave pública
wget -qO - https://artifacts.elastic.co/GPG-KEY-elasticsearch | sudo apt-key add -

#instalar a package apt-transport-https
sudo apt-get install apt-transport-https

#guardar a definição do repositorio em: /etc/apt/sources.list.d/elastic-7.x.list:
echo "deb https://artifacts.elastic.co/packages/7.x/apt stable main" | sudo tee /etc/apt/sources.list.d/elastic-7.x.list
```

Para iniciar, parar ou verificar o estado dos beats podemos utilizar o comando ***systemctl*** e o nome do ***beat*** que enumeramos: auditbeat, heartbeat-elastic, filebeat, Metricbeat, packetbeat

```
#iniciar o auditbeat automáticamente
sudo systemctl enable nomebeat

#iniciar o serviço
sudo systemctl start nomebeat

#parar o serviço
sudo systemctl stop nomebeat

#verificar o estado do serviço
sudo systemctl status nomebeat
```

- ✓ **Filebeat**

No Linux o processo de instalação é muito simples, basta executar o seguinte comando:

```
#instalar o filebeat
sudo apt-get install filebeat
```

```
:~$ sudo apt-get update && sudo apt-get install filebeat
Hit:1 http://ppa.launchpad.net/yannubuntu/boot-repair/ubuntu bionic InRelease
Hit:2 http://archive.ubuntu.com/ubuntu bionic InRelease
Hit:3 http://archive.ubuntu.com/ubuntu bionic-updates InRelease
Get:4 http://archive.ubuntu.com/ubuntu bionic-backports InRelease [74.6 kB]
Get:5 http://security.ubuntu.com/ubuntu bionic-security InRelease [88.7 kB]
0% [Connected to artifacts.elastic.co (151.101.194.222)]
```

Para iniciar automaticamente o serviço executamos o seguinte comando:

```
#iniciar o filebeat automáticamente
sudo systemctl enable filebeat
```

Depois de instalado é necessário editar o ficheiro ***filebeat.yml***. Nas configurações do Filebeat é necessário definir quais os ficheiros de *logs* que vamos enviar para o Elasticsearch.

```
#editar o ficheiro de configuração do filebeat
sudo vi /etc/filebeat/filebeat.yml
```

Para testar as configurações do Filebeat executa-se o seguinte comando:

```
#para testar o ficheiro de configuração do filebeat
filebeat test config
```

Caso se pretenda iniciar o serviço basta executar o seguinte comando:

```
#iniciar o serviço filebeat
sudo systemctl start filebeat
```

- ✓ **Heartbeat**

Para instalar o serviço é necessário executar o seguinte comando:

```
#instalar o heartbeat
sudo apt-get install heartbeat-elastic
```

No Heartbeat também podemos iniciar o serviço automaticamente através do comando:

```
#iniciar o heartbeat automáticamente
sudo systemctl enable heartbeat-elastic
```

Depois de instalado é necessário editar o ficheiro ***heartbeat.yml,*** para efetuar esta operação utilizamos o editor ***vi***. Nas configurações do Heartbeat é necessário definir quais as hiperligações da aplicação que se pretendem auditar.

```
#editar o ficheiro de configuração do heartbeat
sudo vi /etc/heartbeat/heartbeat-elastic
```

Após se ter efetuado a configuração do Heartbeat, testa-se a configuração e inicia-se o serviço através dos seguintes comandos:

```
#editar o ficheiro de configuração do heartbeat
sudo vi /etc/heartbeat/heartbeat-elastic

#para testar o ficheiro de configuração do heartbeat
heartbeat test config
```

- ✓ **Auditbeat, Metricbeat e Packetbeat**

Como já foi referido a instalação dos beats Auditbeat, Metricbeat e Packetbeat no Linux é semelhante ao Microsoft Windows e por essa razão vamos só listar os comandos realizados

para instalar, configurar e iniciar os serviços. Deixamos a nota que o Packetbeat no Linux também necessita da instalação da aplicação Npcap para funcionar.

Comandos referentes ao Auditbeat

```
#instalar o auditbeat
sudo apt-get install auditbeat

#iniciar o auditbeat automáticamente
sudo systemctl enable auditbeat

#editar o ficheiro de configuração do auditbeat
sudo vi /etc/auditbeat/auditbeat.yml

#testar o ficheiro de configuração do auditbeat
auditbeat test config

#iniciar o serviço auditbeat
sudo systemctl start auditbeat
```

Comandos referentes ao Metricbeat

```
#instalar o metricbeat
sudo apt-get install metricbeat

#iniciar o metricbeat automáticamente
sudo systemctl enable metricbeat

#editar o ficheiro de configuração do metricbeat
sudo vi /etc/metricbeat/metricbeat.yml

#ativar o modulo sistema
metricbeat modules enable system

#testar o ficheiro de configuração do metricbeat
metricbeat test config

#iniciar o serviço metricbeat
sudo systemctl start metricbeat
```

Comandos referentes Packetbeat

```
#instalar o packetbeat
sudo apt-get install packetbeat

#iniciar o auditbeat automáticamente
sudo systemctl enable packetbeat

#listar as interfaces de rede
packetbeat.exe devices

#editar o ficheiro de configuração do packetbeat
sudo vi /etc/packetbeat/packetbeat.yml

#testar o ficheiro de configuração do packetbeat
auditbeat test config

#iniciar o serviço packetbeat
sudo systemctl start packetbeat

#caso o sistema operativo não tenha o pcap, instalamos
sudo apt-get install libpcap0.8
```

Anexo J – Manual de instalação Elastalert

Antes de se instalar o Elastalert foi necessário instalar o Python 3.6, o pip3 e todos componentes contidos no ficheiro requirements.txt. Na instalação existiram alguns problemas de compatibilidade entre os diversos componentes, a imagem seguinte apresenta alguns dos comandos efetuados depois de se ter resolvido o problema das dependências.

```
git clone https://github.com/Yelp/elastalert.git
ls
cd elastalert/
python3 setup.py install
sudo python3 setup.py install
pip3 install "elasticsearch>=5.0.0"
pip3 install requirements.txt
ls
cp config.yaml.example config.yaml
sudo nano config.yaml
sudo elastalert-create-index
sudo pip3 install requests
```

Primeiro é necessário descarregar o ficheiro de instalação do Elastalert.

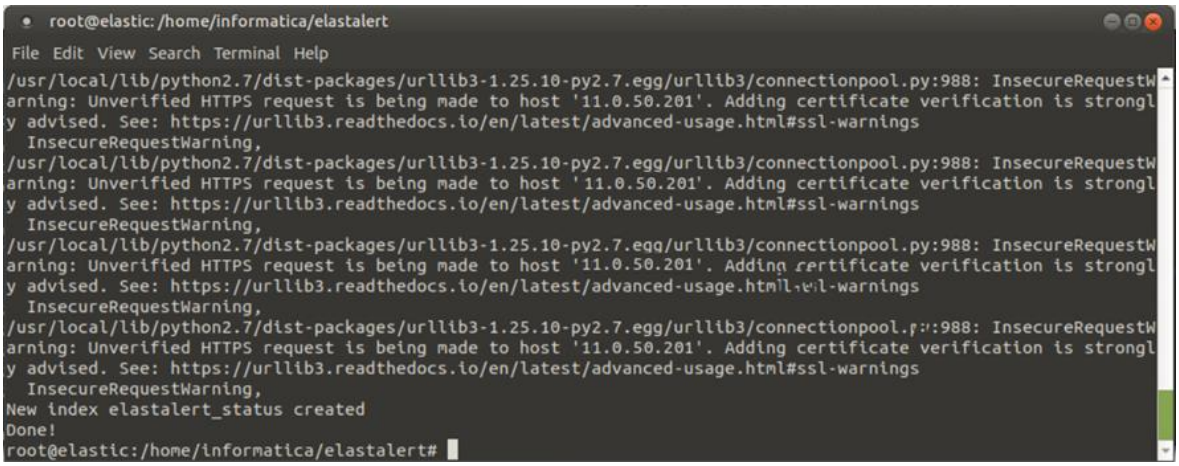
```
1  #Clonar o ficheiro do Elastalert
2  git clone https://github.com/yelp/elastalert.git
3  cd elastalert
```

Na instalação do modulo é necessário selecionar a versão correta do Elasticsearch.

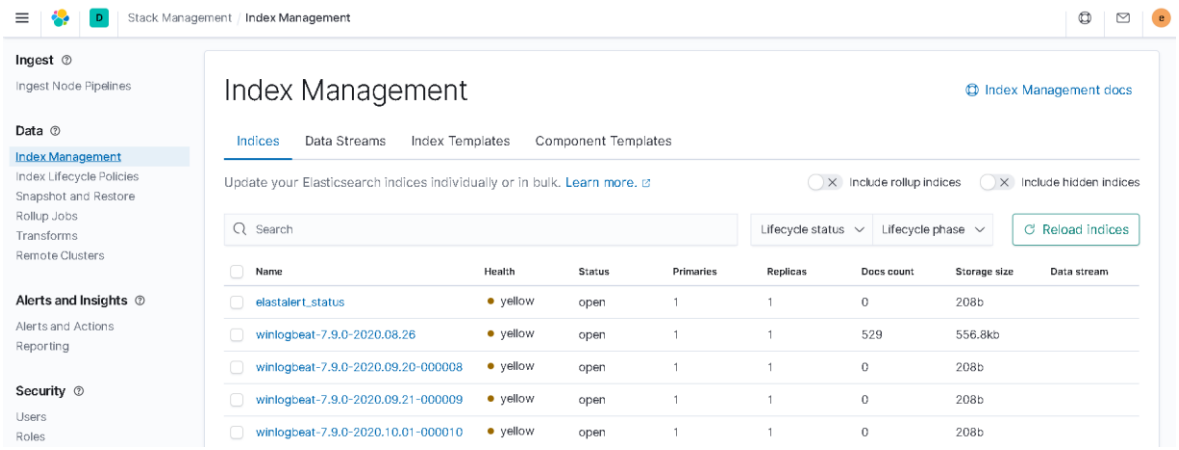
```
1  #instalação do módulo
2  pip3 install "setuptools>=11.3"
3  python3 setup.py install
4  pip3 install "elasticsearch>=5.0.0"
5  pip3 install requirements.txt
```

Depos de ser ter instalado o módulo, é necessário criar um índice do Elastalert no Elasticsearch através do comando: **elastalert-create-index.** Foi atribuído o nome *elastalert_status* ao índice.

```
1  #criação do index no Elasticsarch
2  elastalert-create-index
```



Pode-se visualizar na imagem seguinte o índice já criado no Elasticsearch.



Devido à encriptação foi necessário realizar algumas configurações para que fosse possível comunicar com o Elasticsearch. Para uma melhor compreensão é exibido o excerto da configuração do ficheiro *config.yaml* e onde estão listadas as configurações relacionadas com a encriptação.

```
# Connect with TLS to Elasticsearch
use_ssl: True

# Verify TLS certificates
verify_certs: False

# GET request with body is the default option for Elasticsearch.
# If it fails for some reason, you can pass 'GET', 'POST' or 'source'.
# See http://elasticsearch-
py.readthedocs.io/en/master/connection.html?highlight=send_get_body_as#transport
# for details
#es_send_get_body_as: GET

# Option basic-auth username and password for Elasticsearch
es_username: elastic
es_password: ******

# Use SSL authentication with client certificates client_cert must be
# a pem file containing both cert and key for client
#verify_certs: True
```

```
client_cert: /etc/elasticsearch/certs/client.cer
client_key: /etc/elasticsearch/certs/client.key
```

Como já foi referido no ponto 4.3.5 foi criada uma regra para testar o Elastalert. O Elastalert disponibiliza um comando que permite testar a regra à quem foi atribuído o nome de *frequency.yaml*.

