Operating System Selection For Information System Practicum: A Comparative Study Of Rocky Linux, Fedora, And Kali Linux Using The Technique For Order Of Preference By Similarity To Ideal Solution

Yohanes Karunia Desdelka Simatupang
Information System Department
Telkom University
Bandung, Indonesia
yoha7288@gmail.com

Muhammad Fathinuddin
Information System Department
Telkom University
Bandung, Indonesia
muhammadfathinuddin@telkomuniversity.ac.id

Umar Yunan Kurnia Septo Hediyanto
Information System Department
Telkom University
Bandung, Indonesia
umaryunan@telkomuniversity.ac.id

***Abstract* — The research compares three Linux distributions; Kali Linux, Fedora and Rocky Linux to determine which Linux Operating Systems are best tailored to support practicum activities in Information Systems Department in Faculty of Industrial Engineering Telkom University. Given the rise in importance of Linux in today's learning institutions due to its open source nature, configurability, and stability, choosing the right operating system to run laboratories on is important. The evaluation approach applies the TOPSIS method of multi-criteria decision analysis using the direct rating method to assign weights to the criteria. The evaluation is done across four areas: tool compatibility, systems defaults, user experience and performance & stability. Simulated practicum were done within five laboratory modules to evaluate each OS in the context of realistic practices. These results indicate that Rocky Linux as the first OS is highest in readiness, compatibility and reliability, with Fedora and Kali Linux following closely behind. These results can be used by educational institutions to better design system environments to support engineering education and good/secure/sustainable learning environments.**

***Keywords*—Operating System, Practicum, TOPSIS, Direct Rating, Practicum, Rocky Linux, Fedora, Kali Linux, Information System**

I. INTRODUCTION

Operating systems are essential for managing hardware and software resources on a computer and are an integral part of engineering education. In the Industrial Engineering Faculty, operating systems are used to support modules that are more practical in nature, allowing students to simulate industry settings, manage IT infrastructure, and practice with actual tools used in the field. Consistent with the philosophy of Education 4.0 and the requirements of Industry 4.0, education needs to deliver a stable, safe and flexible systems platform that enables learning project-based learning settings [1], [2].

In particular, Linux distributions have been increasingly popular for use in educational contexts for their open source nature, community development and high customizability. The Linux kernel performs such important functions as allocating memory and resources, controlling interfacing with hardware, running simulations and managing processes in a system, and enabling industrial applications. Linux also allows for this sort of hands-on experience as students can physically experience the inner workings of the operating system from architecture to packages to sys admin skills.

Some of the more popular distributions include Fedora, Kali Linux, and Rocky Linux. Fedora is known for regular updating, being developer friendly, and for innovation. It also comes with strong security mechanisms like SELinux, and has the backing of Red Hat. Kali Linux, on the other hand, is purpose built for penetration testing and cyber security modules; it is for teaching ethical hacking and system secure design. At the same time, Rocky Linux was created to continue the tradition of CentOS' previous support policy, providing enterprise-level stability and support while remaining 100% compatible with RHEL (Red Hat Enterprise Linux).

Even with these systems in place, there are still inconsistencies throughout the labs of the Information Systems Department. The use of different OS distributions can cause various compatibility issues, different performance levels as well as configuration differences that makes the practicum difficult to implement. Different tool dependencies, module failures, or user experience issues that each lab may encounter due to the underlying OS could eventually impact the consistency of instruction and student performance. [8]

In an effort to overcome these issues this study will assess and compare Fedora, Kali Linux and Rocky Linux as potential operating systems for lab practicum work. The central research questions are; (1) how has practicum utilizing operating systems been implemented in the present? (2) Which OS is best suited to support practicum environments in terms of compatibility, default setup, ease of use, and performance? (3) What recommendations can be made to standardize laboratories and improve student learning experiences?

