CEFET/RJ
DIRETORIA DE PESQUISA E PÓS-GRADUAÇÃO

UMA ABORDAGEM BASEADA EM WORKFLOW PARA PRODUÇÃO DE DATASETS SINTÉTICOS DE FLUXOS DE ACESSOS COM MARCAÇÕES PARA IDENTIFICAÇÃO DE ATAQUES DE NEGAÇÃO DE SERVIÇO

Valeria Farias Alves

Dissertação de Mestrado apresentada ao Programa de Pós-graduação em Tecnologia, Centro Federal de Educação Tecnológica Celso Suckow da Fonseca, CEFET/RJ, como parte dos requisitos necessários à obtenção do título de mestre.

Orientadores:

Eduardo Soares Ogasawara

Rafael Santos Carlos Machado

Rio de Janeiro

Maio de 2015

UMA ABORDAGEM BASEADA EM WORKFLOW PARA PRODUÇÃO DE DATASETS SINTÉTICOS DE FLUXOS DE ACESSOS COM MARCAÇÕES PARA IDENTIFICAÇÃO DE ATAQUES DE NEGAÇÃO DE SERVIÇO

Dissertação de Mestrado apresentada ao Programa de Pós-graduação em Tecnologia, Centro Federal de Educação Tecnológica Celso Suckow da Fonseca, CEFET/RJ, como parte dos requisitos necessários à obtenção do título de mestre.

Valeria Farias Alves

Aprovada por:

_____

Presidente Prof. Eduardo Soares Ogasawara, D.Sc.

_____

Prof. Raphael Santos Carlos Machado, D.Sc. (INMETRO)

_____

Prof. Cristina Gomes de Souza, D.Sc.

_____

Prof. Eduardo Bezerra da Silva, DSc

_____

Prof. Davidson Rodrigo Boccardo, D.Sc. (INMETRO)

Rio de Janeiro

Maio de 2015

Ficha catalográfica elaborada pela Biblioteca Central do CEFET/RJ

A474 Alves, Valeria Farias
Uma abordagem baseada em workflow para produção de datasets sintéticos de fluxos de acessos com marcações para identificação de ataques de negação de serviço / Valeria Farias Alves.—2015.
xii, 49f. : il.color. , grafs. , tabs. ; enc.

Dissertação (Mestrado) Centro Federal de Educação Tecnológica Celso Suckow da Fonseca , 2015.
Bibliografia : f.46-49
Orientadores : Eduardo Soares Ogasawara
Rafael Santos Carlos Machado

1. Redes de computadores – Medidas de segurança. 2. Proteção de dados. 3. Fluxo de trabalho. 4. Banco de dados. I. Ogasawara, Eduardo Soares (Orient.). II. Machado, Rafael Santos Carlos (Orient.). III. Título.

CDD 005.8

Dedico este trabalho a minha filha Carol Alves Pinto, por ser a minha eterna razão de viver.

À Deus, que permitiu minha existência e oportunidades para minha evolução.

À minha família, por ser o meu pilar mestre.

À minha filha, pelos meus momentos de ausência.

Ao Prof. Dr. Eduardo Soares Ogasawara e ao Prof. Dr. Raphael Santos Carlos Machado pela orientação, amizade, respeito, confiança e oportunidade a mim oferecida. Obrigado por todo conhecimento que me proporcionaram.

Ao meu superior imediato, Flavio Silva, que propiciou as condições necessárias para a realização deste curso.

Aos meus parceiros de estudos Riccardo Campisano, Carlos Magno, Erika de Freitas, Luiz Magalhães e Viktor Hugo.

Agradeço.

RESUMO

UMA ABORDAGEM BASEADA EM WORKFLOW PARA PRODUÇÃO DE DATASETS SINTÉTICOS DE FLUXOS DE ACESSOS COM MARCAÇÕES PARA IDENTIFICAÇÃO DE ATAQUES DE NEGAÇÃO DE SERVIÇO

Valeria Farias Alves

Orientadores:

Eduardo Soares Ogasawara

Raphael Santos Carlos Machado

Resumo da Dissertação de Mestrado submetida ao Programa de Pós-graduação em Tecnologia do Centro Federal de Educação Tecnológica Celso Suckow da Fonseca, CEFET/RJ, como parte dos requisitos necessários à obtenção do título de mestre.

A evolução tecnológica está proporcionando novas formas de comunicação e troca de informações entre pessoas e corporações por meio da conexão de milhares de dispositivos. Estas comunicações acontecem em tempo real e possibilitam a troca de dados e de informações diversas, sejam elas de cunho profissional ou particular, acadêmico ou de diversão. Juntamente com a evolução dessa conectividade surgem as formas de ataques cibernéticos, do inglês *cyber attacks*. Os ataques podem ter cunho social, político ou financeiro e se caracterizam por acessos não autorizados, roubo de informações e paralisação de serviços. Na busca pelo resguardo das informações, são implementadas medidas de proteção definidas como segurança da informação, que buscam mitigar os riscos envolvidos nas várias formas de comunicação, objetivando a proteção das mais diversas informações contidas nos meios eletrônicos. O primeiro passo no tratamento de um ataque cibernético é a sua identificação, que pode ser realizada por meio de equipamentos específicos, definidos com IDS (*Intrusion Detection System*), onde é possível analisar o tráfego de rede com o objetivo de identificar anomalias. Este trabalho apresenta o *Access Collector* um *workflow* para a construção de *dataset* (banco de dados), de forma automática e sistematizada, com base em fluxo de acesso com marcação, para a análise de ataques de segurança da informação, com foco em ataques de negação de serviço. Tal estudo visa colaborar na reprodução de *datasets* com marcação, possibilitando o treinamento de sistemas de detecção, com a característica de possuir os fluxos rotulados.

Palavras-Chave:

Fluxo de Dados; Ataques Cibernéticos; *Workflow*s

Rio de Janeiro

Maio de 2015

ABSTRACT

A WORKFLOW BASED APPROACH TO PRODUCTION OF SYNTHETIC DATASETS FLOWS ACCESSES WITH TAGS TO IDENTIFICATION DENIAL OF SERVICE ATTACK

Valeria Farias Alves

Advisors:

Eduardo Soares Ogasawara

Raphael Santos Carlos Machado

Abstract of dissertation submitted to Programa de Pós-graduação em Tecnologia - Centro Federal de Educação Tecnológica Celso Suckow da Fonseca, CEFET/RJ, as partial fulfillment of the requirements for the degree of master.

The technological developments are providing a new form of communication and exchange of information between people and organizations through the connection of thousands of devices. These communications happen in real time enabling the exchange of data and various information, whether professional or private nature, academic or fun. Along with the evolution of this connectivity arise forms of cyber attacks. The attacks may have social, political or financial nature and are characterized by unauthorized access, information theft and stop services. In search by the guard of the information, protective measures are implemented defined as Security Information, seeking to mitigate the risks involved in various forms of communication, aiming at the protection of the most diverse information contained in electronic media. The first step in the treatment of a cyber attack is to be identified, which can be performed using specific equipment, set with IDS (Intrusion Detection System), where you analyze network traffic in order to identify anomalies. This paper presents the Collector Access a workflow for building dataset, automatically and systematically, based on marking with free flow for the analysis of information security attacks, focusing on DoS attacks service. This study aims to collaborate in the reproduction of datasets with marking, enabling the training of detection systems, with the characteristic of having the streams labeled.

Keywords:

Security Information; Cyber Attacks; Scientific *Workflow*s

Rio de Janeiro

2015, May

Sumário

Capítulo I - Introdução 1
Capítulo II - Sistemas de Detecção de Intrusão 6
II.1 Ataque Cibernético 6
II.1.1 Classificação dos Ataques Cibernéticos 7
II.2 Detecção de Intrusão 8
II.3 Métodos de detecção de intrusão 8
II.3.1 Método de detecção baseado em conhecimento ou assinatura 10
II.3.2 Método de detecção baseado em comportamento 11
II.4 Formação de *Datasets* para Detecção de Intrusão 11
II.4.1 *Datasets* Baseados em Pacote de Dados 12
II.4.2 *Datasets* Baseados em Fluxo de Dados 12
Capítulo III - Workflows 15
III.1 Introdução 15
III.2 Formas de Representação de *Workflows* 16
III.2.1 Diagrama de Atividades 16
III.2.2 Álgebra de Workflows 17
III.3 Sistemas de *Workflows* 20
Capítulo IV - Análise Bibliográfica 25
IV.1 *Datasets* para Classificação de Ataques Orientados a Pacotes 25
IV.2 *Datasets* para Classificação de Ataques Orientados a Fluxo 27
IV.3 *Datasets* para Identificação de DoS 28
IV.4 Processos para Produção de DataSets 28
IV.5 Síntese dos trabalhos relacionados 30
Capítulo V - Access Collector: Um *workflow* para produção de datasets 32
V.1 Arquitetura do Workflow Access Collector 32
V.2 Diagrama de Atividades do Access Collector 33
Capítulo VI - Avaliação Experimental 35
VI.1 Ambiente 35

VI.2 Geração do Tráfego 35
VI.3 Captura do Tráfego 37
VI.4 Processo de Marcação dos Fluxos 38
VI.5 O Processo de conversão em Fluxo de Dados 39
VI.6 O Dataset Criado 39
VI.7 Analise Preliminar por meio de fluxo de dados 40
Capítulo VII - Considerações Finais 44
VII.1 Conclusão 44
VII.2 Publicações e Trabalhos Futuros 45

Lista de Figuras

Figura I.1 - Crescente Aumento de Incidentes de Segurança 1
Figura I.2 - Incidentes Reportados ao CERT.br em 2014 2
Figura I.3 - Funções do Firewall e IDS 3
Figura II.1 - Wireshark 9
Figura II.2 - Cabeçalho de um pacote 12
Figura II.3 - Padrão Netflow 13
Figura III.1 - *Dataset* relacional 17
Figura III.2 -Exemplo da Operação MAP 18
Figura III.3 - exemplo de Operação SplitMap 19
Figura III.4 - Exemplo de Operação de Reduce 19
Figura III.5 - Operação Query 20
Figura III.6 - Tela inicial do Sagitarii 21
Figura III.7 - Tabela de dados 22
Figura III.8 - Atividade no Sagitarii 23
Figura III.9 - Tela de criação de atividades 23
Figura IV.1 - Gráfico Gerado pelo Netflow 30
Figura V.1 - Arquitetura do Access Collector – Visão dos Processos 33
Figura V.2 - Diagrama de Atividades 34
Figura VI.1 - Representação da topologia do ambiente 36
Figura VI.2 - *Three Way Handshake* 37
Figura VI.3 - Visão geral do Access Collector 39
Figura VI.4 - Identificação do tráfego legítimo 40
Figura VI.5 - Identificação do Ataque SYN Flood 41
Figura VI.6 - Análise do Fluxo 41
Figura VI.7 - Fluxo geral do servidor 42
Figura VI.8 - Ataques por um período de 10 minutos 43

Lista de Tabelas

Tabela III.1 - Operação Algébrica 18
Tabela IV.1 - *Datasets* baseados em pacotes 27
Tabela IV.2 - Resumo dos Artigos 31
Tabela VI.1 - Dados dos pacotes capturados 38
Tabela VI.2 - Atributo dos Fluxos 38

Lista de Abreviaturas e Siglas

ABNT – Associação Brasileira de Normas Técnicas

CERT – Centro de Estudos, Resposta e Tratamento de Incidentes de Segurança do Brasil

CSV – *Comma Separated Values*

DoS – *Deny of Service*

Gbps – Giga bits por segundo Gbps

HDs – *Hard Disks*

HIDS –*Host Intrusion Detection System*

HTTP – *Hypertext Transfer Protocol*

ICMP – *Internet Control Message Protocol*

IDS – *Intrusion Detection System*

IETF – *Internet Engineering Task Force*

IPFIX – *IP Flow Information Export*

IPS – *Intrusion Prevention System*

KDD – *knowledge-discovery in databases*

NIDS – *Network Intrusion Detection System*

OSI – *Open Systems Interconnection*

RFC – *Request for Comments*

SQL – *Structured Query Language*

TCP/IP – *Transmission Control Protocol / Internet Protocol*

UDP – *User Datagram Protocol*

UDP – *Coordinated Universal Time*

UML – (*Unified Modeling Language)*

WFMC – *Workflow Management Coalition*

Capítulo I - Introdução

A Internet vem sendo utilizada cada vez mais na comunicação para troca de e-mails, pesquisas educacionais, acesso a redes sociais ou até mesmo gestão de infraestrutura críticas, como distribuição de água e energia. Uma miríade de dispositivos ficam conectados a um único ambiente virtual. Tal espaço, utilizado como facilitador na conectividade entre pessoas, empresas e dispositivos é denominado como espaço cibernético. O espaço cibernético pode ser considerado como um espaço não físico, composto por meios computacionais, onde pessoas físicas e jurídicas podem se comunicar (MANDARINO JUNIOR; CANONGIA, 2010).

O aumento da conectividade, proporcionado pela evolução tecnológica, está facilitando a realização de serviços e formas de trabalho, mas também abrindo portas para roubos de informações, acessos indevidos e ações sigilosas ilícitas de efeitos grandiosos.

De acordo com os dados do Centro de Estudos, Resposta e Tratamento de Incidentes de Segurança do Brasil (CERT.BR, 2014) pode-se observar (Figura I.1), o aumento significativo dos incidentes relacionados à segurança da informação. No ano de 2014, conforme observado na Figura I.2, as maiores incidências de ataque foram fraude, *scan*, e *DoS (Denial of Service)* (CERT.BR, 2014).

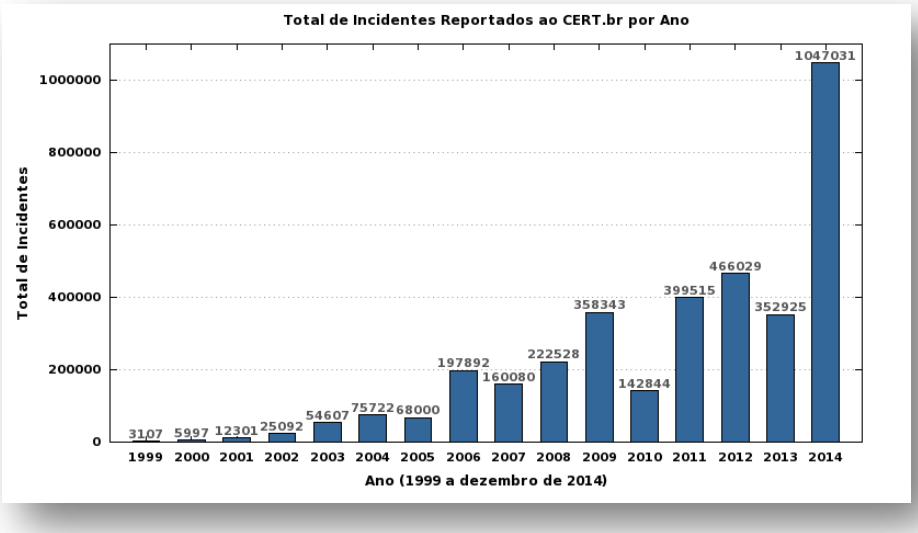


Figura I.1 - Crescente Aumento de Incidentes de Segurança

Fonte: (CERT.BR, 2014)

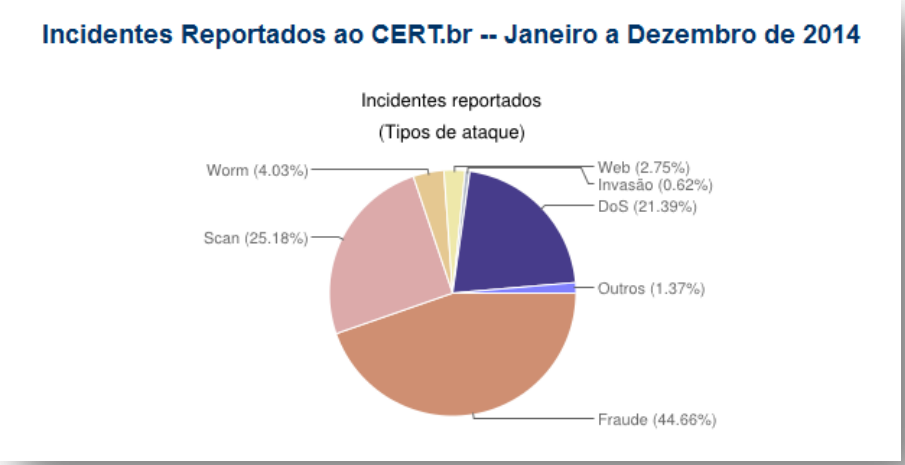


Figura I.2 - Incidentes Reportados ao CERT.br em 2014

Fonte: (CERT.BR, 2014)

Com o objetivo de ampliar a proteção das redes e mitigar os riscos envolvidos nas trocas de informações, novas tecnologias de proteção surgiram, sendo compostas por hardware e/ou software. Nesse contexto, temos os *firewalls*, IPSs (*Intrusion Prevention System*), IDSs (*Intrusion Detection System*), dentre outros. Os *firewalls* são dispositivos constituídos pela combinação de software e hardware, utilizados para segmentar e controlar o acesso entre redes de computadores, permitindo ou negando o tráfego na rede.

Os IDSs são software, hardware ou combinação de ambos utilizados para detectar tráfego suspeito. Um IDS pode ter diferentes capacidades dependendo de quão complexo e sofisticado sejam os seus componentes. Um IDS pode ser classificado operacionalmente como baseado em estação (HIDS – *Host IDS*), onde são monitoradas as características de uma única estação ou baseado em rede (NIDS – *Network IDS*), onde é monitorado o tráfego de rede (STALLINGS; BROWN, 2014). Os IDSs são instalados em pontos estratégicos de uma rede de computadores, para escutar todo o tráfego e assim poder analisa-lo.