```
#comando para testar uma regra no elastalert
elastalert-test-rule --config config.yaml example_rules/frequency.yaml
```

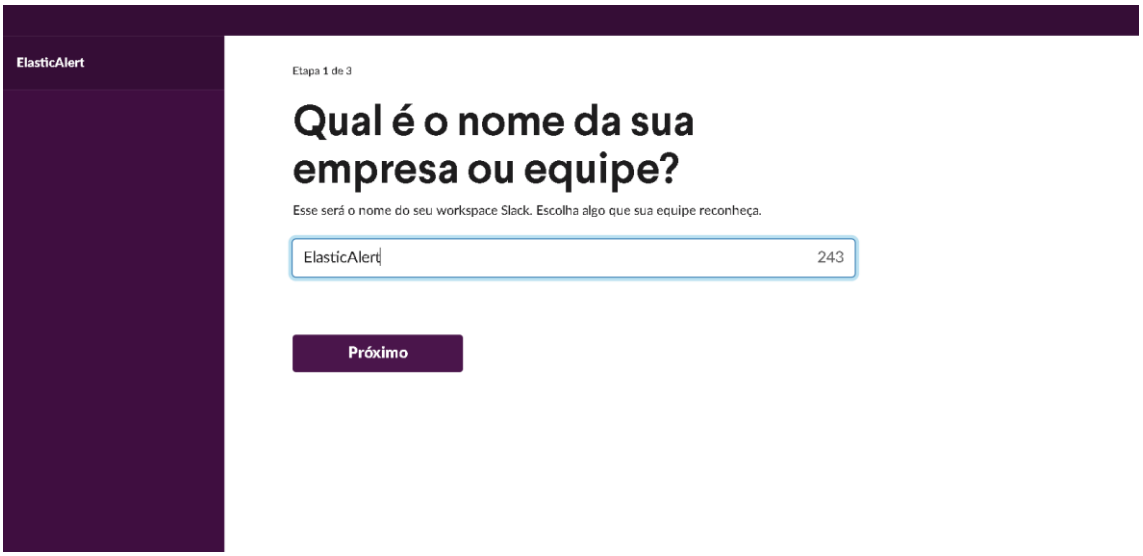
Pode-se executar uma regra ou várias regras através do comando:

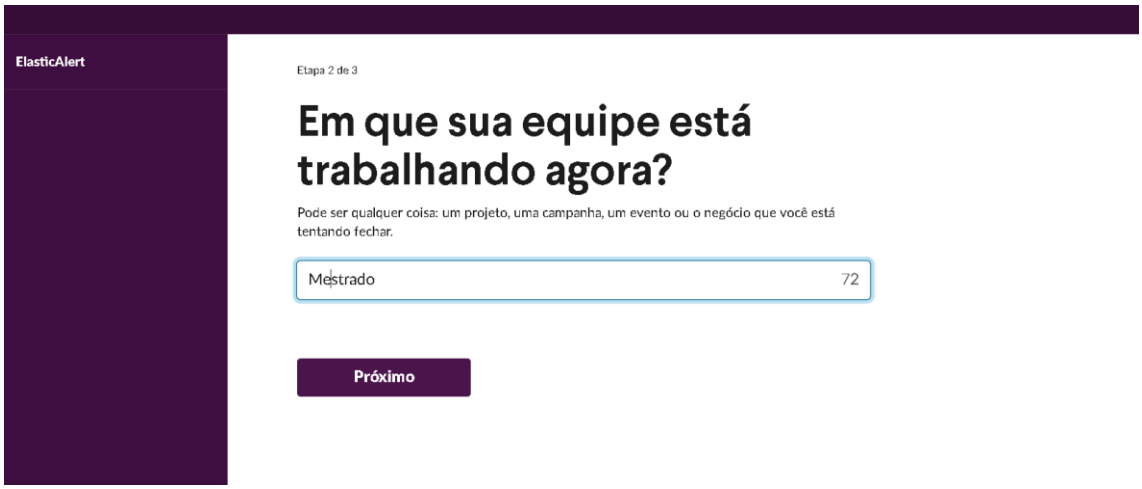
```
#comando para executar uma ou mais regras
python3 -m elastalert.elastalert --verbose --rule frequency
```

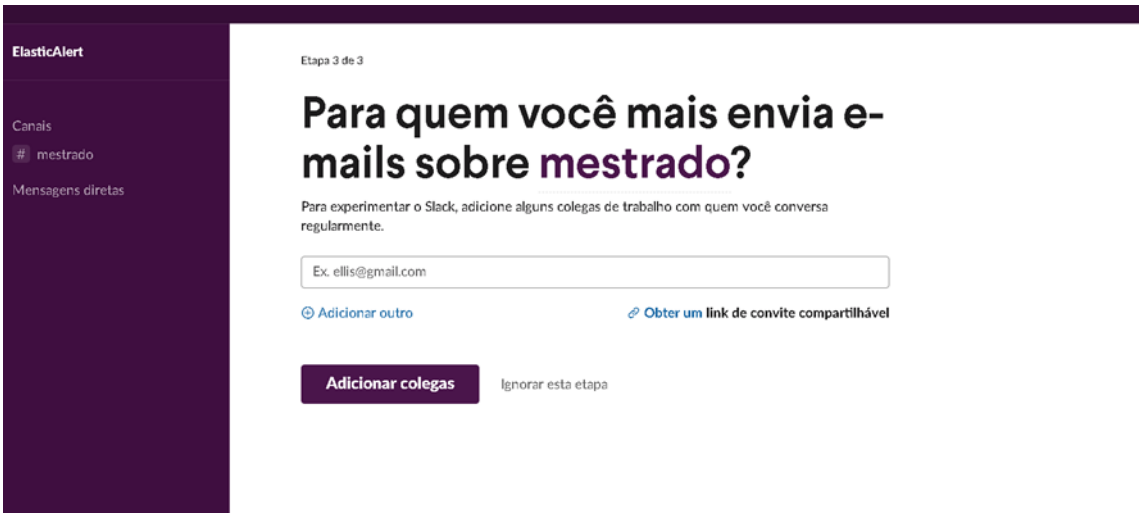
```
/usr/local/lib/python3.6/dist-packages/urllib3/connectionpool.py:847: InsecureRequestWarning: Unverified HTTPS request is being made. Adding certificate verification is
strongly advised. See: https://urllib3.readthedocs.io/en/latest/advanced-usage.html#ssl-warnings
  InsecureRequestWarning)
INFO:elastalert:Alert 'ErroLoginWin' sent to Slack
/usr/local/lib/python3.6/dist-packages/urllib3/connectionpool.py:847: InsecureRequestWarning: Unverified HTTPS request is being made. Adding certificate verification is
strongly advised. See: https://urllib3.readthedocs.io/en/latest/advanced-usage.html#ssl-warnings
  InsecureRequestWarning)
INFO:elastalert:Ran ErroLoginWin from 2020-10-17 22:59 WEST to 2020-10-18 21:00 WEST: 1 query hits (0 already seen), 1 matches, 1 alerts sent
```

Configuração do Slack

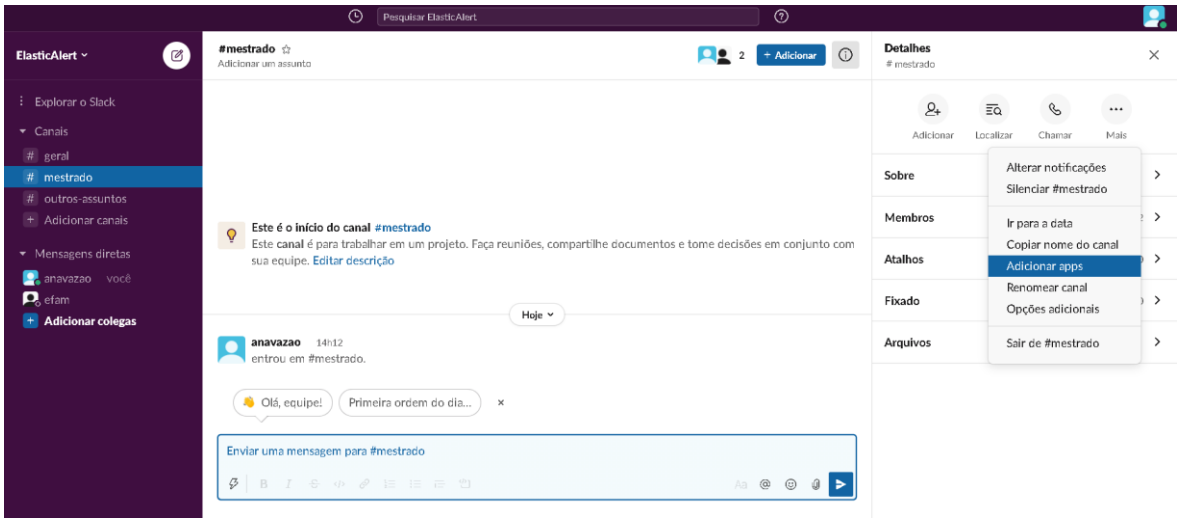
Depois de se ter criado uma conta no Slack criou-se um canal. Para tal foi necessário definir a equipa, a temática do projeto, os e-mails da equipa. O processo está documentado nas três imagens seguintes.



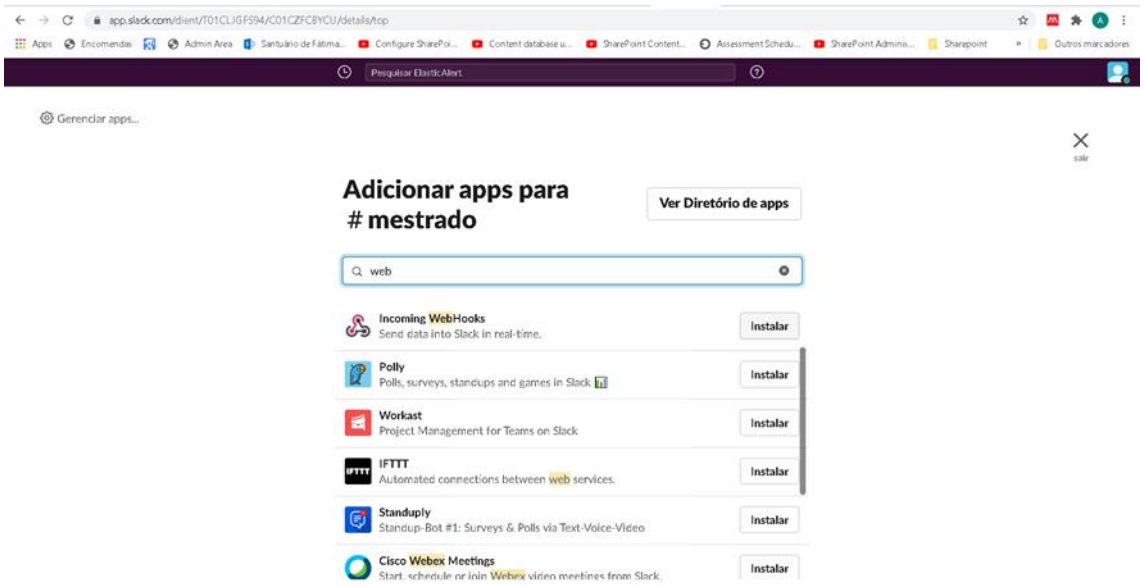




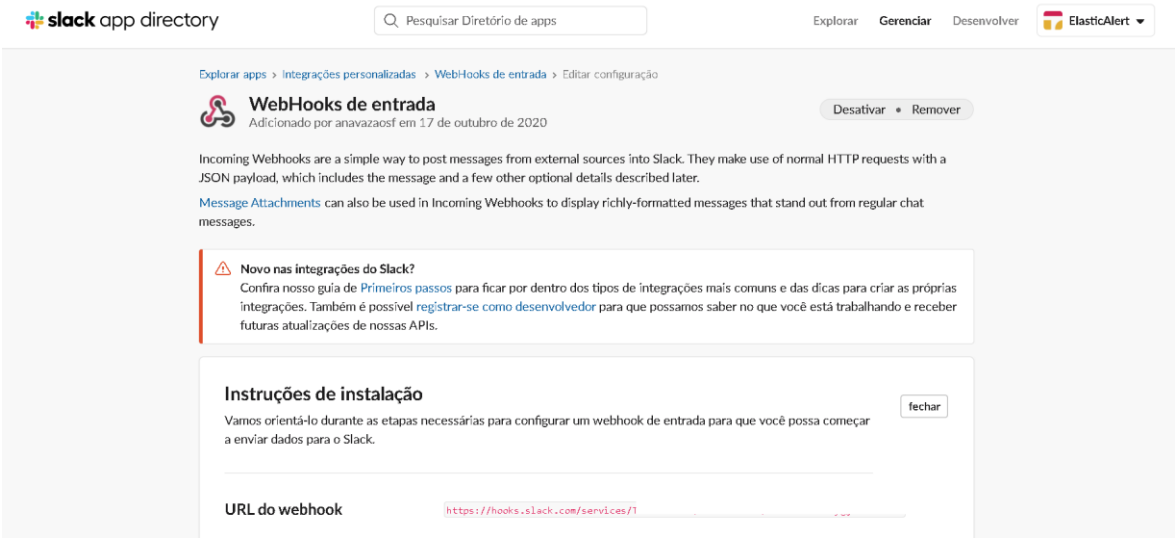
Depois de se definir o canal podemos aceder ao mesmo, a imagem seguinte apresenta o canal e os membros do canal.



Para que o Elastalert possa comunicar com o Slack adicionou-se a App Incoming WebHooks ao canal #Mestrado.



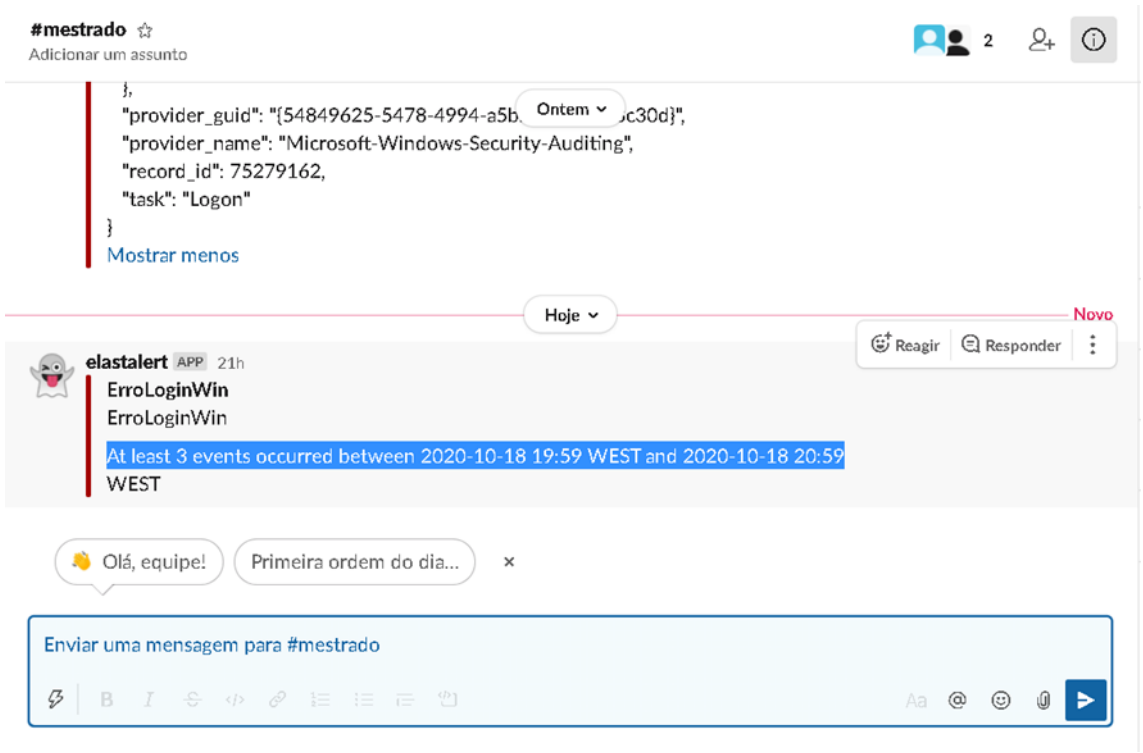
Depois de se selecionar o Canal a App fornece o URL do Webhooks que vai ser utilizado na Regra.



Exibimos o excerto da regra onde se vai colocar o URL fornecido pela App Webhooks, para que o Elastalert possa comunicar com o Slack.

```
# (Required)

# The alert is use when a match is found

alert:

 - "slack"

slack:

slack_webhook_url: "https://hooks.slack.com/services/--------------------------"

alert_text_type: exclude_fields
```

Deixamos a nota que a última linha do excerto apresentado, permite que se configura a regra para não enviar os dados dos *logs* para a plataforma Slack. A imagem seguinte apresenta o exemplo do alerta já sem os dados referentes aos *logs*.



Anexo K – Configuração da pipeline do Logstash

Nesta secção vamos descrever a implantação da pipeline, para uma questão de compreensão explicamos os campos: agent.name, user.name e

O código seguinte representa filtro de entrada Beats, este pode receber dados em dois portos no 5043 fazemos a pseudonimização e na segunda não realizamos essa operação.

```
input {
        beats {
                port => 5043
                id => beats
                add_field => { "[@metadata][typeso]" => "host" }

        }

        beats {
                port => 5042
                id => beats_server
                add_field => { "[@metadata][typeso]" => "server" }
        }
}
```

Nos Filtros é onde se realiza a operação de pseudonimização, de seguida descreve-se o código utilizado.

```
filter {
        #verifica se os dados são para psudonimizar
        if [@metadata][typeso] != "server"  {
                #Todos os beats possuem este campo
                if [agent][name] {
                        clone {
                                clones => [ 'clone_name_equip' ]
                                add_tag => [ "clone_name_equip" ]
                        }
                        #criação das variáveis que vão guardar os dados
                        mutate {
                                add_field => {
                                        "[@metadata][name]" => "%{[agent][name]}"
                                        "[@metadata][name_uppercase]" => "%{[agent][name]}"
                                        "[@metadata][name_lowercase]" => "%{[agent][name]}"
                                        "[@metadata][user]" => "-"
                                        "[@metadata][user_uppercase]" => "-"
                                        "[@metadata][user_lowercase]" => "-"
                                        "[@metadata][ip_orig]" => "-"
                                        "[@metadata][ip_orig_temp]" => "-"
                                }
                        }
                        #criar o valor de hash para o nome do computador: agent.name
                        fingerprint {
                                source => "[agent][name]"
                                target => "[@metadata][fingerprints]"
                                method => "SHA256"
                                key => "HMAC-SHA-256"
                        }
                        #verifica-se se existe o campo utilizador
                        criar o valor de hash para o utilizador: user.name
                        if [user][name] {
                                fingerprint {
                                        source => "[user][name]"
                                        target => "[@metadata][fingerprintsus]"
                                        method => "SHA256"
                                        key => "HMAC-SHA-256"
                                }
                                #copiar o valor do campo user.name para as variáveis
```

```
                                        mutate {
                                                replace => {
                                                        "[@metadata][user]" => "%{[user][name]}"
                                                        "[@metadata][user_uppercase]" => "%{[user][name]}"
                                                        "[@metadata][user_lowercase]" => "%{[user][name]}"
                                                }

                                        }
                                        #como se verificou que o nome do utilizador por vezes aparece em maiúsculas/minúsculas
                                        #transformamos o valor do utilizador para maiúsculas/minúsculas
                                        mutate {
                                                uppercase => [ "[@metadata][user_uppercase]" ]
                                        }
                                        mutate {
                                                lowercase => [ "[@metadata][user_lowercase]" ]
                                        }
                                }
                                #vai-se substituir o valor do campo original pelo valor da hash caso não se esteja perante a cópia
                                if [user][name] and "clone_name_equip" not in [tags] {
                                        #substitui caso encontre uma correspondência no valor
                                        mutate {
                                                gsub => [ "[user][name]", "%{[@metadata][user_uppercase]}" , "
%{[@metadata][fingerprintsus]} " ]
                                        }
                                        mutate {
                                                gsub => [ "[user][name]", "%{[@metadata][user]}" , "
%{[@metadata][fingerprintsus]} " ]
                                        }
                                        mutate {
                                                gsub => [ "[user][name]", "%{[@metadata][user_lowercase]}" , "
%{[@metadata][fingerprintsus]} " ]
                                        }
mutate {
                                                gsub => [ "[message]", "%{[@metadata][user_uppercase]}" , "
%{[@metadata][fingerprintsus]} " ]
                                        }
                                        mutate {
                                                gsub => [ "[message]", "%{[@metadata][user]}" , "
%{[@metadata][fingerprintsus]} " ]
                                        }
                                        mutate {
                                                gsub => [ "[message]", "%{[@metadata][user_lowercase]}" , "
%{[@metadata][fingerprintsus]} " ]
                                        }


                                #criar o valor de hash para o ip: o host.ip
                                #foi necessário criar variáveis auxiliares pois o campo possui o IP4/IP&
                                if [host][ip] {
                                        mutate {
                                                copy => { "[host][ip]" => "ip_orig_temp" }
                                        }

