A Survey Of Network Flow Applications

Bingdong Li[a,b,*], Jeff Springer[a], George Bebis[b], Mehmet Hadi Gunes[b]

[a] *Department of Information Technology, University of Nevada-Reno, NV 89557*
[b] *Department of Computer Science & Engineering, University of Nevada-Reno, NV 89557*

Abstract

It has been over 16 years since Cisco's NetFlow was patented in 1996. Extensive research has been conducted since then and many applications have been developed. In this survey, we have reviewed an extensive number of studies with emphasis on network flow applications. First, we provide a brief introduction to sFlow, NetFlow and network traffic analysis. Then, we review the state of the art in the field by presenting the main perspectives and methodologies. Our analysis has revealed that network security has been an important research topic since the beginning. Advanced methodologies, such as machine learning, have been very promising. We provide a critique of the studies surveyed about data sets, perspectives, methodologies, challenges, future directions and ideas for potential integration with other Information Technology infrastructure and methods. Finally, we concluded this survey.

Keywords: Machine Learning; NetFlow; Network Traffic Analysis; Network Security; sFlow;

1. Introduction

Computer networks are playing a very important role in our daily life. Our dependency on computer networks is growing tremendously. Understanding what information flows in a computer network is important not just for network administrators but also for accounting, network planning, network security, forensics and counter-terrorism. Many governments require Internet Service Providers (ISP) to have capabilities of 'lawful interception'

[*] Corresponding Author: bingdongli@unr.edu, Tel: +1(775)682-6805, Fax: +1(775)784-4529

(LI) network traffic [37]. Moreover, network flow can provide information for business relationship [65].

Network flow records high-level descriptions of internet connections but not the actual data transferred. Collection and analysis of network flow information is more efficient than deep packet inspection and protects the privacy of users. This information helps to uncover both external activities as well as internal activities such as network misconfiguration and policy violation. Network flow information is supported by a wide range of products including Cisco Netflow [62], Juniper' cflowd, NetStream, and sFlow. These systems are all similar to NetFlow systems, and will be referred to as NetFlow-like in this survey.

It has been over 16 years since Cisco's NetFlow was developed by Darren and Barry Bruins in 1996 [62]. Research in network flow analysis has become very active in the recent years as observed in figure 1. It is necessary to look back what perspectives have been achieved and what methods have been used and are more effective in order to move forward. This paper presents a survey of NetFlow-like applications that papers published between 1998 and early 2012. Figure 1 shows the paper distribution with respect to publication year. Our objective is to provide a better understanding of major achievements in the field by reviewing state of the art approaches, perspectives, important issues and challenges, and suggesting directions for future research. Note that, we have focused mostly on studies using NetFlow-like data as input, emphasizing some of the latest approaches rather than attempting to provide a complete historical review of network flow applications.

Related reviews discussing similar aspects to this survey but not specific to NetFlow-like applications can be found in [102] for IP-Flow based intrusion detection, [160] for botnet detection, [98] for internet traffic classification using machine learning, and [123] for discussion of using machine learning for network intrusion detection.

Traditionally, NetFlow-like analysis systems have been used for network monitoring, planning and billing. Recent research approaches, however, have focused more on network security analysis with the objective of detecting anomalous activities that traditional security infrastructures, such as intrusion detection systems (IDS), firewalls and anti-virus tools, can not handle. These approaches employ advanced techniques such as machine learning. Moreover, new NetFlpw-like analysis system design is moving toward distributed systems to provide more scalability, robustness and computational power for real-time in depth analysis.

Despite the popularity of sFlow and its wide deployment, few studies have focused on using sFlow as their data source.

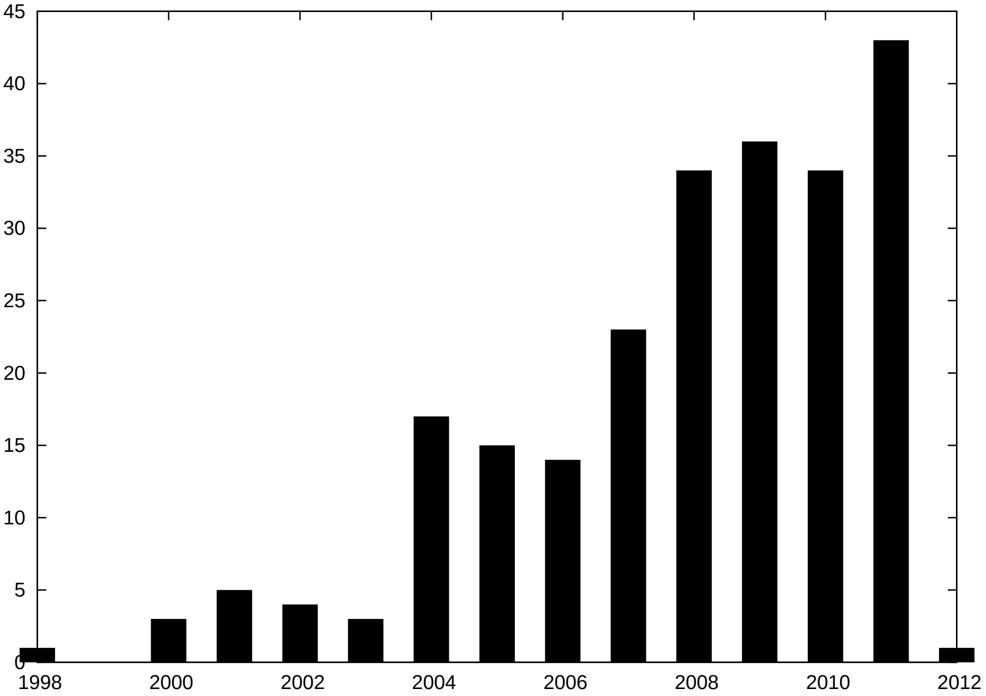

Figure 1: Publications by Year

The rest of this paper is organized as follows. Section 2 provides a brief introduction to NetFlow, sFlow, IPFIX, and network flow analysis. Section 3 reviews the key perspectives which have been addressed in the literature including networking monitoring, analysis and management, application classification, inferring user identity, and network security awareness. Section 4 explains the key methodologies and tools which have been employed for data analysis including statistic, machine learning, profiling, behavior-based approaches, sampling, visualization, and computational infrastructures. Section 5 discusses the limitations of existing approaches, challenges and future directions. Finally, Section 6 presents our conclusions.

2. Background

Network flow is defined as an unidirectional or bidirectional sequence of packets between two endpoints (from server to client or from client to server) with some common attributes. The most important key fields include: first and last time of the flow received, source/destination IP address, source/destination port number, layer 3 protocol type, type of services, bytes transferred, and input logical interface. Additional fields may be included that depend on the NetFlow version or configuration for export. They provide a rich set of traffic statistics including user, protocol, port, and type of services which can be used for a wide variety of purposes such as network security, network monitoring, traffic analysis, capacity planning, traffic classification, accounting, and billing. The general process of working with NetFlow includes capturing, sampling, generating, exporting, collecting, analyzing and visualizing.

There are various systems that capture NetFlow IP operational data from network links or devices: Ntop [26], NG-MON [52], NetFlow, NetFlow-lite, sFlow, cflowd, NetStream, etc. In addition, IPFIX [3] defines the standard IP flow format for exportation. NetFlow and sFlow are widely used systems while IPFIX is a new standard. In this section, we will give a brief introduction about NetFlow, sFlow, IPFIX, and taffic analysis.

2.1. ***NetFlow***

NetFlow is a traffic monitoring technology developed by Darren and Barry Bruins in 1996 at Cisco [62]. It defines how a router exports information and statistics of routed sockets. As a de facto industry standard, it is a built-in feature of most routers and switches from Cisco, Juniper, and other vendors. Network devices look at the packets arriving on the interfaces, and capture traffic statistics per flow based on configuration for sampling or filtering, then they create a flow cache, aggregate and export the data through UDP or Stream Control Transport Protocol (SCTP). NetFlow cache entry is created by the first packet of a flow, maintained for similar flow characteristics, and exported to collectors periodically based on flow timers or flow cache management. The export format were fixed before version 8. After version 9, extensibility and flexibility are added to integrate with MPLS, IPv6 and BGP, and user defined records. NetFlow version 5 and 9 are the most popular versions. Sampled NetFlow is a variant originally introduced by Cisco to reduce computation burden by reducing number of NetFlow. It can be configured as predetermined n^{th} packet or randomly selected interval. Figure 2 presents the basic process of NetFlow formation, exportation, storage and analysis. Due to the great value of network traffic and limited computational resources (memory, CPU and bandwidth), technologies of caching, sampling and UDP exportation were used. These can cause quality issues for the collected NetFlow data: (1) some new flows will not be counted when cache is full; (2) sampling reduces the accuracy of flows, especially when sampling rate is adjusted by the traffic rate; (3) exported flow records do not necessarily correspond to the order in which the flow traffic arrived at the router. There are varieties of NetFlow collectors and analysis tools from commerical vendors such as Cisco, freeware or developed in-house for special purposes [4, 102].

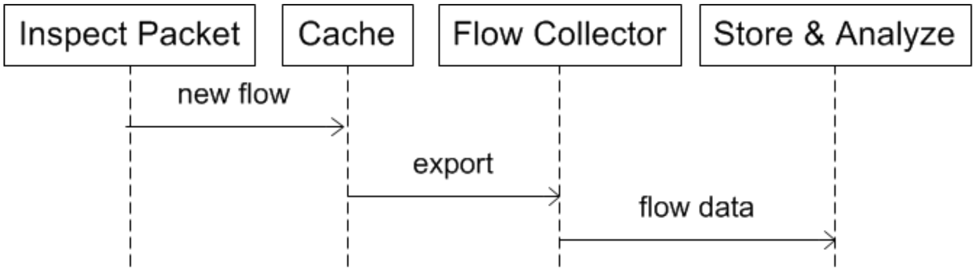

Figure 2: NetFlow Process

2.2. ***sFlow***

Packet sampling of traffic flow [31] has a long history before NetFlow was developed. sFlow was developed by *InMon Inc.* and has become an industry standard defined in RFC 3176. It is a technology using simple random sampling and supported by Alcatel, Extreme, Force10, HP, Hitachi by embedding the sFlow agent within switches and routers. The sFlow agent is a software process that combines interface counters and flow samples into sFlow datagrams and immediately sends them to sFlow collectors via UPD. Immediate forwarding of data minimizes memory and CPU usage. Packets are typically sampled by Application-Specific Integrated Circuits (ASICs) to provide wire-speed performance. sFlow data contains complete packet header and switching/routing information, and provides up to the minute view of the network traffic. sFlow is able to run at layer 2 and capture non IP traffic as well. The sFlow collectors are servers that collect the sFlow datagrams. The official sFlow web site [6] provides a list of available sFlow collectors. Figure 3 present the basic components and processes of sFlow analysis.

