# Data visibility methods in CPEs

Kimmo Virtanen

Master's thesis
November 2018
Technology, communication and transport
Master´s Degree Programme in Information Technology
Degree Programme in Cyber Security

jamk.fi

| Author(s) | Type of publication | Date |
|---|---|---|
| Virtanen, Kimmo | Master's thesis | February 2019 |
| | | Language of publication: English |
| | Number of pages 65 | Permission for web publication: |

Title of publication
**Data visibility methods in CPEs**

Degree programme
Master´s Degree Programme in Information Technology, Cyber Security

Supervisor(s)
Kotikoski, Sampo

Assigned by
AinaCom Oy

Abstract

Flow information is gathered from data packets traversing an observation point. Traditionally, flow information has been used for e.g. billing, capacity planning, and fault finding. Trending technologies such as software defined networking, machine learning and anomaly detection can gain advantage from flow based information.

The task was to search suitable methods to implement flow monitoring in a service provider environment. Customer Premises Equipment was chosen for monitoring points. The objective was to gain more comprehensive visibility into traffic and explore the benefits of flow monitoring. The implementation was a case study where different open source software was evaluated.

It was noticed that the used equipment did not support TLS encryption in flow export. Another observation was that there exists several software for flow data receiving, processing and saving. The best results could be achieved with a message broker that other software uses as a data bus to transfer data. This arrangement decreases the effort in moving from one software to another. It was also noticed that plain flow monitoring did not bring much extra value compared to SNMP monitoring.

Flow data could bring more value to administrators when used in more sophisticated solutions, such as anomaly detection or security evaluations. In any case automation is needed to gain the best benefit from the data.

Keywords/tags (subjects)
NetFlow, IPFIX, CPE

Miscellaneous

jamk.fi

| Tekijä(t) | Julkaisun laji | Päivämäärä |
|---|---|---|
| Virtanen, Kimmo | Opinnäytetyö, ylempi AMK | Helmikuu 2019 |
| | Sivumäärä<br>65 | Julkaisun kieli<br>Englanti |
| | | Verkkojulkaisulupa myönnetty: |

Työn nimi
**Liikennetiedon seuranta asiakaslaitteella**

Tutkinto-ohjelma
Master´s Degree Programme in Information Technology, Cyber Security

Työn ohjaaja(t)
Sampo Kotikoski

Toimeksiantaja(t)
AinaCom Oy

Tiivistelmä

Tietoliikennevoita voidaan valvoa keräämällä tietoa valvontapisteen läpäisevien datapakettien otsikkokentistä. Perinteisesti tekniikkaa on käytetty laskutusperustan luomiseen, kapasiteettisuunnitteluun ja vianetsintään. Vuoinformaatiota käytetään enenevissä määrin myös ohjelmistopohjaisissa verkoissa, koneoppimisessa ja datavuon poikkeamien etsinnässä.

Työn toimeksiantaja oli AinaCom Oy ja tehtävänä oli tutkia soveltuvia tapoja ja ohjelmistoja toteuttaa tietoliikennevoihin perustuva valvonta toimeksiantajan verkossa. Asiakaspäätelaitteet valittiin valvonnan kohteiksi. Tavoitteena oli selvittää mitä lisäarvoa vuovalvonnalla oli saatavissa SNMP verkonvalvontaan nähden.

Käytännön työ toteutettiin asentamalla testattavat ohjelmistot virtuaalipalvelimelle. Erilaisia soveltuvia ohjelmistoja löytyi runsaasti. Toteutuksen aikana todettiin, että parempaan testaustulokseen päästäisiin käyttämällä erityistä viestiväyläohjelmistoa, johon voisi liittää useampia testattavia ohjelmistoja. Toteutuksen aikana todettiin, että käytetyt päätelaitteet eivät tukeneet kaikkia toivottuja tiedonsiirtotapoja. Tämä ei suuresti vaikuttanut työn lopputulokseen. Tutkimuksen tuloksena todettiin, että vuovalvonta ei tuo suurta lisäarvoa SNMP valvontaan nähden.

Vuodataa voidaan hyödyntää esimerkiksi tietoturvaratkaisuissa poikkeamien etsinnässä datavuosta. Tämä vuodatan prosessointi olisi antoisa jatkotutkimuksen kohde.

Avainsanat (asiasanat)
NetFlow, IPFIX, CPE

Muut tiedot

## Contents

**Figures**

**Acronyms**

| | |
|---|---|
| CPE | Customer Premises Equipment |
| (D)DOS | (Distributed) Denial of Service |
| MPLS | Multiprotocol Label Switching |
| ISP | Internet Service Provider |
| IPFIX | IP Flow Information Export |
| ITIL | Information Technology Infrastructure Library |
| SNMP | Simple Network Management Protocol |
| OSI | Open Systems Interconnection |
| ISO | International Organization of Standardization |
| CCITTT | Comité Consultatif International Téléphonique et Télégraphique (International Consultative Committee on Telephony and Telegraphy) |
| TCP/IP | Transmission Control Protocol / Internet Protocol |
| MIT | Management Information Tree |
| MO | Managed Object |
| CMIP | Common Management Information Protocol |
| QOS | Quality of Service |
| TMN | Telecommunications Management Network |
| FCAPS | Fault-management, Configuration, Accounting, Performance, Security |
| eTOM | Enhanced Telecom Operations Map |
| NGOSS | Next Generation Operations Support systems |

| | |
|---|---|
| ISO/IEC | International Organization of Standardization / International Electrotechnical Commission |
| CIA | Confidentiality, Integrity, Availability |
| ASIC | Application Specific Integrated Circuit |
| RFC | Request for Comments |
| IANA | Internet Assigned Numbers Authority |

# 1 Introduction

Information security demands have increased the need for monitoring IT systems and networks. Considering network management and monitoring there are quite a few alternative methods how companies and organizations monitor e.g. bandwidth utilization, protocols, data flows, contents, and behaviour. This research focuses on how data visibility could be increased using information gathered from customer premises equipment (CPE).

## Assigner

The thesis was assigned by author's employer AinaCom Oy. AinaCom provides IT and communications technology solutions for business customers and it is the largest virtual operator in Finland. Solutions consist of talk, mobile, network and IT solutions including connections and equipment. AinaCom Oy has three offices located in Helsinki, Hämeenlinna and Turku. (AinaCom 2018).

AinaCom is a part of Aina Group concern, the main subsidiaries of which are the newspaper company Hämeen Sanomat Oy and AinaCom Oy. AinaCom has over 60 employees. (Aina Group Quarterly report 1-6 2017).

Telia acquired AinaCom in December 2018. After acquisition AinaCom continues its operation as Telias subsidiary. (Härjämäki 2018 a). After the acquisition company name was changed to Telia Communication. (Härjämäki 2018 b)

## Goal of Thesis

This work intends to evaluate methods of gathering information from CPEs. It is considered how different methods are suitable for different purposes. Fault finding, traffic baselining or automatic provisioning could all utilize different methods and

protocols. The research aims to find out what flow-based information could be gathered and for what purposes. The utilization of Network flow information for fault finding purposes was thought as a good topic when first planning the subject for the thesis.

Flow monitoring increases knowledge about network traffic by examining packet headers. Header information is combined into flows and handed over for further analysis. From the security perspective, flow information can be used to detect (D)DOS attacks and network intrusions or attempts of intrusions. Additionally, it is possible to detect anomalies from the flow data.

## Research area

The assignor company provides ISP services to its customers. A relevant portion of overall service sets are Internet connections. The connections can be modified for their physical and logical features to fit each use case. However, the majority of connections are implemented with a CPE router and routed connection. In the basic case, the customer is allocated a public IP subnet and it is routed straight to the Internet or through the firewall. A common implementation is illustrated in Figure 1.

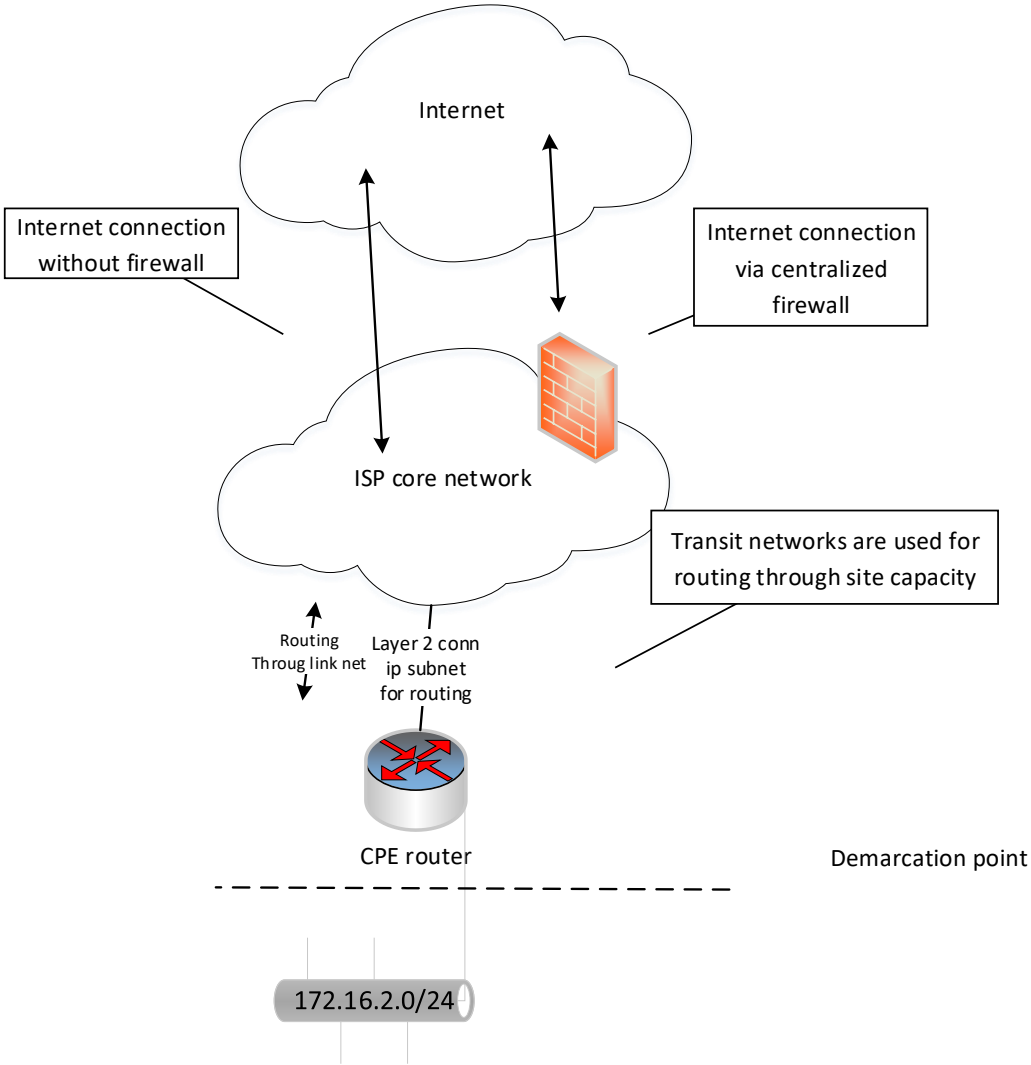


Figure 1. Example of layer 3 MPLS corporate network.

MPLS service provider has several points where it can monitor data going through its network. Figure 1 presents CPE equipment that is usually a router, core equipment and a firewall located at the Internet edge. These basic components are also used in MPLS based corporate networks. If a customer does not want a centralized firewall from the ISP, the customer allocated network is routed straight into the Internet

without trespassing the ISP firewall. Demarcation point illustrates the point where service provider's responsibility ends at the customer site.

## Research problem

Monitoring traffic destined from or to the customer site could be implemented in CPE or in ISP core. This thesis examines CPE as a monitoring point. The research question is to evaluate what additional benefits data flow monitoring in CPEs can provide and how data flow monitoring could be implemented in the assignor´s environment.

## Objective and scope

The research is intentionally limited to netflow monitoring in the CPE equipment; what added value Netflow produces into existing monitoring and how it should be implemented into the assignors´ environment.

The research compares Flow transport methods Netflow and IPFIX. During the research the possibilities of flow information saving methods could be concidered. No additional hardware should be needed and only open source software is used.

Examination of legislation is crucial before implementing monitoring in the live environment. The EU and Finnish legislation set constraints on e.g. how traffic information can be gathered, stored, and handled. Legislation clarification is needed for the knowledge of how monitored data can be used for different purposes. The examination of laws is made apart from the technical study, because legislative restrictions could narrow down the research scope unnecessarily.

The network flow monitoring should be investigated from the network monitoring perspective. There are different standards or guidelines how to implement network

monitoring as a part of business operations. Flow monitoring is compared to the processes of the ITIL framework.

Research methods

The research method is a case study with a qualitative approach. The research is executed by studying flow and Netflow protocols as well as different transport methods. Additionally, different open source software is reviewed for their compatibility for the implementation. After suitable methods and software have been found, they will be implemented and reviewed in a demo environment. The research is made from service provider perspective, and it is meant to serve as a background study for a possible additional security control in customer implementations. After studying the theoretical base, a few flow implementations are are built and tested in the demo environment.

# 2 Theoretical base

The following subchapters introduce acknowledged standards and practises for network management.

## 2.1 Network management

Network monitoring is a part of network management. Network management consists of controlling, planning, allocating, deploying, coordinating and monitoring network resources. (Farrell 2009 Chapter 4.1)

Network monitoring can be divided into passive and active monitoring. Passive monitoring collects statistics from network traffic produced by users. Statistics are usually collected by management software with SNMP protocol. Active monitoring is accomplished by sending data patterns into network and measuring values how the data was transmitted through the network. (Farrell 2009 Chapter 5.1 )

When searching for the roots of network managing and monitoring frameworks, one has to start from the beginning of the modern Internet. Early network management documents have evolved differently for OSI model and TCP/IP model. There was a race between these two networking models in the early days of Internet.

