UNIVERSIDADE DA BEIRA INTERIOR
Engenharia

Analysis of Network Attacks and Security Events using Modern Data Visualization Techniques

Paulo Macedo Pereira

Dissertação para obtenção do Grau de Mestre em

Engenharia Informática

(2º ciclo de estudos)

Orientador: Prof. Doutor Pedro R. M. Inácio

Covilhã, 1 de Outubro de 2015

Dedicatory

To my parents, my sister and my brother in law, for all the help, patience and encouragement throughout my career so far.

Acknowledgments

During my academic path, I had the help and support from many people, either directly or indirectly. It is due to them that I got to this stage of my life.

First of all, I want to thank my parents, Deolinda Macedo and Domingos Pereira, who provided me with an education and a future. They have always supported me and comprised my biggest incentive to write this dissertation. I am also grateful to my sister, Natalia Pereira and my brother in law, Sérgio Marques, who, similarly to my parents, always supported me in every aspect of my academic path, and helped me to keep the motivation to successfully end the dissertation.

Then, I want to acknowledge my supervisor, Professor Doutor Pedro R. M. Inácio, for having accepting me in this masters program, as well as for all the help and time spent in guiding me through the course of the masters program.

Last, but not least, I want to aknowledge the friends that helped, encouraged and motivated me the most, allowing me to overcome this challenge. João Catarino, João Ferreira, Patricia D'amil and Gonçalo Paiva: thank you very much.

Resumo

As técnicas de visualização de dados contêm recursos vitais em várias áreas, desde a pesquisa até há área profissional. Representações eficazes estão frequentemente a contribuir para a compreensão do quadro geral, a partir de um grande volume de dados, às vezes permite novas descobertas ou para uma síntese eficiente. Devido há grande quantidade de dados que os computadores lidam nos dias de hoje, muitas técnicas modernas de visualização de dados foram desenvolvidas para lidar com os grandes conjuntos de dados, permitindo perceber características únicas. Na era da informação, os computadores (e os seus utilizadores) e as redes estão entre as maiores fontes de dados, embora eles também sejam utilizados no seu processamento e armazenamento. Muitos sistemas de monitorização de rede e dispositivos de segurança fazem uso de técnicas de visualização de dados tradicionais para reportar funcionalidades ou para fornecer informações profissionais sobre o estado dos dados.

O âmbito deste trabalho insere-se no cruzamento dos campos, das técnicas de segurança de rede e de visualização dos dados. Os objectivos são estudar abordagens modernas para representar os dados, que podem actualmente ser utilizados em outras áreas, e aplicar uma dessas abordagens na visualização de tráfego de rede e ataques. Avaliar a sua utilidade das visualizações também era um objectivo, juntamente com a constituição de um grande conjunto de representações para várias classes de tráfego e ataques de rede clássicos.

A técnica conhecida como *Circos*, amplamente utilizada para representações genéticas, foi aplicada para alcançar os objectivos deste programa de mestrado. Muitas representações para pelo menos 18 conjuntos de tráfego diferentes foram produzidas ao longo deste trabalho, com muitas analisadas detalhadamente nesta dissertação. Esses conjuntos, contendo tráfego gerados pelas aplicações contemporâneas e ataques clássicos de rede ou actividades de sondagem, foram seleccionados a partir de dois conjuntos de dados. De forma a produzir o *Circos*, um conjunto mínimo de características de tráfego foram identificadas, e foram implementados vários scripts para automatizar o processo. Para a parte final deste trabalho, uma experiência baseada na comparação (humana) entre nove conjuntos conhecidos e nove desconhecidos foram criados. Para demonstrar que as representações foram úteis para identificar as classes de tráfegos ou ataques. Durante a experiência, foi possível identificar correctamente oito, dos nove conjuntos (um dos ataques foi incorrectamente classificado como tráfego *Hypertext Transfer Protocol (HTTP)*), comprovando a utilidade desta técnica nesta área.

Palavras-chave

Ataques Informáticos de Rede, Circos, Classes de Tráfego, Monitorização de Tráfego de Rede, Representação Gráfica dos Dados, Técnicas de Visualização de Dados Modernas

Resumo alargado

Introdução

Esta secção é composta por um resumo alargado sobre o trabalho realizado nesta dissertação. Contém as ideias principais enumeradas ao longo da dissertação. Inicialmente, começa-se por efetuar a descrição do problema abordado e quais os objetivos específicos, assim como são resumidamente descritas e mencionadas as principais contribuições resultantes do trabalho. De seguida, é apresentado um breve resumo de todos os capítulos da dissertação incluindo alguns detalhes mais importantes como a menção a trabalho relacionado na área e sobre outras técnicas de visualização, assim como os conjuntos de dados utilizados para o posterior desenvolvimento dos gráficos. O método utilizado para análise dos conjuntos e as tecnologias utilizadas ao longo do projeto também são aqui descritos, bem como uma sucinta análise dos resultados obtidos na fase final do trabalho de mestrado. Por último, são apresentadas as principais conclusões.

Enquadramento, Descrição do Problema e Objetivos

A área da informática é uma área do conhecimento muito vasta. A tecnologia associada a esta área está em constante evolução, e afeta a vida humana muito significativamente. A proliferação da Internet of Things (IoT), por exemplo, acabará por levar a uma grande conectividade pelo mundo dos dispositivos, capazes de interagir entre si e os seres humanos através da Internet. Essa tecnologia tem o potencial de tornar a vida humana melhor, mas também agravar os problemas de segurança existentes ou gerar novos.

Por causa da enorme utilização da Internet, muitas equipas têm de lidar com incidentes de segurança diariamente [Lou15], que vão desde ameaças amadoras nacionais até ataques estruturados internacionais. Lidar com incidentes de segurança é difícil por muitas razões, mas principalmente porque alguns ataques são projetados ou executados para serem furtivos e difíceis de correlacionar. Por exemplo, um único ataque pode parecer uma série de incidentes isolados. Normalmente, os peritos de segurança recorrem a ferramentas para identificar visualmente ameaças ou fazer correlações, mas estes sistemas não são perfeitos. Muitos ataques são realizados através da rede e podem atingir o objetivo através de diferentes caminhos. Como tal, as técnicas de visualização de eventos de rede ou classes de tráfego compreendem um subconjunto crítico da área acima mencionada, em que esta dissertação se foca. A visualização de tráfego é importante para muitos sistemas de monitorização (não só os relacionados com a

segurança), uma vez que proporcionam formas mais rápidas para entender o comportamento da rede, muito útil para administradores de rede e de sistemas. É comum que os *Network based Intrusion Detection System (NIDS)*, *firewalls* ou dispositivos de segurança criarem gráficos (por exemplo, gráficos de linhas ou de barras) para métricas como a largura de banda, número de ligações por unidade de tempo, as estatísticas sobre pacotes descartados, entre outros, nos relatórios que produzem após os eventos de rede ou segurança. A visualização dos dados está-se a tornar cada vez mais importante na monitorização e segurança das redes e sistemas, porque as redes estão mais complexas, devido ao aumento de dispositivos ligados e ao aumento do processamento de recursos. Apesar das técnicas de visualização tradicionais, tais como os gráficos clássicos de barras ou linhas, serem adequadas para muitos propósitos nesta área, vale a pena fazer o levantamento das técnicas de visualização, adaptá-las e verificar a sua utilidade para este propósito. Para alcançar os objetivos descritos, o trabalho de investigação desta dissertação foi dividido nas seguintes fases:

1. Revisão do estado da arte, em termos de ferramentas e técnicas utilizadas para a visualização de dados, assim como em termos de linguagens de programação para a potencial implementação de *scripts*;

2. Identificação de técnicas de visualização de dados e ferramentas associadas, assim como um estudo aprofundado da técnica seleccionada, e preparação de várias experiências preliminares;

3. Focada na selecção dos dados a serem utilizados e na escolha das características que podem ser utilizadas para produzir as visualizações;

4. Desenvolvimento do(s) *script*(s) para produzir as visualizações sobre os dados;

5. Realização de experiências para avaliar a capacidade de classificar os gráficos produzidos com a técnica escolhida;

6. Escrita da dissertação.

Principais Contribuições

A principal contribuição desta dissertação consistiu na aplicação de técnicas modernas de visualização de dados para classificar ataques na rede. A técnica de visualização conhecida como *Circos* foi utilizada no âmbito deste trabalho. Entre o capítulo 3 e o capitulo 5 estão presentes as várias representações desenvolvidas, sendo que no final são discutidos os resultados. Embora o processo descrito seja manual, é mostrado que é possível classificar gráficos de registos

de tráfego desconhecidos a partir de gráficos de tráfego conhecido. A abordagem fornece, assim, uma prova de conceito e conclui-se que é possível produzir uma classificação automática, utilizando por exemplo, mecanismos de inteligência artificial com processamento de imagem. A segunda contribuição provém da elaboração do estado da arte em termos de conceitos relacionados com a visualização de ataques e da construção de um grande conjunto de visualizações utilizando o *Circos*. O estudo da arte em termos de técnicas de visualização é o tema principal do capítulo 2, enquanto a descrição dos conjuntos de dados, *scripts* para construir o *Circos* e as amostras dos resultados estão incluídos nos capítulos 3, 4 e 5.

Estado da Arte

Para a elaboração do trabalho descrito nesta dissertação foi necessário primeiro estudar abordagens para a visualização de tráfego na rede existentes, assim como o estudo de várias técnicas de visualização de outras áreas. Este estudo é descrito no capítulo 2. Resumidamente é dito que as técnicas utilizadas para a criação de visualizações podem ser as mais variadas possíveis. São apresentadas técnicas de visualização de ataques, que propõem visualizações recorrendo a 3 Dimensions (3D) [JPLL09] [NUCB12] [CGM+13] [NAU+13], estando também discutida uma técnica baseada no motor de um jogo [HA06], onde certos eventos de segurança despoletam várias ações, como por exemplo, a actualização das listas de controlo na *firewall*. As técnicas baseiam-se na visualização das atividades da rede, e quando existe alguma atividade fora dos padrões normais, esta é identificada. Foram também analisadas várias técnicas de visualização de dados. Entre as várias técnicas temos por exemplo, a *Edible Or Medical*, uma técnica de visualização de dados que representa vários tipos de plantas, e se estas são possíveis de serem utilizadas para fins medicinais, comestíveis ou ambas. É possível também observar que existe um gráfico interativo *Paperscape*, que é um repositório online sobre artigos científicos.

Circos e os Dados

O *Circos* [KSB+09] apresentado na secção 3.2 foi inicialmente desenvolvido para a visualização de dados do genoma, mas rapidamente proliferou para outras áreas de investigação. É uma visualização circular e é geralmente ideal para a observação de relações (ou falta delas). Ganhou popularidade devido à sua flexibilidade de representação, definindo as quantidades ou objectos que a superfície circular deve ligar, sendo que as ligações são curvas, de ponto a ponto, dentro de um circulo. Nesta mesma secção, são apresentados os primeiros gráficos *Circos*, obtidos a partir de tráfego *Voice over IP (VoIP)*. Os conjuntos de dados utilizados no âmbito deste trabalho provêm de duas fontes principais: alguns do *Massachusetts Institute of Technology (MIT) Lincoln Laboratory* [MIT14], enquanto que os outros eram propriedade do laboratório em que

este trabalho foi realizado. Os conjuntos de dados do *MIT* contêm vários ficheiros. Para este trabalho, apenas os ficheiros `tcpdump` e *list* foram relevantes. O primeiro contém todos os pacotes que foram capturados, enquanto o segundo contém informação importante sobre o conjunto de dados. Os conjuntos de dados do laboratório são similares aos do *MIT*. Os ficheiros estavam também no formato `tcpdump`, sendo que a única diferença entre eles são que os conjuntos de dados capturados no laboratório foram realizados recorrendo-se a aplicações legítimas e, nos do *MIT*, os conjuntos contêm ataques. Para ser possível observar os dados, utilizou-se a aplicação *Wireshark*.

Método e Ferramentas

No capítulo 4 são mencionadas as ferramentas utilizadas para o desenvolvimento dos gráficos *Circos*. Para realizar este projecto foi necessário o uso de várias tecnologias e *software*. As três principais tecnologias/*softwares* utilizadas foram o *Wireshark*, para a análise, filtragem e processamento do tráfego de rede, *Python*, para criar *scripts* para a manipulação e transformação dos dados, e o *software Circos* (que é desenvolvido em *Perl*) para criar as representações *Circos*. A ferramenta *Wireshark* foi útil nesta parte do trabalho, através do qual foi possível aceder à informação relevante existente nos conjuntos de dados. É capaz de mostrar informação sobre os vários campos de vários protocolos, tais como os endereços *Internet Protocol (IP)* destino e fonte, números de porta, nome do protocolo, entre outras informações. Foram também apresentados os gráficos *Circos* relativos às investidas de uma ferramenta de catalogação de vulnerabilidades em rede, conhecida como *Satan*. Por último são explicados os *scripts* desenvolvidos (em *Python*). Para a execução dos *scripts* é necessária a informação de *output* do *Wireshark*. Após transformar os dados fornecidos pelos *Wireshark*, os *scripts* produzem três ficheiros necessários para o *software Circos*. Cada um desses ficheiros contém informação relevante, sendo que no primeiro contém informação sobre as etiquetas (*IPs* e portas), o segundo, tem informação sobre as ligações entre as várias etiquetas (ponto de origem, ponto de destino, entre outras), e terceiro e último ficheiro guarda o nome das etiquetas e das cores associadas a cada uma.

Análise dos Resultados

No capítulo 5 são inicialmente apresentados os gráficos *Circos* criados a partir de tráfego legítimo. Foram desenvolvidos gráficos para quatro conjuntos de dados legítimos, entre eles incluem-se *VoIP*, *HTTP*, *Peer-to-peer (P2P)* e *Secure Shell (SSH)*. De seguida, apresentam-se e discutem-se os gráficos relativos a registos de tráfego cujo conteúdo é conhecido e se referem a ataques. Entre os ataques temos o *Neptune*, *Back*, *NMAP*, *Dictionary* e por último o *Satan*.

Após a apresentação dos gráficos conhecidos, foi realizada uma classificação através de uma comparação com os gráficos desconhecidos com o objetivo de verificar se era possível identificar a que conjuntos de dados pertenciam os gráficos. Por último, o capítulo tem uma discussão sobre os resultados obtidos. Para ser possível identificar a que conjuntos de dados pertencem os gráficos, é necessário primeiramente saber as características de cada protocolo, assim como de cada ataque. De seguida é necessário observar atentamente os gráficos conhecidos e descobrir quais as características que podem ser únicas e úteis na identificação dos tráfegos. Entre os nove conjuntos de dados desconhecidos utilizados, foi possível identificar oito. O ataque *Satan* foi o único cujo tráfego não foi identificado. Por outro lado, entre os desconhecidos, foram classificados dois conjuntos de dados como pertencendo ao tráfego *HTTP*.

Conclusões e Trabalho Futuro

Esta dissertação aborda o problema de produzir visualizações úteis de grandes quantidades de dados, nomeadamente para o tráfego de rede e ataques relacionados. Tudo começou com a análise de algumas visualizações de ataques na rede, para depois convergir para o objetivo mais específico, o da classificação de tráfego de rede e identificar ataques utilizando a técnica conhecida como *Circos*. Uma das principais conclusões é que o *Circos* pode realmente ser utilizado para esse fim, utilizando características simples de tráfego. Os objectivos deste programa de mestrado foram alcançados: um grande conjunto de gráficos *Circos* para tráfego de rede, e para ataques de rede clássicos, foram construídos durante o decorrer deste trabalho; visualizações com muitos dados foram discutidas nesta dissertação para identificar os aspetos que as visualizações enfatizam; a experiência e os resultados discutidos foram apresentados no capítulo 5 e provado que o *Circos* pode ser utilizado para identificar classes de tráfego de rede e ataques. Muitos gráficos foram criados e analisados. Para cada conjunto de dados foram criados pelo menos seis gráficos *Circos* diferentes. Dependendo da classe de tráfego ou a aplicação associada, a maioria dos gráficos provou ser útil. Na fase final do programa de mestrado, os gráficos foram divididos em dois grupos principais: um com gráficos cuja proveniência era conhecido; outro cuja as classes de tráfego ou ataques eram desconhecidos. Cada um destes grupos foi subdividido em 9 subgrupos, correspondente a 9 classes de tráfego diferentes ou ataques. Foi possível identificar 8 das 9 classes ou ataques, que foi mais que o esperado, tendo em conta o conjunto de características utilizadas. O conjunto que não foi corretamente identificado pertencia a um ataque, com comportamento semelhante ao tráfego *HTTP*. Os resultados argumentam a favor do potencial desta técnica de visualização de dados aplicado ao tráfego de rede.

Abstract

Data visualization techniques comprise crucial resources in many research and professional areas. Effective representations often contribute to the understanding of the overall picture behind a large volume of data, sometimes leading to novel discoveries or to an efficient synthesis. Due to the large amount of data that computers handle nowadays, many modern data visualizations techniques were designed to deal with such large data sets, exhibiting unique characteristics. In the information era, computers (and their operators) and networks are also amongst the biggest sources of raw data, though they are also used in its processing and storage. Many network monitoring systems and security appliances make usage of traditional data visualization techniques in reporting functionalities or to provide practitioners with status information.

The scope of this work falls within the intersection of the fields of network security and data visualization techniques. Its objectives are to study modern approaches to represent data, which may be currently being used in other areas, and apply one of those approaches in the visualization of network traffic and attacks. Assessing the usefulness of the visualizations was also an objective, along with the constitution of a large data set of representations for several traffic classes and classical network attacks.

A technique known as *Circos*, widely used for genomic representations, was the one applied for achieving the objectives of this masters program. Many representations for at least 18 different traffic traces were produced along this work, with many analyzed with detail in this dissertation. These traces, containing traffic generated by contemporary applications and classical network attacks or probing activities, were selected from two datasets. In order to produce the *Circos*, a minimal set of traffic characteristics was identified,and several scripts for automating the processing were implemented. Towards the final part of this work, an experiment based on the (human) comparison between nine labeled and nine unlabeled *Circos* was set up to demonstrate that the obtained representations were useful up to the point of being used to identify traffic classes or attacks. During the experiment, it was possible to correctly identify eight, out of the nine, traces (one of the attacks was incorrectly classified as HTTP traffic), proving the usefulness of this technique in this field.

Keywords

Circos, Graphical Representation of Data, Network Based Attacks, Network Traffic Monitoring, Modern Data Visualization Techniques, Traffic Classes

Contents

List of Figures

List of Tables

Acronyms and Abbreviations

3D 3 Dimensions

2D 2 Dimensions

3DSVAT 3D Stereoscopic Vulnerability Assessment Tool

ACK Acknowledge

ACM Association for Computing Machinery

API Application Programming Interface

ASR Attack Surface Reduction

BPMN Business Process Modeling and Notation

CPAN Comprehensive Perl Archive Network

CSS Cascading Style Sheets

DARPA Defence Advanced Research Projects Agency

DDoS Distributed Denial-of-Service

DoS Denial-of-Service

EAS External Attack Surface

FIN Finished

FPS First Person Shooter

FTP File Transfer Protocol

FRE3DS Framework for Rendering Enhanced 3D Stereoscopic Visualizations for Network Security

HCI Humam-Computer Interaction

HIDS Host based Intrusion Detection System

HTTP Hypertext Transfer Protocol

HVS Human Visual System

IAS Internal Attack Surface

IDE Integrated Development Environment

IDS Intrusion Detection System

IoT Internet of Things

IP Internet Protocol

LLC Logical Link Control

MIT Massachusetts Institute of Technology

NCR National Cyber Range

NIDS Network based Intrusion Detection System

OS Operative System

OSI Open System Interconnection

P2P Peer-to-peer

PCF Partial completion Filters

PERL Practical Extraction and Report Language

PHP Hypertext Preprocessor

PSF Python Software Foundation

QoS Quality of Service

RAVEN Real-time Attack Visualization Environment

RINSE Real-Time Immersive Network Simulation Environment

SNMP Simple Network Management Protocol

SIEM Security Information and Event Management

SIMTEX Simulator Training Exercise Network

SSH Secure Shell

SYN Synchronize

TCP Transmission Control Protocol

TDNN Time Delay Neural Network

UDP User Datagram Protocol

VoIP Voice over IP

Chapter 1

Introduction

This dissertation describes visualization techniques to network attacks and security events. In this chapter, the motivation and the scope to approach this subject is discussed, followed by the problems statement and objectives and how to overcome these problems. The adopted approach to solve the problem, the main contributions are the two next sections. In the last section, the contents of each chapter of the dissertation will be described.

1.1 Motivation and Scope

Computer science is a very dynamic area of knowledge. The technology associated with this area is constantly evolving and effectively changing the life of humans. For example, mobile devices are growing both in number and in terms of processing and storage capabilities [Tri15]. The proliferation of the IoT will eventually lead to a highly connect world of devices, capable of interacting amongst themselves and the humans via the Internet. Prospectively, such technology has the potential to make human life better, but also to worsen existing security problems or spawn new ones. In the internetworked world of today, many response teams have to handle security incidents in a daily basis [Lou15], spanning from amateur national threats to structured international attacks. Dealing with security incidents is difficult due to many reasons, but mostly because some attacks are designed or performed to be stealthy and hard to correlate. E.g., a single attack may look like a series of isolated incidents. Typically, incident response teams or cyber-security experts use tools that aid them to visually identify threats or correlations, though these systems are not perfect.

Many attacks are network based and may reach their target via different paths. As such, techniques for the visualization of network events or traffic classes comprise a critical subset of the aforementioned field, on which this dissertation is focused on. Traffic visualization is important for many monitoring systems also (not only security related), since they provide quicker ways to understand the behavior of a network, very useful for network and system administrators. It is common for NIDS, firewalls or security appliances to output charts (e.g., bar or line charts) concerning bandwidth usage, number of connections per time unit, statistics on dropped packets, amongst others, in the reports that they produce after network or security

events [Cis15] [Spl15] [DEL15] [Rev14], e.g., Allot communications e CISCO]. Data visualization is becoming increasingly important on traffic monitoring and security because networks are also becoming more complex, as the number of connected devices and processing capabilities increase. Though traditional data representation and visualization techniques, such as classic bar or line charts, are suitable for many purposes in this field, it is worthwhile to keep surveying the techniques for data visualization, trying to adapt them, and assess their usefulness.

This mater's dissertation is focused on assessing the usefulness of applying modern data visualization techniques to computer network security and traffic monitoring. More specifically, it will try to assess if particular data visualizations can be used to classify network traffic, detect intrusions or identify attacks. Within the computer science area, it falls in the intersection between the two major axis of traffic analysis and classifications, and security. By choice, it converges to the assessment of the possibility of taking advantage of data visualizations for classification of classical attacks. Under the 2012 version of the Association for Computing Machinery (ACM) Computing Classification System, a *de facto* standard for computer science, the scope of the master's program, reflected in this dissertation, falls within the categories named:

- **Security and privacy∼Intrusion detection systems**
- **Security and privacy∼Denial-of-service attacks**
- *Security and privacy∼Firewalls*
- Networks∼Network monitoring

1.2 Problem Statement and Objectives

This dissertation addresses the problem of finding efficient means to graphically represent large sets of data, namely for the case of network traffic. The specificity of the problem spans from the fact that the data of interest refers, in the scope of this work, to computer based attacks, which are dominated by artificial processes, to the complex nature of source of such data (computer networks). The events that the attacks generate are often difficult to represent in a human friendly manner, but such representations are useful for the detection, classification, understatement and reporting of such events.