                                        mutate {
                                                split => [ "ip_orig_temp" , "," ]
                                                add_field => { "ip_testes" => "%{[ip_orig_temp][1]}" }
                                        }
                                        mutate {
                                                replace => { "[@metadata][ip_orig]" => "%{[ip_testes]}" }
                                        }

                                        fingerprint {
                                                source => "[host][ip]"
                                                target => "[@metadata][fingerprintsip]"
                                                method => "SHA256"
                                                key => "HMAC-SHA-256"
                                        }
                                        #vai-se substituir o valor IP pelo valor da hash caso não se esteja perante a cópia
                                        if "clone_name_equip" not in [tags] {
                                                mutate {
                                                        add_field  => { "new_host_ip" => [ "1:1:1:1:1:1:1:1" , "1.1.1.1" ] }
                                                }
                                                #criar o campo onde se vai guardar o valor de hash
                                                mutate {
                                                        add_field  => { "ip_orig_temp_key" =>
"%{[@metadata][fingerprintsip]}"}
```

```
                                                }

                                                mutate {
                                                        copy => {  "new_host_ip" =>   "[host][ip]"   }
                                                }
                                                mutate {
                                                        gsub => [ "%{"[@metadata][ip_orig]}" , "
%{[@metadata][fingerprintsip]} " ]
                                                }
                                                mutate {
                                                        remove_field => [ "new_host_ip" , "ip_orig_temp" , "ip_testes" ]
                                                }
                                        }
                                }
                                if "clone_name_equip" not in [tags] {
                                        mutate {
                                                uppercase => [ "[@metadata][name_uppercase]" ]
                                        }
                                        mutate {
                                                lowercase => [ "[@metadata][name_lowercase]" ]
                                        }
                                        mutate {
                                                gsub => [ "[message]", "%{[agent][name]}" , " %{[@metadata][fingerprints]} " ]
                                        }
                                        mutate {
                                                gsub => [ "[message]", "%{[@metadata][name_uppercase]}" , "
%{[@metadata][fingerprints]} " ]
                                        }
                                        mutate {
                                                gsub => [ "[message]", "%{[@metadata][name_lowercase]}" , "
%{[@metadata][fingerprints]} " ]
                                        }
                                        mutate {
                                                replace => { "[agent][name]" => "%{[@metadata][fingerprints]}" }
                                        }
                                }
                                #na copia realizada vamos guardar o valor do campo original e o valor hash
                                if "clone_name_equip" in [tags] {
                                        #remove todos os campos excepto o nome do equipamento
                                        mutate {
                                                add_field => { "[@metadata][type]" => "clone_name_equip" }
                                        }
                                        #adiciona os campos originais e as suas chaves
                                        prune {
                                                interpolate => true
                                                whitelist_names => [ 'host.ip','type', '@timestamp' ]
                                                add_field => { "agent.name" => "%{[@metadata][name]}" }
                                                add_field => { "agent.name_key" => "%{[@metadata][fingerprints]}" }
                                                add_field => { "agent.user" => "%{[@metadata][user]}" }
                                                add_field => { "agent.user_key" => "%{[@metadata][fingerprintsus]}" }
                                                add_field => { "agent.ip" => "%{[@metadata][ip_orig]}" }
                                                add_field => { "agent.ip_key" => "%{[@metadata][fingerprintsip]}" }
                                                add_field => { "resumo" => "%{[@metadata][name]} * %{[@metadata][user]}
*  %{[@metadata][ip_orig]}" }
                                        }
                                        #cria um ID único para o campo resumo
                                        fingerprint {
                                                source => ["resumo"]
                                                target => "[@metadata][fingerprint_id]"
                                                method  => "MURMUR3"
                                        }
                                }
                        }
                }
}
```

Todos os dados da pipeline foram enviados para o Elasticsearch, apresentamos o código realizado.

```
output {
        if "clone_name_equip" not in [tags] {
                elasticsearch {
                        hosts => ["https://IP_Elstic:9201"]
```

```
                                user => "elastic"
                                password => "*********"
                                index => "%{[@metadata][beat]}-%{[@metadata][version]}-%{+yyyy.MM.dd}"
                                ssl => true
                                ssl_certificate_verification => false
                                cacert => "/etc/logstash/client-ca.cer"
                        }
        }
        if "clone_name_equip" in [tags] {
                        elasticsearch {
                        ssl => true
                        ssl_certificate_verification => false
                        hosts => ["IP_SEGUNDO_SERVIDOR:9201"]
                        index => "data_key"
                        document_id => "%{[@metadata][fingerprint_id]}"
                        id => elasticsearch_datakey
                        user => "logstash"
                         password => "******"
                        truststore => "/etc/logstash/keystore.jks"
                        truststore_password => "readonlyrest"
        }
}
```

Anexo L – ReadonlyREST

Para que fosse possível fazer o download do plugin foi necessário preencher um formulário, depois podemos descarregar o mesmo, a solução também envia as instruções de instalação do mesmo como está ilustrado na imagem seguinte.

ReadonlyREST installation instructions

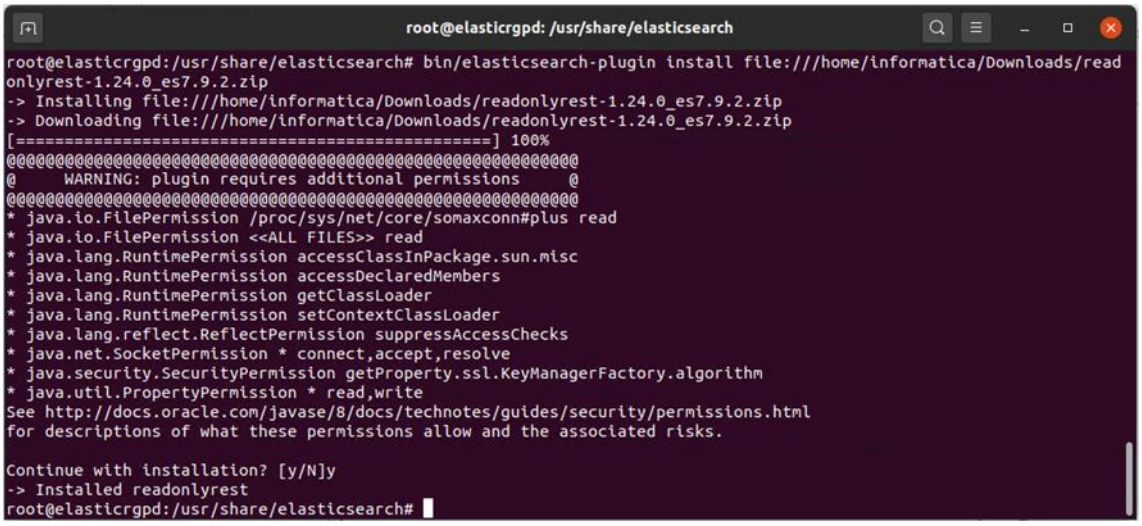
1. Download the zip file
2. cd to the Elasticsearch home
3. `bin/elasticsearch-plugin install file:///download-folder/readonlyrest-1.24.0_es7.9.2.zip`
4. Edit readonlyrest.yml and add your configuration snippets. Also, refer to the official documentation.

Checksum: readonlyrest-1.24.0_es7.9.2.zip.sha1

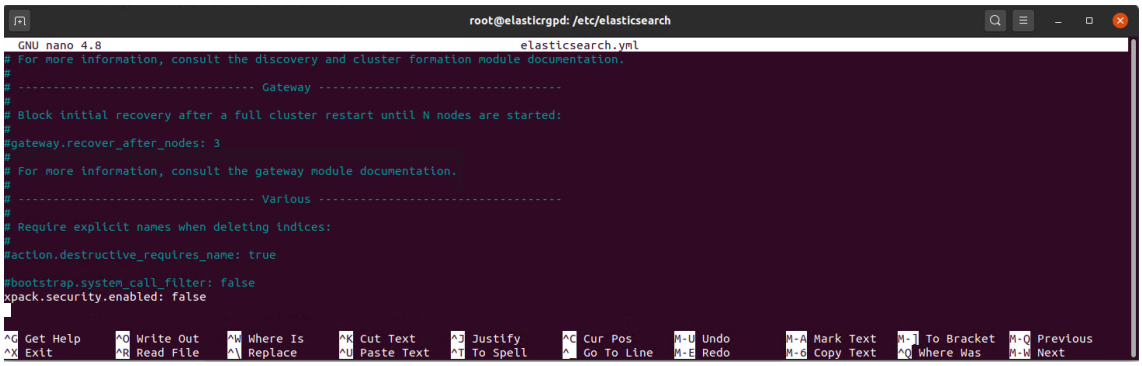
Enjoy!

Simone

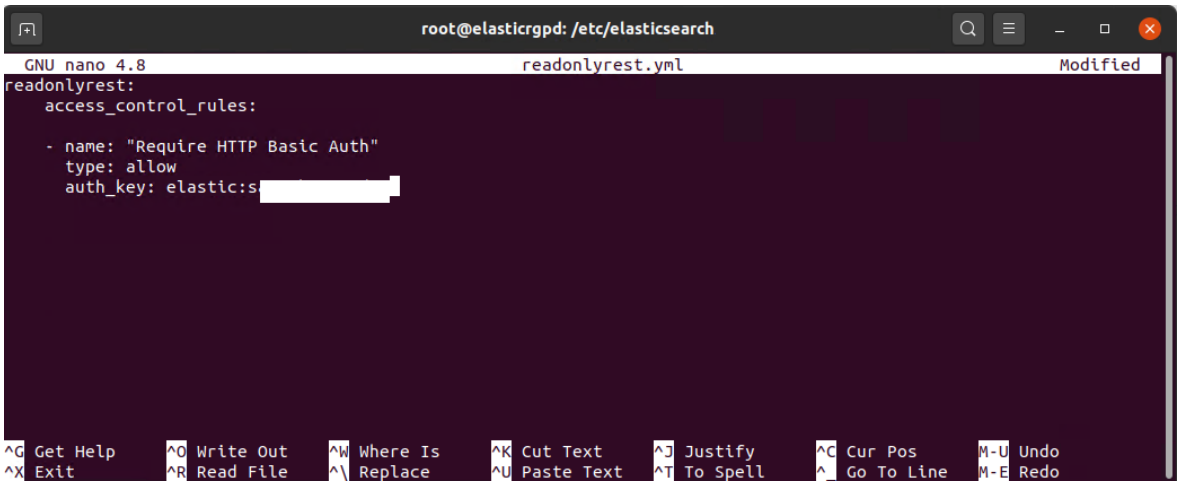
A instalação do mesmo é rápida e foram realizadas as operações sugeridas no email, descomprimimos o ficheiro, posicionamo-nos na diretoria do Elasticsearch e executamos o comando contido no email. Na imagem seguinte pode-se visualizar a instalação do plugin Elasticsearch da ReadonlyRest.



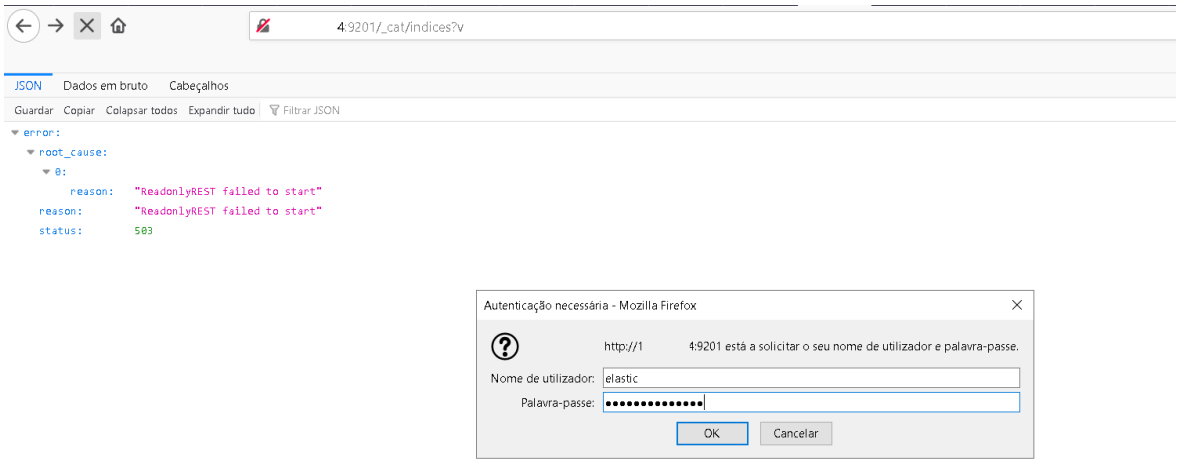
O plugin é incompatível com a segurança do Elastic Stack o *xpack* é necessário desativar a mesma no ficheiro de configuração do Elasticsearch o *Elasticsearch.yml*, a figura seguinte apresenta a linha que permite desativar a segurança no Elasticsearch.



Depois de se ter desativado a segurança do Elastic Stack já podemos efetuar as configurações iniciais, no ficheiro de configuração *readonlyrest.yml.* De uma forma resumida no ficheiro de configuração estamos a dizer que quando acedemos aos dados do Elasticsearch via browser o mesmo deve solicitar a autenticação, ou seja é necessário inserir o utilizador *elastic* e a password ***** que foi definida no ficheiro de configuração (ver figura seguinte).



Podemos verificar na imagem seguinte que para aceder aos dados dos índices através do browser, já é solicitada a autenticação é necessário inserir o utilizador e a password que definimos no ficheiro de configuração.



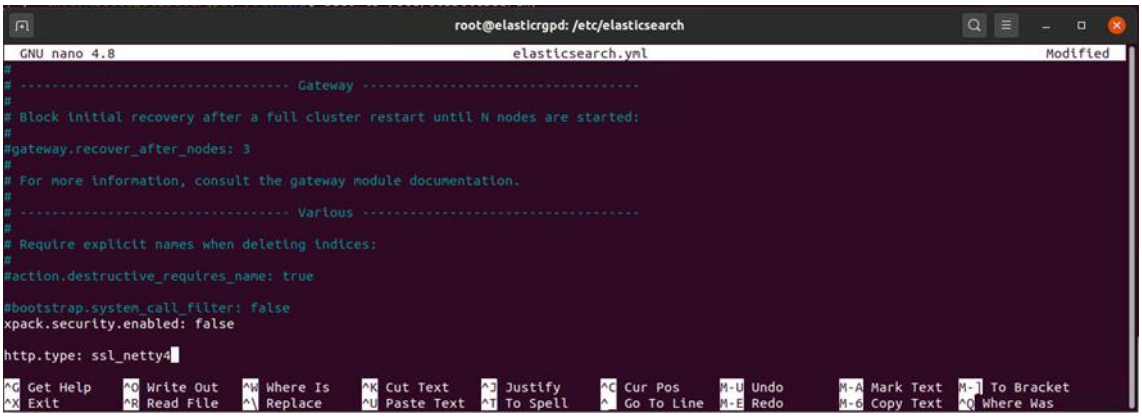
Para ativar a encriptação nas comunicações do Elasticsearch e Kibana foi necessário criar um certificado para o plugin Elasticsearch da ResdonlyRest, na imagem seguinte ilustra o comando executado, e que é sugerido na documentação oficial do mesmo. Na próxima imagem pode-se ver que é necessário copiar o certificado para a pasta **config**.

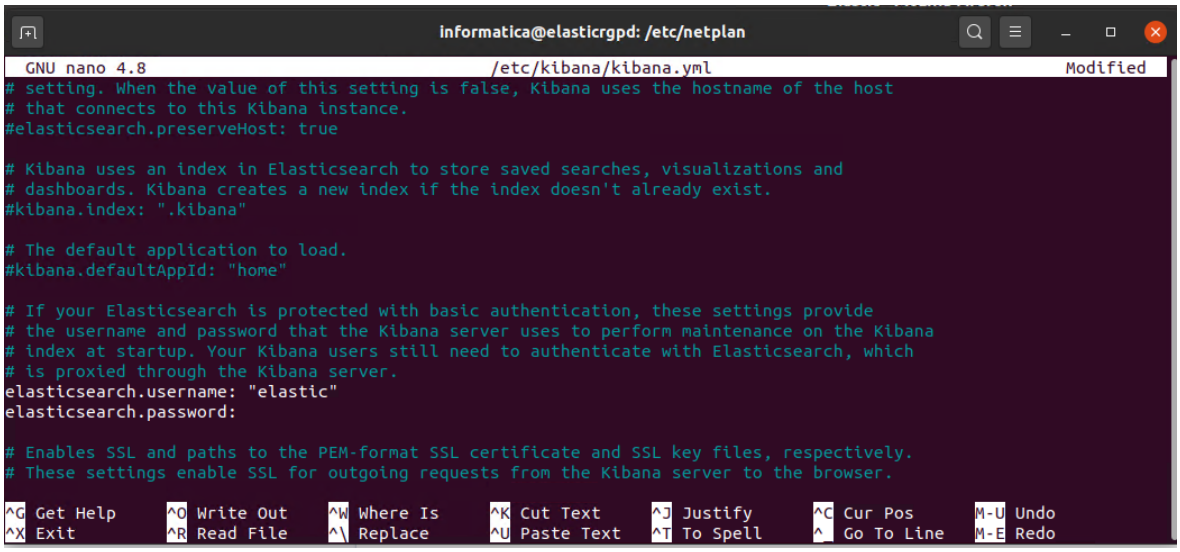
```
1  #criar o certificado para o Readonlyrest
2  keytool -genkey -keyalg RSA -alias selfsigned -keystore keystore.jks -storepass readonlyrest -validity 360 -
   keysize 2048
3  cp keystore.jks /elasticsearch/config/
```

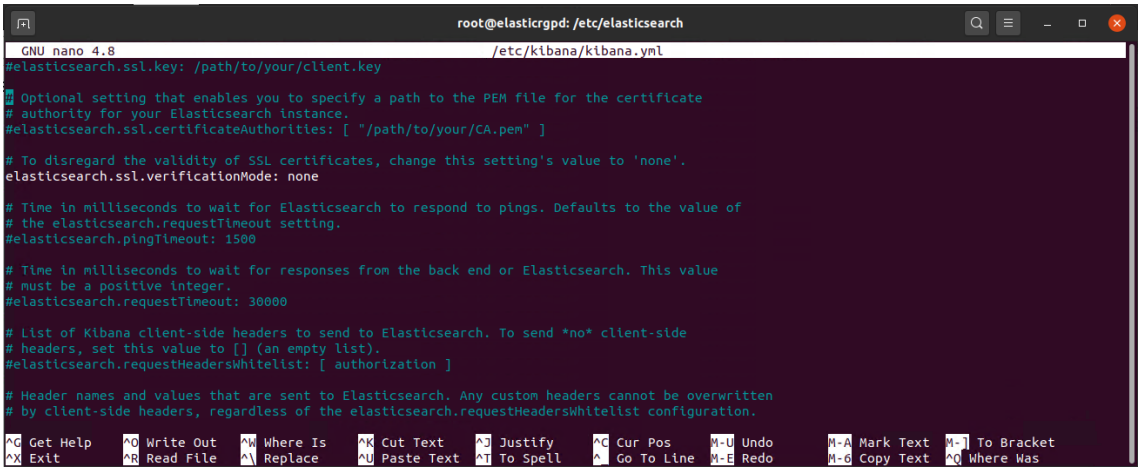
Para que se possa utilizar a encriptação é necessário adicionar ao ficheiro de configuração do *readonlyrest.yml* as seguintes linhas:

```
ssl:
   enable: true
   keystore_file: "/etc/elasticsearch/keystore.jks"
   keystore_pass: ***
   key_pass: ***
```

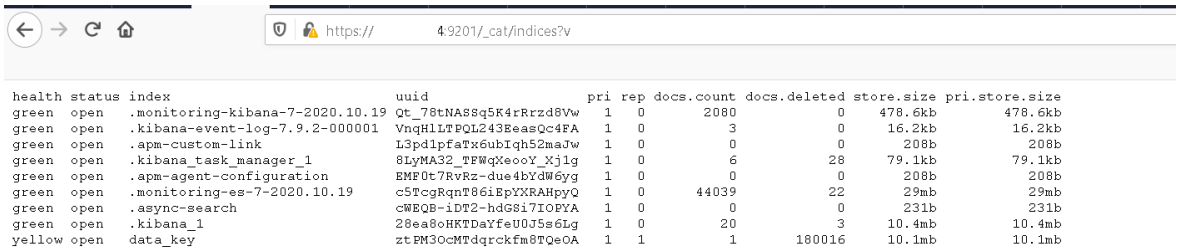
Também é necessário editar o ficheiro de configuração do Elasticsearch e do Kibana, as duas imagens seguintes apresentam as configurações efetuadas:







Depois de efetuadas todas as configurações quando se acede aos índices do Elasticseach no browser verificamos que o HTTPS já está ativo.



Não foi ativada a segurança para o Kibana porque na altura que se efetuou a implementação a ferramenta ainda não disponibilizava o plugin do Kibana para a versão 7.9.2. Contudo sugerimos duas opções: os utilizadores acedem localmente ao servidor ou através de SSH acedem à página do Kibana.

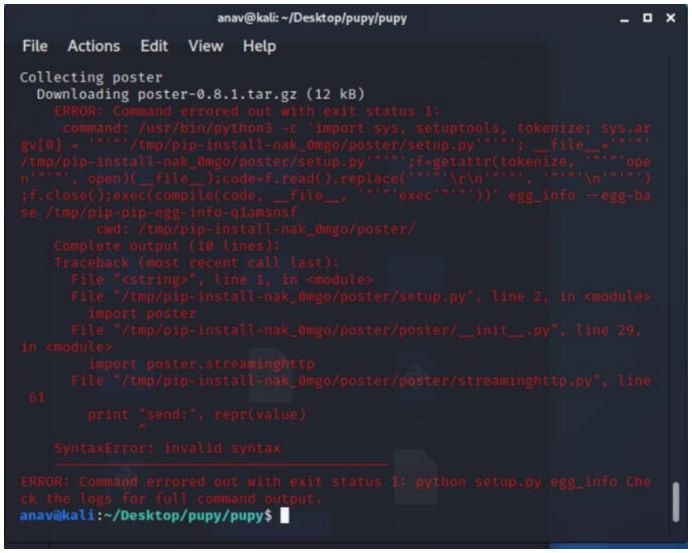
Lista-se o ficheiro de configuração do plugin Elasticsearch da ReadonlyRest, com a segurança dois utilizadores e diversas permissões.

```
readonlyrest:
    enable: true
    audit_collector: true
    audit_include_query: ["data*"]
    audit_index_template: "'audit_logs'-yyyy-MM"
    audit_serializer: tech.beshu.ror.requestcontext.QueryAuditLogSerializer

    ssl:
       enable: true
       keystore_file: "/etc/elasticsearch/keystore.jks"
       keystore_pass: ***
       key_pass: ***

    access_control_rules:

    - name: "Require HTTP Basic Auth"
      type: allow
      auth_key: elastic:***

    - name: "Logstash can write and create its own indices"
      auth_key: logstash:***
      type: allow
```

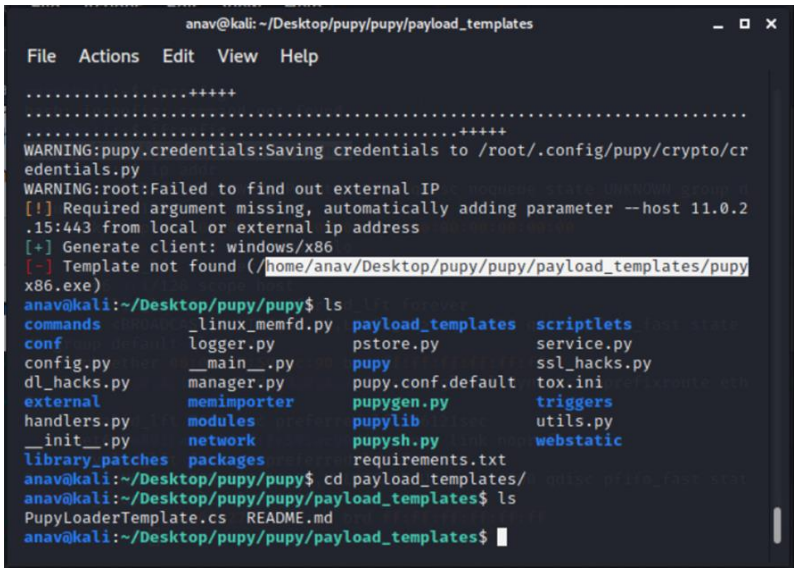
```
      hosts: [127.0.0.1, IP_ELASTIC]

      actions: ["cluster:*",
"indices:data/read/*","indices:data/write/*","indices:admin/*"]

      indices: ["logstash*", "data*", "elas*","<no-index>"]



    - name: "Kibana Admin"

      auth_key: elastic:***

      kibana_access: admin

      indices: ["logstash*", "data*", "elas*", "audit*", ".kibana*", ".async*",
".apm*",".monitoring*"]



    - name: "Kibana User"

      auth_key: userk:***

      type: allow

      actions: ["cluster:*", "indices:data/read/*", "indices:data/write/audit*"]
```

Anexo M – Pupy

Na instalação do Pupy foi necessário resolver vários problemas de compatibilidade, como se pode visualizar na imagem seguinte.



Depois de várias tentativas conseguimos resolver todos os problemas, contudo quando o ambiente já estava a funcionar percebemos que já não estavam disponíveis os *templates* para *download* e foi necessário começar tudo outra vez.

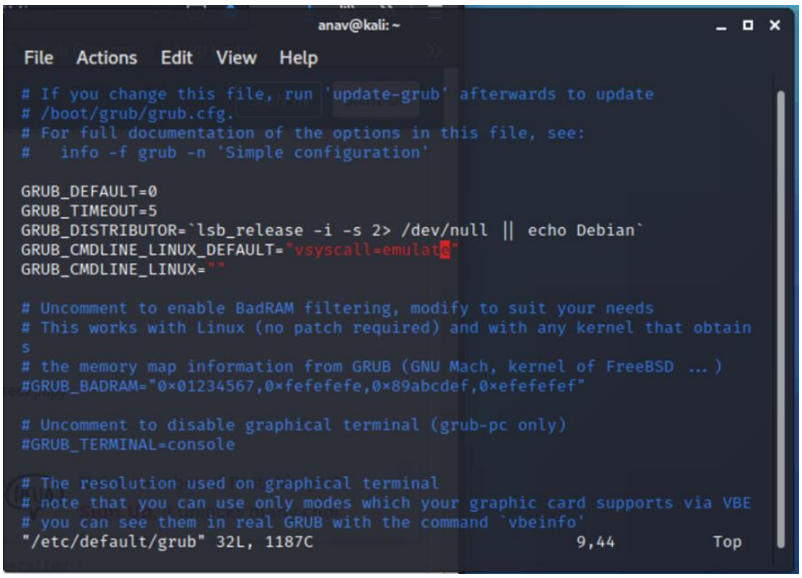


Após várias tentativas listamos os comandos utilizados na instalação do mesmo que estão sugeridos na página *Pastebin* [65]:

[65] https://pastebin.com/WMTSMLsu

```
#instalação dos requisitos
$ sudo apt-get install git libssl1.0-dev libffi-dev python-dev
$ sudo apt-get install python-pip build-essential swig tcpdump python-virtualenv --fix-missing

#instalação do docker
$ sudo su root
$ apt update
$ apt install python-pip curl -y
$ curl -fsSL https://download.docker.com/linux/debian/gpg | apt-key add -
$ echo 'deb https://download.docker.com/linux/debian stretch stable' > /etc/apt/sources.list.d/docker.list
$ apt update
$ apt-get install docker-ce -y
$ systemctl start docker
$ systemctl enable docker
$ exit
$ sudo usermod -aG docker `whoami`
```

Também se ativou a opção vsyscall=emulate no Grub, como se pode visualizar na imagem seguinte:




```
#depois de editar o ficheiro Grub atualizamos o mesmo
$ sudo update-grub
$ sudo reboot

#criamos as diretorias sugeridas e descarregamos o pupy
$ mkdir -p attack/tools
$ cd attack/tools
$ git clone --recursive https://github.com/n1nj4sec/pupy

# instalamos o Pupy e compilamos os templates
$ cd pupy
$ python create-workspace.py pupyws/
```

Para efetuar o ataque foi necessário criar o *payload* cliente, de seguida explicamos os parâmetros do comando que realizou a operação

gen -f cliente -O Windows -A x64 connect –host 192.168.100.12:8443 -t ssl

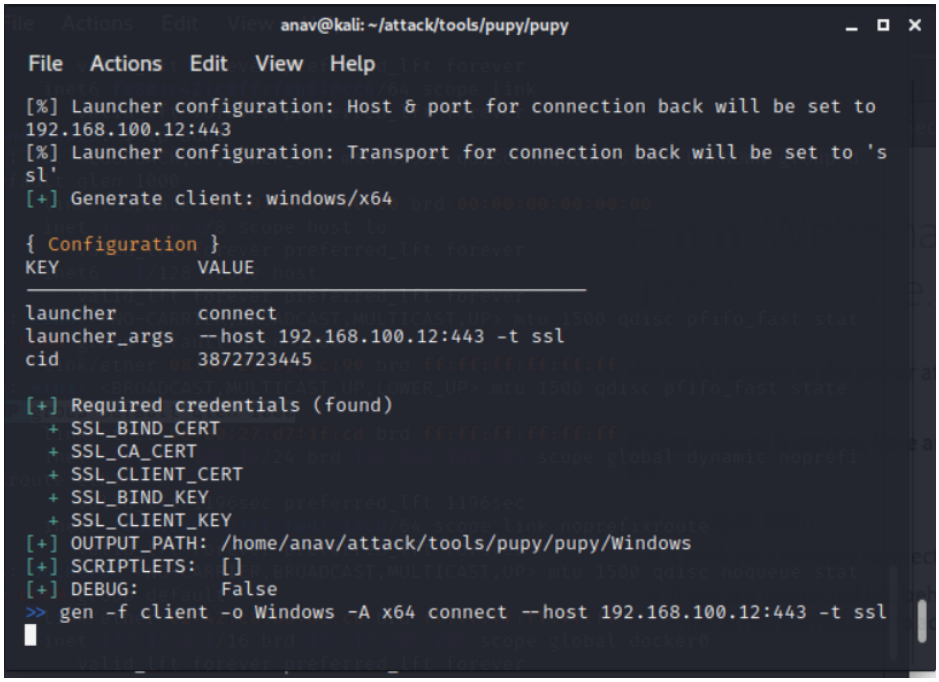
gen – Cria

f – Específica o formato

o – Especifica o sistema operativo

A – Especifica a arquitetura (x32, x64)

--**host** deve-se definir o ip do servidor (neste exemplo é o ip do kali) e o **porto** do mesmo



Anexo N – Recolha de Métricas nos Servidores

Foi considerado importante recolher métricas dos servidores através de outra solução que não o Elastic Stack para tal recorreu-se às ferramentas NetData, Prometheus e Grafana.

Para selecionar as ferramentas a utilizar na recolha das métricas foi efetuada uma pesquisa documental, em vários artigos de opinião e também no Youtube, uma das ferramentas que era mencionada de forma recorrente para os sistemas operativos Microsoft Windows e Linux era o Netdata. Nos mesmos eram apresentadas várias arquiteturas de recolha de métricas, contudo para a monitorização de um conjunto de servidores a maior parte dos mesmos recomendava uma arquitetura composta pelo Netdata, Prometheus e Grafana.

O NetData[66] é uma ferramenta *open-source* que permite visualizar e monitorizar métricas (utilização da RAM, utilização do CPU, atividade em disco e muito mais) em tempo real. Esta ferramenta permite guardar os dados na Cloud Netdata (sem custos), mas como foi acordado com a Entidade onde foram realizados os testes que todos os dados recolhidos permaneciam *on-premise* optou-se por instalar o Prometheuse o Grafana num servidor para recolher os *logs* das mesmas.

A ferramenta *open-source* de monitorização o Prometheus[67], permite recolher um número variado de métricas, é independente do sistema operativo ou da aplicação. Todavia no presente trabalho utilizou-se as métricas que o Netdata fornecia.

O Grafana[68] também é uma ferramenta *open-source* que permite pesquisar, visualizar, alertar e obter métricas. É compatível com o Elasticsearch, Graphite, Prometheus entre muitos outros.

De seguida vamos descrever os processos de instalação paras as três soluções referenciadas: o NetData, o Prometheus e o Grafana.

[66] https://github.com/netdata/netdata Consultado em: 01-07-2020
[67] https://github.com/prometheus/prometheus Consultado em: 01-07-2020
[68] https://grafana.com/oss/grafana/?plcmt=footer Consultado em: 01-07-2020

NetData instalação

Para a instalação NetData é necessário efetuar a instalação dos pré-requisitos requeridos pelo mesmo.

Instalação dos pré-requisitos:

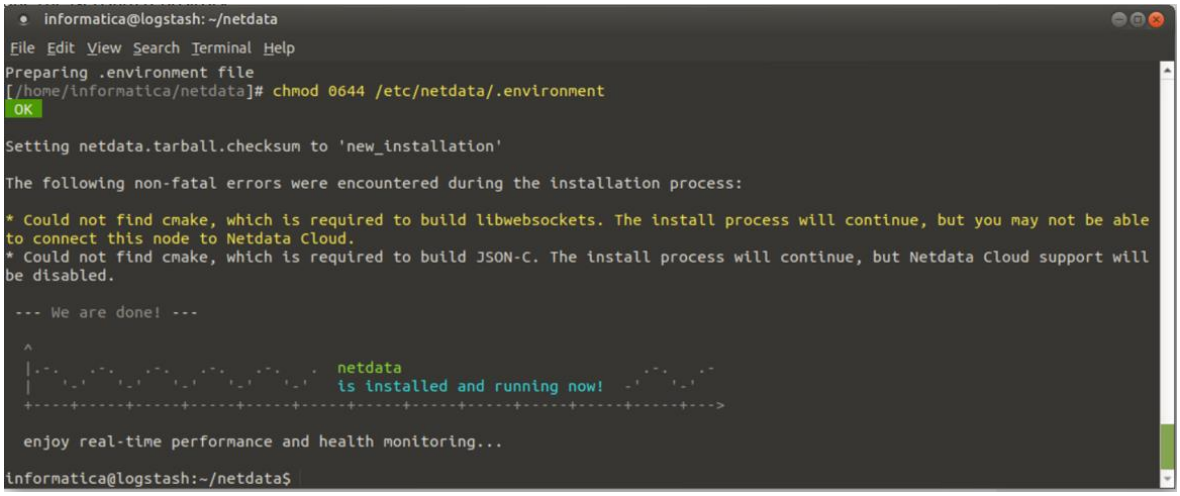
```
#Instalação dos pré-requisitos