2.3. ***IPFIX***

IP Flow Information Export protocol (IPFIX) is an IETF standard for exporting network flow based on NetFlow version 9, and is defined in RFC 5101

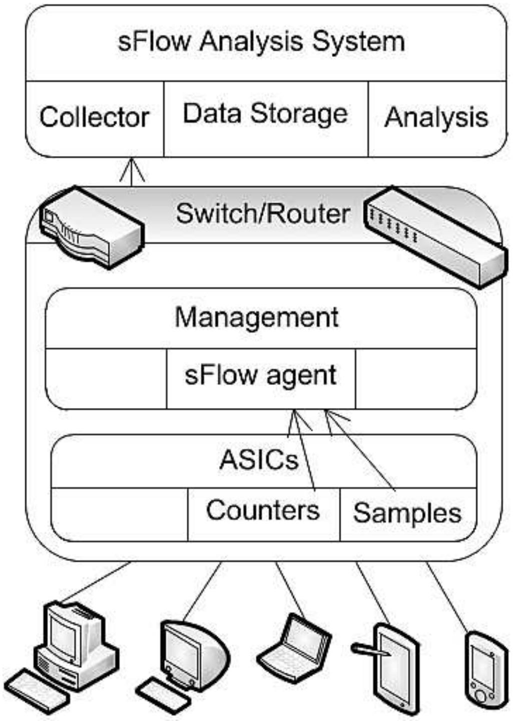

Figure 3: sFlow Process

for information transmitting protocols , RFC 5102 for information model, and RFC 5103 for exporting bidirectional flow. IPFIX was designed to meet the fast growing requirements to observe network traffic, provide an extensible and flexible data model that can be customized, and support reliable and secure data transfer through SCTP, TCP and UDP. IPFIX flow definition is less restrictive than traditional flow key definition. As standardization is underway, more vendors are going to support IPFIX.

*2.4. **Network Flow Analysis***

Network flow analysis is the process of discovering useful information by using statistics or other sophisticated approaches. The basic process includes capturing, collecting and storing data, aggregating the data for query and analysis, and analyzing the data and results for useful information. This information is mostly related to network management, measurement, and network security. There are different ways to collect network flow data: SNMP, NetFlow, raw packet, or auditing data from network infrastructure such as IDSes, Firewalls, and VPN gateways. Typically, there are two strategies:

depth-first when there is known information and clear purpose, or breadth-first when looking for a general view of the network without a particular purpose.

Deep packet inspection needs packet level information and consumes more computational resources. Flow level analysis, such as NetFlow and sFlow, consume less computational resources. There are many products and tools developed by industry or open source community. Analysis of network flow information has become crucial as the internet has become the living blood in our society and is expanding at a fast pace around the world. There are many challenges in analyzing network flow data such as, huge amount of data due to networks becoming larger and faster, limited high-level information, and complex statistical properties. As discussed in sections 3 and 4, various perspectives have been analyzed and many algorithms have been developed.

3. Perspectives

In this section, we survey the main research perspectives of network flow applications. In particular, we cover network monitoring, measurement and analysis, application classification, user identity inferring, security awareness and intrusion detection, and issues related to error and bias in NetFlow collection and analysis. Figure 4 shows the distribution of papers with respect to four main perspectives: monitoring, classification, security, and issues related to errors. As it can be observed, network security has been the main research topic using NetFlow data.

3.1. ***Network Monitoring, Measurement and Analysis***

Network monitoring and measurement provides valuable information to network administrators, as well as ISPs and content providers. Compared to other technologies, such as SNMP or Windows Management Instrumentation(WMI), network flow data contain additional information for further analysis. For example, they can provide bandwidth analysis, specific protocol monitoring, and system performance, etc. Monitoring based on NetFlow can be categorized as:

- *Network monitoring*: provides information about routers and switches as well as network-wide basis view, and is used for problem detection along with efficient troubleshooting.

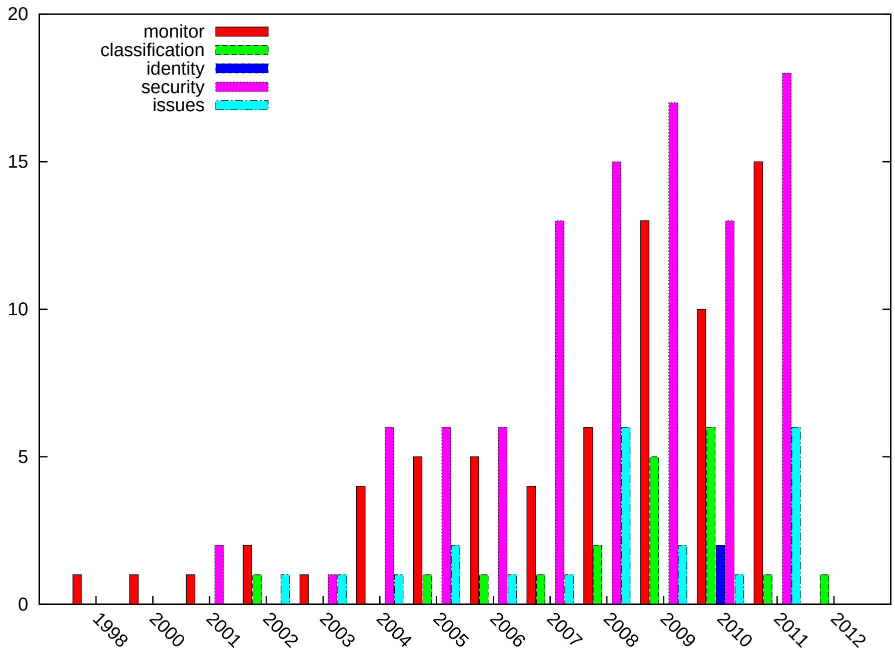

Figure 4: Analyzed Perspectives by Years

- *Application monitoring*: provides information about application usage over the network, and is used for planning and allocation of resources.
- *Host monitoring*: provides information about user utilization of network and applications, and is used for planning, network access control, violating security policy.
- *Security monitoring*: provides information about network behavior changes, and is used to identify DoS attacks, viruses and worms, and network anomalies.
- *Accounting and Billing*: provides network metering, and is used for billing.

In this section, we focus on network, application, user and resource monitoring while section 3.4 focuses on security related monitoring. In the following, we discuss specific research perspectives.

3.1.1. Network Monitoring

Many aspects of networks have been studied using NetFlow data, including network performance based on round-trip time [128], delay measurement [67, 78], connectivity [115], misuse of bandwidth [85], traffic characterization [71], finding heavy hitters [133], monitoring for special purpose QoS [80], and diagnostic of troubleshooting [129].

3.1.2. Application Monitoring

Liu and Huebner [82] investigated the stochastic characteristics of distributions of flow length, packet size, throughput, etc. for the popular and bandwidth consuming applications. *Kalafut et al.* [59] proposed a heuristic method to differentiate wanted and unwanted traffic based on the sampled NetFlow data.

3.1.2.1. VoIP. Voice over IP (VoIP) service is widely used, however, it introduces security threats that include Session Initiation Protocol (SIP) scan, SIP flooding, and Real-time Transport Protocol (RTP) flooding. *Lee* [76] developed a system that can monitor VoIP service and detect VoIP network threats based on NetFlow statistics and behaviors. *Kobayashi* [68] presented a method for measuring VoIP traffic fluctuation by using NetFlow and sFlow based on the variance of the interval of the target RTP packets. *Lucas* [27] provided an open source VoIP monitoring system based on protocol characteristics.

3.1.2.2. Mobile Network. Sinha et al. [121] analyzed the flow-level upstream traffic behavior from Broadband Fixed Wireless (BFW) and Digital Subscriber Link (DSL) to provide traffic characteristics of these networks. *Moghaddam* [90] used wireless NetFlow data to measure and simulate user behavior and provide information for future mobile network design.

3.1.2.3. IPv6. With the transition from IPv4, there is a need to understand IPv6 usage including user behavior, traffic volume, transitional technologies, assignment of IPv6 address, IPv6 percentage of network traffic. NetFlow can provide this information including application types, usage of transitional technologies of IPv6 to IPv4, interface identifier assignment schemes, etc [119, 156].

3.1.3. Host Monitoring

Host profile and relationships in the network can be used for resource planning as well as for network security analysis. *Caracas et al.* [16] proposed an algorithm based on NetFlow data to describe the dependencies among computer systems, software components, and services. *Kind et al.* [65]

presented a method to uncover the relationships between IT infrastructures using NetFlow data. *Chen et al.* [20] developed novel heuristics to analyze characteristics and correlations between inter-data centers and client traffic, provide insights into data center design and operation. Several methods have been proposed for profiling behaviors on the end hosts [143, 148, 149]. Behavior-based approaches will be discussed in detail in section 4.4.

3.2. ***Network Application Classification***

Network application classification classifies network traffic into certain application categories which can be coarse- or fine-grained. Network application classification is a challenging task because of obfuscation techniques such content encryption, dynamic ports, and proprietary communication protocols. Classification approaches can be divided into four categories: port based, payload-based, heuristic based using transport layer statistics, and machine learning based. Port based approaches are no longer reliable because of certain applications that randomly assign ports. Payload based approaches do not work on encrypted traffic, and are resource-intensive and scale poorly with high bandwidth. Approaches based on heuristic and machine learning approaches provide alternative methods.

There are many reasons for network traffic classification. Network administrators need information about applications running at the network (e.g., file sharing), if they are legitimate users or worms. ISPs and content providers need the information for quality of service assurances. Research in this area has existed for over ten years, but is still growing. There is a list of 68 papers and 86 data sets collected in *CAIDA* web pages [2]. There are several surveys on network application classification using traffic classification approaches [63] and machine learning approaches [98]. We discuss machine learning approaches in detail in section 4.2.1. It is worth mentioning that *Perelman et al.* proposed a method that investigates the application signatures of web browsers, mail client, or media-players in network flow [104]. Peer to peer networks have become a major security concern and the focus of most network classification studies. We discuss peer to peer classification in detail below.

3.2.1. Peer to Peer Network

Peer to Peer (P2P) networks have been widely utilized for file-sharing, video distribution and voice communications. They consume more internet traffic than traditional applications, and have been a concern for network administrators and a challenge for network security. There is interest from ISPs and network administrators to identify and control the P2P network traffic [49, 154]. NetFlow provides an alternative approach that is more efficient in terms of storage and processing than deep packet inspection (DPI). Recently, there has been considerable effort on NetFlow P2P analysis. These include methods based on: (a) default P2P port for heavy-hitters [138], (b) port usage pattern of specific P2P network such as BitTorrent [13, 49], (c) flow statistic characteristics such as packet length and time-interval [13, 107, 147], (d) TCP flags that a host, as both client and server, send/receive a packet with both SYN and ACK at the same time [58], (e) machine learning that using features such as IP address and port, packet size, bytes exchanged. Among machine learning approaches we discuss in section 4.2.1, six out of eight classifiy P2P traffic.

3.3. ***User Identity Inferring***

Identifying a person based on extrinsic biometric is not new; well-known examples include signatures and keystrokes. Inferring user identity based on network flow patterns however is a new field. *Melnikov et al.* [87] discussed the potential of inferring user identity using NetFlow feature distribution and cross-correlation of various trace parameters and relationships among packets. Even though the reported results were preliminary, additional research will yield more promising results.

3.4. ***Security Awareness and Intrusion Detection***

In this section, we focus on security related awareness, detection and monitoring. Table 1 provides a list of studies that provide perspectives on security awareness and intrusion detection. Table 3 also lists approaches that use machine learning approaches. IDSes can be categorized based on how they identify intrusions: anomaly-based, misused-based (knowledge-based or signature-based), or combination of both anomaly and misuse-based [126]. Alternatively, IDSes can be categorized based on what they target: host-based, network-based or both.

Network anomaly detection refers to finding patterns that are not expected users behaviors, also known as anomaly-based IDS. Compared with misuse-based IDS, these patterns are previously unknown. Most content-oriented systems belong to knowledge-based detection, which looks for known signatures of malware by inspecting traffic packets. Most behavior-oriented systems belong to anomaly-based detection, which differentiate anomalous behavior from normal behavior. NetFlow based IDSes use existing NetFlow data and limited information and avoids privacy issues compared to content-oriented approaches. However, NetFlow based IDSes are more difficult because of limited variables in the NetFlow data. Consequently, recent research has shown that machine learning approaches are better than statistical and streaming methods. *Sperotto et al.*s [126] conducted an overview of IP flow-based intrusion detection that focused on flow-based IDS, concept of flows, classification of attacks, and defense technique. In the following, we discuss perspectives of security awareness and intrusion detection that can be achieved using NetFlow data.