The main objectives of this work are thus to study modern data visualization techniques, applied in other areas of science with different purposes, and assess the applicability of at least one of those techniques in the context of the analysis of network attacks. Secondary objectives

of the master's program include identifying the best set of network traffic characteristics for producing the representations; constructing a large set of data visualizations for contemporary network traffic and analyzing them; delivering fresh visual representations for well known classical attacks using the implemented technique(s); and implementing a proof-of-concept that demonstrates its usefulness. Achieving such goal will most certainly require the definition of datasets for analysis and investigate how to adapt the data related with computer networks, attacks and security incidents to these techniques.

1.3 Adopted Approach for Solving the Problem

To accomplish the objectives described in the previous section, the research work of this masters program was divided into the following phases:

- **Phase 1** consisted in the revision of the state-of-the-art in terms of data visualization techniques and tools, intrusion detection, security events correlation and Security Information and Event Management (SIEM) systems; getting acquainted with the research problem, identify and get to know the tools and technologies useful in the context of this work, namely in terms of potential programming languages for implementing scripts and required software packages;

- **Phase 2** was focused on the identification of data visualizations techniques and associated tools, which led to the in-depth study of the selected data visualization technique and related software library and to the preparation of several preliminary experiments for getting acquainted with the way of functioning of such library;

- **Phase 3** was focused on the selection of data to be analyzed, namely of traffic traces containing normal and attack related traffic. The traffic characteristics that would be used to produce the data visualizations were also identified in this phase, which required using and adapting the software for producing the data visualization technique, studying the correct syntax for input data and how to control the outputs;

- **Phase 4** included the prototyping of scripts to automatically process the datasets and produce a structured set of graphical representations for the characteristics identified in Phase 3 using the technique selected in Phase 2. This phase included also a preliminary analysis of the results;

- **Phase 5** consisted in conducting a series of experiments for showing that it is possible to classify network traffic and attacks using the data visualizations produced in the scope of

this work;

- **Phase 6** was devoted to the structuring of the obtained material and to the writing of the master's dissertation.

1.4 Main Contributions

The main contribution of this master's program consisted on applying modern data visualization techniques to network traffic classification, namely to the identification and classification of network attacks. The data visualization technique known as Circos was used within the scope of this work. Data representations of several aspects of network traffic are included in the dissertation from Chapter 3 to Chapter 5, with the main results showing that the classification is possible discussed in the later. Though the procedure described herein is completely manual, it shows that by comparing data visualizations from both known and unknown traces, it is possible to classify the unknown ones. It provides a proof of concept, on which it is possible to elaborate in order to produce an automatic classification procedure, e.g., combining artificial intelligence mechanisms with image processing.

The survey over the state-of-the-art in terms of concepts related with visualization of network attacks and the constitution of a large structured dataset of data visualizations using Circos, for contemporary network traffic and for classical attacks, comprise secondary contributions of this work. The study of the state-of-the-art in terms of techniques for data visualization and their applications in network monitoring is the main subject of chapter 2, while the description of the datasets, scripts for building the Circos and samples of the resulting representations are included in chapters 3, 4 and 5. Notice that this work included the identification of the most meaningful traffic characteristics, in terms of the output of Circos, and the preparation of the suitable inputs for its packages. Additionally, it was necessary to adapt some functions of the libraries in order for them to support larger inputs.

1.5 Dissertation Overview

This dissertation is organized in six main chapters. The body of this dissertation is constituted by four chapters, preceded by the introductory chapter and succeeded by the conclusions and future work. The contents of each one of the chapters can be summarily described as follows:

- Chapter 1 – **Introduction** – presents the scope and the motivation behind the work de-

scribed in this dissertation, as well as the problems and objectives. The adopted approach for solving the problem is also outlined in this chapter, along with main contributions of the underlying research work. The organization of the dissertation is the last subsection in this chapter.

- Chapter 2 - **Related Work** - contains a discussion on visualizations used for network attacks and a overview of data visualization techniques. It also contains a review of works on this area, with focus on the related works about network attacks and visualizations.

- Chapter 3 – **Data Visualization Techniques and Datasets** – contains a description of the visualization techniques used in the scope of this work. The focus is on the visualizations using the so-called Circos technique, explained in detail in an initial section of the chapter. One of the sections of this chapter is dedicated to the description of the datasets using along this work to produce the several visualizations and obtaining the results discussed afterwards.

- Chapter 4 – **Method and Experimental Setup** – outlines the technologies and libraries utilized to automate the analysis and produce data visualizations. The flow evolves to the explanation of the procedures used for the analysis of network traffic from the datasets and, at the end of the chapter, some of the most important scripts implemented in the scope of this work are briefly discussed.

- Chapter 5 – **Analysis of the Results** – starts by presenting many visualizations for several legitimate network traffic classes, complementing the set of data representations included in other chapters of the dissertation. Subsequently, it includes the analysis of Circos obtained for classical network based attacks. Towards the end of the chapter, a discussion on how the data visualizations can be used to easily classify some of the attacks and network traffic is included.

- Chapter 6 – **Conclusions and Future Work** – is devoted to presenting the main conclusions of this dissertation, with the focus on the results obtained, and to pointing out the potential future lines of research.

Chapter 2

Related Work

2.1 Introduction

The community working in computer security, specially intrusion detection, has been actively working on means to visualize and detect attacks, signaling the importance of the visualization aspect in this field. Data visualization results normally in improved detection, correlation, classification and reporting of attacks. This subject, nonetheless, is not that explored yet, though there are several different and interesting approaches in the literature. This chapter is devoted to the description of some of those approaches, namely to some of the most peculiar ones (e.g., one based in a First Person Shooter (FPS) game), which may offer helpful insights to the work presented herein. The data visualization technique used in the scope of this work (Circos) will be described in the following chapter. The discussion is divided into two main parts: section 2.2 presents related works in the specific topic of network attacks visualization; while section 2.3 presents an overview of other interesting data visualization techniques.

2.2 Related Work on Visualization of Network Attacks

In this section, some publications related to visualization of network attacks and network attacks are introduced and summarized. While it is possible to find numerous publications regarding network attacks in scientific databases, papers on the theme that adds the data visualization keywords are more scarce.

Article [KLS13] describes a visualization technique that analyzes the HTTP headers of requests and their responses to detect and visualize web attacks. In the results are included visualizations of scanning vulnerabilities, password brute force, and position tracking of the attacker. In one of the proposals, the system structure a database of patterns of attacks using SNORT. When an anomaly is detected, an alarm is activated. In the case of the vulnerability be password brute force or scanning is used the 'Nikto', which is a scanning tool to discover vulnerabilities in the server or application. In another test, where attackers insert 'random' text string to perform login, the frequency of requests and the text is saved and may indicate an attempted

attack. The tool includes the striker position display through the use of Google Maps Application Programming Interfaces (APIs), and the position displayed on the world map. The article focuses on the research, visualization and threat detection.

The IDSRadar [ZZF+13] monitors the network using pie charts in real-time using five categories of entropy functions to analyze irregular behavior. Summarizes the interactions and filtered possible to detect intruders. First, the Intrusion Detection System (IDS) logs (captured with the SNORT) are stored in a MySQL database, then the IDSRadar reads the data (source IP, destination IP and the types of alerts) in user-defined time intervals, calculating statistical information on the attacks. Then the IDS presents the alerts and information calculated in graph form. IDSRadar allows to view a variety of information, including the begin of the attacks, and who are the attackers and victims. The IDSRadar is a tool to assist in understanding the IDS alerts and identify abnormal behavior in the network in order to reduce the number of false positives.

Paper [MMB06] presents a port-based display system (Transmission Control Protocol (TCP) and User Datagram Protocol (UDP)) for efficient identification of network and port scan. It also presents some guidelines to incorporate this view in IDS systems. Among other features, the port scans and network scans can usually be readable with the methods presented, despite the limitations of the data collected. You can identify characteristics about safety. The tool used was divided into three levels: high, mid and low-level. At each level the dataset is different. In the high-level is used throughout the dataset, the mid-level displays all the ports in a time interval, and the low-level its possible to visualize a port in any time interval. The approach allows the display patterns at a level, which can go unnoticed on a different level. According to the authors, port scans that occur in a slow manner and with a random order will always be difficult to detect. However, this problem may be overcome by reducing the dataset. Overall, the tool is able to provide the network status without compromising the network infrastructure.

The authors of [FMK+08] present a system to analyze NetFlow data through a relational database system. The NetFlow data is interconnected with an IDS alerts to explore suspicious activities on the network. Information is displayed through TreeMap. In the case studies is checked how the tool can be used to highlight the importance of alerts, revealing distributed attacks and analyze the services used in the network. Authors present a NFlowVis system providing an abstract visualization of the entire network and aggregate visualizations of the IDS data for future analysis of the goal, your network traffic, and what are the victim systems. The article uses the combination of TreeMap view a clustering algorithm and Hierarchical Edge Bundle to group information in a meaningful way. The Hierarchical Edge bundle is a flexible and generic method that can be used in conjunction with existing visualization techniques tree, allowing users to choose the tree technique, facilitating their integration. This technique reduces the visual clutter when dealing with a large number of edges.

Article [CDK+09] is a survey of the state of the art detection and defense against network attacks. Due to the increasing development of systems and computer networks, also the number and complexity of threats and vulnerabilities of these systems is increasing, fulfilling a major concern of large companies. This evolution makes them more dependent on systems, extending from horizon to attack by malicious individuals, who are also more and more. There are certain attacks that are easy to identify through the network traffic analysis, others are easily detected on the machines that contain the attacked application. The article concludes that its necessary to continue to invest in community analysis of available data, because the attacks continue to evolve. The intrusion detection and network monitoring are the most important ways to approach in the context of security.

The authors of [AH13] propose a real-time NIDS unattended for high-speed networks that work in normal or encrypted communications through the network behavior monitoring. In normal communications, the system own the ability to detect Denial-of-Service (DoS), Distributed Denial-of-Service (DDoS) attacks or scanning, obtaining a huge amount of data. In encrypted communications, the system lists the traffic of the attackers to discover similarities from previous communications in order to detect possible Bot-Masters. On the lines of future work, the authors report that will still implement your model and test it with traffic samples with different types of attacks as well as use the model simulations by connecting it to a public network. Your objective is to test it in real-time.

The main objective of the authors of [ECS05] is to identify, in advance, the attacks before they are successful, by monitoring and visualization. Majority systems relating to intruders failure to provide a good view to interpret and generate information. Historically, the visualization techniques are applied to monitoring and analysis of network, mainly for quality and performance of the network, the individual packets and even email messages, but it does not provide enough details about the attack. In the case of the attacks that are more sophisticated, such as the case of a port-scan camouflaged, that will always become more difficult to detect, display offering allows the accumulation of temporal events for further analysis. The authors suggest that new techniques must be added in order to increase the range of attacks that can be analyzed. While this work focuses on the identification and analysis of attacks, its necessary to implement the future defense of these attacks.

According to the authors of [vHPBI13], two methodologies were presented in order to test the effects of attacks and defense mechanisms. One approach consists in reproduce the effects via simulation. The second approach is to construct a test bed on which the operating systems and applications are installed and utilized. That last approach allows a more precise analysis. In this paper the authors describe a virtualized testbed network for creating test scenarios attacks and malware detection. This article aims not only to exhibit the ease with which the simulated

traffic can be added to a test network, but also to illustrate the importance of the generated traffic. In a typical network may be easier to detect anomalies, since any increase in traffic could indicate an attack. The simulated traffic is described in this study simulates accurate volumes of a small-scale enterprise, without the use of expensive tools.

In article [NJKJ05], some new visualization techniques are described to deal with the complexity of attack graphs. A visual way is introduced that reduces the complexity and improve the representation. The described techniques may be applied individually or in combination, provide multiple visualizations of the attacks. Each visualization own its strengths, and their combination can be applied to increase its effectiveness. The authors introduced a technique for filtering attacks, restrictions based on a hierarchy that allows the user to navigate the hierarchy to interactively control the subset (network) which was attacked. Its also able to link events detected the topology, allowing his exploration to trace and realize the impact of the attack. These techniques can be adapted to other techniques (common) visualizations based on network vulnerabilities and/or detection systems.

The authors of [AJA14] propose an intrusion detection system based on neural networks Time Delay Neural Network (TDNN). In the first test, the authors compare your system with the SNORT, and conclude that your system has recognized all the attacks described in the article, while the SNORT failed some of these attacks. Its assumed, with new rules SNORT would detect the attacks that have failed. In the second test, with the Defence Advanced Research Projects Agency (DARPA) datasets from MIT, the proposed system was able to recognize once again all attacks with a gear to SNORT. Tests show that the proposed system can detect attacks on different timelines. In the pre-processing phase, relevant features are extracted from various attacks to the neural network, which produces outputs that represent possible attacks.

Paper [SJ14] starts by differentiating External Attack Surface (EAS) Internal Attack Surface (IAS) and Attack Surface Reduction (ASR), and IAS is larger than the EAS. Administrators can usually reduce the EAS, by having access to the entire network and its components, while the attackers necessitate to discover the vulnerabilities of the system. In this article, the authors propose to extend the EAS instead of reducing the IAS, making false vulnerabilities that act as bait for attackers, so that these waste your time on these vulnerabilities, rather than on the real vulnerabilities of the system. The authors discuss three case studies. In Case 1, Expanding Surface Attack through Virtual Identities propose create a number of IPs virtual, modifying them periodically, so that the actual IP system mix the virtual. Case 2, Expanding Attack Surface through Secret Moving Proxy, the authors propose to create many proxies, and only some will be utilized, chosen at random. Case 3, Expanding Attack Surface through Dynamic Virtual Networks, is proposed to constantly modify IP, network topology, control access and routing through a virtualized network, thus producing many false systems, which gives the

possibility to analyze the behavior of the attackers as the administrators attack the virtual components.

Article [KSV07] focuses in three types of attacks: the Partial Completion Attacks, the Attacks That Do Scanning and Bandwidth Attacks. The authors propose a new technique called Partial completion Filters (PCF) to be scalable detection of attacks. In this article, are described the algorithm PCF, their behavior and their use in detection, and demonstrated their effectiveness in various types of attacks mentioned above. It also referred to the importance of network security, and its necessary evolution, as compared to other Internet functions, this is delayed compared to the forwarding, classification, Quality of Service (QoS).

Article [LPCB11] is divided into two parts: one dedicated to the private sector and academic research, and the second part focused on the area simulation cyber attacks. In the private and academic part starts by talking about cyber attacks utilizing various network simulation tools and attacks in order to be able to analyze future data. Among these tools are the ARENA, the Real-Time Immersive Network Simulation Environment (RINSE), the SECUSIM, the OPNET and NetEngine. The biggest problem encountered for the applications used in academic research is that does not allow users to realize the impact that the attacks can cause companies or countries. In the public sector, the tools mentioned and produced by government organizations around the world are Simulator Training Exercise Network (SIMTEX) [McB07], the CAAJED [ML08], the Cyber Storm [DoHS15] (I, II and III), DARPA National Cyber Range (NCR) [Age15] and India's Divine Matrix [Sin10]. In this sector, governments have increasingly invested in staff training and the defenses cons various scenarios in cyber security that may happen. In conclusion, there are many efforts to increase the quality of the simulations and better understand the problems of cyber attacks, both private and public level.

In [JPLL09], the authors propose a visual interactive network connection system called NetViewer. This tool is designed in 3D view for representing traffic activities from the network flows and their patterns. The system is based in 3 parts, capture subsystem, database and network security visualization framework. WildPackets and OmniPeek are used to capture subsytem. NetViewer have 3 main interactive methods used: (1) Selection technique, this method make user view the data easily. (2) Filter technique, enables users to change the set of data based on some conditions. (3) Reconfiguration technique, allow users to change the way data items are arranged in order to provide different perspectives on the dataset. The experiments show that NetViewer can detect DDoS attacks, network scans and port scans.

In [SW04], it is explained how to specify and analyze attack models on the network, making the data input into the tool that generates charts automatically, and analyzes system vulnerabilities, the authors always refer to the toolkit. The authors created, implemented and tested algorithms to automatically generate chart attacks, to perform different types of analysis of

vulnerabilities that can be found,it also was created a support toolkit to create these graphics. At this point the authors are conducting various types of experiences, from different settings, in different sub-sets to determine different objectives of the attackers. The toolkit was created in two integrated systems, MITRE Corp.'s Outpost and the Lockheed Martin's ANGI.

The authors of [NUCB12] introduced the stereoscopic 3D visual framework called Framework for Rendering Enhanced 3D Stereoscopic Visualizations for Network Security (FRE3DS). This framework utilizes state-of-the-art 3D rendering techniques to assist secure visualizations in applications running over the network. They propose, through the framework, the 3D Stereoscopic Vulnerability Assessment Tool (3DSVAT) tool, which is an assistant for the rapid detection and correlation of vulnerabilities. The tool provides the monocular and binocular visualizations, which are different visualization levels for improving the administrator experience. This tool is able to perform a quick analysis of the vulnerabilities of data on the network, reduces visual noise, which is detected in some visualizations. The tool also reveals vulnerability characteristics in local networks and correlate information between nodes. An example of the visualization produced using 3DSVAT is included in Figure 2.1.

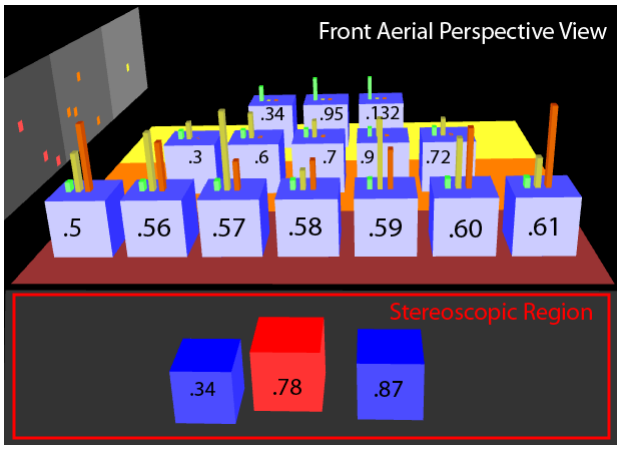

Figure 2.1: Aerial front view visualization produced by 3DSVAT.

The authors of [CGM+13] propose a tool called CyberVis, which combines the icons of traditional network diagrams with Business Process Modeling and Notation (BPMN). This logic connects the network layer, business processes and tasks, offers a flexible framework of support of any of them in intrusion alerts. Instead of filling the user with excessive information, CyberVis abstracted the visual part, to exhibit important information about attacks, and indicates the potential impact on the network and the tasks of the company. The CyberVis was designed in accordance with the guiding lines of the Human Visual System (HVS), resulting in emphasizing of serious attacks, or many small attacks with great impact, and relationship to other components in the visualization. The tool possesses two more functions. Deep-Dive, which allows the use of data, similar to a database interface, and the Forensic Mode, which allows playback (in the style of a film) of alerts that have passed on user-defined settings for analysis. Figure 2.2

includes a screenshot of the tool.

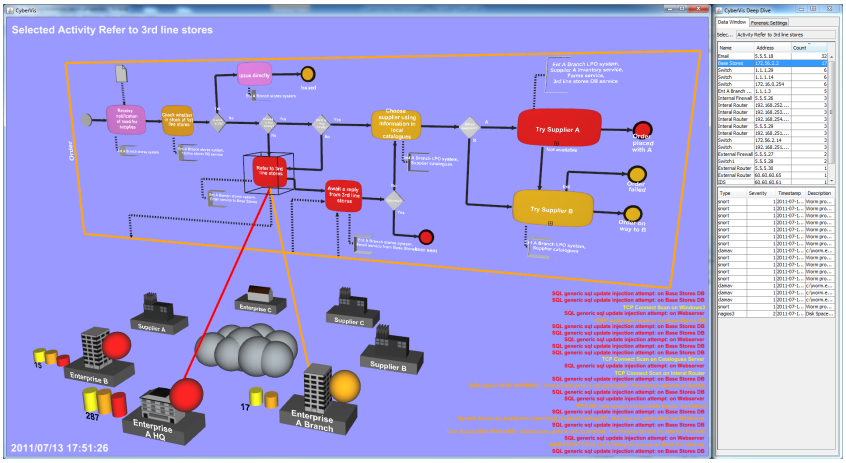

Figure 2.2: Screenshot of CyberVis (taken from [CGM+13]).

Visual Firewall, discussed [LTG+05], has 4 implemented views: real-Time traffic, visual signature, statistics and IDS alarm. These four views provide various levels of detail and temporal information that the system administrator needs to monitor systems in a passive and active manner. Each of the four visualizations, separately, provides specific details about the network traffic, the flow of packets, transfer rates and suspicious activities. The four perspectives combine to form a coherent network status illustration. Figure 2.3 contains an image of one of the views of the Visual Firewall.

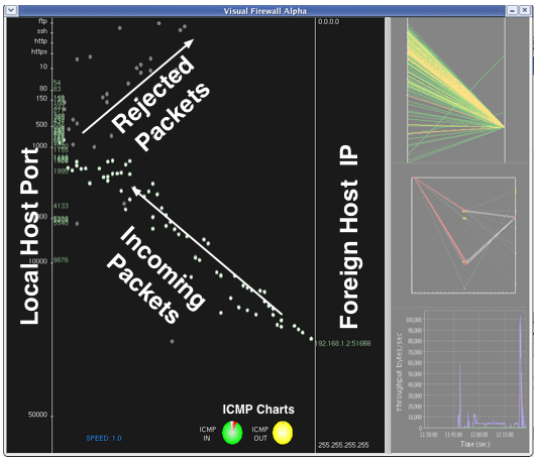

Figure 2.3: Real-Time Traffic View of Visual Firewall (taken from [LTG+05]).

The authors of [NAU+13] developed NAVSEC, a visualization module, which is a prototype system for navigating security visualization tools in a 3D network. The NAVSEC is a tool to reduce the noise that sometimes is included in views, especially for beginners. NAVSEC is illustrated in a case where a probation (scan) attack is disguised in a transfer of files with multiple connections, showing that, when using the tool, even a novice user is able to detect attacks that only experienced users were able to. NAVSEC utilize advanced visualization techniques based on a

database of interaction sequences from a community of experts to improve the experience of a novice user, provide an easy interpretation of the 3D network security, and perform a quick analysis of data network, increasing its efficiency. Figure 2.4 contains an illustration of what a port scan may look like in NAVSEC.

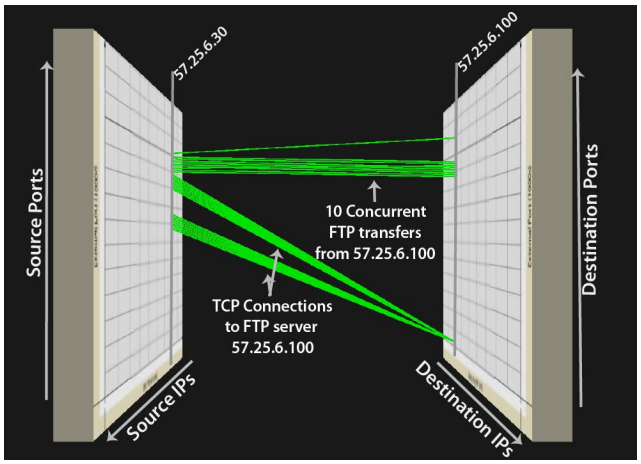

Figure 2.4: A potential stealthy port scan (taken from [NAU+13]).

