Master Thesis

Czech
Technical
University
in Prague

F3

Faculty of Electrical Engineering
Department of Telecommunications Engineering

Monitoring System Based on Telemetry in Network CESNET3

Bc. Ladislav Loub

Supervisor: Ing. Jan Kubr, Ph.D.

June 2023

Acknowledgements

I would like to thank my supervisor Ing. Jan Kubr, Ph.D., for all the help and valuable advice while writing this thesis. Secondly, I would like to thank the CESNET company for inciting this topic and allowing me to use all the necessary resources needed for this thesis. Enormous thanks belong to my family for supporting me throughout my studies and helping me with everything that was in their capacity. Namely my mother, Jana Loubová, my father, Ing. Ladislav Loub, and my brother Mgr. Tomá² Masa°, Ph.D.. The last thanks belong to my girlfriend Tereza for her patience and support during writing this thesis.

Declaration

I hereby declare that the presented thesis is my own work and that I have cited all sources of information in accordance with the Guideline for adhering to ethical principles when elaborating an academic nal thesis.

I acknowledge that my thesis is subject to the rights and obligations stipulated by the Act No. 121/2000 Coll., the Copyright Act, as amended. In accordance with Article 46 (6) of the Act, I hereby grant a nonexclusive authorization (license) to utilize this thesis, including any and all computer programs incorporated therein or attached thereto and all corresponding documentation (hereinafter collectively referred to as the Work), to any and all persons that wish to utilize the Work. Such persons are entitled to use the Work in any way (including for-pro t purposes) that does not detract from its value. This authorization is not limited in terms of time, location and quantity. However, all persons that makes use of the above license shall be obliged to grant a license at least in the same scope as de ned above with respect to each and every work that is created (wholly or in part) based on the Work, by modifying the Work, by combining the Work with another work, by including the Work in a collection of works or by adapting the Work (including translation), and at the same time make available the source code of such work at least in a way and scope that are comparable to the way and scope in which the source code of the Work is made available.

Abstract

The thesis focuses on designing and deploying a monitoring system based on Model-Driven Telemetry technology. This technology is relatively recent. The RFC 9232, partly describing it, was released in may 2022. The thesis is the continuation of my bachelor thesis, in which I have researched Model-Driven Telemetry and its possible usage in the future.

The goal of the system is to monitor operational state data from network devices managed by CESNET. Primary information is data concerning the condition of the devices and statistics about their network interfaces.

The system consists of several parts to be able to ful ll requirements. The parts are: collectors for assembling data, a time-series database for storing them, visualization tools for displaying the data in a user-friendly way and, nally, management of the whole system. The colletors are deployed Telegraf container instances in Docker, In uxDB is used as a time-series database, Grafana and Nagios are used as visualization tools, and Ansible is used for the management of all the systems.

The result of this work is a system that serves as monitoring of the CESNET network. The thesis could serve as inspiration for other service providers that are considering deploying similar monitoring system.

Keywords: Model-Driven Telemetry, Streaming Telemetry, monitoring system, YANG, SNMP

Supervisor: Ing. Jan Kubr, Ph.D.
Department of Computer Graphics and Interaction,
Resslova 307/9,
12000 Praha 2

Abstrakt

Cílem této práce je návrh a nasazení monitorovacího systému zaloºeného na technolgii Model-Driven telemetrie. Tato technologie je pom¥rn¥ nová, kdy RFC 9232, které ji £áste£n¥ popisuje bylo vydáno teprve v kv¥tnu 2022. Tato prace navazuje na moji bakalá°skou práci, ve které jsem zkoumal telemetrii a její moºné nasazení do budoucna.

Navrºený monitorovací systém bude mít za úkol sb¥r stavových dat ze sí´ových za°ízení provozovaných sdruºením CESNET. Sbírané informace budou primárn¥ ohledn¥ stavu za°ízení samotných a jejích sí´ových rozhraní.

Systém se sestává z n¥kolika £ástí. Tyto £ásti jsou kolektory pro sb¥r dat, £asová databáze pro uloºení dat, visualiza£ní nástroje pro zobrazení statistik v p°ív¥tivé form¥ a správa celého systému. Kolektory jsou nasazeny pomocí softwaru Telegraf jako kontejnery v systému Docker. ƒasová databáze je pouºita In uxDB. Pro vizualizaci jsou pouºity nástroje Nagios a Grafana. Celá správa systému je ud¥lána pomocí nástoje Ansible.

