

## Collecting and processing data of network devices impacting system load in terms of monitoring and warning system implementation

### Keywords

network device, data collection, data processing, statistics, system monitoring, warning system

### Abstract

*Basic information about the network monitoring process is introduced. Two monitoring methods for data collection from network devices are distinguished. Logs and metrics are described as the elements containing information about the current state of the network. A description of metropolitan networks in Poland, the solutions they apply and the specificity of the network are presented. The monitoring systems are discussed in terms of the scope of collected and processed data. The analysis of the collection and processing of network device data and the impact on its load is presented. For this purpose, the statistical data collected by Juniper MX router concerned the system load are processed. Moreover, the measurement metric used and the obtained results for the selected network device are presented. Finally, the conclusions are discussed in terms of monitoring and warning systems implementation.*

### 1. Introduction

Monitoring, in terms of IT, is an extensive branch of obtaining information on the state of computer systems and networks. Depending on whether it will be the supervision of application performance or the state of end devices, monitoring will differ in the choice of appropriate tools selection and the methods of data collection are used. Thus, in the chapter, the monitoring systems are first discussed in terms of the scope of collected and processed data. Next, using the literature presented in the bibliography, establishing close cooperation with Centre of Informatics Tricity Academic Supercomputer and networkK (CI TASK) in Gdańsk and conducting a research on network monitoring, it was possible to collect and process selected data of a particular network device. For this purpose, using the obtained statistical data, the non-parametric hypotheses of the distributions of metric representing the CPU load for the 10 tests were verified

and an exemplary evaluation of the distribution parameters for the first test was made.

The chapter is composed of Introduction, Sections 1–5 and Conclusion.

First in Section 2, the basic definitions concerned with monitoring and warning systems are introduced, two monitoring methods for data collection from network devices are distinguished and the network monitoring process steps are determined. Moreover, the warning systems are described and the data possible to be collected from network devices are presented. Next in Section 3, the basic elements containing information about the current state of the network, i.e. logs and metrics are described. In Section 4, a short description of metropolitan networks in Poland, the solutions they apply and the specificity of the network is presented. Then, in Section 5, the analysis and evaluation of the data gathered of the considered network device is performed. For this purpose, the statistical data collected by Juniper MX router on the system load are pro-

cessed. Moreover, the metrics used and the obtained results for the selected network device are presented. There are verified the non-parametric hypotheses regarding the distributions of these metrics and the statistical parameters are estimated. In conclusion, the results are discussed in terms of monitoring and warning systems implementation and the research further perspective is given.

## 2. Network monitoring and warning systems

Monitoring is the constant long-term observation and control of certain process or phenomena, with the use of a camera, video recorder or measuring devices (Markowski, 2012). We can monitor the changing climate, various species of animals, as well as locate courier parcels that we are waiting for. Companies may provide monitoring for the changing environment in order to anticipate and take advantage of potential opportunities or threats that may harm the enterprise. Due to the significant progress of digitization and the widespread access to the Internet, network monitoring is becoming more and more important for many companies, especially for those in the IT industry (Tapscott, 2008).

Network monitoring, which is a subset of network management, is the systematic checking of a computer network (Nowicki & Uhl, 2016). In the case of any problem or a break in Internet access, the role of the monitoring system is to immediately notify about the event and provide complete information to the network administrator (Dąbrowski, 2021).

The main task of the warning system is to early notice anomalies by setting appropriate notification thresholds. Monitoring of the relevant system elements allows to predict and warn before a failure occurs and to provide an appropriate time buffer for preventive actions.

### 2.1. Network monitoring process – methods

There can be distinguished two methods of network monitoring: active and passive. They differ in the way data is collected.

The basic feature of active network monitoring is the system initiating a connection in order to obtain the expected information. The administrator, using a tool, for example Simple Network Management Protocol – SNMP (Mauro & Schmidt, 2005), sends a request for specific in-

formation to the device, and then receives a response with the result. Such a procedure allows the administrator to observe and measure the features that are important to him at a given moment (Chowdhury et al., 2014). With appropriately high parameters of the device and link bandwidth as well as the appropriate frequency of requests, the generated traffic may be negligible. Active monitoring reflects the so-called *pull model*, which is characterized by the need to query the monitored device for specific data.

Passive monitoring is characterized by the elimination of the constant polling of the device for the necessary information. Instead of continuous communication between a monitoring system and the device, there can be installed one or more agents on the device, whose task is to collect and send data at a predetermined frequency to a monitoring station.

Besides the above method, a streaming solution, colloquially called telemetry, has appeared on the market, which is often a hardware solution. Both of these methods represent the so-called *push model*, which is characterized in that the data is pushed by the monitored device and received by the monitoring station. Such a solution requires fewer hardware resources on both sides, because it does not require the queries to be generated by the monitoring station and the queries processed by the monitored device (Aida et al., 2003).

### 2.2. Network monitoring process – steps

The network monitoring process consists of the following steps:

- data collection,
- data representation,
- station reporting,
- results analysing,
- data processing.

The first step is data collection by the device. Then, in the data representation step, the collected data is pre-processed and placed in a specific format.

The next step is reporting, i.e. sending the collected data by a network device to the management station. Then the measurement data is analyzed and the results are interpreted. The last step is to present the processed data, most often in a graphic or text manner.

