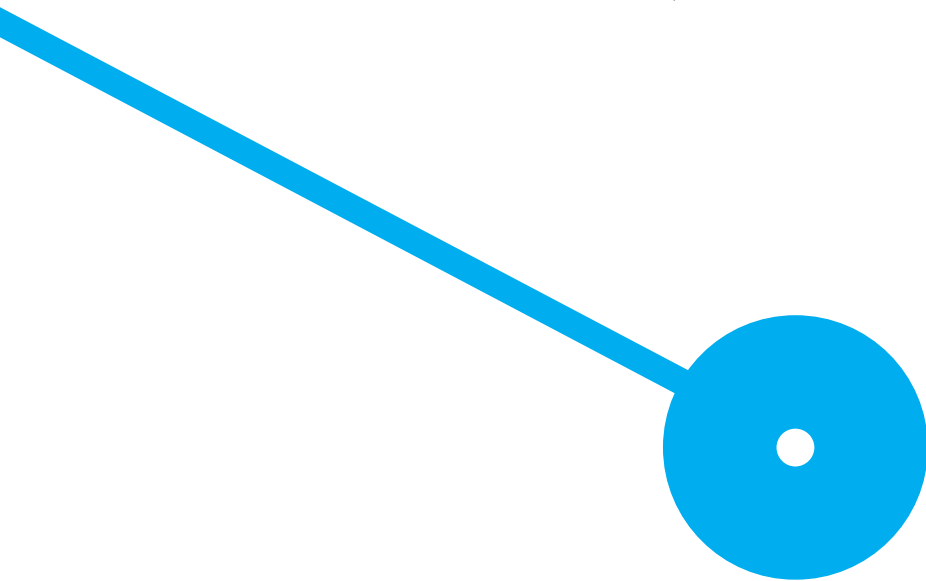




Plataforma de Testes de Resiliência em Cibersegurança

Christopher Pereira Meder

06/2022





Plataforma de Testes de
Resiliência em Cibersegurança
Christopher Pereira Meder
João Paulo Magalhães

Agradecimentos

O meu agradecimento ao meu coordenador e orientador do projeto, Doutor João Paulo Magalhães pela sua ajuda, disponibilidade e sugestões dadas durante esta Licenciatura.

A todos os meus colegas de curso que se tornaram mais do que colegas, são amigos para vida que me apoiaram neste percurso académico.

Quero também agradecer a todos os professores que tive ao longo desta jornada por tudo que nos foi ensinado.

Um especial agradecimento a minha família que reside no estrangeiro, local onde residi até aos meus 11 de idade, por me apoiarem, tentar ajudar no que pudessem e motivar a finalizar esta etapa importante da minha vida, obrigado.

Resumo

O mundo da tecnologia está em rápida evolução e expansão, e como tal, os ciberataques estão a crescer e as empresas estão em perigo a partir deles.

Este projeto foi criado para ajudar a criar plataformas para testar a resiliência contra ciberataques, ao longo deste relatório ira ser apresentado diversas ferramentas que são utilizadas no dia-a-dia nas empresas que utilizam este tipo de plataformas para se proteger contra os ataques que podem acontecer a qualquer momento.

Para alem de falar sobre diversas ferramentas, também vai ser explicado como uma plataforma para testar a resiliência pode ser criada, sendo que foi automatizado ao máximo possível e também como criar uma bateria de testar para testar a resiliência da nossa plataforma por fim tem uma demonstração da nossa plataforma em funcionamento.

Palavras-chave: Cibersegurança, ciberataques, Testes de Resiliência

Índice

Agradecimentos	3
Resumo.....	4
Índice de Figuras	6
Lista de Abreviaturas	8
1. Introdução	10
1.1 Estrutura do Relatório	11
2. Estado de Arte e Ferramentas Relacionadas	12
2.1 Ameaças Cibernéticas - Exemplos.....	15
2.1.1 Malware	15
2.1.2 Ransomware	16
2.1.3 Phishing/Social Engineering.....	17
2.1.4 Distributed denial-of-service attacks.....	18
2.2 Proteção contra ciber ataques	19
2.2.1 OpenIOC.....	19
2.2.2 STIX.....	19
2.2.3 TAXII	21
2.2.4 IDS.....	22
2.2.5 Sandbox IOCs	23
2.2.6 IPS.....	24
2.2.7 Firewalls.....	25
2.2.8 Snort.....	26
2.2.9 Cortex.....	26
2.2.10 TheHive	27
2.2.11 Malware Information Sharing Platform	28
3. Metodologia de desenvolvimento do Projeto.....	29
4. Implementação do Projeto	31
4.1 Snort	31
4.2 MISP	37
4.3 Plataforma de Testes.....	43
4.3 Testes de resiliência	50
5. Conclusão	53
Referências Bibliográficas.....	54
Anexos.....	57

Índice de Figuras

Figura 1 - Percentagem de Organizações comprometidas a pelo menos 1 ataque com sucesso - comparitech.com	13
Figura 2 - Valor dos Danos causados de Cibercrime - europeanmemories.net	14
Figura 3 - Total de Malware nos últimos 10 anos - av-test.org	15
Figura 4 - Pagamentos de Ransomware - coveware.com	16
Figura 5 - % de Phising Attacks hospedados em HTTPS - comparitech.com	17
Figura 6 – Aumento da % de ataques DDoS - securelist.com	18
Figura 7 – Esquema STIX - oasis-open.github.io	20
Figura 8 - Modelo principal TAXII - anomali.com	21
Figura 9 - Diagrama do STIX e TAXII - researchgate.net	22
Figura 10 - Funcionamento de uma Firewall - cutewallpaper.orgl	25
Figura 11 - Snort Logo	26
Figura 12 - Ciclo do TheHive	27
Figura 13 - Interação MISP - misp-project.org	28
Figura 14 - Esquema do Projeto	29
Figura 15 - Script para transferência das regras	30
Figura 16 - Informação Snort	31
Figura 17 - Escolha de Interface	32
Figura 18 - Atribuição da Rede	32
Figura 19 - Criação do backup do snort.conf	33
Figura 20 - Adicionar a rede ao ficheiro snort.conf	34
Figura 21 - Verificar se a configuração esta valida	35
Figura 22 - Incluir as Regras no Snort	35
Figura 23 - Inicio da Instalação do MISP	37
Figura 24 - Download do Script	37
Figura 25 - Fim da Instalação do MISP	38
Figura 26 - Web UI Login	38
Figura 27 - Alteração da palavra-passe	39
Figura 28 - MISP Feeds	39
Figura 29 - Criar o nosso Evento	40
Figura 30 - Evento Criado	40
Figura 31 - Adicionar objetos ao Evento	41
Figura 32 - Criação do Objeto	42
Figura 33 - Continuação da criação do Objeto	42
Figura 34 - Export final do Evento	42
Figura 35 - Criação de Scripts	43
Figura 36 - Listar Opções	44
Figura 37 - Adicionar scripts a pilha de testes	44
Figura 38 - Correr a pilha de Testes	45
Figura 39 - Remover a pilha de Testes	45
Figura 40 - Menu principal do Script	45
Figura 41 - Script de Ping	46
Figura 42 - Script DNS Lookup	47
Figura 43 - Script NMAP	47
Figura 44 - Script SNMP	48
Figura 45 - Script Login FTP	48
Figura 46 - Login em serviço SSH	49

Figura 47 - Script em funcionamento	50
Figura 48 - Alimentar a pilha de testes.....	50
Figura 49 - Scripts dentro da pilha de testes	51
Figura 50 - Resultado dos Testes I	51
Figura 51 - Resultado dos Testes II.....	52

Lista de Abreviaturas

AI – Artificial Intelligence

CERT – Computer Emergency Response Team

CIRCL – Computer Incident Response Center Luxembourg

CPU – Central Processing Unit

CSIRT – Computer Security Incident Response Team

DDoS - Distributed denial-of-service

DMZ – Demilitarized Zone

DNS - Domain Name System

FTP - File Transfer Protocol

GCC – GNU Compiler Collection

HTTPS - Hyper Text Transfer Protocol Secure

ICMP - Internet Control Message Protocol

IDE - Integrated Development Environment

IDS – Intrusion Detection System

IPS – Intrusion Protection System

IOC – Indicators of Compromise

IoT – Internet of Things

IP – Internet Protocol

ISO – Optical Disc Image

MISP - Malware Information Sharing Platform

NCIRC – NATO Computer Incident Response Capability

NIDS – Network Intrusion Detection System

NMAP - Network Mapper

OPSEC – Operational Security

OTAN/NATO – Organisation du Traité de l'Atlantique Nord ou North Atlantic Treaty

Organisation

PC – Personal Computer

PII – Personally Identifiable Information

RAM – Random Access Memory

REST API – Representational State Transfer Application Program Interface

SIEM – Security Information and Event Management

SNMP - Simple Network Management Protocol

SOC - Security Operations Center

SSH - Secure Shell

TLP – Traffic Light Protocol

TTP - Tactics, Techniques, and Procedures

URL – Uniform Resource Locator

VM – Virtual Machine

VS Code - Visual Studio Code

Wi-Fi – Wireless Fidelity

1. Introdução

A Internet é a tecnologia dominante da Era das TI, tal como o motor elétrico foi na Era Industrial. A Internet é uma rede global de redes interligadas que fornece comunicação interativa sem fios. A Internet trouxe muita coisa positiva, tal como fornecer uma comunicação eficaz utilizando serviços de correio eletrónico e de mensagens instantâneas para qualquer parte do mundo. Por exemplo, ir ao Banco ou efetuar compras *online* tornou-se muito menos complicado, as interações e transações comerciais melhoraram, poupando muito tempo entre muitas outras coisas.

Apesar do lado positivo da Internet existe também muita coisa negativa, tal como por exemplo, os ataques cibernéticos. Estes ataques têm vindo a aumentar todos os anos e estima-se que até 2025 os danos causados por cibercrime irão chegar aos 10.5 triliões de dólares em escala curta, isto é, desde 2020 todos os anos o custo desses ataques aumenta 15% [1].

Embora as empresas se tentem proteger dos ciberataques, com a evolução rápida da tecnologia, existem sempre novas superfícies de ataque. Para cada solução encontrada para mitigar um ataque podem-se abrir novas maneiras para ser atacado, fazendo com que uma proteção na empresa a 100% seja impossível.

Sendo impossível conter todos os ciberataques, as organizações e os especialistas da área da cibersegurança tem vindo a mudar de um paradigma de cibersegurança para ciberresiliência. A ciberresiliência caracteriza-se pela capacidade do negócio das organizações se manter operacional em cenários de ciberataque e qual o tempo de recuperação de um desses ataques. Neste âmbito testar a resiliência das organizações perante cenários de ciberataque torna-se importante para afinar os procedimentos de defesa e resposta a incidentes. Neste projeto propõe-se a criação de uma “Plataforma para Teste de resiliência em Cibersegurança”. O objetivo é conceber um sistema que permita criar casos de teste, de forma a averiguar o comportamento da execução dos testes, ou seja, se perante um determinado ciberataque os sistemas informáticos em uso na organização são capazes de lidar com o ataque. A execução deste objetivo pressupõe a concretização de um conjunto de passos, entre os quais destacam-se:

- A criação de um ambiente IT de testes composto por vários sistemas e mecanismos de segurança, tais como firewall e IDS (Intrusion Detection System);
- A configuração desse ambiente por forma a que o mesmo seja atualizado com as regras de segurança disponibilizadas pela comunidade (e.g. MISP)
- A criação de uma plataforma composta por:

- Scripts de teste;
- Criação de baterias de testes (junção de vários scripts de teste);
- Execução automática dos testes.

Uma plataforma de testes de resiliência é uma mais-valia para as organizações. Através da mesma as organizações podem analisar o comportamento do seu ambiente perante a ocorrência de ciberataques, determinando o seu nível de resiliência e contribuindo para a definição de práticas e procedimentos de deteção e de recuperação de ciber incidentes.

1.1 Estrutura do Relatório

No capítulo 2 é apresentado o Estado da Arte e Ferramentas Relacionadas. O capítulo 3 foca na metodologia adotada para o desenvolvimento do projeto, incluindo a escolha das ferramentas, no capítulo 4 é descrita a Plataforma de Teste de Resiliência em Cibersegurança desenvolvida e os testes realizados. Por fim, no quinto capítulo é feita a conclusão do projeto.

2. Estado de Arte e Ferramentas Relacionadas

A cibersegurança, de acordo com [7], é a proteção de sistemas conectados a internet tais como sistemas *hardware*, *software* e dados. Esta área está a crescer em importância devido ao número crescente de utilizadores, dispositivos e programas que são utilizados numa empresa também devido a dependência de dispositivos "inteligentes" como por exemplo telemóveis, televisões, e os muitos pequenos dispositivos que compõem a Internet of Things (IoT).

As medidas de cibersegurança são concebidas para combater ameaças contra sistemas e aplicações em rede, quer essas ameaças provenham de dentro ou de fora de uma organização.

Em 2020 [11], o custo médio de uma brecha de dados foi de 3.665 milhões de Euros a nível mundial, já em 2021 o custo médio aumentou para 4.026 milhões de Euros. Estes custos foram calculados com a informação dos lucros perdidos dependendo das horas em que a empresa ficou parada devido a essa brecha, o dano causado devido a essa brecha incluindo a reputação que perdeu fazendo com que ao longo dos seguintes anos a empresa em vez de estar a lucrar está a recuperar dos danos causados estagnando o processo dessa empresa e também inclui o valor da descoberta e resposta e essa brecha. Os cibercriminosos têm como alvo os PII (Personally Identifiable Information) dos clientes isto inclui nomes, moradas, números de identificação nacional, e informações sobre cartões de crédito que depois são vendidos nos mercados digitais ilegais tal como o Mercado Negro.

Com o panorama de ameaças sempre a mudar, é importante compreender como os ataques informáticos estão a evoluir e que controlos de segurança e tipos de treino funcionam. A este respeito em [2] é apresentado um estudo que revela que:

- Houve 153 milhões de novas amostras de *malware* entre Março de 2021 e Fevereiro de 2022, um aumento de quase 5% em relação ao ano anterior, que foi de 145,8 milhões.
- Em 2019, 93,6% do *malware* observado era polimórfico, o que significa que tem a capacidade de alterar constantemente o seu código para se esquivar à deteção.
- Quase 50% dos PC de empresas e 53% dos PC de consumidores que foram infetados uma vez foram reinfectados no mesmo ano.
- Um estudo de 2007 concluiu que os hackers maliciosos estavam anteriormente a atacar computadores e redes a uma taxa de um ataque a cada 39 segundos. O relatório 2020 do Centro de Reclamações contra Crimes na Internet descobriu que houve 465.177 incidentes reportados nesse ano, o que resulta num ataque bem-sucedido a cada 1,12 segundos.

- Em 2021 86,2% das organizações inquiridas foram afetadas através de um ciberataque bem-sucedido.

Na Figura 1 é apresentado a percentagem de organizações desde 2014 até 2021, que foram comprometidas a pelo menos um ataque com sucesso.