3.4.1. Top N

Top N refers a set of statistic and models of NetFlow data. They reflect the basic network status. It is relatively simple with NetFlow analysis. It can be used to find the big talkers or heavy-hitters. It also can be used for abnormal traffic detection [155].

3.4.2. Port Scan

Port scanning is the act of systematically scanning a computer's ports, and is usually done by using small packets that probe the target machines. In most network attacks, port scanning is the first reconnaissance step. Scans can be classified in three categories: scanning many ports on a single host, scanning a single port on many hosts, and combination of both. Detection of port scan is addressed in most studies cited in table 1. Approaches include host incoming/outgoing connections, probability of entropy, Bayesian logistic regression, distances from baseline models, and machine learning.

3.4.3. Denial of Service

A denial-of-service (DoS) or distributed denial-of-service (DDoS) attack is an attempt to make the target host or network resource unable to respond to its requests. Detection of DOS or DDoS is addressed in most flow based IDSes. *Gao et al.* [46] proposed a resilient DoS detection based on sketch-based schemes that use a hash table for storing aggregated flow measurement. *Kim et al.* [64] described different DoS attacks based on traffic patterns and presented a network anomaly detection method that can detect flooding attacks. New developments include *Yin et al* [61] who use novel dynamic entropy to measure the anomaly, *Galtsev* [44] who presented an attack detection method based on statistic aggregation that can detect DDoS and port scanning. Table 1 provides a list of related studies.

3.4.4. Worms

A worm is a standalone malicious program that replicates across the networks by exploiting software vulnerabilities or tricking users to execute it by social engineering. Worms can cause mildly annoying effects, damaging data or software, DoS, stealing data, etc. Detection of worms can be categorized as trap-oriented, packet-oriented and connection-oriented [18]. Detection of scans is one of the important steps for worm detection, and hence many similar approaches are used in both types of detection. NetFlow-like approaches are connection-oriented and include: analysis of host behavior on the basis of incoming and outgoing connections [29], correlation between NetFlow data and honeypot logs [28], and detecting hit-list worms using protocol graphs [25]. *Chan et al.* [18] proposed FloWorM system that includes tracker, analyzer and reporter based on NetFlow data. *Abdulla et al.* [7] presented a Support Vector Machines (SVM) method based on the fact that a scanning activity or email worm initiates a significant amount of traffic without DNS.

3.4.5. Botnet

Botnets are malware at the infected target and controlled by a remote entity known as bot-master. They have become one of the major security threats credited for DDoS, spamming, phishing, identity theft, and other cyber crimes. Many botnets rely on communication channels varying from centralized IRC and HTTP to decentralized P2P networks. Detection of a botnet is relatively more difficult than detection of port scan and worms. *Zhu et al.* [18] conducted a survey on understanding, detection and tracking, and defending against botnets. Recent approaches use advanced methodologies and combine host and network level information. *Zeng et al.* [153] proposed a method that combined host and network-level information with protocol-independent detection. BotCloud's detection is based on MapReduce and combining host and network approaches [42]. BotTrack's tracking is based on PageRank of NetFlow data and host behavior model [41]. Finally, *Barsamian* uses a network statistical behavioral model for botnet detection [10], and *Weststrate* uses heuristic methods to find botnet servers [145].

3.4.6. Policy Validation

Peer-to-peer networks can be used legitimately, or misused by botnet, or violate network usage policy. Section 3.2.1 details peer-to-peer classification using NetFlow data. *Krmicek et al.* [70] proposed an approach to detect the use of unauthorized Network Address Translation (NAT) via a heuristic method based on NetFlow data. NetFlow data can also provide information about legitimate flows denied by the security policy and help network administrator with troubleshooting. *Frias-Martinez* [43] proposed a behavior-based network access control mechanism with a true rejection rate of 95%.

3.5. ***Issues of Data Error***

NetFlow data is exported using UDP. Data can be lost due to overloaded segments between routers and collectors, an overload of collectors with benign traffic increases, burst nature of NetFlow traffic, or attacks in progress. Similarly, errors may happen in the process of sampling, transporting and collecting. In order to address these problems, several methods have been proposed. *Cohen et al.* [24] proposed a framework for calculating confidence intervals to address the estimation errors in a multistage combination of sampling and aggregation. *Trammell et al.* [132] characterized, quantified, and corrected timing errors, which are consequence of Cisco NetFlow version 9 protocol design that estimates the true base time from derived base time information. *Rohmad et al.* [112] proposed an enhanced NetFlow version 9 using nProbe GPL. *Fioreze et al.* [38] investigated the trustfulness of NetFlow measurements and found that octets and packets are reliably reported but the flow duration of samples are shorter than the actual duration. *Zhu et al.* [159] studied the errors of utilized bandwidth measurement of NetFlow and provided guidance for correctly estimating the utilized bandwidth. Finally, *Ricciato et al.* [111] described a methodology to estimate one-way packet loss from IPFIX or NetFlow flow records.

4. Methodologies

In this section, we review various methodologies used to analyze NetFlow data. Figure 5 provides a chronological summary of the methodologies discussed in this section. As it can be observed, a considerable number of studies have focused on using machine learning algorithms and real time analysis.

Table 1: Summary of Security Awareness and Intrusion Detection

Year	Methodology	Perspective
2001 [35]	Histogram and chart	IDS
2001 [69]	Statistic	DoS and DDos
2004 [151]	Links between machines or domains	IDS
2004 [64]	Statistic patterns	DoS and DDoS
2005 [29]	Host behavior based	Worm outbreaks
2005 [30]	IP aggregation	Detection and monitoring
2006 [110]	Flow aggregation	IDS
2007 [109]	Trust and reputation model	IDS
2007 [28]	Flow signature and honeypot logs	Worm detection
2008 [18]	Heuristics	Worm detection
2008 [158]	Statistic	Anomaly detection
2009 [70]	Heuristics	NAT detection
2009 [137]	Decision tree	Dictionary attack
2009 [43]	K-means	Behavior-based NAC
2009 [150]	Information theory	Risk detection
2009 [155]	Statistic	Top N detection
2009 [136]	Statistic	Spam machines
2010 [135]	NBA	Malware
2010 [54]	Spatial-temporal aggregating	Malicious website detection
2011 [114]	Statistic of host behavior	Attack Detection
2011 [61]	Dynamic entropy	DoS
2011 [44]	Statistic	DDoS and port scan
2011 [41]	Host behavior and PageRank	Botnets detection
2011 [125]	Time series	IDS
2011 [42]	PageRank	Botnets detection

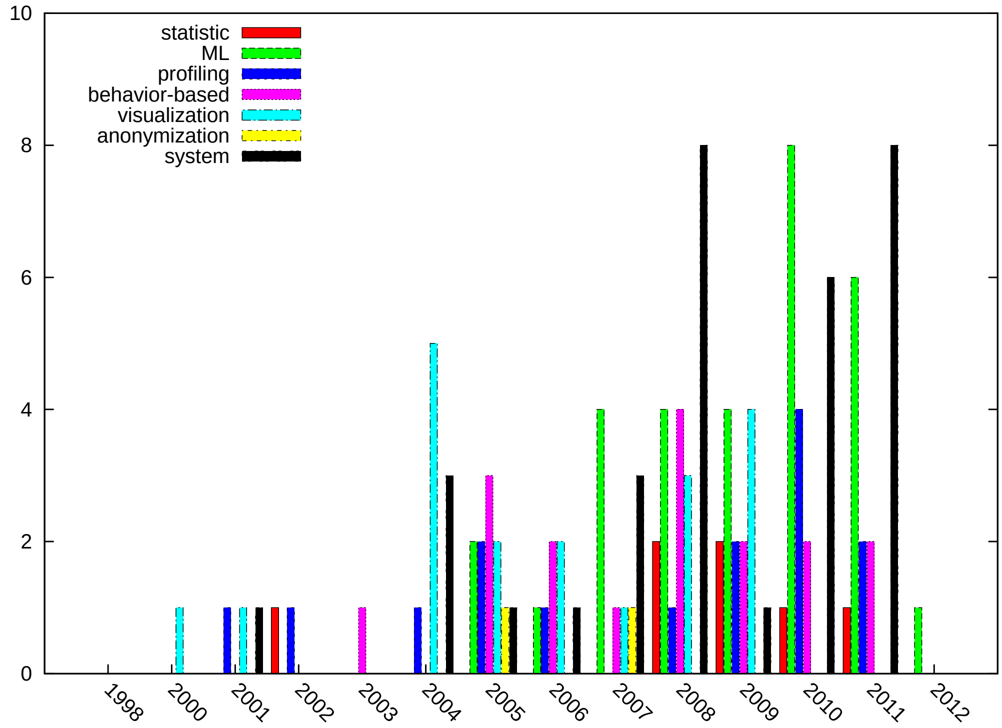

Figure 5: Methods by Years

4.1. ***Statistics***

Statistic approaches are the most common method in NetFlow analysis. In general, it is the basic step before applying heuristic-based approaches, machine learning and visualization. NetFlow data contains statistics of network flow information generated and exported from routers. *Duffield et al.* [34] investigated the resource usage of NetFlow formation and exportation as well as statistical properties of original traffic from sampled traffic data. *Proto et al.* [106] proposed a statistical model for network intrusion detection system. *Sawaya et al.* [114] proposed an approach of attack detection based on traffic flow statistics of hosts. *Barsamian* [10] proposed a botnet detection method using statistical signatures. *Liu et al.* [12] proposed an analysis and monitoring system using NetFlow statistic, and an IDS based on variance similarity.

Compared to other approaches, statistical approaches are usually easier to implement, provide accurate results and consume less resources. However, statistical approaches are good only for known cases and lack the ability to adapt to new cases.

*4.2. **Machine Learning***

Machine learning represents a collection of methods for discovering knowledge by searching for patterns. Machine learning refines and improves its knowledge base by learning from experience. The basic learning types are listed below:

- *Classification*: classify inputs to labeled outputs.
- *Clustering*: group inputs into clusters.
- *Association*: discover interesting relations between features.
- *Prediction*: predict outcome in terms of a numeric quantity.

Machine learning schemes include information theory, neural networks, support vector machines, genetic algorithms, and many more [123]. Machine learning applications require the collection of training and test data sets and depend on algorithms for feature extraction, feature selection, and learning. Initially, the system is trained using example data to learn specific data associations; then, the system is deployed in a similar environment where test data is used for classification. In this section, we provide a survey of machine learning approaches in NetFlow applications, which include traffic classification, anomaly detection, and security awareness.

Selecting an appropriate set of features for a specific problem is critical. Example of features are shown in Tables 2 and 3 and categorized as: (1) *basic features* such as NetFlow data fields, source and destination IP address and port, network interface, transport protocol, type of service, start and finish timestamps, cumulative TCP flags, number of bytes and packets transmitted, and MPLS labels; (2) *derived features* such as flow length (finish time - start time), average packet size (bytes / number of packets), average flow rate (bytes / length), average packet rate (number of packet / length), aggregation of IP subnet and traffic load bytes, percentage of traffic load on a node, percentage of traffic load at the current sub-tree with time period and aggregation threshold [140, 141]; (3) *application specific heuristics* such as webmail traffic [116] that has properties as close service proximity, daily and weekly pattern, and duration of client session; and (4) *advanced features* such as abacus signature, degree distribution, self-similarity of flow interval, entropy, kernel function, mutual information and Hellinger distance [134], or data fusion with other log files such as Snort, DNS related requests [7] (number of DNS requests, response, normals, and anomalies for each host over a certain period of time).