The system described in [HA06] contains a 3D game engine that is used to transform network events into game entities. A user of the system is able to control the parts and features of the network that were translated into the game by playing it and interacting with the *visual metaphors* that was created. abnormal behaviors are targets for the players (administrators) that may resort to in-game weapons or actions, such as *shooting* or *cure*, to fight the threats and defend the network. These actions are translated back to appropriate network operations (for example, update the access control in the firewall). The orthogonal *visual metaphors* are actually embedded features with particular functions (e.g., detecting abnormal behavior on the network) and characteristics learned from network monitoring activities and NIDSs. Administra-

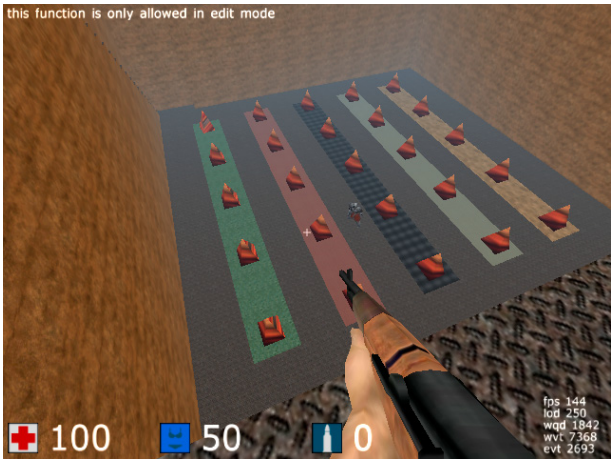

Figure 2.5: Virtual representation of the greynets with 25 addresses (taken from [HA06]).

tors can trigger (gun), heal (syringe) and adapt (pliers). These tools allow a user to interact with the virtual environment and, at the same time, make changes in real time to the network.

Shooting puts access control lists on the firewall, the pliers allows restrictions on bandwidth and heal undo operations. Figure 2.5 shows a virtual representation of the 3D game engine.

This article [CA04] examines visual fingerprints forgotten in several popular attack tools to better understand the methods used by the attackers, as well as the characteristics that identify these tools. These techniques are applied by attacker in a passive and virtually undetectable manner. This research explores various applications of visualization techniques and their usefulness to identify the tools utilized by attackers without the typical IDSs. The results demonstrate that these popular tools can be easily detected by passively sniffing and representing the resulting data with appropriate views. A major concern is that some tools, like *NMAP*, are extremely flexible, and experienced users can build attacks that trick this system. The visualization produced by passive visual fingerprint is shown in Figure 2.6.

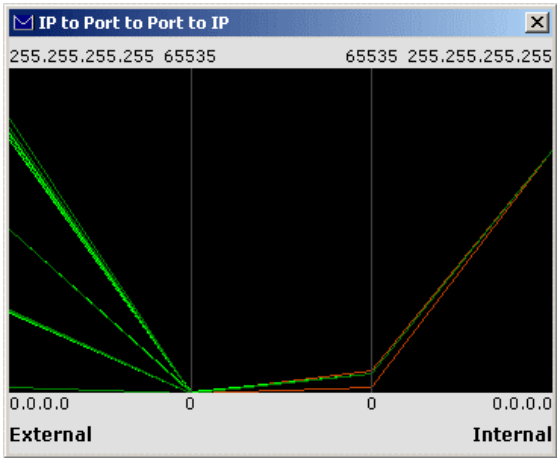

Figure 2.6: Passive Visual Fingerprinting for the information flow starting at the external IP, passing to the external port, then to the internal port, and finally to the internal IP (taken from [CA04]).

Riad et al. [REHA11] present a new framework with a visualization technique for the analysis of data coming from SNORT. This framework utilizes Hypertext Preprocessor (PHP) and Cascading Style Sheets (CSS) to accomplish its objectives. The intrusion detection is an intensive process and can not be performed without the aid of a computer. It is necessary to analyze potentially large amounts of data in real-time to be able to report abnormal use of networks and systems. The proposed framework creates bar charts from traffic, dividing the bars by protocols. The framework was proven effective to visualize intrusions detected by a SNORT system. A screenshot of the framework during execution is shown in Figure 2.7.

The authors of [HLH11] propose a research tool based on gestures named Real-time Attack Visualization Environment (RAVEN). It offers analysis features combined with a graphical environment for multiple users. Initially, the tool was designed to generate graphical representations of attacks and was later integrated in an IDS with improved visualization techniques and interaction, and adapted to the network model University of Tulsa. The architecture includes four

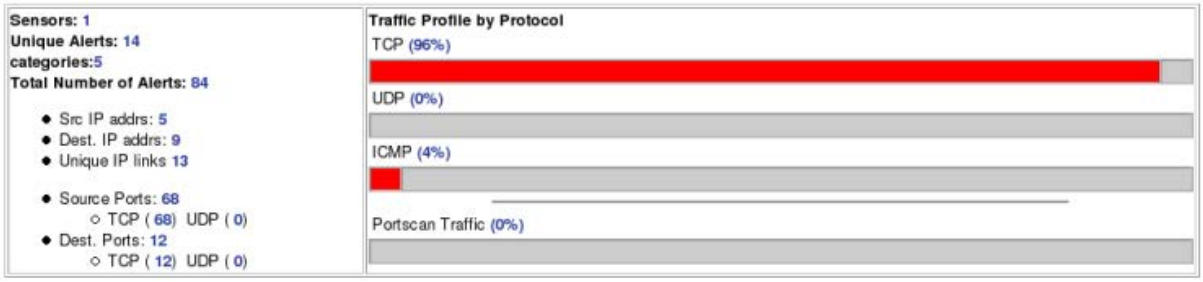

Figure 2.7: Screenshot of the framework developed by Riad et al. [REHA11].

main components: (i) the acquisition of the model, (ii) creation of attack graphs, (iii) visualization and (iv) analysis. RAVEN was proven to be an effective platform to assess the impact of technical Humam-Computer Interaction (HCI) on the analysis of graphs concerning attacks. RAVEN requires the network system to be analyzed to be modeled first. Only then, the system is ready to start producing the network visualizations and perform analysis. Each component of the visualization has a different colored badge, denoting the level of risk associated with each condition. It was argued that these visualizations are suitable to be used in professional environments after minor improvements. Figure 2.8 contains an image illustrating the dependency graph produced by RAVEN.

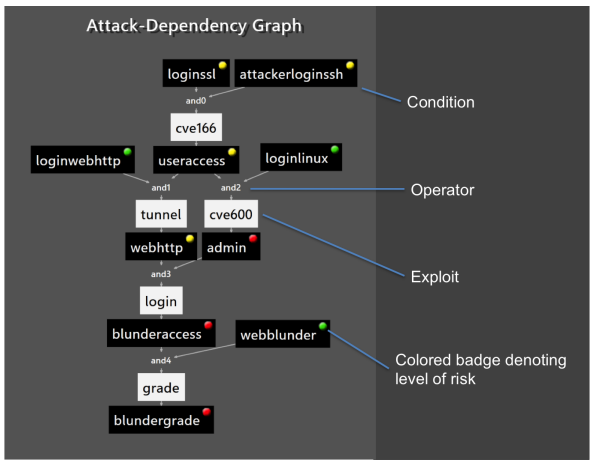

Figure 2.8: Attack dependency graph outputted by RAVEN (taken from [HLH11]).

2.3 Interesting Data Visualization Techniques

This section presents some contemporary data visualization techniques used in other areas of knowledge. It is interesting and sometimes inspiring to see how researchers create or adapt (known) techniques for their own purpose. The included data representations are amongst those that the author thinks that better transmit the objective of a good visualization technique, notably the ability to emphasize details and help learn novel facts from the data.

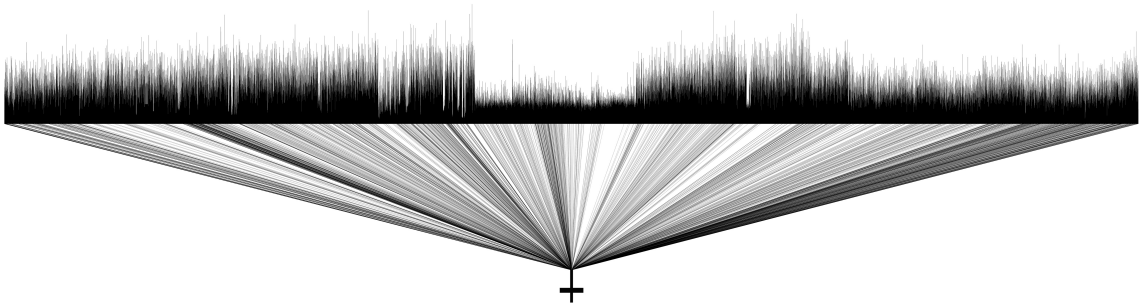

Figure 2.9: Representation for any mention of the word *God* in the Bible (taken from [Ein14]).

The first chart (Figure 2.9) of this section is a bar chart with an unusual amount of bars. In order to create the chart, the author represented the length of all verses of the Bible as bars, to then connect the word *God* to all of the places where it was found in this lengthy classical text. The result is a beautiful image, where the apparently Gaussian nature of several parts of the Bible are emphasized. In many sections of the chart, it is possible to see what seems like to be Gaussian noise. The several books composing the Bible are also seen in the figure. The word *God*, with uppercase *G*, was found 4,500 times.

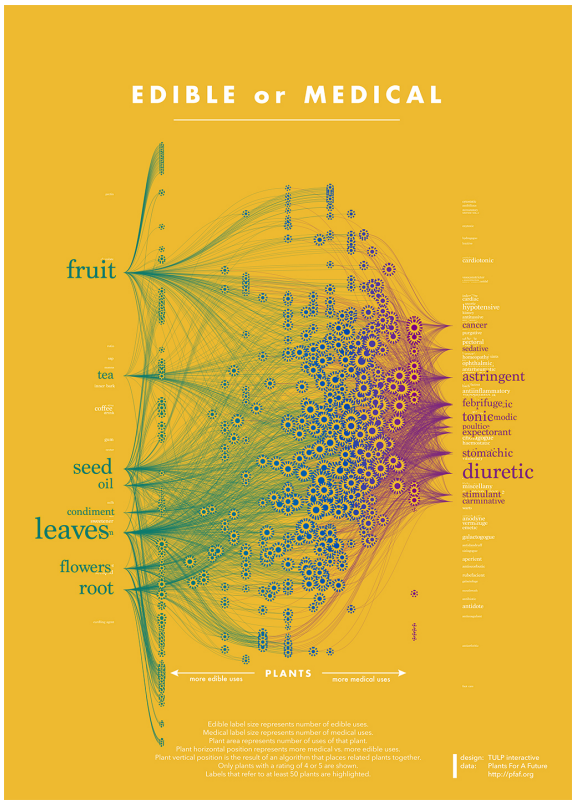

Figure 2.10: Representation for any Edible or Medical plant (taken from [Tul14]).

Figure 2.10 visually represents various types of plants. Plants can be edible, medicinal, or of other use for humans. The image shows the names/types of the edible plants and, on the left side, it shows the respective medical usefulness. In case more than 50 plants are connected to a feature, this feature is highlighted in the representation, in a directly proportional manner,

resulting in a very expressive highlight of the best edible plants which, in this case, are the leaves and fruits. The diuretic effect is the medicinal effect easier to achieve.

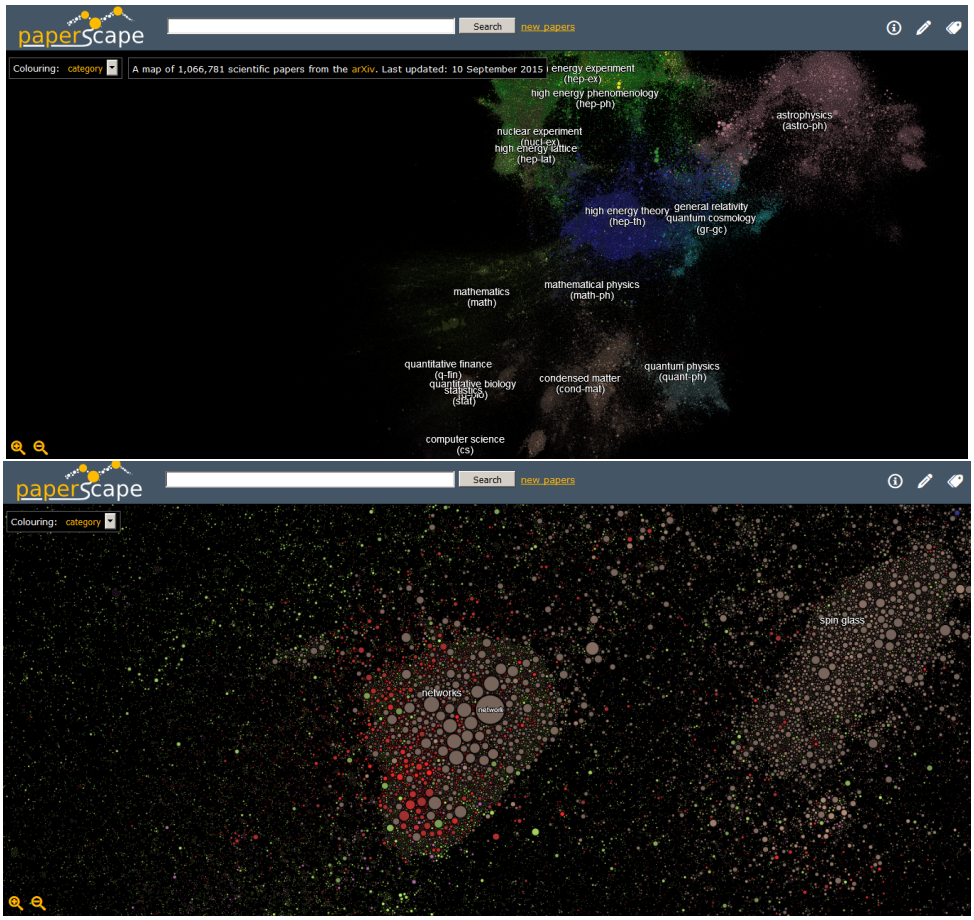

Figure 2.11: Representation of an interactive graphic to visualize an online repository for scientific research papers (taken from [DG13]).

Figure 2.11 is composed by two screenshots of the interactive graphic known as Paperscape [DG13]. Paperscape is an online repository for scientific research papers. The upper screenshot shows constellations named after the areas in which the several papers orbit. As one zooms into the figure, the sub-areas come into focus. At the end, several large or smaller circles populate the screen, whose size is directly proportional to the importance of the paper they represent in that specific field. he larger the circle, the more relevant the paper is, since the size is related with the number of citations of that paper.

Figure 2.12 contains a graphical representation of the the organization of Wikipedia in terms of categories, subcategories and articles. The representation is superb in emphasizing the subdivisions and the number of articles at the edges. For example, according to this chart, 2,424,305 Wikipedia articles are about people.

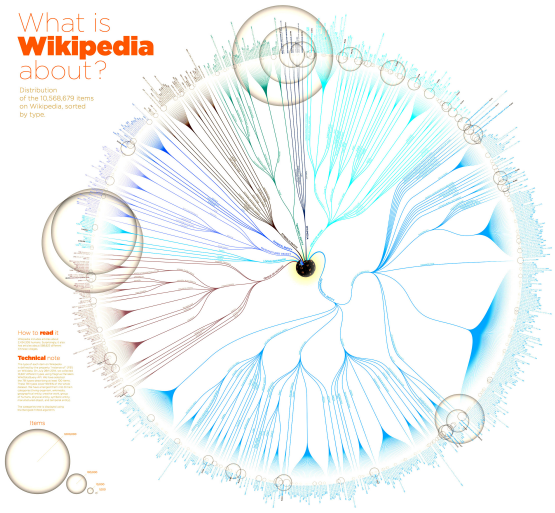

Figure 2.12: Representation of just over 10 million Wikipedia articles divided into categories(adapted from [PAC14]).

2.4 Conclusions

This chapters discussed some of the works in the specialized literature using different visualization techniques to model networks, represent traffic and attacks, and even manage the infrastructure and systems. The discussion shows that many researchers contributed to the development of new, and sometimes exquisite, techniques for different types of visualizations, also recognizing the importance it may have for the complex computer networking scenario. In spite of these efforts, there is still room for research and development in this field, namely in the network traffic monitoring and intrusion detection topics. From the analysis, and given the dynamic nature of the subjects at hands (computer networks and attacks), it can be concluded that it is important to constantly try to find new means that help minimize damage, categorize traffic and threats and help maintain increasingly large networks.

It is possible to find both 2 Dimensions (2D) and 3D data visualization techniques applied to network traffic monitoring, analysis and administration, as well as intrusion detection. Some of these techniques were developed under the assumption that they could ease the process of detecting attacks, for example for novices in the area. One proposal is based on translating the network administration into a FPS, which is uncommon, but proves that imagination may also be helpful in this area. Most works end proving that different data visualization techniques are useful to emphasize some aspects of the attacks, many times easing the process of detecting them.

This chapter provides a more detailed context to this master's program, enabling to better position this work and also justify its motivation. The next chapters can now be focused on the experimental part of the work, starting from the description of the visualization technique and

datasets used within its scope.

Chapter 3

Data Visualization Techniques and Datasets

3.1 Introduction

The data visualization technique used in the scope of this master's program, known as Circos in the specialized literature, is introduced in this chapter. As it will become clearer later on and briefly mentioned in chapter 1, one of the objectives was to produce visualizations of network traffic so as to identify attacks, on the one hand, and to classify them, on the other. The classification consists in naming the attacks starting from the produced representations. The intention of aiming for the traffic classification is to maximize proneness to potentially interesting characteristics of those representations. The idea is to assess if the visualizations are useful in the sense that they enhance those characteristics. Circos is described in section 3.2.

To better quantify the aforementioned usefulness of the technique, a typical classification exercise was designed. It required the construction of two datasets: one containing (labeled) traces of traffic generated by known applications and attacks; and another one containing traces of traffic with no information regarding its provenance, though it was generated by the same set of applications or attacks. Both datasets are described in this chapter also, namely on section 3.3.

3.2 Circos – The Data Visualization Technique Used in the Scope of this Work

Circos [KSB+09] was initially developed to visualize genomic data but it quickly proliferated into other areas of research. Circos is a way of representing data in a circular form, and it is typically accepted that it is ideal so observe relations (or lack of). Its popularity is mostly due to the flexibility of the representation since, at its core, the technique only defines that quantities or items at a circular surface should be connected by curves within the circle. This chapter and the following two include several Circos. The first charts can be found in figure 3.1, below.

By adjusting some settings in Circos and simply feeding new data, it rapidly became ready for new flights.It is possible to find several examples of areas where the Circos technique was applied in [CH14]. Some examples show that Circos was inclusively used by The New York Times in December 2007 to show the relationship between political debates and the name of the respective candidates. Another interesting example where Circos was applied to concerns *Human Migration Patterns*. In [NS14], four Circos charts were produced, each of them representing a five-year period since 1990 to 2010. They illustrate the recorded migration volume, including immigration and emigration of 123 countries in a very intuitive manner. The work reported in [oC14] about *Urban Planning* in Caceres, Spain, performed in January of 2011, provides also clear evidence of the flexibility and widespread applicability of the technique. In this case, they used Circos to visualize the relation of their urban planning strategy between businesses. The examples presented here exhibit how versatile Circos can be. This was one of the main reasons for its choice for this work, along with the typically beautiful and revealing representations it produces,

Circos is nowadays distributed in the form of software packages. It is freely available to download from `http://circos.ca/`. Its was natively written in Perl, but it is also possible to find an implementation in R programming. During correct execution and given proper input, this software converts data and associated information into a 2D circular chart. One of the benefits of circular visualizations like Circos is that it favors the visualization of the relationship between the data and the information, improving the underlying analysis process, unlike traditional charts, which are sometimes affected by noise in the presence of medium size datasets, which hinder their perception, crippling the objective of the chart. Another design goal of Circos was to develop a framework to producing high quality graphics with a beautiful aspect, which could facilitate human interpretation. If the charts are easier on the eye, human analysis will be less cumbersome.

In order to demonstrate the capabilities of Circos technique, several charts produced with the Circos software for some traffic traces used in the scope of this work are included below. Several hundreds of Circos were produced during the course of this master's program. Some of them were used for fine-tuning and debugging purposes only, while others were used for traffic and attack classification. To improve the explanation flow and avoid having all charts conglomerated in chapter 5, it was decided to include the charts and their respective discussion along the dissertation, when appropriate, even if within a different scope. Obviously, not all charts made it to this document. The Circos included in Figure 3.1 were obtained for a trace containing VoIP traffic. VoIP connections are typically one-to-one connections. If compared with many other charts included in this dissertation, it can be concluded that they possess really distinctive characteristics in relation to all of them, which is perfect for classification. As for the representations, both show relations regarding the TCP destination port. On the left

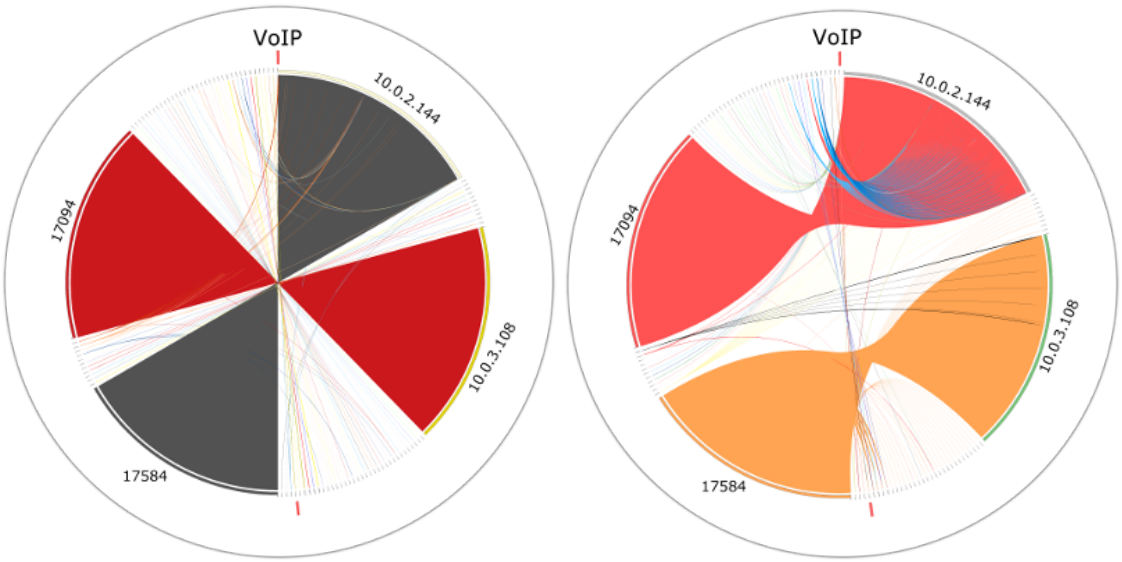

Figure 3.1: Circos representation obtained for destination port information from VoIP traffic.

side of the figure, the chart shows the relation between the destination port and destination IP addresses while, on the right, the relationship between the destination port and the source IP address is represented (i.e., the relation between the destination port and the IP address from which the packet came). They both show that the TCP ports 17094 and 17584 are the most used ones. The IP addresses `10.0.3.108` and `10.0.2.144` appear in both charts as both the source and destination of the communications, since VoIP is bidirectional (justifying their appearance in both charts simultaneously). In this dissertation, these charts are the first evidence that Circos has to the potential to transform raw data into beautiful visual representations.