OSI reference model to standardize telecommunications was published by International Organization for Standardization (ISO) in 1984 followed by management extensions. At the time of the preparation, OSI was collaborating with CCITT. CCITT had many participants from traditional circuit switched telecommunications companies. Packet, or datagram switched networks was being developed, so ISO had to include both, datagram switched and circuit switched techniques in its standard. OSI model was criticized for this decision, which made it more complex than its rival TCP/IP solution. In addition ISO was criticized for slow and bureaucratic procedures. (Russell 2013)

OSI management model uses Common Management Information Protocol (CMIP) as a protocol to communicate the managed information. Managed equipment, e.g. network switches, maintain Management Information Tree (MIT) database. MIT includes managed objects (MO). Managed equipment maintains several MOs that can be viewed or changed with CMIP protocol. (Yemini 1993)

For example, the following publications describe OSI management; OSI Management Framework ISO/IEC 7498-4, Systems Management Overview ISO/IEC 10040 and Management Information Model, ISO/IEC 10165-1. Protocol is described in Common Management Information Service (CMIS) Definition ISO/IEC 9595, and the Common Management Information Protocol (CMIP) ISO/IEC 9596-1. (The Open Group 1997).

CCITT was the predecessor of International Telecommunications Union (ITU-T). It provided guidelines for national telecommunication companies to manage an interoperate their networks. ITU-T replaced CCITT in 1993. ITU-T continued CCITT's work in developing a methodology for network management. ITU-T pioneered network management and created Telecommunications Management Network (TMN). It was published in May 1996 in ITU-T recommendation M.3010. (Adeel, Madani, Siddiqui 2011 Chapter 1) TMN has included similarities with OSI management concepts. (Pras, Beijnum & Sprenkels 1999, 19-20)

TMN (Telecommunications Management Network) is defined in publications which are given identification numbers between M.3000-M.3599. The papers include principles, architecture, definitions and specifications to implement any kind of TMN. TMN´s purpose is summarized in M.3010 as follows. "The basic concept behind a TMN is to provide an organized architecture to achieve the interconnection between various types of Operations Systems (OSs) and/or telecommunications equipment for the exchange of management information using an agreed architecture with standardized interfaces including protocols and messages." (ITU-T M.3010) TMN is a separate management network with several connections to telecommunications network to send and receive traffic from the managed systems or equipment. Later work with TMN has been done within Telemanagement Forum (TMF) (ibid. 2,20)

IETF published Simple Network Management Protocol (SNMP) in 1988. It was the management solution for TCP/IP. Since TCP/IP became the most adopted technology in networks, also SNMP became the most used management framework in packet switchet networks. It is still widely in use today. (Sathyan 2010, Chapter 6.2)

Both management models, OSI management model and SNMP framework utilize ASN.1 (Abstract Syntax Notation One) to define the managed objects. OSI model uses complete ASN.1, specification and SNMP framework uses only a subset of the specification, which has made OSI model management more versatile than its counterpart. (Park, Choi, Jung & Sunwoo 1993, 147).

SNMP is simpler, cost-effective and open in standards. That is why it is the most popular network management protocol. TMN is model is more complex and concentrates on reliablity and stability of networks. It was initially proposed for managament of telecommunication networks. (Jianguo 2010)

## 2.2 The structure of TMN model

ITU-T´s TMN introduces logical layers. A hierarchical model reduces complexity by splitting the management functionalities into layers: business management, service management, network management, element management and network element layers. The layers support different kinds of management sets called management information models. These models enable the management of different kinds of systems. The actions between layers are possible by defined means. (Architectural and Framework Standards: The TMN/FCAPS Model (ITU-T) 2008 - 2018)

*Network element layer* is the most physical layer. It represents elements that are being managed. *Element management layer* deals with the management of individual network elements. It should provide consistent and full access management view into multivendor network. Element management layer can also group the managed equipment. Statistics and logs are collected within the scope of the layer. *Network management layer* takes care of network management and it is supported by the element management layer, providing technology transparent view into wide geographical networks. Network management layer controls all elements in its scope and provisions networks to support customer services. Network wide statistics and performance information is collected. *Service management layer* manages customer faced services. Service creation, order handling, implementation, monitoring, complaint handling and invoicing are included. Also, interactions with other networks, e.g. operator connections are managed in this layer. *Business management layer* is responsible for the whole enterprise. It has access to other layers; however, other layers should not have access to business management layer functions. It provides information for business management such as planning

investments, manpower or budget. (Architectural and Framework Standards: The TMN/FCAPS Model (ITU-T) 2008 - 2018)

ITU-T published their recommendation M.3400 in 1997. It is one publication of the series that specifies TMN. It brought the ITU-T concept FCAPS into TMN. FCAPS had been introduced earlier in the ITU-T OSI model. FCAPS categorizes management into five functional areas. The first letters of the functional areas form the abbreviation FCAPS. All logical layers should be treated with their own FCAPS domain. (Sathyan 2010, Chapter 2.6) Functional areas are introduced in the following: (Sathyan 2010, Chapter 3)

*Fault management* is a functional area for error logs, error notifications and diagnostic tests. In addition to logging and taking notice of abnormal action it is also the area which takes corrective actions to fix faults.

*Configuration management* associates names with managed objects and sets the parameters that makes routine operation of the system. It obtains announcements of changes in the condition of network elements and changes configuration of the open system.

*Accounting management* functional area enables charges to be established as resources are being used. It makes possible to identify the origin of the costs and set limits for the usage of resources due to the accounting costs. This area also informs users from costs and combines costs where several resources together form the actual cost. *Performance management* collects statistics, maintains logs for system state, determines system performance and makes changes to systems in the name of performance management.

*Security management* controls security mechanisms and the distribution of security related information. In addition, security relevant reporting is accomplished in this functional area.

FCAPS management structure where logical layers are treated with its own FCAPS domain is shown in Figure 2. (Introduction to eTOM 2009)

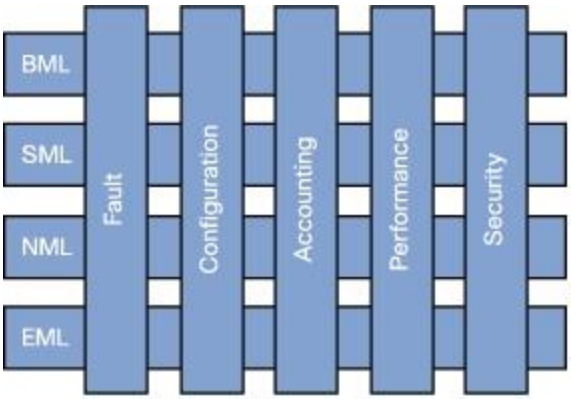


Figure 2. TMN FCAPS structure

## 2.3 Network management forum

eTOM was developed by TeleManagement Forum (TM Forum). It is global non-profit industry association that aims to help communications service providers and their suppliers to digitally transform and thrive in the digital era. It has over 850 members from across 180 countries. (About us 2018)

Cisco describes TM Forum function as follows “The TM Forum is an industry association focused on transforming business processes, operations, and systems for managing and monetizing online information, communications, and entertainment services.” (Introduction to eTOM 2009)

TOM was created between 1995 and 1999 and it evolved into eTOM between 2000 and 2002. eTOM was also published by ITU-T in recommendation M.3050. (Introduction to eTOM 2009) ITU-T papers from M.3000 to M.3599 include Telecommunications management network recommendations. (ITU-T recommendations 2018)

TM Forum was founded as Network management forum. The name was changed in 1998. Computerworld wrote in 1989 that: “The OSI/NMF, whose membership currently numbers more than 60 network and computer vendors, was formed last July to select and recommend certain subsets of OSI protocols to ensure consistent implementations of the standard among its members´ products.” (Horwitt 1989)

TMN and eTOM are different from the view of point they are made. TMN guidelines network management from bottom to up. On controversary eTOM guidelines supporting processes of entire enterprise, from top to down. (Introduction to eTOM 2009)

eTOM and its associated instruments aim to guide the shape of business processes, agree on information that needs to be transmitted between business activities, identify environment to interconnection of operational support systems and enable development of products for integrating and automating telecom operator processes. eTOM is part of NGOSS framework (Next Generation Operations Support Systems). (van Bon. Verheijen, Chapter 17.1) NGOSS is comprehensive framework for developing, procuring and deploying OSSs, BSSs (Business Support Systems) and software. It includes in addition to eTOM business process map, Shared Information/Data (SID) model for common language for all data, Technology Neutral Architecture guidelines, Compliance and Conformance Criteria guidelines and Lifecycle and Methodology processes for solutions development. (Introduction to eTOM 2009)

The highest conceptual layer of eTOM shows strategic, infrastructure and product (SIP) as well as operational processes in two different areas. The key functional areas are situated horizontally over strategic and operative process areas. The enterprise processes are their own area. Internal and external influencers are marked with ovals. Figure 3 shows this level 0 conceptual view of the model. (ITU-T M.3050.1 2007)

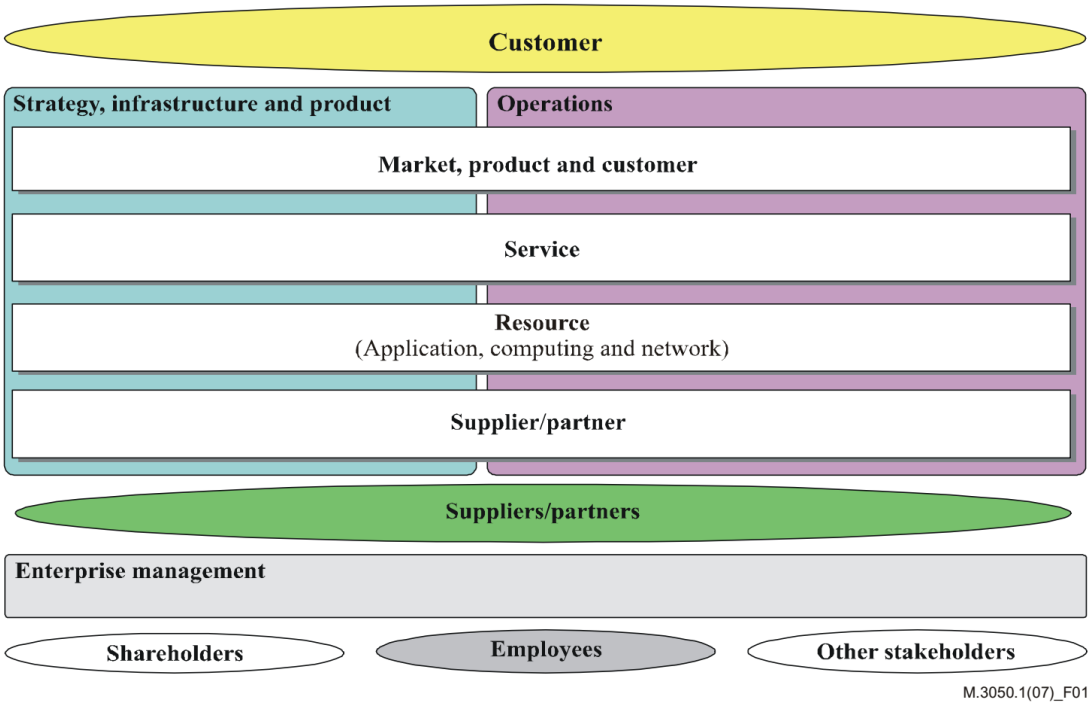


Figure 3. eTOM conceptual layer

Figure 4 shows again the three process areas. Level 0 process areas are now divided into level 1 process groupings. Seven vertical process groupings are end-to-end processes that are needed to support customers and manage the business. Functional process groupings are drawn horizontally. This level 1 figure gives an overall view of the framework. When forwarding into more precise divisioning, i.e. layer 2, functional process groupings are divided into more accurate tasks. (ITU-T M.3050.1 2007)