### 2.3. Warning systems

Early Warning Systems (EWS) are frequently applied as cost-effective risk mitigation measures against natural hazards, which provide timely information on future or ongoing events to reduce loss of life and damages (UNISDR, 2014). In informatics, the principle of warning systems is to give a proper time for reaction and to support preventive activities due to early notifications. The observed anomalies occurring at the level of the system and its components are captured, a warning message is sent and therefore, irregularities/threats can be noticed in advance. This reduces risk of adverse events and permits eventual mitigation actions (Koyuncugil et al., 2010).

### 2.4. Data collected from devices

The following data can be obtained from a device using telemetry.

- Forwarding
  - interface counters,
  - filter / Policer counters,
  - ingress LSP statistics.
- Platform
  - optical power levels,
  - power consumption and temperature,
  - NPU / Line Card CPU and memory,
  - sampling process statistics.
- Routing
  - BGP peer information,
  - RSVP Protocol statistics,
  - routing process memory consumption.
- Protocols
  - LLDP state,
  - LACP state,
  - ARP / NDP state.
- QoS
  - buffer usage.

### 3. Characteristics of collected data

Event logs, commonly known as logs, are events that have taken place, and the metric is a measure of the health of the system. In many monitoring structures, the emphasis is on detecting errors, i.e. whether a specific event or system state has occurred. When administrator receives a notification of a specific system event, he can usually check any collected data to find out exactly what

happened and why. It is a good practice for the event log to have a specific structure that will tell a kind of a story about what happened.

Properly used metrics provide a dynamic real-time view of the infrastructure health that helps to manage and make good decisions.

Additionally, by detecting anomalies and analyzing patterns, e.g. through warning systems, metrics can potentially be used to identify defects or problems before they occur or before a specific system event takes place that indicates a failure (Reichert, 2018).

In addition, the logs should contain separators that will clearly separate the elements contained in the message string from each other. The first and most critical element of almost any log syntax is the *when* timestamp. The timestamp is important as it tells you exactly when an event happened on the system and was logged. Without this component, you need to rely on log analysis software to timestamp it, based on the sending date when it receives an event log. Adding a timestamp where an alert is recorded will ensure that the alert is consistently and accurately placed at the right moment when the event happened. RFC 3339 is used to define the standard Internet time and date format (Klyne & Newman, 2012). The timestamp should include the year, month, day, hour, minute, second, and time zone. To find out what happened, the log must include data such as: event priority, success or failure, status codes, resource URI or anything else that will help the administrator to identify the event accurately. The analyst should be able to retrieve a single log message or entry from a log file and know most or all of the critical information without relying on the log file name, storage location, or automatic meta-tagging from the tool.

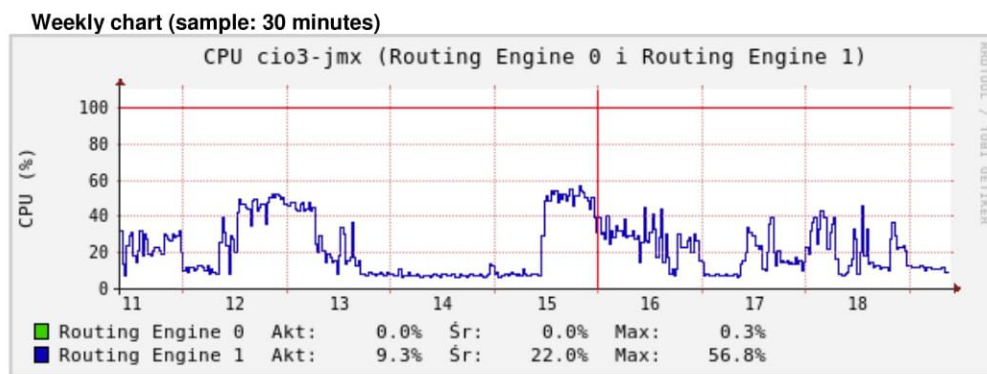
Metrics are measures of system health i.e. the properties of the software or hardware components. To be useful, data is tracked by recording data points or observations over time. An observation consists of the following elements: a value, a timestamp and sometimes a series of properties that describe it, such as source or tags. Sequences of values of observations in successive units of time are called time series. A classic example of data that we may collect as a time series is a number of website visits. We periodically collect observations about visits to a selected website, recording the number of visits and the times of observation. The collected data at



fixed intervals is called resolution. The interval can be, for example a second, five minutes, or 60 minutes. Choosing the appropriate resolution for recording the metric is critical. Details can be easily overlooked when selecting a size grain that is too coarse, e.g. long intervals are highly unlikely to identify anomalies in the data. Alternatively, choosing a high resolution may result in the need to store and interpret large amounts of data (Meghanathan & Natarajan, 2018). The time series of the values of a given metric is generally a chronologically ordered list of observations, containing the measurement data of one

metric. For example, the metric might be temperature or CPU load. Many realisations can be collected and a time series can be created from them.

Time series of metric values are often visualized as a two-dimensional graph with data values on the Y axis and time on the X axis, which is depicted in Figure 1. Often, if someone wants to present data in larger time intervals, he/she aggregates them, using certain mathematical functions and methods for this purpose. Thanks to this, large amounts of data in a smaller scope can be presented.