Methods for feature selection include symmetric uncertainty [57], information gain [127], subgroup, keyword selection, gradually reduction based on efficiency [83], and rough sets. The type of data sets and features being employed are very important for a successful machine learning approach. Typically, a large dataset is necessary to cover various relations in the data, including temporal and spatial relations. Training data has to be attack-free or attack-specific, both of which are difficult to obtain. The data sets in Table 3 can be categorized as (a) internet backbone of more than one week period, (b) internet backbone of less than one week period, (c) intranet of more than two weeks, and (d) simulated data or honeypot log.

4.2.1. Application Classification

Nguyen et al. [98] surveyed the application of machine learning techniques for traffic classification from 2004 to 2007; Even though NetFlow was not specified as analysis data set, but the basic methodologies are applicable to NetFlow data. *Kim et al.* [63] conducted an evaluation of traffic classification using traces with collected payloads. Their evaluation included seven machine learning algorithms: *Naive Bayes (NB), Naive Bayes Kernel Estimation (NBKE), Bayesian Network (BN), C4.5 Decision Tree (DT), k-Nearest Neighbors, Neural Networks, and Support Vector Machines (SVM).* They concluded that SVM consistently achieved higher accuracy. *Soysal et al.* [124] conducted more specific evaluations and comparisons of *BN, DT* and *Multilayer Perceptons* on flow-based network traffic classification using flow trace data. They concluded that *BN* and *DT* are suitable for internet traffic flow classification.

Nor et al. [99] evaluated a large number of machine learning algorithms in terms of their performance on NetFlow data with the objective of classifying HTTP, gmail, and video streaming. The highest accuracy machine learning algorithms had an accuracy more than 99.33%. Unfortunately, they did not provide information about the features used. Table 2 summarizes the algorithms, accuracy, features and data types for traffic classification using NetFlow data. Since accuracy varies considerably, there is a need to evaluate these algorithms and features on the same data set.

Table 2: Summary of Machine Learning Approaches of Network Application Classification

Year	Algorithm	Accu.(%)	Feature	Application
2007 [57]	NBKE	91	Basic[a] and derived[b]	P2P, email, Multimedia
2009 [17]	DT	90	Basic	P2P, VoIP, DNS, email, FTP
2010 [19]	Clustering	90	Application[c]	SNMP, email, DNS, IRC
2010 [113]	SVM	90	Advanced[d]	P2P
2010 [116]	SVM	94	Application	Webmail
2010 [9]	DT	90	Basic and derived	P2P, HTTP, VoIP, DNS, FTP, email, games
2011 [134]	SVM	70	Advanced	P2P
2012 [81]	BN	95	Derived	BULK, email, P2P

[a] Basic NetFlow data fields

[b] Calculation and aggregation of basic features

[c] Application specific properties from basic and derived features

[d] Abstract information from basic and derived features

4.2.2. Security Awareness and Anomaly Detection

Table 3 provides a summary of machine learning algorithms for anomaly detection in terms of algorithms, features, and research perspectives. The highest reported detection rate is 98% [146]. *Sommer et al.* [123] found that applying machine learning for network anomaly detection is harder than in other domains. This is mainly due to the great variety of traffic and the fundamental nature of machine learning approaches that are better suited at finding similarities than identifying relationships that are not present in the training data.

4.3. ***Profiling***

Network profiling is an important step for further analysis. Various profiling levels have been discussed in the literature including user, application, host, and network profiling.

- *User profiling*: There is limited work on the user profiling based on NetFlow data. *Melnikov et al.* [87] proposed a set of correlation and distribution of user flow data related to time and packet to identify a

Table 3: Summary of Machine Learning Approaches of Anomaly Detection

Year	Algorithm	Feature	Dataset	Perspective
2005 [72]	Cluster	Advanced[a]	Internet	Anomaly
2007 [83]	Multiclass SVM	Advanced	internet	NSSA
2008 [142]	GA-based	Derived[b]	non-NetFlow[c]	DDoS
2010 [141]	Kernel	Derived	Internet	Monitoring
2010 [127]	SVM	Derived	Intranet	Masquerade
2011 [7]	SVM	Application[d]	non-NetFlow	Worm
2011 [140]	SVM	Derived	Ineternet	Attacks
2011 [139]	SVM	Advanced	Internet	Attacks
2011 [146]	SVM	Basic[e] and derived	non-NetFlow	IDS

[a] Abstract information from basic and derived features

[b] Calculation and aggregation of basic features

[c] Simulation or log data

[d] Application specific properties from basic and derived features

[e] Basic NetFlow data fields

users. Different user behavior based approaches have employed various features that are discussed in section 4.4. User profiling research, however, may provide helpful information in future.

- *Application profiling*: *Liu and Huebner* [82] discussed the stochastic characteristics of some of the most popular applications (i.e., FTP, HTTP, SNMP, NNTP, DNS, and napster): flow length and time by probability density function and tail distribution, average packet size distribution, and average throughput distribution. *Karagiannis et al.* [60] proposed traffic patterns of social behavior, function (provider or consumer), and application ports that were used to classify traffic based on heuristic rules.

- *Host profiling*: *Wei et al.* [143] proposed an approach for internet host profiling using a data structure that can be expressed in XML-like format at listing 1, where the communication similarity is the average of Dice similarity values for the host. *Kuai et al.* [148] proposed an approach based on bipartite graphs to represent host communication and one-mode projection of bipartite graphs to capture the social-behavior similarity of end hosts as figure 6. For networks with few hosts, we need more detailed information for further analysis. *Minarik et al.* [89] proposed a host behavior profiling based on the bi-directional NetFlow that use communicating peers (number of servers contacted, clients answered, and single flows), amount of traffic (amount of requests, replies, and single flows), and traffic structure (number of client, server and single flows). *Frias-Martinez et al.* [43] defined a host behavior profile that contains seven features: the total number of flows, average flow size, average flow duration, total number of packets contained in all flows, average number of packets per flow, total number of unique IP addresses contained in all flows, and average packet size.

- *Network profiling*:*Cho et al.* proposed *Aguri* tree [22], an aggregation-based traffic profile that aggregates small volume flows with a fixed number of nodes in an IP tree for spatial measurement. *Jiang et al.* [56] characterized network prefix-level traffic profiling as daily traffic volume, distributions (over time, direction, applications, and flow size), and ratio of upload-download. *Lakhina et al.* [73] described *Origin-Destination* flows using a routing metric, and further analyzed using *Aguri* tree to include time, features (i.e., source and destination address, source and destination port) and volume to represent both time and space attributes [72] .

4.4. ***Behavior-based Approaches***

Recently, behavior-based approaches to network security have received attention [47]. Compared to signature-based approaches, behavior-based approaches first learn normal behaviors, and then detect anomalies. This approach has been applied with many research perspectives: application classification, anomaly detection, zero day attack detection, network access control [43], and network design [121]. Types of behavior-based approaches include threshold, statistical and learning-based. Levels of behavior-based approaches include ISP-based internet backbone behavior [29, 148, 149], network behavior [29, 108, 136], user behavior [87], host behavior [29, 60, 84, 130, 135, 148] and application (or protocol) behavior [60, 76, 124].

```
<host>
  ip_address
  daily_destination_number
  daily_byte_number
  average_TTL
  <tcp_service>port1,port2, ...</tcp_service>
  <udp_service>port1,port2, ...</udp_service>
  <communication>
    <tcp_communication>
      destination_address
      daily_byte_num
      daily_connection_num
      average_duration_time
      <port>port1,port2, ...</port>
    </tcp_communication>
    <udp_communication>
      destination_address
      daily_byte_num
      daily_packet_number
      <port>port1,port2, ...</port>
    </udp_communication>

  </communication>
  communication_similarity
</host>
```

Listing 1: Internet Host Profile (Courtesy of [143])

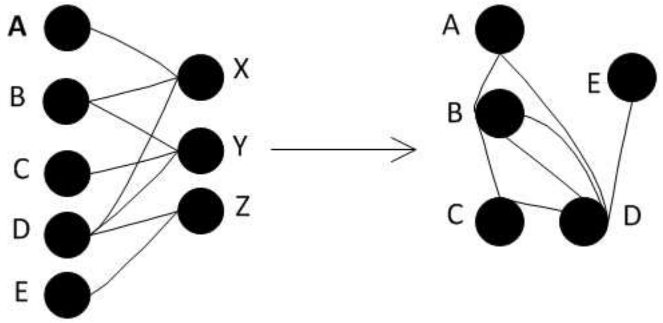

Figure 6: Bipartite Graph (left) and One-mode Projection (right)

4.5. ***Visualization***

Network visualization provides interactive visual displays for exploration of network traffic. Visualizing a large amount of information and providing sufficient level of detail to be meaningful and useful is a challenging task. Visualization can be at different levels of network abstraction (i.e., whole network, individual machine, and between whole network and individual machine), and described by different mechanisms (histogram, chart or glyph-based and 3D graph). Table 4 presents a chronological summary of related studies with their abstract level, mechanism of data processing and visualization, and research issues. Most applications use statistics and chart methods; whereas few applications use advanced methodologies such as machine learning, graph theory and quad-tree. In terms of research perspectives, most of them focus on security detection while others provide network monitoring.

Besides the approaches summarized in table 4, several other projects are worth mentioning. *NFSen* [5] is an open source, graphical web based front-end tool. It aggregates network traffic by protocols, direction or hosts using charts, and is used for network investigation. *AURORA* [1] is an IBM research project for traffic analysis and visualization. It was designed for large networks, supports multiple levels of abstraction, and uses chart and graph to visualize traffic, anomaly detection or real time traffic flow. Finally, the Spinning Cube of Potential Doom [75] is a 3D display of network links for anomaly detection and visualized as a cube.

4.6. ***Anonymization***

There is a need to anonymize NetFlow data to protect the privacy when the data is shared among parties. There are several approaches for NetFlow specific anonymization. *Slagell et al.* [122] proposed an anonymization tool for sharing network logs for computer forensics. Their tool can anonymize the common fields in multiple ways. Similiarly, *Foukaraki et al.* [40] proposed an anonymization tool with flexible features and high-performance.

4.7. ***Analysis Systems***

As the traffic volume is very large, methodologies to improve performance of capturing, collection, and analysis are needed. There are three commonly used methods to reduce data size: aggregation, filtering, and sampling [31]. In the following, we will survey optimization, sampling, and distributed analysis systems.

Table 4: Summary of NetFlow Visualization Applications

Year	Abstract	Mechanism	Perspective
2000 [105]	Network	Aggregate traffic volume of protocols, chart	Network protocol and traffic amount
2001 [35]	Multiple	Histogram and chart	IDS
2004 [151]	Multiple	Links between machines or domains, graph	IDS
2004 [74]	Multiple	Activities of IP, histogram, glyph-based graph	Security situational awareness
2004 [8]	Multiple	Map between internal and external traffic, graph	Security
2004 [86]	Network	Aggregate based on port, chart and graph	Security event detection
2005 [30]	Network	IP aggregation of traffic bursts, chart & graph	Worm detection and backbone monitoring
2005 [103]	Network	Manifold learning, chart	Monitoring, detection
2006 [100]	Individual	Extended the quad-tree,3D navigation and playback	Internet traffic
2006 [101]	Network	Network statistics of protocols, chart	Network statistics
2006 [110]	Multiple	Statistic, flow aggregation, chart & graph	IDS
2007 [84]	Multiple	Host behavior regard protocols, Force-directed graph	Host behavior of security
2008 [88]	Individual	Graph theory, graph	Network traffic
2008 [39]	Network	TreeMap with splines, chart and graph	Network security monitor
2008 [130]	Multiple	Aggreate data per port, 3D graph	Intrusive behavior
2009 [120]	Network	Based on Simple K-Means clustering, chart	Detect anomalies
2009 [23]	Network	Pattern of shape, graph	Network attacks
2009 [131]	Multiple	Aggregate and Map, graph and chart	Network monitoring
2009 [48]	Multiple	Aggregate, tree view, Geolocation, chart and graph	Network security
2012 [118]	Multiple	Sphere	Network traffic

4.7.1. Optimization

Optimization can be applied in many stages of the NetFlow analysis process: capturing, collecting and analyzing. *Bouhtou and Klopfenstein* [14] proposed mathematical models to select the NetFlow interfaces based on robust optimizations to deal with probabilistic constraints. *Sagnol et al.* [115] proposed a method of *Successive c-Optimal Design* to select NetFlow interfaces and find the optimal sampling rates. *Hu et al.* [55] proposed an entropy based adaptive flow aggregation algorithm to improve efficiency of storage and export, and improve the accuracy of legitimate flows. *Zadnik et al.* [152] proposed an architecture of network flow monitoring adapter based on hardware platform *COMBO6*, which is able to monitor one million simultaneous flows on an 2Gbps link. *Nagaraj et al.* [97] proposed an efficient aggregation techniques to speed up queries based on attributes and filter condition of queries.