O IDS tem como objetivo detectar atividades suspeitas, impróprias, incorretas ou anômalas, sendo capaz de detectar ataques, trabalhando como uma câmera, monitorando todo tipo de tráfego podendo realizar a detecção com base na análise de pacotes ou a partir do comportamento de acesso, identificado pela análise dos logs dos tráfegos (NAKAMURA; GEUS, 2007). Dentre as funções de um IDS podemos destacar a coleta, análise, armazenamento e em alguns casos a resposta a atividades suspeitas. Conforme mostrado na Figura I.3, o tráfego pode ser inspecionado pelo IDS, que possui a capacidade de identificar por comportamento ou assinatura, um tráfego suspeito e gerar um alerta.

A identificação realizada por assinatura consiste na comparação de um tráfego que é similar a um ataque anteriormente identificado e catalogado no banco de assinaturas. A identificação baseada no comportamento compara a situação de uma rede definida como normal, de acordo com uma análise prévia, a um comportamento definido como diferente do padrão.

Com o objetivo de detectar ataques, sejam eles por assinatura ou por comportamento suspeito, os IDS realizam a inspeção do conteúdo dos pacotes de rede. A grande dificuldade na realização da análise de pacotes em uma rede de alta velocidade, como por exemplo, redes que trafegam a velocidade de Giga bits por segundo (Gbps), é o grande volume de dados gerados. Face ao grande volume, a análise em tempo real requer equipamentos de alto desempenho, com grande poder de processamento (SPEROTTO et al., 2010).

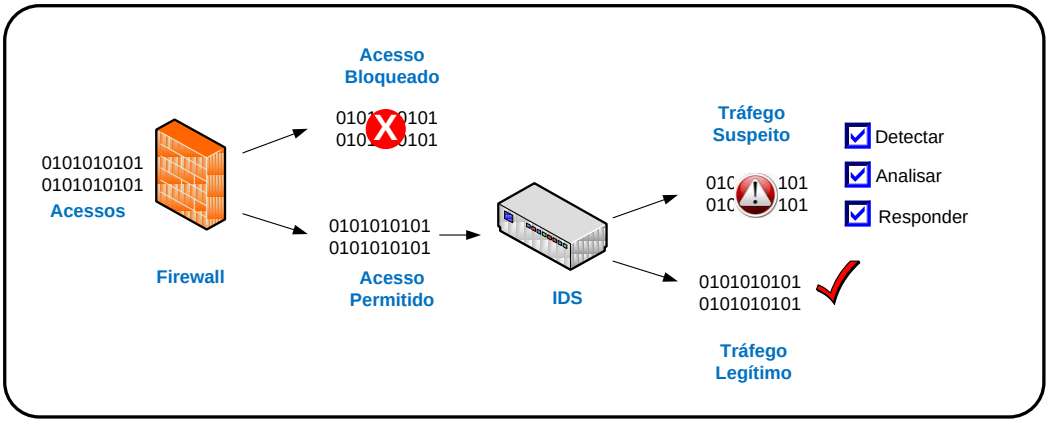


Figura I.3 - Funções do Firewall e IDS

Fonte: Próprio Autor

A análise baseada em comportamento é uma alternativa para a análise de rede de alto desempenho, assim como uma alternativa para a detecção de ataques de negação de serviço. Para este tipo de detecção, pode ser utilizada a análise baseada no fluxo de pacotes, que consiste no agrupamento de informações referentes a uma comunicação específica na rede, onde é possível identificar o par de endereços IP e informações agregadas, como IP de origem e destino, porta de origem e destino, assim como o protocolo utilizado na comunicação (SPEROTTO et al., 2010).

Considerando ainda a detecção de ataques por meio da análise de pacotes, deve-se considerar a grande dificuldade de detecção de ataques de negação de serviço, utilizando este método, pois tal ataque pode ser realizado com o uso de pacotes de dados que não gerem nenhum tipo de ação suspeita. Os ataques de negação de serviço (*Denial of Service* – DoS) se caracterizam por consumirem os recursos de servidores e roteadores e impedem que usuários legítimos tenham acesso a um determinado serviço. Desta forma, podem enviar pacotes

definidos como normais, porém em uma grande quantidade, o que provocará a lentidão ou paralisação de determinado servidor ou serviço.

Para realizar a análise de *dataset* de fluxo voltadas à identificação de ataques, em especial aqueles relacionados à negação de serviço, faz-se necessária a utilização de ferramentas capazes de realizar descobertas em um determinado *dataset* (HAN; KAMBER, 2011). As técnicas de mineração de dados (ou *Data Mining*) podem auxiliar na classificação destes fluxos visando à identificação de ataques.

Para tanto, torna-se vital a existência de *datasets* baseados em fluxos de dados com comportamentos variados, considerando tráfego dito normal e tráfego relativos a ataques, de forma a colaborar na avaliação das marcações dos fluxos, proporcionando o auxílio aos classificadores na identificação e diferenciação de ataque e acessos normais, propiciando uma boa forma de treinamento de IDS.

A possibilidade de construção de dataset de acordo com a característica necessária para qualquer tipo de avaliação é uma das proposições desta pesquisa, visando uma forma sistemática e automática para a geração dos dados. Com a utilização do *workflow*, os *datasets* poderão ser gerados de forma automática, considerando o fluxo de dados, por possuir um volume menor de informações, possibilitando a sua reprodução com as características necessárias para cada avaliação. A possibilidade de marcação dos fluxos é o diferencial desta pesquisa, propiciando a facilidade na identificação dos fluxos relacionados a ataques e os fluxos definidos como acesso comum. Tal estudo visa colaborar na reprodução de *datasets* com marcação, possibilitando o treinamento de sistemas de detecção, visto a grande dificuldade de encontrar bases de dados para este fim e com a característica de possuir os fluxos rotulados.

O presente trabalho analisa tal problema e estabelece uma abordagem para geração de *dataset* de fluxo de dados marcados para identificação de ataques de negação de serviço, tendo como objetivo geral a construção de datasets para treinamentos de IDs. Para tanto, foi desenvolvido um *workflow* definido como *Access Collector*, de forma a garantir a definição de métodos sistemáticos para a geração do *dataset,* considerando nesta pesquisa a análise de ataques de negação de serviço do tipo Syn Flood. A geração do dataset de forma sistematizada é de grande relevância, uma vez que abre espaços para a produção de novos *datasets* com outros perfis de ataques de modo automático. Os *datasets* gerados poderão ser utilizados para a validação e treinamento dos sistemas de detecção, definidos como detectores de intrusão.

Além desta introdução, o trabalho está dividido em mais seis capítulos. O capítulo II apresenta o referencial teórico, contendo uma introdução aos ataques cibernéticos, seus conceitos gerais e demais definições necessária para o entendimento do experimento realizado. O capítulo III apresenta os conceitos referentes a *workflow*, as formas de

representação e os sistemas de *workflow,* destacando a apresentação do sistema de *workflow* Sagitarii, utilizado na pesquisa. O capítulo IV apresenta os trabalhos relacionados a este estudo. O capítulo V apresenta o *workflow* desenvolvido para a criação de *datasets*, definido como *Access Collector*. O capítulo VI, apresenta a avaliação experimental, contemplando todos os passos executados para a realização do experimento. O capítulo VII apresenta os resultados, contribuições e proposta para estudos futuros.

Capítulo II - Sistemas de Detecção de Intrusão

A área de segurança da informação tem despertado cada vez mais interesse de organizações e de pessoas que visam proteger um de seus maiores bens: a informação. A segurança da informação é a área responsável pela segurança e proteção das informações. A informação é um dos maiores ativos de uma organização, portanto, um bem que deve ser protegido, tendo em vista a sua importância e seu valor. Os requisitos de confidencialidade, integridade, disponibilidade, autenticidade e legalidade orientam a análise, o planejamento, a implantação e o controle da Segurança da Informação (MANOEL, 2014).

Neste diapasão, este capítulo tem como objetivo descrever a fundamentação teórica das tecnologias relacionadas para o desenvolvimento da abordagem. Na seção II.1 são descritos os conceitos relativos a ataques cibernéticos e as respectivas classificações. Na seção II.2 é abordado o conceito de detecção de intrusão e em seguida, na seção II.3, os métodos de detecção. Na seção II.4 é apresentado as formas de produção de *datasets* para avaliar detecção de intrusão. Finalmente, a seção II.5 apresenta o conceito de mineração de dados e as técnicas relativas a esta área.

II.1 Ataque Cibernético

A temática relacionada a ataque cibernético está tomando força a cada dia, devido à dependência cada vez maior da tecnologia para a realização das mais diversas atividades, sejam elas profissionais ou pessoais, no mundo da internet. Os crimes cibernéticos caracterizam-se pela invasão de sistemas para a realização de roubo de informações ou paralisação de serviços. Não se limitam apenas aos danos diretos, como a interrupção de determinado serviço, que poderia provocar danos à determinada empresa, mas a danos indiretos a toda a sociedade, como é o caso das espionagens e outros atos antiéticos utilizados por empresas privadas e governos de alguns países, de forma a se beneficiarem de informações privilegiadas.

Com objetivos que podem ter cunho financeiro, político ou até mesmo pessoal, por meio de programas maliciosos, informações podem ser obtidas indevidamente, computadores podem ser manipulados silenciosamente e paralisações de serviços podem ser realizadas. Estes programas maliciosos podem chegar até os sistemas de várias formas, podendo gerar danos irreparáveis, interrupções de serviços por várias horas, provocando assim grandes perdas financeiras, seja por roubo de informações ou por danos causados a uma marca.

O crescente número de ataques realizados na ultima década demonstra a quantidade de ameaças. Várias corporações enfrentam ataques diversos com o objetivo de imobilizá-la, introduzir vírus, coletar dados confidenciais ou impedir as comunicações (NAÍM, 2013).

Conforme Branquinho et al. (2014), a ciberguerra, também conhecida como guerra cibernética, é uma modalidade de guerra na qual o conflito não ocorre com armas físicas, mas por intermédio da confrontação com meios eletrônicos e informáticos no chamado ciberespaço.

II.1.1 Classificação dos Ataques Cibernéticos

Os ataques podem ser classificados como físico ou lógico. Os ataques físicos, conforme (CAVALCANTI, 2014), se caracterizam pelo roubo de equipamentos como: HDs (*Hard Disks*), *pen drives*, entre outros. Os dispositivos podem ser retirados de empresas ou roubados de executivos com o objetivo de se obter informações confidenciais e sigilosas, de forma a se obter algum tipo de benefício.

Os ataques lógicos se caracterizam por acessos a sistemas de computadores, com objetivo de roubar, modificar informações ou paralisar serviços. Como exemplo de ataques lógicos pode citar o *Scan* que é uma técnica que realiza a varredura de uma determinada rede para identificar computadores e serviços ativos e os *Worms* que são programas semelhantes ao vírus, com a diferença de que são capazes de se propagarem automaticamente entre máquinas conectadas a rede ou através de *pen drives*.

Os ataques por negação de serviço (DoS), do inglês *Denial of Service*, diferentemente da maioria dos ataques, não visa invadir um computador para extrair informações confidenciais, como números de cartões de crédito e senhas bancárias, ou modificar o conteúdo armazenado em computadores. Estes ataques têm como objetivo tornar indisponíveis os serviços. Nenhum dado é roubado, nenhuma informação é alterada. O equipamento atacado simplesmente deixa de oferecer o seu serviço aos clientes legítimos.

Os ataques de DoS se caracterizam pelo uso de ferramentas automáticas, que enviam comando de solicitações a um determinado servidor, tornando-o sobrecarregando, gerando assim a sua indisponibilidade ou lentidão. Conforme Jorge e Wendt (2013) ataque consiste na perda de desempenho proposital de serviços ou sistemas, de forma a impossibilitar o uso por pessoas que possuem permissão para acessá-los.

O ataque de DoS utiliza-se de uma das características do protocolo TCP/IP (*Transmission Control Protocol / Internet Protocol),* onde um computador para estabelecer uma conexão com um servidor envia um sinal TCP conhecido como SYN (*Syncronize*) e o servidor responde a solicitação enviando um sinal de ACK (*Acknowledgement*). Um dos tipos de ataque de DoS conhecido como *Syn Flood* que é caracterizado pelo envio de vários sinais de SYN para um determinado servidor, que não consegue responder a tantas solicitações e então passa a recusar novas solicitações. Solicitações estas que podem ser legítimas, oriundas de

outros computadores, tornando desta forma o servidor sobrecarregado, podendo gerar a paralização total da comunicação e indisponibilidade de serviços.

Uma subcategoria do DoS, denominada DDoS (*Distributed DoS*), é um tipo de ataque de grandes dimensões, ou seja, que utiliza até milhares de computadores para atacar uma determinada máquina, distribuindo a ação entre elas. Ataques deste tipo já provocaram indisponibilidade em grandes sites, como CNN, Amazon, Yahoo, Microsoft e eBay (CAVALCANTI, 2014). A identificação dos ataques DDoS constituem operações com um grau de dificuldade elevado. Este fato se deve ao grande número de máquinas atacantes que podem estar envolvidas, ao uso de técnicas para forjar endereços IP (*IP Spoofing*), que escondem a origem verdadeira dos pacotes, e também à similaridade entre o tráfego legítimo e o tráfego de ataque. Desta forma, os ataques distribuídos de negação de serviço (DDoS) estão entre os ataques mais difíceis de serem prevenidos e detectados, contribuindo para um tempo de resposta mais elevado.

II.2 Detecção de Intrusão

Um sistema de detecção de intrusão tem como objetivo analisar o tráfego de rede a fim de detectar tentativas de invasões ou exploração de vulnerabilidades, sem a necessidade de interferirem no tráfego da rede. Mediante a identificação de alguma anormalidade, uma notificação é disparada (GUIMARAES et al., 2006).

De acordo com o seu funcionamento e posicionamento na rede, o IDS pode ser classificado em dois tipos: rede (NIDS – *Network IDS*) e host (HIDS – *Host IDS*). Os NIDS monitoram o tráfego da rede e devem ser instalados em pontos estratégicos para análise dos pacotes. Operam em modo promíscuo, isto é, analisam todos os pacotes que passam pela rede, destinados a qualquer endereço. Já os HIDS são instalados em um servidor específico, com o objetivo de monitorar anomalias da máquina onde está instalado. Estes analisam ações realizadas pelo sistema operacional e seus serviços. Operam em modo não promíscuo, isto é, verificam apenas os pacotes destinados ao servidor onde está instalado.

Quando o IDS atua de forma integrada com outros dispositivos de segurança existe a possibilidade da atuação de bloqueio após a detecção de anomalias. Este caso é possível, quando existe a integração com um firewall, por exemplo. O firewall possui a capacidade de bloqueio de tráfego em uma rede.

II.3 Métodos de detecção de intrusão

O IDS realiza a análise dos pacotes que trafegam na rede, com o objetivo de identificar ações suspeitas. Para tal ação, se faz necessário o uso de *sniffer*s. De acordo com Jorge e Wendt (2013), os *sniffers* possuem a finalidade de monitorar todo o tráfego de rede, de modo

que todos os dados transmitidos possam ser interceptados e analisados. Dentre os softwares com esta finalidade, podemos citar o *Wireshark*.

O software *Wireshark*, anteriormente conhecido como *Ethereal* é uma ferramenta livre utilizada para analisar o tráfego de rede. Tem capacidade de interceptar os pacotes que trafegam nas redes e de exibir detalhes das suas informações de controle e conteúdo. Desta forma é possível identificar todo o tráfego de rede, analisando os pacotes, portas e protocolos que são requisitados e enviados por um determinado computador. Os pacotes IPs (*Internet Protocol*) possuem informações sobre origem, destino e dados da mensagem. A Figura II.1 apresenta a tela de captura de dados do *sniffer* Wireshark, onde é apresentado o endereço de origem, endereço de destino, a porta utilizada e informações referentes a troca de comunicação.