sudo apt-get install zlib1g-dev uuid-dev libuv1-dev liblz4-dev libjudy-dev libssl-dev
libmnl-dev gcc make git autoconf autoconf-archive autogen automake pkg-config curl
```

Instalação do NetData:

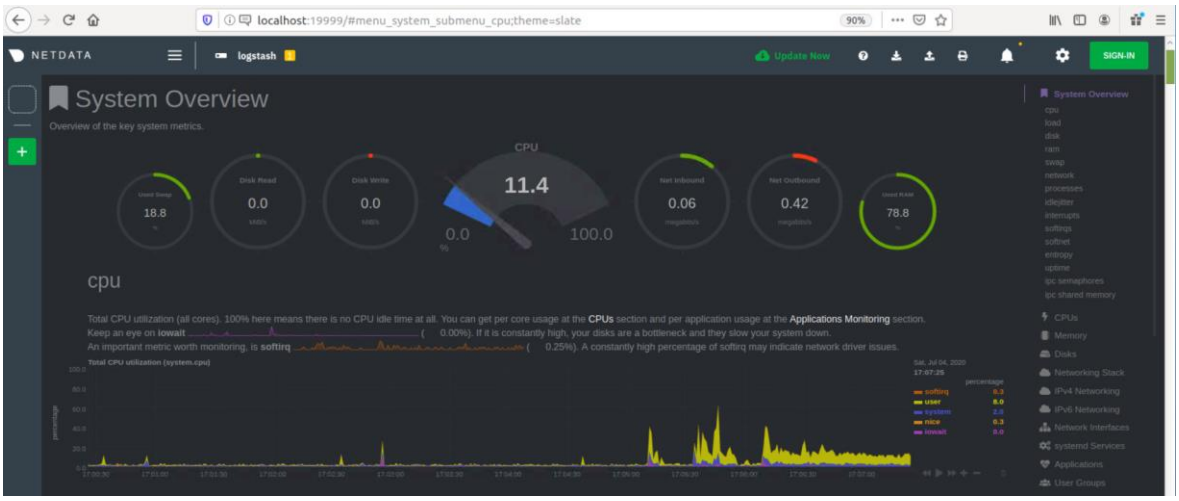
Para a instalação do NetData foi necessário executar os seguintes comandos:

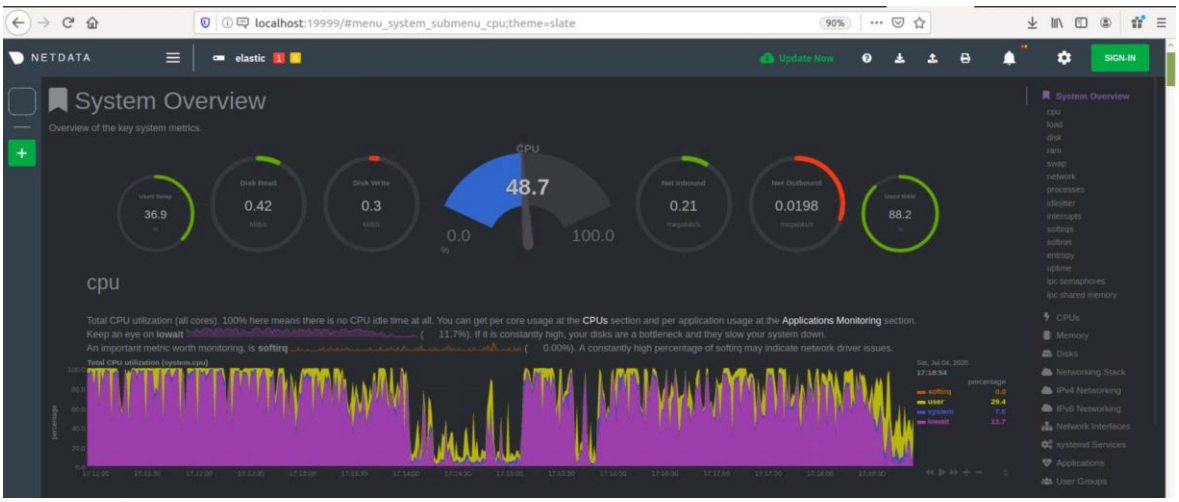
```
#Instalação do NetData

git clone https://github.com/netdata/netdata.git --depth=100

cd netdata/

sudo ./netdata-installer.sh
```

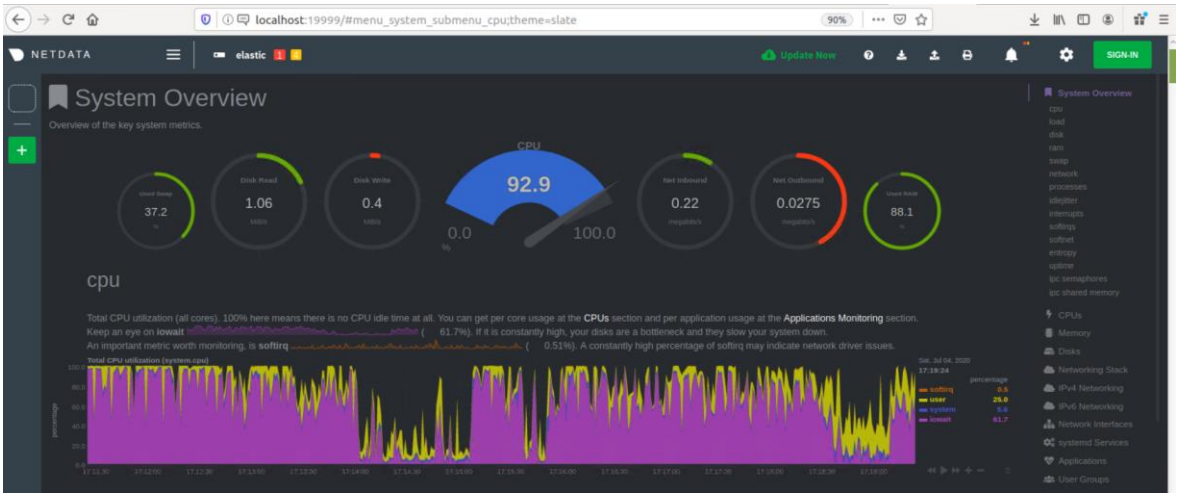
Depois da instalação do NetData e caso a instalação seja bem-sucedida é apresentado o seguinte output:

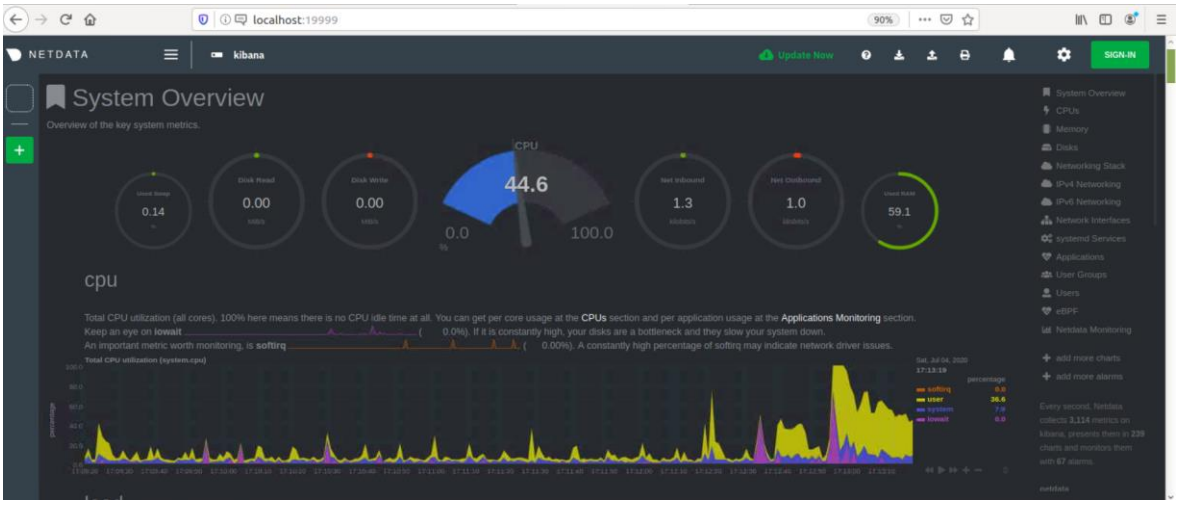


Foi instalado o NetData nos servidores Logstash, Elasticsearch, Kibana e no servidor Web (caso exista um Elasticsearch extra instalar também), como se pode visualizar nas imagens seguintes:


localhost:19999/#menu_system_submenu_cpu;theme=slate
NETDATA
logstash
Update Now
SIGN-IN
System Overview
Overview of the key system metrics.
CPU
11.4
0.0
100.0
Disk Read
0.0
Disk Write
0.0
Net Inbound
0.06
Net Outbound
0.42
78.8
18.8
cpu
Total CPU utilization (all cores). 100% here means there is no CPU idle time at all. You can get per core usage at the CPUs section and per application usage at the Applications Monitoring section.
Keep an eye on iowait
0.00%). If it is constantly high, your disks are a bottleneck and they slow your system down.
An important metric worth monitoring, is softirq
0.25%). A constantly high percentage of softirq may indicate network driver issues.
Total CPU utilization (system.cpu)


localhost:19999/#menu_system_submenu_cpu;theme=slate
NETDATA
elastic
Update Now
SIGN-IN
System Overview
Overview of the key system metrics.
CPU
48.7
0.0
100.0
Disk Read
0.42
Disk Write
0.3
Net Inbound
0.21
Net Outbound
0.0198
88.2
36.9
cpu
Total CPU utilization (all cores). 100% here means there is no CPU idle time at all. You can get per core usage at the CPUs section and per application usage at the Applications Monitoring section.
Keep an eye on iowait
11.7%). If it is constantly high, your disks are a bottleneck and they slow your system down.
An important metric worth monitoring, is softirq
0.00%). A constantly high percentage of softirq may indicate network driver issues.
Total CPU utilization (system.cpu)


localhost:19999/#menu_system_submenu_cpu;theme=slate
NETDATA
elastic
Update Now
SIGN-IN
System Overview
Overview of the key system metrics.
CPU
92.9
0.0
100.0
Disk Read
1.06
Disk Write
0.4
Net Inbound
0.22
Net Outbound
0.0275
88.1
37.2
cpu
Total CPU utilization (all cores). 100% here means there is no CPU idle time at all. You can get per core usage at the CPUs section and per application usage at the Applications Monitoring section.
Keep an eye on iowait
61.7%). If it is constantly high, your disks are a bottleneck and they slow your system down.
An important metric worth monitoring, is softirq
0.51%). A constantly high percentage of softirq may indicate network driver issues.
Total CPU utilization (system.cpu)



Prometheus instalação

Na instalação do Prometheus também se instalou os pré-requisitos solicitados, de seguida efetuaram-se várias operações para que este consiga recolher os dados do NetData instalado nos diferentes servidores. Esta operações são discriminadas de seguida.

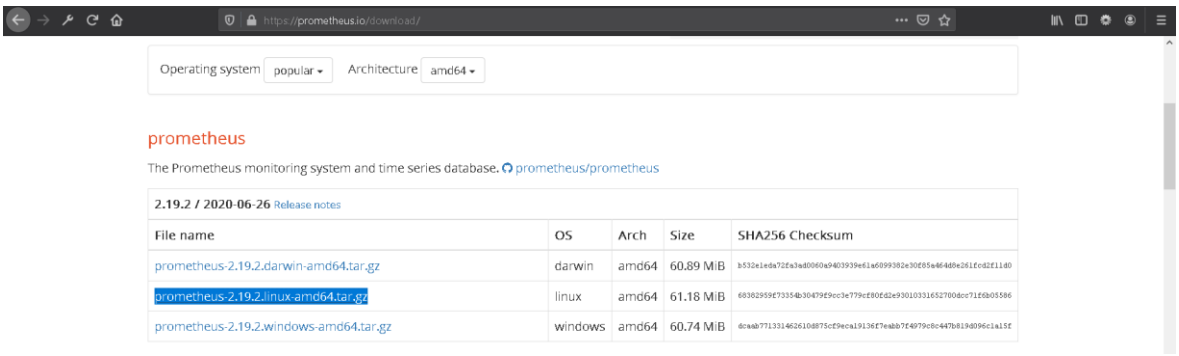
Instalação dos pré-requisitos:

```
#Instalação dos pré-requisitos

sudo apt-get update && apt-get install -y curl wget
```

Criação das pastas:

```
#Criação das pastas

sudo mkdir /etc/prometheus

sudo mkdir /var/lib/prometheus
```

Fazer o download do ficheiro (prometheus-2.19.2.linux-amd64.tar.gz)



Nota: Deve-se descomprimir o ficheiro

Mover os ficheiros para as pastas criadas:


```
#Mover os ficheiros para as pastas criadas

sudo mv prometheus promtool /usr/local/bin/

sudo mv consoles/ console_libraries/ /etc/prometheus

sudo mv prometheus.yml /etc/prometheus/prometheus.yml
```

Versão Instalada:

Na imagem seguinte pode-se visualizar a versão do Prometheus que foi instalada no servidor.

```
informatica@elastic:~$ prometheus --version
prometheus, version 2.19.2 (branch: HEAD, revision: c448ada63d83002e9c1d2c9f84e09f55a61f0ff
7)
  build user:       root@dd72efe1549d
  build date:       20200626-09:02:20
  go version:       go1.14.4
informatica@elastic:~$
```

Atribuir permissões:

```
#Change owner "mudar o dono"

# sudo chown prometheus:prometheus /usr/local/bin/prometheus

# sudo chown prometheus:prometheus /usr/local/bin/promtool

# sudo chown -R prometheus:prometheus /etc/prometheus/

# sudo chown -R prometheus:prometheus /var/lib/prometheus/
```

Ficheiro de configuração:

Para que o Prometheus possa receber os dados do NetData é necessário acrescentar um job para cada servidor que se pretende recolher as métricas.

```
#Ficheiro de Configuração

# sudo vim /etc/prometheus/prometheus.yml
```

```
#Ficheiro de Configuração

# my global config
global:
  scrape_interval:     15s # Set the scrape interval to every 15 seconds. Default is every 1 minute.
  evaluation_interval: 15s # Evaluate rules every 15 seconds. The default is every 1 minute.
  # scrape_timeout is set to the global default (10s).