4.7.2. Sampling

Sampling network flow reduces the burden of handling massive volumes of flow data in collection, storage and analysis. *Duffield* [31] conducted a review of internet measurement sampling in 2004, focusing on classical sampling methods, new applications and sampling methods, and applications areas. In 2007, *Haddadi et al.* revisited the issues of NetFlow sampling which focuses on data distortion and techniques for the compensation of data distortion.

Sampling methods, impact of sampling, integration of system-wide sampling, and recovering sampled data from distortion are mentioned in below studies. *Duffield et al.* [31, 33] developed a size-dependent sampling scheme suitable for billing purposes. *Estan et al.* [36] proposed an Adaptive NetFlow which dynamically adapts the sampling rate to achieve robustness without sacrificing accuracy. *Brauckhoff et al.* [15] evaluated the impact of sampling on anomaly detection metrics using flows with the *Blaster* worm, and found that entropy-based features are less affected. *Barlet-ros et al.* [9] analyzed the impact of sampling on the accuracy of traffic classification using machine learning methods, and proposed a solution to reduce the impact. *Cheng et al.* [21] proposed a resource-efficient sampling system that combines three models: a pre-sampling model that records the estimated value rather than the measured value, a sampling and holding model that process the sampled packets to update the cache, and a non-uniform sampling model and keep the long flows in cache. *Hao et al.* [53] developed a sampling scheme based on sampling two-runs to improve time and memory efficiency. *Han et al.* [51]

proposed a *pFlours* tool that fetches a packet and performs sampling to eliminate the synchronization problem during network traffic sampling. *Duffield et al.* [32] discussed trajectory sampling, methods to eliminate duplications, and methods to join incomplete trajectories. *Sekar et al.* [117] presented a system-wide approach that samples as a router primitive. To identify high-rate flow, *Zhang el at.* [157] developed two methods: fixed sample size test which uses user specified accuracy, and truncated sequential probability test through sequential sampling. *Lee et al.* [77] proposed a method for related sampling where flows from the same application session are given higher probability. *Bartos et al.* [11] proposed adaptive, feature-aware statistical sampling techniques to reduce the impact of sampling on anomaly detection.

4.7.3. Distributed Analysis System

More applications demand real time analysis, advanced detection and classification. Centralized analysis systems face the difficulties of performance, scalability, and robustness. Although sampling provides an approach to reduce those burdens, there are tasks that can not be based on sampling data. Distributed systems provide new mechanisms for capturing, accounting and monitoring [94]. Several distributed analysis systems have been mentioned below. *Kitatuji et al.* [66] proposed a real-time system with a bit-pattern based flow definition and round-robin mechanism to balance packet steams. *Sekar et al.* [117] proposed cSamp, a monitoring tool based on a coordinating mechanism for flow sampling, hash-based packet selection, and workload distributed. *Morariu et al.* [95] proposed a distributed IP traffic analysis system. *DiCAP* [93], a flow capturing system, uses round-robin and a Distributed Hash Table (DHT) to distribute the workload and uses off-the-shelf hardware at network links. *DIPStorage* [91] is a distributed flow storage platform for IP flow records based on DHT. *SCRIPT* [92] is a distributed flow analysis framework that distributed flow records equally to multiple nodes. Others used peer-to-peer communication infrastructure [41, 45] and *map-reduce* for efficient computation. Recent studies use the existing *Hadoop* based clustering platform and the *map-reduce* framework. *Lee et al.* [79] proposed using *Hadoop* based *map-reduce* to process packet trace files. *Francois et al.* [41] proposed botnet detecting system based on *Hadoop* based clustering and *PageRank*. *Morken* [96] compared two *map-reduce* frameworks of *Apache Hadoop* and *Nokia Disco*, and concluded that *Nokia Disco* provides fast response time while *Hadoop* provides rich features, and map-reduce model is a very good approach for flow filtering and aggregation.

5. Discussion

Even though a large body of research has focused on traffic flows, many issues remain open. In particular, NetFlow data analysis is challenging because of the difficulty in collecting real data, huge data sets with limited information, and lack of systematic methodologies. In this section, we discuss our view about datasets, research perspectives, methodologies, challenges, and possible future research directions.

5.1. ***Datasets***

Because of privacy and other concerns, researchers lack effective traffic flow data sets. Simulated data and other log data have been used as alternatives. Even though there is some real data, this data is either old or does not cover a large enough time period. Acquiring training data sets is another challenge for supervised machine learning. Moreover, there is no publicly available data for comparing different methodologies. Accurate analysis depends on real-time data collection. In surveyed papers , very few discuss a real time data collection solution [45].

Despite the popularity of sFlow and its wide deployment, few papers have focused on sFlow as their data source.

5.2. ***Research Perspectives***

Current studies have covered most perspectives of network monitoring, measurement, and network security. Application of network data in network monitoring is more successful than in network security, while real time network security is in high demand for network management. Basic top N data is not enough to understand the current complex network security situations. More specific perspectives such as referring user identity will provide clear information for security and forensic purpose. New perspectives will probably from network security because network security is becoming more important and challenge.

5.3. ***Methodologies***

Heuristic approaches are easier to implement and seem more effective than machine learning approaches; however, practical experiences and findings are difficult to gain. Statistical approaches with heuristic methods give accurate results for known situations. For situations involving anomaly, more research is needed to develop advanced approaches leveraging information theory, machine learning and data mining. Much of the work has been been limited to specific problems such as port-scan, DoS, or worms. A system that covers wide network security situations is needed for network security administrators. Moreover, visualization needs to focus more on IT operations and provides easy to understand and helpful information. For machine learning approaches, feature selection is a very important step that needs to be specific to the problem. Currently, there is no study available for understanding and comparing the affect of feature selection in the context of NetFlow data. Integrating data from other IT infrastructures will provide more information. As there is no publicly available dataset for comparing different approaches, researchers use their own private data sets in their experiments.

5.4. ***Challenges***

With the constantly changing nature of networks and new applications and protocols being added to the internet, network analysis will have to keep up with the speed of change. For example, IPv6 addresses can be randomly generated and may not be identified as a unique host or user. Since IPv6 over IPv4 packets can bypass firewalls [50], new approaches for IPv6 measurements are needed. New applications and protocols, faster internet speeds with increased backbone bandwidth, and more complex content will make the analysis more difficult. In particular, the cloud computing that is based on moving contents to cloud services will make the analysis more complicated. In the following, we discuss specific challenges.

5.4.1. Feature Representation and Selection

Representing and selecting a set of appropriate features is challenging because NetFlow data provides the header information only. For a specific task, the key question is how to effectively represent and extract features, and how to select the right features for a specific problem. With NetFlow version 9, it would be important to effectively leverage these new information.

5.4.2. Real Time Analysis and Data Storage

Analysis results need to be available in real time or within some fairly short period of time as the traffic is flowing. Furthermore, data needs to be continuously stored for certain amount of time for future need. Real time data collection is a challenging task because of the data size and the nature of the network traffic. Real time analysis requires understanding the dynamic nature of network traffic. As David [144] pointed out "that is the face of knowledge in the age of the Net: never fully settled, never fully written, never entirely done".

5.5. ***Future Directions***

Despite significant work in the field, future research is needed to address the above mentioned challenges.

5.5.1. Distributed Data Collection and Analysis

Real time analysis is in high demand in network security. Centralized analysis systems have difficulty dealing with huge data and real time analysis. Scalability and robustness is required to analyze data from multiple collectors. New technologies, such as *Apache Hadoop* related distributed data collection and analysis systems, open up more opportunities for re-thinking the network traffic analysis. Distributed applications and *map-reduce* model will provide more power and bring more insight and understanding.

5.5.2. Advanced Analysis Methodologies

Advanced methodologies using behavior-based features have the potential to mine helpful information. As *Sommer et al.* [123] pointed out, machine learning algorithms excel at finding similarity rather than at identifying anomalous behaviors. To make machine learning approaches more accurate and efficient, there is a need for better understanding of different types of features and heuristics for specific goals. In practice, selecting and understanding an effective set of features is challenging and labor-intensive.

5.5.3. Integration

Integration with existing network infrastructures (e.g., IDS,firewall and VPN gateway), integration with log file event activities, as well as integration with host IDS (e.g., meta-events) all show a trend. NetFlow analysis can fill in the gap that IDS, firewall and host-based anti-virus tools can not provide. It can provide monitoring, reporting, security altering, validating policy and configuration, assisting for forensic investigation, and serving as complimentary approaches for other network applications. Correlation with existing network infrastructures (e.g., NIDS may alert for an attack then NetFlow data will validate the alert) can give a high probability factor to remove false positives. *Liu et al.* [83] proposed a method using Snort logs and NetFlow data fusion with SVMs to create network security awareness. Integrated together with other approaches (such as deep packet inspection) NetFlow-like approaches can provide a breadth-first approach for early investigation, and cover more hierarchies of network layers.

6. Conclusion

In this paper, we performed a comprehensive survey of network flow applications. First, we provided a brief background information on sFlow, NetFlow, and network traffic analysis. We covered the state of the art in network monitoring, analysis and management, application classification, user identity inferring, and network security awareness. We found that network security has been an important research topic, and has covered various aspects of network security issues. We then surveyed the state of the art of methodologies related to statistics, machine learning, profiling, behavior-based approaches, visualization, anonymization, and analysis systems. We found that advanced methodologies such as machine learning has been an important approach, applied mostly on application classification and network security awareness. Then, we critiqued the surveyed research with emphasis on data sets, research perspectives, methodologies, challenges, and pointed out possible directions for future research.

Acknowledgments:

We are grateful to Dong Yu of Microsoft Research for valuable feedback. George Bebis is a Visiting Professor in the Department of Computer Science at King Saud University (KSU), Riyadh, Saudi Arabia.

References

[1] AURORA: Traffic analysis and visualization, http://www.zurich.ibm.com/aurora/, June 27, 2012.

[2] Internet Traffic Classification, http://www.caida.org/research/traffic-analysis/classification-overview/, June 27, 2012.

[3] IPFIX, http://datatracker.ietf.org/wg/ipfix/, June 27, 2012.

[4] NetFlow applications, http://netflow.caligare.com/applications.htm, June 27, 2012.

[5] NFSen - Netflow Sensor, http://nfsen.sourceforge.net/, June 27, 2012.

[6] sFlow Collectors, http://www.sflow.org/products/collectors.php, June 27, 2012.

[7] S. A. Abdulla, S. Ramadass, A. Altaher, and A. A. Nassiri. Setting a Worm Attack Warning by using Machine Learning to Classify NetFlow Data. *International Journal of Computer Applications*, 36(2):49–56, Dec. 2011.