**Figure 1.** Example of metric plot for Routing Engine.

Providing a visual representation of critical data is relatively easier to interpret than viewing the same data as a list of values. Graphs over time also give a historical picture of what is being monitored – they show what has changed and when. Nevertheless, we can of course use both of these possibilities to understand what is happening in our environment and when it happened. The visualisation can be done with different types of charts or graphs such as indicators, counters and timers (Turnbull, 2016).

- Indicator – the first and most common type of chart is an indicator. Indicators are numbers that change over time, they are essentially a snapshot of a specific measurement. Classic metrics for CPU, memory, and temperature usage are usually presented as indicators. In the case of network monitoring, it will be the current saturation of the link on the monitored port.
- Counter – the second type of chart that might be seen frequently is the counter. Counters are numbers that increase over time and never decrease. The counters can sometimes be reset to zero and start incrementing again. Good

examples of application or infrastructure counters are system uptime, the sum of bytes sent and received by the device, or the number of logins.

- Timer – the visualization also shows a small selection of timers. They keep track of how long something has lasted. They are commonly used for application monitoring – for example, you can embed a timer at the beginning of a particular method and stop the timer at the end of the method. Each call to the method would result in the measurement of the method's execution time.

In the next section, there will be shortly presented the metropolitan networks in Poland and discussed the wide range of services provided that require monitoring.

## 4. Metropolitan networks in Poland

### 4.1. History of metropolitan networks in Poland

The year 1991 is symbolically considered the beginning of the Internet in Poland (Nakonieczny, 2019). The Academic Computer Centre –

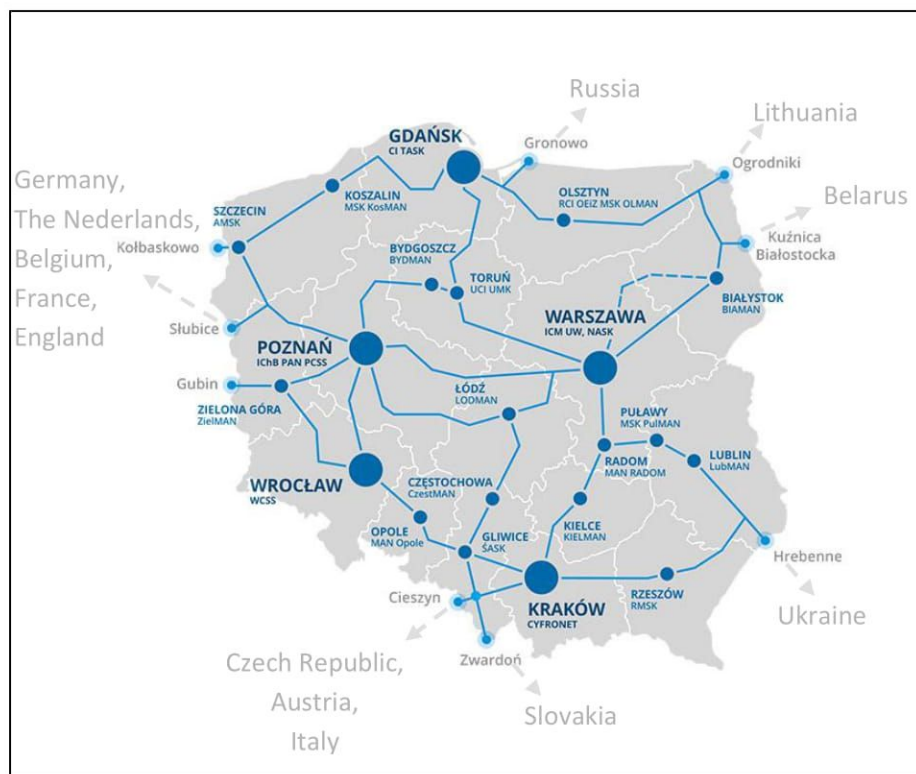
CYFRONET AGH, established in 1973, operates in Cracow.

The state-owned NASK Research Institute was established in Warsaw – the Research and Academic Computer Network. Then the Poznań Supercomputing and Networking Center in 1993, the Tri-City Academic Computer Network in 1994 and the Wrocław Centre for Networking and Supercomputing in 1995 are established.

Another milestone was the signing of the Agreement on cooperation between leading units in the use and development of the ATM 34 Mb/s domestic network on July 15, 1997. The cooperation between the units culminated in the Polish national research and education network Consortium – PIONIER, which was established on October 25, 2003 in Kazimierz Dolny on the Vistula River. It is a formal agreement concluded between the leading units of MAN (Metropolitan

Area Network) and KDM (in Polish: *komputery dużej mocy* – High Performance Computers) and includes PIONIER fiber optic network, which is one of the most modern in the world.

The aim of the Consortium was to build the PIONIER network and to develop, on the basis of this network and IT infrastructure belonging to Leading Units (members of the network), activities serving the implementation of the statutory objectives of the Leading Units and for the development of the Information Society. To meet this goal, the Consortium developed a number of rules regulating the purpose of the Polish academic computer network PIONIER, defining the ownership of the infrastructure, the rules of operation, cooperation with external non-budgetary institutions and cooperation with foreign countries (Dąbrowski, 2021).