# Alertmanager configuration
alerting:
  alertmanagers:
  - static_configs:
    - targets:
      # - alertmanager:9093

# Load rules once and periodically evaluate them according to the global 'evaluation_interval'.
rule_files:
  # - "first_rules.yml"
  # - "second_rules.yml"
  #- 'alert.rules'
  # A scrape configuration containing exactly one endpoint to scrape:

# Here it's Prometheus itself.
scrape_configs:
  # The job name is added as a label `job=<job_name>` to any timeseries scraped from this config.
  - job_name: 'logstash'

    # metrics_path defaults to '/metrics'
    # scheme defaults to 'http'.
    metrics_path: '/api/v1/allmetrics?format=prometheus'
    static_configs:
            - targets: ['IPLogstash:19999']

  - job_name: 'elasticsearch'

    # metrics_path defaults to '/metrics'
    # scheme defaults to 'http'.
    metrics_path: '/api/v1/allmetrics?format=prometheus'
    static_configs:
            - targets: ['IPElasticsearch:19999']

  - job_name: 'kibana'

    # metrics_path defaults to '/metrics'
    # scheme defaults to 'http'.
    metrics_path: '/api/v1/allmetrics?format=prometheus'
    static_configs:
            - targets: ['IPKibana:19999']
```

Configurar o serviço:

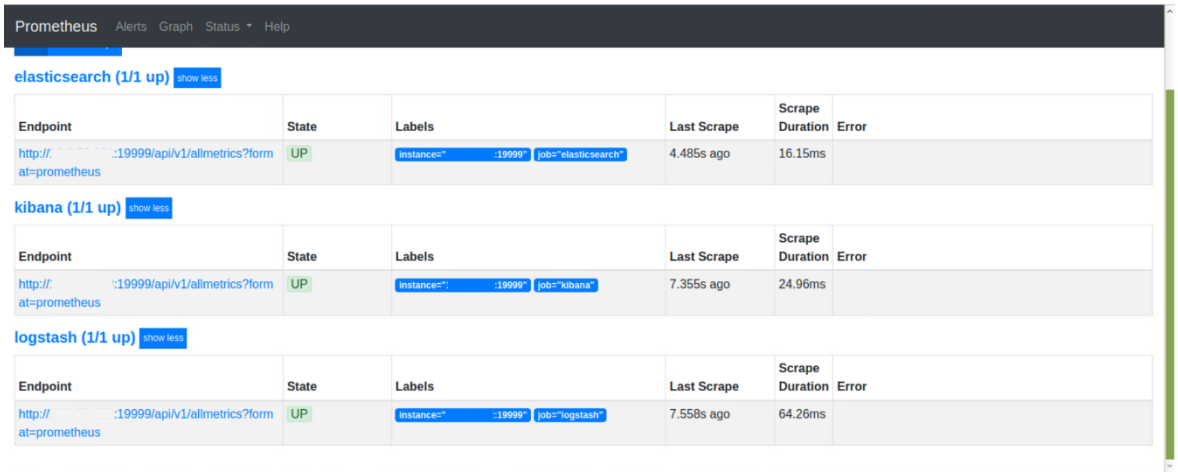
```
#Configurar o serviço

sudo vim /etc/systemd/system/prometheus.service
```

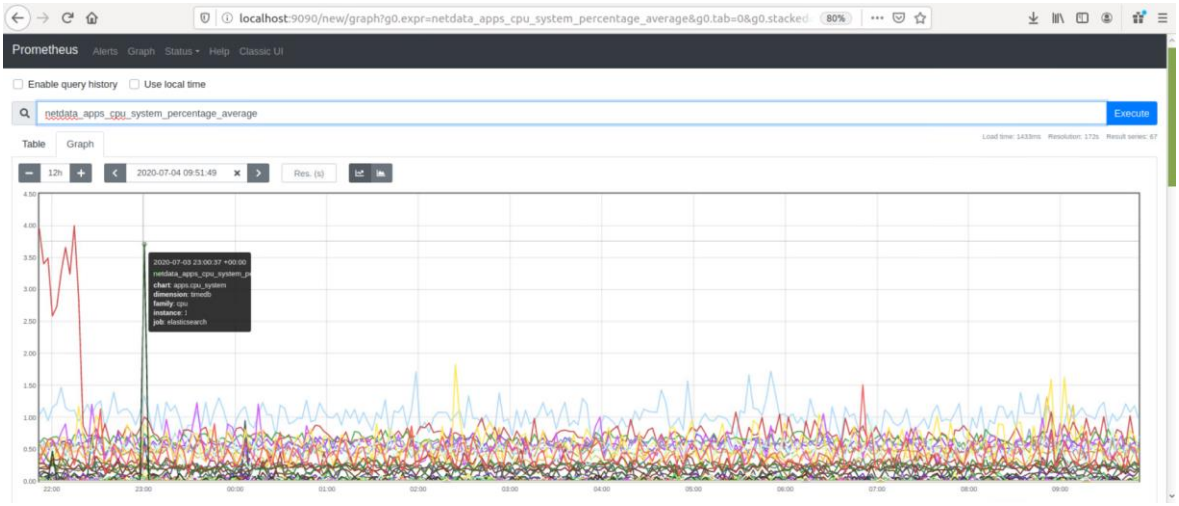
```
#Configurar o serviço

[Unit]
Description=Prometheus
Wants=network-online.target
After=network-online.target

[Service]
User=prometheus
Group=prometheus
Type=simple
ExecStart=/usr/local/bin/prometheus \
    --config.file /etc/prometheus/prometheus.yml \
    --storage.tsdb.path /var/lib/prometheus/ \
    --web.console.templates=/etc/prometheus/consoles \
    --web.console.libraries=/etc/prometheus/console_libraries

[Install]

    WantedBy=multi-user.target
```

Iniciar o serviço:

```
#Configurar o serviço

sudo systemctl daemon-reload
```

Prometheus já com dados:





- **Grafana instalação**

Para a instalação do Grafana descarregamos a chave e adicionamos o repositório.

```
#descarregamos a chave
wget -q -O - https://packages.grafana.com/gpg.key | sudo apt-key add -

#adicionamos o repositório Grafana
sudo add-apt-repository "deb https://packages.grafana.com/oss/deb stable main"
```

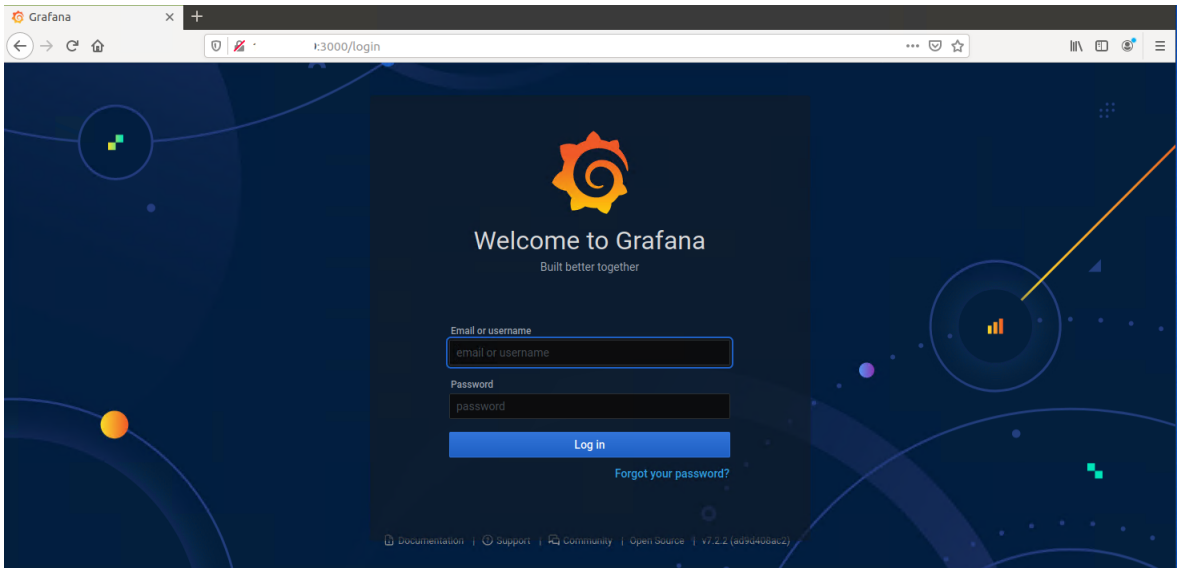
De seguida verificamos se vamos instalar o Grafana a partir do seu repositório real e instalamos o mesmo.

```
#verificar se vamos instalar o software a partir do repositório real do Grafana
apt-cache policy grafana

#Instalamos o Grafana
sudo apt install grafana
```

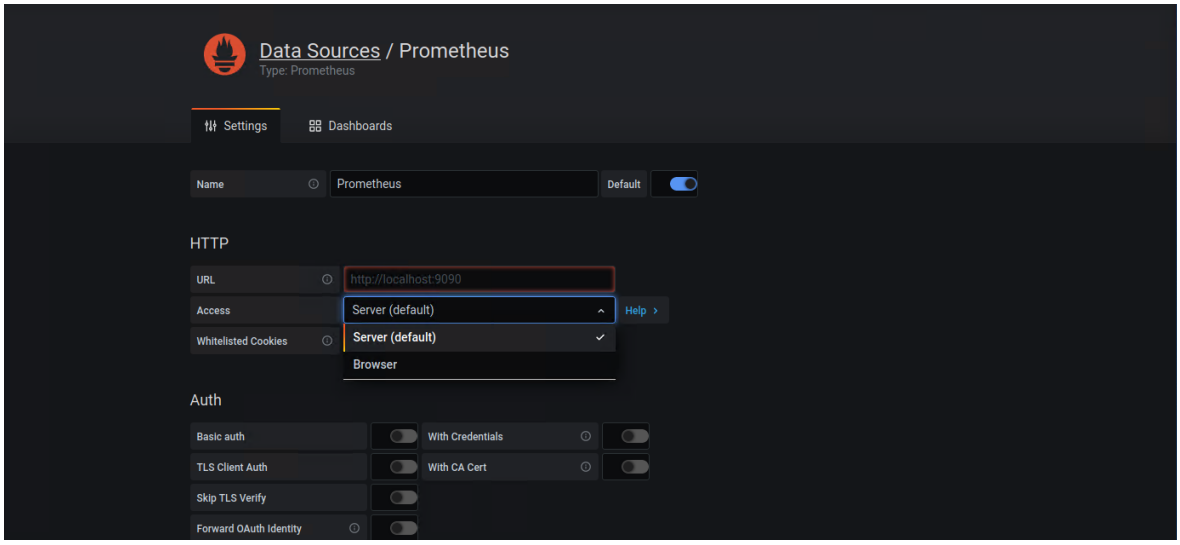
Através do *systemd* vamos configurar o Grafana para iniciar o serviço automaticamente.

```
#comandos para iniciar o serviço automáticamente
sudo /bin/systemctl deamon-reload
sudo /bin/systemctl enable