This study proceeds to establish a decision making framework using TOPSIS (Technique for Order of Preference by Similarity to Ideal Solution) with Direct Rating weighting. It also combines hands-on experience, technical assessments and lab work to curate a list with statistical rankings. Their findings should help the Faculty of Industrial Engineering adopt a better aligned, more effective and technically long-lasting OS standard for future practicum.

II. LITERATURE REVIEW

Menyajikan dan menjelaskan teori-teori yang berkaitan dengan variabel-variabel penelitian. Poin subjudul ditulis dalam abjad.

A. Definition of Operating System

An Operating System (OS) is the collection of software that manages the hardware and software resources on a computer and provides services to application programs. It is the link between the hardware and the user. Uppala [9] states that modern OSs are starting to use more artificial intelligence in order to enhance performance, security and stability. Sholademi et al. In addition, [10] point out that the machine learning abilities are also built in to adapt itself dynamically to new cyber security threats, thus making the OS also very much relevant in real time. Also, Aslam and Ahmad [11] also point out that modern operating systems have grown to become compliant platforms with data pipelines.

The fact that they are open-source computer platforms based on the Linux operating system contributes to their use in education since they are flexible, modifyable, have an open source code, and have collaborative development networks around the world. This flexibility, as Silchenko states, makes Linux good for experimentation, research and training in technology [12].

B. Operating System Category

Types of operating systems can be defined by their common purpose and application; Desktop, Server and Security OS.

1) Desktop Operating System

Desktop Operating Systems Desktop OSs are for everyday activities like browsing, multimedia and productivity applications. These include Windows, macOS and various Linux distributions such as Fedora Workstation. Desktop OSs provide GUIs, a wide range of hardware, and ease of use. Stelz [13] argues that desktop usability requires things like GUI-based user environments, support for peripherals and updates on a regular basis.

2) Server Operating System

Server Operating Systems Server OSs are optimized for backend services, such as web hosting, database and virtualization. Examples include Rocky Linux, Fedora Server, and Windows Server. Zhao et al. [14] indicates that such systems often rely on command line interfaces (CLI) to increase performance, and support virtualization through technologies such as Docker and Podman.

3) Security Operating System

Security Operating Systems Security focused OSs for threat detection, forensics, and penetration testing. Kali Linux is a good example with tools for security checking and ethical hacking pre-installed. These OSs are deployed in the government and enterprise sectors due to their strong security auditing capabilities, as mentioned by Sedelnikov and Koltygin [15].

C. Development and Evolution of Linux

The Linux kernel, originally developed by Linus Torvalds in 1991, is the operating system kernel at the core of many open source operating systems. According to Guo, Liang, and Long [16], the rapid innovative aspects of the open source model of Linux has become intrinsic within modern IT infrastructure.

Distributions are adapted to use in a given domain, as for example Ubuntu is useful on desktops, CentOS and Rocky Linux for enterprise and servers. Apokin and Khandozhko [17] mention scalability of Linux, as it is able to support systems from embedded up to high performance computing. Feng et al. [18] add that Linux is now at the core of software-defined infrastructure and cybersecurity, thanks to its use of containerization, virtualization, and cloud-native technologies like eBPF.

Distributions such as Rocky Linux arose as stable, community-supported alternatives after CentOS's change in support. Singh [19] describes this agility in augmenting the ability of the Linux ecosystem to address both enterprise and academic needs.

D. Overview of Compared Operating System

The present research will narrow its scope to three different Linux distributions: Fedora, Kali Linux and Rocky Linux. They both have different functions in the academic environment:

1) Fedora

Fedora is a modern Linux distribution known for its cutting-edge updates, GNOME desktop environment, and robust package management system.It comes with SELinux and is great for development and educational labs. Lachance and Mosser [20] support the view that it serves as a testbed for enterprise features that are later incorporated into Red Hat Enterprise Linux (RHEL)..