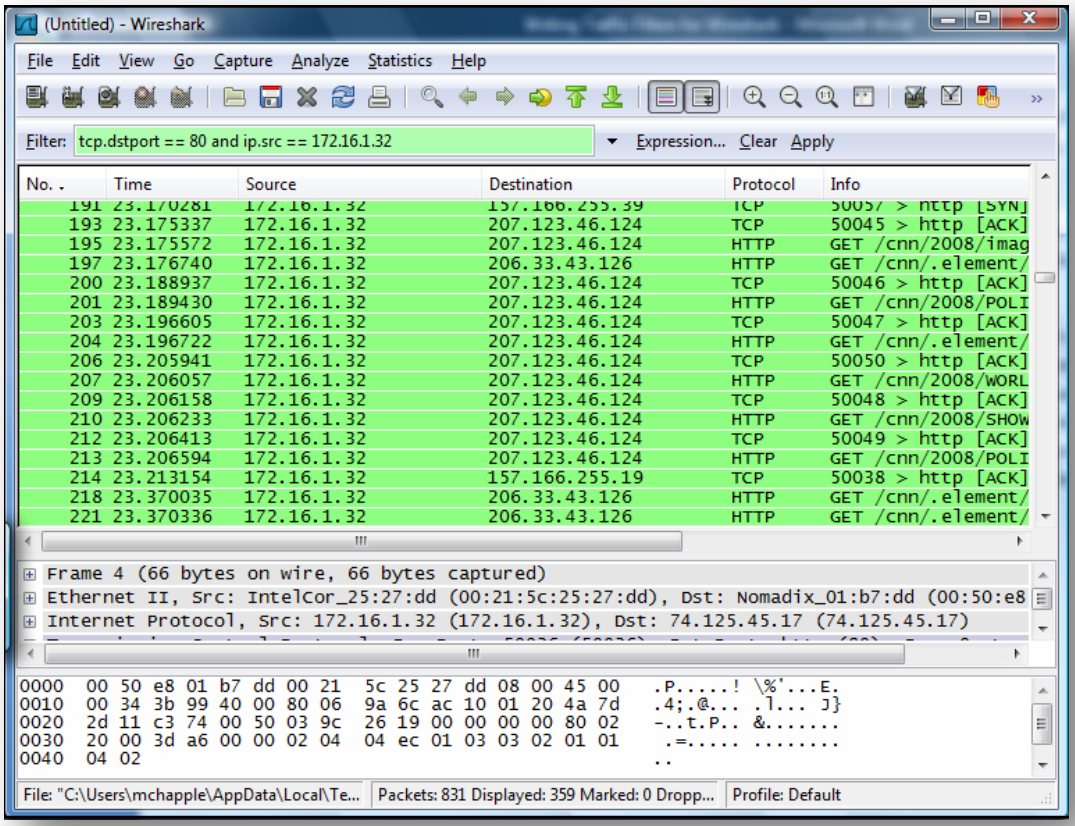


Figura II.1 - Wireshark

Fonte: Software Wireshark

Os *sniffers* capturam todos os pacotes trafegados na rede de forma a serem analisados. A análise e identificação de anomalias podem ser feita por dois métodos distintos: análise baseada em conhecimento ou análise baseada em comportamento.

II.3.1 Método de detecção baseado em conhecimento ou assinatura

A identificação baseada em conhecimento ou assinatura detecta um ataque por meio de comparações entre assinaturas de ataques. Baseia-se na observação de eventos que se assemelham a comportamentos intrusivos já conhecidos, comumente chamados de assinaturas de intrusão.

As assinaturas ou padrões de ataques são identificados e catalogados como ataques conhecidos, onde o comportamento é registrado, servindo como referência para identificação de ataques similares. “A detecção de assinatura procura detectar um ataque através de comparações de padrões de assinaturas de ataques. Quando os padrões de assinatura para um ataque correspondem ao tráfego observado na rede, um alerta é gerado.” (GUIMARAES et al., 2006).

Neste tipo de detector de intrusão uma assinatura é utilizada como parâmetro de busca para os eventos detectáveis. Desta forma, as informações são analisadas e comparadas com as assinaturas e caso seja encontrada uma correspondência, o evento em questão é detectado e considerado como uma possível ameaça.

As assinaturas são caracterizadas por registros que representam exatamente o evento o qual se deseja identificar. Por exemplo, uma assinatura pode ser baseada na análise de *strings* do campo de dados dos pacotes trafegados. Quando uma determinada *string* ocorrer em uma assinatura e for identificada em um determinado pacote, este representa uma possível atividade maliciosa (DOKAS et al., 2002). Para tal identificação, faz-se necessário desencapsular os pacotes para sua análise, o que pode provocar interferência no funcionamento da rede.

Outra desvantagem neste tipo de análise é a incapacidade de detectar ataques novos ou desconhecidos, exigindo que suas bases de regras e assinaturas sejam atualizadas frequentemente (CANDIDO; CANSIAM; SAUDE, 2005). A limitação deste tipo de sistema é que a detecção dos eventos baseia-se em ataques já conhecidos, sendo, portanto ineficiente na detecção de comportamentos intrusivos que ainda não foram descobertos, como é o caso dos novos ataques, definidos *como zero day*. A eficiência deste método depende da frequência da atualização das assinaturas de intrusão do sistema.

Outra análise é a que consiste na verificação dos cabeçalhos de pacotes, com o objetivo de identificar os mal formados. Nestes pacotes, os cabeçalhos não estão de acordo com os padrões dos protocolos, definidos nos *Request for Comments (*RFCs*),* que são os documentos que descrevem os padrões dos protocolos. Considere que pacotes TCP podem conter uma combinação de *flags* não definidas no padrão e serem utilizados para exploração de uma vulnerabilidade em aplicações que não estejam preparadas para receber esta combinação. A desvantagem desta abordagem é não possuir detalhes sobre o evento,

baseando-se apenas na checagem de cabeçalhos, apresentando assim, dificuldades na detecção de eventos como *DoS* e *DDoS* (DOKAS et al., 2002).

II.3.2 Método de detecção baseado em comportamento

A identificação baseada em comportamento ou anomalia é caracterizada por um tráfego de rede fora do padrão previsto para o ambiente. Esta técnica considera que todo comportamento caracterizado como ataque é anômalo e desta forma o sistema reage a tal comportamento que não se enquadre nos comportamentos ditos normais. A premissa básica desta análise é que atividades de intrusão ou ataques fazem parte do subconjunto composto por atividades anômalas.

A anomalia é um evento que causa uma alteração em relação ao perfil padrão do sistema e pode ser caracterizada por um ataque, falha de equipamentos, problemas de configuração, sobrecarga da rede ou uso inadequado de algum serviço ou recurso. Por meio deste método é possível identificar até mesmo as mudanças sutis em uma função (ou conjunto de dados). Essa capacidade permite detectar variações do comportamento na rede, podendo ser um indício de ataque.

Por meio de observações prévias das características de um sistema durante um determinado período de tempo torna-se possível definir o perfil de comportamento de seus diversos componentes (rede, usuários, estações, aplicações). Desta forma, esta metodologia de detecção se baseia na observação de eventos que ocorrem no sistema e sua comparação com um perfil já definido para uma possível identificação de desvios do comportamento normal, ou seja, identificação de anomalias no comportamento do sistema.

A grande vantagem deste tipo de abordagem é a possibilidade de se detectar técnicas de intrusão ainda desconhecidas, pois embora não se conheça o seu comportamento, esse deve se distinguir naturalmente do comportamento esperado baseado no perfil e análise previamente realizada. Esta abordagem vem recebendo uma crescente atenção devido a sua capacidade de detectar novos ataques, colaborando para uma atuação rápida e efetiva.

A detecção por anomalias também apresenta problemas, que é o número de falsos positivos, isto é, classificação de comportamentos normais como ataques. Isto ocorre porque nem toda atividade não usual é ilegítima ou representa um ataque.

II.4 Formação de *Datasets* para Detecção de Intrusão

A produção de *datasets* voltados para apoiar a detecção de intrusão é de grande relevância para o desenvolvimento de novos IDS. Para tanto, é necessário à existência de um elemento coletor. Este coletor normalmente é posicionando em um ponto estratégico da rede, onde consegue observar e coletar todos os dados trafegados.

No caso da avaliação de ataques relacionados à segurança da informação, o *dataset* gerado pode ser utilizado para a identificação de ataques com base na assinatura (pacotes) ou em comportamento (fluxo). Com o objetivo de padronizar a forma de coleta dos dados, protocolos como IPFIX (*IP Flow Information Export*), desenvolvido pela IETF (Internet Engineering Task Force) podem ser utilizados.

II.4.1 *Datasets* Baseados em Pacote de Dados

Todos os dispositivos conectados a rede compartilham o mesmo mecanismo para a transmissão e recepção de informações. Trabalham com o mesmo conjunto de regras que permitem a comunicação uns com os outros. Conforme (RAINER; CEGIELSKI, 2012), esse conjunto de regras e procedimentos que regulam as transmissões utilizando a rede se chama protocolo.

O protocolo TCP/IP, *Transmission Control Protocol / Internet Protocol* ou Protocolo de Controle de Transmissão / Protocolo de Internet, é responsável por gerenciar a troca de informações entre os dispositivos conectados a rede. As informações trocadas são decompostas em pequenos segmentos denominados pacotes de dados, que são transmitidos sequencialmente pela rede.

Os pacotes são divididos basicamente em duas partes: o cabeçalho que possui as informações de endereçamento de origem e destino e os dados, que é a mensagem a ser transmitida propriamente dita. A Figura II.2, apresenta a composição de um cabeçalho de um pacote e seus respectivos campos.

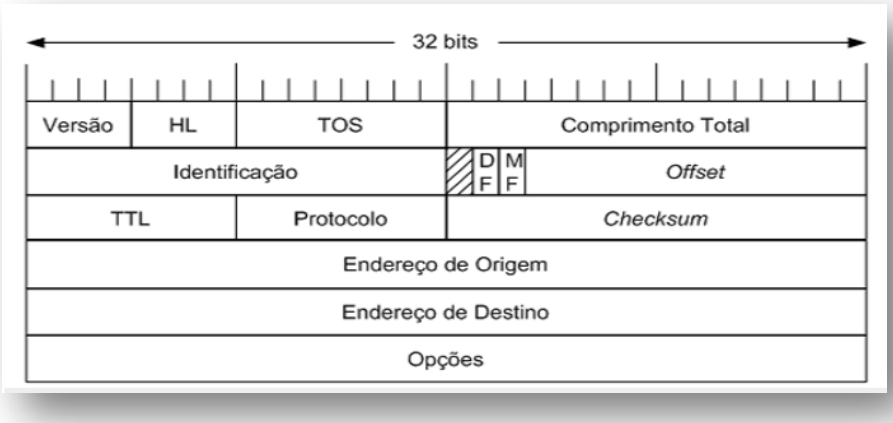


Figura II.2 - Cabeçalho de um pacote

Fonte: Adaptado de (TANENBAUM, 2003)

II.4.2 *Datasets* Baseados em Fluxo de Dados

O conceito de fluxos de dados surgiu da necessidade de uma metodologia para facilitar a análise de rede. O IETF (*Internet Engineering Task Force – Internet Society),* órgão regulador

de padrões para Internet, propôs a criação do protocolo definido como IPFIX (*IP Flow Information Export*) cuja finalidade era estabelecer uma arquitetura para análise de tráfego na Internet. O IETF definiu o IPFIX “como um conjunto de pacotes IP que passam por um ponto comum de observação na rede durante um determinado intervalo de tempo”, por meio das RFCs 3917 (IETF.ORG, 2004) e 5101 (IETF.ORG, 2008). Todos os pacotes pertencentes a um determinado fluxo possuem um conjunto de propriedades comuns.

O IPFIX, cada vez mais utilizado por administradores de rede, permite a análise e monitoramento do tráfego de uma rede de computadores de larga escala. Suas metodologias de análise consomem baixo custo computacional se comparadas à metodologia de análise de pacotes. O IPFIX fornece uma série de especificações para a sumarização de informações da rede, possibilitando de forma ampla a análise do tráfego, incluindo a detecção de eventos de segurança.

Padrões similares como o Netflow foram implementados em roteadores da *Cisco Systems*. O Netflow da Cisco analisa sete atributos do tráfego: Ip de origem, ip de destino, porta de origem, porta de destino, protocolo, classe de serviço e interface de entrada (CISCO, 2014). Considerando-se um intervalo de tempo entre requisições, todos os pacotes com o mesmo IP de origem/destino, portas de origem/destino, protocolo e classe de serviço são agrupados em um fluxo de pacotes e então são contados e condensados em um *dataset* chamado cache Netflow. A Figura II.3 apresenta o processo de coleta e armazenamento dos dados, de acordo com o padrão Netflow.

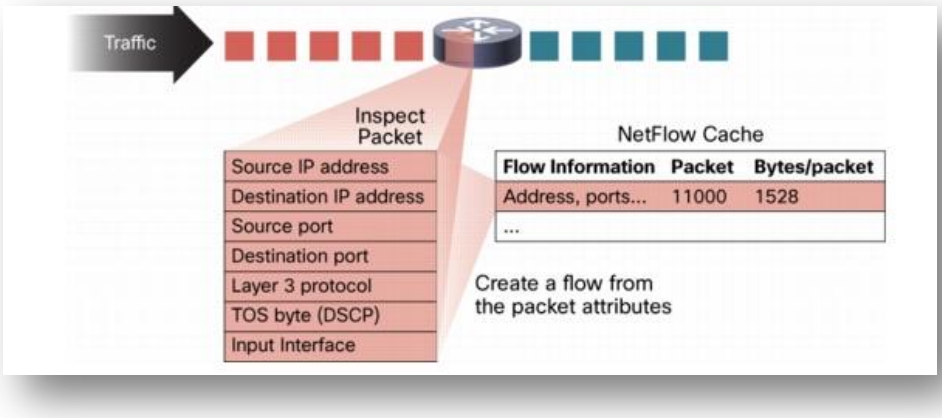


Figura II.3 - Padrão Netflow

Fonte: (CISCO, 2014)

Uma vez que os fluxos gerados durante o tráfego podem ser exportados e armazenados, passam a constituir uma fonte valiosa de informações, pois são representações concisas das sessões em uma rede. É possível obter detalhes de cada conexão ou sessão estabelecida por qualquer máquina pertencente ao ambiente monitorado, fator este extremamente relevante na análise de segurança.

Os fluxos de dados garantem a escalabilidade do processo de análise de pacotes. Esta tecnologia gera informações sobre o tráfego do ambiente baseada apenas nos cabeçalhos dos pacotes, não necessitando analisar seus conteúdos, o que poderia provocar latência na rede. Isto permite o monitoramento de um grande perímetro de rede (DOKAS et al., 2002).

Capítulo III - Workflows

Este capítulo tem como objetivo descrever a fundamentação teórica relacionada a modelagem e desenvolvimento de *workflows.* Na seção III.1 são descritos conceitos de *workflow*. Em seguida, na seção III.2 são descritas as formas de representação dos *workflows*, considerando o diagrama de atividades e a álgebra dos *workflows,* contemplando suas operações. Na seção III.3 são apresentados os sistemas de *workflows*, enfatizando os sistemas baseados em álgebra de workflows.

III.1 Introdução

A definição de *workflow* surgiu incialmente para detalhar processos administrativos realizados em um escritório, decompondo-os em atividades. A *Workflow Management Coalition* (WFMC, 2000), organização global formada por analistas, consultores, universitários e pesquisadores de gerenciamento dos processos de negócios, definiu *workflow* como sendo a automação de processos de negócios, nos quais atividades são distribuídas entre os participantes de acordo com um conjunto de regras.

Workflows podem ser implementados, portanto, para definição de processos de negócios e também para a estruturação de dados, com o objetivo de se obter resultados. Sendo assim, um *workflow* envolve a combinação de dados e processos, em uma sequência estruturada de passos, visando à solução de um problema. É composto pelo sequenciamento, parametrizações dessas atividades, bem como pela definição dos dados a serem processados e pelo controle dos fluxos necessários. Após a execução do *workflow*, pode-se realizar a análise dos resultados. Durante o processo de análise, o especialista deve consultar a execução das atividades e identificar se ocorreu algum erro na execução ou se existe algum fator que mereça destaque na avaliação. É de grande importância que os testes e seus resultados possam ser reproduzidos, com o objetivo de se obter a validação do processo automatizado.

Desta forma, a ciência da computação vem colaborando na realização e exploração de dados pela automatização dos processos experimentais (DEELMAN et al., 2009). A medida que os computadores evoluem, proporcionam um processamento cada vez maior de dados, possibilitando o aumento da complexidade e volume dos dados a serem processados, assim como as atividades a serem encadeadas para a execução dos *workflows*. Os *workflows* visam a apoiar os analistas, de modo que eles possam manter o foco na realização da ciência e não nos aspectos computacionais (OGASAWARA et al., 2011)

III.2 Formas de Representação de *Workflows*

Os *workflows* podem ser representados de formas variáveis, podendo tomar diferentes formatos e se apresentarem em diferentes níveis de abstração (MATTOSO et al., 2008). De forma a apoiar a composição dos modelos de *workflow,* ferramentas visuais permitem a modelagem e execução das atividades a serem realizadas por estes sistemas.

Os *workflows* podem ser modelados e instanciados para execução em várias linguagens de representação. Dentre as formas de representação para *workflows*, destaca-se o Diagrama de Atividades implementado na Linguagem de Modelagem Unificada (do inglês, UML) (FOWLER, 2004). No que tange a instanciação para execução existe uma miríade de linguagens (DEELMAN et al., 2009) e sistemas para execução (CALLAHAN et al., 2006; DEELMAN et al., 2007; HULL et al., 2006; WANG; CRAWL; ALTINTAS, 2009), cada um tem sua vantagem e plataforma computacional. Visando abstrair esses aspectos, o foco deste trabalho consiste em adotar as representações baseadas em álgebra de *workflows* (OGASAWARA et al., 2011), que introduz uma relação de consumo e produto entre as atividades, mas ao mesmo tempo, mantém um isolamento das questões de plataformas computacionais e de linguagens para instanciação de *workflows*.

III.2.1 Diagrama de Atividades

O modelo UML (*Unified Modeling Language)* é a linguagem de modelagem que permite representar um sistema orientado a objetos de forma padronizada, auxiliando na visualização do desenho e a comunicação entre objetos. Permite que o produto do desenvolvimento seja representado por diagramas padronizados.

Uma das formas de modelagem presentes na UML é o diagrama de atividades, que pode ser usado para representar as dependências entre as atividades, sendo utilizada também na modelagem de *workflows*. O diagrama de atividades pode ser usado tanto para elicitação (MARTINHO et al., 2009), que é o processo de criação do *workflows*, quanto para execução (QIN; FAHRINGER; PLLANA, 2007) de *workflows*.