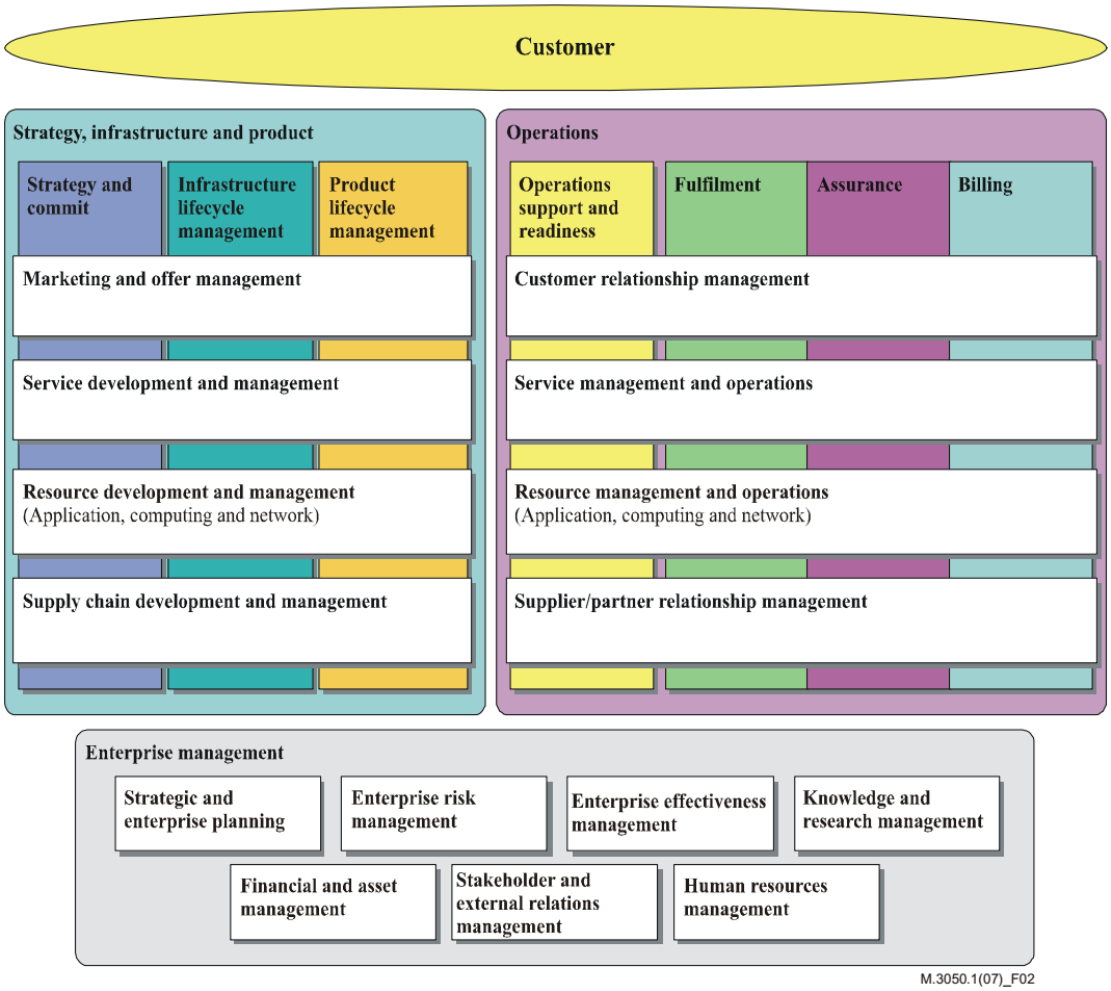


Figure 4. eTOM level 1 illustration

eTOM and ITIL framework have some overlapping. TM Forum describes eTOM and ITIL relationship as follows: “ITIL encompasses a set of ‘good practices’ that are widely recognized and applied and shows how these can be orchestrated in a service management lifecycle. TM Forum’s Business Process (eTOM) and Information (SID) Frameworks deliver a reusable, agreed, and widely adopted service-oriented architecture, with processes that can be linked directly with ITIL’s good practices” (Relationship to ITIL)

## 2.4 ISO 27000 series of information security standards

Not least for security comparability purposes, an ISO standard for security management has been introduced. ISO/IEC 27000 family provides standard for Information Security Management System ISMS. IEC stands for International Electrotechnical Commission. The standard includes features that an expert committee has discussed to be International state of the art in Information Security Management Systems. It is intended to assist organizations to implement and run an ISMSM. (SFS-EN ISO/IEC 27000:2017) The roots of the standard are in Britain. Before ISO, the concept was standardized by British Standard Institution. (An Introduction to ISO 27001, ISO 27002....ISO 27008 2018)

Standard 27000 that bundles all 27k standards together defines; “An ISMS is a systematic approach for establishing, implementing, operating, monitoring, reviewing, maintaining and improving an organization´s information security to achieve business objectives. It is based upon a risk assessment and the organization´s risk acceptance levels designed to effectively treat and manage risks.“ (SFS-EN ISO/IEC 27000:2017). In other words, the assets are identified and risks are evaluated. After evaluation, controls are implemented according to assets and risks. Controls are maintained actively and assets are periodically re-evaluated. All progress is done according to standard guidelines.

Currently ISMS standard consists from 19 different 27XXX numbered standards. The standard family is introduced and overviewed in number 27000. The rest of the standards can be divided into three types: requirement specification, description of general guidelines and sector-specific guidelines. (ISO/IEC 27000:2017)

Standards 27001 and 27006 are two standards that specify the requirements for fulfilling the standard demands. Standard number 27001 is intended for the implementor and standard number 27006 is meant for the accreditor party that accredits the implementations of the standard. (ISO/IEC 27001:2017)

## 2.5 Information Technology Infrastructure Library (ITIL)

Information Technology Infrastructure Library (ITIL) was originally developed in the 1980s. British government Central Computer and Telecommunications agency was responsible for making the guidelines for efficient and financially responsible IT. It was meant to serve the government and the private sector. (In a Nutshell: A Short History of ITIL 2005) Nowadays joint venture company AXELOS, created in 2013, manages and develops the portfolio. ITIL is actively maintained and the most acknowledged IT service management approach in the world (About AXELOS). ITIL has reached its third evolution step and the fourth version is under development. (ITIL-update)

ITIL is a collection of IT service management best practices that have been written from the service point of view. ITIL consists of five publications that follow service lifetime stages. (What is ITIL Best Practice)

- ITIL service strategy
- ITIL service design
- ITIL service transition
- ITIL service operation
- ITIL continual service improvement.

Service lifecycle is illustrated in Figure 5. (Stages of ITIL Service Lifecycle 2016) Every stage of a lifecycle must be handled as an embedded part of the service lifecycle. Each stage affects the others and is interdependent. To introduce connections between service stages few examples will follow.

Service strategy defines e.g. service portfolio, demand management, service budgets. Operations provide risks and cost information backwards to strategy. Design confines the operations through e.g. service catalogue, SLAs, security policies, procedures. Operations give the design valuable monitoring and reporting information.

A service has to be launched and modified into its environment during its lifetime. Service transition takes care that the service launches, other changes are

implemented fluently and transition outcomes are monitored. It also takes care of service validation, testing, and quality assurance. Service transition ensures for its own part that the service fits its purpose. The final part of service lifecycle is continual service improvement. This stage utilizes input from any service lifecycle stage. Among other information, service operations provide invaluable information from service performance. (Morris & Gallacher 2016, Chapter 32)

Network management fits into service operation stage and therefore it will be introduced more closely.

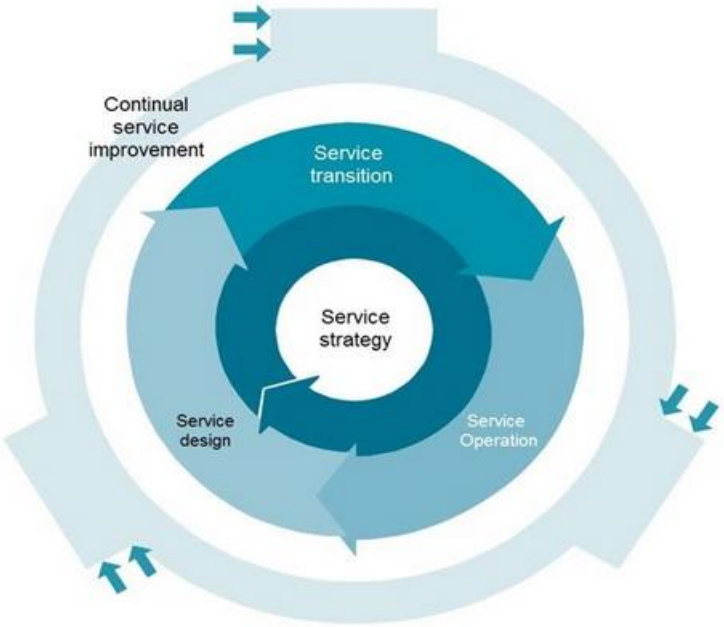


Figure 5. ITIL service lifecycle

Service operation stage of service lifecycle is the most visible stage for service users and for the business. It takes care of the daily tasks concerning the service. Service operation takes place after the service has been designed and transitioned into

production. The processes in service operation's lifecycle stage aim to deliver and support the service at the agreed service level. In practice, this stage handles all queries, interruptions, malfunctions that could occur in running IT service. In addition to supporting service, service operation is responsible for trying to reduce incidents or interruptions and monitoring and baselining are tools for this. Also, the access management is on service operations responsibility. (Morris & Gallacher 2016, Chapter 32)

ITIL is a collection of processes. Service operation includes several processes. (Morris & Gallacher 2016, Chapter 32)

- Event management
- Incident management
- Problem management
- Request fulfilment
- Access management

ITIL defines a team of people focusing on a task as a function. Service operation lifecycle introduces these functions. (Morris & Gallacher 2016, Chapter 32)

- Service desk
- Technical management
- Operations management
- Application management

Before exploring these areas more carefully, it is essential to clarify event, incident and problem. The most minor, however, not unnecessary at all is event. Events indicate any change of state in the managed object. Events do not necessarily trigger any actions. Often, they are indicators from a normal operation. Reporting and fault finding utilize events. Miessler (2015) defines events as follows: "An event is an observed change to the normal behavior of a system, environment, process, workflow or person." Successful or unsuccessful authentication logging is a typical event. Events that require actions and are flagged to support staff are called alerts. An example of alert could be data traffic crossing capacity threshold in a corporate Internet link.

Incident is an unplanned disruption in the managed service. In addition to quite obvious service outage, also service degradation or redundancy loss are considered as an incident. Branch office leased line breakdown, server raid arrays single hard drive failure or Internet capacity congestion are all incidents. Miessler's (2015) interpretation of incident follows: "An incident is an event that negatively effects the confidentiality, integrity, and/or availability (CIA) at an organization in a way that impacts the business". There is an obvious difference in incident and problem in the ITIL terminology. It defines a problem as an underlying cause of one or more incidents. (Morris & Gallacher 2016, Chapter 34)

### 2.5.1 Event management

Event management is one of the main operations in IT. It monitors changes in service states in managed services. It baselines the normal activities and may rise alerts for changes that cause more actions. Event data is the basis for service reporting, health monitoring and performance control. Event management baselining is a valuable resource when resolving incidents or problems. In addition to passive event listening, event management might carry out active monitoring and even automatic updating. Active monitoring is for example pinging or HTTP requests sourced from the monitoring system. Event management provides signals e.g. for incident, problem and change management. (Morris & Gallacher 2016, Chapter 36)

The process of event management describes handling of events throughout their lifecycle. It detects state changes and should describe appropriate action for every event. Control action could be automated e.g. increased logging after failure notification. At the same time, failure event could have started an incident process. Event management requires automation. No NOC (Network Operating Centre) can handle all events by hand. Automation is tireless and less error prone when detecting issues. (ibid.)

Classifying can help to filter unnecessary events to overwhelm an event management system. Events are classified into three categories. Informational event does not

require any action, even though it might trigger some predetermined action. Often it is an indication of normal activity. Warning event should be evaluated if actions are needed. It could be an indication that capacity threshold of an internet link has exceeded. The third event type is exception. An example of an exception could be a user attempting to log in with an incorrect password. Three event types are not strictly defined. It is an organization's choice how to divide events into these types. (Morris & Gallacher 2016, Chapter 36)

Event management process flow starts when an event occurs. It has to be noted somehow. Some events are reported automatically straight from service component and some are results of active polling. Process flow names the informing of event management engine as an event notification. When the event management system receives the event, event detection and logging take place. The next phase is correlation and filtering. Correlation evaluates the importance of the event. After the evaluation has been carried out, the event is classified. Based on classification, the common actions are as follows. Informational events are closed, warning events are passed forward to further evaluation, and exceptions rise incident or some other service management process. (Morris & Gallacher 2016, Chapter 36)

### 2.5.2 Incident management

Incident as defined in ITIL is an unplanned outage, reduction of quality or any failure in service that has not yet affected user experience. When service is returned to normal state towards users, incident is solved. Incident resolving does not include investigating the reason why incident was raised. Only thing that matters is that service is restored to agreed service level. Most incidents are raised by user announcements or event manager. If the cause of failure that raised incident needs to be investigated, problem management process is initiated. (Morris & Gallacher 2016, Chapter 34)

Incident management guides organizations to use standardized methods walking through incidents. The whole process benefits from common practices when

organization resolves incidents. Standardized methods return improved resolve times, detailed visibility and consistent reporting among others benefits. Incident prioritization should be performed based on business needs. “All incidents must be efficiently responded to, analyzed, logged, managed, resolved, and reported.” (Morris & Gallacher 2016, Chapter 34) The process itself aims to keep the customer satisfied even though something unwanted has happened. Eventually, the incident management process, as IT usually, is in the service of business goals. (Morris & Gallacher 2016, Chapter 34)

Because incidents are an outage or a degradation of service, it is essential that incidents are handled without unnecessary delays, which is why incident resolving times have to be monitored. Service level agreement is the main guideline that has to be fulfilled. It usually has at least service uptime defined. The time that the service desk handles the ticket is measured as operational level agreement. Often IT systems have subcontractors or other companions. If incident resolving requires third party intervention, their response and action time is agreed in underpinning contracts. (ibid.)

ITIL defines two different escalation methods. Functional escalation is needed when the first line realizes that an issue cannot be resolved with their knowledge or tools. In addition, a timer counting incident resolving time can trigger functional escalation. There can be several support levels. No matter what level is investigating an incident, the ownership of the incident is always in the service desk. The service desk keeps relevant parties informed about the incident. Hierarchic escalation is conducted in case of a major incident. In hierarchic escalation, appropriate management is informed about the incident. Management can then make the necessary decisions about e.g. prioritization and resources. (ibid.)

### 2.5.3 Problem management

Problem management researches the reasons for incidents. If the reason for an incident is known, problem management is not needed since there is no problem.

Problem management tries to find final fix for recurring or single incidents. If final resolution is not achieved fast enough, also workarounds can be used. In addition to root cause investigation for incidents, the problem management tries to prevent incidents or problems occurring in the first place. When incidents or problems cannot be prevented, they should be mitigated so that in case of occurrence, the hit for the service would be as minimal as possible. (Morris & Gallacher 2016, Chapter 34)

Problem management updates the database about problem root causes. Other processes such as incident management can use known problems database to solve incidents more efficiently. If problem is not known and it is not in known problems, the database incident management needs to trigger problem management process to resolve the issue. When a problem is solved, it is added into a database and the incident management does not anymore trigger problem management for the same kind of incident. Incident management could trigger problem management process also if it recognizes a rising trend in certain kind of incidents. (ibid.)