[8] R. Ball, G. A. Fink, and C. North. Home-centric visualization of network traffic for security administration. In *Proceedings of the 2004 ACM workshop on Visualization and data mining for computer security*, VizSEC/DMSEC '04, pages 55–64, New York, NY, USA, 2004. ACM.

[9] P. Barlet-ros and A. Cabellos-aparicio. Analysis of the impact of sampling on NetFlow traffic classification. *Methodology*, 55(5):1083–1099, 2010.

[10] A. V. BARSAMIAN. Network characterization for botnet detection using statistical-behavioral methods. Master's thesis, Thayer School of Engineering, Dartmouth College, USA, June 2009.

[11] K. Bartos, M. Rehak, and V. Krmicek. Optimizing flow sampling for network anomaly detection. In *Wireless Communications and Mobile Computing Conference (IWCMC), 2011 7th International*, pages 1304–1309, July 2011.

[12] L. Bin, L. Chuang, Q. Jian, H. Jianping, and P. Ungsunan. A NetFlow based flow analysis and monitoring system in enterprise networks. *Computer Networks*, 52(5):1074–1092, 2008.

[13] X. Bo, C. Ming, L. Fei, and W. Na. P2P flows identification method based on listening port. In *Broadband Network Multimedia Technology, 2009. IC-BNMT '09. 2nd IEEE International Conference on*, pages 296–300, 2009.

[14] M. Bouhtou and O. Klopfenstein. Robust Optimization for Selecting NetFlow Points of Measurement in an IP Network. *IEEE GLOBECOM 20072007 IEEE Global Telecommunications Conference*, pages 2581–2585, 2007.

[15] D. Brauckhoff, B. Tellenbach, A. Wagner, M. May, and A. Lakhina. Impact of packet sampling on anomaly detection metrics. *Proceedings of the 6th ACM SIGCOMM on Internet measurement IMC 06*, page 159, 2006.

[16] A. Caracas, A. Kind, D. Gantenbein, S. Fussenegger, and D. Dechouniotis. Mining semantic relations using NetFlow. In *Business-driven IT Management, 2008. BDIM 2008. 3rd IEEE/IFIP International Workshop on*, pages 110–111, Apr. 2008.

[17] V. Carela-Espanol, P. Barlet-Ros, and J. Solé-Pareta. Traffic classification with sampled netflow. Technical Report 2, Technical report, Universitat Politecnica de Catalunya, 2009, 2009.

[18] Y.-T. Chan, C. A. Shoniregun, and G. A. Akmayeva. A NetFlow based internet-worm detecting system in large network. In *Digital Information Management, 2008. ICDIM 2008. Third International Conference on*, pages 581–586, 2008.

[19] U. K. Chaudhary, I. Papapanagiotou, and M. Devetsikiotis. Flow classification using clustering and association rule mining. In *Computer Aided Modeling, Analysis and Design of Communication Links and Networks (CAMAD), 2010 15th IEEE International Workshop on*, pages 76–80, 2010.

[20] Y. Chen, S. Jain, V. K. Adhikari, Z.-l. Zhang, and K. Xu. A First Look at Inter-Data Center Traffic Characteristics via Yahoo ! Datasets. *wwwuserscsumnedu*, pages 1620–1628, 2011.

[21] G. Cheng and J. Gong. A Resource-Efficient Flow Monitoring System. *Communications Letters, IEEE*, 11(6):558–560, June 2007.

[22] K. Cho, R. Kaizaki, and A. Kato. Aguri: An Aggregation-Based Traffic Profiler. In *Proceedings of the Second International Workshop on Quality of Future Internet Services*, COST 263, pages 222–242, London, UK, UK, 2001. Springer-Verlag.

[23] H. Choi, H. Lee, and H. Kim. Fast detection and visualization of network attacks on parallel coordinates. *Computers Security*, 28(5):276–288, 2009.

[24] E. Cohen, N. Duffield, C. Lund, and M. Thorup. Confident estimation for multistage measurement sampling and aggregation. *Proceedings of the 2008 ACM SIGMETRICS international conference on Measurement and modeling of computer systems SIGMETRICS 08*, (i):109, 2008.

[25] M. P. Collins and M. K. Reiter. Hit-list worm detection and bot identification in large networks using protocol graphs. In *Proceedings of the 10th international conference on Recent advances in intrusion detection*, RAID'07, pages 276–295, Berlin, Heidelberg, 2007. Springer-Verlag.

[26] L. Deri. ntop,http://www.ntop.org.

[27] L. Deri. Open Source VoIP Traffic Monitoring, 2006.

[28] F. Dressler, W. Jaegers, and R. German. Flow-based Worm Detection using Correlated Honeypot Logs, 2007.

[29] T. Dubendorfer and B. Plattner. Host Behaviour Based Early Detection of Worm Outbreaks in Internet Backbones. *14th IEEE International Workshops on Enabling Technologies Infrastructure for Collaborative Enterprise WETICE05*, (c):166–171, 2005.

[30] T. Dubendorfer, A. Wagner, and B. Plattner. A Framework for Real-Time Worm Attack Detection and Backbone Monitoring. *First IEEE International Workshop on Critical Infrastructure Protection IWCIP05*, pages 3–12, 2005.

[31] N. Duffield. Sampling for Passive Internet Measurement: A Review. *Statistical Science*, 19:472–498, 2004.

[32] N. Duffield and M. Grossglauser. Trajectory Sampling With Unreliable Reporting. *IEEE/ACM Transactions on Networking*, 16(1):37–50, Feb. 2008.

[33] N. Duffield, C. Lund, and M. Thorup. Charging from sampled network usage. In *Proceedings of the 1st ACM SIGCOMM Workshop on Internet Measurement*, IMW '01, pages 245–256, New York, NY, USA, 2001. ACM.

[34] N. Duffield, C. Lund, and M. Thorup. Properties and prediction of flow statistics from sampled packet streams. In *Proceedings of the 2nd ACM SIGCOMM Workshop on Internet measurment*, IMW '02, pages 159–171, New York, NY, USA, 2002. ACM.

[35] R. F. Erbacher. Visual Behavior Characterization for Intrusion Detection in Large Scale Systems. In *Proceedings of the IASTED International Conference On Visualization, Imaging, and Image Processing*, pages 54–59, 2001.

[36] C. Estan, K. Keys, D. Moore, and G. Varghese. Building a better NetFlow. *ACM SIGCOMM Computer Communication Review*, 34(4):245, 2004.

[37] F. Baker and B. Foster and C. Sharp. Cisco Architecture for Lawful Intercept in IP Networks, 2004.

[38] T. Fioreze, L. Z. Granville, A. Pras, A. Sperotto, and R. Sadre. Self-management of hybrid networks: Can we trust netflow data? *Integrated Network Management 2009 IM09 IFIPIEEE International Symposium on*, pages 577–584, 2009.

[39] F. Fischer, F. Mansmann, D. A. Keim, and S. Pietzko. Large-scale Network Monitoring for Visual Analysis of Attacks. *Visualization for computer security 5th international workshop VizSec 2008 Cambridge MA USA September 15 2008 proceedings*, 72(1-3):1–8, 2008.

[40] M. Foukarakis, D. Antoniades, S. Antonatos, and E. P. Markatos. Flexible and high-performance anonymization of NetFlow records using anontool. *2007 Third International Conference on Security and Privacy in Communications Networks and the Workshops SecureComm 2007*, pages 33–38, 2007.

[41] J. François, S. Wang, R. State, and T. Engel. BotTrack: Tracking Botnets Using NetFlow and PageRank. *NETWORKING 2011*, 6640:1–14, 2011.

[42] J. Francois, S. Wang, W. Bronzi, R. State, and T. Engel. BotCloud: Detecting botnets using MapReduce. In *Information Forensics and Security (WIFS), 2011 IEEE International Workshop on*, pages 1–6, 2011.

[43] V. Frias-Martinez, J. Sherrick, S. J. Stolfo, and A. D. Keromytis. A Network Access Control Mechanism Based on Behavior Profiles. In *Computer Security Applications Conference, 2009. ACSAC '09. Annual*, pages 3–12, 2009.

[44] A. A. Galtsev and A. M. Sukhov. Network attack detection at flow level. *Aerospace*, 2011.

[45] L. Gao, J. Yang, H. Zhang, B. Zhang, and D. Qin. FlowInfra: A fault-resilient scalable infrastructure for network-wide flow level measurement. In *Network Operations and Management Symposium (APNOMS), 2011 13th Asia-Pacific*, pages 1–8, 2011.

[46] Y. Gao, Z. Li, and Y. Chen. A DoS Resilient Flow-level Intrusion Detection Approach for High-speed Networks. In *Distributed Computing Systems, 2006. ICDCS 2006. 26th IEEE International Conference on*, page 39, 2006.

[47] D. Geer. Behavior-based network security goes mainstream. *Computer*, 39(3):14–17, Mar. 2006.

[48] J. R. Goodall and M. Sowul. VIAssist: Visual analytics for cyber defense. In *Technologies for Homeland Security, 2009. HST '09. IEEE Conference on*, pages 143–150, May 2009.

[49] A. M. Gossett, I. Papapanagiotou, and M. Devetsikiotis. An apparatus for P2P classification in Netflow traces. In *GLOBECOM Workshops (GC Wkshps), 2010 IEEE*, pages 1361–1366, 2010.

[50] M. Gregr, P. Matousek, M. Sveda, and T. Podermanski. Practical IPv6 monitoring-challenges and techniques. In *Integrated Network Management (IM), 2011 IFIP/IEEE International Symposium on*, pages 650–653, May 2011.

[51] B.-J. Han, J.-H. Lee, S.-G. Sohn, J.-H. Ryu, and T.-M. Chung. pFlours: A New Packet and Flow Gathering Tool. In *Advanced Communication Technology, 2008. ICACT 2008. 10th International Conference on*, volume 1, pages 731–736, 2008.

[52] S.-h. Han, M.-s. Kim, H.-t. Ju, and J. W.-k. Hong. The Architecture of NG-MON: A Passive Network Monitoring System. In *InProceeding*

of 13th IFIP/IEEE International Workshop on Distributed Systems: Operations and Management, pages 16–27, 2002.

[53] F. Hao, M. Kodialam, T. V. Lakshman, and S. Mohanty. Fast, Memory Efficient Flow Rate Estimation Using Runs. *Networking, IEEE/ACM Transactions on*, 15(6):1467–1477, 2007.

[54] H.-W. Hsiao, D.-N. Chen, and T. J. Wu. Detecting hiding malicious website using network traffic mining approach. In *Education Technology and Computer (ICETC), 2010 2nd International Conference on*, volume 5, pages V5–276 –V5–280, June 2010.

[55] Y. Hu, D.-m. Chiu, J. C. S. Lui, and S. Member. Entropy Based Adaptive Flow Aggregation. *IEEE/ACM Transactions on Networking*, 17(3):698–711, 2009.

[56] H. Jiang, Z. Ge, S. Jin, and J. Wang. Network prefix-level traffic profiling: Characterizing, modeling, and evaluation. *Comput. Netw.*, 54(18):3327–3340, Dec. 2010.

[57] H. Jiang, A. W. Moore, Z. Ge, S. Jin, and J. Wang. Lightweight application classification for network management. *Proceedings of the 2007 SIGCOMM workshop on Internet network management INM 07*, page 299, 2007.

[58] W. Jinsong, L. Weiwei, Z. Yan, L. Tao, and W. Zilong. P2P Traffic Identification Based on NetFlow TCP Flag. In *Future Computer and Communication, 2009. ICFCC 2009. International Conference on*, pages 700–703, Apr. 2009.

[59] A. J. Kalafut, J. Van Der Merwe, and M. Gupta. Communities of Interest for Internet Traffic Prioritization. *IEEE INFOCOM Workshops 2009*, pages 1–6, 2009.