**Figure 2.** The backbone of the nationwide PIONIER network (based on <https://pionier.net.pl>).

#### 4.2. PIONIER Network in Poland

At present, PIONIER connects 21 centers of the MANs and 5 centers of Supercomputers using its own fiber optic lines. PIONIER is the first national academic network in Europe to use its own optical fibers for data transmission. Currently, the PIONIER network uses Dense Wavelength Division Multiplexing – DWDM technology

with 400 Gbps, 100 Gbps and 10 Gbps data rates. Projects implemented within the consortium allow for the steady and continuous development of the MANs involved and access to innovative technologies. As a result, units with no budget possibilities are able to provide high-quality IT services in their academic environments.



The expansion of the network infrastructure has extended the possibilities of connecting more scientific units to the environmental scientific networks of MAN and enabled the already connected scientific units to use lines with a higher bandwidth data transmission.

Thanks to modern technologies, there has been an increase in the level of reliability of the operation of MAN municipal networks and KDM centers, and the network infrastructure has enabled research units connected to MAN municipal networks to conduct research requiring Internet access and sending information with the highest global parameters, as at that time.

Based on the tenders organized in 2017 and 2019 by PCSS for the purchase of support, it can be seen that until 2019 the main devices used in the centers are Juniper products, including MX80, MX400 and MX960.

### 4.3. Specificity of network in CI TASK

Looking at the Networks Department of CI TASK, one can notice that access to the link has been divided into commercial and educational one. The commercial link is intended for companies that prefer to be treated as partners and which are open to the client's needs. Educational links are intended for academic environments and institutes, ensuring easy access to high bandwidths directly between units. Such an extensive network requires effective monitoring of end and backbone network devices.

KDM Department – High Performance Computers Department takes care of the supercomputer TRYTON in CI TASK. It is a computing cluster which is one of the fastest computers in Europe. It is used for scientific calculations, simulations, analysis and data processing (Big Data), plus it is housed in over 40 cabinets. The computing power of the supercomputer TRYTON amounts to 1480 TFLOPS (Dąbrowski, 2021). Such a cluster requires not only monitoring of the consumption of the processor, RAM or disk occupancy, but also paying attention to the temperature of the devices, the operation of the entire cooling system, as well as the power consumption.

The last branch is cloud solutions, supervised by the Application Department of CI TASK. Here, monitoring works already at the application level, which requires a more modern approach to

the monitoring issue. These solutions have their physical endings on machines, which also require constant control and monitoring.

## 5. Analysis and evaluation of collected data

In order to get a broad view on the topic, the tests were performed in the production environment. A cooperation was established with CI TASK, located at the Gdańsk University of Technology. CI TASK has agreed to make some of its network equipment available for laboratory purposes. The device from which the data was collected was a Juniper router: MX80 Universal Routing Platform. It is one of the smallest routers of the MX type.

In order to improve the performance of tests and reduce the potential human error, an own written bash script was prepared. This script was executing the snmpbulkwalk commands for the specified OIDs (Object Identifiers).

### 5.1. Collection of network device data

A study of the CPU load (understood here as the Routing Engine) was carried out and the tests were performed. To eliminate the chances of measurement errors, the tests were carried out over a longer period of time, with each test being repeated several times. Figure 3 shows an example of a query for the Routing Engine status.

As a result, there were obtained the following data:

- device temperature,
- CPU temperature,
- RAM utilization,
- CPU utilization with average for 5 seconds, 1 minute, 5 minutes and 15 minutes, expressed as a percentage.

The impact of a large number of SNMP queries on the router load was investigated. The snmpbulkwalk method was applied using the SNMP protocol. The queries were generated using the own written software with the frequency: 10, 5, 4, 3 and 2 [seconds]. The data gathered during the research were presented in the form of metrics, i.e. numerical values for the CPU were obtained. They represent positive integers expressed as a percentage of the CPU load (Routing Engine). The base load on the CPU was 7%.

```

root@telemetry-jmx> show chassis routing-engine
Routing Engine status:
  Temperature           37 degrees C / 98 degrees F
  CPU temperature       50 degrees C / 122 degrees F
  DRAM                  2048 MB (2048 MB installed)
  Memory utilization    35 percent
  5 sec CPU utilization:
    User                25 percent
    Background          0 percent
    Kernel              25 percent
    Interrupt           1 percent
    Idle                49 percent
  1 min CPU utilization:
    User                16 percent
    Background          0 percent
    Kernel              18 percent
    Interrupt           1 percent
    Idle                65 percent
  5 min CPU utilization:
    User                7 percent
    Background          0 percent
    Kernel              11 percent
    Interrupt           1 percent
    Idle                82 percent
  15 min CPU utilization:
    User                4 percent
    Background          0 percent
    Kernel              8 percent
    Interrupt           1 percent
    Idle                87 percent
  Model                 RE-MX80
  Serial ID             S/N ZM2713
  Start time            2021-08-23 18:51:50 UTC
  Uptime                46 days, 21 hours, 25 minutes, 47 seconds
  Last reboot reason    Router rebooted after a normal shutdown.
  Load averages:       1 minute  5 minute  15 minute
                       0.31      0.20      0.16
    
```

**Figure 3.** Sample result of the query for the status of Routing Engine.

From the point of view of obtaining the lowest possible load, the values of the metrics should be as low as possible. For one resource, 121 measurement data were obtained during one test lasting 10 minutes. Such a large amount of data allowed for a very accurate estimation of the parameters of the distribution of metrics for a narrow confidence interval at the designated confidence level of 95%.