2) Kali Linux

Kali Linux is a Debian-based distribution tailored for cybersecurity training.It comes with a whole suite of penetration testing tools. Sedelnikov and Koltygin [15] point out that Kali Linux is mostly used in ethical hacking and digital forensics, and it is not a general-purpose or an end-user OS.

3) Rocky Linux

Rocky Linux was born to be a drop-in replacement for CentOS, which had transitioned to CentOS Stream. It is community supported long term and is compatible with RHEL. According to Singh, Rocky Linux shines in servers and enterprise deployments where resources are minimal and decent default configurations are critical.

E. Kernel Level Difference

All three of the operating systems studied here are built on the Linux kernel, but there is considerable amount of difference in system-wide configuration. Fedora runs the most current upstream kernels, so it is better for development than for production. Kali Linux patches the kernel for

penetration testing tools and drivers including packet injection modules, while Rocky Linux is conservative, with LTS support and tuned enterprise-grade kernels. Such differences affect the behavior of the system, support for tools and update policies. In Borges et al. As [23] points out, kernel options can make or break stability, performance and compatibility in a given field across different distributions.

III. METHODOLOGY

In this chapter, a comparison of the applicability of Fedora, Kali Linux, and Rocky Linux as operating systems for Practicum activities will be outlined. It contains the methodology and theoretical framework utilized in the research, as well as the organized methodology of solving the research problem with the TOPSIS (Technique for Order of Preference by Similarity to Ideal Solution) with Direct Rating in determining the weights.

A. Conceptual Framework

TOPSIS, proposed by Hwang and Yoon, is the traditional widely used MCDM method employed in the study. This approach considers several possible solutions by assessing their relative distance from an ideal (best) and anti-ideal (worst) solution. This study uses four evaluation criteria:

- C1: Tools Compatibility
- C2: Default System Configuration
- C3: User Experience
- C4: Performance & Stability

Weights are assigned using the Direct Rating technique, which requires a good knowledge and direct experience of the laboratories settings. It begins with scoring each criterion subjectively and normalizing it into weights based on percentages.

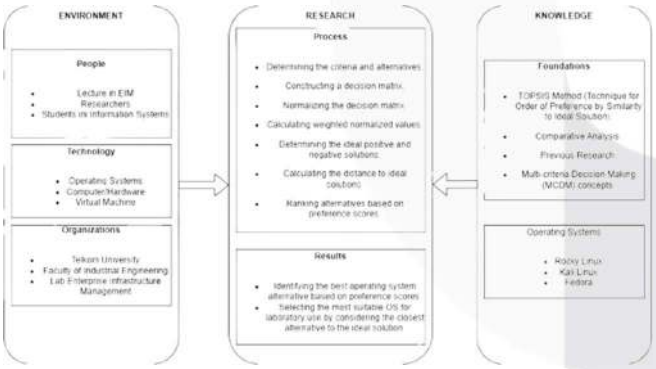

FIGURE 1

1) Environment

The testing took place in an isolated environment deploying virtual machines over a Windows 11 host with Oracle VirtualBox. In the interest of fairness, all operating systems (Fedora 42, Kali Linux 2025.1a, Rocky Linux 9.5) were given the same configuration (4GB of RAM, 2-core CPU, 32GB of storage). The practicum modules that were tested consisted of five labs: EDM, EIM, EISD, ERP, and SAG.

2) Research

This research identifies and evaluates each OS according to these four criteria. The technique for order of preference by similarity to ideal solution (TOPSIS) method is utilized to evaluate the best overall performing OS by creating a weighted decision matrix, determining distance from ideal and negative-ideal solutions and determining a preference score.

3) Knowledge

This research combines theoretical knowledge of the characteristics of OS as well as actual performance data from real lab implementations. It is a method that helps on a structured analysis and decision making on which is the best possible OS.

B. Systematization of Problem Solving

This section presents a taxonomy to determine the ideal operating system to be utilized within the Faculty of Industrial Engineering practicum. The study follows the Design Science Research Methodology of Ken Peffers which is customized to suit the employed combination of TOPSIS and Direct Rating methods.

