

SELECTING NETWORK MONITORING SYSTEM SOFTWARE WITH ANALYTICAL HIERARCHY PROCESS METHOD

Ahmad Sulhi¹, Abdul Muis Sobri², Cecep Nurul Alam³

¹Information Technology Department, Faculty of Science and Technology
State Islamic University Syarif Hidayatullah Jakarta

²Informatics Engineering Department, Faculty of Information Technology
Budi Luhur University

³Informatics Engineering Department, Faculty of Science and Technology
State Islamic University Sunan Gunung Jati Bandung

¹sulhi@uinjkt.ac.id, ²abdul.muis@budiluhur.ac.id, ³cecep@uinsgd.ac.id

ABSTRAK

Saat ini banyak sekali perangkat lunak Network Monitoring System (NMS) yang handal serta mudah dalam penggunaannya, misalnya CiscoWork, HP-OpenView, dan IBM-Tivoli. Hanya saja *software* tersebut cukup mahal, karena memang tergolong produk berbayar. Untungnya ketersediaan produk NMS tidak sebatas berbayar, tetapi juga banyak alternatif lain yaitu produk-produk yang tergolong *Free and Open Source Software* (FOSS). Sayangnya, kebanyakan produk FOSS tersebut tidak saja susah dalam implementasi, baik dari segi instalasi maupun konfigurasi, tetapi juga keterbatasan dalam jumlah *node* serta jenis yang dimonitor, meliputi perangkat jaringan, *server*, dan aplikasi. Metode pemilihan perangkat lunak NMS berbasis FOSS ini didasarkan pada metode penelitian Oyku Alanbay dan ISO 7498-4 melalui pendekatan *Analytical Hierarchy Process* (AHP) dengan menggunakan *Expert Choice tool*. Di antara GroundWork, Hyperic, Nagios, OpenNMS, dan Zenoss, berdasarkan hasil penelitian ini menempatkan Nagios sebagai perangkat lunak yang cukup *reliable* untuk *monitoring* jaringan, *server*, dan aplikasi.

Kata Kunci: *NMS, FOSS, Oyku Alanbay, ISO 7498-4, AHP*

ABSTRACT

Nowadays there are lots of Network Monitoring System (NMS) software that are reliable and easy to use, such as CiscoWork, HP-OpenView, and IBM-Tivoli. It is just that the software is quite expensive because it has classified as a commercial product. Fortunately, the availability of NMS products is not limited to commercials, but also many other alternatives, namely products that classified as Free and Open Source Software (FOSS). Unfortunately, most of these FOSS products are not only difficult to implement, both in terms of installation and configuration, but also limitations in the number of nodes and types of monitored, including network devices, servers, and applications. The method of selecting FOSS-based NMS software based on the Oyku Alanbay and ISO 7498-4 research methods through the Analytical Hierarchy Process (AHP) approach using the Expert Choice tool. Among GroundWork, Hyperic, Nagios, OpenNMS, and Zenoss, based on the results of this study put Nagios as a fair reliable software for monitoring networks, servers, and applications.

Keywords: *NMS, FOSS, Oyku Alanbay, ISO 7498-4, AHP*

I. INTRODUCTION

In today's era of global competition, computer networks are the backbone of information systems, which can measure whether a company is competitive or not. The vital of this computer network, so the failure will result in loss of productivity and financial losses. Therefore, companies that want to remain competitive want a computer network infrastructure that is reliable, and can work optimally with the lowest possible failure rate.

Computer networks are complex. The complexity is not only limited to the number of devices but also the variety of systems and technologies that are in it, as well as the wide coverage of network areas including Local Area Network (LAN), Wireless Local Area Network (WLAN), Wide Area Network (WAN), Metropolitan Area Network (MAN) and Campus Area Network (CAN) [1].

Even reliable infrastructure cannot be used as a guarantee that the network function can work optimally without the support of good management. Ourselves will never be able to observe with the naked eye and then report what happens just like that, especially when there are obstacles in the network. All these needs can be met with a program that calls itself an NMS.