## 5.2. Estimation of parameters

Using the obtained statistical data, there can be verified the non-parametric hypotheses regarding the distributions of metric (representing the CPU load) for the examined 10 tests. To do this, a large statistical sample for each random variable is needed. A sample of at least thirty implementations in each set determined as a result of the study is considered to be such.

For the purposes of this chapter, detailed calculations for the CPU load metric measured with the Bulkwalk 5  $T_1$  test (first test with the frequency of 5 seconds) are presented, then the detailed results for Bulkwalk 5  $T_2$  (second test with the

frequency of 5 seconds) test are given and next, the average values for 10 performed tests, i.e. Bulkwalk  $i T_j$ ,  $i = 10,5,4,3,2$ ;  $j = 1,2,\dots,10$  (10 tests with the frequency of 10, 5, 4, 3 and 2 seconds), are calculated.

The sample has a sufficient number of realizations ( $n_{T_1} = 121$ ). These realizations of the random variables are marked as  $\theta_{T_1}^k$ ,  $k = 1,2,\dots,121$ . An exemplary evaluation of the distribution parameters for the Bulkwalk 5  $T_1$  test was performed below.

The realisations for the Bulkwalk 5  $T_1$  test are presented in Table 1. There can be noted that among the realizations the first value is atypical (Table 1). It will be treated as an outlier because it will prevent from carrying out a correct analysis.

In this subsection, the formulae contained in the book: *Reliability and Safety of Complex Technical Systems and Processes: Modeling – Identification – Prediction – Optimization* (Kołowrocki & Soszyńska-Budny, 2011) are used.

**Table 1.** Realisations for Bulkwalk 5  $T_1$  test

Number of realizations $n_{T_1}$	Realizations $\theta_{T_1}^k, k = 1, 2, \dots, 121$ [%]
121	7; 40; 40; 40; 39; 40; 41; 40; 40; 40; 40; 40; 40; 40; 40; 41; 40; 40; 40; 40; 40; 39; 40; 41; 40; 40; 40; 40; 40; 39; 40; 41; 40; 39; 39; 40; 40; 39; 40; 41; 40; 39; 40; 40; 40; 40; 39; 41; 40; 40; 40; 40; 40; 40; 40; 41; 40; 39; 39; 40; 40; 40; 40; 41; 40; 39; 40; 40; 40; 40; 40; 41; 40; 40; 40; 40; 40; 40; 40; 41; 40; 40; 40; 40; 40; 40; 40; 41; 40; 40; 39; 40; 40; 39; 40; 41; 40; 40; 40; 40; 40; 40; 39; 41; 40; 39; 40; 40; 40; 40; 40; 41; 40; 40; 40; 40; 41; 40; 40; 41; 40; 40.

In order to determine the realization of the average value  $M_{T_1}$ , the formula

$$M_{T_1} = \frac{1}{n} \sum_{k=1}^n \theta_{T_1}^k \quad (1)$$

should be used, receiving

$$M_{T_1} = \frac{1}{120} \sum_{k=1}^{120} \theta_{T_1}^k = \frac{4801}{120} \cong 40.00833.$$

Then, using the formula

$$\bar{r} \cong \sqrt{n}, \quad (2)$$

the number of intervals

$$\bar{r}_{T_1} \cong \sqrt{120} \cong 10.95 \approx 11,$$

can be calculated, which is needed to create a histogram.

Then there can be found the number  $\bar{R}_{T_1}$ , which is the difference between the maximum and minimum values of  $\theta_{T_1}^k$ :

$$\bar{R}_{T_1} = \max_{1 \leq k \leq 120} \theta_{T_1}^k - \min_{1 \leq k \leq 120} \theta_{T_1}^k = 41 - 39 = 2.$$

To calculate the length  $d_{T_1}$  of intervals, the following formula was used:

$$d_{T_1} = \frac{\bar{R}_{T_1}}{\bar{r}_{T_1} - 1} = \frac{2}{11 - 1} = 0.2.$$

Expressions

$$a_{T_1}^1 = \max\left\{ \min_{1 \leq k \leq n} \theta_{T_1}^k - \frac{d_{T_1}}{2}, 0 \right\}; \quad (3)$$

$$b_{T_1}^j = a_{T_1}^1 + j d_{T_1}, \quad j = 1, 2, \dots, \bar{r}_{T_1}; \quad (4)$$

$$a_{T_1}^j = b_{T_1}^{j-1}, \quad j = 2, 3, \dots, \bar{r}_{T_1}; \quad (5)$$

allow to define the ends  $a_{T_1}^m, b_{T_1}^m$  of intervals  $I_m = \langle a_{T_1}^m, b_{T_1}^m \rangle, m = 1, 2, \dots, 11$ :

$$a_{T_1}^1 = \max\left\{ \min_{1 \leq k \leq 30} \theta_{T_1}^k - \frac{d_{T_1}}{2}, 0 \right\}$$