### 2.5.4 Request fulfilment

Requests indicating a fault or a degradation in service are incidents. There are also many other requests such as password reset requests, information requests or equipment relocation requests. These requests fall into request fulfilment process. The process handles repeatedly occurring questions where the service provider can provide customer what they want. ITIL defines that the request fulfilment process handles frequently occurring, low cost and low risk requests. (Morris & Gallacher 2016, Chapter 35)

In contrary to incidents request fulfilment has often a lead time. Incidents are always resolved as soon as possible. Like the incident process, also request fulfilment includes prioritization and escalation possibilities and users are informed about progress. Request fulfilment should use predefined patterns or practices. Dealing with a request is more efficient when the same kind of requests are always handled

in the same way. Standardized request fulfilment is opportune for automation. (Morris & Gallacher 2016, Chapter 35)

### 2.5.5 Access management

Access management takes care that authorized users can use the service and unauthorized users cannot. In addition to adding, removing or revoking authorizations, access management monitors that rights are used according to company policies. Authentication tracking is needed, to trigger a security incident if needed. All authorization related tasks should be taken care in the same organization to keep up consistency. (Morris & Gallacher 2016, Chapter 37)

## 2.6 Flow protocols

Flow monitoring can be conceptually divided into several phases. First phase in the prosess is Packet Observation. Packets are captured in an Observation Point and are preprosessed. The second phase is Flow Metering & Export. In Metering Process flows packets are aggregated into flow information. Exporting Process delivers created flow information to configured destination. The third phase is Data Collection. It receives, storages and preprocesses the flow data. Collecting phase processing could include e.g aggregation, filtering, data compression and summary generation. The fourth and final phase is Data Analysis. Analysis could be automated on manual. Analysis could include e.g traffic profiling, classification and anomaly and intrusion detection. (Hofstede, Čeleda, Trammell, Drago, Sadre, Sperotto & Pras 2014, 4)

The term NetFlow is used widely when describing flow monitoring. It refers to all Cisco-proprietary flow export protocol versions e.g NetFlow version 9, as well as to the Cisco´s flow capture and metering technology. Flow data is sent from exporter to collector with flow export protocol. (ibid., 3) In the following text, NetFlow is first

discussed as a technology and later NetFlow version 9 and IPFIX export protocols are introduced in their own subchapters.

The Cisco NetFlow technology can be used as an example when introducing flow monitoring, since it uses well known flow protocols and other protocols tend to be interoperable with it. Netflow was developed in 1996 by Cisco Systems. (NetFlow gives network managers a detailed view of application flows on the network 2003, 2) It has been used for accounting of data traffic, billing, capacity planning and bandwidth monitoring among others monitoring tasks. Additionally, security monitoring has utilized NetFlow. (Santos 2016, 1-2)

Santos describes the general principle of a Flow (Santos 2016, 4)

> *A flow is a unidirectional series of packets between a given source and destination. In a flow, the same source and destination IP addresses, source and destination ports, and IP protocol are shared. This is often referred to as the five-tuple.*

The five-tuple refers to the five parameters used to classify traffic into flows. It is the minimum amount of information that NetFlow uses. The five-tuple refers to source address, destination address, source port, destination port and protocol fields. Depending on its version, NetFlow can gather more data to classify traffic into more granular flows. In the beginning, the information was collected only from IP packet headers; however more recent versions can also include other parameters. For example, layer 2 header, IPv4 header, IPv6 header or Ethernet port information could be used. (Santos 2016, 4-5) Classifying traffic based on different traffic parameters is not unique for NetFlow. For example, OpenFlow specifes as a default 44 different match fields to classify network traffic. (OpenFlow Switch Specification Version 1.5.1 2015, 79)

Packets classified into the same flow update the flow database held in the NetFlow utilizing device memory. NetFlow information records are called flow records and the

storage is called flow cache. Database could be viewed on the observing network equipment, however the best usability is achieved when flow information is sent to the flow collector. (Santos 2016, 5-6)

Cisco systems proprietary NetFlow has evolved through several versions. The most recent versions of NetFlow are NetFlow v9 and Flexible Netflow. Flexible NetFlow did not introduce new flow export protocol version. (Santos 2016, 40) NetFlow is supported by many vendors; however, also other flow monitoring methods exist. For example, Juniper Networks have their own proprietary Jflow. Juniper Networks´s proprietary Jflow is fully interoperable with NetFlow capable collectors (Juniper Flow Monitoring 2011, 3). Similarly Huawei supports Netstream protocol that serves quite the same way than as NetFlow. (Netstream 2015)

In addition to NetFlow and Jflow, there is sFlow. sFlow does not observe flows; it is rather a packet sampling technology. It was first developed by HP and Inmon. Nowadays Inmon is responsible for taking care of sFlow brand, development and licensing. Many networking vendors, such as Dlink, Extreme networks and others support this technology. Licensing for free may have affected its favour among networking vendors. (Trost 2009)

Unlike NetFlow, sFlow is a hardware-based technology. The largest part of it is programmed into equipment ASICs. This method requires less computing power than examining packets at operating system level. sFlow is a packet sampling technology and it examines packet at configured interval. From sampled packet it collects full headers and TCP flags and up to the first 80 bytes of the payload. (Trost 2009)

### 2.6.1 NetFlow version 9

The most popular NetFlow version is 9. (Muniz & Santos 2017) NetFlow version 9 data export format is defined in informational RFC3954. NetFlow examines packets traversing a monitoring point, usually a router- or a switchport. The actual point where data is monitored is called an observation point. Several observation points

may belong to the same observation domain. For example, router line card could have each of its interfaces as an observation point, and the line card with all its observation points could form an observation domain. Every observation point is associated to an observation domain. (RFC3954)

Packets with common predefined parameters are classified to belong to a certain flow. The equipment responsible for monitoring, creating flow-cache and sending flow records to central database is called the exporter. Equipment gathering flow info from exporters is called a collector. The flow between source and destination can be timed out via a predefined timeout or via TCP protocol FIN or RST bit. Common configuration is that after the observed flow has been timed out, it is stored as a flow record into flow-cache and then sent to collector. (Santos 2016, 43-44; RFC3954)

NetFlow version 9 data export format is designed to be independent from the underlying transport protocol. The RFC, however, states that UDP is used for exporter processing efficiency. Exporter is capable to export to multiple collectors using independent transport protocols. NetFlow v9 export protocol is not designed to transport flow records over public networks. There is no security built in the protocol. (RFC3954)

The main difference between version 9 transport protocol and its predecessors is that version 9 is based on templates. Earlier NetFlow versions have hardcoded the information that NetFlow fields carry. The templates in version 9 enable customization of exported fields. New features can be implemented while simultaneously supporting legacy implementations. For example, syslog messages can be sent over NetFlow9 (*Cisco ASA Series General Operations CLI Configuration Guide, 9.2* Chapter: NetFlow Secure Event Logging).

NetFlow v9 record format comprises template FlowSets and data FlowSets. Template FlowSets describe and define fields transferred in data FlowSets. Netflow v9 packet header format is illustrated in Figure 6 (RFC3954).

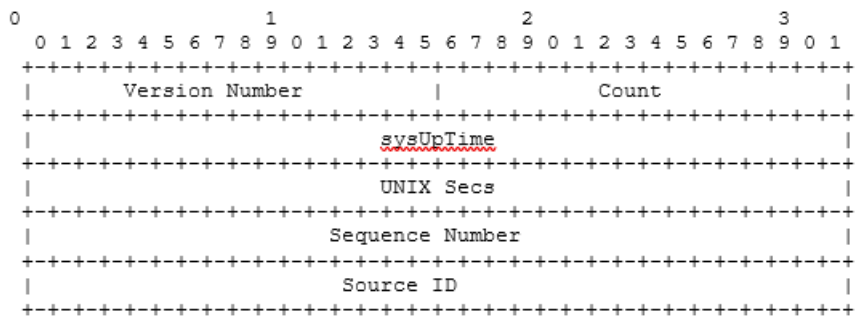


Figure 6. Netflow v9 packet header format

Packet header includes the version, which is 9 for version 9. Count field expresses the sum of any kind of records in a packet. Field sysUpTime shows milliseconds from last equipment restart. UNIX secs shows seconds from 0000 UTC 1970 when the packet leaves exporter. Sequence number is counter of all export packets from the current observation domain to the current exporter. Lastly, the source id field identifies the exporter observation domain. Observation domains can be used to differentiate several export streams from same exporter. (RFC3954)

Export packet includes one or more FlowSets. FlowSet is a term for a collection of flow records that have similar structure. Three types of FlowSets exist. Template FlowSets describe what NetFlow information data FlowSets include. Data FlowSets include the actual NetFlow data. Collector knows how to handle data FlowSets because it has received template FlowSets from the exporter. The third FlowSet type is Options Template FlowSet, which makes it possible to transfer NetFlow process information, such as sampling interval or configured timers. Options information is sent regularly in data FlowSets to the collector. (RFC3954)

NetFlow version 9 records contain fields. The field types from the RFC 3954 are listed in Appendix 1.

### 2.6.2 Flexible NetFlow

Flexible NetFlow is Cisco´s next-generation flow technology. Flexible NetFlow can be configured to use export method NetFlow v5, NetFlow v9 or IPFIX. New export protocol format or version is not introduced. Flexible NetFlow has the possibility to maintain several flow cache databases in exporter. Different databases can use different collectors. In addition to several flow caches, three different types of flow caches are available there. They are called normal, permanent and immediate cache. Normal cache is a traditional cache used in the earlier versions of NetFlow. Permanent cache keeps increasing flow counters and the information not zeroized. Immediate cache will age out every flow as soon as it is created. Immediate age out will produce one packet size flow records. (Cisco IOS Flexible NetFlow Technology Q&A 2006)

Flexible NetFlow separates data parameters that form a flow into key-fields and non-key fields. Key fields are those parameters that must match in different packets in the same flow. Key parameters are user definable. Non-key parameters are to be collected into flow information but non-key parameters do not have to match in different packets within a flow. (Santos 2016, 61-63)

### 2.6.3 IPFIX

IPFIX (IP Flow Information Export) is IETF standard protocol for exporting flow information from exporters to collectors. It is based on NetFlow version 9. Its protocol version numbering continues Netflow versions and its packet header field marks number 10. Like Netflow, it is a push protocol and does not need any requests from data receiver before transmitting. (Claise & Trammell 2013)

IPFIX was developed for flow exporting purposes, however, it has been developing towards general information push protocol (ibid.). IETF Working Group for IPFIX was concluded in 2015. It published several drafts and there is still more than 20 active RFCs. RFCs 7011 and 7012 are the core elements of the protocol. RFC 7011 defines

the protocol and RFC 7012 describes the information elements that IPFIX transports (IETF 2012).

IPFIX standard defines supported underneath transport protocols. SCTP support is mandatory for compliant implementations. More accurately it further defines the Partially Reliable SCTP (PR-SCTP) extension. TCP and UDP may be supported on a compliant implementation, but only SCTP is mandatory in the standard. By default port, 4739 is listened on the collector end. For secure connections, port 4740 is used. (RFC7011)

RFC 7011 defines the terminology and concepts for the protocol. Observation domain is the largest aggregate into which flow information can be grouped by a metering process. The metering process is the process that observes data traffic or flows and its characteristics at observation points. IPFIX does not define metering process. Every observation domain has it unique ID per IPFIX exporting process, and it is recommended that those IDs should be unique per IPFIX device. (ibid.)

RFC 7011 defines packet flow with the following parameters. The definition applies to any length flow and for sampled flows. Parameters combining all packets in a flow are called flow keys. (ibid.)

> *A Flow is defined as a set of packets or frames passing an Observation Point in the network during a certain time interval. All packets belonging to a particular Flow have a set of common properties. Each property is defined as the result of applying a function to the values of:*
>
> *1. one or more packet header fields (e.g., destination IP address), transport header fields (e.g., destination port number), or application header fields (e.g., RTP header fields [RFC3550]).*
>
> *2. one or more characteristics of the packet itself (e.g., number of MPLS labels, etc.).*
>
> *3. one or more of the fields derived from Packet Treatment (e.g., next-hop IP address, the output interface, etc.).*

Flow records include information about a specific flow. Flow records are generated by a metering process that gets flow information from observation domains and observation points. The observed packet headers and characteristics are inputs to the metering process. The metering process includes e.g. packet header capturing, timestamping, sampling and creating, classifying and maintaining flow records. A flow record contains the measured properties of the flow and usually characteristic properties of the flow. The measured property could be the sum of the byte count of all packets in the flow. An example of a characteristic property is the source IP address. Metering process usually sends flow records to exporting process that sends flow records to collecting processes. (RFC 7011)

An equipment that hosts one or more exporting processes is called an exporter. Collecting process runs on a collector and receives IPFIX messages from one or more exporters. The equipment that has one or more exporting processes running is called an IPFIX device. Observation domains or metering processes are not an attribute for an IPFIX device. IPFIX is protocol that delivers information from exporter process to collecting process. (ibid.)

IPFIX is protocol is based on templates. Template is an ordered sequence of type and length information. It is used to specify the structure of specific data. (ibid.)

IPFIX packets

IPFIX packet header is constructed from five fields. The Fields are illustrated in Figure 7 (RFC7011). The message header fields include the version number field that is 10 in IPFIX messages. The length field shows message length including header. Export time field indicates timestamp when message header leaves from exporter. Sequence number field shows incremental number of all data records sent from certain observation domain by exporting process. Only data records increment this value.

Observation domain ID field includes observation domain id that the information is collected from. Id is unique per exporting process. Collecting process should use the transport session and the observation domain id value to separate streams originating from the same exporting process. (RFC7011).