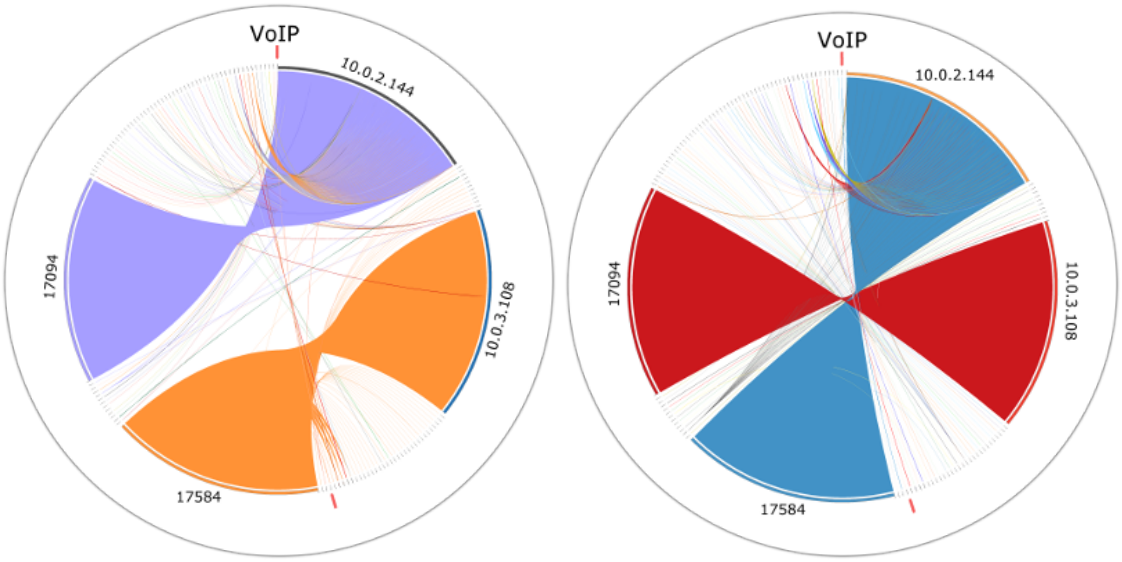

Figure 3.2: Circos representation obtained for source port information from VoIP traffic.

The Circos in Figure 3.2 were obtained for the aforementioned VoIP traffic trace also but, this time, the destination port number was replaced by the one the source port. The left chart shows the relationship between source ports and destination IPs while the chart on the right illustrates

the relationship between source ports and source IPs. As in the first two VoIP related Circos, ports 17094 and 17584 handle most connections. The involved IPs are the same. Notice that it is possible to represent different quantities or items in different parts of each Circos (e.g., TCP ports on the left side and IPs on the right side of each chart). Every entry on the traffic trace (i.e., each line on the trace file) corresponds to a curve in a Circos. This immediately leads to the conclusion that prevalent connections will be more visible in the produced representations and that sporadic connections will not achieve as much emphasis.

The charts in Figure 3.2 are similar to the ones in in Figure 3.1, mostly because, in this case (as in many others), the communications are two-way. This means that, for each destination and source IP addresses and ports combination, it is highly probable that the inverse combination is found in the same trace also. Because of this fact, and to avoid repetition within the dissertation, only one pair of these four charts will be included in most cases. Notice however that, for some attacks, specially DoS, connections may not be two-way, since the victim machine may be unable to answer to requests. In such cases, the four charts may be useful.

The last chart of this section shows the relations between the TCP destination and source ports of VoIP traffic. It is possible to see that the connection is established between the two ports; but also that there were short lived connections from 17584 to other ports. In order to produced cleaner charts, the resulting image was edited before being included herein. For example, only the most important port numbers or IPs are visible in the representations; while the software outputs images with all port numbers in the trace file.

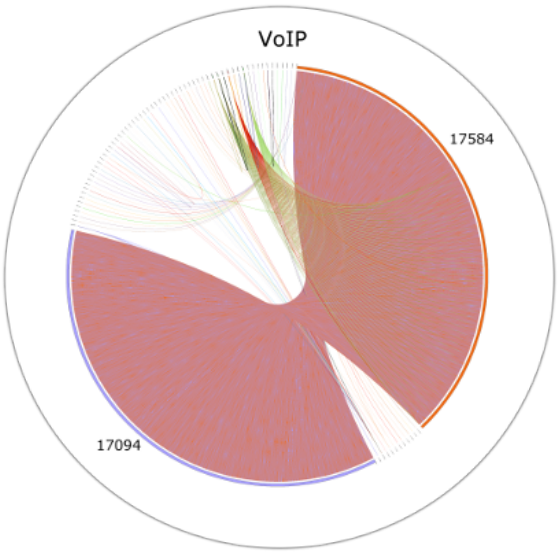

Figure 3.3: Circos representation obtained for destination and source port information from VoIP traffic.

3.3 Datasets

To test if Circos could be used to classify traffic and network attacks using only simple characteristics of IP packets, a classification exercise was idealized. It consists on producing charts for both labeled and unlabeled traces. It is obvious that it was necessary to sometimes simulate the situation of the unlabeled traces, since traffic was coming from known repositories and from laboratory sniffing experiments. In such conditions, it was normally possible to obtain the ground truth of the traffic. To overcome that problem, scripts for automatically construct the charts for unlabeled traffic were implemented, discussed afterwards with more detail.

The MIT datasets were created in 1998 and are divided into seven sets of data, each corresponding to one week (seven weeks of traffic traces). Each of these sets of data is further divided into five sets, one for each useful day of the week. Information about the topology and about equipment configuration is available with the datasets. Moreover, MIT included also a table with the attacks simulated in their laboratory set up. The table is comprised by two columns: the first has the attack name; while the second has an description of the attack. Despite this set of data is several years old, it contains classical attacks and probe activities, some of which are still used nowadays. Some of those attacks have changed or evolved, allowing them to have a larger impact. Every one-day dataset contains at least one attack.

The MIT datasets contain multiple files for each day. For this work, only the `tcpdump` [tcp15] and the `list` files were interesting. The first includes all packets that were captured for the given day; while the second contains useful information about the trace. Using Wireshark (or the `tcpdump` tool for Unix or Unix-like Operative Systems (OSs)), it was possible to filter the trace file and save the information for building Circos in text files. The information saved included IP addresses and TCP or UDP port information, and also TCP flags. These datasets have been used in many research works, e.g., to test the efficiency of methods for intrusion detection [MIT15].

The selected characteristics constitute a minimal set that is typically used to identify network traffic flows (except for the TCP flags and adding the protocol, which is implicit in this case). This was the main reason for choosing them. For most Circos produced in this study, the lines represent an association between two of those characteristics given by the information in a packet, flowing from one machine to another. Each line of the aforementioned text files contain the information of a single packet transmission.

The datasets produced in the laboratory are similar to the ones of the MIT. The traces were in the `tcpdump` format and their pre-processing was thus performed using Wireshark also. The metadata for these files was included in separate files. The minor difference to the MIT traces is that this dataset contains several traces of traffic generated by legitimate applications, since

its initial purpose was to study both P2P and attack related traffic.

The Circos included in the remaining of the chapter were obtained for legitimate HTTP related communications. For charts including the destination TCP port (see Figure 3.4), this visual representation emphasizes the client-server nature of the communications (many curves apparently flowing to a small number of points). Port 80, the default for HTTP servers, is the end of more curves than the remaining ones. The charts in this section show that the amount of data used to construct the chart can be overwhelming. Nonetheless, one must not forget that these charts are representing single packets, and not aggregated statistics such as bandwidth per time unit.

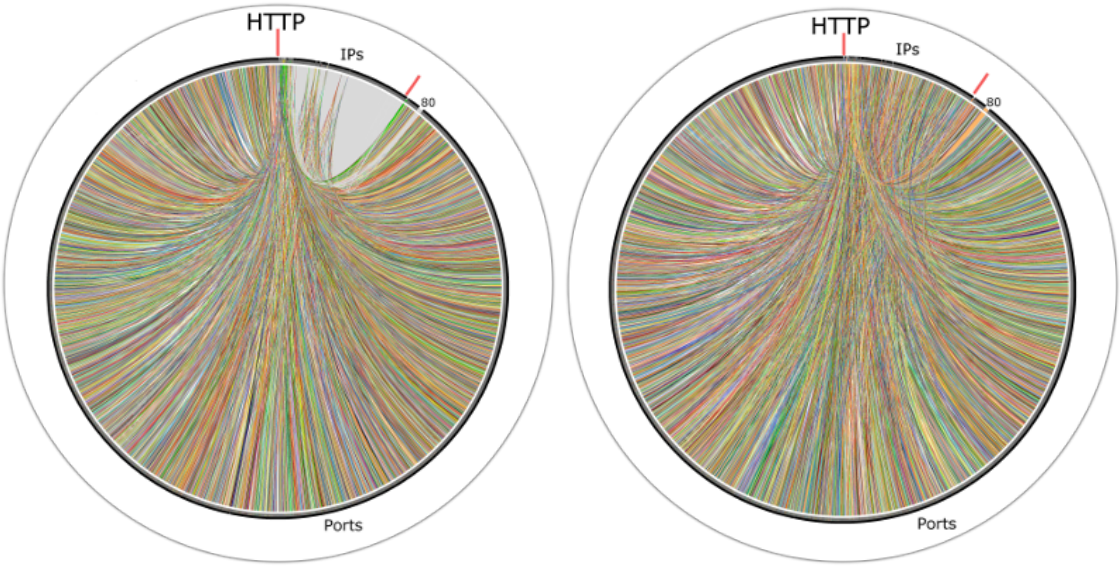

Figure 3.4: Circos representation obtained for destination port information from HTTP traffic.

The charts of Figure 3.5 contain an analogous representation to the previous ones, but focusing on the TCP source port. In this case, it is interesting to notice that both charts suggest randomness in the source port side, corroborating the normal functioning of TCP connections (in which the initiator chooses a random source port at the start of the communication). Nonetheless, while on the left side (concerning the destination IP address), there seems to be more convergence on the server, on the right side, the range of possible IP addresses is entirely covered by the several curves.

The relations between source and destination ports are depicted in Figure 3.6. In these communications, most packets have the number 80 in either the source or the destination port, originating a Circos where the majority of the curves end or start at this item, which takes more space in the circumference. The remaining surface of this Circos suggests a uniform distribution of the port numbers over the circumference.

Figure 3.7 contains a visual representation of the relationship between TCP interactions for

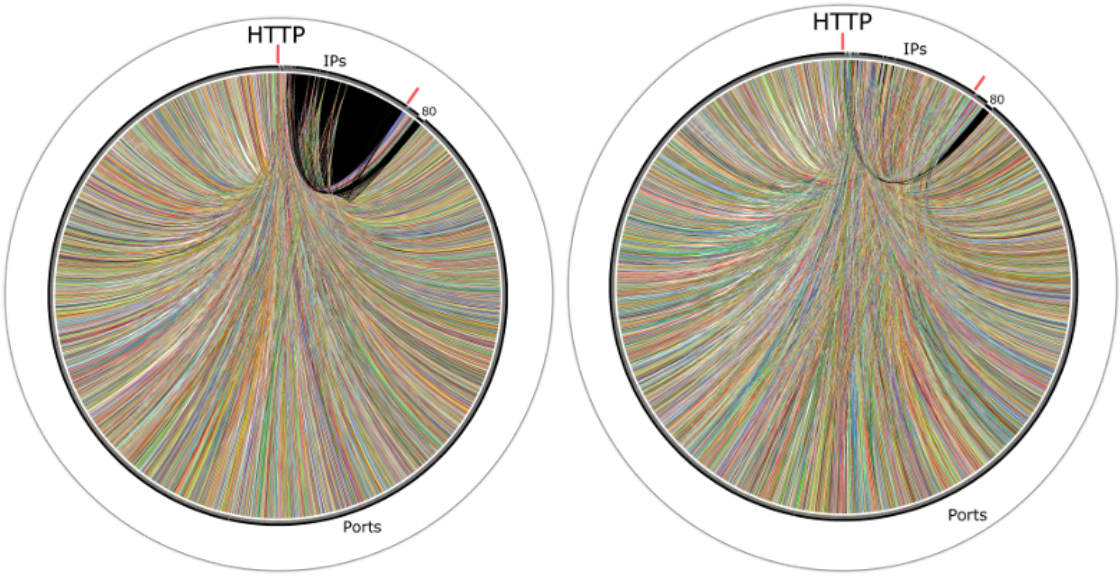

Figure 3.5: Circos representation obtained for source port information from HTTP traffic.

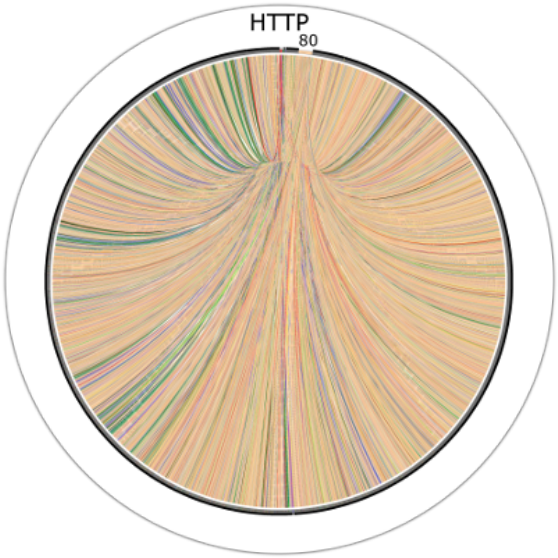

Figure 3.6: Circos representation obtained for destination and source port information from HTTP traffic.

HTTP traffic. This type of chart was produced by making a connection between the flag in the header of a TCP segment and the next. This representation was considered because it had the potential to emphasize abnormal behavior, e.g., due to Synchronize (SYN) floods or port scans. The chart included in the figure shows that, in this class of traffic, Acknowledges (ACKs) and SYNs are always followed by SYN-ACK; Finisheds (FINs) are typically followed by FINs also.

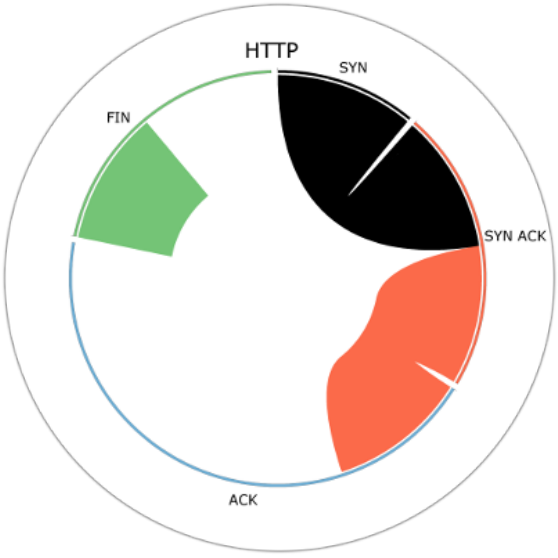

Figure 3.7: Circos representation obtained for TCP flag information from HTTP traffic.

3.4 Conclusions

This chapter presents the data visualization techniques used in the scope of this work as well as the datasets used to create the visualizations. Much of the research effort was placed on the technique known as Circos, on the adaptation of the software and respective inputs to the available data, and to the analysis of the resulting charts. Circos was chosen because of its flexibility, graphical expression and ability to produce beautiful representations. Part of the work described herein was also devoted to the definition of the traffic characteristics, and their combinations, that could be used to produce useful visualizations. Given the nature of the problem at hands and also the main objectives of this master's program, the set of characteristics that usually defines network flows in TCP/IP was selected: source and destination IP addresses; source and destination transport layer ports and protocol. Additionally, TCP flags were used to obtain some of the visualizations. Some Circos for legitimate traffic were already included to show how they look for the available data and characteristics and also to help explain some of their advantages and disadvantages when applied in this field.

The provenance of the data used to produce the data visualizations and the results discussed in chapter 5 was also discussed here. The datasets were constituted by traffic traces from the well-known MIT Lincoln experiments and from previous experiments on the laboratory where this work was performed. While the former contain traffic from classical attacks, the later contains packets generated by more recent network applications also. At this point, the goal was to better scrutinize the procedure to automatically obtain Circos from the traces, since at least 6 charts per trace were to be produced. The work performed in this phase paved the way to the implementation of the scripts for that purpose and for the assessment if the results could be used for traffic classification and attack identification.

Chapter 4

Method and Experimental Setup

4.1 Introduction

One of the most important aspects of this master's program concerns the methods and tools used for the constructions of the Circos representations. In order to be able to create these representations, it was necessary the use some scripting related technologies, as well as traffic analysis tools. The technologies and libraries used to create data processing scripts are explained in section 4.2. Section 4.3 then includes the description of the method used to analyze network traffic and also several charts for one of the attacks in the datasets: the *Satan* attack. Finally, the prototype of the developed script is briefly explained in section 4.4.

4.2 Technologies and Libraries

This project required the usage of several technologies and software. The three main technologies/software that were used were Wireshark, for the analysis, filtering and processing of network traffic, Python, to create scripts for handling and transforming the data, and the Circos software (which is development in Perl), to create the Circos representations.

Wireshark is a well-known network traffic analyzer, supporting a large panoply of network protocols from many layers of the Open System Interconnection (OSI) model. This tool allows examining the smallest details of network packets, as well as capturing traffic in real-time and processing traffic traces in, e.g., `tcpdump` format, among many other features. In this work, Wireshark was used for the interpretation of traffic traces, on the one hand, and for applying some filtering to the data on the other. By using such features, it was possible to eliminate irrelevant data to the project. The data that was necessary for producing charts was then stored in text files, a feature also provided by this tool. The files resulting from this step were later fed to the developed scripts.

The Python language was used to develop the scripts utilized for processing the data and also for automatizing part of the process for constructing charts. Python is a widely used language

nowadays, mostly because it is fast, easy to learn and because it is open source. Its documentation is also very useful and the writing style inherits many characteristics from other well-known languages. It is also known for being suitable for scripting tasks. To aid in the development of the scripts, the Netbeans Integrated Development Environment (IDE) with the Python4NetBeans802 plugin was installed (natively, Netbeans does not support Python). The developed scripts were for transforming the data outputted by Wireshark into suitable inputs for the Circos software and also for taking the human factor out of the experiments conducted afterwards (for testing the ability to classify traffic with these graphical representations).

Along this project it was also necessary to utilize Perl, since most of the Circos software is written in that programming language. Regarding the specific details of the software, the Circos website contains some tutorials to understand the software and to produce charts. Nonetheless, the software is designed to handle less data than the one produced in some of the experiments of this work, and some minor changes to the source code of Circos was required to produce the charts.

4.3 Method for Analyzing of Network Traffic

At the initial stages of this master's program, a significant amount of time was dedicated to analyzing traffic traces from the datasets. The Wireshark tool was specially useful at this part of the work, via which access to the information of interest was possible. Wireshark shows the contents of the fields for several protocols of IP packets, namely the source and destination IP addresses and port numbers, the name of the protocol and other useful data (which was also relevant for the filtering phase, as explained below).

For the sake of impartiality, the adopted datasets were from two different previous projects. Because of that, much data of the traces had to be filtered out, as it was irrelevant. For example, all frames with no IP packets, namely Logical Link Control (LLC) frames, were removed. Moreover, IP packets containing layer 3 protocols other than TCP or UDP (i.e., without port number information) were discarded almost immediately also. At the end, and since the characteristics that were identified for this first incursion using Circos, the data that made it to the output text files was constituted by IP addresses and TCP or UDP port numbers. Each line of the files corresponds to a single packet, with the several values separated by spaces.

It was also decided to explore the relationship between types of TCP segments (e.g., FIN, SYN, etc.). For that specific purpose, the resulting files were only constituted by TCP related data, namely a label for the flag and a sequential number. Packets containing data from all other protocols were removed. In order to show the Circos obtained for this specific data, and given

that the usefulness of this representation may not be obvious at a first glance, several charts for the *Satan* vulnerability scanner are included below, including the charts obtained for the TCP flags.

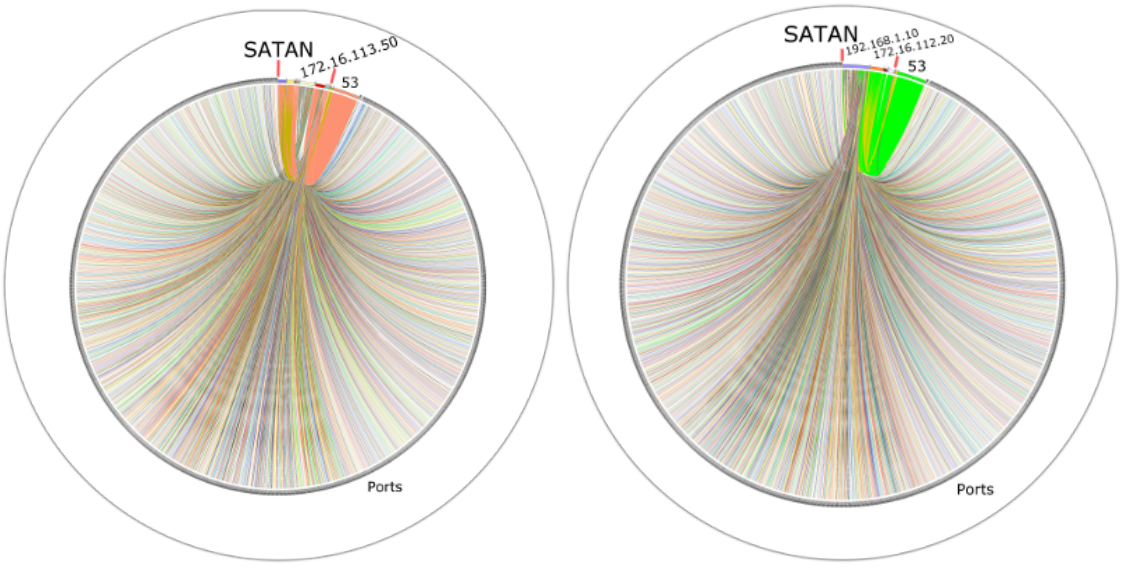

Figure 4.1: Circos representation obtained for destination port information from a trace with a portscan.

Figure 4.1 contains two charts: the chart on the left is a representation obtained using the destination IP addresses and destination port numbers, while the one on the right was obtained using source IP addresses and the destination port numbers. Port 53 stands out amongst all possible port numbers in the first Circos. It is also noticeable that the IP `172.16.113.50` has more connections, although there are a few other IP addresses with multiple connections also. On the right side, it is possible to see that port 53 stands out comparatively to other ports also. The IPs `192.168.1.10` and `172.16.112.20`, are the two IPs that stand out in this representation. The IP the `172.16.112.20` contains almost all connections with port 53 and the IP `192.168.1.10` have some connections to destination port 53. The configurations of the connections and the colors clearly suggest that the IP `192.168.1.10` is the source of the attack, sending packets to a wide spectrum of TCP ports. The target is the system with the IP `172.16.113.50`.

Figure 4.2 contains representations that are similar to the ones in Figure 4.1, but for the source port information: in the left side, the destination IP addresses are plotted against the source port numbers; while the left sides concerns the source IP addresses. The spreading character of a port scan is superbly emphasized in these charts. Ports 123, 53 and 1034 are highlighted in both charts (this tends to happen as communications are bidirectional), and the systems with IPs `192.168.1.10` and `172.16.112.20` seem to be the ones starting the majority of the connections (concluded from the chart on the right side). These connections seem to be coming from ports 53 and 1034. Based on these observations, it can be said that these IPs are probably the source of the attack.