NMS is a software whose main function is monitoring and managing network devices with the aim that reliability and availability are maintained and can be an anomaly or can be terminated early on the network. Current there are lots of software that is reliable and easy to use, such as, CiscoWork, HP Open View, and IBM Tivoli NetView. But the software is quite expensive because it is classified as a commercial product. Fortunately, the availability of NMS products is not limited to commercial (propriety), but also many other alternatives, namely open-source products [2].

Although there are many choices, it's not necessarily no problem. New problems will arise when we want to choose which products are reliable and by the needs of our network infrastructure. Reliability here is at least the same as licensed products (commercial). Because there is no means if the product is is free but not reliable. If this happens, in the end the cost must be greater than the price of the commercial product itself.

The next problem, most open source products are not only difficult to carry out, both

in terms of installation and configuration, but also limitations in the number of nodes and types that are monitored, including monitoring network devices, servers, and applications. This choice is more complex when the software is to use on a large network (enterprise network).

Network Monitoring System (NMS is software whose main function is to watch network devices with the aim that the reliability and availability are maintained and can be an anomaly or can be terminated early on the network. A monitoring system is a process for collecting data from various resources. Usually, the data collected in real-time data [4]. Broadly speaking, the stages in a monitoring system are divide into three major processes as shown in the following pictures:

- Process in collecting monitoring data
- Process in monitoring data analysis
- The process of displaying monitoring data

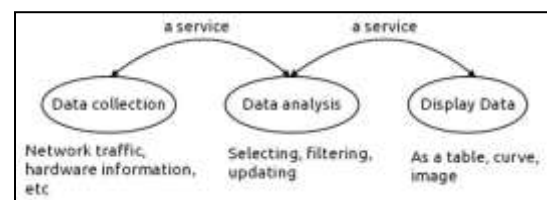


Figure 1. Process in a monitoring system

Action that occurs between the processes in a monitoring system is in the form of service, which is a process that is constantly running at a certain time interval. The processes that occur in a monitoring system starts by collecting data such as data from network traffic, hardware information, etc. Then the data is analyzed in the data analysis process and finally, the data will be display.

The first study to select NMS software as software to be review in this study based on research conducted by Jack Hughes of Tech Teapot. On November 2, 2007, Jack Hughes posted the results of his research in the form of statistics based on how many users have downloaded open source-based NMS software from sourceforge.net [5].

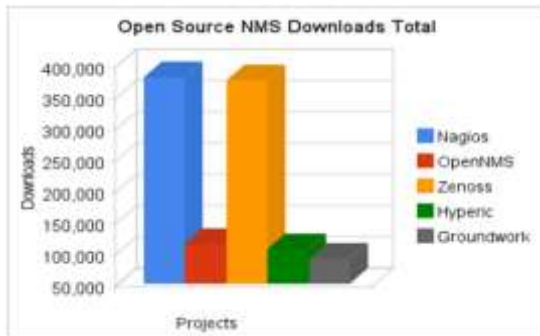


Figure 2. Open source NMS downloads total

From these statistics obtained 5 (five) of the most downloaded software, namely Nagios 379,807, OpenNMS 113,652, Zenoss 374,719, Hyperic 104,616, and GroundWork 89,903.

ISO/IEC 7498-4 is a framework for network management, released in 1989. The Network Management model specified in this ISO covers 5 (five) conceptual areas, namely Fault Management, Configuration Management, Accounting Management, Performance Management, and Security Management, which is abbreviated with the acronym FCAPS. [6]

Fault management means the ability to detect mistakes. This application will watch the device by doing pooling periodically and notifying when the device inaccessible or error.

The main purpose of configuration management is to watch network and system configuration information so that all hardware, software, and configuration versions can be tracked, and all potential problems can be eliminated (or anticipated). Here it is very important to have a standard configuration, so that with just a glance at the network staff will be able to know exactly where this device located, what type of device, what its role is, and so on.