Výsledkem práce je monitorovací systém, který slouºí pro dohled sít¥ ve sdruºení CESNET. Práce m·ºe slouºit jako inspirace pro ostatní poskytovatele sí´ových sluºeb.

Klí£ová slova: Model-Driven telemetrie, Streamovaná telemetrie, monitorovací systém, YANG, SNMP

P°eklad názvu: Monitorovací systém zaloºený na telemetrii v síti CESNET3

Contents

Figures

Tables

Chapter 1

Introduction

This thesis directly follows my bachelor thesis, "Network traffic monitoring using Model-Driven Telemetry" [16]. I have focused on a new way of monitoring network devices named Model-Driven Telemetry. The main reason for researching this technology was the problem that started to appear during the bachelor thesis. All currently used monitoring systems in CESNET are SNMP based and reaching their limits. SNMP stands for Simple Network Management Protocol. It is a protocol used to manage and monitor devices on a network. SNMP is widely used in network management systems to collect information about network devices such as routers, switches, servers, and printers [22]. One system would need a significant architecture overhaul to suffice current trends. The other is paid software, which would require considerable investment. In that order, it was decided to explore options of Model-Driven Telemetry. Based on the results, the decision was made to continue on this path and deploy a functional monitoring system.

Since finishing my bachelor's thesis, some things regarding Model-Driven Telemetry technology have changed. When I started researching it, it was only in the early draft [23], but since then, it has matured into full RFC 9232 [24]. This RFC is not for any particular technological solution but attempts to specify telemetry and all related terms and technologies. In later chapters, I will review the changes this RFC brings, where the term telemetry will be explained.

My bachelor thesis aimed to research and deploy a Model-Driven Telemetry collector. This collector was designed for testing purposes but not for production monitoring. Deployment of the collector was managed by the bash script, which installed all needed parts. Since the goal of the thesis was not any production system, the script constantly deployed an entirely new monitoring stack. The collector's purpose was to measure the differences between SNMP and Model-Driven Telemetry.

The testing offered valuable performance data on differences between SNMP and Model-Driven Telemetry. Based on the results, it was decided to continue with the MDT [1] path. Nevertheless, as with most new systems, it must be backward compatible. The restriction then must be included in the system's design. The final system must be able to collect data using SNMP and

[1] MDT (Model-Driven Telemetry)

simultaneously using MDT.

In the testing setup, the collector consisted of four parts. The data collection part was handled by Telegraf and Pipeline tools. This data was then saved into the InfluxDB database. The last part was the visualization of collected data. The Grafana software was used for this purpose. All tools were available as open-source software, so no purchases was necessary. This thesis' primary analysis task will be to evaluate the tools used in the testing setup for the new monitoring system. The biggest challenge will probably be the scalability of the collector. The collectors must be able to handle traffic from hundreds to thousands of devices in the network. Also, the monitoring system must be easy to deploy and automated as much as possible.

On the previous collector, I measured the impacts of SNMP and Model-Driven Telemetry in the old network CESNET2 with Cisco CRS platform devices. Even though these devices had relatively low RAM[2], implementing a monitoring system based on Model-Driven Telemetry was decided to be beneficial. Recently the old CESNET network is being upgraded to newer devices. These new devices have abundant memory, so MDT usage makes more sense. This technology is becoming a preferred way and gets much more updates on the manufacturer side.

The new CESNET3 network currently consists inclusively of Cisco devices. Regardless, the collector must be able to handle data from multiple vendors. To achieve this requirement, the data collecting part of the system will receive data using SNMP and Model-Driven Telemetry. Since these two technologies are quite different, data normalization will be needed. This task can be achieved by writing a script or finding another possible tool. Incoming data using SNMP and MDT are different in structure. The system must have means to unify similar data structures based on the administrator's description.

[2] One of the disadvantages of MDT compared to SNMP is its higher usage of RAM.

Chapter 2

Analysis

2.1 Current monitoring technologies

Network monitoring is one of the essential parts of keeping the network operational and providing services for customers with adequate quality. Throughout time there were many methods and technologies used for monitoring.

One of the well-known technology today is SNMP. This protocol went through three major versions, which filled in missing features or fixed its issues. But as network management moves more and more toward programmability, the SNMP can no longer provide adequate options.

The new way of monitoring is called Model-Driven Telemetry or Streaming Telemetry. Precise details and explanations of these terms will be in the next chapter. The situation progressed so far that some monitored data are no longer available using SNMP.