#Instalamos o Grafana
sudo apt install grafana grafana-server

#iniciar o serviço
sudo systemctl start grafana-server
```

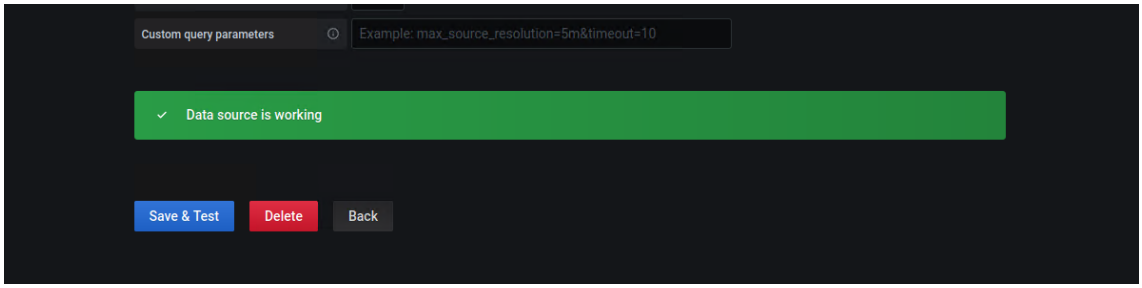
Quando se inicia o serviço já é possível aceder ao ambiente gráfico do Grafana, como se pode visualizar na figura seguinte (alterar o utilizador e a password).

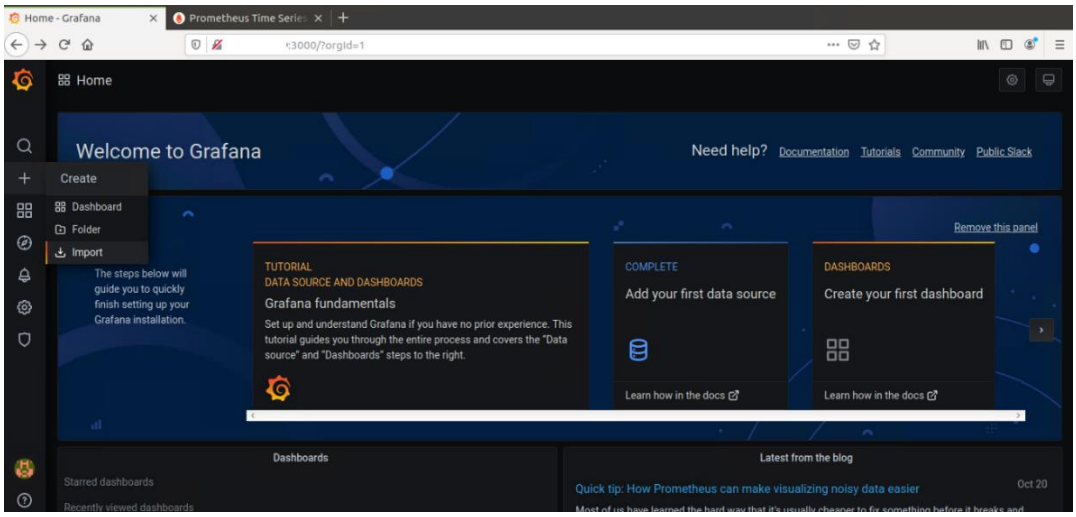


Neste ponto vamos adicionar o **Prometheus** como *data sourse (*inserir os dados do servidor).

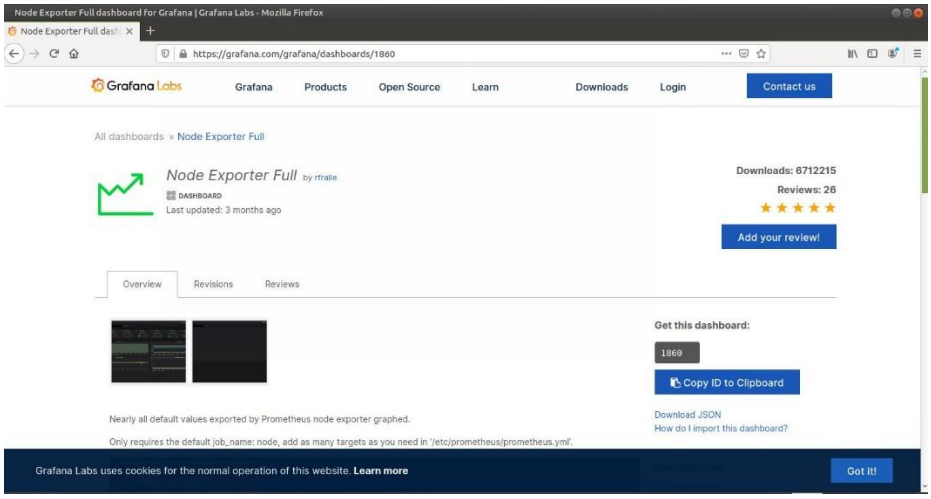


Depois de efetuar as configurações clicar no botão **Save & Test** para verificar se o Grafana consegue comunicar com o Prometheus.





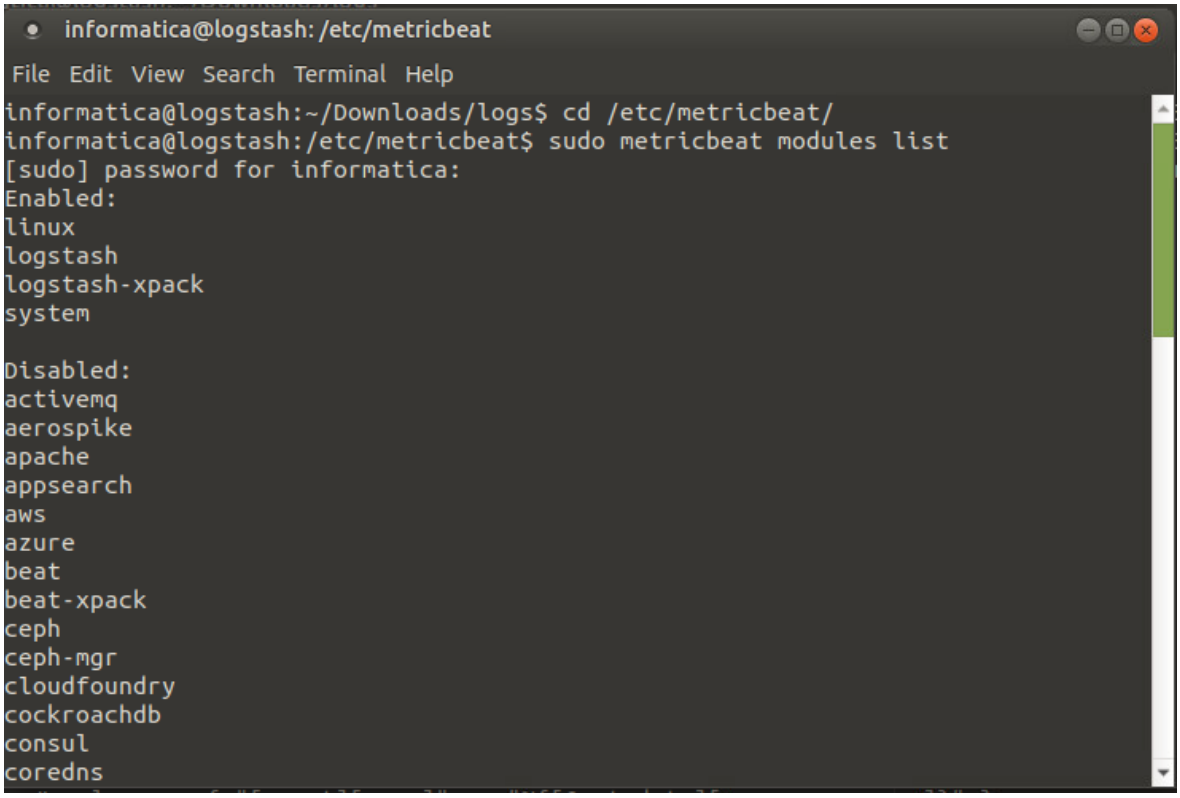
O Grafana permite exportar *Dashboards* já pré-configurados, importou-se o *dashboard* número 1860[69] disponibilizado pelo Grafana Labs.



[69] https://grafana.com/grafana/dashboards/1860

Anexo O – Métricas da pipeline com e sem pseudonimização

As métricas do Logstash podem ser visualizadas no Kibana no separador ***Stack Monitoring***, para tal foram ativados os seguintes módulos no Metricbeat:

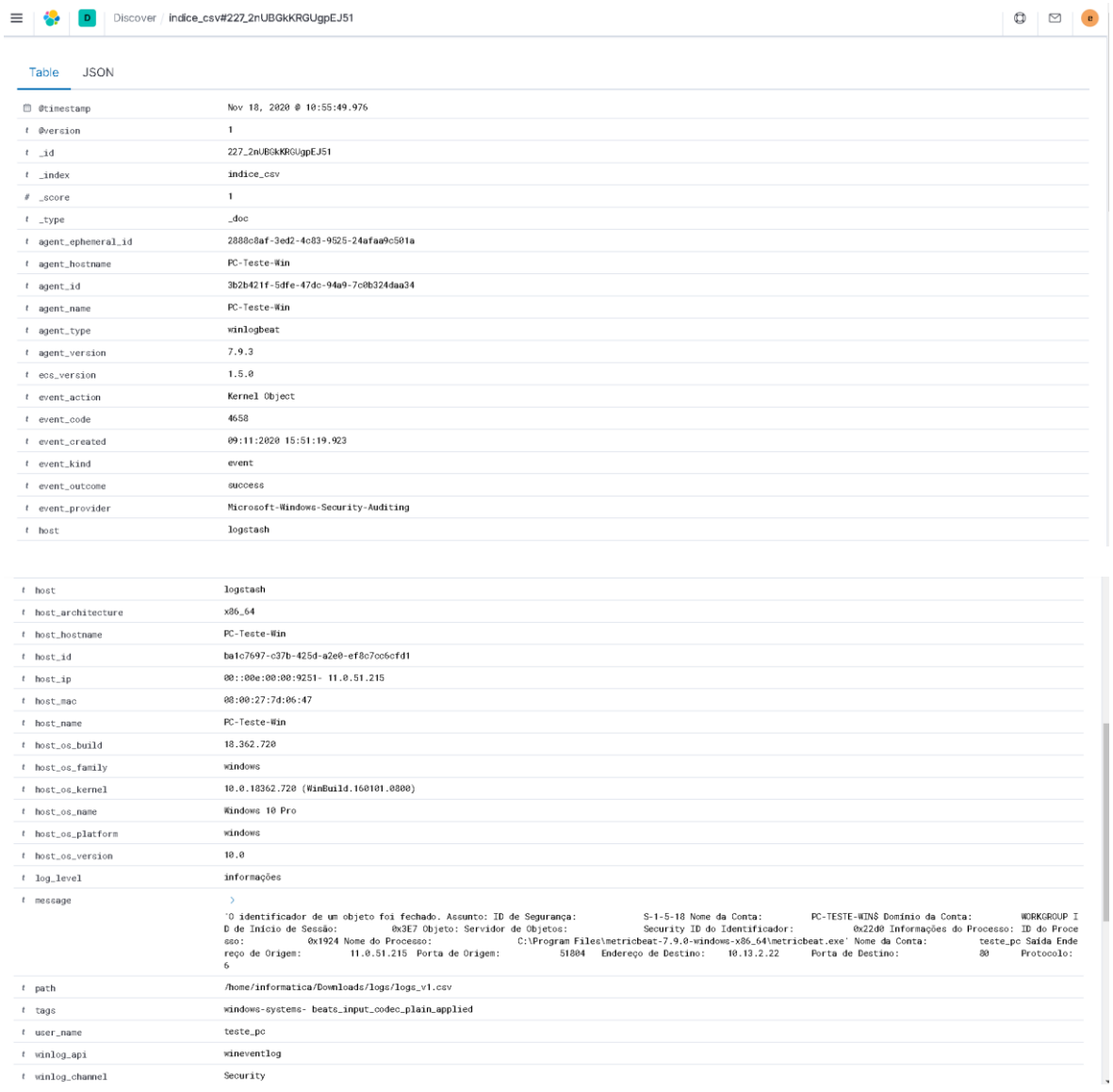


Foi necessário realizar algumas alterações ao ficheiro de configuração da pipeline descrita no Anexo K – Configuração da pipeline do Logstash, alteramos os plugins de entrada de beats para file. Nos plugin filtro utilizamos o filtro CSV para realizar o mapeamento dos campos do ficheiro, e o nome dos campos foi alterado, substituiu-se o "." Pelo "_" apresentamos o excerto do código que ilustra estas alterações:

```
input {
        file {
                start_position => "beginning"
                path => "/home/logs/logs_v1.csv"
                sincedb_path => "/dev/null"
                add_field => { "[@metadata][PC1]" => "V1" }
        }
}