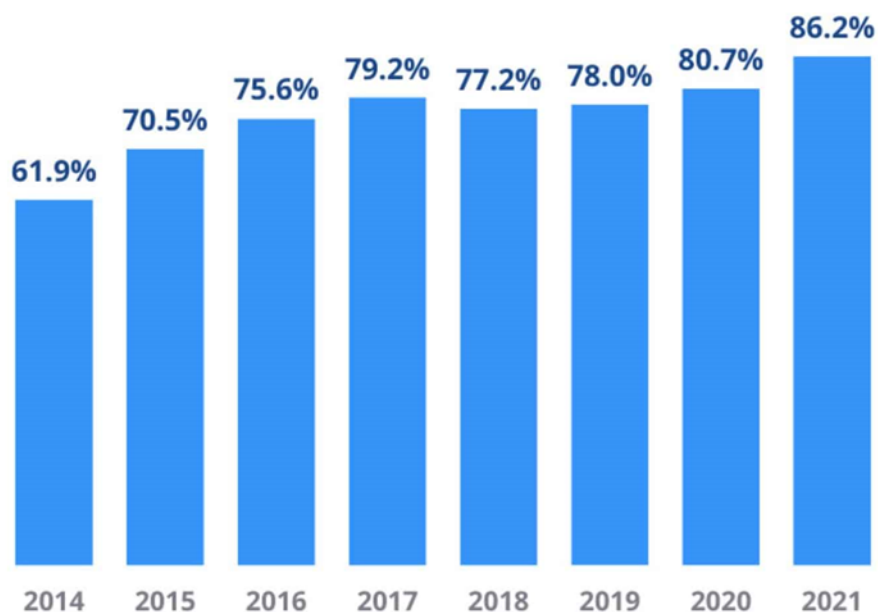


Figure 2: Percentage of organizations compromised by at least one successful attack.

Figura 1 - Percentagem de Organizações comprometidas a pelo menos 1 ataque com sucesso - comparitech.com

A Cybersecurity Ventures, num estudo apresentado em [1], espera que os custos globais do cibercrime cresçam 15% por ano durante os próximos cinco anos, atingindo 10,5 biliões de dólares de escala curta por ano até 2025, contra 3 biliões de dólares em 2015. O dano causado pelos cibercrimes é exponencialmente maior do que os danos infligidos pelas catástrofes naturais num ano, e irá ser mais rentável do que o comércio global de todas as principais drogas ilegais combinadas. Esta estimativa de custos dos danos baseia-se em números históricos do cibercrime, incluindo o crescimento recente de ano para ano das atividades de *hacking* de gangues de crime organizado e patrocinado pelo estado-nação hostil, portanto as empresas privadas e corporações, e uma superfície de ataque cibernético que será uma ordem de magnitude maior em 2025 do que é hoje. A Figura 2 revela a evolução nos danos causados pelo cibercrime entre 2001 e 2020.

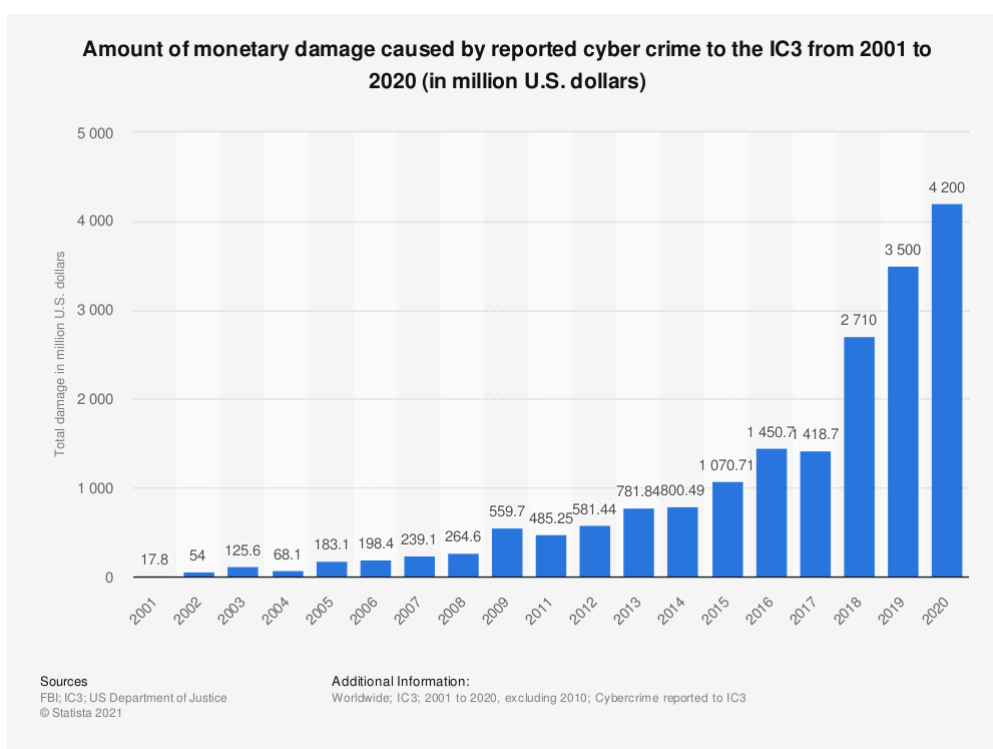


Figura 2 - Valor dos Danos causados de Cibercrime - europeanmemories.net

A complexidade do sistema de segurança, criada por tecnologias diferentes e pela falta de especialização interna, pode amplificar estes custos. Mas as organizações com uma estratégia abrangente de cibersegurança, governada pelas melhores práticas e automatizada utilizando análises avançadas, *Artificial Intelligence* e aprendizagem de máquinas, podem combater mais eficazmente as ameaças cibernéticas e reduzir o ciclo de vida e o impacto das brechas quando estas ocorrem.

2.1 Ameaças Cibernéticas - Exemplos

Ao longo desta seção são apresentados alguns exemplos das ameaças cibernéticas mais comuns.

2.1.1 Malware

O termo *malware* refere-se a variantes de software malicioso - tais como *worms*, vírus, *Trojans* e *spyware* - que fornecem acesso não autorizado ou causam danos a um computador. Os ataques de *malware* são cada vez mais "sem ficheiros", isto são ataques que apenas deixam *footprints* nas memórias, ou seja, um programa tradicional para combater *malware* lê todos os ficheiros de uma aplicação e caso houver um *footprint* já conhecido é marcado como *malware*, este malware "sem ficheiros" funciona na memória de um computador tornando difícil de ser encontrado pelas aplicações tais como ferramentas antivírus ou mesmo profissionais de segurança. A Figura 3 evidencia a evolução em número de *malwares* ao longo dos últimos 10 anos.

Total malware

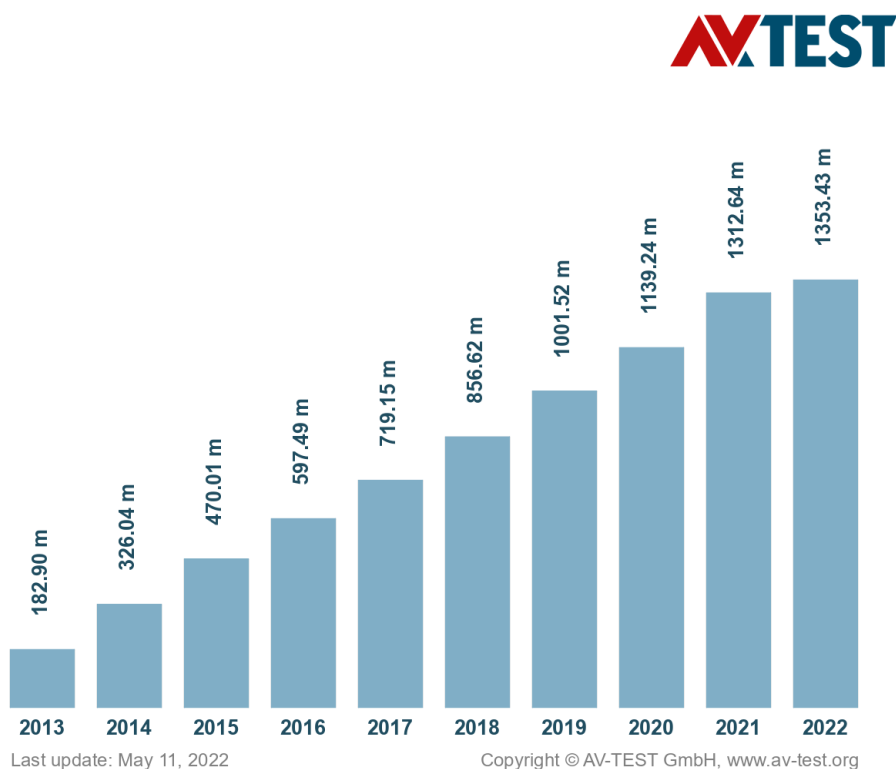


Figura 3 - Total de Malware nos últimos 10 anos - av-test.org

2.1.2 Ransomware

O *ransomware* é um tipo de *malware* que bloqueia ficheiros, dados ou sistemas, e ameaça apagar ou destruir os dados ou tornar os dados privados ou sensíveis públicos a menos que seja pago um resgate aos cibercriminosos que executaram o ataque. Os recentes ataques de resgates visaram organismos governamentais, que são mais fáceis de invadir do que as organizações privadas e estão sob pressão para pagar resgates, a fim de restaurar aplicações e websites nos quais os cidadãos dependem. A Figura 4 apresenta o custo media de resgate durante um ano desde 2018 até 2021.

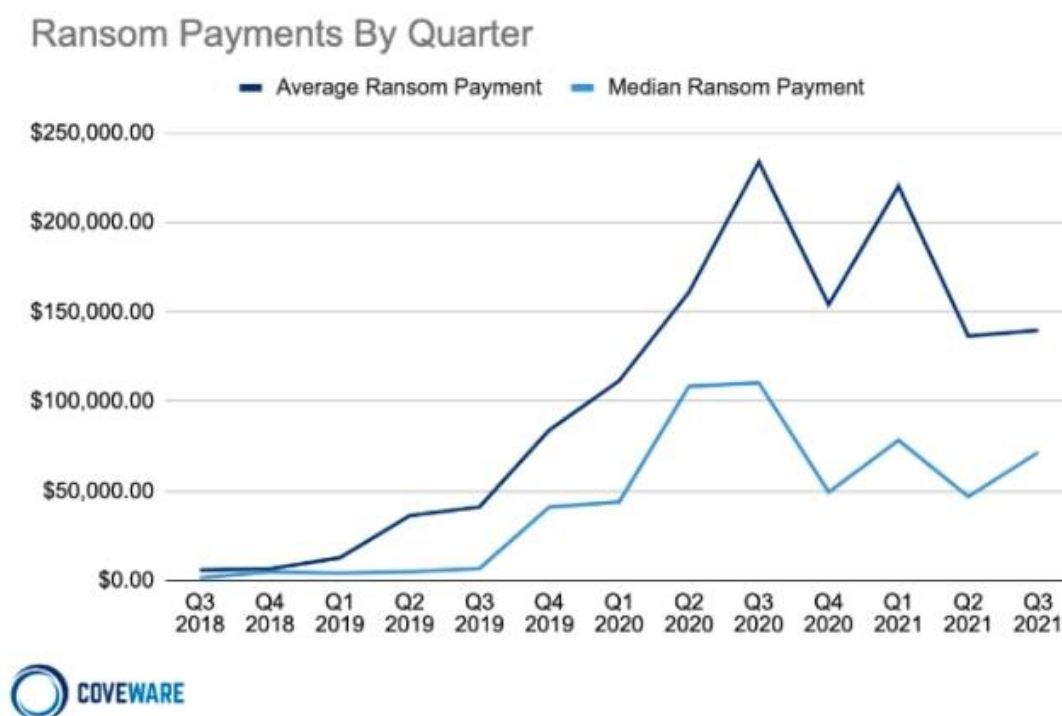


Figura 4 - Pagamentos de Ransomware - coveware.com

2.1.3 Phishing/Social Engineering

Phishing é uma forma de engenharia social que engana os utilizadores a fornecerem as suas próprias informações que identificam pessoalmente ou informações sensíveis. Em esquemas de *phishing*, e-mails ou mensagens de texto parecem ser de uma empresa legítima que pede informações sensíveis, tais como dados de cartão de crédito ou informação de login. Na Figura 5 é demonstrado a percentagem ao longo dos anos de *sites* com protocolo HTTPS, portanto um protocolo seguro, mas estes contêm ataques de *Phishing*.

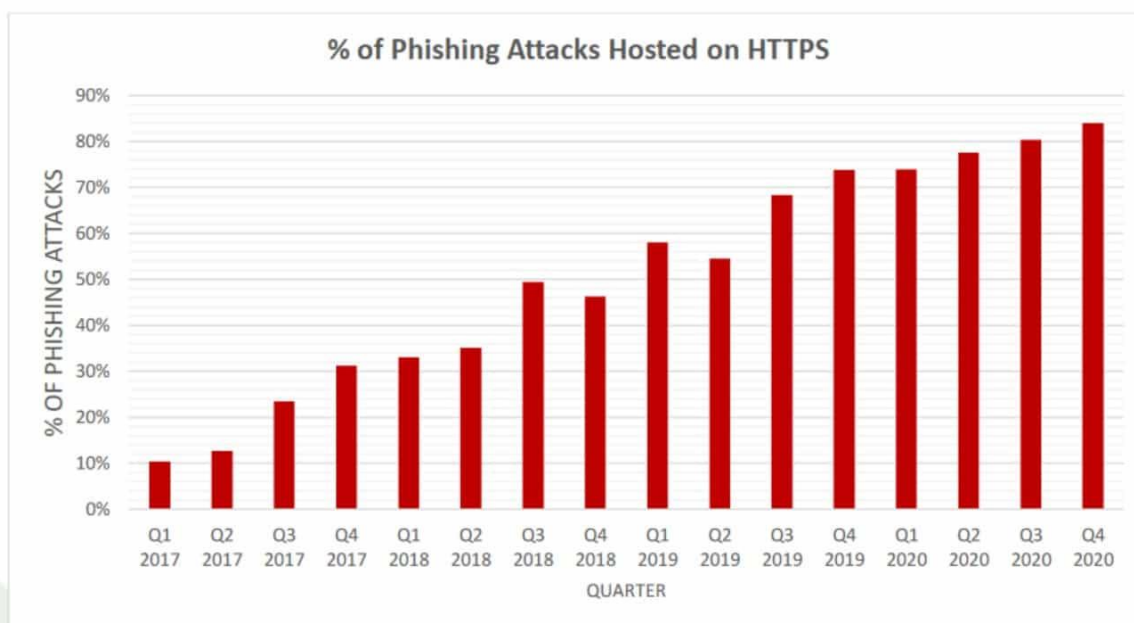


Figura 5 - % de Phishing Attacks hospedados em HTTPS - comparitech.com

2.1.4 Distributed denial-of-service attacks

Um ataque DDoS tem por objetivo causar a falha de um serviço sobrecarregando-o com tráfego, geralmente a partir de múltiplos sistemas coordenados. Na Figura 6 demonstra a percentagem de ataques ocorridos entre o ano de 2021 até 2022.