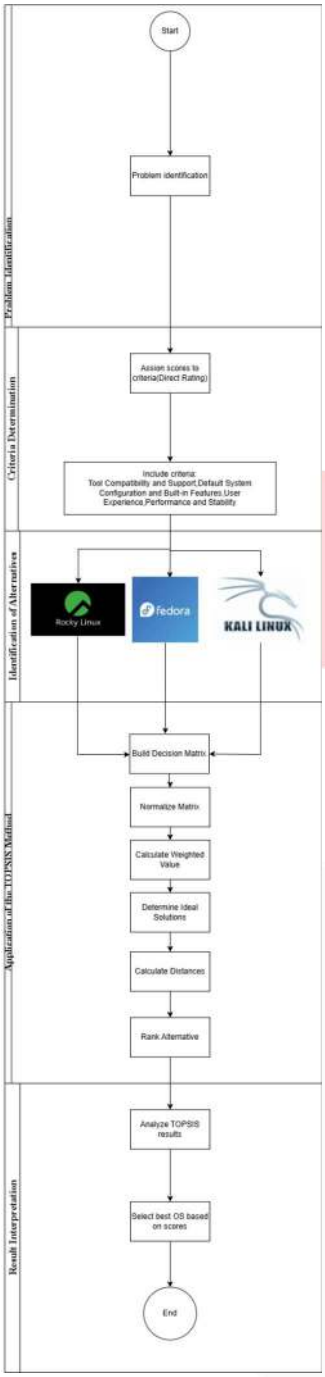

FIGURE 2

1) Identify Problem

OS options for supporting practicum activities need to be decided by the lab. The OS also needs to be compatible, easy to configure, high performing, and friendly.

2) Determine Evaluation Criteria

The next step is to select the criteria of evaluation that will be used to contrast different operating systems. The chosen criterias types are relevant to the operational considerations within a lab environment because:

- Tool Compatibility and Support: This refers to how comfortable and well supported operating system used would allow use of common software and technical tools within the practicum and that it would integrate well with current hardware infrastructure in the lab.
- Default System Configuration and Built-in Features: This looks at the ability of the operating system to be used in an educational context without much set up or installation beyond what is already provided with the system or readily available features.
- User Experience: Associated with how user-friendly and intuitive the operating system is for the students as end-users and for the instructors or lab assistants who run the sessions.
- Performance and Stability: How well and reliably the operating system performs practical tasks without crashing or causing system errors, as well as how stable it is when running in multi-user or resource hungry scenarios.

3) Assign Scores using Direct Rating

The scoring for each criterion reflects the practical importance of that criterion in a lab setting. For instance on the case of compatibility it obtained the highest as it is very important in deciding if practicum tools will be able to run..

4) Normalize Scores to Generate Weights

These scores are then proportionally converted by dividing each individual score by the complete score; for example, 0.36, 0.29, 0.21, 0.14.

5) Input and Score Alternatives

Scores for Fedora, Kali Linux and Rocky Linux are given in accordance to how they performed based on their characteristics within controlled lab modules .

6) Construct Decision Matrix

The evaluation's unprocessed scores are organized in a matrix.

7) Normalize Decision Matrix

Vector normalization is used to normalize the decision matrix.

8) Calculate Weighted Normalized Matrix

Each of the normalized values is then multiplied by the criterion weight associated with that criterion.

9) Determine Ideal Positive and Negative Solutions

For each criterion, the worst and the best values are specified.

10) Calculate Euclidean Distance

The distances to the ideal solutions of each alternative are calculated.

11) Calculate Preference Scores and Rank Alternatives

The alternatives are then ordered according to proximity to the ideal solution.

12) Conclusion and Recommendation

The chosen operating system is the one with the highest scores of preference and is deemed the most appropriate for use in laboratory practicum.