$$= \max\left\{ 39 - \frac{0.2}{2}, 0 \right\} = 38.9,$$

$$b_{T_1}^1 = a_{T_1}^1 + 1 \cdot d_{T_1} = 38.9 + 0.2 = 39.1,$$

$$a_{T_1}^2 = b_{T_1}^1 = 39.1,$$

$$b_{T_1}^2 = a_{T_1}^1 + 2 \cdot d_{T_1} = 38.9 + 0.4 = 39.3,$$

$$a_{T_1}^3 = b_{T_1}^2 = 39.3,$$

$$b_{T_1}^3 = a_{T_1}^1 + 3 \cdot d_{T_1} = 38.9 + 0.6 = 39.5,$$

$$a_{T_1}^4 = b_{T_1}^3 = 39.5,$$

$$b_{T_1}^4 = a_{T_1}^1 + 4 \cdot d_{T_1} = 38.9 + 0.8 = 39.7,$$

$$a_{T_1}^5 = b_{T_1}^4 = 39.7,$$

$$b_{T_1}^5 = a_{T_1}^1 + 5 \cdot d_{T_1} = 38.9 + 1 = 39.9,$$

$$a_{T_1}^6 = b_{T_1}^5 = 39.9,$$

$$b_{T_1}^6 = a_{T_1}^1 + 6 \cdot d_{T_1} = 38.9 + 1.2 = 40.1,$$

$$a_{T_1}^7 = b_{T_1}^6 = 40.1,$$

$$b_{T_1}^7 = a_{T_1}^1 + 7 \cdot d_{T_1} = 38.9 + 1.4 = 40.3,$$

$$a_{T_1}^8 = b_{T_1}^7 = 40.3,$$

$$b_{T_1}^8 = a_{T_1}^1 + 8 \cdot d_{T_1} = 38.9 + 1.6 = 40.5,$$





continuous distributions, there can be formulated the following null hypothesis:

$H_0$ : the metric representing the CPU load measured by the Bulkwalk 5  $T_1$  test has a double-trapezium distribution with the density function

$$f_{T_1}(x) = \begin{cases} 0, & x < a \\ q + \left[ \frac{2 - q(b-a) - w(c-b)}{c-a} - q \right] \cdot \frac{x-a}{b-a}, & a \leq x \leq b \\ w + \left[ \frac{2 - q(b-a) - w(c-b)}{c-a} \right] \cdot \frac{c-x}{c-b}, & b \leq x \leq c \\ 0, & x > c. \end{cases} \quad (6)$$

where

$$0 \leq a \leq b \leq c < \infty, \quad 0 \leq q < \infty, \quad 0 \leq w < \infty, \\ 0 \leq q(b-a) + w(c-b) \leq 2,$$

that is, with the distribution function

$$F_{T_1}(x) = \begin{cases} 0, & x < a \\ qx + \left[ \frac{2 - q(b-2a+c) - w(c-b)}{c-a} \right] \cdot \frac{(x-a)^2}{b-a} - aq, & a \leq x \leq b \\ wx + \left[ \frac{2 - q(b-a) - w(2c-a-b)}{c-a} \right] \cdot \frac{(b-x)(x+b-2c)}{c-b} - bw, & b \leq x \leq c \\ 1, & x > c, \end{cases} \quad (7)$$

As the parameters of this distribution are unknown, they were estimated on the basis of the sample, using the formulae contained in the book (Kołowrocki & Soszyńska-Budny, 2011).

Finally, the null hypothesis takes the form:

$H_0$ : the metric representing the CPU load measured by the Bulkwalk 5  $T_1$  test has a double-trapezium distribution:

$$F_{T_1}(x) = \begin{cases} 0, & x < 39 \\ -0.2210834x^2 + 17.89911x - 361.7368, & 39 \leq x \leq 40 \\ 0.2505095x^2 - 19.83615x - 393.125, & 40 \leq x \leq 41 \\ 1, & x > 41, \end{cases} \quad (8)$$

illustrated in Figure 5.

Then, some subsets of  $\bar{I}_m = \langle a_{T_1}^m, b_{T_1}^m \rangle$  were combined so that the condition concerning the minimum sizes of these subsets, i.e.  $\bar{n}_{T_1}^m \geq 5$  was met. After this operation, the number of these subsets is reduced from

$$\bar{r}_{T_1} = 11$$

to

$$\bar{r}_{T_1} = 3.$$

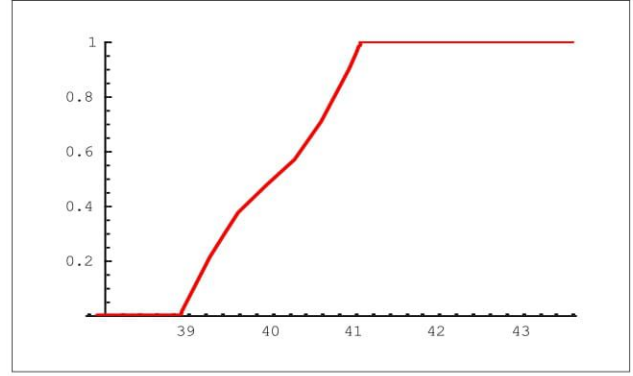


Figure 5. Distribution function  $F_{T_1}(x)$ .