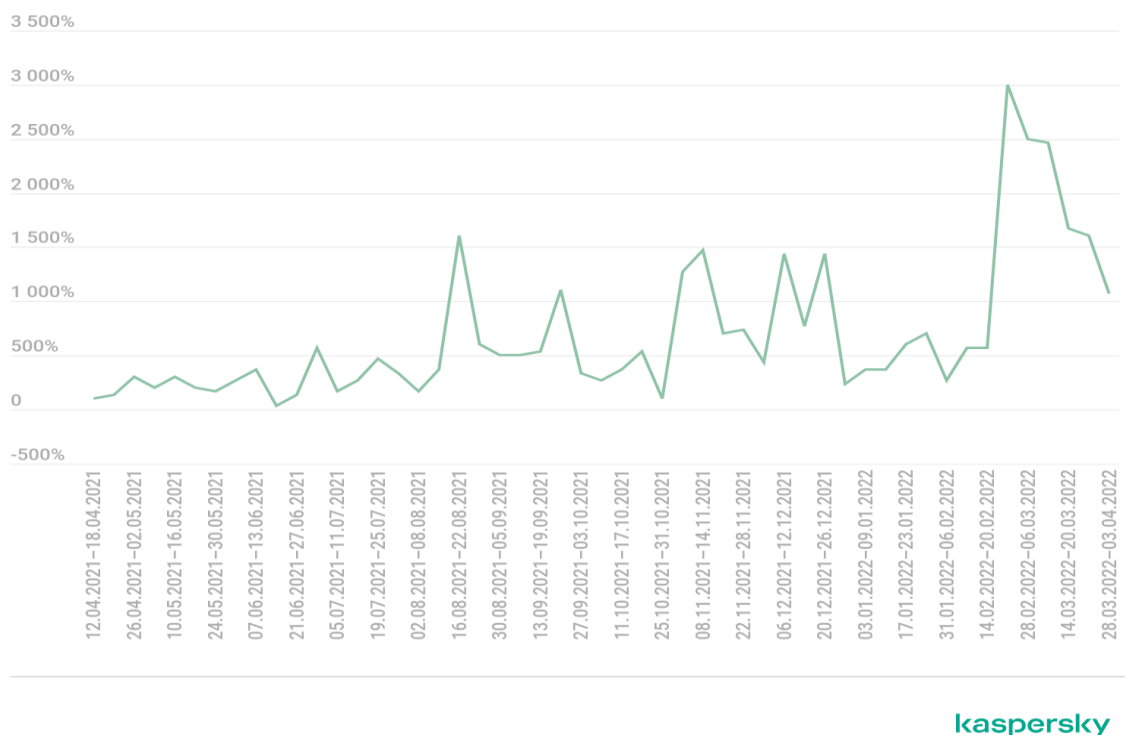


Figura 6 – Aumento da % de ataques DDoS - securelist.com

2.2 Proteção contra ciber ataques

2.2.1 OpenIOC

OpenIOC [16] é uma plataforma livre, destinada à partilha de informações sobre ameaças num formato legível por computadores. Foi desenvolvido por uma empresa americana de ciber-segurança MANDIANT em Novembro de 2011. Está escrito em eXtensible Markup Language (XML) e pode ser facilmente customizado para que as pessoas que respondem a incidentes possam traduzir os seus conhecimentos para um formato comum.

Os métodos convencionais de deteção de falhas de segurança já não são suficientes, uma vez que as assinaturas simples dos IoCs se tornaram muito fáceis de ultrapassar para um intruso. É importante mencionar que os IoCs não são assinaturas, e não se destinam a funcionar como uma assinatura funcionaria, um IoC destina-se a ser utilizado em combinação com a inteligência humana. Várias organizações precisam de ser capazes de comunicar sobre como detetar intrusos nos seus sistemas e nas suas redes, utilizando um formato facilmente compreensível pela máquina permite livrar-se de um atraso humano na partilha de informações.

As organizações têm acesso aos últimos IOCs partilhados por outras organizações. Estes IOCs podem ser facilmente aproveitados por diversas ferramentas de deteção de ameaças, permitindo deteção de ameaças em tempo real.

2.2.2 STIX

Structured Threat Information Expression (STIX) [17] é um formato de linguagem e de serialização desenvolvido pelo MITRE e pelo *Cyber Threat Intelligence* (CTI) da OASIS para a descrição de informações sobre ameaças cibernéticas. [12] Tem sido adotado como um padrão internacional por várias comunidades e organizações de partilha de informações. Foi concebida para ser partilhada através do TAXII, mas pode ser partilhada por outros meios. O STIX está estruturado de modo que os utilizadores possam descrever ameaças.



Figura 7 – Esquema STIX - oasis-open.github.io

- *Attack Pattern* – Um tipo de TTP que descreve as formas como os inimigos tentam comprometer os alvos.
- *Campaign* – Um agrupamento de comportamentos contraditórios que descreve um conjunto de atividades ou ataques maliciosos (por vezes chamados ondas) que ocorrem durante um período de tempo contra um conjunto específico de alvos.
- *Couse of Action* – Uma recomendação de um produtor de inteligência a um cliente sobre as ações que este pode tomar em resposta a essa inteligência.
- *Indicator* – Contém um padrão que pode ser utilizado para detetar atividade cibernética suspeita ou maliciosa.
- *Intrusion Set* – Um conjunto agrupado de comportamentos e recursos hostis com propriedades comuns que se acredita serem orquestrados por uma única organização.
- *Threat Actor* – Indivíduos, grupos ou organizações reais que se acredita estarem a trabalhar com intenções maliciosas.
- *Malware* - Um tipo de TTP que representa código malicioso.
- *Identity* – Indivíduos, organizações ou grupos reais, bem como classes de indivíduos, organizações, sistemas ou grupos.

- *Tool* – Software legítimo que pode ser utilizado por agentes de ameaça para executar ataques.
- *Vulnerability* - Um erro no software que pode ser diretamente utilizado por um hacker para obter acesso a um sistema ou rede.[17]

2.2.3 TAXII

Trusted Automated eXchange of Intelligence Information [12], define como a informação sobre ameaças cibernéticas pode ser partilhada através de serviços e trocas de mensagens. Foi criado especificamente para apoiar a informação STIX, o que faz através da definição de uma API que se alinha com modelos comuns de partilha. Os três modelos principais para o TAXII incluem:

- *Hub and spoke* – um repositório de informação
- *Source/subscriber* – uma única fonte de informação
- *Peer-to-peer* – múltiplos grupos partilham informação

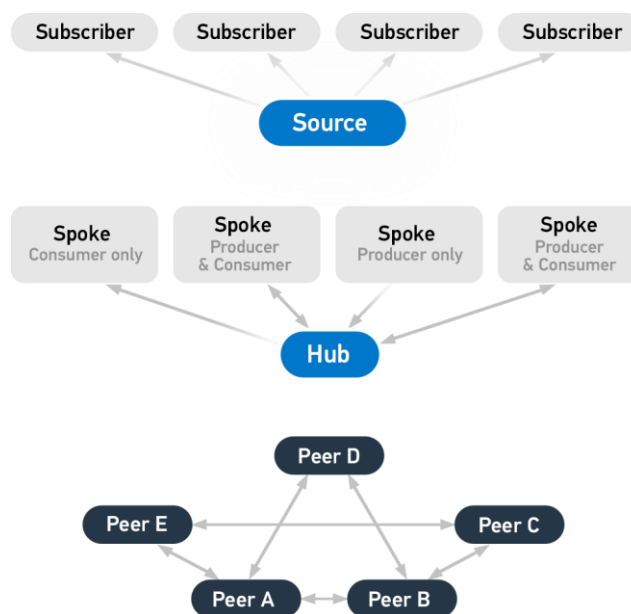


Figura 8 - Modelo principal TAXII - anomali.com

O TAXII define quatro serviços por padrão, sendo que o utilizador pode adicionar os que pretender. Os 4 serviços são:

- *Discovery* – uma forma de aprender que serviços uma entidade apoia e como interagir com eles
- *Collection Management* – uma forma de conhecer e solicitar adesões a recolhas de dados
- *Inbox* – uma forma de receber conteúdo

- *Poll* – uma forma de solicitar conteúdo

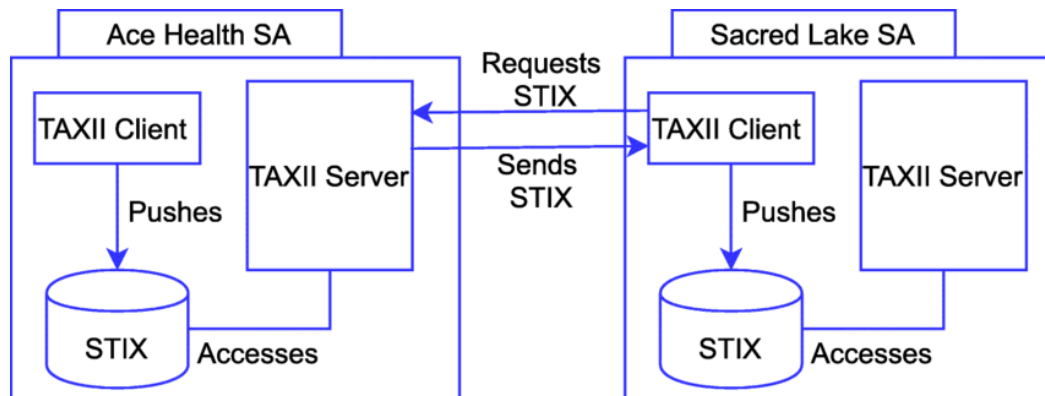


Figura 9 - Diagrama do STIX e TAXII - researchgate.net

2.2.4 IDS

Um IDS ^[14] é um sistema que monitoriza o tráfego de rede em busca de atividades suspeitas e emite alertas quando tal atividade é descoberta. É uma aplicação de software que analisa uma rede ou um sistema em busca da atividade suspeita ou violação de políticas. Qualquer risco ou violação maliciosa é normalmente comunicada a um administrador ou recolhida de forma centralizada utilizando um sistema SIEM. Um sistema SIEM [20] importa resultados de múltiplas fontes e utiliza técnicas de filtragem de alarmes para diferenciar a atividade maliciosa dos falsos alarmes.

Existe 5 tipos de IDS [14]:

- *Network Intrusion Detection System (NIDS)*: são estabelecidos num ponto planeado dentro da rede para examinar o tráfego de todos os dispositivos da rede. Este realiza uma observação de tráfego de passagem em toda a subrede e faz corresponder o tráfego que é passado nas subredes à recolha de ataques conhecidos.
- *Host Intrusion Detection System (HIDS)*: correr em *hosts* ou dispositivos independentes na rede. Um HIDS monitoriza os pacotes de entrada e saída apenas a partir do dispositivo e alertará o administrador se for detetada atividade suspeita ou maliciosa.
- *Protocol-based Intrusion Detection System (PIDS)*: é composto por um sistema que reside consistentemente na parte da frente de um servidor, controlando e interpretando o protocolo entre um utilizador/dispositivo e o servidor.
- *Application Protocol-based Intrusion Detection System (APIDS)*: é um sistema que geralmente se encontra dentro de um grupo de servidores.

- *Hybrid Intrusion Detection System*: é feita através da combinação de duas ou mais abordagens de IDSs.

E tem 2 tipos de Métodos para detetar:

- *Signature-based Method*: deteta os ataques com base nos padrões específicos, tais como número de bytes 1 ou 0 no tráfego da rede. Também deteta com base na já conhecida sequência de instruções maliciosas que é utilizada pelo *malware*.

- *Anomaly-based Method*: foi introduzido para detetar ataques de *malware* desconhecidos uma vez que novos *malwares* são desenvolvidos rapidamente utiliza *Machine Learning* para criar um modelo de atividade de confiança e tudo o que vem a seguir é comparado com esse modelo e é declarado suspeito se não for encontrado no modelo.

2.2.5 Sandbox IOCs

Os IOCs gerados por Sandbox são uma fonte subutilizada de inteligência de ameaças, devido à dificuldade de extrair IOCs acionáveis e de confiança de uma forma eficiente.

Para tal a VMRay foi criada com o objetivo de derrotar o traço mais assustador do malware: a sua capacidade de detetar e escapar aos métodos tradicionais de monitorização, que deixam sinais evidentes da sua presença no ambiente de análise.

Uma sandbox de malware [19] que analisa uma ameaça recolhe pedaços de dados forenses que foram observados durante o tempo de execução da análise. Estes dados recolhidos, frequentemente referidos como "artefactos de análise", incluem normalmente ficheiros, URLs, IPs, processos, e entradas de registo que foram utilizados, criados, ou modificados como parte da execução de *malware*.

Usando o VMRay Analyzer é automatizado o processo de extração de IOCs dos artefactos de análise, assinalando os artefactos relevantes como IOCs. Por exemplo, um URL utilizado como isco para fazer o download de ficheiros será sinalizado como um IOC. Isto significa que os IOCs são agora definidos como um subconjunto de artefactos, acrescentando a cada artefacto uma bandeira "IOC".

2.2.6 IPS

Um IPS [15] é uma aplicação de segurança de rede que monitoriza as atividades de rede ou sistema para atividades maliciosas. As principais funções dos sistemas de prevenção de intrusão são identificar a atividade maliciosa, recolher informação sobre esta atividade, denunciá-la e tentar bloqueá-la ou impedi-la. Tal como um IDS um IPS tem 4 tipos de prevenção:

- *Network-based intrusion prevention system (NIPS)*: monitoriza toda a rede em busca de tráfego suspeito, analisando a actividade do protocolo.
- *Wireless intrusion prevention system (WIPS)*: monitoriza uma rede sem fios para detetar tráfego suspeito, analisando os protocolos de rede sem fios.
- *Network behavior analysis (NBA)*: examina o tráfego de rede para identificar ameaças que geram fluxos de tráfego invulgares, tais como ataques de negação de serviço distribuídos, formas específicas de malware e violações de políticas.
- *Host-based intrusion prevention system (HIPS)*: é um pacote de software incorporado que opera um único anfitrião para atividades duvidosas através da digitalização de eventos que ocorrem dentro desse hospedeiro.

As principais diferenças [14] entre um IPS e um IDS são que os IPS encontram-se em linha e são capazes de prevenir ou bloquear ativamente as intrusões que são detetadas, pode tomar tais ações como enviar um alarme, eliminar pacotes maliciosos detetados, reiniciar uma ligação ou bloquear o tráfego do endereço IP ofensivo e um IPS também pode detetar erros de verificação de redundância periódica (CRC), desfragmentar fluxos de pacotes, mitigar problemas de sequenciamento TCP e limpar opções indesejadas de transporte e camada de rede.

2.2.7 Firewalls

As *firewalls* [13] existem desde finais dos anos 80 e começaram como filtros de pacotes, que eram redes criadas para examinar pacotes, ou bytes, transferidos entre computadores. Embora as *firewalls* de filtragem de pacotes ainda estejam em uso hoje em dia, as *firewalls* percorreram um longo caminho à medida que a tecnologia se desenvolveu ao longo das décadas.

Um IDS e uma Firewall estão ambos relacionados com a segurança da rede, mas um IDS é diferente de uma firewall, uma vez que uma firewall procura intrusões no exterior a fim de as impedir de acontecer. As *firewalls* restringem o acesso entre redes para evitar intrusões e, se um ataque for de dentro da rede, não dá sinal. Um IDS descreve uma suspeita de intrusão depois de ter ocorrido e depois assinala um alarme.

Uma Firewall é uma parte necessária de qualquer arquitetura de segurança e retira o trabalho de suposição das proteções de nível de hospedeiro e confia-as ao seu dispositivo de segurança de rede. As *Firewalls*, e especialmente as *Next Generation Firewalls*, concentram-se no bloqueio de malware e ataques de camada de aplicação, juntamente com um IPS integrado, estas *Next Generation Firewalls* podem reagir de forma rápida e sem problemas para detetar e reagir a ataques externos em toda a rede. Podem definir políticas para melhor defender a sua rede e realizar avaliações rápidas para detetar atividade invasiva ou suspeita, como *malware*, e bloqueá-la.