Accounting management is the process of measuring network use so that network users can set up for accounting or billing purposes. Like performance management, the first step is to measure the use of network devices. The measurement results can be used to get network usage patterns for SLA (Service Level Agreement) guarantees.

Performance management is closely related to SLA (Service Level Agreement), which is a written agreement between the service provider and its customers about the promised network performance level. The SLA consists of several parameters agreed by both parties. Values in the SLA must realistic,

meaningful, and measurable by both parties. All measurements entered into the SLA, such as packet drops, latency, bandwidth, and so on.

The main purpose of security management is to regulate access to network resources in accordance with agreed rules so that the network cannot be intentionally or unintentionally intruded. A security management subsystem, such as, can watch all users who enter the network, and reject users who do not have the correct access codes [7].

GroundWork Open Source (GWOS) is one of the NMS software that is quite widely used in monitoring network devices, servers, and applications. In accordance with its mottoes The Open Enterprise Monitoring Solution This software classified as NMS for enterprise-class network monitoring. [8]

GWOS Community Edition provides essential features needed to watch networks, servers, and applications, with the following features:

- Easy Installation Wizard Wizard for easy installation.
- Point-and-Click Configure the node with a Web-based interface
- Portal-Based Interface can customize according to the needs of the portal.
- Availability Monitoring Provides alerts when a device is functioning or not.
- Heterogeneous Monitoring can monitor various operating systems, servers, devices, and applications.
- Performance Monitoring Provides information about the performance of the device monitored.
- Auto-Discovery and Web-Based Configuration Tools Automatic identify devices.
- Open Architecture Open architecture, so that it can develop further without any investment costs.

Hyperic HQ is one of the most complete servers, applications, and network resource monitoring and management tools. Equipped with more than 70 plugins that are ready to use according to your needs. Hyperic HQ built based on the Java platform, so it can be used on almost all operating systems including Microsoft Windows, Linux, Sun Solaris, and Mac OS. [9]

Features included in the open-source edition include:

- a. Auto-Discovery
- b. Monitor Metrics
- c. Track Performance, Configuration, and Security Changes
- d. Historical Charting and Graphing Performance
- e. Customizable Dashboard
- f. JMX MBean Query
- g. Availability Alerting
- h. Event Alerting
- i. Escalation Schemes
- j. User-Initiated Control Actions
- k. Nagios Integration
- l. Plugin Development Kit for Resource Plugins
- m. HQU Framework for UI, Automation, and Integration Plugins
- n. Web Services API
- o. Support Services (Documentation and Community).

Nagios is an application for monitoring networks, hosts, and services contained on a network. This application can tell users before there is a network problem in the client or end-user. Although this application is design to run on Linux operating systems, in general, this application can run on most * NIX variants equally well [10].

Nagios already has several features, which makes this monitoring application quite powerful. Some features of Nagios include:

- a. It can be used to watch network services (SMTP, POP3, HTTP, NNTP, PING).
- b. It can be used to watch the use of host resources (processor load, memory, and hard disk usage, running processes, log files, and so on).
- c. It has a simple plugin design, which allows users to develop easily their host and service inspection methods.
- d. It has the ability to define a network host hierarchy, which allows detection and distribution between host computers that are down, and those that are not down.
- e. It has the ability to define handling events that will run, before a problem occurs on the host computer.
- f. Having the ability to recognize problems through a web-based display.

The support found in the Nagios project is also very good. Various support services ranging from documentation, FAQs, mailing lists, and forums, have been providing properly by the Nagios.

Like Hyperic HQ, OpenNMS based on a java platform (java-based platform). Other similarities, they both released two editions, open-source (free) and enterprise (commercial). Although service area coverage covers network infrastructure, servers, and applications, the difference is that OpenNMS is more focused on monitoring network devices, while Hyperic HQ focuses primarily on application monitoring (software).