Realization

$$u_{T_1} = u_{T_1}(\theta_{T_1}^1, \theta_{T_1}^2, \dots, \theta_{T_1}^k)$$

of the statistic  $U_{T_1}$ , which is a measure of the divergence between the empirical and hypothetical distribution, is calculated using Table 4.

Table 4. Numbers of realizations of metric (representing CPU load) in combined intervals

Intervals $\bar{I}_m = \langle a_{T_1}^m, b_{T_1}^m \rangle$	Number of realizations $\bar{n}_{T_1}^m$
$\langle 0; 39.1 \rangle$	89
$\langle 39.1; 40.9 \rangle$	15
$\langle 40.9; \infty \rangle$	16

The probabilities that the metric representing the CPU load takes values within the range given in Table 4 are calculated as follows:

$$p_1 = F_{T_1}(39.1) - F_{T_1}(0) = 0.124 - 0 = 0.124,$$

$$p_2 = F_{T_1}(40.9) - F_{T_1}(39.1) = 0.881 - 0.124 = 0.757,$$

$$p_3 = F_{T_1}(\infty) - F_{T_1}(40.9) = 1 - 0.881 = 0.119.$$

The implementation of Pearson's statistics  $\chi^2$  (chi-square) is calculated using Table 5:

$$u_{T_1} = \sum_{m=1}^3 \frac{(\bar{n}_{T_1}^m - n_{T_1} p_m)^2}{n_{T_1} p_m} \cong 0.245.$$

**Table 5.** Calculation of statistics  $U_{T_1}$

$m$	1	2	3
$\bar{I}_m$ $= \langle a_{T_1}^m, b_{T_1}^m \rangle$	(0; 39.1)	(39.1; 40.9)	(40.9; $\infty$ )
$n_m$	15	89	16
$p_m$	0.124	0.757	0.119
$np_m$	14.88	90.84	14.28
$n_m - np_m$	0.12	-1.84	-1.72
$(n_m - np_m)^2$	0.0144	3.3856	2,9584
$\frac{(n_m - np_m)^2}{np_m}$	0.000968	0.037270	0.207171
$u_n$		0.245	

The significance level of the test was assumed  $\alpha = 0.05$ , the number of degrees of freedom was determined

$$\bar{r}_{T_1} - l - 1 = 3 - 0 - 1 = 2.$$

From the distribution  $\chi^2$  –Pearson's tables the value  $u_\alpha$  was read for the previously assumed significance level  $\alpha = 0.05$  and degrees of freedom  $\bar{r}_{T_1} - l - 1 = 2$ , such that the equality

$$P(U_{T_1} > u_\alpha) = \alpha = 0.05,$$

is satisfied and

$$u_\alpha = 5.99.$$

Then the critical interval was determined as

$$(5.99, +\infty).$$

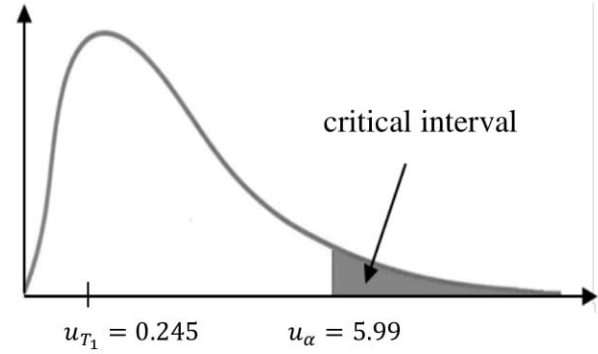
The realisation

$$u_{T_1} = 0.245$$

of the statistics  $U_{T_1}$  was compared with the value  $u_\alpha = 5.99$ . Thus,  $u_{T_1} = 0.245$  does not belong to a critical interval (Figure 6), i.e.

$$u_{T_1} = 0.245 \leq u_\alpha = 5.99.$$

Therefore, there is no reason to reject the hypothesis  $H_0$  for the considered critical level.



**Figure 6.** Interpretation of critical interval and acceptance interval for  $\chi^2$  test.

When the distribution has been identified, the main characteristics can be determined, i.e.:

- expected value:

$$M = \int_0^\infty (1 - F(x))dx \approx 40.00465 \approx 40.005,$$

- second moment

$$M2 = 2 \int_0^\infty x(1 - F(x))dx \approx 1606.736,$$

- variance

$$Var = M2 - M^2 \approx 6.363824,$$

- standard deviation

$$D = \sqrt{Var} \approx 2.522662 \approx 2.523.$$

For the  $T_2$  data set, the distribution and the numerical characteristics were analogically determined for the significance level  $\alpha = 0.05$ , i.e. there was assumed the confidence interval equal to

$$1 - \alpha = 95\%.$$

The mean values and standard deviations of the random variables were obtained and are presented in Table 6 for the two selected CPU load tests.

The average CPU load values for tests  $T_1$  and  $T_2$  were 39.986%.