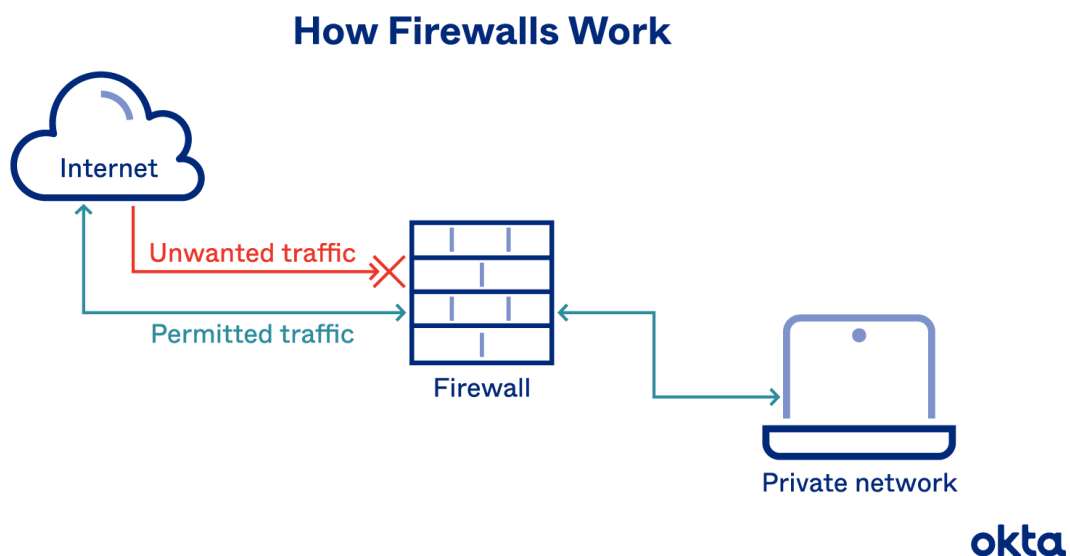


Figura 10 - Funcionamento de uma Firewall - cutewallpaper.org/

2.2.8 Snort

O Snort [8] é o maior IPS *open source* do mundo. Ele utiliza uma série de regras que ajudam a definir a atividade maliciosa da rede e utiliza essas regras para encontrar pacotes que correspondem a elas e gera alertas para os utilizadores.

O Snort também pode ser implantado em rede para parar estes pacotes. O Snort tem três usos primários: Como um *sniffer* de pacotes como o *tcpdump*, como um *logger* de pacotes que é útil para a depuração do tráfego de rede, ou pode ser usado como um sistema completo de prevenção de intrusão de rede.

No dia 21 de Junho de 2021 foi lançado a versão 3 do Snort trazendo atualização com melhorias e novas características resultando num melhor desempenho, processamento mais rápido, melhor escalabilidade para a sua rede e uma variedade de mais de 200 plugins para que os utilizadores possam criar uma configuração personalizada para a sua rede.



Figura 11 - Snort Logo

2.2.9 Cortex

Cortex [5], é um software *open source*, escrito em Scala e gratuito, foi criado pelo TheHive Project para este mesmo fim. Os parâmetros observáveis, tais como IP e endereços de e-mail, URLs, nomes de domínio, ficheiros ou *hashes*, podem ser analisados um-a-um ou em modo de volume utilizando uma interface Web. Os analistas também podem automatizar estas operações graças ao Cortex REST API.

Cortex tenta resolver problemas comuns frequentemente encontrados por SOCs, CSIRTs e investigadores de segurança no decurso da inteligência de ameaças, forense digital e resposta a incidentes.

2.2.10 TheHive

O TheHive [9] é uma Plataforma de Resposta a Incidentes de Segurança 3-em-1 sendo elas TheHive, Cortex e MISP para além disso é uma plataforma gratuita, concebida para facilitar a vida aos SOCs, CSIRTs, CERTs e a qualquer profissional de segurança da informação que lide com incidentes de segurança que necessitem de ser investigados e de ser rapidamente tratados.

Também permite analisar dezenas ou centenas de observáveis em poucos cliques, aproveitando uma ou várias instâncias do Cortex, dependendo das nossas necessidades e requisitos de desempenho do OPSEC.

TheHive tem a capacidade de identificar automaticamente os objetos observáveis que já tenham sido vistos em casos anteriores. Os observáveis também podem ser associados a um TLP e à fonte que os forneceu ou criou utilizando etiquetas. Os analistas podem também marcar facilmente os observáveis como IOCs e isolar-lhos utilizando uma consulta de pesquisa e depois exportá-los para pesquisa num SIEM ou outros armazéns de dados.

Para além disso pode ser configurado para importar eventos de uma ou múltiplas instâncias de MISP. Também pode-se utilizar TheHive para exportar casos como eventos MISP para um ou vários servidores MISP.

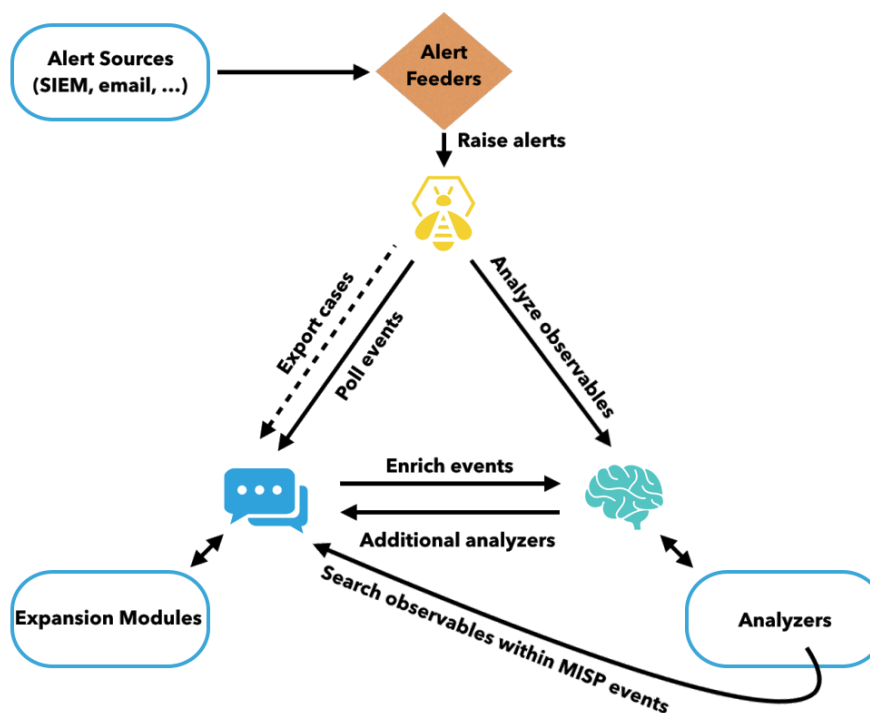


Figura 12 - Ciclo do TheHive

2.2.11 Malware Information Sharing Platform

Malware Information Sharing Platform mais conhecido como MISP [3] é uma plataforma de código aberto que permite a partilha, armazenamento e correlação de IOC de ataques direcionados, informações sobre ameaças, informações sobre fraudes financeiras, informações sobre vulnerabilidades ou mesmo informações sobre contraterrorismo.

MISP foi desenvolvido por uma equipa de programadores da CIRCL, Forças Armadas da Bélgica, OTAN e NCIRC.

MISP ajuda as equipas de segurança a ingerir e analisar dados de ameaças sobre ataques de *malware* detetados, criando automaticamente ligações entre *malware* e as suas características, e armazenando dados num formato estruturado. Além disso, o MISP [21] também ajuda a elaborar as regras para os NIDS e permite a partilha de informação *malware* com outros utilizadores da plataforma.

A estrutura desta plataforma consiste em eventos, *feeds*, comunidades e subscritores. Um evento é uma entrada de ameaça contendo informações relacionadas com a ameaça e os IOCs associados. Uma vez criado um evento, um utilizador atribui-o a um *feed* específico que funciona como uma lista centralizada de eventos pertencentes a uma organização específica e contendo determinados eventos ou especificações de agrupamento.

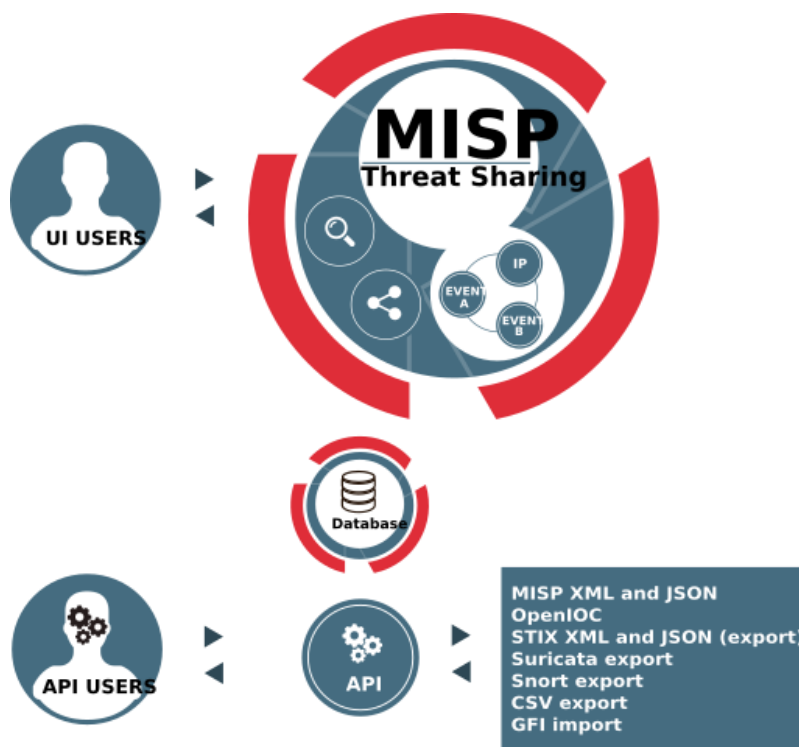


Figura 13 - Interação MISP - misp-project.org

3. Metodologia de desenvolvimento do Projeto

Neste capítulo será explicado a metodologia que foi utilizada para o desenvolvimento do Projeto.

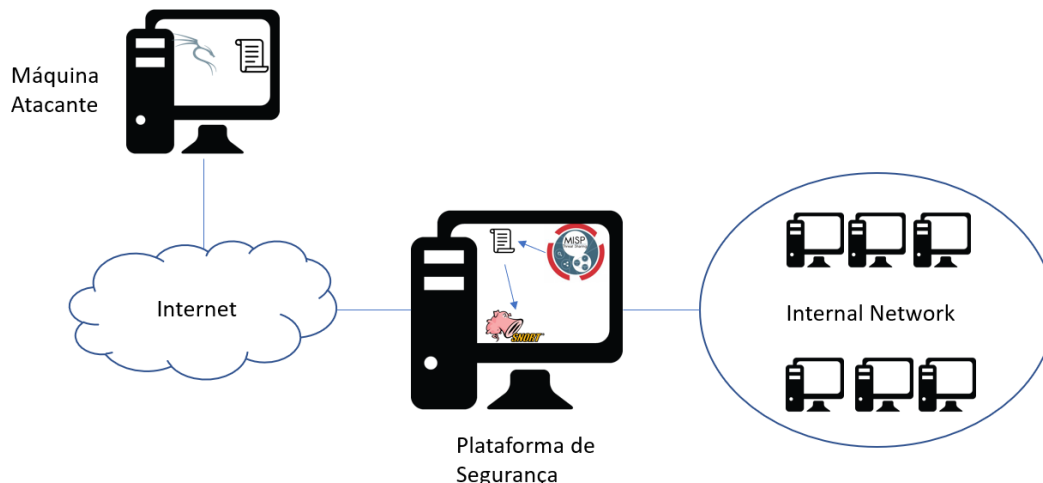


Figura 14 - Esquema do Projeto

No esquema da Figura 15 podemos ter uma visão de como foi implementado o projeto. Na máquina cuja o nome é *Kali* temos o sistema operativo Kali Linux que irá correr a bateria de testes, a máquina cuja o nome é Plataforma de Teste será a máquina onde a plataforma o MISP e o SNORT estão instalados. O SNORT é atualizado automaticamente com base num um *script* que efetua a transferência das regras a partir do MISP.

A premissa de um projeto assume uma visão preliminar do mesmo, onde fatores sem prova são considerados verdadeiros para fins de planeamento. Inicialmente existe um esboço do trabalho a realizar que, progressivamente, se vai adaptando de modo a acompanhar as descobertas e implementações feitas. O plano inicial para criar a “plataforma de teste de resiliência” era criar a nossa própria rede interna com vários computadores ligados a essa rede. Nisso ficou claro que precisamos de várias máquinas para criar a nossa rede.

Para as máquinas seria possível utilizar qualquer sistema operativo que existe, foi utilizado o sistema operativo de Ubuntu pois há mais suporte no sentido de haver mais informação em caso de existir problemas para a criação do projeto. Agora já temos uma pequena ideia de como a estrutura irá ser.

Com conhecimentos adquiridos ao longo dos anos já havia uma ideia de que seria necessário haver uma máquina que iria fazer a interceção da rede, neste caso uma máquina que irá utilizar o *Snort*. Para além disso também existe uma máquina que irá servir como a bateria de teste pois já tem todas as ferramentas instaladas por defeito, essa máquina será então o Kali.

A partir do esquema da solução passou-se à fase de implementação. Ao longo da implementação começaram a aparecer diversos problemas sendo elas que a máquina física que hospedava essas máquinas não aguentava com todas as VM ao mesmo tempo. Para ultrapassar o problema reduziu-se o número de máquinas a utilizar, ficando então apenas a máquina para os testes e a máquina com o *Snort*.

Para a implementação da plataforma optou-se pela utilização de Bash scripting. A escolha recai após uma análise de várias alternativas e considerou-se nessa análise a simplicidade de implementação e manutenção que o Bash script permite. A plataforma propriamente dita é apresentada mais à frente neste relatório.

Outra *feature* desenvolvida no âmbito do projeto foi a automatização da transferência das regras do MISP para o IDS. Como não há informação suficiente acerca disso na Internet tornou-se difícil a implementação. Foi ponderada a utilização do *PyMISP* que é ferramenta mais comum para fazer tal coisa. Mas tal como a pouca informação que havia para transferir dados para o ambiente de testes de segurança numa forma automática também havia pouca informação de como utilizar o *PyMISP*, o que se revelou uma limitação. Lendo e relendo a documentação do MISP foi descoberta uma maneira de fazer a transferência das regras de uma forma automatizada tirando partido do *cURL*. O *script* apresentado na Figura 16 foi criado apenas utilizando a ferramenta do *cURL* e só pode ser executado pelo administrador. O script permite retornar em formato de *Snort* um ficheiro com um limite de 20.000 regras. Após a transferência do ficheiro ele irá automaticamente guardar as regras na pasta onde todas as regras do *Snort* estão guardadas, atualizando desta forma o *Snort*.

```
#!/bin/bash
curl -d '{"returnFormat":"snort","page":"1","limit":"20000"}'
-H "Authorization: I7TzZggfm8QhrQ2o0ZeRck0oYqsrMQwQQx34rbjF"
-H "Accept: application/json"
-H "Content-type: application/json"
-X POST https://localhost/attributes/restSearch >> /etc/snort/rules/misp_attribues.rules
-k
```

Figura 15 - Script para transferência das regras

4. Implementação do Projeto

Após existir uma ideia do projeto que se pretende criar e com todas as pesquisas efetuadas, é necessário começar a implementação do mesmo.