O diagrama de atividades pode ser usado para representar fluxos de atividades simples, que ocorrem de forma sequencial ou para representar fluxos simultâneos. Possui muitas semelhanças com os antigos fluxogramas utilizados para desenvolver a lógica de sistemas. É utilizada para representar uma sequencia de atividades ou um fluxo de dados dentro de determinado sistema. No caso do *workflow*, representa o conjunto de atividades a serem realizadas, que podem incluir a manipulação de dados existentes ou a geração de novos dados.

III.2.2 Álgebra de Workflows

Os *workflows* podem ser representados por meio de uma abordagem algébrica onde as atividades são regidas por operações algébricas e os dados consumidos e produzidos são representados por relações (OGASAWARA et al., 2011). A abordagem algébrica possibilita abordar o problema de execução paralela de *workflow* de modo análogo à otimização de consultas em bancos de dados relacionais.

Os bancos de dados relacionais modelam os dados de forma a serem representados por tabelas, ou, mais formalmente, relações. O modelo relacional consiste no uso de relações. As linhas contidas nas relações são definidas como tuplas. Cada coluna das relações corresponde a um atributo. As relações se relacionam entre si por atributos chaves, que pode ser um ou mais atributos que determinam a unicidade de cada tupla. A Figura III.1 apresenta um banco de dados relacional, onde as tabelas representadas por relações são relacionadas por atributos chave (primária e estrangeira).

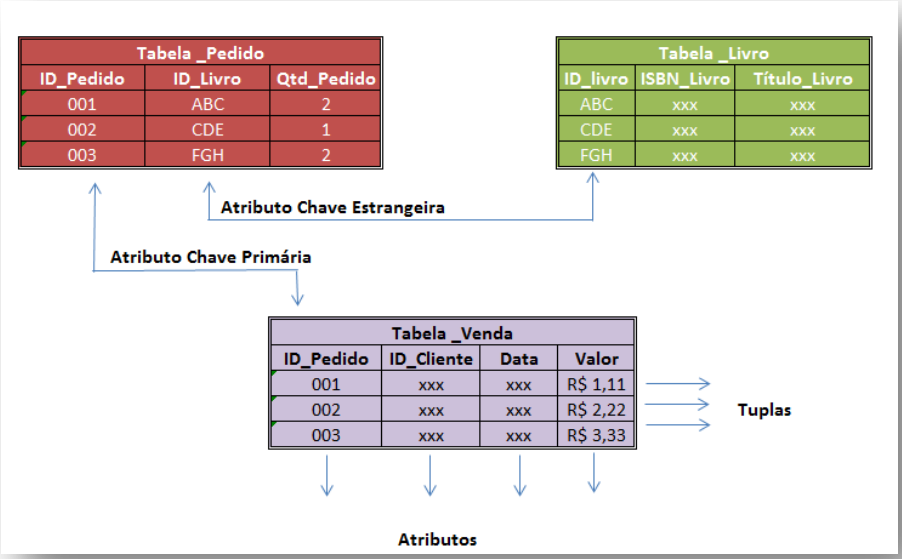


Figura III.1 - *Dataset* relacional

Fonte: Próprio Autor

No contexto da álgebra de *workflows*, os dados são uniformemente representados por meio de relações e as atividades do *workflow* são representadas por operações algébricas que possuem semântica sobre a produção e consumo dessas relações. A representação das operações algébricas possibilita a reescrita de *workflows*, criando expressões equivalentes, porém, com desempenho distinto.

A álgebra definida neste estudo compreende quatro operações: *Map*, *SplitMap*, *Reduce* e *Query,* representadas na Tabela III.1 (OGASAWARA et al., 2011). A diferença básica entre

estes operadores está no tipo de atividade executada e na indicação dos dados produzidos e consumidos pelas atividades.

Tabela III.1 - Operação Algébrica

Operação	Tipo de Atividade	Consumo/Produto	Relação
Map	Programa	Tupla a Tupla	1:1
Split Map	Programa	Produz Múltiplas Tuplas	1:n
Reduce	Programa	Agrega Tuplas	n:1
Query	Exp. Algébrica relacional	Consulta	

Fonte: Próprio Autor

A operação *Map* consome a relação de entrada, uma tupla por vez. Para cada tupla consumida, tem-se uma tupla produzida. Para exemplificar, considere uma relação com tuplas onde são listados os pedidos de venda de uma determinada loja. A medida que cada tupla com os pedidos é lida, uma instância de atividade para coleta de material no estoque é ativada (OGASAWARA et al., 2011), conforme apresentado na Figura III.2. Para cada tupla consumida, gera-se uma tupla como resultado da alocação do pedido para um caminhão com a sua respectiva data de entrega.

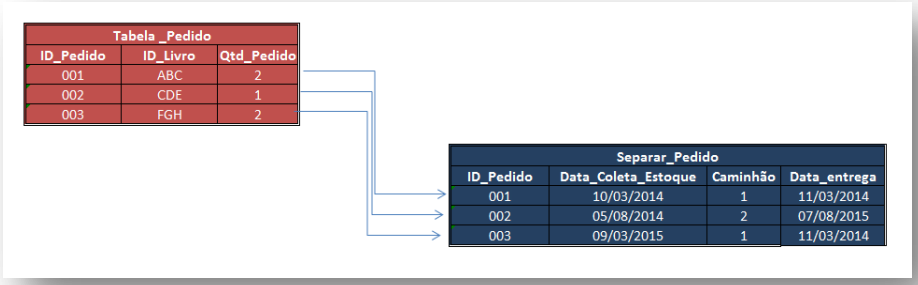


Figura III.2 -Exemplo da Operação MAP

Fonte: Próprio Autor

A operação *SplitMap* produz múltiplas tuplas a medida que for realizando o consumo das tuplas da relação de entrada. Desta forma, conforme apresentado na Figura III.3, podemos considerar que a realização de uma determinada atividade executando a operação *SplitMap*, onde o primeiro pedido da relação de entrada produz três tuplas associadas na relação de saída. Exemplificando, para cada pedido realizado, duas ou mais tuplas são geradas, considerando as respectivas datas de pagamento que são parceladas.

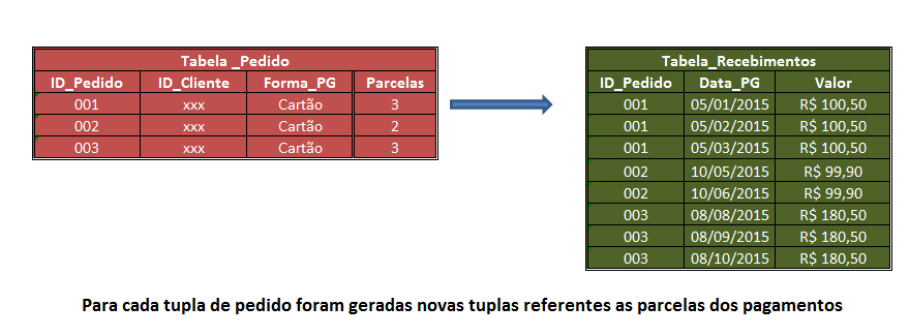


Figura III.3 - exemplo de Operação SplitMap

Fonte: Próprio Autor

A operação *Reduce* atua de forma oposta a *SplitMap*, agregando as tuplas de uma relação de entrada e consolidando-as na relação de saída, considerando um atributo de agrupamento. A Figura III.4. apresenta a relação dos pedidos realizados, onde cada tupla é lida e agrupada de acordo com o atributo data de entrega.

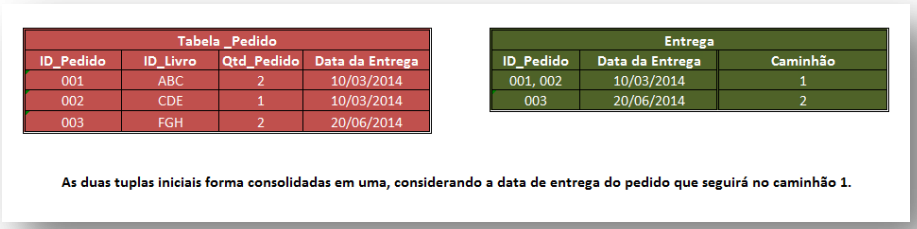


Figura III.4 - Exemplo de Operação de Reduce

Fonte: Próprio Autor

Na operação *Query* buscas são realizadas nas relações com o objetivo de consultar informações desejadas. A consulta pode ser especificada por meio de uma linguagem declarativa, como o SQL (*Structured Query Language*). A operação de *Select* busca informações de acordo com a consulta definida. Considerando a consulta a relação que contém informações dos clientes, se deseja identificar quais clientes estão com informações incompletas. A Figura III.5 apresenta a execução do comando *Select* que retornará as tuplas referentes aos clientes que possuem informações incompletas no cadastro, como endereço, e-mail e telefone.

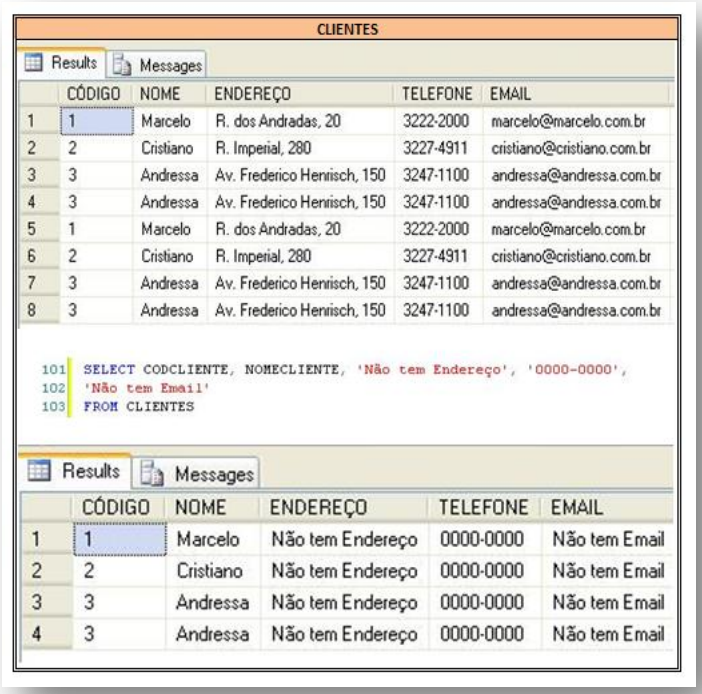


Figura III.5 - Operação Query

Fonte: Próprio Autor

III.3 Sistemas de *Workflows*

Com o intuito de permitir a composição, execução e monitoramento de *workflows*, os Sistemas de Gerência de *Workflows* (SGWf) foram desenvolvidos. Utilizando-se os sistemas de gerencia de *workflows* é possível realizar a execução dos programas e serviços, assim como a manipulação dos dados existentes num *workflow*, de acordo com o processo que se deseja representar. Os sistemas permitem a construção e execução de sequencia de atividades, normalmente apresentadas em uma interface gráfica, que abstrai parte da programação que está por trás do fluxo, auxiliando o usuário a construir um *workflow* sem precisar se preocupar com linguagens de programação.

A composição de um *workflow* requer o encadeamento de atividades, a parametrização destas, as definições dos dados de entrada e a inserção de fluxos de controle necessários. Os sistemas disponibilizam uma lista de atividades que podem ser encontradas nas bibliotecas específicas para a construção do *workflow*, que na fase de criação, podem ser escolhidas para serem realizadas, sendo parametrizadas e concatenadas por meio de ligações que representam fluxos de dados ou fluxo de controle.

Conforme mencionado anteriormente, existe uma grande quantidade de sistemas *workflows*. Entretanto, visando um isolamento entre as linguagens de *workflows* e a sua

representação abstrata, o foco do trabalho concentra-se nos sistemas que apoiam a representação por meio de álgebra de *workflows*. Pode-se destacar os sistemas algébricos de gerenciamento de *workflows*, como o Chiron (OGASAWARA et al., 2013), Scicumulus (DE OLIVEIRA et al., 2010) e recentemente o Sagitarii (ABREU; OGASAWARA, 2015), que tem como diferencial apoiar o cenário de Ciência de Dados (do inglês, *Data Science*), que é o estudo da extração de conhecimento a partir dos dados.

O sistema de *workflows* Sagitarii possibilita a execução paralela de experimentos variados em cenários de grande volume de dados. A partir dele, pode-se executar quantos experimentos forem necessários, sendo que cada experimento pode ter a sua composição de suas atividades alteradas para melhor conveniência do usuário. A Figura III.6 apresenta a tela inicial do sistema de *workflow*s Sagitarii.

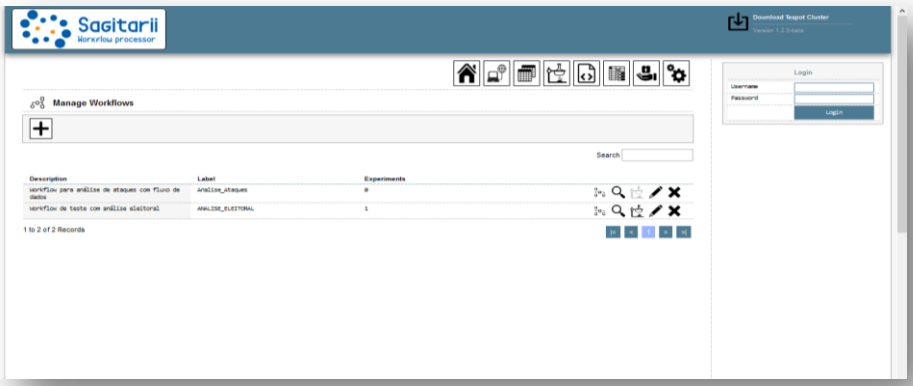


Figura III.6 - Tela inicial do Sagitarii

Fonte: Software Sagitarii

O Sagitarii faz uso do modelo relacional onde armazena a sua proveniência usando banco de dados *PostgreSQL* (POSTGRESQL, 2015), sendo necessária criação de tabelas para input e output de informações a serem consumidas e geradas, onde serão manipuladas por meio dos operadores algébricos.

A implementação do *workflow* no sistema de gerenciamento de *workflows* Sagitarii começa com uma carga da configuração da execução feita em um arquivo no formato CSV (*Comma Separated Values*). As tabelas são criadas diretamente pelo Sagitarii, conforme mostrado a Figura III.7, sendo definidas como *MAP*, compostas por uma coleção de tuplas a serem manipuladas.

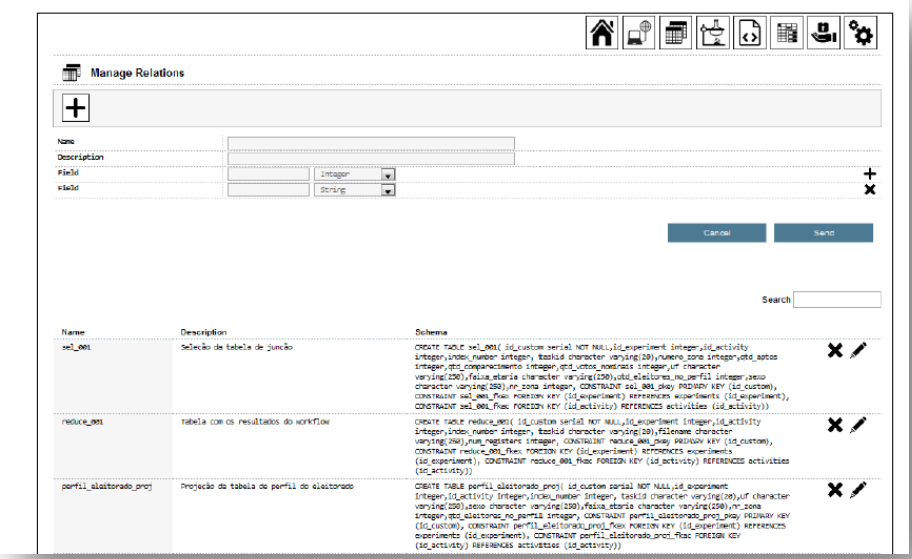


Figura III.7 - Tabela de dados

Fonte: Software Sagitarii

Durante a execução de atividades, as tabelas são consumidas de acordo com o tipo de operador da álgebra indicado. As atividades definidas no *workflow* podem ser executadas pelo servidor onde está instalado o Sagitarii ou enviadas aos nós de execução, que são computadores registrados ao servidor do Sagitarii .

As atividades executadas pelos nós computacionais são definidas como *wrapper*. Um *wrapper* é um programa, normalmente desenvolvido em Java ou Python, que recebe os dados no formato utilizado pelo Sagitarii (CSV), invoca o programa desejado e retorna com as informações de resultado da atividade executada. Os *wrappers* executam instruções que no *workflow* são definidas como uma atividade. O Sagitarii recebe os dados resultantes da execução da atividade, transformando-a no formato padrão reconhecido e encaminha para que seja armazenado no servidor.

Para que os *wrappers* sejam executados devem ser declarados no Sagitarii, para que o sistema seja capaz de enviar estes programas executáveis aos nós de execução e os nós de execução possam processar as atividades de maneira correta.

Para a realização de uma atividade no Sagitarii é

utilizado o "*Activation Executor*". Uma atividade usa um executor para cumprir a execução da atividade, que pode ser um programa externo ou uma atividade interna, como por exemplo a exeução de operadores algébricos: *MAP, SplitMap, Reduce* ou *Query* (OGASAWARA et al., 2011). O *Activation Executor* dispara a execução da atividade que será capaz de ler um *dataset* tabular com os parâmetros de configuração de entrada, produzindo como resultado outro *dataset* tabular.