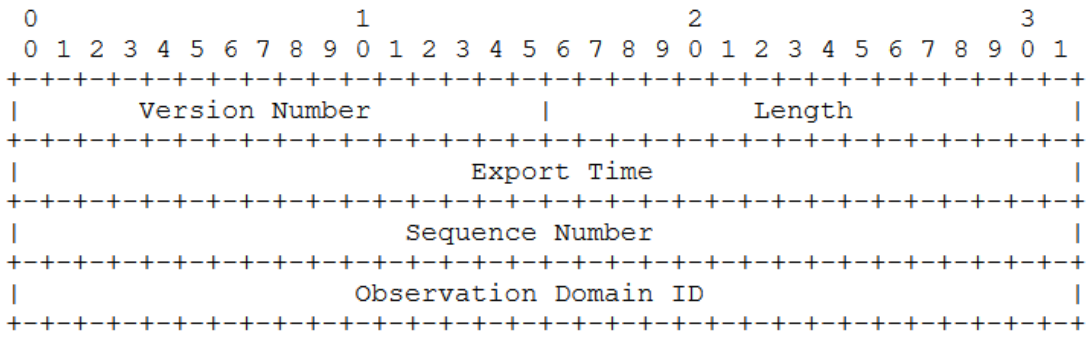


Figure 7. IPFIX header format

IPFIX messages carry records by wrapping them into sets. Set is a term for a collection of records with a similar structure. The message can include zero or more sets. Three different set types exist: template sets, options template sets and data sets. Each expresses that IPFIX set includes one or more times the same record type. In usual usage, template sets do not need to be sent in every message. When collector already has received template information it knows what information data records include. The majority of messages consist solely from data sets. Message format is pictured in Figure 8 (RFC7011).

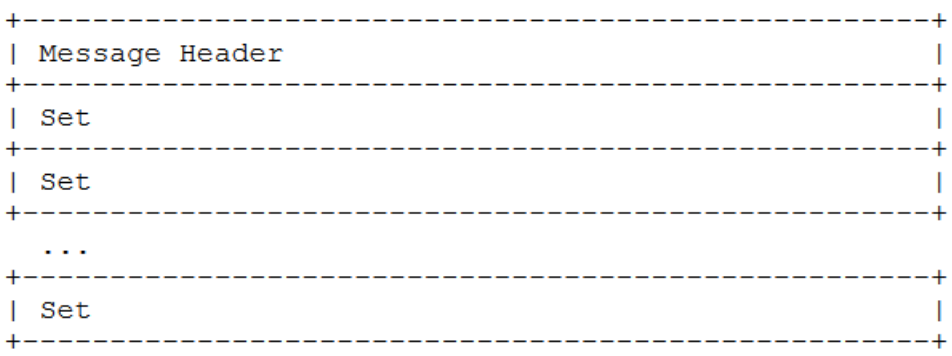


Figure 8. IPFIX Message format

Information element is the description of an IPFIX attribute. Information elements are registered by IANA and each has its own identifier. The type of Information element defines its encoding and what it could contain. Different information element types were originally defined in RFC5102. In addition to IANA registered values, also vendor specific identifiers exist. The enterprise bit with information element identifier id and enterprise number field compose the field specifier. Field specifiers are sent from exporter to collector in template sets nested in template records. (RFC7011, RFC7012)

Sets wrap records in themselves. Three different record formats exist: template record format, options template record format and data record format. The three set types explained earlier follow the naming of records. (RFC7011)

A template record includes one or more field specifiers. Field specifiers show what information elements are used in IPFIX transmission. Each record can mix IANA registered standard and vendor specific information elements. Collector process does not need to receive the description of data sent in every message. Actual data records are sent more often than template records. Every template record has its unique id per transport session and the observation id. (ibid.)

Options template records are much like template records. They provide additional information from the flow records and metering process. Options template records introduce the scope parameter. The scope differentiates some field specifiers to be handled as options template information. Field specifiers marked as a scope introduce extra information from IPFIX exporter, such as reliability information. (RFC7011)

Data records are wrapped in data sets. The data records are stripped of extra information and only include field values. Collector interprets field values with the set id included in the data set header. Set id equals to template id that defines field values in data records included in data sets. (ibid.)

### 2.6.4 Netflow vs IPFIX

NetFlow version 9 export protocol and IPFIX export protocol, are both transport layer independent. NetFlow uses UDP as default and IPFIX defines SCTP support as a mandatory requirement for implementations. (Data Collection Protocols: Netflow Version 9 and IPFIX Export Protocols)

IPFIX requirements document RFC3917 specifies that protocol should provide mechanisms for Confidentiality, Integrity and Authenticity (CIA). IPFIX defines the support for data encryption as a mandatory requirement. DTLS for SCTP and UDP and TLS for TCP must be supported. In addition, allowed IP ranges can be used as a security measure. IPFIX also instructs to protect collectors and collected data, even if it is not in the scope of the RFC. Netflow v9 export method does not define encoding of transmitted data. (RFC3954; RFC7011)

IPFIX sends data records in a format of Information Elements, which include information about the transmitted data. NetFlow calls these elements as a Field Type. Different Information elements have unique id from 1 to 32767. However, ids

from 1 to 127 are reserved for compatibility with NetFlow version9. IPFIX also allows creation of vendor specific information elements. IPFIX uses same IANA registered private enterprise numbers (PEN) as SNMP. (RFC7011, RFC7012) Currently the last register element holds number 482. (IP Flow Information Export (IPFIX) Entities 2018) Cisco lists over a hundred predefined field types as well as vendor specific field type (Netflow Version 9 Flow-Record Format 2011)

IPFIX allows variable length fields in its information elements. This allows optimization of messages that include short information elements. On the other hand, it makes possible to attach longer patterns into information element field. NetFlow does not have this feature. (RFC7011, RFC3954)

## 2.7 Legislation

A Finnish service provider has to obey the Finnish and EU legislation, laws and regulations. The major legislation reform called Information Society Code came into effect in 2015. The new code replaced several laws about electrical communications (Neuvonen 2015)

Information Society Code defines concepts of traffic data. It means data used in information delivery from wich a user can be identified, or information can be associated with a user. For example ip address or routing or time of a connection could be identification data. Baseline of the law is that, traffic data may only be processed as much as it is necessary for transmitting message. (Processing of traffic data 2018), (Information Society Code 2014, Chapter 17 section 137)

Other terms relevant for the scope of thesis are communications provider "*communications provider means a telecommunications operator, corporate subscriber or other party that conveys electronic communications for other than personal or comparable customary private purposes*" and corporate or association subscriber "*corporate or association subscriber means an undertaking or organisation which subscribes to a communications service or an added value service*

*and which processes users' messages, traffic data or location data in its communications network*" (Information Society Code 2014, 4)

It is important to notice that handling traffic data has many restrictions applied by the Information security code. On the other hand, if the definition of traffic data is not fulfilled, i.e the data cannot be associated with a specific user, statistical data may be produced also for other purposes that are mentioned in the law text. The law also introduces responsibilities. Communications provider and corporate or association users´ obligations concerning especially the scope of thesis are introduced in next chapter.

Information Security Code section 247 defines that communications providers must maintain the information security of their services, messages, traffic data and location data. Corporate or association subscribers have to maintain the same aspects of security towards their users. (Information Society Code 2014, 93)

Section 243 defines how public communication networks and communications services shall be planned, built and maintained. The section defines among other regulations that services must be implemented in a way that quality and reliability of functionality can be monitored and significant security violations or threats against them and other defects and significant disruptions can be detected. Information security measures ensuring the security of operations, communications, equipment, programmes and security information material shall be commensurated with the seriousness of threats, the level of technical development to defend the threat and costs incurred by the measures. (Information Society Code 2014, 90-91)

The communications provider general processing principles are defined in section 137. It is enacted that processing electronic messages and traffic data is only allowed to the extent necessary of the purpose, and it may not limit the confidentiality of messages or the protection of privacy any more than necessary. Information may only be disclosed to parties entitled to process them in the given situation. After processing, data must be destroyed or anonymized in a way that it cannot be

associated with the subscriber or user involved, unless otherwise provided by law. (Information Society Code 2014, 54-55)

Communications provider has oblication to maintain information security. (Information Society Code 2014, 93) A telecommunications operator or corporate or association subscriber has right to automatic processing of traffic data and message contents to detect, prevent or investigate information security disruptions. Also, communication possibilities ensurement and payment card fraud prevention justify automatic analysis. The contents of single message may only be processed manually, if it is evidently malicious and automatic measures cannot reach the beforementioned goals. Manual handling of messages shall be implemented with care and message participants shall be informed of the processing. (Information Society Code 2014, 102-103)

In respect of Information Society Code operators can use traffic data to form statistics for pricing and financial planning. After statistics are analysed all identifiable data must be destroyed or anonymized. Statistics can be formed for other purposes if they cannot be associated to specific user. (Processing of identification data – The rights and obligations of corporate and association subscribers 2015)

EU wide General Data Protection Regulation came into effect on 25 May 2018. It applies to any organization that stores data relating to EU residents. GDPR divides data handling parties as data controllers and data processors. Data controllers define why data is collected and how it is handled, however, they might not themselves participate in data handling. The data processor, instead, does not own the data; it just collects or uses the data. This division into data controllers and processors is important when defining GDPR responsibilities between companies. (What is GDPR 2018)

GDPR has six principles. The first is transparency and lawfulness of the gathered data. Secondly, the organizations should gather only data that has a specific purpose. The third principle is data minimization. Only the minimum amount of data to reach the purpose should be collected. The fourth is accuracy: the saved data should be kept on date, and inaccurate or incomplete data should not be saved. The fifth principle is storage limitation. This principle rules that information that is not needed anymore should be deleted. The last one is integrity and confidentiality. Personal data must stored securily and must be protected against unauthorized usage or accidential loss. (Irwin 2018)

## 2.8 Flow based IDS

IDS in computer networks is a technology that detects vulnerability exploits against a protected asset. Intrusion Prevention System (IPS) adds the prevention aspect on the top of the plain detection of IDS. (Paloaltonetworks 2018)

Intrusion detection systems can be divided into Network based (NIDS) and Host based (HIDS) Intrusion detection systems. As the name indicates, network based IDS examines traffic traversing observation point that is in some network equipment. Host based IDS systems are installed in network endpoints, such as end user computers. The development of NIDS was started in the 1990s soon after host based IDSs were introduced. (Stênico & Lee 2014, Chapter 2)

Intrusion detection systems can be classified into two different methods. Signature based IDSs search already known patterns from the monitored data. Therefore signature based implementations need to have knowledge about the patterns that are known to be malicious. Usually this is achieved with database holding information about pre-known patterns. Another type of IDS is anomality-based Intrusion Detection. This type of IDS tries to accomplish expected behavior model, a baseline about the monitored traffic. Any difference in normal behavior or data

pattern is handled as an anomality and alert is raised. (Stênico & Lee 2014, Chapter 2)

Traditionally, IDSs have examined whole packets including the payload. For larger traffic amounts, this is not efficient enough. This is where flow based IDS comes up. Flow information does not include payload data and is therefore more lightweight to examine than whole IP packets. A flow based NIDS receive less data than traditional NIDSs. On the other hand, the payload processing NIDSs have more information to search exploits and are therefore more accurate. These two types of NIDS can be considered as complements.

# 3 Review of the monitoring methods

## 3.1 Background to study

In the current situation, all CPEs are monitored with commercial network monitoring software. All routers are monitored with ICMP ping and SNMP pull based monitoring. Email and/or SMS alerts are triggered when ping loss occurs. Selected SNMP mibs are collected in five minutes interval. MIBs include interface counters, router health and environmental information. History information for all measurements is collected. This is the basic monitoring set that is provisioned for all CPEs. For more demanding monitorings the software allows different customizations. Customer portal allowing customers to view their connection´s monitoring graphics is also in use.

The collected information is saved for later usage. Excluding ping loss alerting, there is no real-time analysis of data. Saved data is mainly used for fault finding by service desk and for SLA availability statistic calculations.

SNMP pull based monitoring can transfer various information from monitored equipment. Enterprise specific MIBs allow e.g. DPI application information being

transferred via SNMP. When only standard MIB information is collected, the information consists mainly from equipment system, health, environment and counter information. The probability of the collected data being identified to a single user is small.

SNMP provides real time info via its trap mechanism. Traps inform from state changes or threshold crossings. Nevertheless, ordinary counter based measurements are always averaging packets or bits based on poll interval. Flow monitoring can provide nearly realtime information from capacity usage. In addition to time advantage, also more in depth information from traffic flows can be gathered. Flow monitoring could be placed between counter based SNMP polling and different DPI methods when examining depth of information gathering. When handling more detailed data, the possibilities to perform an analysis for different purposes are increased, as well as regulatory obligations.

In this study, flow exporters are Cisco 800 series ISR routers. The routers function as Customer Premises Equipment and handle all data entering or leaving a site. The monitoring points are limited to CPE equipment. Flow monitoring is tested and the information gathered could be viewed and analyzed with open source software. No extra equipment is needed since routers can be configured to function as flow exporters and the software is installed on an existing platform. The tested routers support IPFIX flow transport when Flexible NetFlow is used. Even though IPFIX can be used as transport protocol, only UDP can be used as a transport layer protocol. The lack of TCP and SCTP transport protocol support does not fulfill the standard.

In addition to the lack of TCP and STCP support, also TLS and DTLS are not supported on the equipment. This is a drawback when flow monitoring data should be exported over the Internet. In that case, some other protection method must be implemented.

Collector receives flow data from exporters. It must support transport methods that exporters utilize. Optionally, it performs flow data aggregation, filtering or other actions before forwarding data to other instances. Other instances could be, for example, a database, an analyzer or a flat file.