Para tal, começa-se com a instalação das nossas VM com os sistemas operativos Kali Linux e Ubuntu.

4.1 Snort

Sendo que os sistemas operativos estão a funcionar começa-se com a instalação do *Snort*.

Para começar a instalação da aplicação utiliza-se o seguinte comando:

```
sudo apt-get install snort -y
```

Ao correr esse comando no terminal, será apresentado a informação cujo aparece na Figura 16.

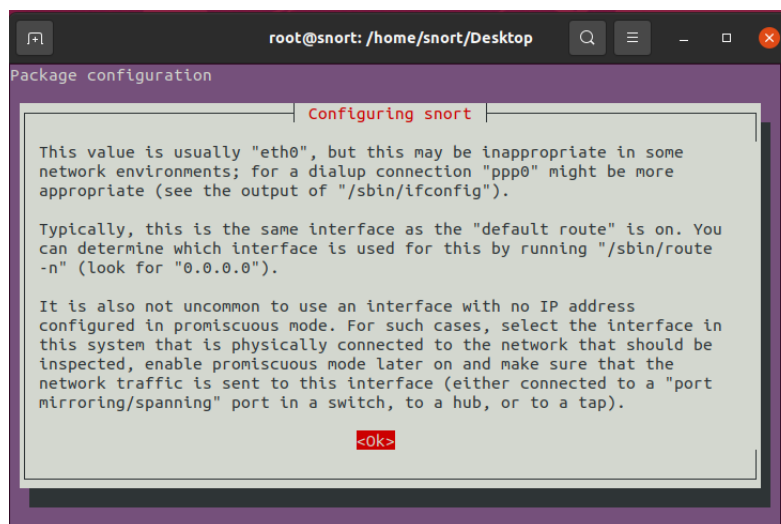


Figura 16 - Informação Snort

Utilizando a tecla “TAB” e “Enter” avança-se para a página seguinte que irá perguntar em que *interface* é pretendido que o Snort corra, neste caso pretende-se que o *Snort* corra na interface de “enp0s8”.

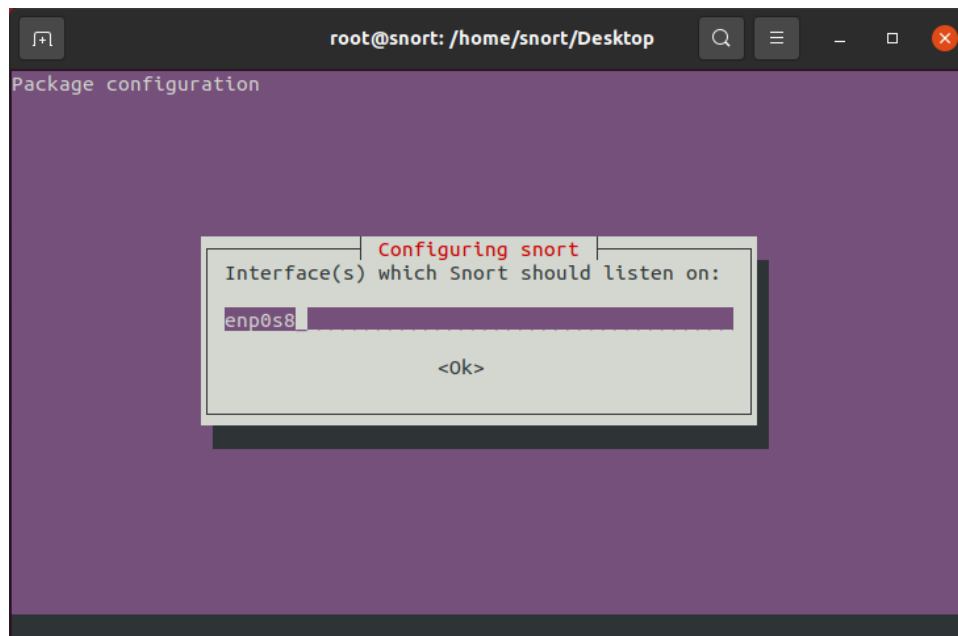


Figura 17 - Escolha de Interface

De seguida é necessário introduzir a rede que será monitorizada pela Snort. Neste projeto a rede irá ser 10.10.10.0/24 que significa que todos os IPs de 1-254 dentro da rede 10.10.10.0/24 irão ser monitorizados (Figura 18).

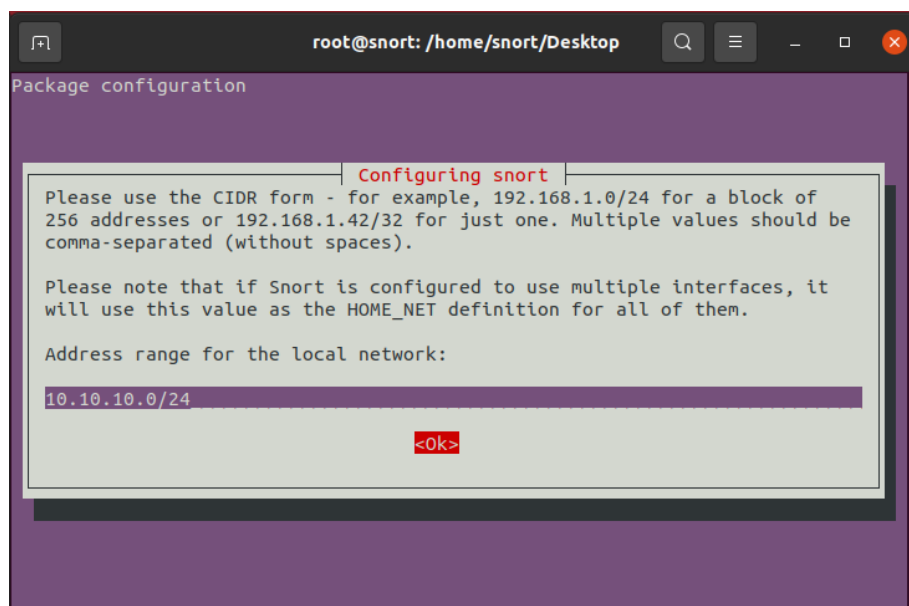


Figura 18 - Atribuição da Rede

Quando a instalação acabar não sendo necessário acede-se a pasta onde o Snort foi instalado para criar um *backup* da configuração do *Snort* para que caso haja um problema já existe um ficheiro pronto para substituir sem ter que voltar a instalar tudo do zero.

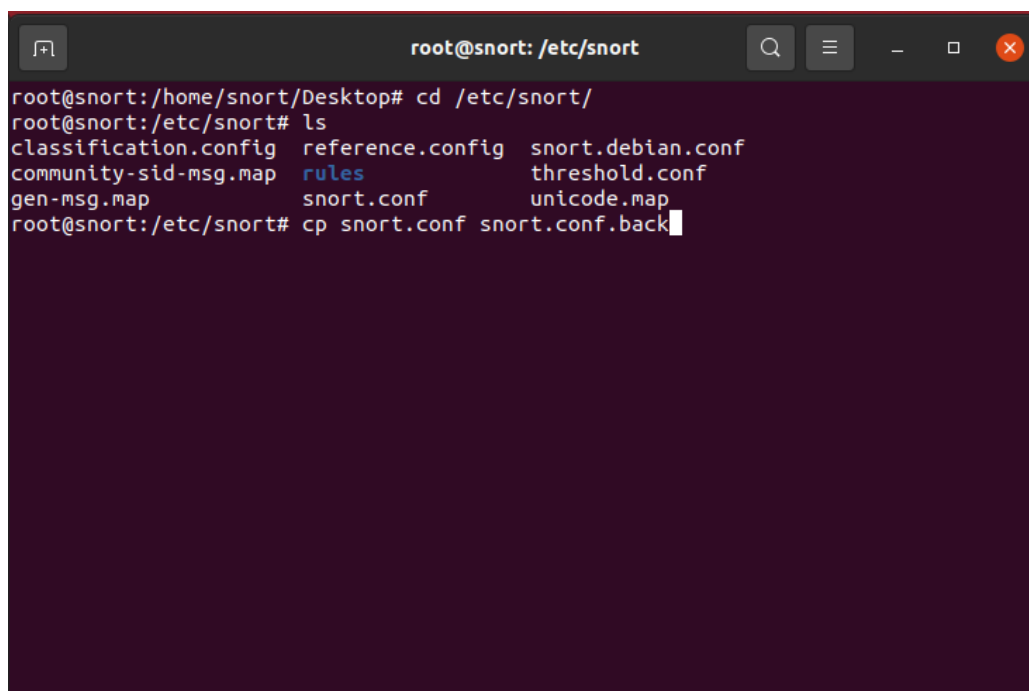
Para criar o ficheiro de *backup* introduz-se o seguinte comando no terminal para aceder ao sítio onde os ficheiros estão instalados:

```
sudo cd /etc/snort
```

Agora para criar o ficheiro de recuperação no terminal escreve-se o seguinte:

```
sudo cp snort.conf snort.conf.back
```

Este comando irá então copiar o ficheiro cujo o nome é de “snort.conf” e cria um novo com o nome de “snort.conf.back” (Figura 19).

A terminal window titled 'root@snort: /etc/snort' with search, menu, and window control icons. The terminal shows the following commands and output:

```
root@snort:/home/snort/Desktop# cd /etc/snort/  
root@snort:/etc/snort# ls  
classification.config  reference.config  snort.debian.conf  
community-sid-msg.map rules            threshold.conf  
gen-msg.map           snort.conf       unicode.map  
root@snort:/etc/snort# cp snort.conf snort.conf.back
```

Figura 19 - Criação do backup do snort.conf

Agora que existe um ficheiro de recuperação será possível aceder ao ficheiro “snort.conf” e adicionar uma linha que irá atribuir o IP da rede interna a uma variável chamada

“HOME_NET” (Figura 20), na qual essa variável ira ser utilizada para todas a regras criar daqui em diante.

```
#
# Additional information:
# This configuration file enables active response, to run snort in
# test mode -T you are required to supply an interface -i <interface>
# or test mode will fail to fully validate the configuration and
# exit with a FATAL error
#-----
#####
# This file contains a sample snort configuration.
# You should take the following steps to create your own custom configuration:
#
# 1) Set the network variables.
# 2) Configure the decoder
# 3) Configure the base detection engine
# 4) Configure dynamic loaded libraries
# 5) Configure preprocessors
# 6) Configure output plugins
# 7) Customize your rule set
# 8) Customize preprocessor and decoder rule set
# 9) Customize shared object rule set
#####
#####
# Step #1: Set the network variables. For more information, see README.variables
#####
# Setup the network addresses you are protecting
#
# Note to Debian users: this value is overridden when starting
# up the Snort daemon through the init.d script by the
# value of DEBIAN_SNORT_HOME_NET s defined in the
# /etc/snort/snort.debian.conf configuration file
#
ipvar HOME_NET any
ipvar HOME_NET 10.10.10.0/24
# Set up the external network addresses. Leave as "any" in most situations
ipvar EXTERNAL_NET any
# If HOME_NET is defined as something other than "any", alternative, you can
# use this definition if you do not want to detect attacks from your internal
# IP addresses:
#ipvar EXTERNAL_NET !$HOME_NET
# List of DNS servers on your network
ipvar DNS_SERVERS $HOME_NET
# List of SMTP servers on your network
ipvar SMTP_SERVERS $HOME_NET
# List of web servers on your network
```

Figura 20 - Adicionar a rede ao ficheiro snort.conf

Também é possível verificar se estão todas a regras a funcionar em condições, para tal no terminal usa-se o seguinte comando, caso estiver tudo em condições vai ser apresentado o resultado da Figura 21:

```
sudo snort -T -i enp0s8 -c /etc/snort/snort.conf
```

```
[ Number of packets truncated to 20 bytes: 1039 ]
pcap DAQ configured to passive.
Acquiring network traffic from "enp0s8".

---= Initialization Complete ===

o''_)- *~ Snort! <*-
'''   Version 2.9.7.0 GRE (Build 149)
      By Martin Roesch & The Snort Team: http://www.snort.org/contact#team
      Copyright (C) 2014 Cisco and/or its affiliates. All rights reserved.
      Copyright (C) 1998-2013 Sourcefire, Inc., et al.
      Using libpcap version 1.9.1 (with TPACKET_V3)
      Using PCRE version: 8.39 2016-06-14
      Using ZLIB version: 1.2.11

Rules Engine: SF_SNORT_DETECTION_ENGINE Version 2.4 <Build 1>
Preprocessor Object: SF_DNS Version 1.1 <Build 4>
Preprocessor Object: SF_REPUTATION Version 1.1 <Build 1>
Preprocessor Object: SF_MODBUS Version 1.1 <Build 1>
Preprocessor Object: SF_SSH Version 1.1 <Build 3>
Preprocessor Object: SF_DCERPC2 Version 1.0 <Build 3>
Preprocessor Object: SF_DNP3 Version 1.1 <Build 1>
Preprocessor Object: SF_GTP Version 1.1 <Build 1>
Preprocessor Object: SF_SIP Version 1.1 <Build 1>
Preprocessor Object: SF_FTPTELNET Version 1.2 <Build 13>
Preprocessor Object: SF_SMTP Version 1.1 <Build 9>
Preprocessor Object: SF_SSLPP Version 1.1 <Build 4>
Preprocessor Object: SF_SDF Version 1.1 <Build 1>
Preprocessor Object: SF_IMAP Version 1.0 <Build 1>
Preprocessor Object: SF_POP Version 1.0 <Build 1>

Snort successfully validated the configuration!
Snort exiting
root@snort:/etc/snort#
```

Figura 21 - Verificar se a configuração esta valida

Tendo em conta que será para implementar as regras do MISP, dentro do ficheiro “snort.conf” terá de ser adicionado o seguinte código apresentado na Figura 22 para o *Snort* poder utilizar as regras.

```
#MISP RULES
include $RULE_PATH/misp_attributes.rules
```

Figura 22 - Incluir as Regras no Snort

A finalizar caso seja pretendido criar as regras para o SNORT a filosofia é bastante simples utilizando as seguintes ações dependendo do que se pretende das regras:

Ação protocolo IP (do atacante) port <> (direção: "out", "in") IP (da nossa rede) port (msg:"Teste", sid: "id da regra do Snort")

As ações podem ser as seguintes:

- **alert** gerar um alerta utilizando o método de alerta selecionado, e depois regista o pacote
- **log** regista o pacote
- **pass** ignora o pacote
- **activate** alerta e depois aciona outra regra dinâmica
- **dynamic** permanecer inativo até ser ativado por uma regra de ativação, depois agir como uma regra de registo
- **drop** bloqueia e regista o pacote
- **reject** bloqueia o pacote, regista-o, e depois envia um TCP *reset* se o protocolo for TCP ou manda uma mensagem a dizer "*ICMP port unreachable*" se o protocolo for UDP.
- **sdrop** bloqueia o pacote, mas não o regista.