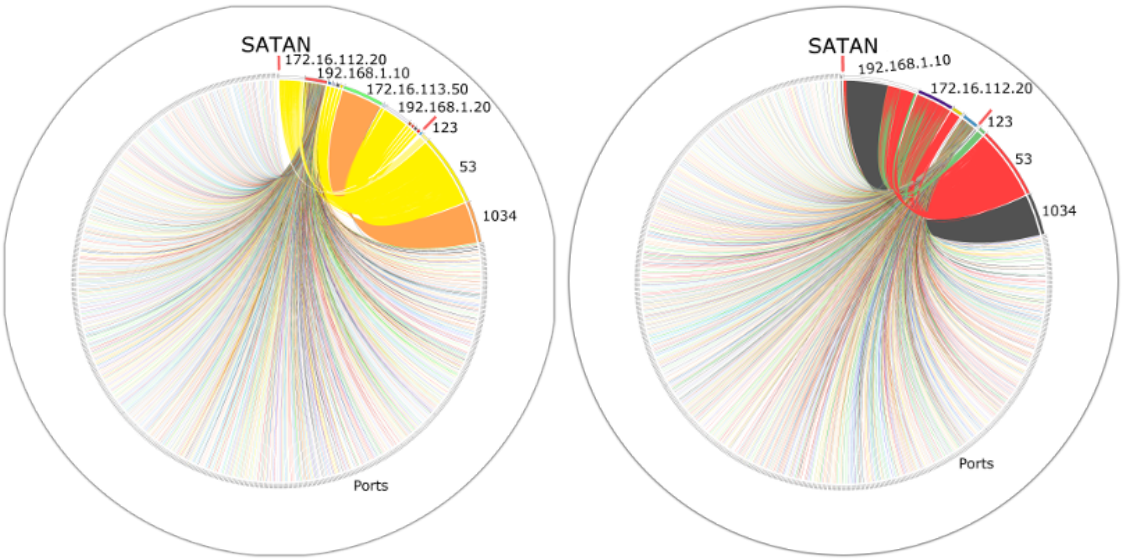

Figure 4.2: Circos representation obtained for the source port information from a trace with a portscan.

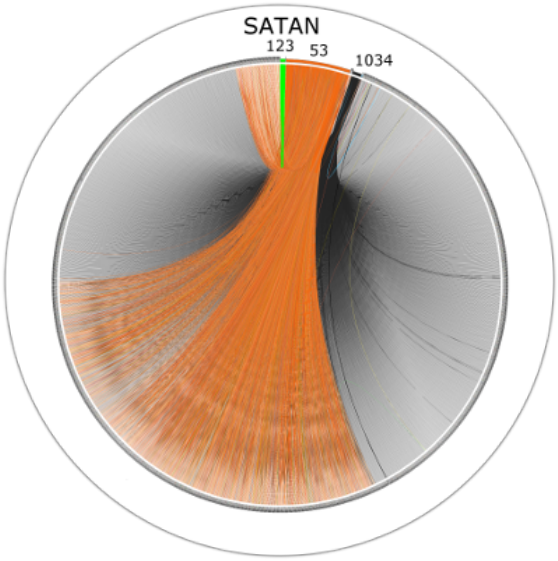

Figure 4.3: Circos representation obtained for destination and source port information from a trace with a portscan.

The Circos in Figure 4.3 emphasizes the relation between source and destination ports of the TCP segments involved in the portscan. The chart highlight the prominent role of ports 53, 123 and 1034. Port 53 is the one with more connections, followed by port 1034 and lastly port 123. The aspect of the representation suggests that these ports are the source of the scan (i.e., the ports from which the probes are sent).

In the last chart of this chapter, depicted in Figure 4.4, it can be observed that some connections were not completed. This is noticeable because the area concerning the SYN flag has halfway connections.

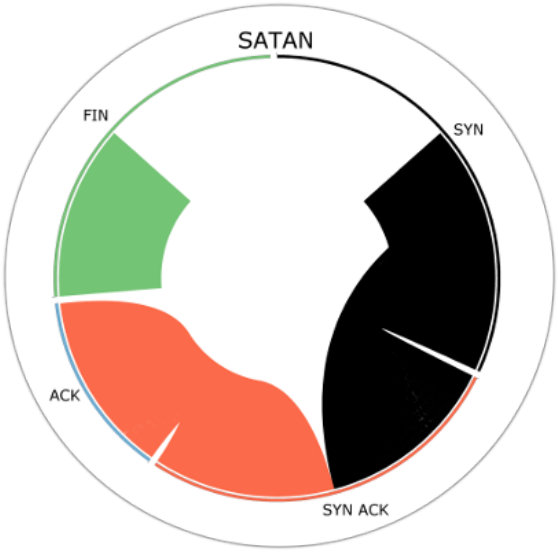

Figure 4.4: Circos representation obtained for TCP flag information from portscan traffic.

4.4 Scripts Prototype

This section describes, in a brief manner, all of the conceptual steps of the most important scripts implemented along this work. As previously mentioned, these scripts were developed in Python. The ones developed for processing the outputs of Wireshark are constituted by six main steps and require two inputs to function properly: the text file with the data for a given traffic trace, and a file containing the references of the colors that the Circos software is able to recognize.

The scripts start by reading and parsing the two supplied files, and proceed with dividing the data by port numbers and IP addresses (depending on the charts that are to be performed). Then, they save all IP addresses or ports numbers occurring in the data file without repetitions, thus obtaining unique entries for each one. With this data stored, the script can then count the number of times that each port number or IP address appears in the main input file,. The number resulting from this counting allows knowing the number of connections for all IP addresses or ports. As this count is done for all data, at the end of this process, the script knows how many connections will be required for the represented IP addresses and port numbers. Such connections need to be colored. The scripts choose the colors randomly from the entire set of possible colors in the second input file, for each one of the represented connections.

In the final part, the scripts write three different files. The first file is called a `karyotype`, which is the default name used by the Circos software for information on labels (in this case, it is whether IP, ports or TCP flags). This file contains different information on labels, such as the name, number zero (the count of connections starts at zero and increases with each connection that the label contains), number of occurrences of the represented IP or port (the maximum

number of connections that it contains), etc. The second file, called `segdup`, has information about the connections. The information presented in the file is the label of the connection, the source of the connection (a number that increments for every connection made by that source label), the destination label, and the target connection number (such as the source connection number, it increments at every connection concerning that label). Lastly, the third file contains he label names and the corresponding colors for each of them.

The process is slightly different for TCP related information. In this case, the developed scripts start with counting the number of occurrences for each flag of the TCP protocol in the trace under analysis. For every SYN segment found on the file, a search for the corresponding SYN ACK segment is carried out, as well as a search for the corresponding ACK segment after that. If a SYN segment does not contain a correspondent SYN ACK segment, the connection is halfway, representing a SYN request unanswered (this situation happens frequently during some attacks). The process of saving the files is similar to ones described above.

4.5 Conclusions

This chapter presented some of the most important resources used in the course of this work, apart from the main datasets. The several steps, tools and technologies involved in the construction of the Circos representations were discussed to highlight the overall method that was followed to achieve such purpose. In this process, Wireshark, Python, Perl and Circos software played major roles. Wireshark was used during initial analysis of the traffic traces and for filtering out unwanted information; python scripts were developed to process the filtered content, outputted by Wireshark, and produce the inputs to the Circos software. Different scripts were developed for port and IP and for TCP related information and some tweaks to the source code of the Circos software were required for it to handle the amount of data of some of the traces.

The developed scripts supported the construction of a large number of representations for all traffic traces in the datasets and also conducting the experiments discussed in the next chapter. For example, they enabled producing the charts for half of the dataset in a completely automated and blind manner. The resulting Circos had no explicit mention to the trace that generated them, so that the effectiveness of classification of traffic resorting to these representations could be measured, as discussed in the following chapters.

Chapter 5

Analysis of the Results

5.1 Introduction

One of the most important outcomes of this work consisted in the elaboration of a considerable amount of data visualizations for different traffic classes and attacks. This chapter reflects part of the effort made to accomplish that objective. As such, it contains and discusses many Circos obtained for the traces of the utilized datasets. Notice that, in order to maintain a certain equilibrium in terms number of pages for each one of the chapters, several charts were already included and briefly described before. This chapter is structured as follows. Section 5.2 contains and discusses charts obtained for legitimate traffic. Section 5.3 is similar to the previous one, but for known attacks related traffic. Section 5.4 is devoted to the experiment of finding out if it was possible to classify traffic or identify attacks only resorting to human analysis of the charts and, finally, section 5.5 discusses the results obtained.

5.2 Data Visualizations of Legitimate Network Traffic

In this section presents many Circos concerning legitimate network traffic. In this work, four different traces of traffic concerning legitimate applications were considered: VoIP, HTTP, P2P and SSH. Notice that the VoIP related charts were already presented in section 3.2 and that HTTP Circos are in section 3.3. As such, this section contains data visualizations for the remaining classes only, namely P2P and SSH traffic only.

The Circos included in Figure 5.1 were obtained for traces containing SSH traffic. SSH connections involve two machines. Both charts exhibit the connections between destination ports and IPs addresses. On the left side of the figure, the Circos shows the relationship between the destination port numbers and the destination IP addresses, while on the right side, the relationship between the destination port numbers and the source IP addresses is depicted. Both charts show that the ports 22 and 57826 are the most used, although it is also possible to find connections on port 53869. The IP addresses `10.0.2.178` and `10.0.3.108` appear in both charts, as both source and destination of the communication.

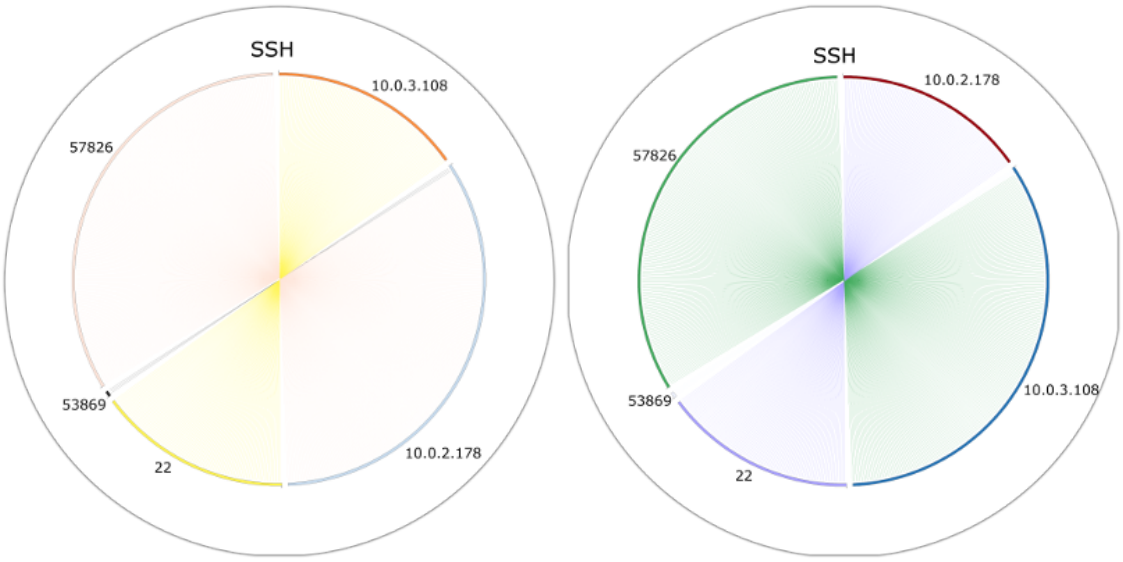

Figure 5.1: Circos representation obtained for destination port information from SSH traffic.

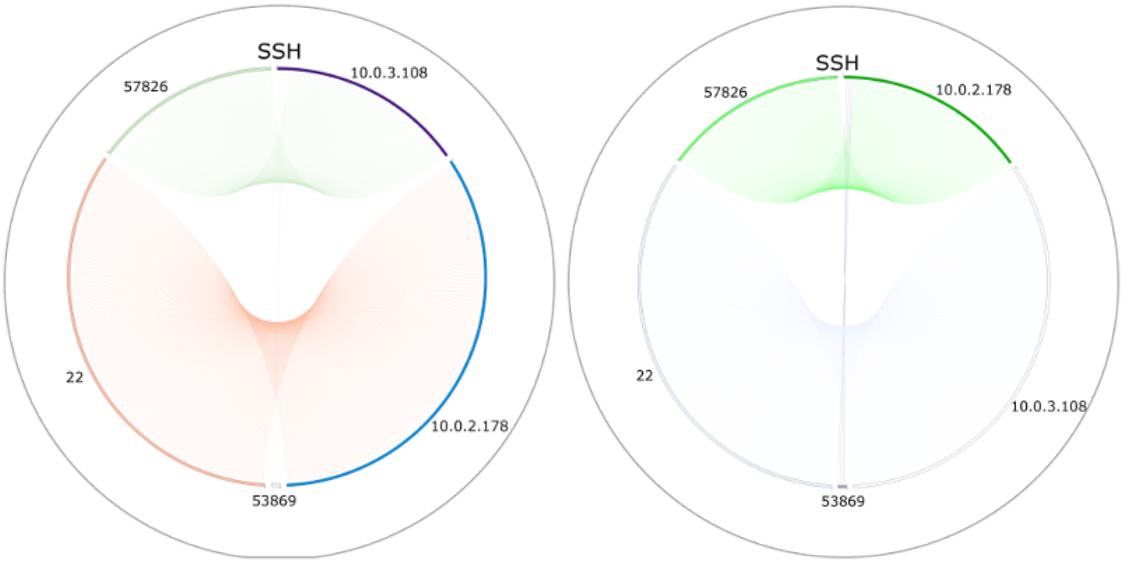

Figure 5.2: Circos representation obtained for source port information from SSH traffic.

The charts in Figure 5.2 were also obtained for SSH traffic, but this time for source port instead of the destination port information. The left chart illustrates relations between source port numbers and destination IP addresses, while the right chart contains a similar representation for the source IP addresses. As in the first two Circos for SSH, ports 22 and 57826 dominate the majority of the connections. The IPs emphasized in these graphs are the same, but the representations are different and pleasing to the eyes.

The last chart concerning SSH traffic is presented in Figure 5.3, in which the relations between source and destination port numbers is illustrated. This type of chart, though very simple, is important to show how a single well behaved connection for a client-server application should work.

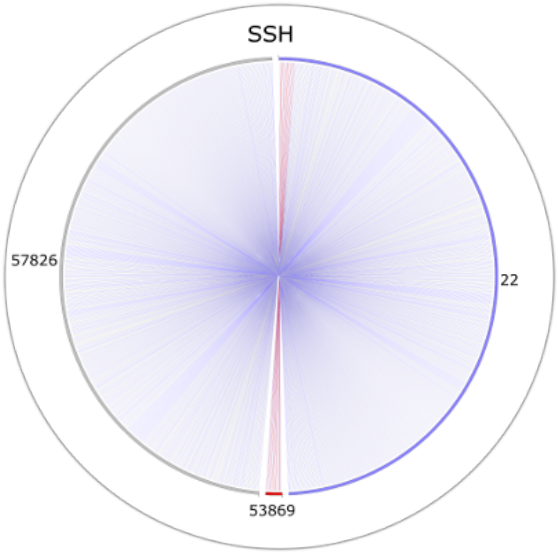

Figure 5.3: Circos representation obtained for destination and source port information from SSH traffic.

The next set of data visualizations concerns P2P file-sharing traffic. Similarly to what was done before, the following two charts (included in Figure 5.4) emphasize the relations between IP addresses and destination port numbers only. The chart on the left is for destination IP addresses, while the chart on the right was obtained for source IP related information. These charts are fundamentally different from those included before. The number of port numbers and IP addresses involved are such that the charts are completely full of connections with different colors. It can be said that the representation seems quite random. The Circos on the right side is divided into two regions, clearly emphasizing that there seems to be only two IPs in the local network connecting to all other ports.

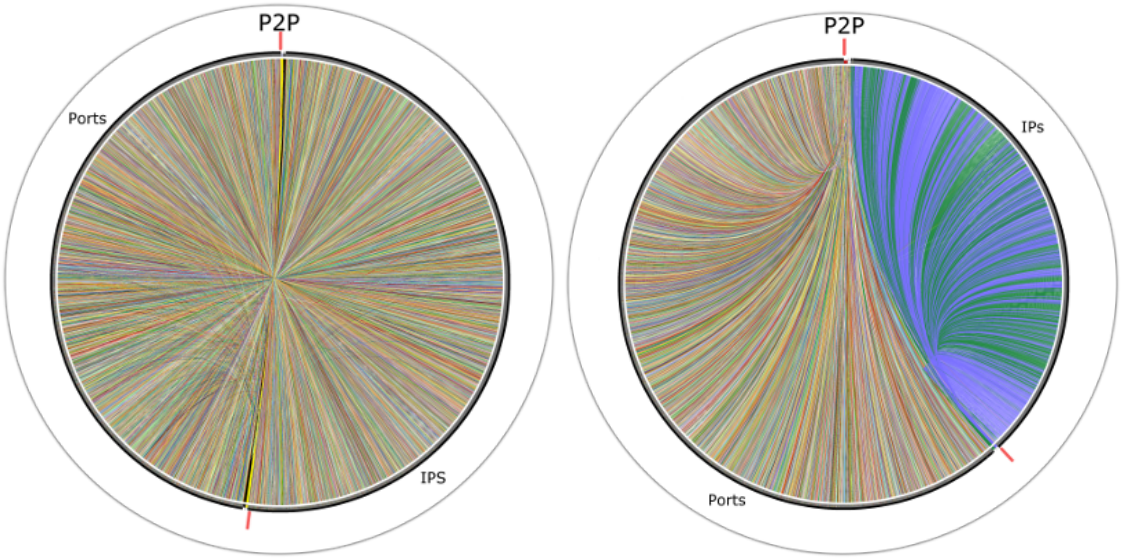

Figure 5.4: Circos representation obtained for destination port information from P2P traffic.

Relations between the destination and source port numbers for P2P traffic are represented in

the chart of Figure 5.5. In these communications, most packets have the ports 19782 and 44424 either as a source or as a destination port number, leading to a vortex effect towards these numbers in the representation.

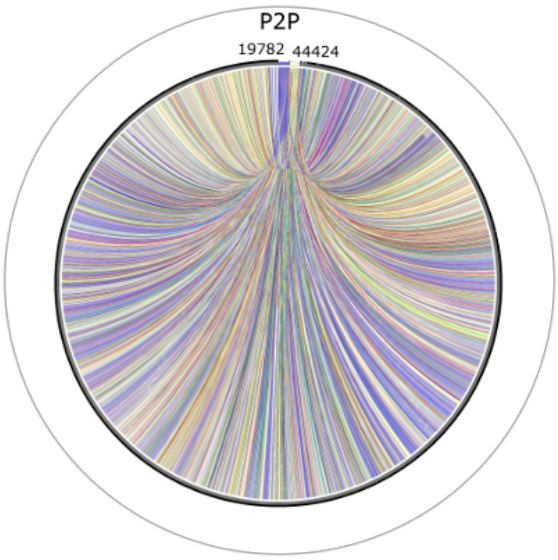

Figure 5.5: Circos representation obtained for destination and source port information from P2P traffic.

5.3 Data Visualization of Known Attacks

In this section, the charts concerning traffic that contains known network based attacks will be presented. Part of the motivation for all this work was precisely in obtaining such representations, to see if they would eventually emphasize useful characteristics of network intrusions and malicious activities. The MIT dataset was used to produce the Circos in this section, namely the subset of traces with the attacks labeled as *Neptune*, *Back*, *NMAP*, *Dictionary* and *Satan*. Notice that *Satan* (or the representations obtained) was already subject of discussion in section 4.3.

Both Circos in Figure 5.6 concern the destination port information of a trace containing traffic from the *Back* attack. *Back* is a DoS attack against the web server, and works by repeatedly and quickly requesting pages at a rate that may be superior to the one supported by the victim, ending up slowing down the server and eventually affecting its ability to respond to legitimate requests. The traffic under analysis concerns a service with a client-server architecture and, as such, the charts illustrate the situation where many connections are converging to a small number of ends. Both charts highlight the popularity of port 80, which is the one where the HTTP service is typically listening. The chart on the left shows the relationships between destination IP addresses and source port numbers, while an analogous representation for the source IP addresses is on the right. Both charts let one know that the victim of the attack is the

system with IP `172.16.114.50`, which is receiving connections from several sources.

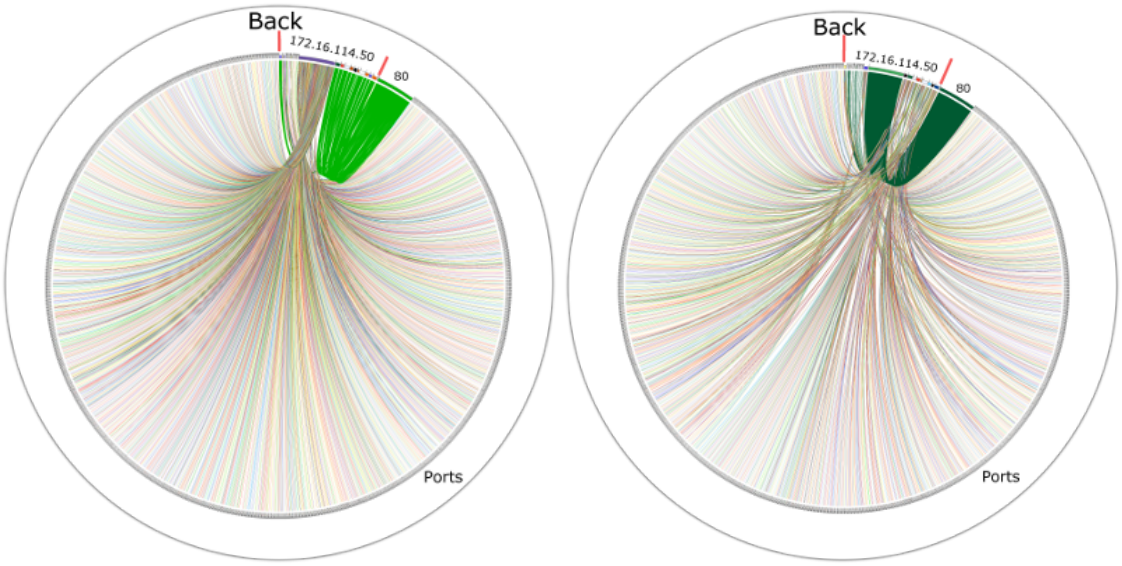

Figure 5.6: Circos representation obtained for source port information from a DoS attack against a webserver.

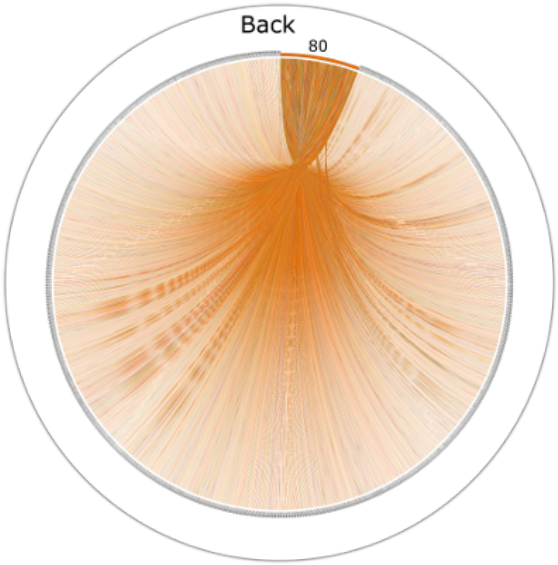

Figure 5.7: Circos representation obtained for destination and source port information from a DoS attack against a webserver.