## 3.2 Research software

Most diverse open source flow monitoring implementations consist of different software elements. These components can be combined based on one´s needs. Each software takes care of its own functionality in the architecture. Figure 9 shows the example architecture.

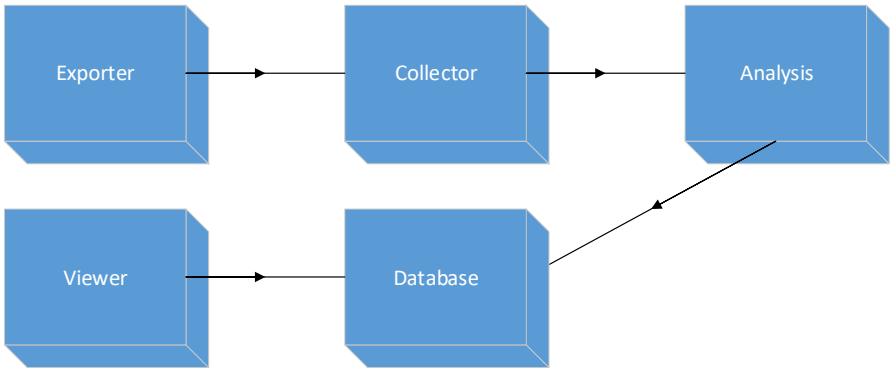


Figure 9. Example of flow monitoring architecture

When exploring the field of open source flow tools, it was noticed that there is a wide variety of software for this purpose. During this research, it was not possible to test them all. Instead, the focus is on finding a frame that works and could be developed and used to try other software as well. Software packages such as ELK and TICK stacks work; however, on their own, they are not as diverse and developer friendly as open source software could be.

Flow information usage is increasing as its usage is expanding, for example, to routing decisions and security purposes. Among other things, the possibilities of IPFIX to transport a wide variety of data support this evolution. It could be useful to build a system that can flexibly import new software for different purposes. Apache Kafka is

one solution to tie several software together. Figure 10 shows flow monitoring architecture with message broker.

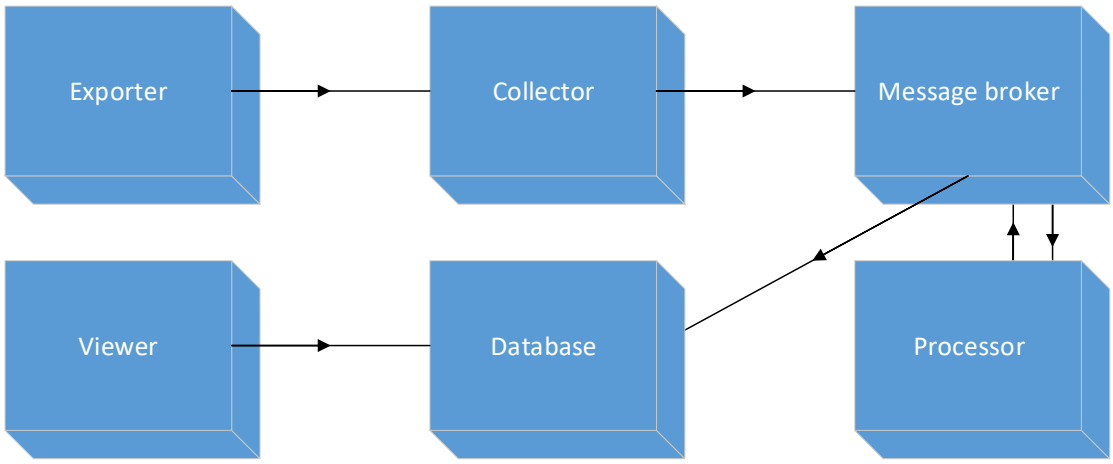


Figure 10. Flow monitoring architecture with message broker.

The collected information should be anonymized and stored, for example forensics could utilize flow data. The storage could be flat files or somekind of a database. There is a variety of database solutions available. Netflow information is strongly time based. There are time-based oriented databases that should give an efficient and scalable solution. However, the amount of flow information is not expected to be large; hence, performance optimization is not the main focus.

### 3.1.1 ELK Stack

ELK stack is an open source database with GUI search and analysis engine. It consists of three main software elements. It uses Logstash for collector, Elasticsearch for database and Kibana for graphics tool. Elastic.co company leads the development of

these software. Base software is open source; however, Elastic.co provides chargeable support and add-ons.

Logstash is an open source collector program that receives data, which is usually log data, processes it to fit the database form and passes it to a database. The first data is input into Logstash from different data sources. Many different kinds of data inputs are supported. After the data has been input into Logstash, it is handled by Logstash filters. Filters modify and unify data to fit into the used database. Filters can also resolve the geo-location of an IP address or anonymize sensitive data, for example. Lastly, Logstash outputs unified data into database. A variety of output forms is supported. Inputs, filters and outputs are highly customizable with over 200 plugins. (Centralize, Transform & Stash Your Data 2018)

Elasticsearch is the database of ELK stack. It is built on Apache Lucene. It scales from a single instance in one server to a cluster with many instances, or nodes in different servers. It is open source and it has REST API. (Elasticsearch 2018)

The stored data is divided into indices. A single index can be divided into several nodes in different servers. The part of index data is called a shard. Shards can be divided into different servers to provide scalability. Single index contains data that has similar characteristic. ES implementation can support hundreds of indices. A basic chunk of indexed data is called a document. For example, one syslog message is saved as a document. Documents are saved in JSON format. (Basic Concepts 2019)
Kibana is a visualization component that shows e.g. graphs, histograms, and charts from Elasticsearch data.

### 3.1.2 Kafka

Kafka is a distributed streaming platform. It holds and transports streams of records. It is used to create real-time data pipelines between systems and applications. Kafka provides four APIs. Producer API is input into Kafka. Producers publish or stream records into Kafka via producer API. Consumer API is used by systems or applications

that read data from Kafka. Streams API makes it possible for a system or an application to consume data from Kafka and put it back into Kafka after processing the data. The last one is connector API. Connector API allows running reusable consumers or producers to connect Kafka into other systems.

## 3.3 System review

Flexible NetFlow was tested with IPFIX export protocol with UDP as transport protocol. Flexible NetFlow is versatile, however, in this case basic monitoring was utilized. Below is an example of general configuration.

Flexible NetFlow allows to define flow key fields and additional fields that provide extra information. It was tested with Cisco router C881G+7-K9. Software 15.6(3)M5. Traditional five-tuple is used to define a flow. Cisco provides ready templates for Flexible NetFlow key parameter defining. Flow record template netflow-original defines key and collected fields as follows:

*Router#sh flow record netflow-original*

*flow record netflow-original:*

*Description: Traditional IPv4 input NetFlow with origin ASs*

*No. of users: 1*

*Total field space: 53 bytes*

*Fields:*

*match ipv4 tos*

*match ipv4 protocol*

*match ipv4 source address*

*match ipv4 destination address*

*match transport source-port*

*match transport destination-port*

*match interface input*

*match flow sampler*

*collect routing source as*

*collect routing destination as*

*collect routing next-hop address ipv4*

*collect ipv4 source mask*

*collect ipv4 destination mask*

*collect transport tcp flags*

*collect interface output*

*collect counter bytes*

*collect counter packets*

*collect timestamp sys-uptime first*

*collect timestamp sys-uptime last*

NetFlow version 9 and IPFIX templates look very similar. This is due to the IPFIX backwards compatibility. Figure 11 shows NetFlow version 9 template when using Cisco Flexible NetFlow built in record template named Netflow-original.

```
Router#sh flow exporter EXPORTER1 templates
Flow Exporter EXPORTER1:
  Client: Flow Monitor MONITOR1
  Exporter Format: NetFlow Version 9
  Template ID    : 256
  Source ID      : 0
  Record Size    : 53
  Template layout
  ----------------------------------------------------------------------------
  |                Field                        |  Type  | Offset |  Size   |
  ----------------------------------------------------------------------------
  | ipv4 source address                         |      8 |      0 |       4 |
  | ipv4 destination address                    |     12 |      4 |       4 |
  | flow sampler                                |     48 |      8 |       4 |
  | interface input snmp                        |     10 |     12 |       4 |
  | transport source-port                       |      7 |     16 |       2 |
  | transport destination-port                  |     11 |     18 |       2 |
  | ip tos                                      |      5 |     20 |       1 |
  | ip protocol                                 |      4 |     21 |       1 |
  | ipv4 source mask                            |      9 |     22 |       1 |
  | ipv4 destination mask                       |     13 |     23 |       1 |
  | transport tcp flags                         |      6 |     24 |       1 |
  | routing source as                           |     16 |     25 |       2 |
  | routing destination as                      |     17 |     27 |       2 |
  | routing next-hop address ipv4               |     15 |     29 |       4 |
  | counter bytes                               |      1 |     33 |       4 |
  | counter packets                             |      2 |     37 |       4 |
  | timestamp sys-uptime first                  |     22 |     41 |       4 |
  | timestamp sys-uptime last                   |     21 |     45 |       4 |
  | interface output snmp                       |     14 |     49 |       4 |
  ----------------------------------------------------------------------------
```

Figure 11 Netflow version 9 template when using Cisco Flexible NetFlow built-in flow record netflow-original

When export protocol is changed to IPFIX, the interoperability between NetFlow v9 and IPFIX is maintained. Data field numberings are not chanded. Figure 12 shows IPFIX template record with same set of exported parameters that is used in NetFlow v9 template record in Figure 11.

```
Router#sh flow exporter EXPORTER2 templates
Flow Exporter EXPORTER2:
  Client: Flow Monitor MONITOR2
  Exporter Format: IPFIX (Version 10)
  Template ID    : 256
  Source ID      : 0
  Record Size    : 53
  Template layout
  ----------------------------------------------------------------------------
  |                Field                |    ID | Ent.ID | Offset |  Size |
  ----------------------------------------------------------------------------
  | ipv4 source address                 |     8 |        |      0 |     4 |
  | ipv4 destination address            |    12 |        |      4 |     4 |
  | flow sampler                        |    48 |        |      8 |     4 |
  | interface input snmp                |    10 |        |     12 |     4 |
  | transport source-port               |     7 |        |     16 |     2 |
  | transport destination-port          |    11 |        |     18 |     2 |
  | ip tos                              |     5 |        |     20 |     1 |
  | ip protocol                         |     4 |        |     21 |     1 |
  | ipv4 source mask                    |     9 |        |     22 |     1 |
  | ipv4 destination mask               |    13 |        |     23 |     1 |
  | transport tcp flags                 |     6 |        |     24 |     1 |
  | routing source as                   |    16 |        |     25 |     2 |
  | routing destination as              |    17 |        |     27 |     2 |
  | routing next-hop address ipv4       |    15 |        |     29 |     4 |
  | counter bytes                       |     1 |        |     33 |     4 |
  | counter packets                     |     2 |        |     37 |     4 |
  | timestamp sys-uptime first          |    22 |        |     41 |     4 |
  | timestamp sys-uptime last           |    21 |        |     45 |     4 |
  | interface output snmp               |    14 |        |     49 |     4 |
  ----------------------------------------------------------------------------
```

Figure 12. IPFIX template when using Cisco Flexible NetFlow built-in flow record netflow-original

All tested software components were installed on virtual Linux server. The virtual server used for testing was running CentOS version 7.4.1708 (Core). Four virtual CPUs and 8 Gigabytes of memory were allocated. The author did not have much experience with Unix or Linux systems, which made testing challenging.

The first implementation was ELK stack. ELK stack uses Elasticsearch as its database, Logstash as aggregator and filter and Kibana for visualization. The installation of ELK stack is quite well documented, and the software is built to work together. The basic set provides visibility, however, many features have to be installed separately as a plugin. For example, alerting and Kibana web UI security could be installed with x-pack software plugin. There were some commercial parts in x-pack and also some incompatibility with the software versions. It was decided not to install the x-pack. To gain security for Kibana, web UI NGINX was installed as web proxy to provide HTTPS

connection to Kibana. Later, the x-pack was renamed as Stack Features. ELK stack has good visualizations through Kibana. In addition, Timelion integrated in Kibana is a software for time-based database graphing.

There are quite many settings when installing and configuring Logstash and Elasticsearch to receive flow data. Open source plugin Elastiflow, by Koiossian, makes settings ready and creates a ready to use dashboard for graphics.

The second implementation was built with collector pmacct and database InfluxDB. Apache Kafka was installed as message broker. Nfacct that is NetFlow and IPFIX portion of Pmacct, was used to collect data and deliver it to Kafka. Between Kafka and InfluxDB there was implementation of Telegraf. Telegraf read data from Kafka and delivered it into InfluxDB. To gain graphical view, Chronograf was also installed. Implementation involved several different software with their configurations. Data was successfully sent into InfluxDB and viewed with Chronograf. Due to the complexity of the implementation, there were no further configuration adjustments after basic flow data was successfully saved into the database.

## 3.3 Service operation point of view

When discussing ITIL processes, new services are formed on the ground of Service Strategy and Service Design. When bringing new tools into existing processes, Service Transition processes alone could give adequate control. It must be concidered that before building new tools into production, careful planning should be conducted. New software must be implemented at least with regulation, legislation and business targets in mind. These aspects must be emphasized especially when any privacy sensitive data is handled and/or saved. In case of bringing flow monitoring system into service toolbox in addition to SNMP, at least Service Design processes are required because of privacy consideration.

Monitoring tools have an important role in ITIL Service Operations processes. The most efficient usage of monitoring prevents uncontrolled degradation of service.

Network monitoring indications in ITIL Event Management can be used to create a baseline of the current state. Any exceptions in status quo can be passed on to a proper process. The necessary manual or automated actions are followed and in the best case, service is not affected at all. The minimum set of needed automation is automated events and/or alerts from monitoring system. For example, in network monitoring capacity thresholds and in flow monitoring DDOS indicators should raise automated events.