The features of OpenNMS include:

- a. Service polling - Information service on data availability and latency and measurement.
- b. Data collection - Collection, storage, and reporting of data derived from nodes that are monitor through the protocol, including SNMP, JMX, HTTP, WMI (Windows Management Instrumentation).
- c. Thresholding - Evaluating polled latency data or collected performance data against configurable thresholds, creating events when these are exceeded or rearmed (Evaluate)
- d. Event management - Setting events both internal and external events, including SNMP traps
- e. Alarms and automation - Equipped with alarms with centralized automatic control.
- f. Notifications - Sending messages based on events via email, SMS, and other media [11].

Zenoss (Zenoss Core) is a network application and monitoring system for the enterprise-class, which can simplify the work of network administrators and system administrators, who can effectively manage the status and performance of the network contained in an infrastructure. This application also can automatically search, agent-less data collection, reporting, and real-time information access through AJAX-based web portals.

The first Zenoss project made by Erik Dahl and released as an open-source project at SourceForge, in March 2006. Because of the growing popularity of the application, this project acquired by Zenoss Inc., which is also a member of the Open Management Consortium.

As an Enterprise Monitoring application, Zenoss has all the technologies found in monitoring software. This application made based on Python-Zope Application Server. Zenoss has all the advantages found in other open-source application technologies, such as

Net-SNMP, data stored in a MySQL database, and log data retrieved using RRDtool. Zenoss can collect data via SNMP, or if a Windows server uses Windows Management Instrumentation (WMI) [12].

The purpose of this study was to choose a reliable FOSS-based NMS software, by determining selection criteria and sub-criteria based on the combining of Oyku Alanbay theory [3] with ISO 7498-4. The objective was to select a reliable FOSS-based NMS software to be used as a computer network monitoring tool. The following two research questions were then proposed for guiding the implementation of the study.

RQ1. : Does the software selection method based on Oyku Alanbay Theory and ISO 7498-4 will produce reliable NMS software?

RQ2. : Which NMS software is more reliable based on the research method above?

This paper divided into five sections. The first one elucidates the research programs of the study, then followed by the literature review, research method, results and discussions, and the conclusion parts in the last section.

II. RESEARCH METHOD

2.1 Conceptual Framework

The thinking framework in determining selection strategy of FOSS-based NMS software that is reliable and can be implemented properly as a monitoring tool for networks, servers, and computer applications as illustrated in the mindset framework as follows:

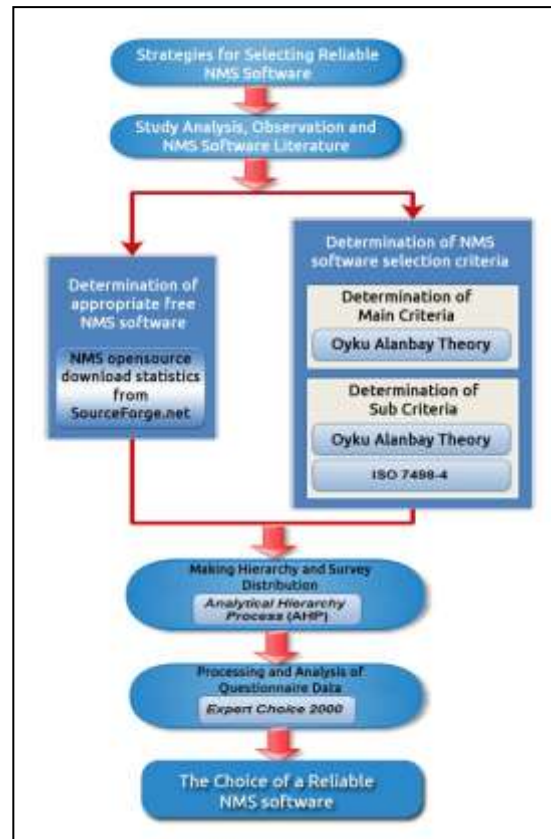


Figure 3. Mindset strategy for FOSS-based NMS software selection strategy

From the picture above it can see that choice strategy of a reliable FOSS-based NMS software determined by criteria based on the Alanbay Research Method and ISO 7498-4, where the criteria are prepared using the AHP approach method and then questionnaire/questionnaire data from respondents experts obtained and analyzed with Expert Choice 2000. While the determination of alternative software to studied was obtained based on the statistics of Open Source NMS downloads from the sourceforge.net site.