**Table 6.** Characteristics of random variables for two selected CPU load tests

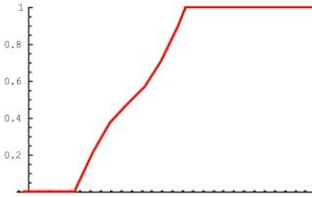
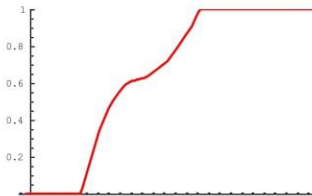
Test	Mean [%]	Standard deviation [%]	Distribution function graph
Bulkwalk 5 $T_1$	40.005	2.523	
Bulkwalk 5 $T_2$	39.787	4.915	

Table 7 presents the average values of 10 performed tests for the CPU load.

**Table 7.** Average values of 10 tests for CPU load

Test	Bulkwalk 5
10 s	23.55%
5 s	39.99%
4 s	47.62%
3 s	61.89%
2 s	89.25%

For the proper work of warning system and monitoring, data must be collected at the appropriate frequency (every 1 or 5 minutes is too rarely). The above results show that too high frequency causes too much load for the monitored device, which also directly affects the reliability of the network. Telemetry may solve the problem of excessive load by changing the data collection model and automating the processes.

## 6. Conclusion

The chapter presents issues related to the subject of monitoring and warning system implementation. This subject was discussed on the basis of the example of MAN Units – Academic IT Centers. Using the literature presented in the bibliography, establishing close cooperation with CI TASK and conducting research on network monitoring, it was possible to draw the following

conclusions. Based on the CI TASK unit, it can be noted that the multitude of services provided for the administrators of specific departments requires diverse approach to the issue of monitoring. The network department mainly focuses on monitoring the input/output data of the link and the load on end devices. Moreover, the research revealed that analyzing data based on a graph can be supported by data analysis tools. Nowadays, the real-time data reporting techniques for sampling CPU or memory usage replaces the old reporting (every 5 minutes or 1 minute).

The study allowed to state that despite the technical possibilities of frequent SNMP polling by bulkwalk method, it significantly influences the load of the monitored device and therefore the reliability of the network as well. The author's further research is concerned with analysing more data obtained from a device (listed in Section 2.4) as well as analysing and comparing telemetry solutions offered by other companies such as Cisco or Arista.

## References

- Aida, M., Miyoshi, N. & Isnibashi, K. 2003. A scalable and lightweight QoS monitoring technique combining passive and active approaches. *INFOCOM 2003. Twenty-Second Annual Joint Conference of the IEEE Computer and Communications. IEEE Societies*, Volume 1, IEEE, 125–133.
- Chowdhury, S.R., Bari, M.F., Ahmed, R. & Boutaba, R. 2014. Payless: a low cost network monitoring framework for software defined networks. *2014 IEEE Network Operations and Management Symposium (NOMS)*, IEEE, 1–9.
- Dąbrowski, M. 2021. *Analysis and Comparison of Data Collection and Processing Tools for Network Devices*. MSc Thesis. Faculty of Electronics, Telecommunications and Informatics. Gdańsk University of Technology (in Polish).
- Klyne, G. & Newman, C. 2021. Date and Time on the Internet: Timestamps, The Internet Society, 2012, <https://datatracker.ietf.org/doc/html/rfc3339> (accessed 20 Aug 2021).
- Kołowrocki, K. & Soszyńska-Budny, J. 2011. *Reliability and Safety of Complex Technical Systems and Processes: Modeling – Identification – Prediction – Optimization*, Springer, 55–61, 175–182.

- Konsorcjum PIONIER. 2021. [www.pionier.net.pl/online/pl/jednostki/1](http://www.pionier.net.pl/online/pl/jednostki/1) (accessed 20 Apr 2021).
- Markowski, A. 2012. *Kultura języka polskiego. Teoria. Zagadnienia leksykalne*. PWN (in Polish).
- Mauro, D., Mauro, D.R. & Schmidt K. 2005. *Essential SNMP*. O'Reilly Media, 19–37.
- Meghanathan, N. 2018. *Centrality Metrics for Complex Network Analysis: Emerging Research and Opportunities*. IGI Global, 1–34.
- Nakonieczny, M. 2019. *Task Story 1990–2015 (Historia Trójmiejskiej Akademickiej Sieci Komputerowej pisana Internetem)*, TASK Publishing Gdańsk, 3–13.
- Network Monitoring. 2021. <https://avinetworks.com/glossary/network-monitoring>, (accessed 27 Mar 2021).
- Nowicki, K. & Uhl, T. 2016. *Monitorowanie i bezpieczeństwo sieci komputerowych*. Wydawnictwo Naukowe Akademii Morskiej w Szczecinie, 1–11.
- Reichert, D. 2021. *Logs and Metrics: What Are They, and How Do They Help Me*. [www.sumologic.com/blog/logs-metrics-overview](http://www.sumologic.com/blog/logs-metrics-overview) (accessed 10 Apr 2021).
- Tapscott, D. 2008. *Grown Up Digital: How the Net Generation is Changing Your World*. McGraw Hill Professional, 9–39.
- Turnbull, J. 2016. *The Art of Monitoring*, O'Reilly, 21–54.
- UNISDR – The United Nations Office for Disaster Risk Reduction. *Terminology*. Geneva, [www.unisdr.org/we/inform/terminology](http://www.unisdr.org/we/inform/terminology) (accessed 10 Feb 2022).