Incident and Problem Management processes use largely monitoring tools. History information of measured values is valuable, especially when Problem Management process examine root causes of incidents, i.e. problems. These processes involve human interaction. For example, Service desk, Technical or Application Management need to use monitoring tools and they need to have proper training for tools they use. Inadequate training should be visible in key performance indicators, such as resolution times. When attaching the abovesaid into ITIL processes, existing SNMP monitoring and appending flow monitoring, there is need for training. Flow monitoring gives a completely new aspect into dataflows that SNMP monitoring cannot provide. For example, information from geographical distribution of IP traffic can be used in incident solving. The staff is a valuable source of information when practicing Continual Service Improvement. They know best how existing processes, methods and tools could be improved.

# 4 Conclusions

Flow based information gathering is increasing its popularity as a part of more sophisticated solutions. Different kinds of security purposes, routing decisions, even software defined networking, take advantage of information gathered from flows. The popularity can be seen on the volume of open source software. Software for e.g. graphing, alerting, anomaly detection, and machine learning has been developed.

Open source solutions provide functional frameworks, such as ELK stack. However, when certain software stack is installed, the leeway in software choices is constricted. Message broker system is one way to combine the different software for e.g. analysis or anonymization purposis into the implementation more freely.

Flow monitoring gives sight into data transfers and their endpoints. It improves visibility compared to plain counter view gained with SNMP. Straight benefit for fault finding process is the ability to point out hosts that hog the capacity of the line. Flows directed to rarely used addresses or locations can also be a sign of unwanted activity. With analysis, software flow monitoring is a great addition to network monitoring. However, current SNMP monitoring should not be replaced with flow monitoring. For example, environmental or equipment health values are not included in flow monitoring.

Open source software was implemented to gain understanding how it could be used in production environment. Ready software stacks, such as ELK, give the easiest way to implement free open source software. Some of software aggregates are providing support for a fee. Choosing ready software stack might tie the implementor to certain software and the variety of usable software might be narrowed. On the other hand, when an implementor builds its own framework from a variety of software, there could be excessive complexity. The implementor should consider what demands monitoring should fulfill and how critical it is for processes and the entirety. This, with resources available, should guide the decision for software components.

The lack of encryption capability is a drawback. However, most of the equipment is connected through trusted lines and unencrypted flow data export could be used. Nevertheless, encryption in transport should be implemented if possible.

Legislation plays a significant role in accurate network monitoring. In the scope of the research, the assignor company is treated as communications provider providing a communications service. The subscribers would be defined as corporate or association subscribers and in some cases end users could be defined as users. Even though the communications provider is obligated to take care of the availability,

usability and security of the provided service traffic, monitoring data should not be possible to be combined with a user in a normal operation situation. This forces additional attention to information gathering and storing of traffic data. The thesis focuses on adding visibility between MPLS network branch offices. Sometimes offices have only one or few users. In this situation, the gathered traffic data could be pointed to a single user. In some cases, a person can be pointed using his/her IP or MAC address. Information should be gathered and/or stored in a way that it does not fulfill law interpretation of traffic data that can be linked to a user. Also, the storaging time of monitored data should be decided.

The first research problem was to evaluate what additional benefits data flow monitoring in CPEs can provide. The obvious increase in visibility in data flows was discovered. There are possibilities to gain more benefits. Those could be achieved by processing and analyzing gathered data. It would be interesting to explore possibilities of flow data with different kind of analyzers. Also, route monitoring and flow assisted routing is a rising trend and a step towards software defined network.

The second research problem was how data flow monitoring could be implemented in the assignors' environment. Basic installations were conducted with open source software. Proof of concept was achieved. However, security and maintenance issues were not handled and legislation was not concerned in the implementations. All these areas need more in depth examination before the concept is ready for production. It was learnt that it is possible to build a working environment with open source software. This just needs more knowledge and resource, than commercial products.

# References

About AXELOS. Page on Axelos.com website. Accessed 27 February 2018. Retrieved from https://www.axelos.com/about-axelos

About us. 2018. Page on tmforum.org website. Accessed 6 November 2018. Retrieved from https://www.tmforum.org/about-tm-forum/

Adeel, A. Madani, H. & Siddiqui, T. 2011. VoIP Performance Management and Optimization: A KPI-Based Approach to Managing and Optimizing VoIP Networks. Cisco press. Accessed 26 October 2018. Retrieved from https://library-books24x7-com.ezproxy.jamk.fi:2443/assetviewer.aspx?bookid=45401&chunkid=406174150&noteMenuToggle=0&hitSectionMenuToggle=0&leftMenuState=1

Aina Group Plc bulletin 24 August 2018. Page on Ainagroup.fi website. Accessed 8 November 2018. Retrieved from http://www.ainagroup.fi/tiedotus/node/aina-group-myy-ainacomin-telialle

Aina Group Plc Quarterly report January – June 2017. Bulletin August 8 2017. Accessed 26 February 2018. Retrieved from http://www.ainagroup.fi/img/file.php?id=47505

An Introduction to ISO 27001, ISO 27002....ISO 27008. 2018. Page on 27000.org website. Accessed 6 November 2018. Retrieved from http://27000.org/index.htm

Architectural and Framework Standards: The TMN/FCAPS Model (ITU-T). eTutorials.org. Accessed 5 November 2018. Retrieved from http://etutorials.org/Networking/network+management/Part+I+Data+Collection+and+Methodology+Standards/Chapter+3.+Accounting+and+Performance+Standards+and+Definitions/Architectural+and+Framework+Standards+The+TMN+FCAPS+Model+ITU-T/

Basic Concepts 2019. Page on elastic.co. Accessed 2 January 2019. Retrieved from https://www.elastic.co/guide/en/elasticsearch/reference/current/getting-started-concepts.html.

Centralize, Transform & Stash Your Data 2018. Page on elastic.co. Accessed 13 November 2018. Retrieved from https://www.elastic.co/products/logstash

Cisco ASA Series General Operations CLI Configuration Guide, 9.2. Cisco.com. Accessed 1 November 2018. Retrieved from https://www.cisco.com/c/en/us/td/docs/security/asa/asa92/configuration/general/asa-general-cli/monitor-nsel.html

Cisco IOS Flexible NetFlow Technology Q&A. Page on cisco.com. Updated June 19, 2006. Accessed 3 November 2018. Retrieved from https://www.cisco.com/c/en/us/products/collateral/ios-nx-os-software/flexible-netflow/prod_qas0900aecd804be091.html

Claise Benoit, Trammell Brian. Applying IPFIX to Network Measurement and Management. Presentation in IETF 87 Berlin, Germany. 28th July 2013. Accessed 20 January 2018. Recording retrieved from https://youtu.be/2qxZq97TErQ

Data Collection Protocols: Netflow Version 9 and IPFIX Export Protocols. Etutorials.org. Accessed 13 January 2018. Retrieved from etutorials.org/Networking/network+management/Part+I+Data+Collection+and+Methodology+Standards/Chapter+3.+Accounting+and+Performance+Standards+and+Definitions/Data+Collection+Protocols+NetFlow+Version+9+and+IPFIX+Export+Protocols/

Elasticsearch 2018. Page on amazon.com. Accessed 13 November 2018. Retrieved from https://aws.amazon.com/elasticsearch-service/what-is-elasticsearch/

IETF. RFC3954 Cisco Systems NetFlow Services Export Version 9. Accessed 13 January 2018. Retrieved from https://www.ietf.org/rfc/rfc3954.txt

NetFlow gives network managers a detailed view of application flows on the network 2003. Cisco Systems publication. Case Study. Accessed 26 November 2017. Retrieved from https://www.cisco.com/c/dam/en/us/products/collateral/ios-nx-os-software/ios-netflow/prod_case_study0900aecd80311fc2.pdf

Farrell, A. 2009. Network management: Know it all. Morgan Kaufmann Publishers. Accessed 27 February 2018. Retrieved from https://library-books24x7-com.ezproxy.jamk.fi:2443/toc.aspx?bookid=39946

Hofstede, Čeleda, Trammell, Drago, Sadre, Sperotto & Pras. Flow Monitoring Explained: From Packet Capture to Data Analysis with NetFlow and IPFIX. 2014 IEEE communication surveys & tutorials, VOL. 16, NO. 4. Accessed 10 December 2018. Retrieved from https://ieeexplore-ieee-org.ezproxy.jamk.fi:2443/stamp/stamp.jsp?tp=&arnumber=6814316

Horwitt, E. OSI forum carves out its niche. Computerworld 29 May 1989. Accessed 6 November 2018. Retrieved from https://books.google.fi/books?id=VanY7cqYD2wC&pg=PP44&lpg=PP44&dq=%22network+management+forum%22+was+founded&source=bl&ots=OPXUW9_qXC&sig=8R7Pz_fOXnDTAID--ujXAjNF_Jc&hl=fi&sa=X&ved=2ahUKEwjaoOy2n7_eAhWJjCwKHal9DWEQ6AEwBnoECA0QAQ#v=onepage&q=%22network%20management%20forum%22%20was%20founded&f=false

Härjämäki T. AinaCom osaksi Telia-perhettä – AinaComille ja AinaPaylle uusi toimitusjohtaja. 2018. (a) Page on aina.fi website. Accessed 10 February 2019.

Retrieved from http://www.aina.fi/ajankohtaista/ainacom-osaksi-telia-perhett%C3%A4-ainacomille-ja-ainapaylle-uusi-toimitusjohtaja.

Härjämäki T. AinaComista Telia Communication. 2018. (b) Page on aina.fi website. Accessed 10 February 2019. Retrieved from http://www.aina.fi/ajankohtaista/ainacomista-telia-communication

IETF. IPFIX Status Pages. Updated 24 Oct 2012. Accessed 30 December 2017. Retrieved from https://tools.ietf.org/wg/ipfix/

In a Nutshell: A Short History of ITIL 2005. Page on ITIL Central. Accessed 27 February 2018. Retrieved from http://itsm.fwtk.org/History.htm

Introduction to eTOM. 2009. Technology white paper on Cisco.com. Updated June 18, 2009. Accessed 6 November 2018. Retrieved from https://www.cisco.com/c/en/us/products/collateral/services/high-availability/white_paper_c11-541448.html

Information Society Code 2014. Ministry of Transport and Communications, Finland. Accessed 7 January 2018. Retrieved from https://www.finlex.fi/fi/laki/kaannokset/2014/en20140917.pdf

IP Flow Information Export (IPFIX) Entities iana.org Accessed 14 December 2018. Retrieved from https://www.iana.org/assignments/ipfix/ipfix.xhtml

Irwin, L 2018. The GDPR: Understanding the 6 data protection principles 2018. Accessed 10 November 2018. Retrieved from https://www.itgovernance.eu/blog/en/the-gdpr-understanding-the-6-data-protection-principles

ISO/CCITTT and Internet Management Mapping 1997. Systems management Reference Model. The Open Group. Accessed 1 March 2018. Retrieved from http://pubs.opengroup.org/onlinepubs/9279299/apdxb.htm

ITIL-update. Page on Axelos.com website. Accessed 8 November 2018. Retrieved from https://www.axelos.com/itil-update

ITU-T M.3050.1. 2007. Enhanced Telecom Operations map eTOM. Accessed 6 November 2018. Retrieved from https://www.itu.int/rec/T-REC-M.3050.1/recommendation.asp?lang=en&parent=T-REC-M.3050.1-200703-I

ITU-T Recommendations. 2018. Page on ITU-T.int website. Accessed 6 November 2018. Retrieved from https://www.itu.int/itu-t/recommendations/index.aspx?ser=M

Jianguo, D 2010. Advances in Network Management. Auerbach Publications. Chapter 3 - Evolution in Network. Accessed 10 February 2019. Retrieved from https://library-books24x7-com.ezproxy.jamk.fi:2443/toc.aspx?bookid=32104

Juniper Flow Monitoring 2011. J-Flow on J Series Services Routers and Branch SRX Series Services Gateways. Juniper networks. Application note. Accessed 13 January 2018. Retrieved from https://www.juniper.net/us/en/local/pdf/app-notes/3500204-en.pdf

M.3010. Principles for a telecommunications management network. 02/2000. ITU-T Recommendation M.3010. Accessed 3 November 2018. Retrieved from https://www.itu.int/rec/dologin_pub.asp?lang=e&id=T-REC-M.3010-200002-I!!PDF-E&type=items

Me olemme Aina. 2018. Page on AinaCom.fi website. Accessed 26 February 2018. Retrieved from http://www.ainacom.fi/ainacom.