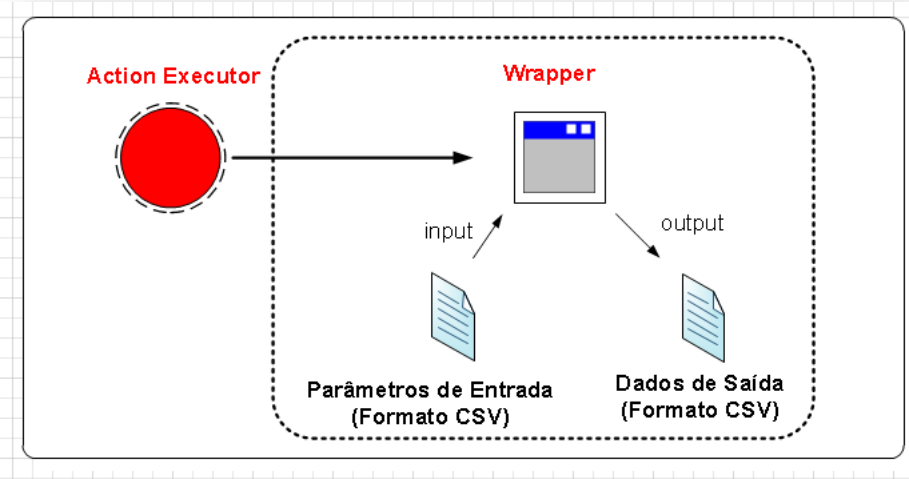


Figura III.8 - Atividade no Sagitarii

Fonte: Adaptado (ABREU; OGASAWARA, 2015)

Após realizar a criação das tabelas, wrappers e executores, é possível criar o *workflow* propriamente dito. Nesta etapa, é informado o encadeamento das atividades, o executor que irá invocar cada atividade, que tabelas servem como fonte de dados de consumo e produção e quais operadores algébricos serão implementados.

Para a organização do sequenciamento de atividades a serem realizadas pelo Sagitarii, é utilizada a opção "*Manage Activities*", onde é definido o fluxo de execução do *workflow.* Realizada a parametrização do fluxo de atividades o *workflow* estará pronto para ser executado.

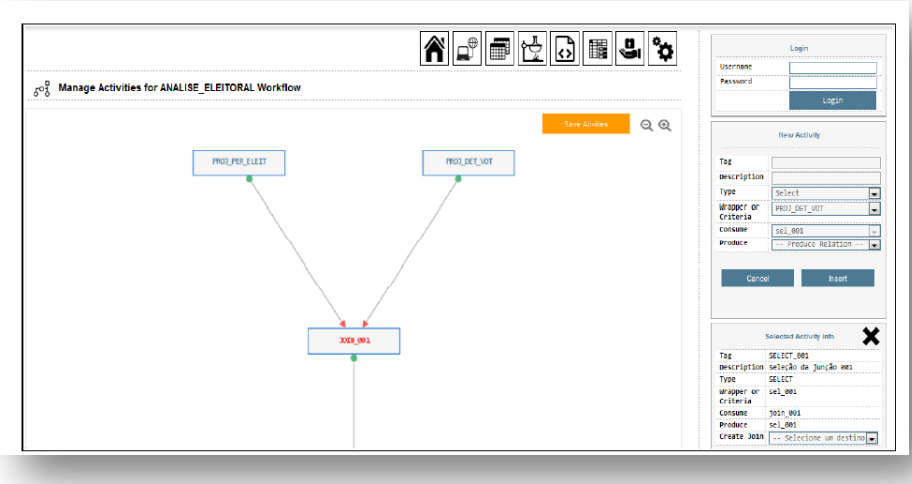


Figura III.9 - Tela de criação de atividades

Fonte: Software Sagitarii

Uma vez tendo iniciada a execução de um *workflow*, as atividades em andamento podem ser acompanhadas por meio de tela específica de acompanhamento no Sagitarii. A partir da tela são apresentados os detalhes da execução do *workflow*, como por exemplo, a fila de processamento de atividades, informando as atividades executadas e as que ainda estão na fila para execução.

Capítulo IV - Análise Bibliográfica

Existe um volume considerável de pesquisas relacionadas à avaliação do tráfego de rede com o objetivo de identificar ataques de DoS. Poucas abordagens estão relacionadas a produção de *datasets*, considerando dados sintéticos baseados em fluxo de dados, conforme apresentado nesta pesquisa.

Para avaliar os diferentes trabalhos relacionados a *datasets* para análise de ataques de DoS, foi realizada uma busca sistemática, datada de 21 de novembro de 2014, na base de dados do IEEExplorer Digital Library (IEEE, 2014), a partir das palavras chaves: "flow-based" ∧ "intrusion detection" ∧ "*dataset*". A pesquisa trouxe 67 artigos contendo informações relacionadas ao tema. A partir da avaliação dos 67 arquivos, 49 foram selecionados para leitura completa. Foram avaliados os artigos que apresentaram proposta de criação e avaliação de *dataset* relacionado a fluxo de dados para identificação de ataques de DoS. A pesquisa foi complementada com buscas usando *backward snow-balling* (JALALI; WOHLIN, 2012), visando às iniciativas relacionadas a *datasets* relacionados a ataques de segurança da informação, principalmente os relacionados a DoS e *datasets* produzidos com fluxo de dados.

IV.1 *Datasets* para Classificação de Ataques Orientados a Pacotes

Existem inúmeras abordagens referentes à análise de ataques de segurança com base na análise de pacotes de dados. Muitos especialistas têm investigado as abordagens para a detecção de ataques considerando para tanto a análise de *datasets* gerados pela coleta dos pacotes IP trafegados na rede.

Para a análise de dados, bases de dados (*dataset*) são geradas para avaliação do tráfego, de forma a identificar se de fato os tráfegos identificados são tráfegos normais de rede ou ataques. Um dos *datasets* mais utilizados para testes e avaliação de pacotes de dados é a KDD *cup*1999. Esse *dataset* foi concebido por meio da simulação em um ambiente de uma rede militar dos EUA, mais precisamente da força aérea dos Estados Unidos, sendo alimentado com conexões TCP (*Transmission Control* Protocol). Em meio a essas conexões foram inseridos múltiplos ataques, definidos como conexões ruins (KDD, 1999). Desta forma o *dataset* KDD *Cup*99 é composto de conexões “maliciosas”, denominadas ataques, e de conexões “legítimas” chamadas normais.

Ataques catalogados no projeto realizado em 1998 e 1999 pelo Laboratório *Lincoln* do MIT – *Massachussets Institute of Technology* - e patrocinado pelo governo americano através da DARPA – *Defense Advanced Research Projects Agency* - gerou um *dataset* baseado em pacote de dados. Os *datasets* foram desenvolvidos com o objetivo de realizar estudos relacionados à detecção de intrusão, com o objetivo de oferecer aos pesquisadores exemplos

de dados baseados em tráfego de redes e ataques. Inicialmente foram gerados dois *datasets*, um em 1998 e outro em 1999, que são utilizadas para testes de detecção de intrusão (DARPA, 2000).

Os testes realizados em 1998 pelo DARPA foram divididos em duas partes: uma avaliação em tempo real e uma avaliação "*off-line*", com dados de tráfego armazenados. Os sistemas de detecção de intrusão foram testados de forma "*off-line*", fazendo uso do dataset gerado pela coleta do tráfego de rede. O sistema de IDS realizou o processamento dos dados com o objetivo de analisar o tráfego normal e os ataques. A amostra do tráfego capturado foi disponibilizada a primeira vez em fevereiro de 1998.

Devido ao crescente interesse em avaliar as técnicas para a detecção de ataques complexos, no ano de 2000, o DARPA desenvolveu um novo conjunto de dados, sendo dividido em cenários com várias etapas distintas. Um destes cenários contempla um ataque distribuído de negação de serviço. Tal ataque foi executado em cinco passos: o atacante investiga a rede, explora uma vulnerabilidade, invade o host vulnerável, instala um *trojan* e lança um ataque DDoS a um servidor fora do site onde o host invadido está instalado (DARPA, 2000). No segundo cenário, também é realizado um ataque distribuído de negação de serviço. Todo o tráfego de rede foi capturado pelo *sniffer.* Tanto os resultados dos testes realizados no cenário um como no cenário dois, transformaram-se em *datasets* que são disponibilizados para avaliação.

O CAIDA (*Center for Applied Internet Data Analysis*) é um centro de estudos dedicado à análise dados da internet. É um dos grupos de pesquisa formado pela colaboração de organizações comerciais, governo e grupos de pesquisas, com base no *Supercomputer Center* (SDSC) localizado na Universidade da Califórnia. O CAIDA realiza a coleta de vários tipos diferentes de dados em diversos locais geograficamente distintos e disponibiliza em seu *site* esses dados para a comunidade de investigação, preservando sempre a privacidade dos indivíduos e organizações que doam estes dados ("CAIDA", 2014).

Outro *dataset* muito utilizado para avaliar IDS é o DefCon ("DEF CON Hacking Conference", 2015). Este *dataset* é gerando na conferência Defcon, durante uma competição entre hackers, denominada *Capture the Flag*. Durante a competição, que ocorre anualmente, cada equipe recebe um host a ser defendido. Em paralelo, outro time da competição tenta invadir o host protegido. Todo o tráfego de rede é gravado e posteriormente disponibilizado ao público. O conjunto de dados Defcon tem propriedades que o tornam muito diferente do tráfego comum identificado no "mundo real", pois contém apenas tráfego de ataques, sendo de grande valia para a avaliação e treinamento de IDS (KRUEGEL; VALEUR; VIGNA, 2005).

Tabela IV.1 - *Datasets* baseados em pacotes

Datasets	Tipo	Objetivos
KDD CUP 99	Pacote de Dados	Simulação: conexões "ruins", denominadas ataques, e de conexões "boas" chamadas normais.
DARPA (Defense Advanced Research Projects Agency) 1998, 1999 e 2000.	Pacote de Dados	Conexões diversas de rede
CAIDA (Center for Applied Internet Data Analysis) 2007	Pacote de Dados	Coleta de tráfego real para análise
DefCon – Capture the Flag (CTF)	Pacote de Dados	Ataques gerados durante competição de hackers

Fonte: Próprio Autor

IV.2 *Datasets* para Classificação de Ataques Orientados a Fluxo

Os diversos estudos relacionados a detecção de ataques considerando *datasets* baseados em pacotes de dados, acarreta em uma análise de um volume muito grande de dados. A análise baseada em fluxo de dados propicia a manipulação de um volume menor de dados e menor poder de processamento dos equipamentos.

Anna Sperotto, em suas diversas publicações, analisa os ataques de segurança da informação baseado na análise de fluxo de dados (SPEROTTO et al., 2010), considerando a análise de redes de alta velocidade. Ela destaca a dificuldade na realização de análise pacotes em redes que atuam em Gigabits por segundo (Gbps), por gerarem um grande volume de dados, exigindo equipamentos com grande poder de processamento para analisar todas as informações.

Os fluxos de dados apresentam as informações agregadas das conexões de rede de forma consolidada, informando apenas IP de origem, IP de destino, porta de origem, porta de destino e protocolo, diferentemente dos pacotes de dados, facilitando assim a capacidade de processamento, devido à quantidade reduzida de dados (SPEROTTO et al., 2010). Destaca que os estudos nesta área ainda estão em evolução, pois os primeiros artigos desenvolvidos, considerando a utilização de análise de origem e destino dos pacotes, abstraindo os seus conteúdos, começaram a surgir em 1990, destacando HEBERLEIN et al. (1990) e STANIFORD-CHEN et al. (1996).

A abordagem realizada por CORRÊA et al.(2007) em seu trabalho, apresenta uma nova metodologia para detecção de eventos em redes de computadores utilizando o protocolo *NetFlow* e o armazenamento de informações em *datasets* de fluxo de dados. Por meio de

consultas SQL aos dados armazenados é possível identificar alguns eventos na rede, incluindo ataques às redes de computadores. O trabalho propõe uma arquitetura para o armazenamento de informações, deixando a cargo de novos trabalhos a busca por metodologias mais robustas de detecção de intrusão aliadas a tal arquitetura.

IV.3 *Datasets* para Identificação de DoS

Estudos realizados por CORRÊA et al. (CORRÊA; PRÔTO; CANSIAN, 2007), apresentam a criação de um *dataset* composto por fluxos de dados. Avaliam os modelos para armazenagem dos dados no padrão IPFIX, de forma a se obter uma boa análise e monitoramento da rede. Propõe assim, a criação de um modelo de armazenamento de fluxo de redes considerando o uso do IPFIX. Também atuam na pesquisa de análise de fluxo de dados para detecção de ataques de negação de serviço (DOKAS et al., 2002), onde abordam um novo modelo de detecção de anomalias e tentativas de intrusão baseado na utilização de fluxos de dados (padrão *Netflow*). Apresentam estudos, cujo objetivo é apresentar um novo modelo para monitoramento de rede de grande porte, com o objetivo de detectar atividades maliciosas, sem afetar a sua eficiência.

Na busca da identificação de ataques de negação de serviço, estudos desenvolvidos por SUBBULAKSHMI (SUBBULAKSHMI et al., 2011) analisam *dataset* com variações de ataques de negação de serviço utilizando técnicas de mineração de dados. No cenário proposto por SUBBULAKSHMI (SUBBULAKSHMI et al., 2011) o *dataset* foi gerado de forma experimental com a inclusão de dez tipos de ataques DDoS diferentes. Usando programas para geração de pacotes foi criada uma rotina para a geração de ataques. Os ataques foram gerados utilizando um ambiente distribuído e em seguida foram capturados em um servidor por um *script*. Utilizando técnicas de mineração de dados, os tráfegos legítimos e os tráfegos de ataque foram classificados.

Os ataques foram gerados considerando a quantidade de ataques e o IP da vítima em um determinado espaço de tempo. Os ataques lançados foram do tipo ICMP *flooding*, UDP *flooding*, TCP *flooding*, Smurf *flooding, Land flooding*, *Port scan*, HTTP *flooding*, *Session flooding* and IP *flooding*. Cada tipo de ataque foi gerado em um determinado intervalo de tempo e todo tráfego gerado foi capturado gerando o dataset para análise.

IV.4 Processos para Produção de DataSets

Um grande desafio na avaliação e realização de testes em dataset relacionados a ataques de negação de serviço é a falta de *datasets* públicos. Desde 2002, apenas três *datasets* públicos são disponibilizados para a análise, entre elas o KDD CUP 99, a base de

ataques disponibilizada pelo DARPA (*Defense Advanced Research Projects Agency*) e o CAIDA, *dataset* gerado em 2007 contendo tráfegos gerados com ataques de negação de serviço. Tais *datasets* públicos são baseadas em pacotes de dados.

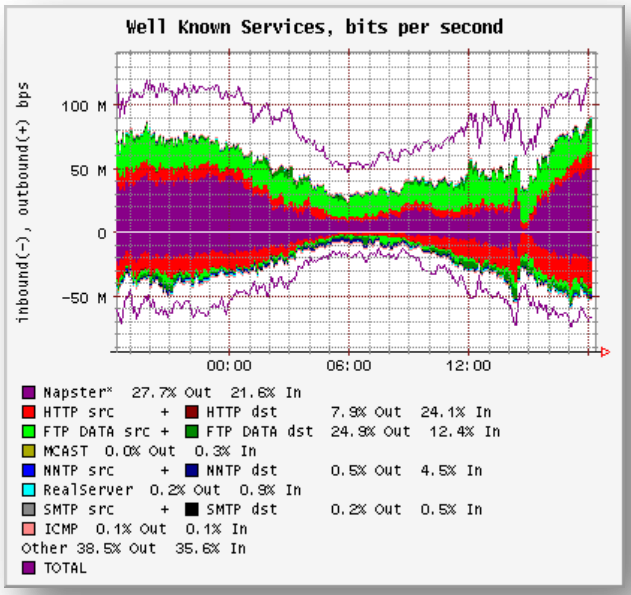
Pesquisas destacadas a seguir analisam o monitoramento de rede e destacam a construção *datasets* de fluxo de dados. Mostram que os sistemas de monitoramento conseguem fornecer informações relacionadas a fluxo de dados, com informações consolidadas, como IP de origem, IP de destino, porta de origem, porta de destino, tempo de execução e bytes trocados. A pesquisa realizada por HOFSTEDE et al (2013) destaca as características para a geração de fluxo de dados utilizando *Netflow* e IPFIX, com o objetivo de identificar ataques de negação de serviço. Os pacotes trafegados na rede são encaminhados por um switch ou roteador ao "*Flow Explorer*" que agrega os pacotes, convertendo-os em fluxos. Quando os fluxos são considerados como concluídos, os dados são enviados para o para o "*FlowCollector*" onde são armazenados e processados. Finalmente, ferramentas de análise, como um IDS, verificam os fluxos com o objetivo de identificar anormalidades (HOFSTEDE et al., 2013). Foi analisado neste estudo, o atraso na geração dos fluxos versus a detecção de ataques de negação de serviço. Propõem uma forma de elaboração de fluxo de dados mais rápida, de forma a ser mais eficiente na detecção de ataques de negação de serviço.

O estudo desenvolvido por ZHONG et al. (2007) apresenta a análise de tráfegos anormais considerando a avaliação de fluxo de dados. O estudo propõe uma forma de agregação de dados de tráfego por fluxo, como uma forma de aperfeiçoar o uso de recursos de processamento, como CPU e memória. Nesse estudo são utilizados cinco campos para compor o fluxo de dados: IP de origem, IP de destino, protocolo, porta de origem e porta de destino. Consideram a análise destes cinco campos suficientes para análise de ataques de negação de serviço, *worms* e *port scan*, pois estes tipos de ataques possuem o mesmo IP de origem ou IP de destino, porém nem sempre tem o número de portas definidos. Consideram a análise dos ataques de negação de serviço onde os pacotes tem o mesmo endereço IP de destino, podendo apresentar vários IPs de origem, como é o caso dos ataques de DDOS (*Distributed Denial of Service*). Analisam os fluxos de ataques por vírus, onde o IP de destino é o mesmo.