2.2 Research Hypothesis

Based on observations made by the author of network monitoring and control software based on free open source, it obtained a hypothesis in this study entitled Method Selection of FOSS-based NMS Software (Free Open Source Software) with analytical Hierarchy Process (AHP) Approach: Study Comparison of GroundWork, Hyperic, Nagios, OpenNMS, and Zenoss. Follows are:

H0: It is a suspect method and evaluation on the selection of reliable FOSS-based NMS software for use in monitoring networks, servers, and applications based on the Oyku

Alanbay research method and change and integration with ISO 7498-4.

H1: It is suspect FOSS - based NMS software that is reliable and complies with the criteria and sub-criteria for use in monitoring networks, servers and applications are Nagios.

2.3 Types of Research

Judging from the type of information that managed, this type of research is a Quantitative Study where the researcher tests the hypothesis with statistical techniques. The statistical data obtained from the questionnaire using the analytical Hierarchy Process (AHP) approach and then tested using the Expert Choice 2000 tool or software.

Meanwhile, if viewed from the purpose of this study, this type of research is Descriptive Research in which this research intends to offer a description or description of a situation as clearly as possible without any treatment of the object to examine.

2.4 Sample Selection Method

In selecting samples, the authors take data from a limited population (limit population) by using purposive sampling, which is taking samples based on certain considerations.

Respondents take a selection of this sample are expert respondents namely network administrators from universities and Internet Service Provider (ISP) companies and academics in the field of networking.

Consideration of sample choice based on the categorization or background of the respondents above, among others, so that evaluation of FOSS-based NMS software is more varied and aim. Besides that, expert respondents in this case are network administrators from colleges/universities, ISPs, and academics in the field of networking because these institutions/companies usually carry out open source products in carrying out their business operations.

The following is a complete list of expert respondents:

Table 1. Table of respondents.

No.	Respondents	Total
1	Admin Network Jakarta State Islamic University	1 Person
2	Admin Network Mercu Buana University	1 Person
3	Admin Network of Personnel Means of Telematics (ISP)	1 Person
4	Admin Network Cyber Network Indonesia (ISP)	1 Person
5	Terrapro Computer Center (Practitioner) Network Admin	1 Person
6.	CNAP Instructors (Cisco Networking Academic Program)	1 Person
Total		6 Persons

2.5 Instrumentation

The main instrumentation used in this study was a questionnaire. The questionnaire arranged in the form of questions by referring to the hierarchy made from the criteria and sub-criteria based on the Saaty scale of 1 9 using the Pairwise Comparison method.

In priority determining of strategic choice step in determining the FOSS-based NMS software, 20 (twenty) sub-criteria proposed which grouped into 3 (three) main criteria.

The preparation and grouping of these main criteria based on the hierarchy compiled by Alanbay. The sub-criteria modified and adjusted to the subject of this study, where Alanbay's research subject is ERP software that is commercial while the research subject that the author studies is FOSS-based NMS software.

The next change is the addition of sub-criteria based on ISO 7498-4 by adding FCAPS (Fault, Configuration, Accountancy, Performance, and Security) elements.

Following the AHP hierarchy in FOSS-based NMS software selection strategy:

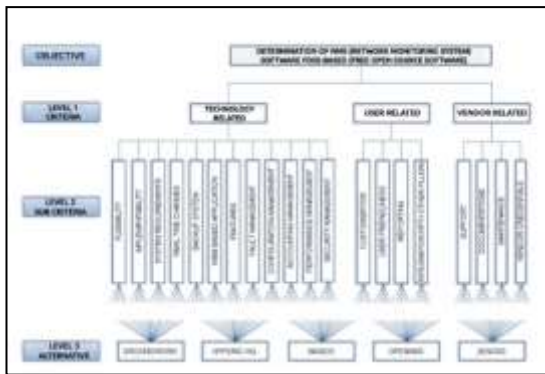


Figure 4. Hierarchy strategy of FOSS-based NMS selection

III. RESULT AND DISCUSSION

Respondents in the FOSS-based NMS software selection research were all expert respondents totaling 6 (six) respondents. The expert respondent's understanding, in this case, is that all respondents really understand the object under study and have implemented the software in their work.