[60] T. Karagiannis, K. Papagiannaki, and M. Faloutsos. BLINC: multilevel traffic classification in the dark. *SIGCOMM Comput. Commun. Rev.*, 35(4):229–240, Aug. 2005.

[61] Y. Ke-xin and Z. Jian-qi. A novel DoS detection mechanism. In *Mechatronic Science, Electric Engineering and Computer (MEC), 2011 International Conference on*, pages 296–298, 2011.

[62] D. R. Kerr and B. L. Bruins. Network flow switching and flow data export. Patent, 2001.

[63] H. Kim, K. C. Claffy, M. Fomenkov, D. Barman, M. Faloutsos, and K. Lee. Internet traffic classification demystified: myths, caveats, and the best practices. In *Proceedings of the 2008 ACM CoNEXT Conference*, CoNEXT '08, pages 11:1—-11:12, New York, NY, USA, 2008. ACM.

[64] M.-S. Kim, H.-J. Kong, S.-C. Hong, S.-H. Chung, and J. W. Hong. A flow-based method for abnormal network traffic detection. In *Network Operations and Management Symposium, 2004. NOMS 2004. IEEE/IFIP*, volume 1, pages 599 –612 Vol.1, Apr. 2004.

[65] A. Kind, D. Gantenbein, and H. Etoh. Relationship Discovery with NetFlow to Enable Business-Driven IT Management. In *Business-Driven IT Management, 2006. BDIM '06. The First IEEE/IFIP International Workshop on*, pages 63–70, Apr. 2006.

[66] Y. Kitatsuji and K. Yamazaki. A distributed real-time tool for IP-flow measurement. In *Applications and the Internet, 2004. Proceedings. 2004 International Symposium on*, pages 91–98, 2004.

[67] J. Kö andgel. One-way delay measurement based on flow data: Quantification and compensation of errors by exporter profiling. In *Information Networking (ICOIN), 2011 International Conference on*, pages 25–30, 2011.

[68] A. Kobayashi and K. Toyama. Method of Measuring VoIP Traffic Fluctuation with Selective sFlow. In *2007 International Symposium on Applications and the Internet Workshops*, pages 89–89. Ieee, 2007.

[69] C. Kotsokalis, D. Kalogeras, and B. Maglaris. Router-based Detection of DoS and DDoS Attacks. In *Proceedings of HP OpenView University Association HPOVUA 8th Annual Workshop*, 2001.

[70] V. Krmicek, J. Vykopal, and R. Krejci. Netflow based system for NAT detection. In *Proceedings of the 5th international student workshop on Emerging networking experiments and technologies*, pages 23–24. ACM, 2009.

[71] S. R. Kundu, S. Pal, K. Basu, and S. K. Das. An Architectural Framework for Accurate Characterization of Network Traffic. *IEEE Transactions on Parallel and Distributed Systems*, 20(1):111–123, 2009.

[72] A. Lakhina, M. Crovella, and C. Diot. Mining anomalies using traffic feature distributions. *SIGCOMM Comput. Commun. Rev.*, 35(4):217–228, Aug. 2005.

[73] A. Lakhina, K. Papagiannaki, M. Crovella, C. Diot, E. D. Kolaczyk, and N. Taft. Structural analysis of network traffic flows. *SIGMETRICS Perform. Eval. Rev.*, 32(1):61–72, June 2004.

[74] K. Lakkaraju, W. Yurcik, and A. J. Lee. NVisionIP: netflow visualizations of system state for security situational awareness. *Proceedings of the 2004 ACM workshop on Visualization and data mining for computer security*, 29:65–72, 2004.

[75] S. Lau. The Spinning Cube of Potential Doom. *Commun. ACM*, 47(6):25–26, June 2004.

[76] C.-y. Lee, H.-k. Kim, K.-h. Ko, and J.-w. Kim. A VoIP Traffic Monitoring System based on NetFlow v9. *International Journal of Advanced Science and Technology*, 4:1–8, 2009.

[77] M. Lee, M. Hajjat, R. R. Kompella, and S. Rao. RelSamp: Preserving application structure in sampled flow measurements. In *INFOCOM, 2011 Proceedings IEEE*, pages 2354–2362, Apr. 2011.

[78] M. Lee, S. Member, and N. Duffield. Opportunistic Flow-Level Latency Estimation Using Consistent NetFlow. *Networking IEEE/ACM Transaction On*, pages 1–14, 2011.

[79] Y. Lee, W. Kang, and Y. Lee. A hadoop-based packet trace processing tool. In *Proceedings of the Third international conference on Traffic monitoring and analysis*, TMA'11, pages 51–63, Berlin, Heidelberg, 2011. Springer-Verlag.

[80] Y. Li. Study of the monitoring model for securities trading network Quality of Service. In *Information Science and Engineering (ICISE), 2010 2nd International Conference on*, pages 1–4, 2010.

[81] C. Liang and G. Jian. Fast application-level traffic classification using NetFlow records. *Journal On Communications*, 33(1):145 – 152, 2012.

[82] D. Liu and F. Huebner. Application profiling of IP traffic. In *Local Computer Networks, 2002. Proceedings. LCN 2002. 27th Annual IEEE Conference on*, pages 220–229, 2002.

[83] X.-W. Liu, H.-Q. Wang, Y. Liang, and J.-B. Lai. Heterogeneous Multi-Sensor Data Fusion with Multi-Class Support Vector Machines: Creating Network Security Situation Awareness. In *Machine Learning and Cybernetics, 2007 International Conference on*, volume 5, pages 2689–2694, 2007.

[84] F. Mansman, L. Meier, and D. A. Keim. Visualization of Host Behavior for Network Security. *Network Security*, pages 187–202, 2007.

[85] F. Mansmann, F. Fischer, D. A. Keim, S. Pietzko, and M. Waldvogel. Interactive Analysis of NetFlows for Misuse Detection in Large IP Networks. In P. Müller, B. Neumair, and G. D. Rodosek, editors, *DFN-Forum Kommunikationstechnologien*, volume 149 of *LNI*, pages 115–124. GI, 2009.

[86] J. McPherson, K.-L. Ma, P. Krystosk, T. Bartoletti, and M. Christensen. PortVis: a tool for port-based detection of security events. In *Proceedings of the 2004 ACM workshop on Visualization and data mining for computer security*, VizSEC/DMSEC '04, pages 73–81, New York, NY, USA, 2004. ACM.

[87] N. Melnikov and J. Schönwälder. Cybermetrics: User Identification through Network Flow Analysis. In B. Stiller and F. De Turck, editors, *Mechanisms for Autonomous Management of Networks and Services*, volume 6155 of *Lecture Notes in Computer Science*, pages 167–170. Springer Berlin / Heidelberg.

[88] P. Minarik and T. Dymacek. NetFlow Data Visualization Based on Graphs. *Visualization for Computer Security*, pages 144–151, 2008.

[89] P. Minarik, J. Vykopal, and V. Krmicek. Improving Host Profiling with Bidirectional Flows. In *Computational Science and Engineering, 2009. CSE '09. International Conference on*, volume 3, pages 231–237, 2009.

[90] S. Moghaddam and A. Helmy. SPIRIT: A simulation paradigm for realistic design of mature mobile societies. In *Wireless Communications and Mobile Computing Conference (IWCMC), 2011 7th International*, pages 232–237, July 2011.

[91] C. Morariu, T. Kramis, and B. Stiller. DIPStorage: Distributed storage of IP flow records. In *Local and Metropolitan Area Networks, 2008. LANMAN 2008. 16th IEEE Workshop on*, pages 108–113, 2008.

[92] C. Morariu, P. Racz, and B. Stiller. SCRIPT: A framework for Scalable Real-time IP Flow Record Analysis. In *Network Operations and Management Symposium (NOMS), 2010 IEEE*, pages 278–285, Apr. 2010.

[93] C. Morariu and B. Stiller. DiCAP: Distributed Packet Capturing architecture for high-speed network links. In *Local Computer Networks, 2008. LCN 2008. 33rd IEEE Conference on*, pages 168–175, 2008.

[94] C. Morariu and B. Stiller. Distributed architecture for real-time traffic analysis. In *Proceedings of the Mechanisms for autonomous management of networks and services, and 4th international conference on Autonomous infrastructure, management and security*, AIMS'10, pages 171–174, Berlin, Heidelberg, 2010. Springer-Verlag.

[95] C. Morariu and B. Stiller. An open architecture for distributed IP traffic analysis (DITA). In *Integrated Network Management (IM), 2011 IFIP/IEEE International Symposium on*, pages 952–957, May 2011.

[96] J. T. Morken. Distributed NetFlow Processing Using the Map-Reduce Model. *Master's thesis, Norwegian University of Science and Technology*, 2010.

[97] K. Nagaraj, K. V. M. Naidu, R. Rastogi, and S. Satkin. Efficient Aggregate Computation over Data Streams. In *Data Engineering, 2008. ICDE 2008. IEEE 24th International Conference on*, pages 1382–1384, Apr. 2008.

[98] T. T. T. Nguyen and G. Armitage. A survey of techniques for internet traffic classification using machine learning. *Communications Surveys Tutorials, IEEE*, 10(4):56–76, 2008.

[99] S. M. Nor and A. B. Mohd. Towards a Flow-based Internet Traffic Classification for Bandwidth Optimization. *International Journal of Computer Science and Security IJCSS*, 3(3):146, 2009.

[100] J. Oberheide, M. Goff, and M. Karir. Flamingo: Visualizing Internet Traffic. In *Network Operations and Management Symposium, 2006. NOMS 2006. 10th IEEE/IFIP*, pages 150–161, Apr. 2006.

[101] A. Oslebo. Stager A Web Based Application for Presenting Network Statistics. In *Network Operations and Management Symposium, 2006. NOMS 2006. 10th IEEE/IFIP*, pages 1–15, Apr. 2006.

[102] T. Overview. Introduction to Cisco IOS ® NetFlow - A Technical Overview, 2007.

[103] N. Patwari, A. O. Hero, and A. Pacholski. Manifold learning visualization of network traffic data. *Proceeding of the 2005 ACM SIGCOMM workshop on Mining network data MineNet 05*, page 191, 2005.

[104] V. Perelman, N. Melnikov, and J. Schönwälder. Flow signatures of popular applications. In N. Agoulmine, C. Bartolini, T. Pfeifer, and D. O'Sullivan, editors, *Integrated Network Management*, pages 9–16. IEEE, 2011.

[105] D. Plonka. FlowScan: A Network Traffic Flow Reporting and Visualization Tool. In *Proceedings of the 14th USENIX conference on System administration*, pages 305–318, Berkeley, CA, USA, 2000. USENIX Association.

[106] A. Proto, L. A. Alexandre, M. L. Batista, I. L. Oliveira, and A. M. Cansian. Statistical Model Applied to NetFlow for Network Intrusion Detection. *Transactions on Computational Science*, 11:179–191, 2010.

[107] W. Qun, D. Xiuyue, and H. Lu. Novelty P2P Flow Analysis System. In *Wireless Communications, Networking and Mobile Computing (WiCOM), 2011 7th International Conference on*, pages 1–4, 2011.

[108] M. Rehak, M. Pechoucek, P. Celeda, V. Krmicek, M. Grill, and K. Bartos. Multi-agent approach to network intrusion detection. In *Proceedings of the 7th international joint conference on Autonomous agents*

and multiagent systems demo papers, pages 1695–1696. International Foundation for Autonomous Agents and Multiagent Systems, 2008.

[109] M. Rehak, M. Pechoucek, and P. Minarik. Collaborative Attack Detection in High-Speed Networks. *Analysis*, 4696 LNAI:73–82, 2007.

[110] P. Ren, Y. Gao, Z. Li, Y. Chen, and B. Watson. IDGraphs: intrusion detection and analysis using stream compositing. *Computer Graphics and Applications, IEEE*, 26(2):28–39, 2006.