Miessler, D. 2015 The Difference Between Events, Alerts, and Incidents. 6 January 2015. Accessed 6 November 2018. Retrieved from https://danielmiessler.com/study/event-alert-incident/

Morris H, Gallacher, L. 2016. John Wiley & Sons. ITIL® Intermediate Certification Companion Study Guide: Service Lifecycle Exams. Accessed 10 March 2018. Retrieved from https://library-books24x7-com.ezproxy.jamk.fi:2443/toc.aspx?bookid=132634

Muniz, J. Santos, O. 2017. NetFlow for Cybersecurity. Cisco Press. Accessed 1 March 2018. Retrieved from http://www.ciscopress.com/articles/article.asp?p=2812391&seqNum=3

Netflow Version 9 Flow-Record Format. Updated 2011. Cisco.com. Accessed 14 January 2018. Retrieved from https://www.cisco.com/en/US/technologies/tk648/tk362/technologies_white_paper09186a00800a3db9.html

Netstream. 2015. Page on support.huawei.com. Accessed 1 November 2018. Retrieved from http://support.huawei.com/enterprise/docinforeader!loadDocument1.action?contentId=DOC0100523133&partNo=10032

Neuvonen Riku. Tietoyhteiskuntakaari yhdisti sähköisen viestinnän lainsäädännön. 2015. Page on website sananvapauteen.fi. Accessed 6 January 2018. Retrieved from https://sananvapauteen.fi/artikkeli/974

OpenFlow Switch Specification Version 1.5.1 (Protocol version 0x06) March 26, 2015. Open Networking foundation publication. Accessed 9 December 2017. Retrieved from https://3vf60mmveq1g8vzn48q2o71a-wpengine.netdna-ssl.com/wp-content/uploads/2014/10/openflow-switch-v1.5.1.pdf

Park J.T, Choi Y.W, Jung J.W & Sunwoo J.S. The integration of OSI Network Management and TCP/IP Internet Management using SNMP. 1993. Published in: Proceedings of 1993 IEEE 1st International Workshop on Systems Management. Date Added to IEEE Xplore: 06 August 2002. Accessed 1 March 2018. Retrieved from http://ieeexplore.ieee.org.ezproxy.jamk.fi:2048/document/315279/

Pras, A. van Beijnum, B & Sprenkels,R. 1999. Introduction to TMN. CTIT Technical Report 99-09 April 1999 University of Twente The Netherlands. Accessed 4 November 2018. Retrieved from http://www.hit.bme.hu/~jakab/edu/litr/TMN/TMN_tutorial.pdf#G32777

Processing of identification data – The rights and opblications of corporate and association subscribers. Updated 24.2.2015 Finnish Communications Regulatory Authority. Accessed 6 January 2018. Retrieved from https://www.viestintavirasto.fi/en/cybersecurity/corporatesubscribersrightsandobligations/processingofidentificationdata.html

Processing of traffic data. Updated 29.05.2018. Finnish Communications Regulatory Authority. Accessed 3 November 2018. Retrieved from https://www.viestintavirasto.fi/en/cybersecurity/telecomsoperatorsrightsandobligations/processingoftrafficdata.html

Relationship to ITIL. tmforum.org. Page from tmforum.org website. Accessed 6 November 2018. Retrieved from https://www.tmforum.org/business-process-framework/relationship-to-itil/

RFC7011 Specification of the IP Flow Information Export (IPFIX) Protocol for the Exchange of Flow Information. IETF. Accessed 15 January 2018. Retrieved from https://tools.ietf.org/html/rfc7011

RFC7012 Information Model for IP Flow Information Export (IPFIX). IETF. Accessed 15 January 2018. Retrieved from https://tools.ietf.org/html/rfc7012

Russell, A. 2013. OSI: The Internet That Wasn't. IEEE Spectrum. Accessed 3 November 2018. Retrieved from https://spectrum.ieee.org/tech-history/cyberspace/osi-the-internet-that-wasnt

Santos, O. 2016. Network Security with NetFlow and IPFIX. Big Data Analytics for Information Security. Indianapolis: Cisco Press.

Sathyan, J. Fundamentals of EMS, NMS and OSS/BSS. 2010. Auerbach Publications. Accessed 5 November 2018. Retrieved from https://library-books24x7-com.ezproxy.jamk.fi:2443/toc.aspx?bookid=36936

SFS-EN ISO/IEC 27000:2017:en. Information security management systems. Overview and vocabulary. Accessed 6 November 2018. Retrieved from https://online.sfs.fi/fi/index/tuotteet/SFS/CENISO/ID2/2/474098.html.stx

SFS-EN ISO/IEC 27001:2017:en. Information security management systems. Accessed 6 November 2018. Retrieved from https://online.sfs.fi/fi/index/tuotteet/SFS/CENISO/ID2/2/474109.html.stx

Stages of ITIL Service Lifecycle. 2016 Page on expediuz.com. Accessed 1 December 2018. Retrieved from http://expediuz.com/stages-itil-service-lifecycle/

Stênico, J. Lee, L 2014. Network Traffic Monitoring and Analysis. Auerbach publications. Accessed 28 January 2018. Retrieved from https://library-books24x7-com.ezproxy.jamk.fi:2443/toc.aspx?bookid=61769

Trost, R. 2009. Practical Intrusion Analysis: Prevention and Detection for the Twenty-First Century. Pearson education. Accessed 10 March 2018. Retrieved from https://books.google.fi/books?id=3y2fhCaJJA0C&pg=PT188&lpg=PT188&dq=sflow+inmon+history+licensing&source=bl&ots=tiF7naSP48&sig=qIWz_n_suMSIGkjMeVKyrpGtE4A&hl=fi&sa=X&ved=0ahUKEwjDrfywvOLZAhWGyaYKHSgDCuwQ6AEIYjAI#v=onepage&q=sflow%20inmon%20history%20licensing&f=false

van Bon, J & Verheijen, T. Fremeworks for IT Management. itSMF-NL 2006. Accessed 6 November 2018. Retrieved from http://www.vanharen.net/Player/eKnowledge/introduction_-_etom_the_enhanced_telecom_operations_map.pdf

What is an intrusion detection system IDS. Paloaltonetworks.com. Accessed 13 January 2018. Retrieved from https://www.paloaltonetworks.com/cyberpedia/what-is-an-intrusion-detection-system-ids

What is GDPR? Everything you need to know, from requirements to fines. 2018. Page on itpro.co.uk. Accessed 27 November 2018. Retrieved from https://www.itpro.co.uk/it-legislation/27814/what-is-gdpr-everything-you-need-to-know

What is ITIL Best Practice? Page on Axelos.com website. Accessed 27 February 2018. Retrieved from https://www.axelos.com/best-practice-solutions/itil/what-is-itil

Yemini, Yechiam. 1993. The OSI network management model. IEEE Communications magazine 31(5):20-29 June 1993. Accessed 4 November 2018. Retrieved from https://www.researchgate.net/publication/3195162_The_OSI_network_management_model

# Appendices

Appendix 1. RFC3954 Netflow version 9 FlowSets.

| Field Type | Value | Length (bytes) | Description |
|---|---|---|---|
| IN_BYTES | 1 | N | Incoming counter with length N x 8 bits for the number of bytes associated with an IP Flow. By default<br>N is 4 |
| IN_PKTS | 2 | N | Incoming counter with length N x 8 bits for the number of packets associated with an IP Flow.<br>By default N is 4 |
| FLOWS | 3 | N | Number of Flows that were aggregated; by default N is 4 |
| PROTOCOL | 4 | 1 | IP protocol byte |
| TOS | 5 | 1 | Type of service byte setting when entering the incoming interface |
| TCP_FLAGS | 6 | 1 | TCP flags; cumulative of all the TCP flags seen in this Flow |
| L4_SRC_PORT | 7 | 2 | TCP/UDP source port number (for example, FTP, Telnet, or equivalent) |
| IPV4_SRC_ADDR | 8 | 4 | IPv4 source address |
| SRC_MASK | 9 | 1 | The number of contiguous bits in the source subnet mask (i.e., the mask in slash notation) |
| INPUT_SNMP | 10 | N | Input interface index. By default N is 2, but higher values can be used |
| L4_DST_PORT | 11 | 2 | TCP/UDP destination port number (for example, FTP, Telnet, or equivalent) |
| IPV4_DST_ADDR | 12 | 4 | IPv4 destination address |

| | | | |
|---|---|---|---|
| DST_MASK | 13 | 1 | The number of contiguous bits in the destination subnet mask (i.e., the mask in slash notation) |
| OUTPUT_SNMP | 14 | N | Output interface index. By default N is 2, but higher values can be used |
| IPV4_NEXT_HOP | 15 | 4 | IPv4 address of the next-hop router |
| SRC_AS | 16 | N | Source BGP autonomous system number where N could be 2 or 4. By default N is 2 |
| DST_AS | 17 | N | Destination BGP autonomous system number where N could be 2 or 4. By default N is 2 |
| BGP_IPV4_NEXT_HOP | 18 | 4 | Next-hop router's IP address in the BGP domain |
| MUL_DST_PKTS | 19 | N | IP multicast outgoing packet counter with length N x 8 bits for packets associated with the IP Flow. By default N is 4 |
| MUL_DST_BYTES | 20 | N | IP multicast outgoing Octet (byte) counter with length N x 8 bits for the number of bytes associated with the IP Flow. By default N is 4 |
| LAST_SWITCHED | 21 | 4 | sysUptime in msec at which the last packet of this Flow was switched |
| FIRST_SWITCHED | 22 | 4 | sysUptime in msec at which the first packet of this Flow was switched |
| OUT_BYTES | 23 | N | Outgoing counter with length N x 8 bits for the number of bytes associated with an IP Flow. By default N is 4 |
| OUT_PKTS | 24 | N | Outgoing counter with length N x 8 bits for the number of packets |

| | | | |
|---|---|---|---|
| | | | associated with an IP Flow. By default N is 4 |
| IPV6_SRC_ADDR | 27 | 16 | IPv6 source address |
| IPV6_DST_ADDR | 28 | 16 | IPv6 destination address |
| IPV6_SRC_MASK | 29 | 1 | Length of the IPv6 source mask in contiguous bits |
| IPV6_DST_MASK | 30 | 1 | Length of the IPv6 destination mask in contiguous bits |
| IPV6_FLOW_LABEL | 31 | 3 | IPv6 flow label as per RFC 2460 definition |
| ICMP_TYPE | 32 | 2 | Internet Control Message Protocol (ICMP) packet type; reported as ICMP Type * 256 + ICMP code |
| MUL_IGMP_TYPE | 33 | 1 | Internet Group Management Protocol (IGMP) packet type |
| SAMPLING_INTERVAL | 34 | 4 | When using sampled NetFlow, the rate at which packets are sampled; for example, a value of 100 indicates that one of every hundred packets is sampled |
| SAMPLING_ALGORITHM | 35 | 1 | For sampled NetFlow platform-wide: 0x01 deterministic sampling 0x02 random sampling Use in connection with SAMPLING_INTERVAL |
| FLOW_ACTIVE_TIMEOUT | 36 | 2 | Timeout value (in seconds) for active flow entries in the NetFlow cache |
| FLOW_INACTIVE_TIMEOUT | 37 | 2 | Timeout value (in seconds) for inactive Flow entries in the NetFlow cache |
| ENGINE_TYPE | 38 | 1 | Type of Flow switching engine (route processor, linecard, etc...) |
| ENGINE_ID | 39 | 1 | ID number of the Flow switching engine |

| | | | |
|---|---|---|---|
| TOTAL_BYTES_EXP | 40 | N | Counter with length N x 8 bits for the number of bytes exported by the Observation Domain. By default N is 4 |
| TOTAL_PKTS_EXP | 41 | N | Counter with length N x 8 bits for the number of packets exported by the Observation Domain. By default N is 4 |
| TOTAL_FLOWS_EXP | 42 | N | Counter with length N x 8 bits for the number of Flows exported by the Observation Domain. By default N is 4 |
| MPLS_TOP_LABEL_TYPE | 46 | 1 | MPLS Top Label Type:<br>0x00 UNKNOWN<br>0x01 TE-MIDPT<br>0x02 ATOM<br>0x03 VPN<br>0x04 BGP<br>0x05 LDP |
| MPLS_TOP_LABEL_IP_ADDR | 47 | 4 | Forwarding Equivalent Class corresponding to the MPLS Top Label |
| FLOW_SAMPLER_ID | 48 | 1 | Identifier shown in "show flow-sampler" |
| FLOW_SAMPLER_MODE | 49 | 1 | The type of algorithm used for sampling data:<br>0x02 random sampling<br>Use in connection with FLOW_SAMPLER_MODE |
| FLOW_SAMPLER_RANDOM_INTERVAL | 50 | 4 | Packet interval at which to sample. Use in connection with FLOW_SAMPLER_MODE |
| DST_TOS | 55 | 1 | Type of Service byte setting when exiting outgoing interface |
| SRC_MAC | 56 | 6 | Source MAC Address |
| DST_MAC | 57 | 6 | Destination MAC Address |
| SRC_VLAN | 58 | 2 | Virtual LAN identifier associated with ingress interface |
| DST_VLAN | 59 | 2 | Virtual LAN identifier associated with egress interface |

| | | | |
|---|---|---|---|
| IP_PROTOCOL_VERSION | 60 | 1 | Internet Protocol Version Set to 4 for IPv4, set to 6 for IPv6. If not present in the template, then version 4 is assumed |
| DIRECTION | 61 | 1 | Flow direction: 0 - ingress flow 1 - egress flow |
| IPV6_NEXT_HOP | 62 | 16 | IPv6 address of the next-hop router |
| BGP_IPV6_NEXT_HOP | 63 | 16 | Next-hop router in the BGP domain |
| IPV6_OPTION_HEADERS | 64 | 4 | Bit-encoded field identifying IPv6 option headers found in the flow |
| MPLS_LABEL_1 | 70 | 3 | MPLS label at position 1 in the stack |
| MPLS_LABEL_2 | 71 | 3 | MPLS label at position 2 in the stack |
| MPLS_LABEL_3 | 72 | 3 | MPLS label at position 3 in the stack |
| MPLS_LABEL_4 | 73 | 3 | MPLS label at position 4 in the stack |
| MPLS_LABEL_5 | 74 | 3 | MPLS label at position 5 in the stack |
| MPLS_LABEL_6 | 75 | 3 | MPLS label at position 6 in the stack |
| MPLS_LABEL_7 | 76 | 3 | MPLS label at position 7 in the stack |
| MPLS_LABEL_8 | 77 | 3 | MPLS label at position 8 in the stack |
| MPLS_LABEL_9 | 78 | 3 | MPLS label at position 9 in the stack |
| MPLS_LABEL_10 | 79 | 3 | MPLS label at position 10 in the stack |

The value field is a numeric identifier for the field type. The following value fields are reserved for proprietary field types: 25,
26, 43 to 45, 51 to 54, and 65 to 69.