The six expert respondents each represented 3 (three) groups/categories, namely the network administrator group at the university, ISP (Internet Service Provider), and practitioners/academics. Each group represented by 2 (two) respondents.

The responses of expert respondents for the questionnaire be seen in the results of the merger of respondents as follows:

Compare the relative importance with respect to: Goal: Pemilihan Software NMS Berbasis FOSS			
	Technology Related	User Related	Vendor Related
Technology Related		2,41827	3,89653
User Related			3,99248
Vendor Related	Incon: 0,09		

Figure 5. The results of combination of respondents to the criteria

Compare the relative importance with respect to: Technology Related											
Priority	Flexibility	Scalability	System Speed	Time	Storage	Up Time	Features	Flexibility	Integration	Performance	Security
Scalability	1,00000	2,21881	1,11716	2,76152	2,00000	2,21754	2,21754	2,40000	2,21754	2,42100	2,14000
System Speed	0,45110	1,00000	1,00000	1,00000	1,00000	1,00000	1,00000	1,00000	1,00000	1,00000	1,00000
Time	0,91020	1,00000	1,00000	1,00000	1,00000	1,00000	1,00000	1,00000	1,00000	1,00000	1,00000
Storage	0,45110	1,00000	1,00000	1,00000	1,00000	1,00000	1,00000	1,00000	1,00000	1,00000	1,00000
Up Time	0,91020	1,00000	1,00000	1,00000	1,00000	1,00000	1,00000	1,00000	1,00000	1,00000	1,00000
Features	0,45110	1,00000	1,00000	1,00000	1,00000	1,00000	1,00000	1,00000	1,00000	1,00000	1,00000
Flexibility	0,45110	1,00000	1,00000	1,00000	1,00000	1,00000	1,00000	1,00000	1,00000	1,00000	1,00000
Integration	0,45110	1,00000	1,00000	1,00000	1,00000	1,00000	1,00000	1,00000	1,00000	1,00000	1,00000
Performance	0,45110	1,00000	1,00000	1,00000	1,00000	1,00000	1,00000	1,00000	1,00000	1,00000	1,00000
Security	0,45110	1,00000	1,00000	1,00000	1,00000	1,00000	1,00000	1,00000	1,00000	1,00000	1,00000

Figure 6. The results of the merger of respondents to the technology relate criteria

Compare the relative importance with respect to: User Related			
	Customizable	User Friendly	Reporting
Customizable		2,36671	1,81717
User Friendly			2,27000
Reporting			1,65911
Incon: 0,01			

Figure 7. The results of the merger of respondents to the user related criteria

Compare the relative importance with respect to: Vendor Related			
	Community Support	Documentation	Maintenance
Community Support		1,66111	1,25911
Documentation			1,76211
Maintenance			1,61801
Incon: 0,01			

Figure 8. The results of the merger of respondents with vendor related criteria

The geometric calculation of the merger of the respondent's data obtained the value of alternative weights as presented in the following graph:

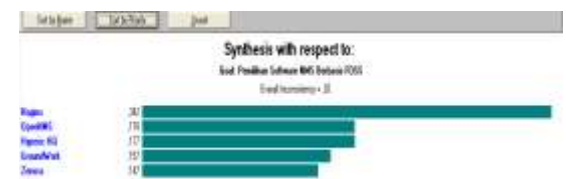


Figure 9. Global priority weight value of alternative priorities based on goals determining FOSS-based NMS software

Figure 9 can be seen that the top or highest priority alternative for determining the FOSS-Based NMS Software is Nagios with a weight value of 0.342 or equivalent to 34.2% of the total alternatives set. The results of these alternative weight values are following the hypotheses made by formulation of the problem in the previous chapter. Then the next alternative priority ranking is OpenNMS (weight value 17.8%), Hyperic HQ (weight value 17.7%), GroundWork (weight value 15.7%), and the lowest priority ranking is Zenoss (weight value 14.7%).