Another technology that monitoring systems can use is CLI[3]. The system can initiate an SSH[4] session to the network device and execute basic show commands. This way is very inconvenient for any machine processing as data received by this method can have various forms and formats. The CLI access is still used mainly by network engineers, but even this access is supposed to be very rare, and the network should be managed from the central management system rather than from the command line.

Next, technology also considered part of device monitoring can be syslog. The primary use of syslog is sending device logs, but in terms of state monitoring, it lacks in various ways. Syslog messages are limited in terms of the information they can convey. While syslog messages can contain basic information such as severity level and timestamp, they do not provide detailed context about the event that triggered the message. The messages are typically sent in a fixed format without any deeper structure, which limits their usefulness for monitoring and analysis. The last disadvantage

[3] Command Line Interface (CLI) is a method of interacting with a network device using text commands typed into a command line interface instead of a graphical user interface (GUI).

[4] Secure Shell (SSH) is a network protocol used for secure remote access to devices over unsecured networks. SSH provides a secure, encrypted communication channel between two devices.

is the scalability of syslog. As the number of network devices and syslog messages increases, it can become challenging to manage and analyze the data generated by syslog. This can result in slower response times, increased storage requirements, and other performance issues.

The current trend is to make the network manageable as much as possible programmatically. The way to achieve this is by using the YANG models. It's essential to explore these new technologies as more vendors implement them. Also, we can observe more common ground between vendors as most of them use at least OpenCon g YANG models. The more signi cant vendors, such as Cisco, Juniper, Huawei, Arista, and Alcatel-Lucent, all have YANG model usage options.

2.2 Network devices

The essential part is the identi cation of network devices from which we are going to collect data. As I've already mentioned, all the devices are network devices, such as routers and switches. The collection of data from servers is not the goal of this system. Other monitoring systems are in place in CESNET for this purpose. The majority of machines are from Cisco. Although there is mostly only one vendor, devices still need to be di erentiated into groups.

I'm going to use several groups. The CESNET3, CESNET2, DWDM and Other. The goal of this thesis is the monitoring of devices in group CESNET3, but I will be designing this system to be able to cover the rest of the groups as well.

The CESNET3 network devices group consists only of Cisco devices. This group is also considered as the IP/MPLS layer of the network[5]. All devices in this group are new and are currently being installed as the next generation of the CESNET network for customers. The core routers in this group are Cisco 8000, ASR9000, and NCS540 series. These devices are running the IOS-XR operating system with the current version 7.5.2. The second part of the network is aggregation switches, which are Cisco Catalyst 9000 series. These are running the IOS-XE system with version 17.06.03. The last part of the network is OOB [6] routers that are ISR1100 series, also running the IOS-XE operating system at version 17.06.03. The OOB routers are used to manage the CESNET3 network. The whole CESNET3 network is designed on private IP addresses, limiting access to the devices.

The CESNET2 group could be considered as older devices that are currently in use. It is also part of the IP/MPLS layer, and its design can be seen in gure 2.1. These devices are being replaced with new ones. Some of the

[5] Internet Protocol Multi-Protocol Label Switching (IP/MPLS) is a routing system that enables fast data switching from one node to the next based on labels. Commonly, the packets in IP networks need to consider the destination addresses to control the function of the routers, which takes time. IP/MPLS nds an alternative to this process by switching packets based on the label [6].

[6] Out-of-Band (OOB) refers to a separate management network that is used to control and monitor devices such as routers, switches, and servers.

old devices probably aren't going to be replaced but will be incorporated into the new network. These devices are, in some cases, limited in functionality regarding connection with this newly designed monitoring system. Some of them are not Model-Driven Telemetry capable, and SNMP remains the only option for monitoring.

Devices in this group are routers Cisco CRS series with IOS-XR 6.6.3 version. I tested Model-Driven Telemetry on some of these devices during my bachelor thesis [16]. Other devices are Alcatel-Lucent routers Nokia 7750 SR with TiMOS-C-16 operating system. These devices have only the SNMP option and are not MDT-capable. However, it doesn't matter, as all Cisco CRS and Nokia 7750 routers are being replaced. The devices in this network, which have some level of probability of staying active, are various portions of Cisco Catalyst switches. Many of these run only on the IOS operating system and are only SNMP capable.