filter {
        if [@metadata][PC1] {
                csv {
```

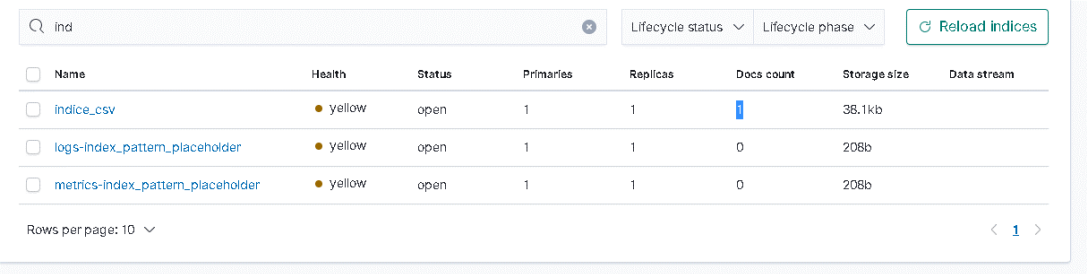
```
                    separator => ","
                    skip_header => "true"
                    columns => [ "agent_ephemeral_id", "agent_hostname", "agent_id",
"agent_name", "agent_type", "agent_version", "ecs_version", "event_action", "event_code", "event_created",
"event_kind", "event_outcome", "event_provider", "host_architecture", "host_hostname", "host_id",
"host_ip", "host_mac", "host_name", "host_os_build", "host_os_family", "host_os_kernel", "host_os_name",
"host_os_platform", "host_os_version", "log_level","message", "tags", "winlog_api", "winlog_channel",
"winlog_computer_name", "winlog_event_data_HandleId", "winlog_event_data_ObjectServer",
"winlog_event_data_ProcessId", "winlog_event_data_ProcessName",
"winlog_event_data_SubjectDomainName", "winlog_event_data_SubjectLogonId",
"winlog_event_data_SubjectUserName", "winlog_event_data_SubjectUserSid", "winlog_event_id",
"winlog_keywords", "winlog_opcode", "winlog_process_pid", "winlog_process_thread_id",
"winlog_provider_guid", "winlog_provider_name", "winlog_record_id", "winlog_task", "user_name",
"winlog_event_data_SubjectUserName" ]
          }
}
```

Para criar o ficheiro CSV, foi utilizado um *logs* real que depois se adicionou os dados que estão ilustrados na Figura 4-46. Podemos visualizar os dados do *logs* selecionado.



t	tags	windows-systems- beats_input_codec_plain_applied
t	user_name	teste_pc
t	winlog_api	wineventlog
t	winlog_channel	Security
t	winlog_computer_name	PC-Teste-Win
t	winlog_event_data_HandleId	0x22d0
t	winlog_event_data_ObjectServer	Security
t	winlog_event_data_ProcessId	0x1924
t	winlog_event_data_ProcessName	C:\Program Files\metricbeat-7.9.0-windows-x86_64\metricbeat.exe
t	winlog_event_data_SubjectDomainName	WORKGROUP
t	winlog_event_data_SubjectLogonId	0x3e7
t	winlog_event_data_SubjectUserName	teste_pc
t	winlog_event_data_SubjectUserSid	S-1-5-18
t	winlog_event_id	4658
t	winlog_keywords	Auditoria de Êxitos
t	winlog_opcode	Info
t	winlog_process_pid	4
t	winlog_process_thread_id	5.496
t	winlog_provider_guid	{54849625-5478-4994-a5ba-3e3b0328c30d}
t	winlog_provider_name	Microsoft-Windows-Security-Auditing
t	winlog_record_id	83014999
t	winlog_task	Kernel Object

O Logstash vai criar o índice automaticamente como está ilustrado na imagem seguinte, contudo foi necessário criar o *Index Pattern (Management → Stack Management → Index Patterns →Create index pattern)* no Kibana, depois de realizar esta operação os índices ficam disponíveis.



O log que está representado na imagem seguinte já está no índice, *índice_csv* do Elasticsearch e estamos a visualizar o mesmo através do Kibana. Este foi tratado pela pipeline do Logstash que converteu o ficheiro CSV e o enviou para o Elasticsearch.

t agent_hostname	PC-Teste-Win
t agent_id	3b2b421f-5dfe-47dc-94a9-7c0b324daa34
t agent_name	PC-Teste-Win
t agent_type	winlogbeat
t agent_version	7.9.3
t ecs_version	1.5.0
t event_action	Kernel Object
t event_code	4658
t event_created	09:11:2020 15:51:19.923
t event_kind	event
t event_outcome	success
t event_provider	Microsoft-Windows-Security-Auditing
t host	logstash
t host_architecture	x86_64
t host_hostname	PC-Teste-Win
t host_id	ba1c7697-c37b-425d-a2e0-ef8c7cc6cfd1
t host_ip	00::00e:00:00:9251- 11.0.51.215
t host_mac	08:00:27:7d:06:47
t host_name	PC-Teste-Win
t host_os_build	18.362.720
t host_os_family	windows
t host_os_kernel	10.0.18362.720 (WinBuild.160101.0800)
t host_os_name	Windows 10 Pro

t host_os_name	Windows 10 Pro
t host_os_platform	windows
t host_os_version	10.0
t log_level	informações
t message	'O identificador de um objeto foi fechado. Assunto: ID de Segurança: S-1-5-18 Nome da Conta: PC-TESTE-WIN$ Domínio da Conta: WORKGROUP ID de Início de Sessão: 0x3E7 Objeto: Servidor de Objetos: Security ID do Identificador: 0x22d0 Informações do Processo: ID do Processo: 0x1924 Nome do Processo: C:\Program Files\metricbeat-7.9.0-windows-x86_64\metricbeat.exe' Nome da Conta: teste_pc Saída Endereço de Origem: 11.0.51.215 Porta de Origem: 51804 Endereço de Destino: 10.13.2.22 Porta de Destino: 80 Protocolo: 6
t path	/home/informatica/Downloads/logs/logs_v1.csv
t tags	windows-systems- beats_input_codec_plain_applied
t user_name	teste_pc
t winlog_api	wineventlog
t winlog_channel	Security
t winlog_computer_name	PC-Teste-Win
t winlog_event_data_HandleId	0x22d0
t winlog_event_data_ObjectServer	Security
t winlog_event_data_ProcessId	0x1924
t winlog_event_data_ProcessName	C:\Program Files\metricbeat-7.9.0-windows-x86_64\metricbeat.exe
t winlog_event_data_SubjectDomainName	WORKGROUP
t winlog_event_data_SubjectLogonId	0x3e7
t winlog_event_data_SubjectUserName	teste_pc
t winlog_event_data_SubjectUserSid	S-1-5-18

t host_id	ba1c7697-c37b-425d-a2e0-ef8c7cc6cfd1
t host_ip	00::00e:00:00:9251- 11.0.51.215
t host_mac	08:00:27:7d:06:47
t host_name	PC-Teste-Win
t host_os_build	18.362.720
t host_os_family	windows
t host_os_kernel	10.0.18362.720 (WinBuild.160101.0800)
t host_os_name	Windows 10 Pro
t host_os_platform	windows
t host_os_version	10.0
t log_level	informações
t message	'O identificador de um objeto foi fechado. Assunto: ID de Segurança: S-1-5-18 Nome da Conta: PC-TESTE-WIN$ Domínio da Conta: WORKGROUP ID de Início de Sessão: 0x3E7 Objeto: Servidor de Objetos: Security ID do Identificador: 0x22d0 Informações do Processo: ID do Processo: 0x1924 Nome do Processo: C:\Program Files\metricbeat-7.9.0-windows-x86_64\metricbeat.exe' Nome da Conta: teste_pc Saída Endereço de Origem: 11.0.51.215 Porta de Origem: 51804 Endereço de Destino: 10.13.2.22 Porta de Destino: 80 Protocolo: 6
t path	/home/informatica/Downloads/logs/logs_v1.csv
t tags	windows-systems- beats_input_codec_plain_applied
t user_name	teste_pc
t winlog_api	wineventlog
t winlog_channel	Security
t winlog_computer_name	PC-Teste-Win
t winlog_event_data_HandleId	0x22d0

t host_os_version	10.0
t log_level	informações
t message	'O identificador de um objeto foi fechado. Assunto: ID de Segurança: S-1-5-18 Nome da Conta: PC-TESTE-WIN$ Domínio da Conta: WORKGROUP ID de Início de Sessão: 0x3E7 Objeto: Servidor de Objetos: Security ID do Identificador: 0x22d0 Informações do Processo: ID do Processo: 0x1924 Nome do Processo: C:\Program Files\metricbeat-7.9.0-windows-x86_64\metricbeat.exe' Nome da Conta: teste_pc Saída Endereço de Origem: 11.0.51.215 Porta de Origem: 51804 Endereço de Destino: 10.13.2.22 Porta de Destino: 80 Protocolo: 6
t path	/home/informatica/Downloads/logs/logs_v1.csv
t tags	windows-systems- beats_input_codec_plain_applied
t user_name	teste_pc
t winlog_api	wineventlog
t winlog_channel	Security
t winlog_computer_name	PC-Teste-Win
t winlog_event_data_HandleId	0x22d0
t winlog_event_data_ObjectServer	Security
t winlog_event_data_ProcessId	0x1924
t winlog_event_data_ProcessName	C:\Program Files\metricbeat-7.9.0-windows-x86_64\metricbeat.exe
t winlog_event_data_SubjectDomainName	WORKGROUP
t winlog_event_data_SubjectLogonId	0x3e7
t winlog_event_data_SubjectUserName	teste_pc
t winlog_event_data_SubjectUserSid	S-1-5-18
t winlog_event_id	4658

O próximo log, também foi tratado pela pipeline do Logstash que converteu o ficheiro CSV num log, na mesma também efetuou a pseudonimização dos campos (nome máquina, utilizador e IP destino) e por fim enviou o log já transformado para o Elasticsearch.

Table JSON

@timestamp	Nov 18, 2020 @ 11:38:37.095
t @version	1
t _id	CW8m23UBGkKRGUgpPIuk
t _index	indice_csv
# _score	1
t _type	_doc
t agent_ephemeral_id	2888c8af-3ed2-4c83-9525-24afaa9c501a
t agent_hostname	bbf561f0c3fbfca05facb5164a2b3c4a5311e2afb196d065cb20cfd21d778dbd
t agent_id	3b2b421f-5dfe-47dc-94a9-7c0b324daa34
t agent_name	bbf561f0c3fbfca05facb5164a2b3c4a5311e2afb196d065cb20cfd21d778dbd
t agent_type	winlogbeat
t agent_version	7.9.3
t ecs_version	1.5.0
t event_action	Kernel Object
t event_code	4658
t event_created	09:11:2020 15:51:19.923
t event_kind	event
t event_outcome	success
t event_provider	Microsoft-Windows-Security-Auditing
t host	logstash

t host	logstash
t host_architecture	x86_64
t host_hostname	bbf561f0c3fbfca05facb5164a2b3c4a5311e2afb196d065cb20cfd21d778dbd
t host_id	ba1c7697-c37b-425d-a2e0-ef8c7cc6cfd1
t host_ip	1:1:1:1:1:1:1:1, 1.1.1.1
t host_mac	08:00:27:7d:06:47
t host_name	bbf561f0c3fbfca05facb5164a2b3c4a5311e2afb196d065cb20cfd21d778dbd
t host_os_build	18.362.720
t host_os_family	windows
t host_os_kernel	10.0.18362.720 (WinBuild.160101.0800)
t host_os_name	Windows 10 Pro
t host_os_platform	windows
t host_os_version	10.0
ⓘ ip_orig_temp_key	⚠ 6d0641a9c5b4bb4a38408947dd0791f078251922fbb776e0b173cb7f58ea8adf
t log_level	informações
t message	'O identificador de um objeto foi fechado. Assunto: ID de Segurança: S-1-5-18 Nome da Conta: bbf561f0c3fbfca05facb5164a2b3c4a5311e2afb196d065cb20cfd21d778dbd $ Domínio da Conta: WORKGROUP ID de Início de Sessão: 0x3E7 Objeto: Servidor de Objetos: Security ID do Identificador: 0x22d0 Informações do Processo: ID do Processo: 0x1924 Nome do Processo: C:\Program Files\metricbeat-7.9.0-windows-x86_64\metricbeat.exe' Nome da Conta: ee23b889d7064e6dab7738ed7be5afb0307d44d804525fa2b80a575429c6cca8 Saída Endereço de Origem: 6d0641a9c5b4bb4a38408947dd0791f078251922fbb776e0b173cb7f58ea8adf Porta de Origem: 51804 Endereço de Destino: 10.13.2.22 Porta de Destino: 80 Protocolo: 6
t path	/home/informatica/Downloads/logs/logs_v1.csv
t tags	windows-systems- beats_input_codec_plain_applied
t user_name	ee23b889d7064e6dab7738ed7be5afb0307d44d804525fa2b80a575429c6cca8

t tags	windows-systems- beats_input_codec_plain_applied
t user_name	ee23b889d7064e6dab7738ed7be5afb0307d44d804525fa2b80a575429c6cca8
t winlog_api	wineventlog
t winlog_channel	Security
t winlog_computer_name	bbf561f0c3fbfca05facb5164a2b3c4a5311e2afb196d065cb20cfd21d778dbd
t winlog_event_data_HandleId	0x22d0
t winlog_event_data_ObjectServer	Security
t winlog_event_data_ProcessId	0x1924
t winlog_event_data_ProcessName	C:\Program Files\metricbeat-7.9.0-windows-x86_64\metricbeat.exe
t winlog_event_data_SubjectDomainName	WORKGROUP
t winlog_event_data_SubjectLogonId	0x3e7
t winlog_event_data_SubjectUserName	ee23b889d7064e6dab7738ed7be5afb0307d44d804525fa2b80a575429c6cca8
t winlog_event_data_SubjectUserSid	S-1-5-18
t winlog_event_id	4658
t winlog_keywords	Auditoria de Êxitos
t winlog_opcode	Info
t winlog_process_pid	4
t winlog_process_thread_id	5.496
t winlog_provider_guid	{54849625-5478-4994-a5ba-3e3b0328c30d}
t winlog_provider_name	Microsoft-Windows-Security-Auditing
t winlog_record_id	83014999
t winlog_task	Kernel Object

t host_id	ba1c7697-c37b-425d-a2e0-ef8c7cc6cfd1
t host_ip	1:1:1:1:1:1:1:1, 1.1.1.1
t host_mac	08:00:27:7d:06:47
t host_name	bbf561f0c3fbfca05facb5164a2b3c4a5311e2afb196d065cb20cfd21d778dbd
t host_os_build	18.362.720
t host_os_family	windows
t host_os_kernel	10.0.18362.720 (WinBuild.160101.0800)
t host_os_name	Windows 10 Pro
t host_os_platform	windows
t host_os_version	10.0
ⓘ ip_orig_temp_key	⚠ 6d0641a9c5b4bb4a38408947dd0791f078251922fbb776e0b173cb7f58ea8adf
t log_level	informações
t message	'O identificador de um objeto foi fechado. Assunto: ID de Segurança: S-1-5-18 Nome da Conta: bbf561f0c3fbfca05facb5164a2b3c4a5311e2afb196d065cb20cfd21d778dbd $ Domínio da Conta: WORKGROUP ID de Início de Sessão: 0x3E7 Objeto: Servidor de Objetos: Security ID do Identificador: 0x22d0 Informações do Processo: ID do Processo: 0x1924 Nome do Processo: C:\Program Files\metricbeat-7.9.0-windows-x86_64\metricbeat.exe' Nome da Conta: ee23b889d7064e6dab7738ed7be5afb0307d44d804525fa2b80a575429c6cca8 Saída Endereço de Origem: 6d0641a9c5b4bb4a38408947dd0791f078251922fbb776e0b173cb7f58ea8adf Porta de Origem: 51804 Endereço de Destino: 10.13.2.22 Porta de Destino: 80 Protocolo: 6
t path	/home/informatica/Downloads/logs/logs_v1.csv
t tags	windows-systems- beats_input_codec_plain_applied
t user_name	ee23b889d7064e6dab7738ed7be5afb0307d44d804525fa2b80a575429c6cca8
t winlog_api	wineventlog
t winlog_channel	Security
t winlog_computer_name	bbf561f0c3fbfca05facb5164a2b3c4a5311e2afb196d065cb20cfd21d778dbd

t host_os_platform	windows
t host_os_version	10.0
ⓘ ip_orig_temp_key	⚠ 6d0641a9c5b4bb4a38408947dd0791f078251922fbb776e0b173cb7f58ea8adf
t log_level	informações
t message	'O identificador de um objeto foi fechado. Assunto: ID de Segurança: S-1-5-18 Nome da Conta: bbf561f0c3fbfca05facb5164a2b3c4a5311e2afb196d065cb20cfd21d778dbd $ Domínio da Conta: WORKGROUP ID de Início de Sessão: 0x3E7 Objeto: Servidor de Objetos: Security ID do Identificador: 0x22d0 Informações do Processo: ID do Processo: 0x1924 Nome do Processo: C:\Program Files\metricbeat-7.9.0-windows-x86_64\metricbeat.exe' Nome da Conta: ee23b889d7064e6dab7738ed7be5afb0307d44d804525fa2b80a575429c6cca8 Saída Endereço de Origem: 6d0641a9c5b4bb4a38408947dd0791f078251922fbb776e0b173cb7f58ea8adf Porta de Origem: 51804 Endereço de Destino: 10.13.2.22 Porta de Destino: 80 Protocolo: 6
t path	/home/informatica/Downloads/logs/logs_v1.csv
t tags	windows-systems- beats_input_codec_plain_applied
t user_name	ee23b889d7064e6dab7738ed7be5afb0307d44d804525fa2b80a575429c6cca8
t winlog_api	wineventlog
t winlog_channel	Security
t winlog_computer_name	bbf561f0c3fbfca05facb5164a2b3c4a5311e2afb196d065cb20cfd21d778dbd
t winlog_event_data_HandleId	0x22d0
t winlog_event_data_ObjectServer	Security
t winlog_event_data_ProcessId	0x1924
t winlog_event_data_ProcessName	C:\Program Files\metricbeat-7.9.0-windows-x86_64\metricbeat.exe
t winlog_event_data_SubjectDomainName	WORKGROUP
t winlog_event_data_SubjectLogonId	0x3e7
t winlog_event_data_SubjectUserName	ee23b889d7064e6dab7738ed7be5afb0307d44d804525fa2b80a575429c6cca8
t winlog_event_data_SubjectUserSid	S-1-5-18

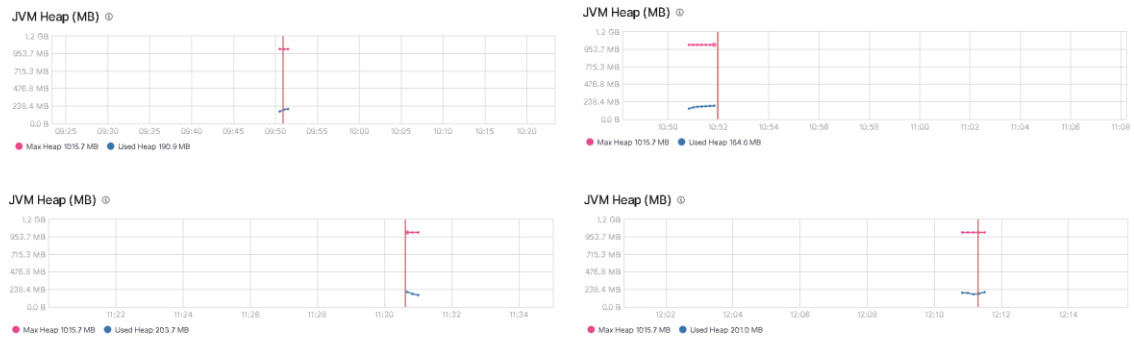
Como já foi mencionado várias vezes com a pseudonimização, configurou-se um servidor secundário, no mesmo foi criado o índice: *data_key*. A imagem seguinte apresenta os dados de recuperação para o cenário das métricas onde foi feita a pseudonimização de alguns campos.

Table JSON

@timestamp	Nov 18, 2020 @ 11:54:40.150
_id	7f57ba9a23f3a5e86459b8bdaae506bdcad660575d7d7c830b1d6401bcccca97
_index	data_key
_score	1
_type	_doc
agent.ip	00::00e:00:00:9251- 11.0.51.215
agent.ip_key	6d0641a9c5b4bb4a38408947dd0791f078251922fbb776e0b173cb7f58ea8adf
agent.name	PC-Teste-Win
agent.name_key	bbf561f0c3fbfca05facb5164a2b3c4a5311e2afb196d065cb20cfd21d778dbd
agent.user	teste_pc
agent.user_key	ee23b889d7064e6dab7738ed7be5afb0307d44d804525fa2b80a575429c6cca8
resumo	PC-Teste-Win * teste_pc * 00::00e:00:00:9251- 11.0.51.215
tags	windows-systems- beats_input_codec_plain_applied, clone_name_equip

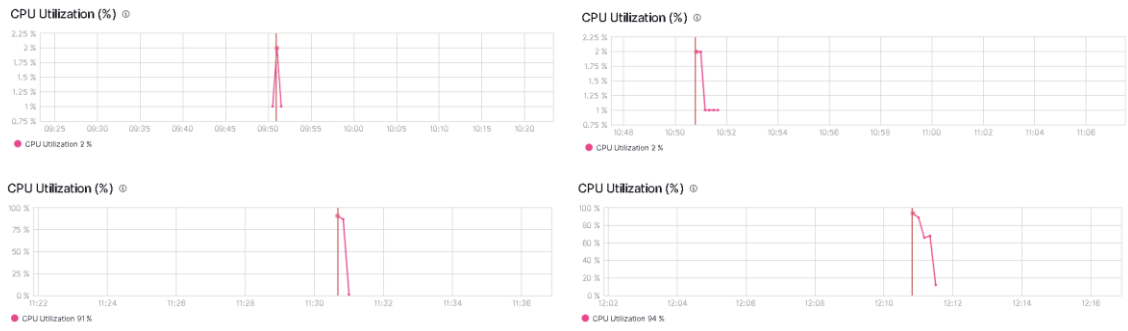
Para comprovar os dados da Tabela 4.3, disponibilizamos de seguida os gráficos criados pelo Kibana com e sem a pseudonimização. Para uma melhor compreensão agrupamos os dados por métricas/cenário. As imagens seguintes apresentam as métricas recolhidas no cenário sem a pseudonimização. De seguida apresenta-se as métricas dos Eventos recebidos.

Na próxima imagem é disponibilizado os gráficos da métrica JVM Heap para os quatro testes (1/100/1000/10000).



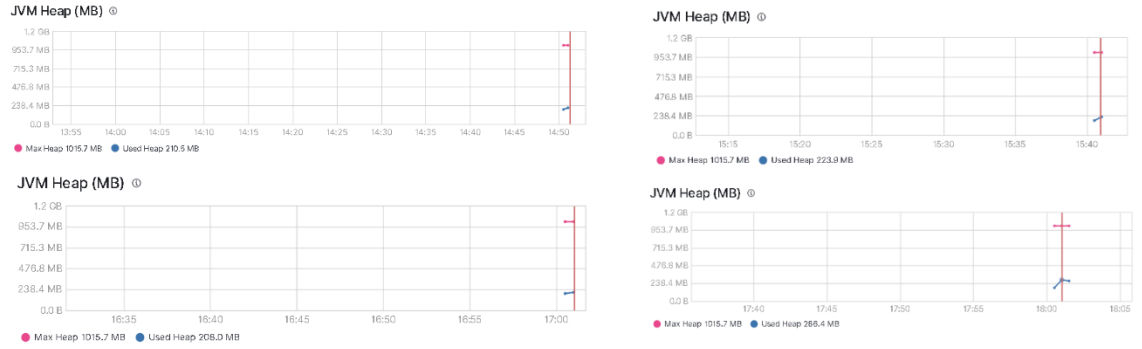
Disponibiliza-se os gráficos as métrica Eventos emitidos para os quatro testes (1/100/1000/10000).

Apresenta-se os gráficos da métrica CPU na próxima figura para os quatro testes (1/100/1000/10000).



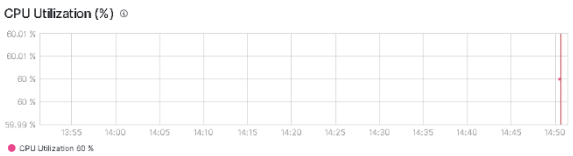
As próximas imagens apresentam as métricas recolhidas no cenário em que foi realizada a pseudonimização. Pode-se observar os gráficos da métrica Eventos recebidos para os quatro testes (1/100/1000/10000).

Na imagem seguinte é disponibilizado os gráficos da métrica JVM Heap para os quatro testes (1/100/1000/10000) no cenário em que se efetuou a pseudonimização.



Disponibiliza-se os gráficos as métrica Eventos emitidos para os quatro testes (1/100/1000/10000) no cenário em que se efetuou a pseudonimização.

Podemos visualizar os gráficos da métrica CPU na próxima figura para os quatro testes (1/100/1000/10000) no cenário em que se efetuou a pseudonimização.


CPU Utilization (%)
60.01 %
60.01 %
60 %
60 %
59.99 %
13:55
14:00
14:05
14:10
14:15
14:20
14:25
14:30
14:35
14:40
14:45
14:50
CPU Utilization 60 %


CPU Utilization (%)
100 %
75 %
50 %
25 %
0 %
15:15
15:20
15:25
15:30
15:35
15:40
CPU Utilization 95 %


CPU Utilization (%)
93.5 %
93 %
92.5 %
92 %
91.5 %
91 %
90.5 %
16:35
16:40
16:45
16:50
16:55
17:00
CPU Utilization 93 %


CPU Utilization (%)
100 %
90 %
80 %
70 %
60 %
17:40
17:45
17:50
17:55
18:00
18:05
CPU Utilization 91 %