3.1 Inconsistency Ratio (CR)

Inconsistency ratio or respondent data inconsistency ratio is a parameter use to check whether a pairwise comparison has done consequent or not. The data inconsistency ratio considered good if the CR value is ≤ 0.1 .

The following shows the inconsistency ratio values for each comparison matrix.

<i>Table 2. Matrix for comparison of elements.</i>		
No	Matrix for comparison of elements	CR Value
1	Comparison of elements of level I criteria based on the goal of determining FOSS-based NMS software	0,09
2	Comparison of level II sub criteria elements based on criteria: determination of FOSS-based NMS software → Technology Related	0,03
3	Comparison of level II sub criteria elements based on criteria: determination of FOSS-based NMS software → User Related	0,01
4	Comparison of level II sub criteria elements based on criteria: determination of FOSS-based NMS software → Vendor Related	0,01
5	Comparison of level III alternative elements based on target-criteria-sub criteria: determination of FOSS-based NMS software → Technology Related → Flexibility	0,04
6	Comparison of level III alternative elements based on target-criteria-sub criteria: determination of FOSS-based NMS software → Technology Related → Implementation	0,02
7	Comparison of level III alternative elements based on target-criteria-sub criteria: determination of FOSS-based NMS software → Technology Related → System Requirements	0,03
8	Comparison of level III alternative elements based on target-criteria-sub criteria: determination of FOSS-based NMS software → Technology Related → Real-time Changes	0,03
9	Comparison of level III alternative elements based on target-criteria-sub criteria: determination of FOSS-based NMS software → Technology Related → Back-up System	0,05
10	Comparison of level III alternative elements based on target-criteria-sub criteria: determination of FOSS-based NMS software → Technology Related → Web Based Application	0,03
11	Comparison of level III alternative elements based on target-criteria-sub criteria: determination of FOSS-based NMS software → Technology Related → Features	0,01
12	Comparison of level III alternative elements based on target-criteria-sub criteria: determination of FOSS-based NMS software → Technology Related → Fault Management	0,04
13	Comparison of level III alternative elements based on target-criteria-sub criteria: determination of FOSS-based NMS software → Technology Related → Configuration Management	0,01
14	Comparison of level III alternative elements based on target-criteria-sub criteria: determination of FOSS-based NMS software → Technology Related → Accounting Management	0,01
15	Comparison of level III alternative elements based on target-criteria-sub criteria: determination of FOSS-based NMS software → Technology Related → Performance Management	0,01
16	Comparison of level III alternative elements based on target-criteria-sub criteria: determination of FOSS-based NMS software → Technology Related → Security Management	0,02
17	Comparison of level III alternative elements based on target-criteria-sub criteria: determination of FOSS-based NMS software → User Related → Customization	0,03

18	Comparison of level III alternative elements based on target-criteria-sub criteria: determination of FOSS-based NMS software → User Related → User Friendliness	0,01
19	Comparison of level III alternative elements based on target-criteria-sub criteria: determination of FOSS-based NMS software → User Related → Reporting	0,01
20	Comparison of level III alternative elements based on target-criteria-sub criteria: determination of FOSS-based NMS software → User Related → Integration with Other plugin	0,00
21	Comparison of level III alternative elements based on target-criteria-sub criteria: determination of FOSS-based NMS software → Vendor Related → Support	0,00
22	Comparison of level III alternative elements based on target-criteria-sub criteria: determination of FOSS-based NMS software → Vendor Related → Documentation	0,01
23	Comparison of level III alternative elements based on target-criteria-sub criteria: determination of FOSS-based NMS software → Vendor Related → Maintenance	0,01
24	Comparison of level III alternative elements based on target-criteria-sub criteria: determination of FOSS-based NMS software → Vendor Related → Vendor Credentials	0,01

It concluded that the pairwise comparison provided by expert respondents has a value of inconsistency ratio that is smaller than 0.1 as the largest limit of inconsistency ratio values. Thus, the results of combined geometric calculations of respondents' data are quite consistent.