IV. RESULT AND DISCUSSION

The outcome of the comparative analysis and the performance of the three Linux-based operating systems along the parameters established in the research are presented in this chapter.

A. Summary of All Laboratories

The assessment took place in five lab modules: EDM, EIM, EISD, ERP and SAG. All modules were tested on the standardized VMs using Fedora, Kali Linux and Rocky Linux.

TABLE 1

OS	C1	C2	C3	C4
Fedora	8	7	7	7
Kali Linux	6	5	5	6
Rocky Linux	9	9	8	9

B. Evaluation using TOPSIS Method

1) Normalized Decision Matrix

Vector normalization was applied to make each criterion comparable, as they are measured on varied scales.

2) Weight Assignment

The weights given using the Direct Rating method were: Compatibility of Tools: 0.36 Default Sys. Config. 0.29 UX: 0.14 Performance & Stability : 0.21

3) Weighted Normalized Matrix

The obtained normalized scores were then weighted.

4) Ideal and Negative-Ideal Solutions

Ideal Solution (A+): The best values for all the alternatives. Negative-Ideal Solution (A): Worst values from all alternatives

5) Distance Calculation

Euclidean distance was applied to compute the distance of each alternative from A^+ and from A^-.

TABLE 2

OS	C1 (0.36)	C2 (0.29)	C3 (0.14)	C4 (0.21)	TOTAL
Fedora	0.288	0.203	0.098	0.147	0.736
Kali Linux	0.216	0.145	0.070	0.126	0.557
Rocky Linux	0.324	0.261	0.112	0.189	0.886

TABLE 3

Rank	OS	Score
1	Rocky Linux	0.886
2	Fedora	0.736
3	Kali Linux	0.557

C. Interpretation

Based on the analysis, Rocky Linux will be the best fit for the different labs in support of practicum activities. It was rated best in all categories with no major compatibility and performance issues. Fedora has decent ratings for being fairly well supported by developers, good usability and requires slight configuration tweaks. Kali Linux, while great for security labs ranked the lowest because it is not compatible by default with general purpose practicum tools.

V. CONCLUSION

According to the comparative assessment using TOPSIS method, Rocky Linux is the best candidate OS for laboratory practicum in the Faculty of Industrial Engineering. It was better across the board in interoperability, defaults, usability and system stability.

Another option for development labs is Fedora, which has a great ecosystem and is up to date very often. Kali Linux was very specialized for Cybersecurity, while excellent for a practicum application it was not as good as general-purpose because of its limited default packages and its learning curve.

REFERENCES

[1] M. F. Kamaruzaman, R. Hamid, S. A. Mutalib, and M. S. Rasul, "Conceptual framework for the development of 4IR skills for engineering graduates," Global Journal of Engineering Education, vol. 21, no. 1, pp. 54–61, 2019.

[2] M. S. Borkar and A. Singh, "Virtualization Technology in Education for Industry 4.0," International Journal of Recent Technology and Engineering (IJRTE), vol. 8, no. 3S, pp. 103–107, 2019.

[3] V. Mishra, S. Aleem, and S. Ahmad, "A Survey of Linux Operating System as an Alternative to Other Operating Systems," International Journal of Scientific Research in Network Security and Communication, vol. 11, no. 6, pp. 9–14, 2023.