4.2 MISP

A instalação do MISP é bastante mais fácil, sendo que apenas é necessário correr um *script* que automatiza a instalação. Este *script* pode ser obtido acedendo ao website do MISP e fazer transferência.

```
snort@snort-VirtualBox:~/Desktop$ wget -O /tmp/INSTALL.sh https://raw.githubusercontent.com/MISP/MISP/2.4/INSTALL/INSTALL.sh
--2022-05-05 17:26:01-- https://raw.githubusercontent.com/MISP/MISP/2.4/INSTALL/INSTALL.sh
Resolving raw.githubusercontent.com (raw.githubusercontent.com)... 185.199.108.133, 185.199.111.133, 185.199.110.133, ...
Connecting to raw.githubusercontent.com (raw.githubusercontent.com)|185.199.108.133|:443... connected
HTTP request sent, awaiting response... 200 OK
Length: 159859 (156K) [text/plain]
Saving to: '/tmp/INSTALL.sh'

/tmp/INSTALL.sh      100%[=====] 156,11K  --.-KB/s    in 0,02s
2022-05-05 17:26:01 (7,49 MB/s) - '/tmp/INSTALL.sh' saved [159859/159859]
```

Figura 24 - Download do Script

```
snort@snort-VirtualBox:~/Desktop$ bash /tmp/INSTALL.sh -A
Next step: Checking if we are run as the installer template
Next step: Checking Linux distribution and Flavour...
Next step: We detected the following Linux Flavour: Ubuntu 20.04
Next step: Checking if we are uptodate and checksums match
sha1 matches
sha256 matches
sha384 matches
sha512 matches
-----
Next step: Setting MISP variables
Next step: Setting generic MISP variables shared by all flavours
groups: 'misp': no such user
The following DB Passwords were generated...
Admin (root) DB Password: 369d6de11092fb08797c8e749903e1a2df5b1568ccba96058de17db87ef7393a
User (misp) DB Password: 7b3484f42c6dd299de1f33318ab7d77c98c5da0b14f284093d28dfb5e69cafe0
Next step: Checking for parameters or Unattended Kall Install
Next step: Setting install options with given parameters.
all
Install on Ubuntu 20.04 LTS fully supported.
Please report bugs/issues here: https://github.com/MISP/MISP/issues
-----
Proceeding with the installation of MISP core
-----
Checking for sudo and installing etckeeper
[sudo] password for snort:
Hit:1 http://pt.archive.ubuntu.com/ubuntu focal InRelease
Get:2 http://pt.archive.ubuntu.com/ubuntu focal-updates InRelease [114 kB]
Get:3 http://security.ubuntu.com/ubuntu focal-security InRelease [114 kB]
Get:4 http://pt.archive.ubuntu.com/ubuntu focal-backports InRelease [108 kB]
Get:5 http://pt.archive.ubuntu.com/ubuntu focal-updates/main i386 Packages [637 kB]
Get:6 http://pt.archive.ubuntu.com/ubuntu focal-updates/main amd64 Packages [1750 kB]
Get:7 http://security.ubuntu.com/ubuntu focal-security/main amd64 Packages [1422 kB]
Get:8 http://pt.archive.ubuntu.com/ubuntu focal-updates/main amd64 DEP-11 Metadata [277 kB]
Get:9 http://pt.archive.ubuntu.com/ubuntu focal-updates/main amd64 c-n-f Metadata [15,0 kB]
Get:10 http://pt.archive.ubuntu.com/ubuntu focal-updates/universe amd64 Packages [921 kB]
Get:11 http://pt.archive.ubuntu.com/ubuntu focal-updates/universe i386 Packages [679 kB]
Get:12 http://pt.archive.ubuntu.com/ubuntu focal-updates/universe amd64 DEP-11 Metadata [390 kB]
Get:13 http://pt.archive.ubuntu.com/ubuntu focal-updates/universe amd64 c-n-f Metadata [20,7 kB]
Get:14 http://pt.archive.ubuntu.com/ubuntu focal-updates/multiverse amd64 DEP-11 Metadata [944 B]
Get:15 http://pt.archive.ubuntu.com/ubuntu focal-backports/main amd64 DEP-11 Metadata [9580 B]
Get:16 http://pt.archive.ubuntu.com/ubuntu focal-backports/universe amd64 DEP-11 Metadata [30,8 kB]
Get:17 http://security.ubuntu.com/ubuntu focal-security/main i386 Packages [428 kB]
Get:18 http://security.ubuntu.com/ubuntu focal-security/main amd64 DEP-11 Metadata [40,7 kB]
Get:19 http://security.ubuntu.com/ubuntu focal-security/main amd64 c-n-f Metadata [10,1 kB]
Get:20 http://security.ubuntu.com/ubuntu focal-security/universe amd64 Packages [700 kB]
```

Figura 23 - Inicio da Instalação do MISP

```

.....
MISP Installed, access here:

User: admin@admin.test
Password: admin
.....
The following files were created and need either protection or removal (shred on the CLI)
/home/misp/mysql.txt
Contents:
Admin (root) DB Password: 369d6de11092fb08797c8e749903e1a2df5b1568ccba96058de17db87ef7393a
User (misp) DB Password: 7b3484f42c6dd299de1f33318ab7d77c98c5da0b14f284093d28dfb5e69cafe0
/home/misp/MISP-authkey.txt
Contents:
Authkey: ZsIFqJGG10mtoGoUPHWbdQARtwqaZh8lPEnvliE8
.....
The LOCAL system credentials:
User: misp
Password: 6cddfaa7b991871f3803eed7cf26e4ddc1e798b69106723f07f92f3882beb45a # Or the password you used
of your custom user
.....
GnuPG Passphrase is: 9a86aae510543ff5e7d6a4c49b871be3f317b4bd9ffffe24dac6adb78c21efb7
.....
To enable outgoing mails via postfix set a permissive SMTP server for the domains you want to contact
:

sudo postconf -e 'relayhost = example.com'
sudo postfix reload
.....
Enjoy using MISP. For any issues see here: https://github.com/MISP/MISP/issues
.....
To run a command as administrator (user "root"), use "sudo <command>".
See "man sudo_root" for details.

misp@snort-VirtualBox:~$

```

Figura 25 - Fim da Instalação do MISP

No final do *script* é fornecido um *User* e uma *Password* esses dados são utilizados para entrar na *interface* de utilizador web. Após inserir os dados na página *Web* é apresentada outra página no qual é necessário alterar a *password default* do administrador.

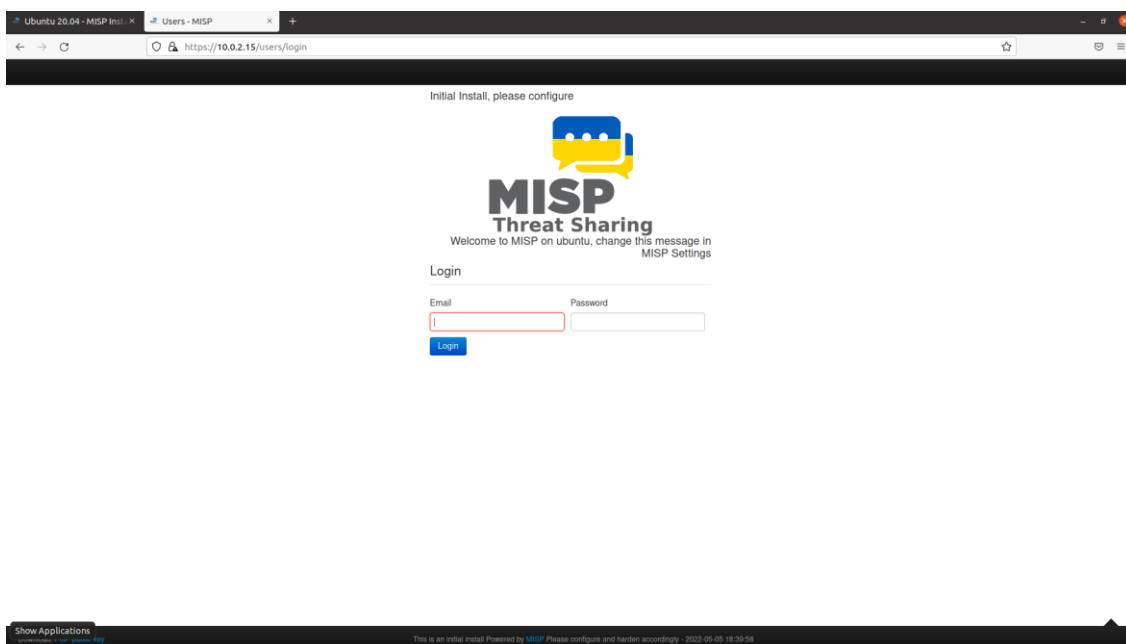


Figura 26 - Web UI Login

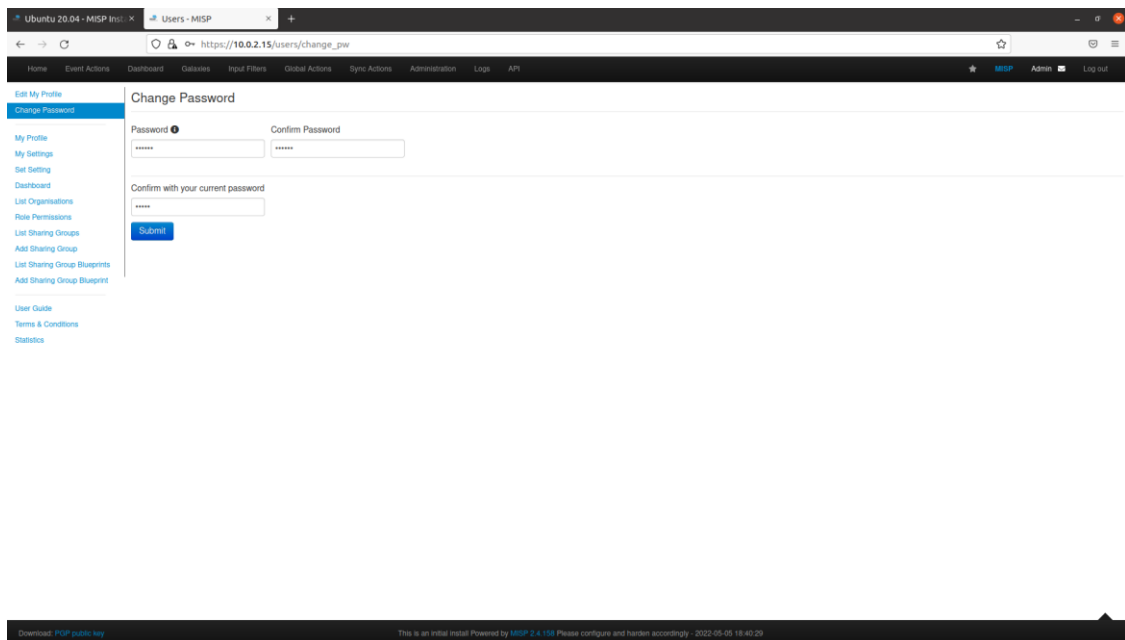


Figura 27 - Alteração da palavra-passe

Para concluir a instalação do MISP necessita-se de obter uma base de dados, sendo assim no *menu* na aba de *Administration* existe uma opção a dizer *Feed* na qual selecciona-se os dois *feeds* presentes e a opção de *Enable Selected*.

Essa opção irá fazer com que seja efetuado a transferência das duas bases de dados que vem por defeito com o MISP.

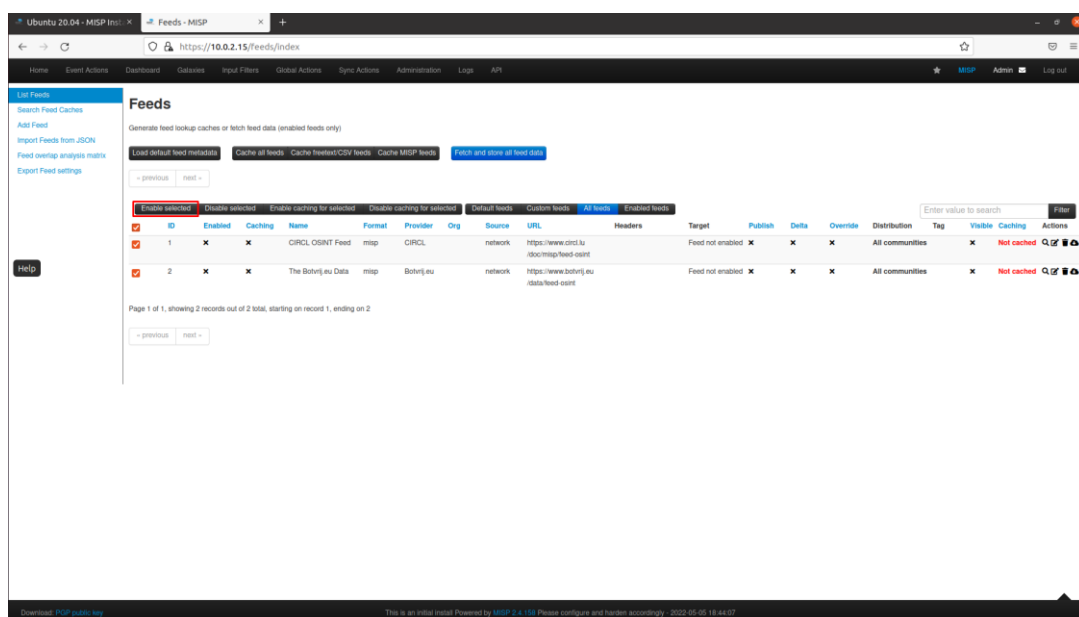


Figura 28 - MISP Feeds

Caso uma empresa encontre um IoC, essa mesma pode aceder ao MISP e criar a sua regra e partilhar com todos os utilizadores do MISP ou então caso não deseje partilhar com outros utilizadores essa regra poderá ficar disponível apenas para a sua empresa. Sendo que é possível criar uma regra que não seja pública para todos os utilizadores pode-se utilizar isso para criar um regra para testar a resiliência de segurança de rede, para tal acede-se a aba de Eventos é procedido a criação de um evento para esse IoC tal como apresentado nas seguintes Figuras 29 e 30.

Date: 2022-06-03

Distribution: Your organisation only

Threat Level: High

Analysis: Initial

Event Info: Ping Flood

Extends Event: Event UUID or ID. Leave blank if not applicable.

Submit

Figura 29 - Criar o nosso Evento

Ping Flood

Event ID	1507
UUID	3f1ed9c5-c484-483d-b9fa-89424225b891
Creator org	ORGNAME
Owner org	ORGNAME
Creator user	admin@admin.test
Protected Event (experimental)	Event is in unprotected mode. Switch to protected mode
Tags	+ +
Date	2022-06-23
Threat Level	High
Analysis	Initial
Distribution	Your organisation only
Warnings	Contextualisation: Your event has neither tags nor galaxy clusters attached - generally adding context to an event allows for quicker decision making and more accurate filtering, it is highly recommended that you label your events to the best of your ability.
Published	No
#Attributes	1 (0 Objects)
First recorded change	2022-06-23 15:36:12
Last change	2022-06-23 15:36:12
Modification map	
Sightings	0 (0) - restricted to own organisation only.

Navigation: Pivots | Galaxy | Event graph | Event timeline | Correlation graph | ATT&CK matrix | Event reports | Attributes | Discussion

Event: 1507: Ping Flood

Galaxies: [+](#) [+](#)

Figura 30 - Evento Criado

Apos criar o evento será necessário adicionar objetos. Estes objetos servem como recipientes que estão em torno de atributos contextualmente ligados, para tal seleciona-se a opção de adicionar objetos e preenche-se com os objetos que este evento irá ter, Figura 31.