[111] F. Ricciato, F. Strohmeier, P. Dorfinger, and A. Coluccia. One-way loss measurements from IPFIX records. In *Measurements and Networking Proceedings (M N), 2011 IEEE International Workshop on*, pages 158–163, 2011.

[112] M. S. Rohmad, F. Azmat, M. Manaf, and J.-l. Manan. Enhanced Netflow version 9 (e-Netflow v9) for network mediation: Structure, experiment and analysis. In *Information Technology, 2008. ITSim 2008. International Symposium on*, volume 3, pages 1–6, 2008.

[113] D. Rossi and S. Valenti. Fine-grained traffic classification with netflow data. *Proceedings of the 6th International Wireless Communications and Mobile Computing Conference on ZZZ IWCMC 10*, page 479, 2010.

[114] Y. Sawaya, A. Kubota, and Y. Miyake. Detection of Attackers in Services Using Anomalous Host Behavior Based on Traffic Flow Statistics. In *Applications and the Internet (SAINT), 2011 IEEE/IPSJ 11th International Symposium on*, pages 353–359, July 2011.

[115] D. Schatzmann, S. Leinen, J. Kögel, and W. Mühlbauer. FACT: Flow-Based Approach for Connectivity Tracking. In N. Spring and G. F. Riley, editors, *PAM*, volume 6579 of *Lecture Notes in Computer Science*, pages 214–223. Springer, 2011.

[116] D. Schatzmann, W. Mühlbauer, T. Spyropoulos, and X. Dimitropoulos. Digging into HTTPS: flow-based classification of webmail traffic. In *Proceedings of the 10th annual conference on Internet measurement*, IMC '10, pages 322–327, New York, NY, USA, 2010. ACM.

[117] V. Sekar, M. K. Reiter, W. Willinger, H. Zhang, R. R. Kompella, and D. G. Andersen. cSamp: A System for Network-Wide Flow Monitoring. In *Proc. 5th {USENIX} {NSDI}*, San Francisco, {CA}, Apr. 2008.

[118] D. S. Shelley and M. H. Gunes. Gerbilsphere: Inner sphere network visualization. *Computer Networks*, 56(3):1016 – 1028, 2012.

[119] W. Shen, Y. Chen, Q. Zhang, Y. Chen, B. Deng, X. Li, and G. Lv. Observations of IPv6 traffic. In *Computing, Communication, Control, and Management, 2009. CCCM 2009. ISECS International Colloquium on*, volume 2, pages 278–282, 2009.

[120] M. P. Singh, N. Subramanian, and Rajamenakshi. Visualization of flow data based on clustering technique for identifying network anomalies. In *Industrial Electronics Applications, 2009. ISIEA 2009. IEEE Symposium on*, volume 2, pages 973–978, 2009.

[121] A. Sinha, K. Mitchell, and D. Medhi. Flow-level upstream traffic behavior in broadband access networks: DSL versus broadband fixed wireless. In *IP Operations and Management, 2003. (IPOM 2003). 3rd IEEE Workshop on*, pages 135–141, 2003.

[122] A. J. Slagell and K. Luo. Sharing network logs for computer forensics: a new tool for the anonymization of netflow records. *Workshop of the 1st International Conference on Security and Privacy for Emerging Areas in Communication Networks 2005*, pages 37–42, 2005.

[123] R. Sommer and V. Paxson. Outside the Closed World: On Using Machine Learning for Network Intrusion Detection. *2010 IEEE Symposium on Security and Privacy*, pages 305–316, 2010.

[124] M. Soysal and E. G. Schmidt. Machine learning algorithms for accurate flow-based network traffic classification: Evaluation and comparison. *Performance Evaluation*, 67(6):451–467, June 2010.

[125] A. Sperotto and A. Pras. Flow-based intrusion detection. In N. Agoulmine, C. Bartolini, T. Pfeifer, and D. O'Sullivan, editors, *Integrated Network Management*, pages 958–963. IEEE, 2011.

[126] A. Sperotto, G. Schaffrath, R. Sadre, C. Morariu, A. Pras, and B. Stiller. An Overview of IP Flow-Based Intrusion Detection. *Communications Surveys Tutorials, IEEE*, 12(3):343–356, 2010.

[127] C. Strasburg, S. Krishnan, K. Dorman, S. Basu, and J. S. Wong. Masquerade Detection in Network Environments. In *Applications and the Internet (SAINT), 2010 10th IEEE/IPSJ International Symposium on*, pages 38–44, July 2010.

[128] F. Strohmeier, P. Dorfinger, and B. Trammell. Network performance evaluation based on flow data. In *Wireless Communications and Mobile Computing Conference (IWCMC), 2011 7th International*, pages 1585–1589, July 2011.

[129] A. M. Sukhov, D. I. Sidelnikov, A. A. Galtsev, A. P. Platonov, and M. V. Strizhov. Active Flows in Diagnostic of Troubleshooting on Backbone Links. *CoRR*, abs/0911.2, 2009.

[130] T. Taylor, S. Brooks, and J. McHugh. NetBytes Viewer: An Entity-Based NetFlow Visualization Utility for Identifying Intrusive Behavior. *VizSEC 2007*, pages 101–114, 2008.

[131] T. Taylor, D. Paterson, J. Glanfield, C. Gates, S. Brooks, and J. McHugh. FloVis: Flow Visualization System. In *Conference For Homeland Security, 2009. CATCH '09. Cybersecurity Applications Technology*, pages 186–198, Mar. 2009.

[132] B. Trammell, B. Tellenbach, D. Schatzmann, and M. Burkhart. Peeling Away Timing Error in NetFlow Data. In N. Spring and G. F. Riley, editors, *PAM*, volume 6579 of *Lecture Notes in Computer Science*, pages 194–203. Springer, 2011.

[133] P. Truong and F. Guillemin. A heuristic method of finding heavy hitter prefix pairs in IP traffic. *Communications Letters, IEEE*, 13(10):803–805, Oct. 2009.

[134] S. Valenti and D. Rossi. Identifying Key Features for P2P Traffic Classification. In *2011 IEEE International Conference on Communications ICC*, pages 1–6. IEEE, 2011.

[135] P. Čeleda, J. Vykopal, T. Plesník, M. Trunečka, and V. Krmíček. Malware Detection From The Network Perspective Using NetFlow Data. In *3rd NMRG Workshop on NetFlow/IPFIX Usage in Network Management*, 2010.

[136] G. Vliek. *Detecting spam machines, a netflow-data based approach.* PhD thesis, 2009.

[137] J. Vykopal, T. Plesnik, and P. Minarik. Network-Based Dictionary Attack Detection. In *Future Networks, 2009 International Conference on*, pages 23–27, Mar. 2009.

[138] A. Wagner, T. Dubendorfer, L. Hammerle, and B. Plattner. Flow-Based Identification of P2P Heavy-Hitters. *International Conference on Internet Surveillance and Protection*, 00(c):15–15, 2006.

[139] C. Wagner, J. François, R. State, and T. Engel. Machine learning approach for IP-flow record anomaly detection. In *Proceedings of the 10th international IFIP TC 6 conference on Networking - Volume Part I*, NETWORKING'11, pages 28–39, Berlin, Heidelberg, 2011. Springer-Verlag.

[140] C. Wagner, J. Francois, R. State, and T. Engel. DANAK: Finding the odd! In *Network and System Security (NSS), 2011 5th International Conference on*, pages 161–168, 2011.

[141] C. Wagner, G. Wagener, R. State, T. Engel, and A. Dulaunoy. Game theory driven monitoring of spatial-aggregated IP-Flow records. In *Network and Service Management (CNSM), 2010 International Conference on*, pages 463–468, 2010.

[142] S. Wang and R. Guo. GA-Based Filtering Algorithm to Defend against DDoS Attack in High Speed Network. In *Natural Computation, 2008. ICNC '08. Fourth International Conference on*, volume 1, pages 601–607, 2008.

[143] S. Wei, J. Mirkovic, and E. Kissel. Profiling and Clustering Internet Hosts. In *DMIN'06*, pages 269–275, 2006.

[144] D. Weinberger. The Machine That Would Predict the Future. *Scientific American*, 305(6):52–57, 2011.

[145] H. Weststrate. Botnet detection using netflow information Finding new botnets based on client connections. *Structure*, 2009.

[146] P. Winter, E. Hermann, and M. Zeilinger. Inductive Intrusion Detection in Flow-Based Network Data using One-Class Support Vector Machines. In *New Technologies, Mobility and Security (NTMS), 2011 4th IFIP International Conference on*, pages 1–5. 2011.

[147] B. Xu, M. Chen, and C. Hu. DEAPFI: A distributed extensible architecture for P2P flows identification. In *Network Infrastructure and Digital Content, 2009. IC-NIDC 2009. IEEE International Conference on*, pages 59–64, 2009.

[148] K. Xu, F. Wang, and L. Gu. Network-aware behavior clustering of Internet end hosts. In *INFOCOM, 2011 Proceedings IEEE*, pages 2078–2086, Apr. 2011.

[149] K. Xu, Z.-L. Zhang, and S. Bhattacharyya. Profiling internet backbone traffic: behavior models and applications. *SIGCOMM Comput. Commun. Rev.*, 35(4):169–180, Aug. 2005.

[150] K. Yin and T. Nianqing. Study on the Risk Detection about Network Security Based on Grey Theory. In *Information Technology and Applications, 2009. IFITA '09. International Forum on*, volume 1, pages 411–413, May 2009.

[151] X. Yin, W. Yurcik, M. Treaster, Y. Li, and K. Lakkaraju. VisFlowConnect: netflow visualizations of link relationships for security situational awareness. In *Proceedings of the 2004 ACM workshop on Visualization and data mining for computer security*, VizSEC/DMSEC '04, pages 26–34. ACM, 2004.

[152] M. Zadnik, T. Pecenka, and J. Korenek. Netflow probe intended for high-speed networks. In *Field Programmable Logic and Applications, 2005. International Conference on*, pages 695–698, 2005.

[153] Y. Zeng, X. Hu, and K. G. Shin. Detection of Botnets Using Combined Host- and Network-Level Information. *Symposium A Quarterly Journal In Modern Foreign Literatures*, pages 291–300, 2010.

[154] W. Zha and J. He. On campus network P2P and its link control. In *Consumer Electronics, Communications and Networks (CECNet), 2011 International Conference on*, pages 5086–5089, Apr. 2011.

[155] H. Zhang. Study on the TOPN Abnormal Detection Based on the NetFlow Data Set. *Computer and Information Science*, 2(3):103–108, 2009.

[156] J. Zhang and S. Meng. A design of NetFlow traffic statistic and analysis system for process of the transition of commercialization of IPV6. In *Computer Science and Service System (CSSS), 2011 International Conference on*, pages 963–965, June 2011.

[157] Y. B. Zhang, B. B. Fang, and H. Luo. Identifying high-rate flows based on sequential sampling. *Ieice Transactions On Information And Systems*, E93-D(5):1162–1174, 2010.

[158] W. Zhenqi and W. Xinyu. NetFlow Based Intrusion Detection System. *2008 International Conference on MultiMedia and Information Technology*, pages 825–828, 2008.

[159] H. Zhu, X. Zhang, and W. Ding. Research on Errors of Utilized Bandwidth Measured by NetFlow. In *Networking and Distributed Computing (ICNDC), 2011 Second International Conference on*, pages 45–49, 2011.

[160] Z. Zhu, G. Lu, Y. Chen, Z. J. Fu, P. Roberts, and K. Han. Botnet Research Survey. In *Computer Software and Applications, 2008. COMPSAC '08. 32nd Annual IEEE International*, pages 967–972, 2008.