[4] G. Ge, P. Tian, M. Yang, and Z. Yang, "Linux Programming Curriculum Reform in the Era of Open Source

Hardware and Software," Frontiers in Educational Research, vol. 7, no. 8, pp. 8–15, 2024.
[5] K. Hitchcock, Linux System Administration for the 2020s. Apress, Springer, 2022..
[6] I. A. Sedelnikov and D. S. Koltygin, "Web application developing methodology for managing robotic complexes," Vestnik of Astrakhan State Technical University. Series: Management, Computer Sciences and Informatics, no. 1, pp. 56–63, 2024.
[7] S. K. Singh, Linux Yourself: Concept and Programming. 1st ed., CRC Press, 2021. [Online]. Available: https://doi.org/10.1201/9780429446047
[8] P. Wu, "Blended Teaching Practice of Linux C Programming for the Cultivation of Practice and Innovation Ability of New Engineering," in Advances in Education, Humanities and Social Science Research, vol. 5, ICEACE, pp. 510–514, 2023.
[9] V. K. Uppala, "The Impact of AI on Architecting Cloud Data Platforms: Enhancing Data Processing and Integration," Journal of Artificial Intelligence, Machine Learning & Data Science, vol. 1, no. 1, pp. 1289–1292, 2022.
[10] D. O. Ihenacho, D. B. Sholademi, V. A. Kehinde, and B. T. Tayor, "Integrating Cybersecurity Intelligence with Data Analytics for Real-Time Decision-Making in Management Information Systems," 2024 IEEE International Conference on MIS Applications, 2024.
[11] E. Aslam and N. Ahmad, "IAM and SOX Compliance in Modern Cybersecurity: How AI and Data Pipelines Shape Advanced Network Security Strategies," Cybersecurity Review, vol. 12, no. 2, pp. 88–96, 2024..
[12] V. Silchenko, "Application Areas of Artificial Intelligence and Customer Service Automation Tools in Restaurant Enterprises," Science and Technology Today, no. 11(39), 2024.
[13] R. Stelz, "Software-Separation on Embedded Hardware – Implementing a Lightweight Hypervisor," FMS-Berichte, vol. 37, pp. 1–7, 2024.
[14] Y. Zhao, S. Li, W. Xue, Y. Zhao, and Y. Wen, "CarePlus: A General Framework for Hardware Performance Counter-Based Malware Detection under System Resource Competition," Computers & Security, vol. 143, 103884, 2024.
[15] A. Rashid and A. Chaturvedi, "Virtualization and Its Role in Cloud Computing Environment," International Journal of Computer Science and Engineering, vol. 7, no. 4, pp. 1131–1136, 2019.
[16] H. Tabrizchi and M. K. Rafsanjani, "A Survey on Security Challenges in Cloud Computing: Issues, Threats, and Solutions," Journal of Supercomputing, vol. 76, no. 12, pp. 9493–9532, 2020.
[17] G. J. Blinowski and P. Piotrowski, "CVE-Based Classification of Vulnerable IoT Systems," in Theory and Applications of Dependable Computer Systems (DepCoS-RELCOMEX), Springer, vol. 15, pp. 82–93, 2020.
[18] J. Guo, H. Liang, and J. Long, "Leveraging File System Characteristics for Ransomware Mitigation in Linux Operating System Environments," IEEE Transactions on Dependable and Secure Computing, 2024.
[19] E. M. Apokin and V. A. Khandozhko, "Software Implementation of a Hierarchical Approach to Solving the Inverse Problem of Kinematics of a Robot Manipulator," Southwest State University Journal, vol. 72, 2024.
[20] M. Feng, J. Zhou, and Y. Tang, "Enhancing Cloud-Native Security Through eBPF Technology," in 2024 IEEE 11th International Conference on Cyber Security and Cloud Computing (CSCloud), pp. 165–168, 2024.
[21] R. Lachance and M. Mosser, "Fedora and Upstream Software in Higher Education: A Pedagogical Analysis," Open Source Education Journal, vol. 2, no. 1, pp. 33–41, 2024.
[22] R. Fox, Linux with Operating System Concepts, 2nd ed. CRC Press, 2021. [Online]. Available: https://doi.org/10.1201/9781003203322
[23] H. Borges, J. A. Pereira, D. Khelladi, and M. Acher, “Linux Kernel Configurations at Scale: A Dataset for Performance and Evolution Analysis,” arXiv preprint arXiv:2505.07487, 2025.