The relation between port numbers during the *Back* attack is illustrated in Figure 5.7. The vortex effect towards port 80 is depicted in this case also. Nonetheless, it seems to be a normal characteristic of service using the client-server architecture.

The following set of charts concern a *Dictionary* attack against a router running Simple Network Management Protocol (SNMP). The objective of the incursion it to obtain access to the router by repeatedly trying to guess the authentication credentials (e.g., the username and password or just the latter). Figure 5.8 presents the Circos obtained for the destination port numbers, with the chart on the left establishing their relation with the destination IP addresses; and the

one on the right showing their relationships with the source IP addresses. Two port number concentrate numerous connections: port 161 has most of the connections, followed by port 53. Connections on port 161 seem to be directed towards the computer with the IP `192.168.1.1`, while the communications to port 53 are arriving at IP `172.16.112.20`.

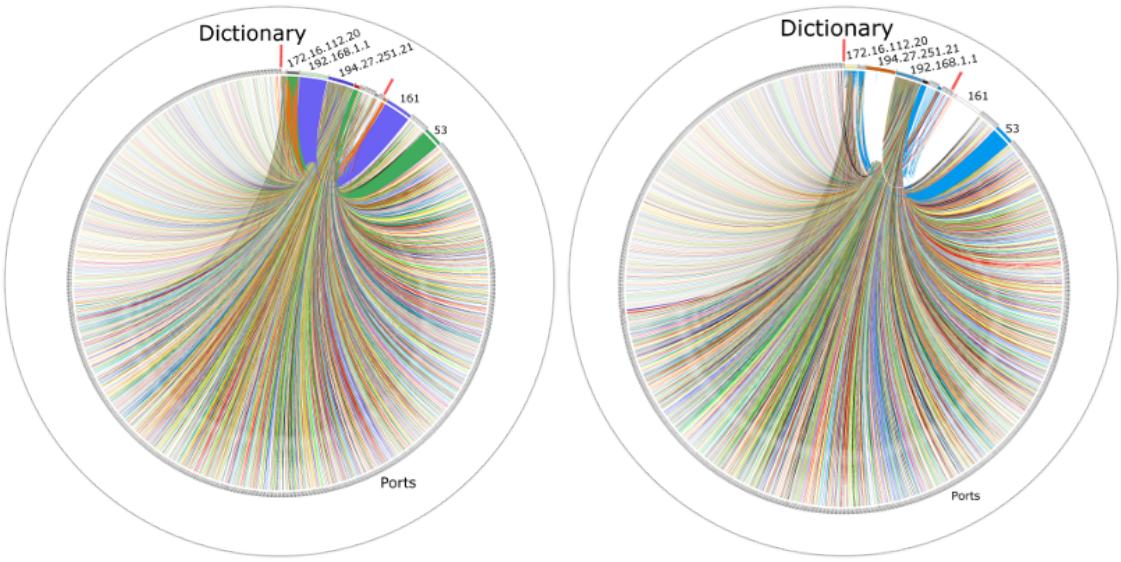

Figure 5.8: Circos representation obtained for destination port information from a trace where a *Dictionary* attack is happening on a router with SNMP.

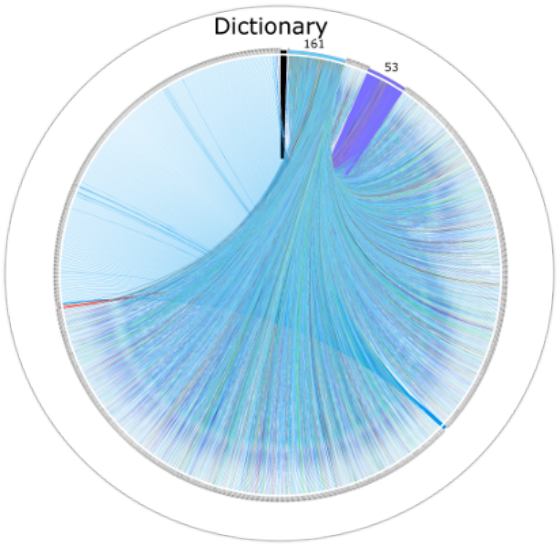

Figure 5.9: Circos representation obtained for destination and source port information from a trace where a *Dictionary* attack is happening on a router with SNMP.

The last chart concerning traffic with a *Dictionary* based attack is presented in Figure 5.9. The depicted relationships between port numbers show that the connections towards port 161 (SNMP) come from many different source ports, typical of attempts coming from a single computer starting (too) many connections in a short amount of time. In the previous charts, it is possible to identify the source as being IP `194.27.251.21`.

The following set of five charts concerns the attack known as *Neptune*. It consists in a SYN flood DoS on one or more ports. Because this attack may result in more asymmetrical connections (non-acknowledged, half-started, etc.), it was decided to include the entire set of charts produced for this trace. The first two Circos, included in Figure 5.10, establish the relationships between destination port numbers and destination IP addresses, on the left side, and between destination port numbers and source IP addresses, on the right. The target of the attack is clearly highlighted on the Circos of the left: `172.16.112.50`. The victim received segments in many ports, probably indicating that the attack was not targeted to a single service only.

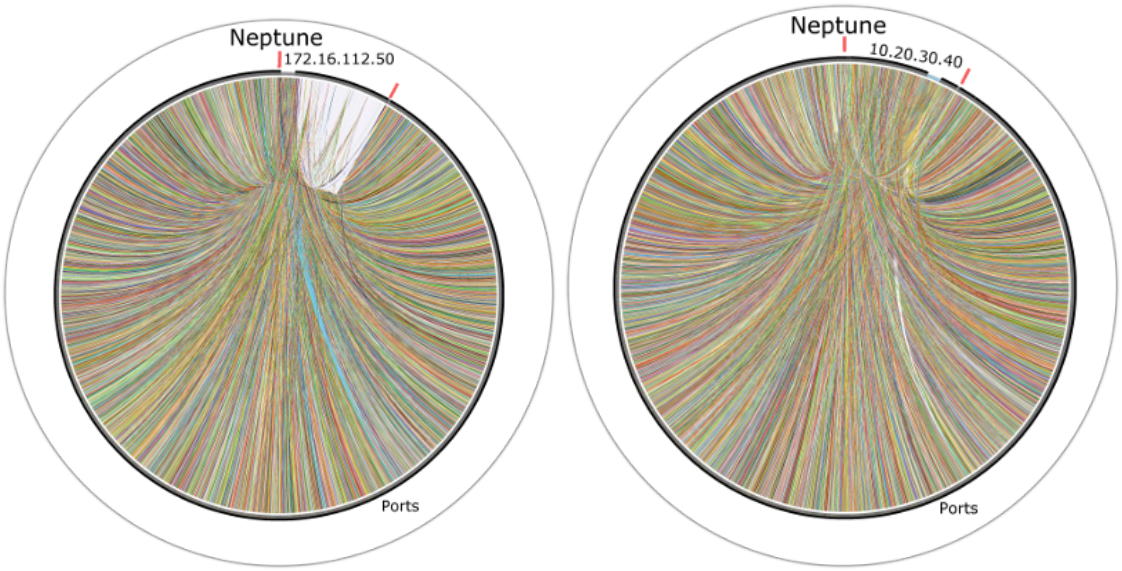

Figure 5.10: Circos representation obtained for destination port information from a SYN flood attack.

The next pair of Circos, in Figure 5.11, plot source port information against destination and source IP related information, respectively. This is one the cases where the charts for source ports are slightly different from the ones of the destination ports, discussed before, precisely because of the effects of the attack. In the previous charts the number of source IP addresses (chart on the right of Figure 5.10) seems higher than on the these ones due to two facts: the source IPs were being spoofed and not all of the packets were answered back.

The chart illustrating the communications between source and destination port number is presented in Figure 5.12. This chart is one of the most random charts included in the dissertation. It would be similar to one generated from random source and endpoints, though there is one port emphasized in the top.

For this attack, it was also decided to include the Circos representing the interactions given by TCP flags in the connections (see Figure 5.13). Since the *Neptune* attack is based on SYN flood, it is possible to observe the dominance of such segments in the chart, and their lack of connection with ACK or SYN/ACK segments demonstrates the existence of a very significant

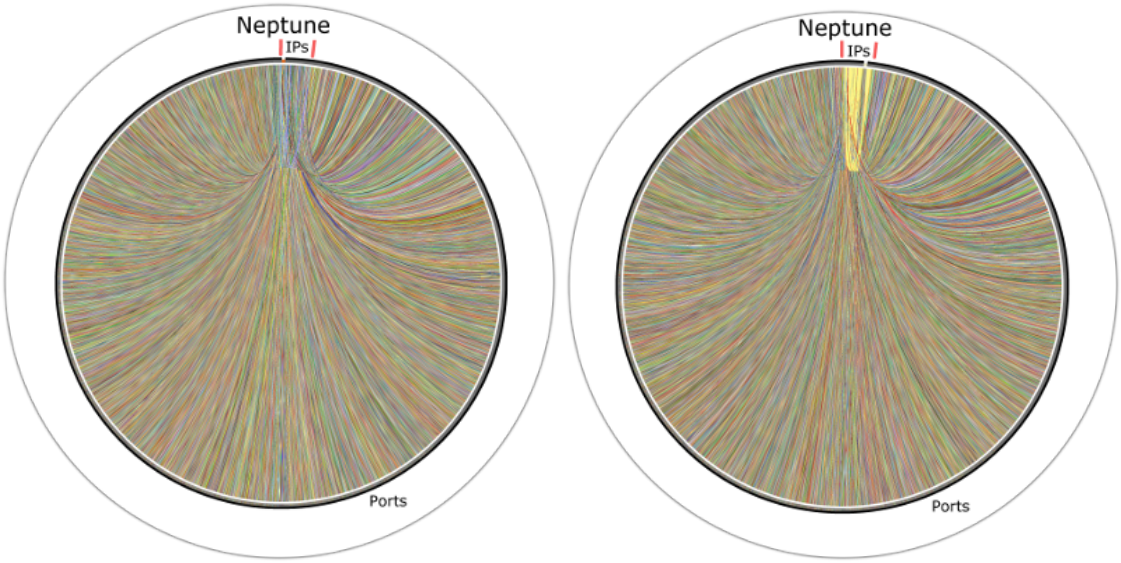

Figure 5.11: Circos representation obtained for source port information from a SYN flood attack.

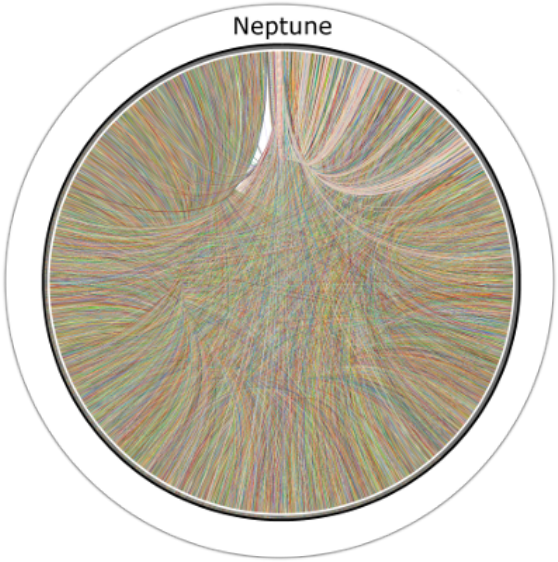

Figure 5.12: Circos representation obtained for destination and source port information from a SYN flood attack.

number of half-way connections, typical of such attacks.

The final set of charts of this section concerns traffic generated with *NMAP*. *NMAP* is a general-purpose tool for performing network scans. The charts in Figure 5.14 both plot destination port information against IP addresses: the one on the left was obtained for source IP addresses, while the one on the right concerns destination IP information. The Circos on the left suggests that a single or a small number of victims are receiving traffic directed towards an strangely high number of ports, a signature of port-scans. On the right, it is possible to see that there is an IP address that aggregates more connections, which is probably the source of the scan (to which the answers flow).

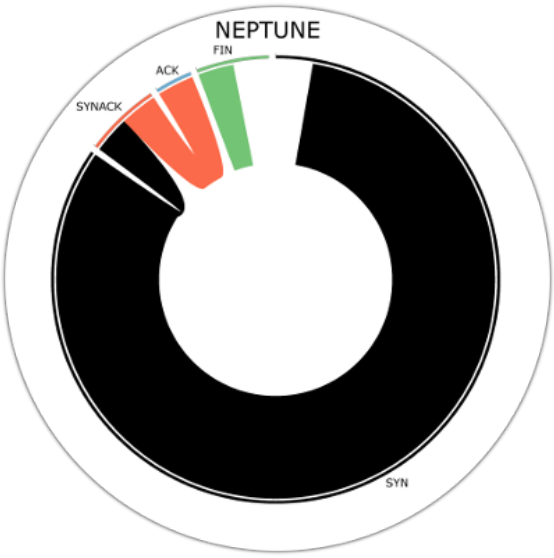

Figure 5.13: Circos representation obtained for TCP flag information from syn flood attack.

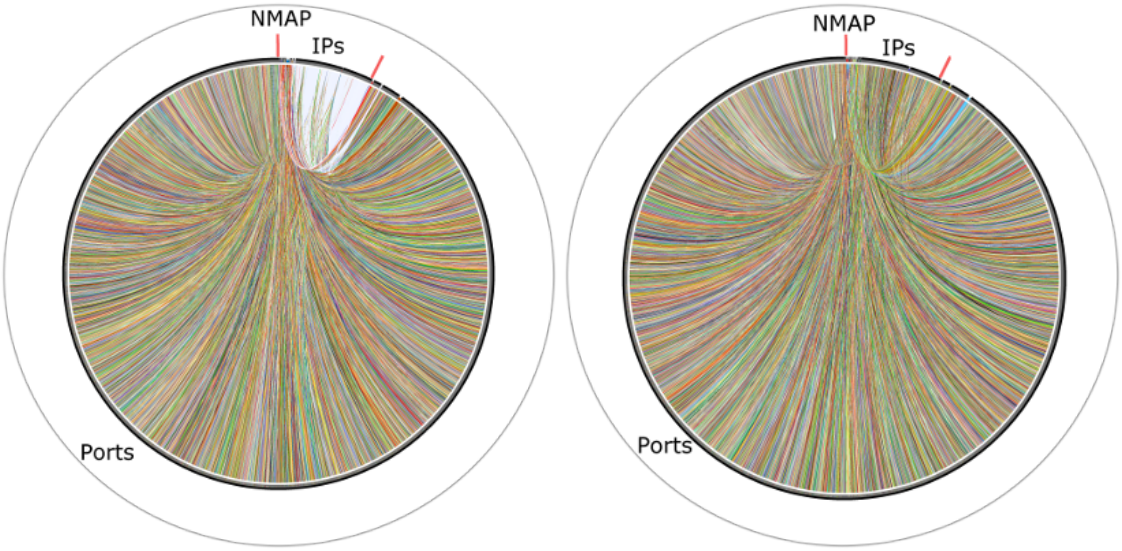

Figure 5.14: Circos representation obtained for destination port information generated by a scan tool.

The relations between source and destination ports for *NMAP* are depicted in Figure 5.15. From their observation it is possible to conclude that most connections converge to a set of ports organized at the top (vortex effect). From the chart colors it is possible to conclude that there are ports accumulating more connections.

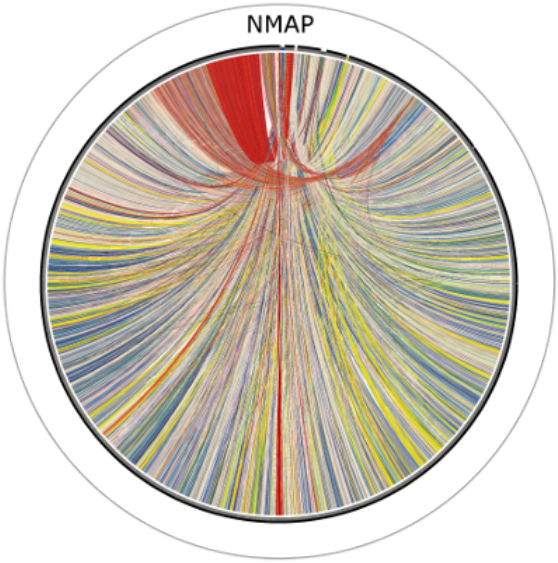

Figure 5.15: Circos representation obtained for destination and source port information generated by a scan tool.

5.4 Classification via Comparison of Data Visualizations

As previously hinted, aiming to quantify the usefulness of the representations produced along this work, a simple experiment was set up. Beyond the charts presented up to this point in the dissertation, several other sets of charts were produced and analyzed. These charts were coming from traces with traffic of which the ground truth was known, but they were generated by an automatic script that, apart from grouping them in directories and subdirectories, does not provide explicit hints from their provenience. The name of the files and subdirectories were random, but each set of charts in a given subdirectory was from a single trace only. The charts were then humanly analyzed and classified taking into account the similarities with previously observed Circos and details. Charts for all previously mentioned attacks and protocols will be presented here.

The Circos included in Figures 5.16 and 5.17 exhibit a very distinctive characteristic. From their look, it is possible to state that they concern VoIP traffic.

The three charts in Figure 5.18 and 5.19 are highly similar to the ones produced for SSH. Beyond these visual similarities, the number of connections that use port 22 (default port used in SSH protocol) let one easy conclude that they belong to SSH traffic. Other client-server simple (in terms of connections) service like this one can be easily identified via the data representation technique used in this work.

By the characteristics presented by the chart on the left of Figure 5.20, and given that the chart is divided into two parts, it is possible to state that these are charts were created from

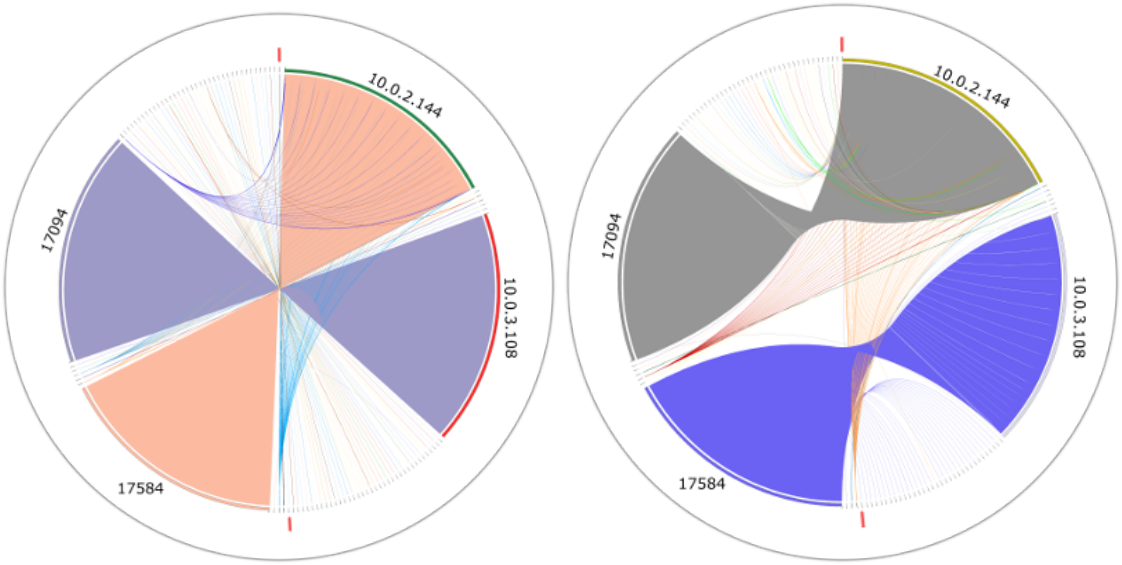

Figure 5.16: Circos representation obtained for destination port information for unknown traffic (later classified as VoIP traffic).

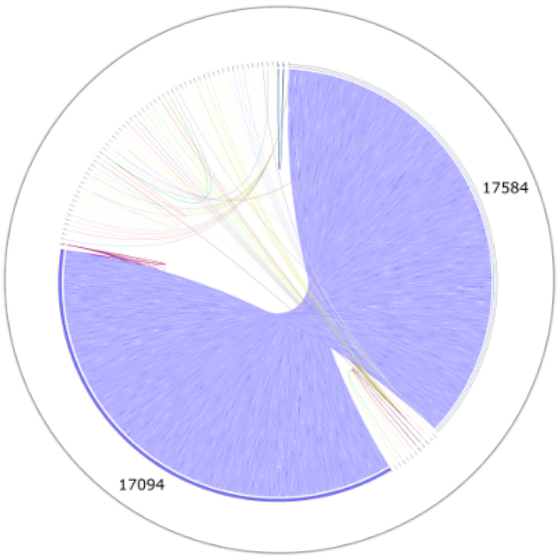

Figure 5.17: Circos representation obtained for destination and source port information for unknown traffic (later classified as VoIP traffic).

P2P traffic. The relations between source and destination ports in Figure 5.21 show the typical randomness created by the numerous connections of P2P traffic.

All of the data representations included in Figure 5.22 and 5.23 highlight port 80. Port 80 is the typical destiny of IP packets carrying the HTTP protocol, but also of the fraudulent packets of the *Back* attack, targeting the associated server. These charts could be coming from either HTTP or from the *Back* attack. However, in this case, they resemble the ones obtained for *Back* closer, from which the classification is issued. Most lines are connecting port 80 with a single IP address on the right chart (the web server) and the chart on the left shows that the connections are coming from a wide spectrum of source TCP port numbers.

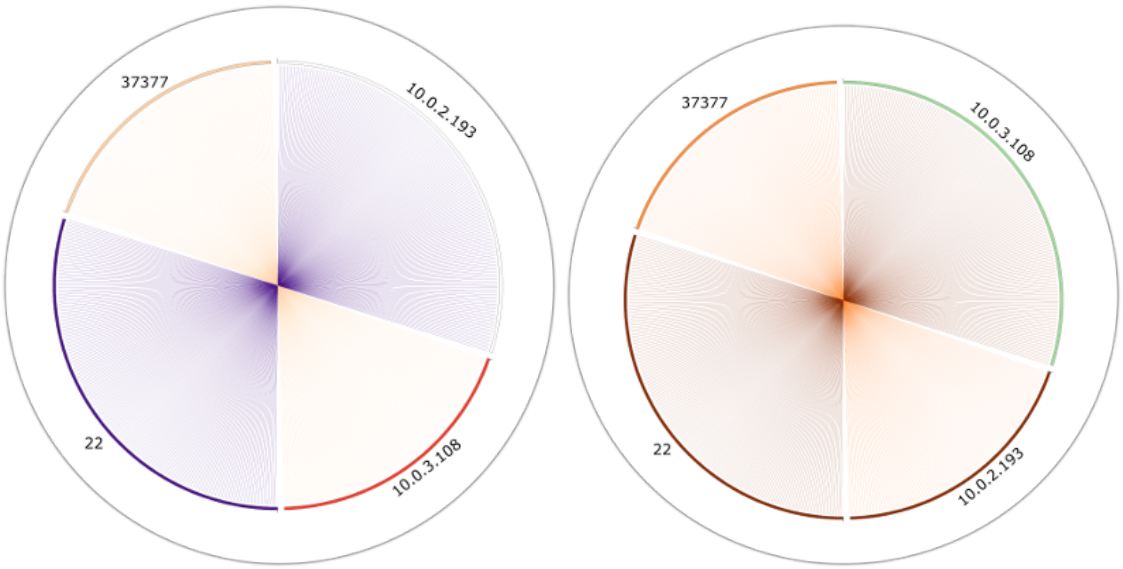

Figure 5.18: Circos representation obtained for source port information for unknown traffic (later classified as SSH traffic).