Neste estudo é utilizado o software fprobe (CSWFPROBE, 2014) para a estruturação dos dados. Tal ferramenta realiza a coleta de dados relacionados ao tráfego de rede, gerando o fluxo. A partir da criação do *dataset* foi criado um cluster para a análise do tráfego de rede, utilizando um algoritmo definido como "*adaptive flow aggregation*" que é comparado com outras soluções para geração de fluxo de dados, como o *Netflow*.

Diversas ferramentas de software são capazes de fazer a construção de fluxo de dados, tendo como entrada os pacotes de rede. Ferramentas como *Flow-toll*, *flowcat*, *flow-nfilter* e

flow-stat trazem resultados interessantes na detecção de comportamentos suspeitos em uma rede (GRÉGIO et al., 2009). Podemos destacar o software *flow-tools* (“Flow-tools”, 2014), que é um software livre, que armazena os fluxos do *NetFlow* em vários arquivos diferenciados por tempo. Um dos recursos disponíveis no *flow-tool* é a ferramenta para detecção de alguns ataques de negação de serviço (*Denial of Service*). A Figura IV.1 apresenta um gráfico com base em protocolos, gerado pelo software *Netflow*.



.

Figura IV.1 - Gráfico Gerado pelo Netflow

Fonte: (“Flow-tools”, 2014)

IV.5 Síntese dos trabalhos relacionados

Neste capítulo foram apresentados trabalhos relacionados à análise de *datasets* disponíveis para pesquisa, a sua separação baseadas em pacotes e fluxo, os *datasets* que apresentavam marcações de ataques e os que foram utilizados para mineração de dados. Também foram identificadas pesquisas relacionadas à produção de *dataset*, considerando a avaliação de pacotes e fluxo de dados para a identificação de ataques relacionados a DoS.

A Tabela IV.2 apresenta de forma resumida as pesquisas relacionadas, os tipos de dados utilizados, se pacote ou fluxo de dados e se realizavam a marcação. Os trabalhos relacionados buscam identificar os ataques de negação de serviço. Durante a pesquisa foram avaliados alguns pontos a serem destacados, como a utilização e produção de *datasets* para pesquisas, o uso de fluxo de dados e as ferramentas de mineração de dados utilizadas para a detecção de ataques de negação de serviço.

Quanto a utilização do *dataset*, foi identificado que grande parte das pesquisas não contemplam a produção do mesmo e sim a utilização de *datasets* disponíveis para estudo, como por exemplo o KDD Cup 99, Defcon e os *datasets* disponibilizados pelo DARPA e CAIDA.

Considerando que as pesquisas baseadas em fluxos de dados, conforme Sperotto (2010) começaram a surgir em 1990, um volume menor de pesquisas relacionadas a produção e avaliação de *dataset* com base em fluxo de dados está disponível. As pesquisas avaliadas considerando a utilização de fluxo de dados destacam a utilização de dois padrões de análise de fluxo: *IPFIX* e *Netflow*.

Para realizar a identificação de ataques de DoS, seja utilizando pacotes de dados ou fluxo de dados, são empregadas ferramentas de mineração de dados com o objetivo de detectar a intrusão, sejam em bases disponíveis no mercado ou em bases produzidas durante a pesquisa.

Tabela IV.2 - Resumo dos Artigos

Artigo	Datasets	Tipo	Marcação	Dado Real/Sintético
(DOKAS et al., 2002)	KDD Cup 99	Pacote	Não	Sintético
(ESKIN et al., 2002)	KDD Cup 99	Pacote	Não	Sintético
(ZHONG; KHOSHGOFTAAR; SELIYA, 2007)	CAIDA	Pacote	Não	Sintético
(ERTÖZ et al., 2004)	Própria	Fluxo	Não	Real
(SPEROTTO et al., 2010)	Própria	Fluxo	Não	Real
(CORRÊA; PRÔTO; CANSIAN, 2007)	Própria	Fluxo	Não	Sintético

Fonte: Próprio Autor

Capítulo V - Access Collector: Um *workflow* para produção de datasets

O presente trabalho visa contribuir por meio de uma abordagem sistematizada de produção de *datasets* baseados em fluxo de dados rotulados, para a detecção de ataques de negação de serviço. A proposta baseia-se em um *workflow* que consiste em três atividades principais: (i) atividade de geração de tráfego de ataques), (ii) atividade de captura dos pacotes e (iii) atividade de transformação de pacotes em fluxo.

Nesta pesquisa, o *workflow* criado foi definido como *Access Collector*. O *workflow* automatiza a criação de *datasets* baseados em fluxo de dados, contendo tráfego normal e de ataque, ambos marcados. É proposta a rotulação do conjunto de dados baseados em fluxo para detecção de ataques de negação de serviço.

Para a implementação do *workflow* é utilizado o sistema de gerenciamento de *workflow*s Sagitarii, que é um software criado para a composição e execução de *workflow* (OGASAWARA et al., 2010). O sistema de *workflows* Sagitarii possui mecanismos específicos para a execução de atividades internas e externas, sendo capaz de disparar acionamentos de atividades para os nós computacionais.

V.1 Arquitetura do Workflow Access Collector

Nesta seção, descrevemos a arquitetura do *workflow* Access Collector, baseado na execução de atividades para a criação de um *dataset* de fluxos de dados com marcação para análise de ataques de negação de serviço. O *workflow Access Collector* executará as etapas definidas na Figura V.1 de forma a gerar o dataset de fluxos: (1) a geração dos pacotes caracterizados por ataques, (2) captura dos pacotes de comunicação com o servidor, (3) conversão dos pacotes em fluxo, a importação dos dados (criação das relações) de forma a se obter o *dataset*.

O servidor a ser atacado é um servidor instalado em um ambiente de produção, que deve executar o software de captura de pacotes em modo promíscuo, ou seja, qualquer pacote que "chegar" ao servidor será capturado e armazenado. A geração dos ataques será realizada pelos nós computacionais, após o envio dos comandos a serem disparados pelo portal de gerenciamento do Sagitarii, o qual denominaremos simplificadamente de servidor Sagitarii.

Cada atividade a ser executada, conforme apresentado na Figura V.1, é previamente definida e configurada no servidor Sagitarii. Durante a execução, a primeira atividade iniciada é o disparo dos acessos caracterizados por ataque, que são realizados pelos nós computacionais (1). Os acessos legítimos são os acessos realizados pelos usuários internos e externos da instituição, utilizando serviços disponíveis no servidor.

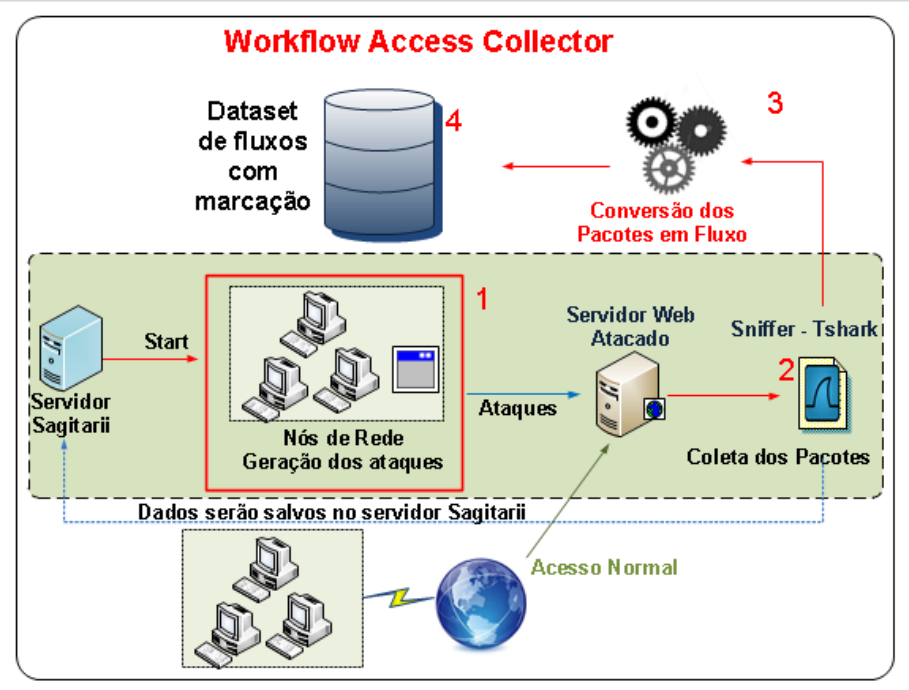


Figura V.1 - Arquitetura do Access Collector – Visão dos Processos

Fonte: Próprio Autor

A atividade de captura dos pacotes (2) consiste na coleta de todos os pacotes de comunicação que são recebidos pelo servidor. Os dados coletados durante a atividade de captura são armazenados em um *dataset* de pacotes para em seguida serem convertidos em fluxos de dados.

Neste cenário, os dados consumidos representam o conjunto de dados referentes aos pacotes de rede que foram capturados e em sequencia serão convertidos em fluxos. Para a realização da conversão são considerados os campos de fluxo de dados definidos pelo protocolo IPFIX, que são IP de origem, porta de origem, IP de destino, porta de destino e protocolo. A próxima sessão apresenta o diagrama de atividades que compõem o experimento.

V.2 Diagrama de Atividades do Access Collector

O diagrama de atividades é uma técnica para descrever um processo lógico, um processo de negócio ou um fluxo de trabalho (FOWLER, 2004). Nele é apresentado a sequencia de atividades a serem realizadas para a construção do *workflow* para a geração do *dataset* para a análise de ataques relacionados à negação de serviço, considerando a detecção com base no comportamento dos pacotes trafegados.

Neste diagrama, o ponto preto indica o início da atividade, as caixas com cantos arredondados representam ações e o circulo preto do final indica o término das atividades. As

setas conectando as caixas de canto arredondados indicam a sequência em que as ações devem ser executadas.

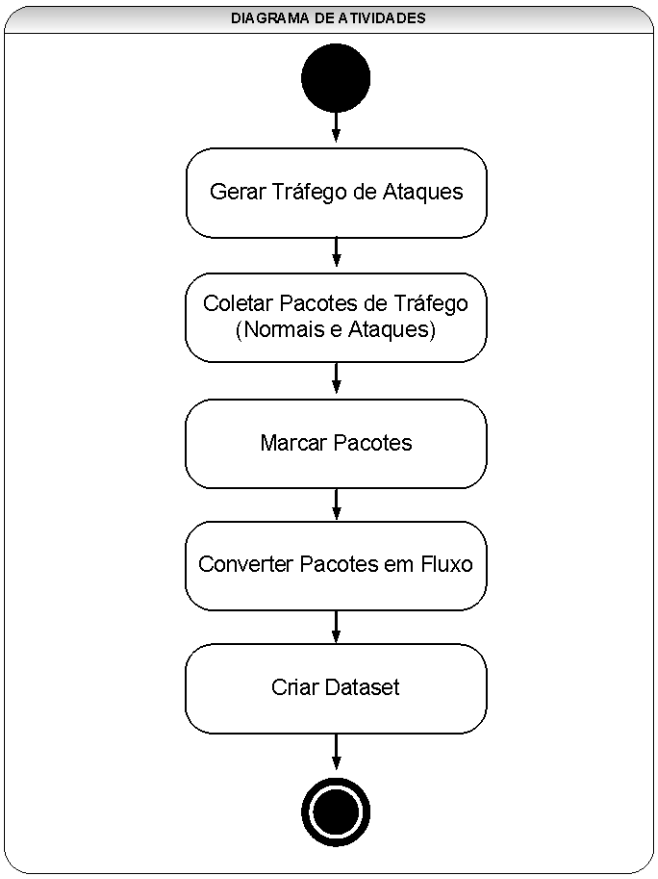


Figura V.2 - Diagrama de Atividades

Fonte: Próprio Autor

O diagrama de atividades representado na Figura V.2 apresenta a sequência de atividades a serem executadas pelo *workflow Access Collector* para a criação do *dataset* para a análise de ataques de negação de serviço, conforme os passos descritos. A instanciação deste processo é descrita na seção de avaliação experimental.

Capítulo VI - Avaliação Experimental

Esta seção descreve a avaliação experimental do *workflow Access Collector*, cujo objetivo é construir um *dataset* para a análise de ataques de negação de serviço com base em fluxo de dados com marcação. Na seção VI.1, é descrito o ambiente composto para a realização do experimento. Na seção VI.2, é descrita a atividade de geração de tráfego. A seção VI.3 apresenta a atividade de captura dos pacotes que constitui o processo de comunicação dos nós que compõe a solução. A seção VI.4 descreve a atividade de marcação dos pacotes. A seção VI.5 apresenta a atividade de conversão de pacotes em fluxo. A seção VI.6 apresenta o *dataset* produzido. E finalmente a seção VI.7 apresenta uma análise do fluxos gerados, assim como os gráficos obtidos como resultados da análise do tráfego.

VI.1 Ambiente

O ambiente utilizado para a realização do experimento foi implementado no laboratório de Pesquisa da Escola de Informática e Computação do Centro Federal de Educação Tecnológica Celso Suckow da Fonseca (CEFET/RJ), sendo composto por um servidor de produção, contendo o Moodle e o portal da Escola de Informática e Computação (EIC) disponíveis para a comunidade de Computação, um servidor para o Sagitarii e os seus respectivos nós computacionais. O servidor com o sistema de gerenciamento de *workflows* Sagitarii, assim como o servidor de produção, possuem a configuração de 8 GB de memória RAM, processador Intel Xeon quad-core e 512 GB de disco rígido. Os nós computacionais são compostos por 4 máquinas de quatro núcleos. Cada nó computacional é responsável pela geração do tráfego de ataque, realizando a simulação de IPs. Os acessos considerados legítimos serão gerados pelo uso natural do servidor, que encontra-se em produção. A Figura VI.1 resume a topologia do ambiente utilizado para a realização do experimento[1].

VI.2 Geração do Tráfego

O processo de geração de tráfego consiste na combinação dos acessos legítimos e de ataque realizados ao servidor web de produção da EIC disponibilizado à comunidade de computação da instituição. Os acessos ao servidor a ser atacado são realizados pela rede interna da instituição e pela internet.

[1] Apesar de todos os wrappers terem sido desenvolvidos e as atividades projetadas e modeladas no Sagitarii, devido a um atraso na disponibilização da versão oficial do Sagitarii para a comunidade, a execução dos experimentos deste capítulo foi feita via script bash, emulando o funcionamento correto do Sagitarii**.**

O tráfego de ataque simula um ataque DDoS, onde vários IPs diferentes, controlados pelo *workflow*, enviam comandos ao servidor de produção com o objetivo de sobrecarregá-lo e causar lentidão ou paralização dos serviços.

A atividade para envio de ataques foi definida no *workflow* como "Enviar_Ataque", onde o *wrapper* denominado "ataque.jar" será executado. O *wrapper* é um programa desenvolvido em Java que executa comandos do aplicativo T50 (T50, 2014) para a geração de tráfego caracterizado pelo ataque de DDoS.

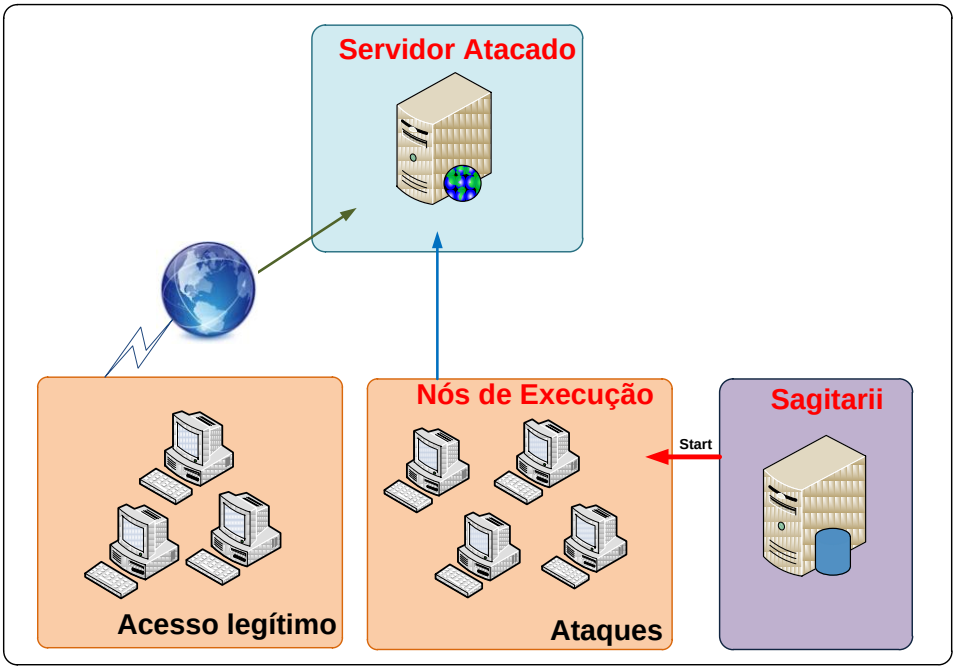


Figura VI.1 - Representação da topologia do ambiente

Fonte: Próprio Autor

Os comandos de acesso ao servidor são realizados por 4 nós computacionais, cada um deles com quatro processadores distintos disparando pacotes para o servidor a ser atacado. Os IPs responsáveis pelos ataques estão na faixa 192.168.5.*.