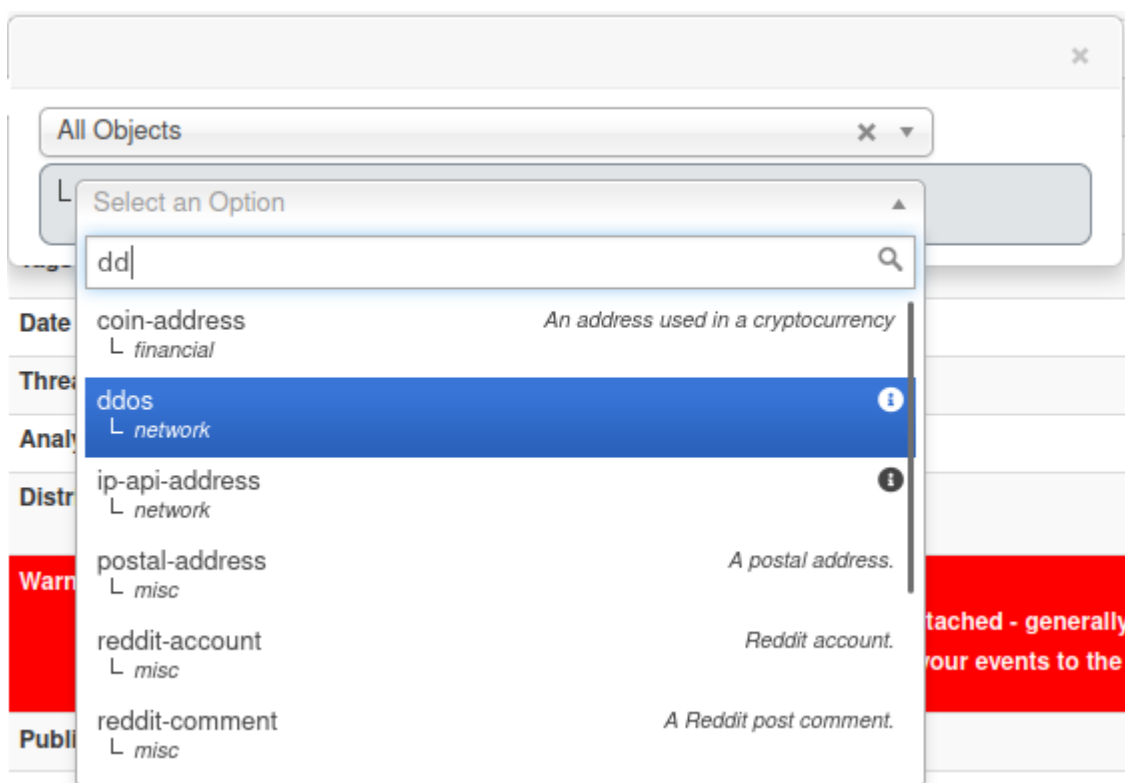


Figura 31 - Adicionar objetos ao Evento

Depois de ser escolhido o tipo de objeto será apresentado um *menu* contendo diversas opções como apresentados nas Figuras 32 e 33, na qual apenas é pretendido que seja preenchido o campo *ip-source* portanto qual é o endereço de onde este ataque origina e qual é o tipo de ataque neste caso ICMP.

Add Ddos Object

Object Template Ddos v9

Description DDoS object describes a current DDoS activity from a specific or/and to a specific target. Type of DDoS can be attached to the object as a taxonomy or using the type field.

Requirements Required one of: ip-dst, ip-src, domain-dst

Meta category Network

Distribution Your organisation only

Comment

First seen date 2022-06-23 **Last seen date** 2022-06-23

First seen time HH:MM:SS.ssssss+TT:TT **Last seen time** HH:MM:SS.ssssss+TT:TT

Expected format: HH:MM:SS.ssssss+TT:TT

Save	Name	Type	Description	Category	Value	IDS	Disable	Correlation	Distribution	Comment
<input type="checkbox"/>	Domain-dst	domain	Destination domain (victim)	Network activity		<input checked="" type="checkbox"/>	<input type="checkbox"/>		Inherit event	
<input type="checkbox"/>	Ip-dst	ip-dst	Destination IP (victim)	Network activity		<input checked="" type="checkbox"/>	<input type="checkbox"/>		Inherit event	
<input checked="" type="checkbox"/>	Ip-src	ip-src	IP address originating the attack	Network activity	10.10.10.3	<input checked="" type="checkbox"/>	<input type="checkbox"/>		Your organisation only	

Figura 32 - Criação do Objeto

<input checked="" type="checkbox"/>	Protocol text	Protocol used for the attack	Other	ICMP	<input type="checkbox"/>	<input checked="" type="checkbox"/>		Your organisation only	
<input type="checkbox"/>	Src-port port	Port originating the attack	Network activity		<input type="checkbox"/>	<input type="checkbox"/>		Inherit event	
<input type="checkbox"/>	Text text	Description of the DDoS	Other		<input type="checkbox"/>	<input checked="" type="checkbox"/>		Inherit event	
<input type="checkbox"/>	Total-bps counter	Bits per second (maximum rate of bits per second measured)	Other		<input type="checkbox"/>	<input checked="" type="checkbox"/>		Inherit event	
<input type="checkbox"/>	Total-bytes-sent counter	Total number of bytes sent by the sources mentioned	Other		<input type="checkbox"/>	<input checked="" type="checkbox"/>		Inherit event	
<input type="checkbox"/>	Total-packets-sent counter	Total number of packets sent by the source mentioned	Other		<input type="checkbox"/>	<input checked="" type="checkbox"/>		Inherit event	
<input type="checkbox"/>	Total-pps counter	Packets per second (maximum rate of packets per second)	Other		<input type="checkbox"/>	<input checked="" type="checkbox"/>		Inherit event	

Figura 33 - Continuação da criação do Objeto

Tendo em conta que todos os eventos das duas bases de dados que estão no MISP já estão disponíveis, apenas resta efetuar a transferência dos eventos do MISP para a máquina de testes.

```
1 # MISP export of IDS rules - optimized for
2 #
3 # These NIDS rules contain some variables that need to exist in your configuration.
4 # Make sure you have set:
5 #
6 # $HOME_NET - Your internal network range
7 # $EXTERNAL_NET - The network considered as outside
8 # $SMTP_SERVERS - All your internal SMTP servers
9 # $HTTP_PORTS - The ports used to contain HTTP traffic (not required with suricata export)
10 #
11 reject ip 10.10.10.3 any -> $HOME_NET any (msg: "MISP e1507 [ ] Incoming From IP: 10.10.10.3"; classtype:trojan-activity; sid:6008901; rev:1; priority:1; reference:url,https://localhost/events/view/1507;)
```

Figura 34 - Export final do Evento

4.3 Plataforma de Testes

A Plataforma de Testes de Resiliência, consiste num conjunto de scripts em Bash controlados por script que define um menu dinâmico. Este script, sempre que se adiciona algo novo na pasta onde está plataforma instalada, faz com que próprio menu se atualize passando a disponibilizar de automaticamente mais opções. O utilizador pode através deste script escolher os testes a realizar, adicionando-os a uma lista que se designa por bateria de testes. É também possível ao utilizador, através deste menu dinâmico e principal,

- Criar *scripts* de testes
- Selecionar *scripts* que irá usar, adicionando-os a uma pilha de testes (bateria)
- Correr a bateria de testes
- Remover scripts da bateria de testes

Na figura 35 é apresentado o código para que os utilizadores possam criar os seus *scripts* utilizando esta bateria de testes, apos inserir o nome dos *scripts* que o utilizador pretende esta irá ser criado e dentro desse ficheiro irão ser acrescentados linhas para que o utilizador possa descrever sobre o que o seu script se trata, como funciona, quando foi criado, quem é o dono desse script e uma pequena descrição do mesmo. O preenchimento dessa informação é obrigatório.

```
#!/bin/bash

criar_script(){
    echo -e ""
    read -p "Nome do Script: " nScript
    touch $nScript
    echo "#!/bin/bash" >> $nScript
    echo "#Owner: " >> $nScript
    echo "#Date: " >> $nScript
    echo "#SortDescription: " >> $nScript
    echo "#Description: " >> $nScript
    echo "#How to Use: " >> $nScript
    nano $nScript
}
```

Figura 35 - Criação de Scripts

A função ilustrada na figura 36 demonstra o código para a criação do menu que lista os *scripts* existentes. O menu é totalmente dinâmico. Através de um ciclo *for* o script lista os vários diretórios que existem no diretório base. Estes diretórios permitem agrupar os scripts de teste por sistema operativo e por tipo de teste (e.g. rede, sistema operativo, aplicação).

```

listar_menu(){
    N=1
    args=()

    echo
    for i in `ls -1 | grep -v script`
    do
        echo $N - $i
        args+=("$i")
        let N=$N+1
    done
}

```

Figura 36 - Listar Opções

Tal como na figura anterior, a função apresentada na Figura 37 é a continuação do menu dinâmico. Esta função permite listar todos os ficheiros dentro de uma pasta que foi acedido anteriormente apresentado a pequena descrição do ficheiro e o nome desse mesmo ficheiro. Após isso é perguntado qual o *script* que se prende que seja adicionado a bateria de testes. Por fim, a lista de testes que compõe a bateria é guardada contendo caminho para a cada script selecionado para ser executado contra o ambiente alvo de testes de resiliência.

```

opcoes_scripts() {

    argsscripts=()

    listar_menu

    read -p "Insira opcao: " op

    N=1
    for k in `ls -1 ${args[$op-1]}`
    do
        Name=`cat ${args[$op-1]}/${k} | grep SortDescription | awk -F ":" '{print $2}'`
        echo $N - $Name \($k\)
        argsscripts+=("${args[$op-1]}/${k}")
        let N=$N+1
    done

    read -p "Qual script deseja adicionar a bateria: " opS
    for y in `ls -1 ${argsscripts[$opS-1]}/${y}`
    do
        #short=${y##*/}
        echo "./${y} >> bateria.sh"
    done

    show_menu
}

```

Figura 37 - Adicionar scripts a pilha de testes

Para terminar este *script* também é necessário existir uma função para correr e eliminar a bateria de testes como apresentado nas figuras 38 e 39.

```

✓ correr_bateria(){
    echo -e ""
    touch bateria.sh
    ./bateria.sh
}

```

Figura 38 - Correr a pilha de Testes

```

remover_bateria(){
    rm bateria.sh
}

```

Figura 39 - Remover a pilha de Testes

Existe ainda um menu para facilitar a escolha das funcionalidades e executar o script, figura 40.

```

show_menu() {
    echo -e "\n  Select an option from menu: "
    echo -e "\n Key  Menu Option:          Description:"
    echo -e " ---  -----"
    echo -e "  1 - Criar                (Cria um script para o utilizador)"
    echo -e "  2 - Adicionar            (Adiciona Scripts a nossa bateria)"
    echo -e "  3 - Correr               (Corre a nossa bateria)"
    echo -e "  4 - Remover              (Remove tudo dentro da bateria)"
    read -n1 -p " Press key for menu item selection or press X to exit: " menuinput

    case $menuinput in
        1) criar_script;;
        2) opcoes_scripts;;
        3) correr_bateria;;
        4) remover_bateria;;
        x|X) echo -e "\n\n Exiting";;
        *) show_menu;;
    esac
}

exit_screen () {
    echo -e "\n All Done! \n"
    exit
}

#Run menus
show_menu
exit_screen

```

Figura 40 - Menu principal do Script

O *menu* dinâmico é uma peça chave na Plataforma de Testes. Associado ao menu existem os scripts de teste. Para efeitos de prova de conceito foram criados os seguintes scripts:

- Ping (Teste ICMP)
- NMAP
- SNMP (Enumeração)
- DNS *Lookup* (Enumeração)
- FTP (*Login*)
- SSH (*Login*)

O *script* do *Ping*, Figura 41, é bastante simples. Este apenas serve para verificar se a máquina que pretendemos testar está acessível ou no caso deste projeto para verificar se as regras implementadas no MISP estão a funcionar corretamente. De forma mais específica, se existe uma regra que foi carregada a partir do MISP que indica que um determinado IP é malicioso, espera-se que a execução do script seja bloqueada evidenciando a resiliência da plataforma IT.

```
#!/bin/bash
#Owner: Christopher
#Date: 10/04/2022
#SortDescription: Ping
#Description: Pings target to see if target is online or to test icmp requests
#How to Use: Insert the target IP

read -p "Targets IP or Domain: " tIp

ping $tIp
```

Figura 41 - Script de Ping

Nas figuras 42, 43 e 44 são demonstrados três scripts que servem para efetuar enumerações a uma rede ou domínio. O *script* apresentado na Figura 42 demonstra o processo de uma enumeração sobre DNS, ou seja, irá verificar todos os domínios que estão registados nesse endereço. Para tal utiliza-se a ferramenta cuja o nome é *dig*, após essa ferramenta terminar, pode-se ou não fazer um mapeamento de todos os dispositivos ligados a essa rede usando o NMAP.

```
#!/bin/bash
#Owner: Christopher
#Date: 11/04/2022
#SortDescription: DNS Lookup
#Description: Lists DNS from a domain or target and if we want we can scan there network with NMAP
#How to Use: Insert the target IP or Domain

read -p "Targets IP or Domain: " tIp
answer=False
dig=`dig $tIp +short`
if [ -z "$dig" ];
then
echo -e "\n "
echo "No IPs found"
sleep 3
else
echo -e "IPs found: \n $dig"
while [ $answer != True ];
do
echo -e '\n Do you want to do NMAP after enumeration? '
read ansDns
if [ "$ansDns" == "y" ] || [ "$ansDns" == "yes" ];
then
answer=True
for ips in $dig;
do
nmap $ips
done
else
if [ "$ansDns" == "n" ] || [ "$ansDns" == "no" ];
then
answer=True
dig $tIp $tDom +short
fi
fi
done
fi
```

Figura 42 - Script DNS Lookup

Como mencionado anteriormente o NMAP, Figura 43, é utilizado para listar todos os dispositivos ligado a uma rede ou então verificar quais são as portas que o dispositivo tem aberto que possam acabar por se tornar numa vulnerabilidade.

```
#!/bin/bash
#Owner: Christopher
#Date: 12/04/2022
#SortDescription: NMAP
#Description: Lists all services from devices connected to a network or a single device
#How to Use: Insert the target IP or Network, #sS = Stealth Scan, T = Time (Slower)1-5(Faster),
#p = Ports, A = Agressive Scan(OS scan, Default Nmap Script, Service Version Scan), oN = Export

read -p "Targets IP or Network: " tIp
read -p 'Do you want to export to a file? ' ansN
if [ "$ansN" == "y" ] || [ "$ansN" == "yes" ];
then
read -p 'File Name: ' nFile
nmap -sS -A -T4 -p- $tIp -oN $nFile.txt
else
nmap -sS -A -T4 -p- $tIp
fi
```

Figura 43 - Script NMAP

Para terminar a apresentação dos *scripts* relativos à prova de conceito sobre enumeração apenas falta o script da Figura 44. Este script permite listar contas, palavras-passe, grupos, dispositivos e nomes de sistema numa máquina. Uma enumeração em SNMP apenas funcionará caso o dispositivo que seja o alvo tenha esse tipo de serviço ativo para verificar tal coisa é necessário utilizar a ferramenta do NMAP.

```
#!/bin/bash
#Owner: Christopher
#Date: 20/04/2022
#SortDescription: SNMP Enumeration
#Description: Enumerates the SNMP from a target, only until version 2c supported
#How to Use: Insert the target IP

read -p "Targets IP: " tIp
answer=False
while [ answer != True ];
do
    read -p 'What SNMP Version you want to use (Version: 1,2c)? ' ansV
    if [ "$ansV" == "1" ];
    then
        answer=True
        snmpwalk -v1 -c public $tIp
    else
        if [ "$ansV" == "2c" ]
        then
            answer=True
            snmpwalk -v2c -c public $tIp
        fi
    fi
done
```

Figura 44 - Script SNMP

Também é possível efetuar *logins* utilizando os scripts SSH e FTP. Estes scripts permitem testar os logins pois uma empresa pode ter serviços SSH tal como o serviço FTP ativos, sendo assim esses serviços podem ser acedidos por outras pessoas caso não haja o controlo suficiente por parte da plataforma de segurança.