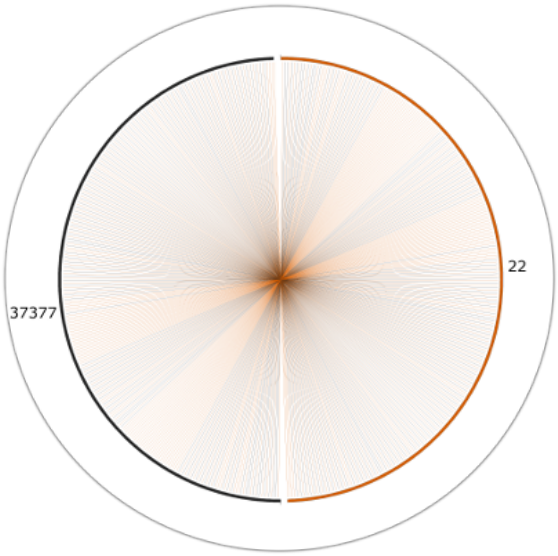

Figure 5.19: Circos representation obtained for destination and source port information for unknown traffic (later classified as SSH traffic).

The Circos included in Figures 5.24, 5.25 and 5.26 were classified as belonging to a *Neptune* attack, mostly due to the chart depicting TCP flag information. Notice that the remaining charts also provide some clues for the classification, but since this attack is probably more modest than the one used to construct the Circos discussed earlier, it was not that easy to get to that conclusion. More legitimate traffic is probably mixed up with the attack, diluting its effects. It was the number of unanswered SYNs that led to this deduction.

The set of Circos included in Figures 5.27 and 5.28 all point out to an existence of an HTTP server. As such, they could either be generated from legitimate HTTP traces or from attacks targeting this specific service. Since there are several IP addresses involved in the communi-

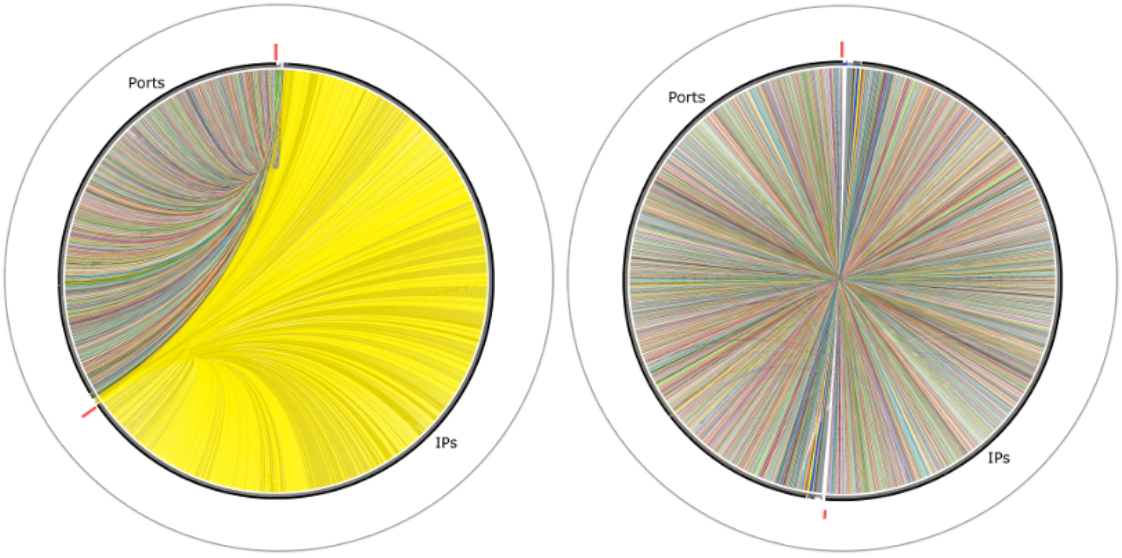

Figure 5.20: Circos representation obtained for source port information for unknown traffic (later classified as P2P traffic).

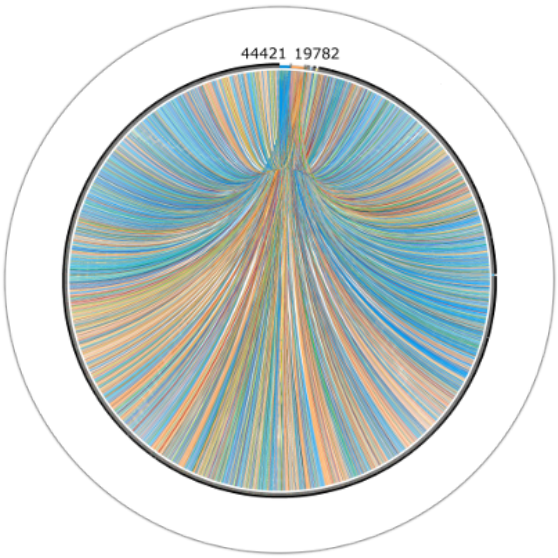

Figure 5.21: Circos representation obtained for destination and source port information for unknown traffic (later classified as P2P traffic).

cations and no further evidenced of malicious activities (e.g., an highlighted IP address), the provenience of these charts was said to be of legitimate HTTP traces.

The group of Circos, in which the ones depicted in Figures 5.29 and 5.30, are included, also suggest that the respective traffic trace is from legitimate HTTP traffic. Unfortunately, this corresponds to a failure in at least one of the classifications, since it was known that there was just only trace containing legitimate HTTP traffic in the unknown sets. Most probably, the attack contained in one of the two traces discussed just now is weak and statistically diluted in the remaining part of the traffic, or that the chosen representations used herein are not able to fully capture and express its behavior.

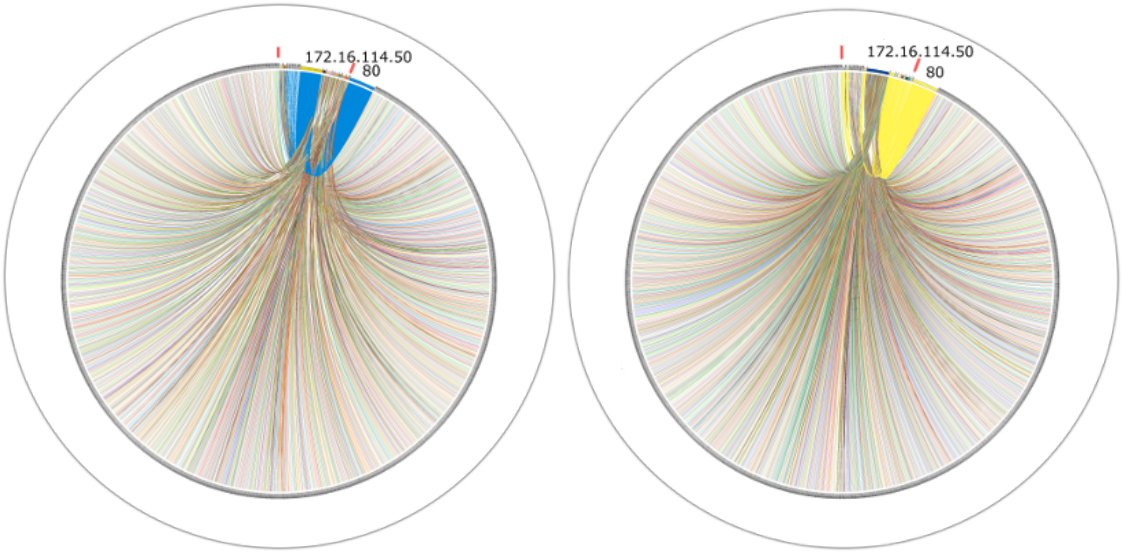

Figure 5.22: Circos representation obtained for destination port information for unknown traffic (later classified as *Back* attack traffic).

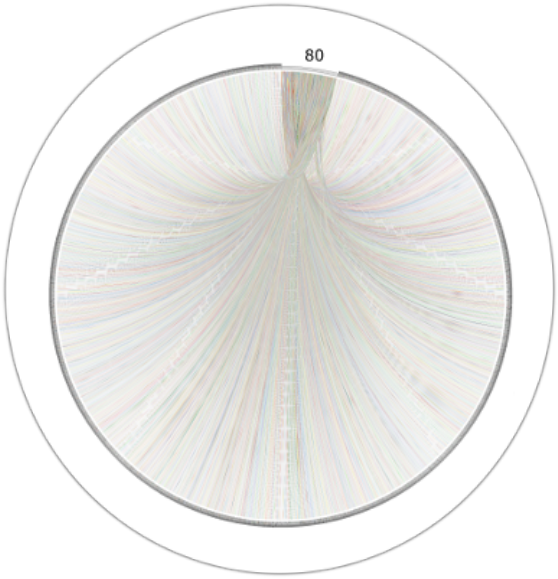

Figure 5.23: Circos representation obtained for destination and source port information for unknown traffic (later classified as *Back* attack traffic).

Even though the Circos in Figure 5.32 seems to suggest the existence of an HTTP server in the trace underlying this group of charts, it is possible to see that the charts in Figure 5.31 are visually similar to the ones included to the *NMAP* portscan in the previous section. Notice that it is possible the the traces contain HTTP traffic also, as the MIT datasets contain a mixture of legitimate and attack related traffic.

The last set of Circos presented in this dissertation are included in Figures 5.34 and 5.33. In these charts, specially in the last one, port 23 is highlighted. This port is where telnet usually runs and given the absurd amount of connections towards that, coming from all over the range of TCP ports, it was concluded that this was due to a *Dictionary* attack on the telnet

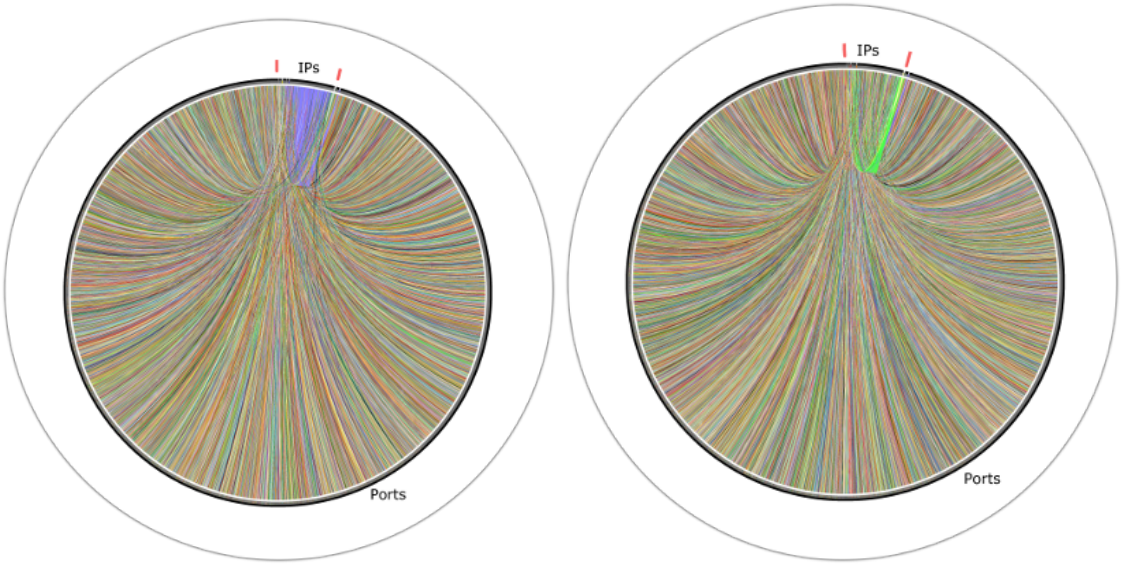

Figure 5.24: Circos representation obtained for destination port information for unknown traffic (later classified as *Neptune* attack traffic).

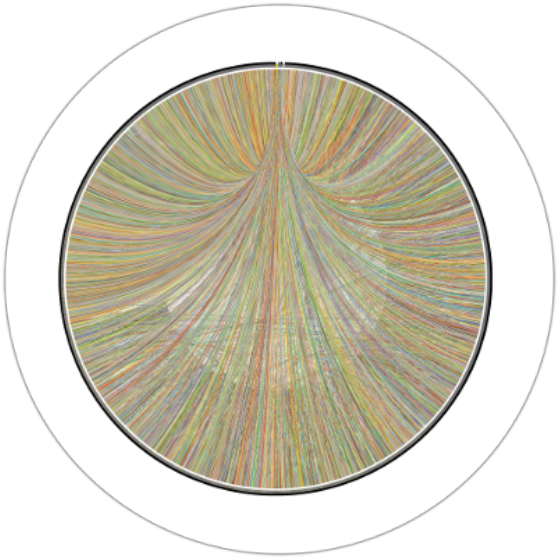

Figure 5.25: Circos representation obtained for destination and source port information for unknown traffic (later classified as *Neptune* attack traffic).

authentication procedure. In this case, it were the details emphasized by Circos and knowledge of the service that led to the conclusion, and not the similarities with other Circos. Some of these representations were also similar to the ones of *NMAP*, except for minor details.

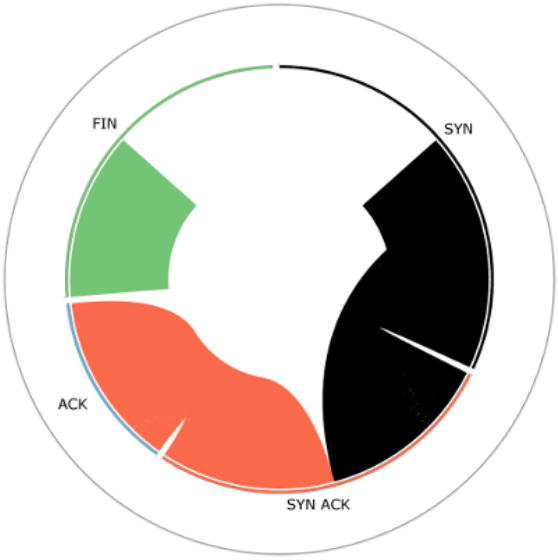

Figure 5.26: Circos representation obtained for TCP flag information for unknown traffic (later classified as *Neptune* attack traffic).

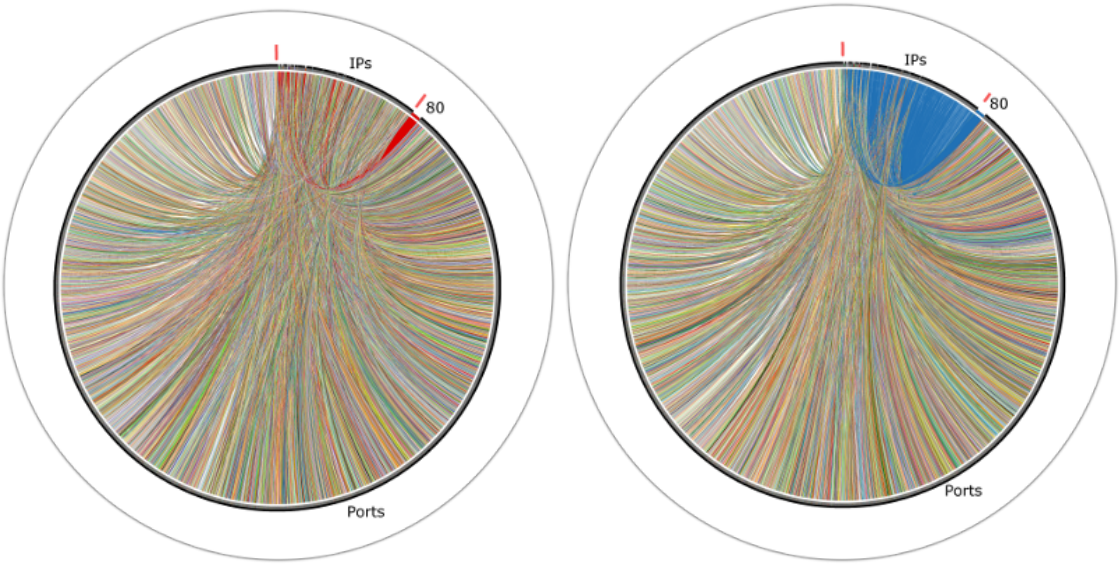

Figure 5.27: Circos representation obtained for source port information for unknown traffic (later classified as HTTP traffic).

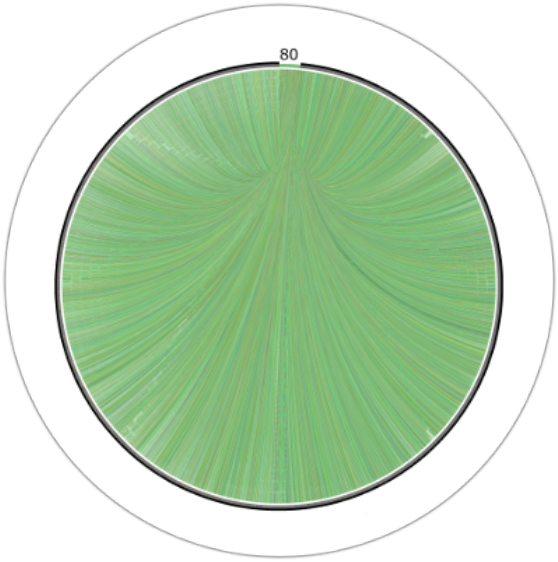

Figure 5.28: Circos representation obtained for destination and source port information for unknown traffic (later classified as HTTP traffic).

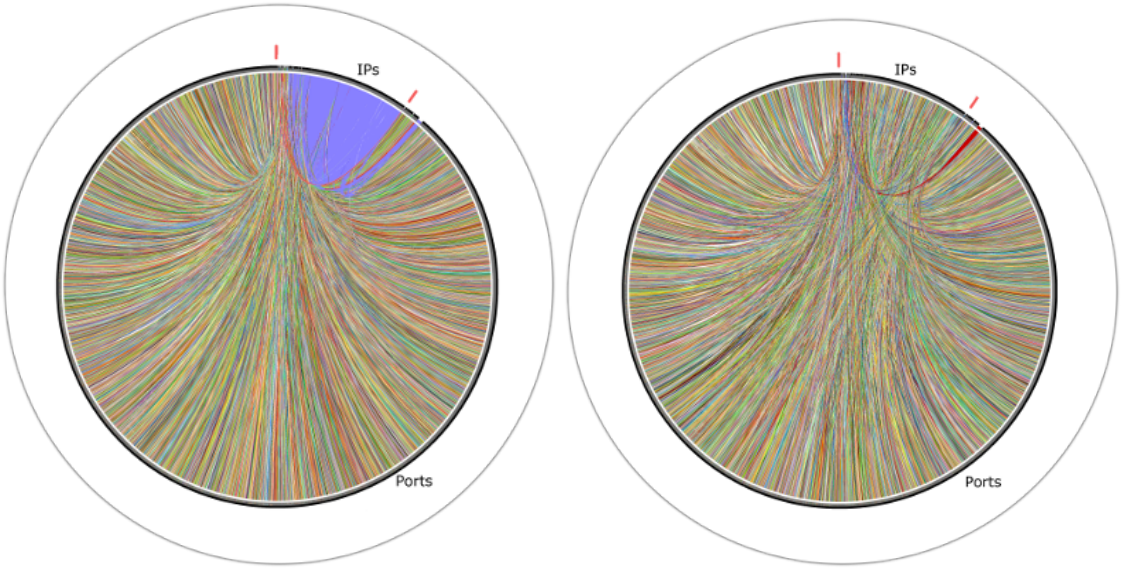

Figure 5.29: Circos representation obtained for destination port information for unknown traffic (later classified as HTTP traffic also).

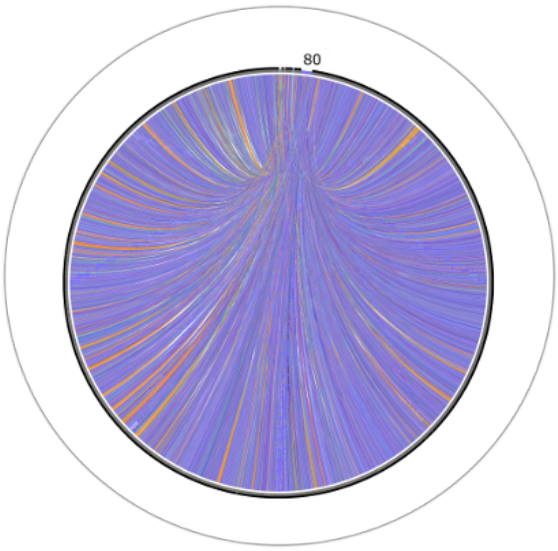

Figure 5.30: Circos representation obtained for destination and source port information for unknown traffic (later classified as HTTP traffic also).

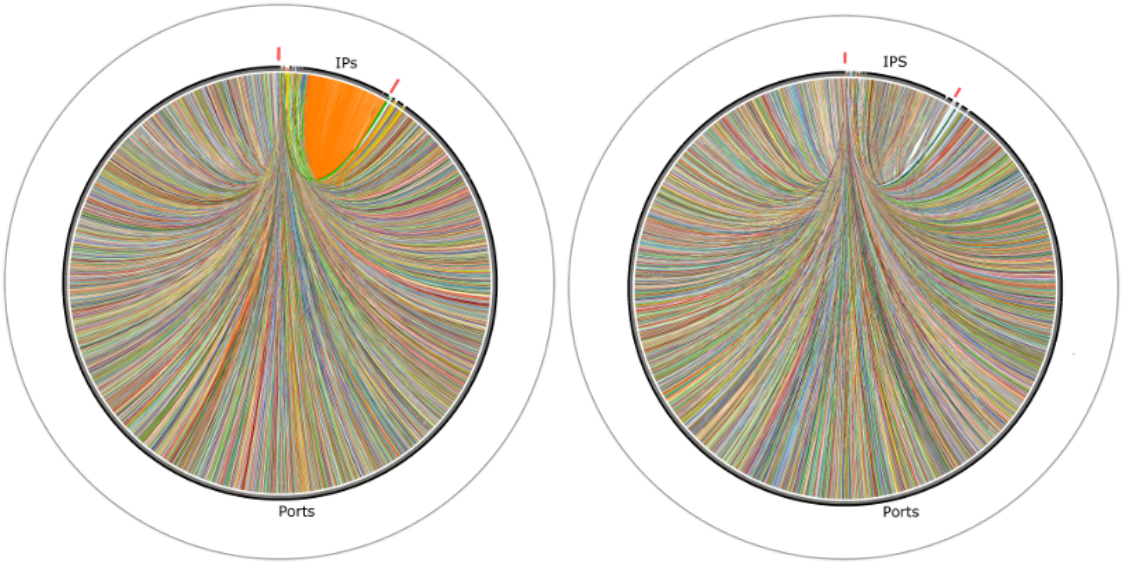

Figure 5.31: Circos representation obtained for destination port information for unknown traffic (later classified as *NMAP* attack traffic).