Os ataques de DDoS simulados são do tipo *SYN Flood*, no qual o atacante utiliza uma das característica do protocolo TCP/IP, realizando uma solicitação do tipo SYN (*synchronize*), aguardando a resposta do servidor que seria um SYN-ACK (*acknowledge*). Como várias solicitações são enviadas ao servidor, o mesmo fica sobrecarregado, gerando lentidão na resposta as solicitações. Em uma atividade normal de comunicação, este processo é definido como aperto de mãos em três etapas (do inglês, *Three-Way Handshake*), no qual o cliente envia ao servidor uma solicitação (SYN), o servidor confirma a requisição mandando um SYN-ACK (*acknowledge*) e o cliente responde com um ACK (*acknowledge*), estabelecendo assim a comunicação, conforme apresentado na Figura VI.2 (TANENBAUM, 2003). A falta da resposta

do cliente com o SYN-ACK, faz com que o servidor aguarde por um certo tempo, alocando recursos, sem terminar o processo de conexão.

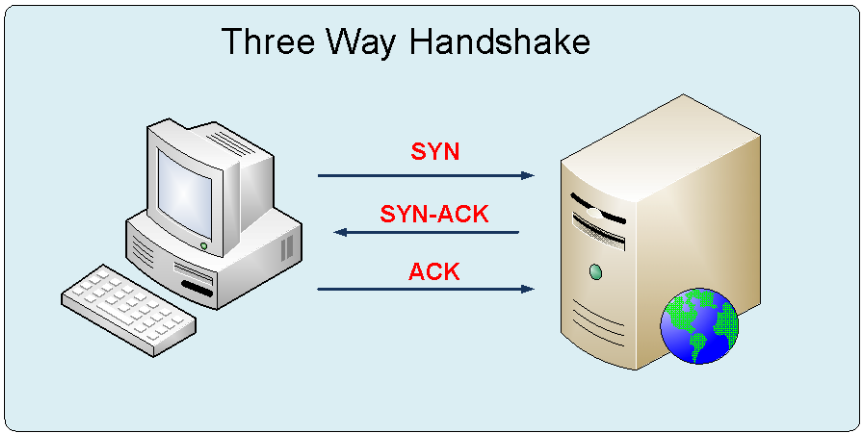


Figura VI.2 - *Three Way Handshake*

Fonte: Próprio Autor

Os comandos enviados são disparados para o servidor a ser atacado, considerando um envio sucessivo de pacotes por um tempo determinado tempo. No *workflow* Sagitarii, a entrada para esta atividade é um arquivo CSV, que contém os parâmetros referentes ao ataque, tais como o endereço de IP do servidor a ser atacado e o tempo de execução do envio dos pacotes.

VI.3 Captura do Tráfego

A atividade referente à captura dos dados é definida no *workflow* Sagitarii como "Executar_captura". Tal atividade é responsável por enviar o comando para iniciar o processo de captura dos pacotes. A coleta ocorre no próprio servidor que está sendo atacado.

Para a realização da atividade de captura, é executado o *wrapper* "sniffer.jar", que executa o aplicativo *tshark,* que é o responsável pela captura dos pacotes. Toda comunicação realizada pelo servidor a ser atacado é coletada. São armazenados os pacotes envolvidos na comunicação, sejam eles de acesso legítimos realizado pelos alunos ou pelos acessos simulados de ataque DDoS.

Para definir os atributos dos pacotes a serem capturados, o arquivo CVS de input de dados no Sagitarii foi definido, conforme apresentado na Tabela VI.1.

Tabela VI.1 - Dados dos pacotes capturados

Id_Pac	Tempo	Ip de Origem	IP de Destino	Porta Origem	Tipo TCP/ UDP	Porta Destino	Tipo TCP/ UDP	Protocolo	Tamanho do pacote	Info do pacote

Fonte: Próprio Autor

Os pacotes de dados foram capturados por um período de 5 dias, gerando como resultado um dataset definido com "data_packets", composto por 23GB de pacotes de dados, considerando o tempo no formato UTC (*Coordinated Universal Time*), de forma a facilitar a manipulação e entendimento dos dados gerados.

Com o objetivo de parar o processo de captura dos pacotes, foi criado no *workflow* a atividade "Parar_Sniffer", onde tal parâmetro pode ser configurado no *workflow Control Access*, previamente, considerando o atributo horas ou dias.

VI.4 Processo de Marcação dos Fluxos

Para a marcação de pacotes a atividade "Executar_rotulo" foi criada no *workflow Access Colletor*, onde com o uso da operação algébrica *Map*, as tuplas de entrada da tabela "data_packets" são lidas, gerando uma nova tupla, com o atributo referente a "acesso normal" ou "ataque".

Para a marcação de cada pacote coletado, foi considerado o atributo IP de origem, onde são identificados os IPs das máquinas que foram responsáveis pelo envio dos ataques. Todos os pacotes cujo IP de origem conhecida com um dos IPs dos nós computacionais, foram marcados como ataques e os demais IPs de origem como acessos legítimos.

O *workflow Access Collector*, após a execução da operação algébrica *Map,* utilizando como *input* a tabela "data_packet" retornou como resultado uma nova tabela definida como "rotulo_packets" onde todos os atributos definidos na Tabela VI.1 foram mantidos, acrescentando-se o atributo "Tipo", conforme apresentado na Tabela VI.2.

Tabela VI.2 - Atributo dos Fluxos

Id_Pac	Tempo	Ip de Origem	IP de Destino	Porta Origem	Tipo TCP/ UDP	Porta Destino	Tipo TCP/ UDP	Protocolo	Tamanho do pacote	Tipo	Info do pacote

Fonte: Próprio Autor

VI.5 O Processo de conversão em Fluxo de Dados

Para executar a atividade de conversão de pacotes de dados em fluxo, foi utilizado o padrão IPFIX desenvolvido pelo IETF. Para armazenamento das informações, considerando o fluxo de dados, foram considerados os atributos: IP de origem, porta de origem, Ip de destino, porta de destino, protocolo.

Considerando o modelo de referencia OSI (*Open Systems Interconnection*), o início de uma comunicação é realizado pela solicitação de uma sessão, definido pelo processo *tree-way handshake*, onde os dois equipamentos que desejam estabelecer a comunicação realizam as medidas de controles necessárias para se estabelecer uma comunicação. Realizado este processo, uma conexão é estabelecida entre os equipamentos que iniciam a troca de informações, que consiste no envio dos pacotes (TANENBAUM, 2003). Em determinadas situações, como no caso dos ataques simulados, são registrados apenas registros de sessões e não de conexões.

Para a realização desta atividade, foi criado no *workflow Access Collector* a atividade "Executar_fluxo", tendo como parâmetros de entrada, ou seja, dados a serem consumidos a tabela "data_packets", que por meio da operação algébrica *Reduce* implementada no *workflow* que reduz as conexões/sessões de pacotes em fluxo de dados.

VI.6 O Dataset Criado

Após a execução das atividades pré-definidas no *workflow*, o dataset com os fluxos de dados marcados é produzido, propiciando a redução de recursos computacionais para a análise de um grande volume de dados.

Figura VI.3 apresenta as atividades definidas no *workflow Access Collector*, assim como os executores, os arquivos de input de informações e as tabelas de dados geradas como resultado final de cada atividade.

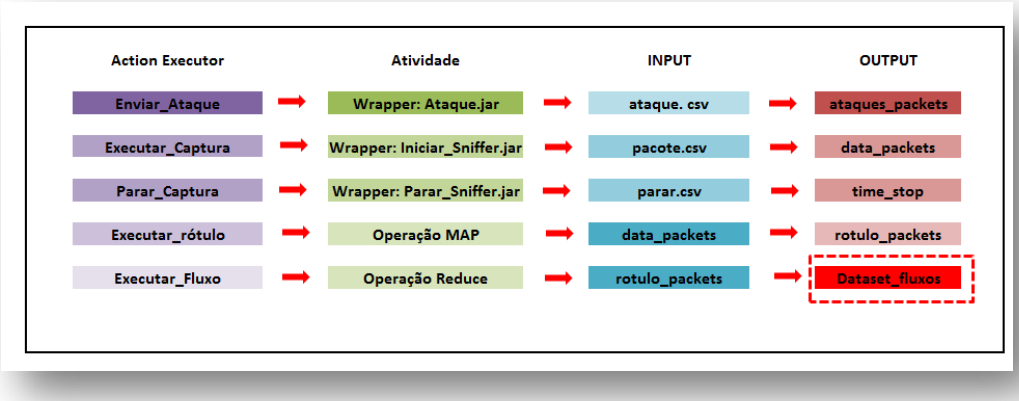


Figura VI.3 - Visão geral do Access Collector

Fonte: Próprio Autor

Uma das contribuições deste estudo é a produção de datasets baseada em fluxo com marcação, para a análise de ataques de DoS, proporcionando a facilidade de geração de *datasets* de forma automática, para prover o auxílio no desenvolvimento e treinamento de IDS capazes de identificar os ataques de negação de serviço.

VI.7 Analise Preliminar por meio de fluxo de dados

A medida que a quantidade de serviços oferecidos em uma rede aumenta, a demanda por desempenho também aumenta, dando origem às redes de alta velocidade. Face a esse cenário, mecanismos de monitoramento, com o objetivo de mitigar os riscos, devem acompanhar a mesma linha de crescimento.

A produção dos fluxos gerados pelo *workflow Access Control* visa a análise do tráfego da rede de forma consolidada, com o objetivo de identificar atividades maliciosas. Considerando que o padrão dos fluxos definidos pelo IPFIX são representações consolidadas das sessões/conexões de uma rede, a análise baseia-se na identificação de fluxos com padrões de comportamento anômalos. Os fluxos marcados como padrões de tráfego legítimo e tráfego de ataques, podem ser utilizados para treinamento de IDS, destacando a vantagem da análise dos fluxos não considerarem a análise individualizada dos conteúdos dos pacotes, diminuindo consideravelmente o volume de dados a serem analisados, agregando a característica de escalabilidade e possibilitando, assim, a análise de redes de alto desempenho.

A medida que os fluxos gerados em uma rede são capturados e arquivados em um *dataset*, passam a constituir um fonte de informação valiosa, pois possibilitam observar detalhes de cada sessão de comunicação, tornando-se uma ferramenta de monitoração e mitigação de riscos para a área de segurança da informação.

O processo inicial de coleta dos dados consiste na captura de todos os pacotes envolvidos na comunicação de dados. A Figura VI.4 mostra a coleta de pacotes que representam um acesso comum do tipo HTTP, onde o cliente envia a requisição (SYN), o servidor responde com SYN-ACK e finalmente o cliente envia o SYN para o estabelecimento da comunicação.



Figura VI.4 - Identificação do tráfego legítimo

Fonte: Software Wireshark

A Figura VI.5 e VI.6 apresentam o ataque de SYN Flood, onde o cliente envia continuamente pacote de SYN para o servidor utilizando o protocolo HTTP e a porta 80.



Figura VI.5 - Identificação do Ataque SYN Flood

Fonte: Software Wireshark

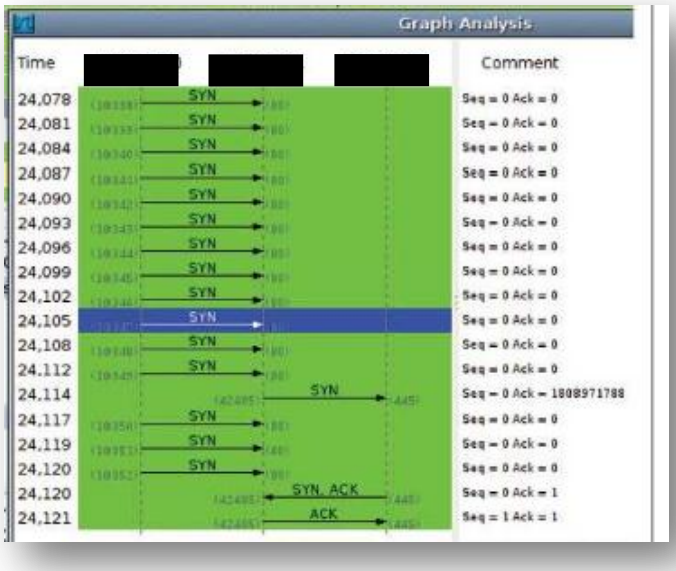


Figura VI.6 - Análise do Fluxo

Fonte: software wireshark

Com o uso de técnicas de mineração de dados, a estratificação das informações pode ser implementada com o objetivo de avaliar os fluxos e apresentar informações consolidadas em relação ao tráfego monitorado. Tais informações podem ser apresentadas de forma gráfica, possibilitando uma melhor identificação dos cenários.

Com a utilização dos fluxos de dados, a quantidade de tuplas existente na tabela "data_packets" são reduzidas, com a utilização da operação algébrica *Reduce* e os conceitos pré-definidos pelo IPFIX. As requisições são consolidadas, identificando o perfil do ataque.

A Figura VI.7 representa o fluxo geral de dados capturados do servidor durante o período de três horas e meia aproximadamente. O eixo x representa os minutos referentes as coletas e o eixo y o volume de fluxos. Foi identificado que os picos maiores de acesso ocorreram nos momentos em que os envios de comando de ataques foram disparados ao servidor, validando a identificação dos acessos ditos.

Durante o processo de análise foram identificados acessos legítimos, que realizavam envio de grande volume de dados, como foi o caso de sessões de SSH estabelecidas pelos clientes, que indicavam também um tráfego anormal. Casos de tráfego anômalo, caracterizado por atividades legítimas são identificados como falso-positivos. Tais acessos são sinalizados.

Os comandos de ataque direcionados ao servidor, durante um período de 10 minutos são obervados na Figura VI.8, que foi gerada utilizando o coneito de média móvel, que considera a análise de dados em um determinado período de tempo. Mediante a análise dos gráficos gerados, foi possível validar que o processo de coleta dos dados, conversão em fluxo e análise, apresentam as atividades simuladas durante o experimento.

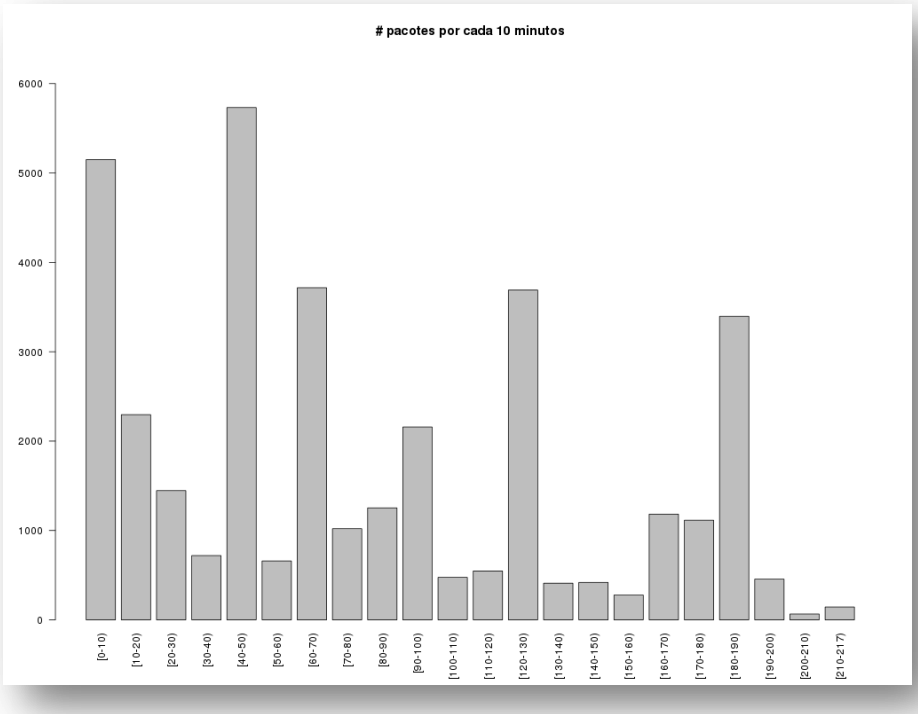


Figura VI.7 - Fluxo geral do servidor

Fonte: Próprio Autor

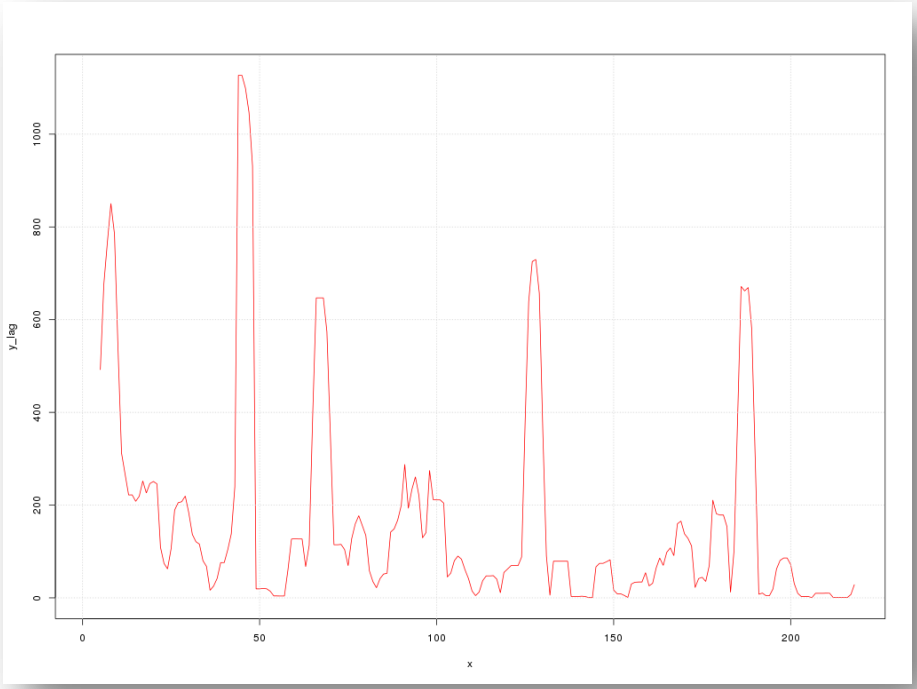


Figura VI.8 - Ataques por um período de 10 minutos

Fonte: Próprio Autor

Como resultado, é apresentado um *dataset* caracterizado por fluxo de dados, onde pode ser identificado os acessos normais e os acessos de ataques, com as respectivas marcações.