The next group is DWDM [7]. The DWDM group consists of devices used in the optical part of the network. To better understand this layer, gure 2.2 visualize both IP/MPLS and DWDM layers. The lowest layer of gure represents physical layer. In case our case this layer is DWDM system. Above this system is then built IP/MPLS layer on gure represented by "Logical Layer (data network)". This layer represents routers and switches in network. The top layer of gure is users layer that represents services provided by whole network to customers.

The DWDM layer serves as an underlay and, in ISO/OSI [8] terms, as the network's physical layer. This part of the network is also relatively new in CESNET. It consists of Cisco NCS 2000 devices. Unfortunately, these devices don't support Model-Driven Telemetry yet. But according to Cisco presentations, this feature is being considered and probably will be released in the future. As of now, these devices support SNMP.

The last group is named Other . The probability of these devices being monitored by this system is relatively low. It consists of various laboratory switches, WiFi controllers, access points, etc. The monitoring capabilities of these devices can vary and will be considered only with the request to be monitored. The fallback solution is considered SNMP.

2.3 Telemetry term explanation

Since May 2022, RFC 9232 Network Telemetry Framework has been available, which gives more information about used terms. The purpose of this RFC is not any technological implementation but an overview of di erent parts that can be considered Network Telemetry. In recent times it's possible

[7] Dense Wavelength Division Multiplexing (DWDM) is a technology used in optical ber networks to increase the network's capacity by allowing multiple data signals to be transmitted simultaneously over a single ber optic cable [19].

[8] ISO/OSI, also known as the OSI model, is a conceptual framework for understanding computer network communication protocols. It stands for Open Systems Interconnection reference model.

Figure 2.1: CESNET2 topology (Source: [1])

Figure 2.2: DWDM and IP/MPLS layers (Source: [4])

to see many uses of the term telemetry, which are entirely valid uses. In section 2.1. Telemetry Data Coverage is speci ed: "Any information that can be extracted from networks (including the data plane, control plane, and management plane) and used to gain visibility or as a basis for actions is considered telemetry data." [24]. According to this definition, almost all data pulled from a device can be considered as Network Telemetry data.

This thesis result won't be using technologies such as Flow Telemetry [27] or Sflow Telemetry [20], which focuses on exporting flow data from network devices[9]. Another not used term is In-band Telemetry [28] as that also revolves around data flow and not state monitoring of devices.

On the other hand, valid terms used in this context are Model-Driven Telemetry, Streaming Telemetry, and Network Telemetry. Model-Driven Telemetry is based on YANG models. Streaming Telemetry describes when devices stream data to the collector without the need of a received event. Another term is Event-Driven Telemetry which is a subset of Streaming Telemetry. The difference is that in an Event-Driven scenario sends data based on events in the network, such as when one interface changes state. Streaming Telemetry transmits this data continuously [10]. Next is Open Telemetry [9]. This is almost synonymous with Model-Driven Telemetry, only that YANG models are developed by OpenConfig.

The context in which the term telemetry is used in this thesis is simply about collecting state data from network devices (such as switches and routers) using YANG data models. The term Model-Driven Telemetry (MDT) is then a type of telemetry that uses a data model to define the structure of the data that is collected and transmitted from a network device. The primary goal of Model-Driven Telemetry is to replace the SNMP. The reason for this goal is to make state monitoring of devices more automated and simplified.

Visualization of the telemetry stack can be seen in pictures 2.3 and 2.4. The first one provides a basic overview of layers of Telemetry. Raw data are modeled on the device into YANG data format. Then are exported using transport protocol to the collector based on the device's configuration.

The second figure visualizes the protocols used. This figure needs explanation since it visualizes two parts in Transport, Encoding, and Protocol sections. The NETCONF protocol, XML encoding[11], SSH, and TCP transport are part of the Model-Driven Configuration rather than Model-Driven Telemetry. It's the element of Closed-loop automation named by Cisco. The main principle is that Model-Driven Configuration should be able to change device configuration based on data received by Model-Driven Telemetry, and vice-versa MDT should be configured using Model-Driven Configuration.

[9] These technologies are also commonly referred to as Netflow data [18].

[10] Further explanation of differences between Streaming Telemetry and Event-Driven Telemetry is explained in my bachelor thesis [16, Sec. 2.3].

[11] Extensible Markup Language (XML) is a markup language and file format for storing, transmitting, and reconstructing arbitrary data. It defines a set of rules for encoding documents in a format that is both human-readable and machine-readable [30].

Figure 2.3: OSI Model of Telemetry (Source: [10])

Figure 2.4: The Model-Driven Manageability Stack (Source: [10])