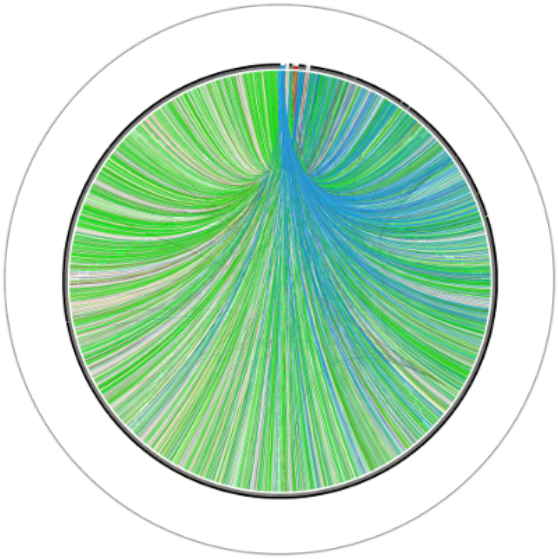

Figure 5.32: Circos representation obtained for destination and source port information for unknown traffic (later classified as *NMAP* attack traffic).

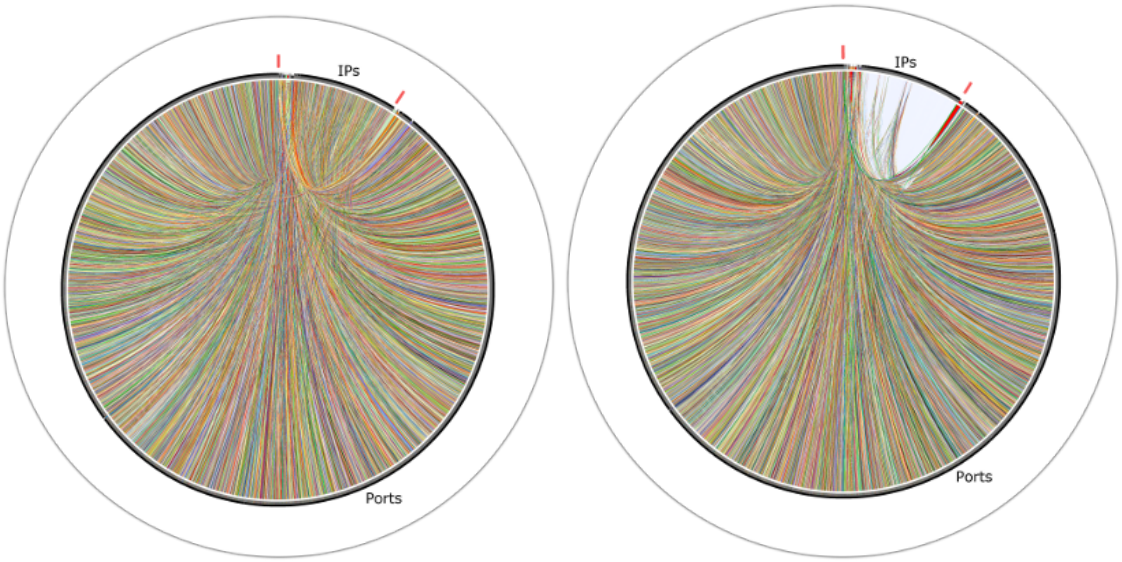

Figure 5.33: Circos representation obtained for source port information for unknown traffic (later classified as *Dictionary* attack traffic).

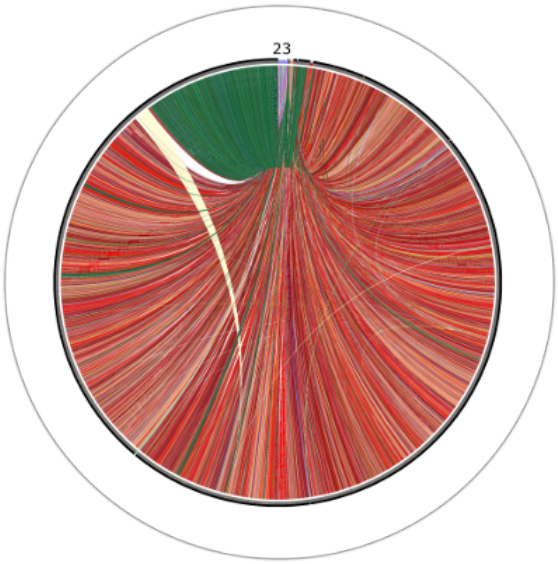

Figure 5.34: Circos representation obtained for destination and source port information for unknown traffic (later classified as *Dictionary* attack traffic).

5.5 Discussion

In the previous section, Circos generated from traces with unknown traffic were analyzed and sometimes compared with the ones that were discussed earlier. The idea was to show the usefulness of the representations in terms of traffic classification and attack detection. Classification was performed either by direct comparison of the charts or by analyzing emphasized details in some of the cases. to achieve this goal, a total of eighteen groups of Circos were created. There were nine coming from traces with ground truth knowledge, and other nine of which provenience was unknown. In each one of these larger subgroups, there were traces with legitimate traffic and with attacks. If there was a given traffic class or attack in one of the larger subgroups, then there was also a similar (but not equal) trace in the other. The charts for the unknown traces were created using automatic scripts, which randomized their names. The discussion started from the presentation of the Circos from labeled legitimate traces, evolved to the ones containing attacks and, finally, discussed the classification decisions for the unknown ones.

Table 5.1: TSummary of the classification experiment using Circos.

HTTP	VoIP	P2P	SSH	NMAP	Back	Satan	Neptune	Dictionary
Correctly Classified (CC)	CC	CC	CC	CC	CC	Incorrectly Classified (IC)	CC	CC

The overall results of this experiment are summarized in Table 5.1. As can be seen, most of the unknown traces were correctly classified, with only one failure. Not all classifications were done via direct comparison of data representations. Some of them were due to artifacts emphasized by these representations, as an abnormal number of connections to a given port, or the number of IP addresses involved, or the number of source ports in relation to the number of IP addresses, etc. On the other hand, if there are many relevant ports on multiple charts, a unique characteristic for one or a pair of those ports may be a sign of abnormal behavior. Obviously, using solely the data visualizations is not enough to produce accurate results, and deep knowledge of the attacks and protocols is necessary, since some attacks will produce traffic similar to legitimate services of applications in many aspects.

The classes that were easier to identify were the ones of VoIP, P2P and SSH, mostly due to the unique characteristics produced in the Circos. Protocols such as SSH and File Transfer Protocol (FTP) should be easy to identify in these charts, as they comprise simpler (in terms of connections) services, in which the ports are highlighted in the charts. VoIP and P2P are also quite dissimilar from all other classes of traffic. Given the results, it is foreseeable that the traffic that is going to be more difficult to identify concerns attacks on popular services, such as HTTP, unless the attack is truly overwhelming. For example, using the available tools, it was not possible to detect the *Satan* attack when in the presence of HTTP traffic or because it was addressing the server.

5.6 Conclusions

This chapter reflects the achievement of two objectives of this master's program. On the one hand, it includes and discusses many of the data visualizations produced using Circos. On the other, it shows the results concerning the usage of such representations to, e.g., classify traffic or identify network based attacks. The discussion suggests that there is work that needs to be done and room for many improvements. For example, in here, entire (offline) traces were used to produce Circos but, probably, the representations would be better if they were confined to smaller parts of the traces. The usage of animated Circos flowing through the data may also help emphasize momentary abnormal behaviors.

Chapter 6

Conclusions and Future Work

The final chapter of this dissertation is divided in 2 sections. The first section wraps up the most important conclusions of the work developed during the master's program; while the second one contains some guidelines and suggestions for future work.

6.1 Main Conclusions

This dissertation addresses the problem of producing useful visualizations of potentially large amounts of data, namely network traffic and attacks related. It started with the overview of some works on network and attacks visualization, to then converge to the more specific objective of classifying network traffic and identify attacks using the technique known as Circos. One of the main conclusions is that Circos can indeed be used for such purpose, using a simple traffic characteristics. The objectives of this master's program were achieved: a large set of Circos for contemporary network traffic and for classical network attacks was constructed during the course of this work; many data visualizations were thoroughly discussed in this dissertation to identify the aspects that the visualization were emphasizing; the experiment and results discussed in chapter 5 prove that they can be used to identify network traffic classes and attacks.

In order to construct the proof-of-concept and show the usefulness of Circos in this area, it was necessary to have datasets of traffic traces containing attacks, as explained in chapter 3.3. The set of traffic characteristics used to produce the data visualization was discussed in that chapter also. It is worth to mention that this work was seeking for methods to represent IP packet related information, and not aggregated statistics about network activity. It is often easier to produce data representations for aggregated statistics, since there is typically less data to represent. Circos proved to be an interesting choice for the purpose at hands, though some minor modifications had to be made to the respective software to handle much larger inputs than it was initially designed to handle.

Thousands of charts were generated and analyzed. For example, for each traffic trace, at least 6 different Circos were produced: destination vs. source transport layer ports; destination port

vs. destination IP; destination port vs. source IP; source port vs. destination IP; source port vs. source IP; and TCP flags sequences. Depending on the traffic class or associated application, most of the charts proved useful to emphasize specific artifacts of the communications. For example, Circos with TCP flags sequences are very good to identify SYN floods or abnormal TCP connections. In the final phase of the master's program, the charts were divided into two main groups: one with charts whose provenance was known; another one whose traffic classes or attacks were unknown. Each of these groups was subdivided into 9 subgroups, corresponding to 9 different traffic classes or attacks. It was possible to identify 8 out of the 9 traffic classes or attacks, which was higher than expected, given the utilized set of characteristics. The set that was not correctly identified belonged to an attack, with similar behavior to HTTP traffic. The results argue in favor of the potential of this data visualization technique applied to network traffic.

6.2 Future Work

It is possible to identify several paths for improving this work or several derivate research directions:

- **New features** – the first suggestion concerns looking for and trying out more traffic characteristics. This will potentially give rise to new visual artifacts that may be helpful in the identification of new attacks or in an increased accuracy in the classification of network traffic;

- **Different visualization techniques** – given the available time frame and the objectives of the proposal, it was decided to focus most of the work on Circos and on the basic characteristics that define network flows. Nonetheless, it would be beneficial to use more contemporary visualization techniques or the more complex versions (combined with other charts) of Circos;

- **New attacks** – the number of menaces is constantly increasing, with computer networks comprising one of the targets or means to perform them. It is not possible to humanly study every possible attacking approaches, but it would be interesting to extend this study to a wider range of menaces, namely to DDoS. In such case, it would be useful to take snapshots of the traffic at different network distances of the target, so as to obtain an idea of the different characteristics that could be used at each stage of the incursion;

- **Better automation** – many of the data visualizations were obtained using automated scripts, which processed the traces, prepared the inputs and ran the Circos software.

Nonetheless, given the size of the traces, many charts took several hours to be produced. The author believes that it is possible to improve this procedure, namely by further pre-processing the traces and perhaps reduce the input data before the critical part of creating the charts. If Circos are to be used in real-time monitoring systems, then an approach to refresh the charts in a packet-by-packet basis over a fixed size iterative window would probably be the best option. Implementing it would require changing (or extending) the Circos software;

- **Machine learning** and **Image Processing** – the human analysis of charts with the dataset of labeled Circos enabled the author to identify several classes of network traffic and attacks. Automating this process is a matter of future research work, and it can probably be done resorting to *image processing* and *machine learning* techniques;

- **Integration with a SIEM system** – a the end of this dissertation, it is argued that the produced data visualizations would be useful to attack identification and even traffic classification. The inclusion of Circos graphics in a SIEM system would be an interesting path to follow, perhaps as an extension to an open source one like the AlienVault OSSIM [Ali15].

Bibliography

[Age15] Defense Advanced Research Projects Agency. National Cyber Range Rapidly Emulates Complex Networks, March 2015. Last Access: March 29, 2015. Available from: http://www.darpa.mil/NewsEvents/Releases/2012/11/13.aspx. 11

[AH13] P.V. Amoli and T. Hamalainen. A Real Time Unsupervised NIDS for Detecting Unknown and Encrypted Network Attacks in High Speed Network. In *Proceedings of the 2013 IEEE International Workshop on Measurements and Networking Proceedings (M N)*, pages 149–154, October 2013. 9

[AJA14] O. Al-Jarrah and A. Arafat. Network Intrusion Detection System using Attack Behavior Classification. In *proccedings of the 2014 5th International Conference on Information and Communication Systems (ICICS)*, pages 1–6, April 2014. 10

[Ali15] AlienVault. AlienVault OSSIM: The World's Most Widely UsedOpen Source SIEM, 2015. Last Access: September 4, 2015. Available from: https://www.alienvault.com/products/ossim. 59

[CA04] Gregory Conti and Kulsoom Abdullah. Passive Visual Fingerprinting of Network Attack Tools. In *Proceedings of the 2004 ACM Workshop on Visualization and Data Mining for Computer Security*, VizSEC/DMSEC '04, pages 45–54, New York, NY, USA, 2004. ACM. xix, 15

[CDK+09] Georg Carle, Falko Dressler, RichardA. Kemmerer, Hartmut Koenig, Christopher Kruegel, and Pavel Laskov. Network Attack Detection and Defense - Manifesto of the Dagstuhl Perspective Workshop, March 2nd-6th, 2008. *Computer Science - Research and Development*, 23(1):15–25, 2009. 9

[CGM+13] S. Creese, M. Goldsmith, N. Moffat, J. Happa, and I. Agrafiotis. CyberVis: Visualizing the Potential Impact of Cyber Attacks on the Wider Enterprise. In *Proceedings of the 2013 IEEE International Conference on Technologies for Homeland Security (HST)*, pages 73–79, November 2013. xi, xix, 12, 13

[CH14] Jonathan Corum and Farhana Hossain. Naming Names, March 2014. Last Access: June 14, 2015. Available from: http://circos.ca/intro/general_data/. 22

[Cis15] Cisco. Cloud Managed Security, 2015. Last Access: September 25, 2015. Available from: https://meraki.cisco.com/products/appliances. 2

[DEL15] SonicWall DELL. The Network Security SonicOS Platform, August 2015. Last Access: September 25, 2015. Available from: http://www.sonicwall.com/us/en/products/Network-Security-Platform.html. 2

[DG13] Rob Knegjens Damien George. Paperscape, July 2013. Last Access: September 3, 2015. Available from: http://paperscape.org/. xix, 18

[DoHS15] National Cyber Security Division Department of Homeland Security. Cyber Storm: Securing Cyber Space, September 2015. Last Access: September 25, 2015. Available from: http://www.dhs.gov/cyber-storm-securing-cyber-space. 11

[ECS05] R.F. Erbacher, K. Christensen, and A. Sundberg. Designing Visualization Capabilities for IDS Challenges. In *Proceedings of the IEEE Workshop on Visualization for Computer Security, 2005. (VizSEC 05).*, pages 121–127, October 2005. 9

[Ein14] Max Einstein. GOD, February 2014. Last Access: September 3, 2015. Available from: http://datalooksdope.com/god/. xix, 17

[FMK+08] Fabian Fischer, Florian Mansmann, DanielA. Keim, Stephan Pietzko, and Marcel Waldvogel. Large-Scale Network Monitoring for Visual Analysis of Attacks. In *Computer Science on Visualization for Computer Security*, volume 5210 of *Lecture Notes in Computer Science*, pages 111–118. Springer Berlin Heidelberg, 2008. 8

[HA06] Warren Harrop and Grenville Armitage. Real-time Collaborative Network Monitoring and Control Using 3D Game Engines for Representation and Interaction. In *Proceedings of the 3rd International Workshop on Visualization for Computer Security*, VizSEC '06, pages 31–40, New York, NY, USA, 2006. ACM. xi, xix, 14

[HLH11] Zach Harbort, G. Louthan, and J. Hale. Techniques for Attack Graph Visualization and Interaction. In *Proceedings of the 7th Annual Workshop on Cyber Security and Information Intelligence Research*, (CSIIRW) '11, pages 74:1–74:1, New York, NY, USA, 2011. ACM. xix, 15, 16

[JPLL09] Zhang Jiawan, Yang Peng, Lu Liangfu, and Chen Lei. NetViewer: A Visualization Tool for Network Security Events. In *Proceedings of the International Conference on Networks Security, Wireless Communications and Trusted Computing, (NSWCTC) 2009.*, volume 1, pages 434–437, April 2009. xi, 11

[KLS13] Bonhyun Koo, YangSun Lee, and Taeshik Shon. A Novel Approach to Visualize Web Anomaly Attacks in Pervasive Computing Environment. *The Journal of Supercomput-*

ing, 65(1):301–316, 2013. 7

[KSB+09] Martin I Krzywinski, Jacqueline E Schein, Inanc Birol, Joseph Connors, Randy Gascoyne, Doug Horsman, Steven J Jones, and Marco A Marra. Circos: An Information Aesthetic for Comparative Genomics. *Genome Research*, 2009. Available from: http://genome.cshlp.org/content/early/2009/06/15/gr.092759.109.abstract. xi, 21

[KSV07] Ramana Rao Kompella, Sumeet Singh, and George Varghese. On Scalable Attack Detection in the Network. *IEEE/ACM Transaction Networks*, 15(1):14–25, February 2007. 11

[Lou15] Idalécio Lourenço. A Cibersegurança pode sair cara, mesmo muito cara (entrevista a Eng. José Alegria), July 2015. Last Access: July 15, 2015. Available from: http://www.cio.pt/2015/07/14/a-ciberseguranca-pode-sair-cara-mesmo-muito-cara/. ix, 1

[LPCB11] Sylvain P. Leblanc, Andrew Partington, Ian Chapman, and Mélanie Bernier. An Overview of Cyber Attack and Computer Network Operations Simulation. In *Proceedings of the 2011 Military Modeling & Simulation Symposium*, (MMS) '11, pages 92–100, San Diego, CA, USA, 2011. Society for Computer Simulation International. 11

[LTG+05] C.P. Lee, J. Trost, N. Gibbs, R. Beyah, and J.A. Copeland. Visual Firewall: Real-Time Network Security Monitor. In *Proceedings of the IEEE Workshop on Visualization for Computer Security, 2005. (VizSEC 05).*, pages 129–136, October 2005. xix, 13

[McB07] Aaron McBride. Air Force Cyber Warfare Training. *The Defense Standardization Program Journal*, pages 9–13, 2007. 11

[MIT14] DARPA Intrusion Detection Evaluation Massachusetts Institute Technology. 1998 Training Data Attack Schedule, 2014. Last Access: October 4, 2014. Available from: http://www.ll.mit.edu/ideval/docs/attacks.html. xi

[MIT15] DARPA Intrusion Detection Evaluation Massachusetts Institute Technology. Intrusion Detection Related Publications, 2015. Last Access: October 24, 2015. Available from: http://www.ll.mit.edu/ideval/pubs.html. 25

[ML08] Raphael S Mudge and Scott Lingley. Cyber and Air Joint Effects Demonstration (CAAJED). Technical report, DTIC Document, 2008. 11

[MMB06] Chris Muelder, Kwan-Liu Ma, and Tony Bartoletti. Interactive Visualization for Net-

work and Port Scan Detection. In *volume 3858, Lecture Notes in Computer Science on Recent Advances in Intrusion Detection*, pages 265–283. Springer Berlin Heidelberg, 2006. 8

[NAU+13] Troy Nunnally, Kulsoom Abdullah, A. Selcuk Uluagac, John A. Copeland, and Raheem Beyah. NAVSEC: A Recommender System for 3D Network Security Visualizations. In *Proceedings of the 10th Workshop on Visualization for Cyber Security*, (VizSec) '13, pages 41–48, New York, NY, USA, 2013. ACM. xi, xix, 13, 14

[NJKJ05] S. Noel, M. Jacobs, P. Kalapa, and S. Jajodia. Multiple Coordinated Views for Network Attack Graphs. In *Proceedings of the IEEE Workshop on Visualization for Computer Security, 2005. (VizSEC 05).*, pages 99–106, October 2005. 10

[NS14] Ramon Bauer Nikola Sander, Guy J. Abel. The Global Flow of People, February 2014. Last Access: June 15, 2015. Available from: `http://www.global-migration.info/`. 22

[NUCB12] T. Nunnally, A.S. Uluagac, J.A. Copeland, and R. Beyah. 3DSVAT: A 3D Stereoscopic Vulnerability Assessment Tool for Network Security. In *Proceedings of the 2012 IEEE 37th Conference on Local Computer Networks (LCN).*, pages 111–118, October 2012. xi, 12

[oC14] Town of Caceres. Urban Planning, March 2014. Last Access: June 15, 2015. Available from: `http://circos.ca/intro/general_data/`. 22

[PAC14] Arnaud Picandet Paul-Antoine Chevalier. What is Wikipedia About?, August 2014. Last Access: September 3, 2015. Available from: `http://www.informationisbeautifulawards.com/showcase/608-what-is-wikipedia-about`. xix, 19

[REHA11] Alaa El-Din Riad, Ibrahim Elhenawy, Ahmed Hassan, and Nancy Awadallah. Data Visualization Technique Framework for Intrusion Detection. *International Journal of Computer Science Issues (IJCSI)*, 8(5), 2011. xix, 15, 16

[Rev14] ReversingLabs. Network Security Appliance, 2014. Last Access: September 25, 2015. Available from: `http://reversinglabs.com/products/network-security-appliance.html`. 2

[Sin10] Rahul Singh. Divine Matrix: Indian Army fears China Attack by 2017, February 2010. Last Access: March 29, 2015. Available from: `http://www.infowar-monitor.net/`

2010/02/divine-matrix-indian-army-fears-china-attack-by-2017/. 11

[SJ14] Kun Sun and Sushil Jajodia. Protecting Enterprise Networks Through Attack Surface Expansion. In *Proceedings of the 2014 Workshop on Cyber Security Analytics, Intelligence and Automation*, SafeConfig '14, pages 29–32, 2014. 10

[Spl15] Splunk. Splunk Enterprise Security, August 2015. Last Access: September 25, 2015. Available from: https://splunkbase.splunk.com/app/2800/. 2

[SW04] Oleg Sheyner and Jeannette Wing. Tools for Generating and Analyzing Attack Graphs. In *volume 3188 of Lecture Notes in Computer Science Formal on Methods for Components and Objects*, pages 344–371. Springer Berlin Heidelberg, 2004. 11

[tcp15] tcpdump. TCPDUMP & LIBPCAP, June 2015. Last Access: September 24, 2015. Available from: http://www.tcpdump.org/. 25

[Tri15] Rob Triggs. How Far we've Come: a Look at Smartphone Performance Over the Past 7 Years, July 2015. Last Access: September 25, 2015. Available from: http://www.androidauthority.com/smartphone-performance-improvements-timeline-626109/. 1

[Tul14] Jan Willem Tulp. Edible or Medical, November 2014. Last Access: September 3, 2015. Available from: http://www.naturalrecall.org/jan-willem-tulp/. xix, 17

[vHPBI13] R. van Heerden, H. Pieterse, I. Burke, and B. Irwin. Developing a Virtualised Testbed Environment in Preparation for Testing of Network Based Attacks. In *Proceedings of 2013 International Conference on Adaptive Science and Technology (ICAST).*, pages 1–8, November 2013. 9

[ZZF+13] Ying Zhao, FangFang Zhou, XiaoPing Fan, Xing Liang, and YongGang Liu. IDSRadar: a Real-Time Visualization Framework for IDS Alerts. *Science China Information Sciences*, 56(8):1–12, 2013. 8