Na Figura 45 é apresentado o script que efetua o login num serviço FTP.

```
#!/bin/bash
#Owner: Christopher
#Date: 10/04/2022
#SortDescription: FTP Login
#Description: Trying to login to a FTP server
#How to Use: Insert the target IP

read -p "Targets IP or Domain: " tIp
ftp $tIp
```

Figura 45 - Script Login FTP

Para terminar alguns exemplos de *scripts* para a pilha de testes da plataforma será apresentando na Figura 46, o *script* para efetuar logins em serviços SSH.

```
#!/bin/bash
#Owner: Christopher
#Date: 10/04/2022
#SortDescription: SSH Login
#Description: Trying to login to a SSH server
#How to Use: Insert the target IP

read -p "Targets IP: " tIp
answer=False
while [ answer != True ];
do
    read -p 'Username: ' user
    read -p 'Port: ' port
    if [ -z $user ] || [ -z $port ];
    then
        echo -e "You need to set a Username and Port"
    else
        answer=True
        if [ -z $tIp ];
        then
            ssh $user@$tIp -p $port
        fi
    fi
done
```

Figura 46 - Login em serviço SSH

4.3 Testes de resiliência

Tendo em conta que a plataforma de testes já criada é necessário testar para verificar se a solução implementada tem o funcionamento pretendido.

Para tal, procedeu-se aos testes da plataforma contra o ambiente IT criado. Na Figura 47 observa-se a plataforma de teste no seu correto funcionamento.

```
$ ./script.sh

Select an option from menu:

Key  Menu Option:      Description:
---  -
1 - Criar            (Cria um script para o utilizador)
2 - Adicionar        (Adiciona Scripts a nossa bateria)
3 - Correr           (Corre a nossa bateria)
4 - Remover          (Remove tudo dentro da bateria)
Press key for menu item selection or press X to exit: █
```

Figura 47 - Script em funcionamento

A bateria de testas foi criada utilizando a plataforma, como se ilustra na Figura 48. Para demonstração, foram seleccionados vários scripts, sendo estes scripts que compõe a bateria de testes apresentados na figura 49.

```
$ ./script.sh

Select an option from menu:

Key  Menu Option:      Description:
---  -
1 - Criar            (Cria um script para o utilizador)
2 - Adicionar        (Adiciona Scripts a nossa bateria)
3 - Correr           (Corre a nossa bateria)
4 - Remover          (Remove tudo dentro da bateria)
Press key for menu item selection or press X to exit: 2
1 - bateria.sh
2 - Enumeration
3 - Login
4 - teste.sh
Insira opcao: 2
1 - DNS Lookup (dns_lookup.sh)
2 - NMAP (nmap.sh)
3 - Ping (ping.sh)
4 - SNMP Enumeration (snmp_enum.sh)
Qual script deseja adicionar a bateria: █
```

Figura 48 - Alimentar a pilha de testes

```
1 ./Enumeration/ping.sh
2 ./Enumeration/nmap.sh
3 ./Login/ssh.sh
4 |
```

Figura 49 - Scripts dentro da pilha de testes

Finalmente, a bateria de testes é executada. A opção três no menu permite correr todos os scripts que constam da bateria de testes. Como é possível verificar na figura 50 e 51 o programa tem o correto funcionamento. Para além disso consegue-se verificar que a regra implementada do MISP, que rejeitava pedidos ICMP, também esta a funcionar devidamente, mas pode-se verificar que é possível fazer enumerações usando o NMAP a essa máquina evidenciando que existe uma porta SSH aberta. Sendo que não foi implementada nenhuma regra contra a enumerações das portas da máquina ou contra acesso não autorizado por SSH consegue-se ver que é possível aceder através de SSH à máquina, confirmando que a regra do MISP está a funcionar com é devido e a plataforma de testes também.

```
Select an option from menu:
Key Menu Option: Description:
1 - Criar (Cria um script para o utilizador)
2 - Adicionar (Adiciona Scripts a nossa bateria)
3 - Correr (Corre a nossa bateria)
4 - Remover (Remove tudo dentro da bateria)
Press key for menu item selection or press X to exit: 3
Targets IP or Domain: 10.10.10.2
PING 10.10.10.2 (10.10.10.2) 56(84) bytes of data.
From 10.10.10.2 icmp_seq=2 Destination Port Unreachable
From 10.10.10.2 icmp_seq=1 Destination Port Unreachable
From 10.10.10.2 icmp_seq=2 Destination Port Unreachable
From 10.10.10.2 icmp_seq=2 Destination Port Unreachable
From 10.10.10.2 icmp_seq=2 Destination Port Unreachable
--- 10.10.10.2 ping statistics ---
4 packets transmitted, 0 received, + 4 errors, 0% packet loss, time 1001ms
rtt min/avg/max/mdev = 0.180/0.188/0.196/0.008 ms
Targets IP or Network: 10.10.10.2
Do you want to export to a file? n
Starting Nmap 7.92 ( https://nmap.org ) at 2022-07-12 12:27 EDT
Nmap scan report for 10.10.10.2
Host is up (0.00021s latency).
Not shown: 65532 closed tcp ports (reset)
PORT STATE SERVICE VERSION
22/tcp open ssh OpenSSH 8.2p1 Ubuntu 4ubuntu0.5 (Ubuntu Linux; protocol 2.0)
|_ ssh-hostkey:
|_ 3072 a5:96:bb:6e:8b:e0:16:69:c5:7a:f4:6b:73:2e:48:5e (RSA)
|_ 256 70:01:69:f0:fe:45:dc:18:2b:69:45:d5:73:83:aa:71 (ECDSA)
|_ 256 96:e4:bf:e8:57:28:ba:74:f8:0e:8c:75:26:9a:d8:3f (ED25519)
80/tcp open http Apache httpd 2.4.41 ((Ubuntu))
|_ http-title: Did not follow redirect to https://10.10.10.2/
|_ http-server-header: Apache/2.4.41 (Ubuntu)
443/tcp open ssl/http Apache httpd 2.4.41 ((Ubuntu))
|_ http-robots.txt: 1 disallowed entry
|_ /
|_ http-title: Users - MISP
|_ Requested resource was https://10.10.10.2/users/login
|_ http-trace-info: Problem with XML parsing of /evox/about
|_ ssl-cert: Subject: commonName=misp.local/organizationName=Organization/stateOrProvinceName=State/countryName=LU
|_ Not valid before: 2022-06-13T15:32:19
|_ Not valid after: 2023-06-13T15:32:19
|_ http-server-header: Apache/2.4.41 (Ubuntu)
|_ ssl-date: TLS randomness does not represent time
MAC Address: 08:00:27:46:74:25 (Oracle VirtualBox virtual NIC)
Device type: general purpose
Running: Linux 4.X|5.X
OS CPE: cpe:/o:linux:linux_kernel:4 cpe:/o:linux:linux_kernel:5
OS details: Linux 4.15 - 5.6
Network Distance: 1 hop
Service Info: OS: Linux; CPE: cpe:/o:linux:linux_kernel
```

Figura 50 - Resultado dos Testes I

```
TRACEROUTE
HOP RTT ADDRESS
1 0.21 ms 10.10.10.2

OS and Service detection performed. Please report any incorrect results at https://nmap.org/submit/ .
Nmap done: 1 IP address (1 host up) scanned in 17.59 seconds
Targets IP: 10.10.10.2
Username: snort
Port: 22
The authenticity of host '10.10.10.2 (10.10.10.2)' can't be established.
ED25519 key fingerprint is SHA256:7/jeaL4FqYqCHYSbBmij/t/OkL+5MhvaR/TgZNqJhMk.
This key is not known by any other names
Are you sure you want to continue connecting (yes/no/[fingerprint])? yes
Warning: Permanently added '10.10.10.2' (ED25519) to the list of known hosts.
snort@10.10.10.2's password:
Welcome to Ubuntu 20.04.4 LTS (GNU/Linux 5.13.0-48-generic x86_64)

 * Documentation:  https://help.ubuntu.com
 * Management:    https://landscape.canonical.com
 * Support:        https://ubuntu.com/advantage

0 updates can be applied immediately.

The list of available updates is more than a week old.
To check for new updates run: sudo apt update
Your Hardware Enablement Stack (HWE) is supported until April 2025.
Last login: Tue Jul 12 17:17:42 2022 from 10.10.10.3
snort@snort-VirtualBox:~$
```

Figura 51 - Resultado dos Testes II

5. Conclusão

Os ataques informáticos não param, nem a exploração de novos métodos de descoberta de vulnerabilidades, nem todas as empresas têm as suas *Firewalls*, DMZ ou mesmo os suas IDS com as mais recentes atualizações contra essas vulnerabilidades, tornando-se vulneráveis a ataques informáticos.

Este tópico é importante não só para aqueles que estão conscientes da área da Cibersegurança, mas para todos aqueles interessados em criar ou proteger a infraestrutura de informática da empresa.

A tecnologia faz parte das nossas vidas e todos nós podemos ser alvos de ataques cibernéticos, por isso é importante ter uma proteção ou algum conhecimento sobre como proteger contra estes ataques.

Para este projeto foi desenvolvido uma plataforma para testar a resiliência contra-ataques cibernéticos numa empresa. Como tal foi criado um ambiente virtual com o MISP e o SNORT para simular o ambiente de uma empresa. Além disso foi desenvolvido uma plataforma de testes para testar a resiliência que permite criar uma bateria de testes a executar contra o sistema alvo de análise. Esta bateria de testes serve para verificar se o ambiente está seguro contra-ataques cibernéticos conhecidos.

Uma plataforma para testar a resiliência de ataques cibernéticos é muito importante numa empresa, pois sem existir uma plataforma destas a empresa irá ficar vulnerável a ataques cibernéticos podendo levar a fuga de informação importante de clientes, funcionários ou mesmo de fornecedores fazendo com que a sua reputação seja má ou levar a mesma a falência.

Referências Bibliográficas

- [1] "Cybercrime To Cost The World \$10.5 Trillion Annually By 2025," *Cybercrime Magazine*, Dec. 08, 2018. <https://cybersecurityventures.com/cybercrime-damages-6-trillion-by-2021/> (accessed Jun. 19, 2022).
- [2] "300+ Terrifying Cybercrime & Cybersecurity Statistics (2022)," *Comparitech*. <https://www.comparitech.com/vpn/cybersecurity-cyber-crime-statistics-facts-trends/> (accessed Jun. 19, 2022).
- [3]
"CIRCL » MISP - Open Source Threat Intelligence Platform."
<https://www.circl.lu/services/misp-malware-information-sharing-platform/> (accessed May 21, 2022).
- [4]
"Malware Statistics & Trends Report | AV-TEST." <https://www.av-test.org/en/statistics/malware/> (accessed Jun. 19, 2022).
- [5]
License. TheHive Project, 2022. Accessed: May 20, 2022. [Online]. Available:
<https://github.com/TheHive-Project/CortexDocs>
- [6]
MISP, "MISP Open Source Threat Intelligence Platform & Open Standards For Threat Information Sharing," *MISP Open Source Threat Intelligence Platform & Open Standards For Threat Information Sharing*. <https://www.misp-project.org/> (accessed May 10, 2022).
- [7]
"Segurança de computadores," *Wikipédia, a enciclopédia livre*. Feb. 26, 2022. Accessed: Jun. 18, 2022. [Online]. Available:
https://pt.wikipedia.org/w/index.php?title=Seguran%C3%A7a_de_computadores&oldid=63094552
- [8]
"Snort - Network Intrusion Detection & Prevention System." <https://www.snort.org/> (accessed May 10, 2022).
- [9]
Try it. TheHive Project, 2022. Accessed: May 21, 2022. [Online]. Available:
<https://github.com/TheHive-Project/TheHive>
- [10]
C. Labs, "What is (MISP) Malware Information Sharing Platform | Cyware Educational Guides | Educational Guides," *Cyware Labs*. <https://cyware.com/educational-guides/cyber->

[threat-intelligence/what-is-malware-information-sharing-platform-misp-b28e](https://www.ibm.com/topics/cybersecurity) (accessed May 21, 2022).

[11]

“What is Cybersecurity? | IBM.” <https://www.ibm.com/topics/cybersecurity> (accessed Jun. 18, 2022).

[12]

“What Are STIX/TAXII?” <https://www.anomali.com/resources/what-are-stix-taxii> (accessed Jul. 04, 2022).

[13]

“What is a Firewall? The Different Types of Firewalls,” *Check Point Software*. <https://www.checkpoint.com/cyber-hub/network-security/what-is-firewall/> (accessed Jul. 04, 2022).

[14]

“What is an Intrusion Detection System? | Barracuda Networks.” <https://www.barracuda.com/glossary/intrusion-detection-system> (accessed Jul. 05, 2022).

[15]

“What is Intrusion Prevention System? | VMware Glossary,” *VMware*. <https://www.vmware.com/topics/glossary/content/intrusion-prevention-system.html> (accessed Jul. 05, 2022).

[16]

C. Labs, “What is Open Indicators of Compromise (OpenIOC) Framework | Educational Guides | Educational Guides,” *Cyware Labs*. <https://cyware.com/educational-guides/cyber-threat-intelligence/what-is-open-indicators-of-compromise-openioc-framework-ed9d> (accessed Jul. 04, 2022).

[17]

“Introduction to STIX.” <https://oasis-open.github.io/cti-documentation/stix/intro.html> (accessed Jul. 04, 2022).

[18]

“Intrusion Prevention System (IPS),” *GeeksforGeeks*, Apr. 09, 2019. <https://www.geeksforgeeks.org/intrusion-prevention-system-ips/> (accessed Jul. 05, 2022).

[19]

alexandra, “VMRay Feature Brief - Automated IOC Generation,” *VMRay*. <https://www.vmrays.com/resource/vmrays-feature-brief-automated-ioc-generation/> (accessed Jul. 04, 2022).

[20]

“Security information and event management,” *Wikipedia*. Jul. 02, 2022. Accessed: Jul. 05, 2022. [Online]. Available:

https://en.wikipedia.org/w/index.php?title=Security_information_and_event_management&oldid=1096153732

[21]

MISP Project, 2022. Accessed: May 10, 2022. [Online]. Available:

<https://github.com/MISP/MISP>

Anexos

Anexo1 – Script para Testes - <https://github.com/chr1sM/MISP/tree/main/Scripts>

Anexo 2 – Script para download das Regras -

https://github.com/chr1sM/MISP/blob/main/misp_auto.sh