IV. CONCLUSIONS

4.1 Conclusions

Some conclusions drawn from the implementation of this thesis are as follows:

- a. The method and evaluation in determining the choice of FOSS-based NMS software as an effective network, server, and application monitoring tool is the Analytical Hierarchy Process (AHP) approach which determines selection criteria based on the Oyku Alanbay research method with change and integration of ISO 7498-4.
- b. Nagios is a FOSS-based NMS software that is reliable enough to use as a network monitoring tool, server, and computer application because Nagios meets all the elements of the criteria and sub-criteria specified in this study by getting the highest weight.

4.2 Suggestions

Based on the results of the research implications presented in the previous chapter, the authors recommend:

- a. This research can be used as a reference in determining reliable FOSS-based NMS software for monitoring network devices, servers, and computer applications.
- b. For software developers, especially NMS software vendors are advised to pay more attention to matters on the user (User Related), both in terms of ease of use and in terms of customization, so users can adjust to existing needs, both from the network, server and application side.
- c. This research can further be developed, among others by exploring or breaking down in terms of its features, because on a choice of NMS software, often the first thing to look at is the completeness of the features.

REFERENCES

- [1] U. Raza, P. Kulkarni, and M. Sooriyabandara, "Low Power Wide Area Networks: An Overview," *IEEE Commun. Surv. Tutorials*, vol. 19, no. 2, pp. 855–873, 2017, doi: 10.1109/COMST.2017.2652320.
- [2] Y. Yu *et al.*, "Fault management in software-defined networking: A

- survey,” *IEEE Commun. Surv. Tutorials*, vol. 21, no. 1, pp. 349–392, 2019, doi: 10.1109/COMST.2018.2868922.
- [3] O. Alanbay, “Erp Selection Using Expert Choice Software,” *ISAHP July 8-10, 2005*, no. 47, pp. 1–14, 2005, [Online]. Available: <http://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.579.9207&rep=rep1&type=pdf>.
- [4] F. Wang, S. Tan, J. Peng, and Y. Chang, “Process monitoring based on mode identification for multi-mode process with transitions,” *Chemom. Intell. Lab. Syst.*, vol. 110, no. 1, pp. 144–155, 2012, doi: 10.1016/j.chemolab.2011.10.013.
- [5] J. Hughes, “Open source network management download comparison,” 2020. <https://techteapot.com/open-source-network-management-download-comparison/>.
- [6] Cisco, “Network Management System : Best Practices White Paper,” 2011. <https://www.cisco.com/c/en/us/support/docs/availability/high-availability/15114-NMS-bestpractice.pdf>.
- [7] M. . A. Jabraeil Jamali, *IoT Security - Huawei*. 2020.
- [8] GroundWork, “Unified Monitoring,” *Environment*, 2020. <https://www.gwos.com/features/unified-monitoring/>.
- [9] vmware, “Hyperic HQ,” 2016. <http://www.vmware.com/products/vrealize-hyperic.html>.
- [10] Nagios, “Application Monitoring With Nagios □,” 2020. .
- [11] OpenNMS.com, “The OpenNMS Platform,” 2020. <https://www.opennms.com/opennms-platform/>.
- [12] Zenoss.com, “Application-Centric Network Monitoring & Analyti,” 2020. <https://www.zenoss.com/product/network-monitoring>.

of third parties. The author agrees that the decision to publish or not to publish a manuscript in the journal submitted by the author, is entirely the right of the journal manager. Prior to the final acceptance of the manuscript, the author is required to assert in writing, that the written post is copyright of the author and assign this copyright to the journal manager.

Copyright

All unpublished manuscripts can be submitted elsewhere. The author is responsible for the publication's permission or recognition of images, tables and numbers in the manuscript he sent. The manuscript is not a result of plagiarism and does not violate any other rights