Capítulo VII - Considerações Finais

Este capítulo apresenta as considerações finais da pesquisa, contemplando a seção VII.1, referente as conclusões obtidas neste estudo e a seção VII.2, contemplando as publicações realizadas e os trabalhos futuros.

VII.1 Conclusão

Com o objetivo de realizar a detecção e análise de invasão, mecanismos de defesa são implementados. O processo de análise das redes que consiste na coleta, análise e detecção de anomalias tem sido um desafio constante, principalmente devido à quantidade de dispositivos conectados, à variedade de protocolos e serviços, ao volume elevado de tráfego, bem como às características intrínsecas do tráfego padrão.

A presente pesquisa propôs uma forma de criação de um dataset baseado em fluxo de dados com marcação, considerando o modelo de armazenamento utilizando *datasets* relacionais, de forma a ser implementado com o uso de um *workflow* que utiliza parâmetros algébricos. O resultado obtido foi um dataset sintético, baseado nos fluxos de dados.

O presente estudo visa contribuir com a proposta de geração de *dataset* de fluxo de dados de forma automatizada, com o objetivo de realizar a identificação de ataques de negação de serviço, possibilitando a sua implementação em ambientes variados. O *workflow Access Control* apresenta como principais características a adaptabilidade a qualquer tipo de ambiente, detecção de anomalias baseadas em comportamento, escalabilidade e automatização de processos.

Para a criação do *dataset* foi utilizado o conceito de *workflow*, onde foi utilizado o sistema de gerenciamento de *workflows* Sagitarii, que implementa a realização das atividades internas de forma algébrica, considerando os seus operadores próprios. Apresenta-se como uma abordagem de destaque para a modelagem de experimentos baseados em simulações, sendo necessário para tanto, determinar as características de seus componentes, onde todas as atividades são previamente definidas, para que as atividades possam ser executadas de forma automática e com pouca intervenção.

A pesquisa apresenta três contribuições de destaque: i) A geração do tráfego com acessos normais e ataques, que podem ser repetidos em laboratório, considerado os parâmetros desejados. ii) Em segundo lugar, podemos citar a forma de conversão dos pacotes em fluxo, com as marcações dos fluxos de ataques e fluxos normais. iii) A criação de um *datasets* para treinamento de IDS, que pode ser gerado conforme a necessidade.

A maior vantagem proporcionada pela pesquisa é a execução de todos estes passos de forma integrada e automática, podendo ser reproduzida em qualquer ambiente com a utilização do conceito de *workflow,* garantindo a reprodução sistemática do experimento.

VII.2 Publicações e Trabalhos Futuros

Este trabalho produziu como resultado dois artigos que foram publicados no Simpósio de Excelência em Gestão e Tecnologia (SEGET), realizado em novembro de 2014. O estudo intitulado "Segurança Cibernética e Política Públicas no Brasil" contribuiu como motivador para a pesquisa relacionada à segurança da informação e as suas forma de proteção, onde foi realizada uma análise geral das ações e políticas públicas que estão sendo estudadas e implementadas de forma a proteger as informações dos cidadãos.

O segundo artigo definido como "Uso de *Workflows* para Apoiar a Elaboração de Técnicas de Predição de Invasão de Sistemas" expõe a metodologia para o uso *workflows* de forma apoiar a elaboração de técnicas de predição de invasão de sistemas (FARIAS et al., 2014).

Finalmente, pretende-se consolidar a pesquisa, realizar e explorar um pouco mais os cenários de ataques de forma a serem gerados novos *datasets,* considerando o conceito *workflows* para a sua implementação, de forma a disponibilizarmos para a comunidade. Poderão ser considerados nos estudos futuros a utilização de novos cenários para geração de *datasets*, considerando formas de ataques diversas, com reprodução de datasets com características distintas.

Referências Bibliográficas

ABREU, C. M. DE; OGASAWARA, E. **Sistema de Workflows Sagitarii**. Disponível em: <https://sourceforge.net/p/gpca/wiki/Sagitarii/>. Acesso em: 31 mar. 2015.

BRANQUINHO, MARCELO. SEIDL, JAN; AZEVEDO, JARCY JUNIOR; MORAES, LEONARDO; BRANQUINHO, THIAGO. **Segurança de Automação Industrial e SCADA**. 1ª. ed. Rio de Janeiro: Campus, 2014.

CAIDA. Disponível em: <http://www.caida.org/data/passive/ddos-20070804_dataset.xml>. Acesso em: 21 mar. 2014.

CALLAHAN, S. P.; FREIRA, JULIANA, SANTOS, EMANUELE; SCHEIDEGGER, CARLOS; SILVA, CALUDIO T. **VisTrails: visualization meets data management.** Proceedings of the 2006. ACM SIGMOD international conference on Management of data. **Anais**: ACM, 2006

CANDIDO, A. J.; CANSIAM, A. M.; SAUDE, A. M. **Uso de Redes Neurais para Detecção de Anomalias em Fluxo de dados**, VI Simpósio de Guerra Eletronica, São José dos Campos, 2005.

CAVALCANTI, K. **Internet para todos: Cuide da sua segurança com segurança**. 2. ed. Olinda - Brasil: Livro Rápido, 2014.

CERT.BR. **Centro de Estudos, Resposta e Tratamento de Incidentes de Segurança no Brasil**. Disponível em: <http://www.cert.br/>. Acesso em: 20 ago. 2014.

CISCO. **Cisco IOS NetFlow - A Technical Overview**. Disponível em: <http://www.cisco.com/c/en/us/products/collateral/ios-nx-os-software/ios-netflow/prod_white_paper0900aecd80406232.html>. Acesso em: 20 ago. 2014.

CORRÊA, J. L.; PRÔTO, A.; CANSIAN, A. M. **Modelos de armazenamento de fluxos de rede para análises de tráfego e de segurança**, 2007.

CSWFPROBE. **Tool that collect network traffic data**. Org. Disponível em: <http://www.opencsw.org/package/fprobe/>. Acesso em: 20 ago. 2014.

DARPA. **DARPA Intrusion Detection Data Sets**. Disponível em: http://www.ll.mit.edu/ideval/data. Acesso em: 20 ago. 2014.

DEELMAN, E.; MEHTA, D.; SINGH, G.; SU, H.; VAHI, V. Pegasus: Mapping Large-Scale Workflows to Distributed Resources. In: **Workflows for e-Science**. USA: Springer, 2007. p. 376–394.

DEELMAN, E.; GANNON, D.; GANNON, M.; TAYLOR, I. Workflows and e-Science: An overview of workflow system features and capabilities. **Future Generation Computer Systems**, v. 25, n. 5, pp. 528–540, maio 2009.

DEF CON Hacking Conference. Conferencia Annual de Hacking. Las Vegas. Disponível em: <https://www.defcon.org/>. Acesso em: 9 maio. 2015.

DE OLIVEIRA, D.; OGASAWARA, E. BAIÃO, F.; MATOSO, M. **SciCumulus: A Lightweight Cloud Middleware to Explore Many Task Computing Paradigm in Scientific Workflows.** 2010 IEEE 3rd International Conference on Cloud Computing (CLOUD). In: Anais IEEE 3RD INTERNATIONAL CONFERENCE ON CLOUD COMPUTING (CLOUD). jul. 2010

DOKAS, P.; ERTOZ, L.; KUMAR, V.; LAZAREVIC, A. SRIVASTAVA; TAN, P. N. **Data Mining for Network Intrusion Detection.** In Proceedings of the NSF Workshop on Next Generation Data Mining, 2002.

ERTÖZ, L.; EILERTSON, E.; LAZAREVIC, A.; TAN P. N.; KUMAR, V.; SRIVASTAVA, J.; DOKAS, P. et al. **MINDS - Minnesota Intrusion Detection System**. In: MIT PRESS. 2004

ESKIN, E.; ARNOLD, A.; PRERAU, M.; PORTNOY, L.; STOLFO, S. **A Geometric Framework for Unsupervised Anomaly Detection: Detecting Intrusions in Unlabeled Data**. In: IN APPLICATIONS OF DATA MINING IN COMPUTER SECURITY. 2002

FARIAS, V.; SOUZA, R.; ROCHA, V.; SOUZA, C.; BEZERRA, E.; MACHADO, R.; OGASAWARA, R. **Uso de workflow científico para apoiar a elaboração de técnicas de predição invasão de sistemas**. In: SIMPÓSIO DE EXCELÊNCIA EM GESTÃO E TECNOLOGIA (SEGET). 2014

Flow-tools. Software Livre. Disponível em: <https://code.google.com/p/flow-tools/>. Acesso em: 21 mar. 2014.

FOWLER, M. **UML Distilled A brief guide to the standard Object Modeling Language**. 3rd. ed. Boston: Person Education, 2004.

GRÉGIO, A. R. A.; FILHO, B. P.; MONTES, A.; SANTOS, R. **Técnicas de Visualização de Dados aplicadas à Segurança da Informação**. In: IX SIMPÓSIO BRASILEIRO EM SEGURANÇA DA INFORMAÇÃO E DE SISTEMAS COMPUTACIONAIS. 2009

GUIMARAES, A. G.; OLIVERIA, A.; LINS, R. D.; OLIVEIRA, R. **Segurança em Redes Privadas Virtuais - VPNs**. Rio de Janeiro: Brasport, 2006.

HAN, J.; KAMBER, M. **Data Mining: Concepts and Techniques, Third Edition**. 3 edition ed. Burlington, MA: Morgan Kaufmann, 2011.

HEBERLEIN, L. T.; DIAS, G. V.; LEVITT, K. N.; WOOD, J.; WOLBER, D. **A network security monitor.** Oakland, CA: IEEE, 7 maio 1990

HOFSTEDE, R; BARTOS, V.; SPEROTTO, A.; PRAS, A. **Towards Real-Time Intrusion Detection for NetFlow and IPFIX**. In: NETWORK AND SERVICE MANAGEMENT (CNSM), 2013 9TH INTERNATIONAL CONFERENCE, 2013

HULL, D.; WOLSTENCROFT, K.; STEVENS, R.; POCOCK , M.; LI, P.; OINN, T. **Taverna: a tool for building and running workflows of services.** Nucleic acids research, v. 34, n. suppl 2, p. W729–W732, 2006.

IETF.ORG. **RFC 3917 - Requirements for IP Flow Information Export (IPFIX)**. Disponível em: <https://www.ietf.org/rfc/rfc3917.txt>. Acesso em: 6 maio. 2015.

IETF.ORG. **RFC 5101 - Specification of the IP Flow Information Export (IPFIX) Protocol for the Exchange of IP Traffic Flow Information**. Disponível em: <https://tools.ietf.org/html/rfc5101>. Acesso em: 6 maio. 2015.

JALALI, S.; WOHLIN, C. **Systematic literature studies: Database searches vs. backward snowballing.** ACM-IEEE International Symposium on Empirical Software Engineering and Measurement (ESEM). set. 2012

JORGE, H. V. N.; WENDT, E. **Crimes Cibernéticos - Ameaças e Procedimentos de Investigação**. 2ª Ed. 2013 ed. Brasport, 2013.

KDD. **KDD Cup 1999 Data**. Data Mining and Knowledge Discovery competition. Disponível em: http://kdd.ics.uci.edu//databases/kddcup99/kddcup99.html . Acesso em: 20 ago. 2014.

KRUEGEL, C.; VALEUR, F.; VIGNA, G. **Intrusion Detection and Correlation: Challenges and Solutions**. 1. ed. Santa Barbara, California: Springer US, 2005. v. 14

MANDARINO JUNIOR, R.; CANONGIA, C. **O Livro Verde: Segurança Cibernética no Brasil**. Disponível em: <http://dsic.planalto.gov.br/documentos/publicacoes/1_Livro_Verde_SEG_CIBER.pdf>. Acesso em: 16 jun. 2015.

MANOEL, S. DA S. **Governança de Segurança da Informação: Como criar oportunidades para o seu negócio**. 2014. ed. Brasport, 2014.

MARTINHO, W.; OGASAWARA, E.; OLIVEIRA, O.; CHIRIGATI, F.; SANTOS, I.; TRAVASSOS, G. H. T.; MATTOSO, M. **A Conception Process for Abstract Workflows: An Example on Deep Water Oil Exploitation Domain. Anais** EEE International Conference on e-Science. 2009

MATTOSO, M.; WERNER, C.; TRAVASSOS, G.; BRAGANHOLO, V.; MURTA, L. **Gerenciando Experimentos Científicos em Larga Escala**. In: SEMISH - CSBC. Belém, Pará - Brasil. 2008

NAÍM, M. **O Fim do Poder**. 1ª. ed. LEYA BRASIL, 2013.

NAKAMURA, E. T.; GEUS, P. L. DE. **Segurança de Redes em Ambientes Cooperativos**. São Paulo: Novatec, 2007.

OGASAWARA, E.; MATOSO, M.; WERNER, C.; OLIVEIRA, D.; CRUZ, S.; BRAGANHOLO, V. **Towards supporting the life cycle of large scale scientific experiments**. In: INTERNATIONAL JOURNAL OF BUSINESS PROCESS INTEGRATION AND MANAGEMENT 01/2010; 5:79-92. 2010

OGASAWARA, E.; DIAS, J.; SILVA, V.; CHIRIGATI, F.; OLIVEIRA, D.; PORTO, F.; VALDURIEZ, P. MATTOSO, M. **Chiron: a parallel engine for algebraic scientific workflows.** Concurrency and Computation: Practice and Experience, v. 25, n. 16, p. 2327–2341, 1 nov. 2013.

OGASAWARA, E.; DIAS, J.; SILVA, V.; CHIRIGATI, F.; OLIVEIRA, D.; PORTO, F.; VALDURIEZ, P. MATTOSO, M. **An Algebraic Approach for Data-Centric Scientific Workflows**. PVLDB, v. 4, n. 12, p. 1328–1339, 2011.

POSTGRESQL. **PostgreSQL**. Open source Object-relational Database System. The PostgreSQL Global Development Group. Disponível em: <http://www.postgresql.org/>. Acesso em: 8 abr. 2015.

QIN, J.; FAHRINGER, T.; PLLANA, S. **UML based Grid Workflow Modeling under ASKALON.** Distributed and Parallel Systems.2007

RAINER, R. K.; CEGIELSKI, C. **Introdução a Sistemas de Informação**. Elsevier, 2012. v. 3 Edição

SPEROTTO, A.; SCHAFFRATH, G.; SADRE, R.; MORARIU, C.; PRAS, A.; STILLER, B.; **An Overview of IP Flow-based Intrusion Detection**, 2010.

STALLINGS, W.; BROWN, L. **Segurança de Computadores - Princípios e Práticas**. ELSEVIER, 2014.

STANIFORD-CHEN, S.; CHEUNG, S.; CRAWFORD, R.; DILGER, M.; FRANK, J.; HOAGLAND, J.; LEVITT, K.; WEE, C.; YIP, R.; ZERKLE, D. **GrIDS - A Graph-Based Intrusion Detection System for Large Networks**. In: IN PROCEEDINGS OF THE 19TH NATIONAL INFORMATION SYSTEMS SECURITY CONFERENCE. 1996

SUBBULAKSHMI, T.; SHALINIE, S. M.; GANAPATHISUBRAMANIAN, V.; BALAKRISHNAN, K.; ANANDKUMAR, D.; KANNATHAL, K. **Detection of DDoS attacks using Enhanced Support Vector Machines with real time generated dataset**. In: ADVANCED COMPUTING (ICOAC), THIRD INTERNATIONAL CONFERENCE ON 14-16 DEC. 2011. IEEE, 2011. Disponível em: <http://ieeexplore.ieee.org/stamp/stamp.jsp?tp=&arnumber=6165212&isnumber=6165149>

T50. **T50**. Open Source Software. SourceForge community. Disponível em: http://t50.sourceforge.net/, Acesso em: 15 fev. 2014.

TANENBAUM, A. S. **Rede de Computadores**. 4. ed. Editora Campus, 2003.

WANG, J.; CRAWL, D.; ALTINTAS, I. **Kepler + Hadoop: A General Architecture Facilitating Data-intensive Applications in Scientific Workflow Systems.** Proceedings of the 4th Workshop on Workflows in Support of Large-Scale Science. **Anais**: WORKS '09.New York, NY, USA: ACM, 2009. Disponível em: <http://doi.acm.org/10.1145/1645164.1645176>. Acesso em: 22 fev. 2015

WFMC, I. **Binding, WfMC Standards**. Disponível em: <http://www.wfmc.org>. Acesso em: 27 fev. 2015.

ZHONG, S.; KHOSHGOFTAAR, T.; SELIYA, N. **Clustering-based Network Intrusion Detection**. In: INTERNATIONAL JOURNAL OF RELIABILITY, QUALITY AND SAFETY ENGINEERING. 2007