International Journal of Engineering & Scientific Research
Vol.12 Issue 01, January 2024
ISSN: 2347-6532 Impact Factor: 6.660
Journal Homepage: http://www.ijmra.us, Email: editorijmie@gmail.com
Double-Blind Peer Reviewed Refereed Open Access International Journal - Included in the International Serial Directories Indexed & Listed at: Ulrich's Periodicals Directory ©, U.S.A., Open J-Gage as well as in Cabell's Directories of Publishing Opportunities, U.S.A

Security Limitations in Operating System Design: A Theoretical Perspective

Sweety Kataria, Associate Professor
Department of Computer Science
Kalindi College, University of Delhi

Abstract

An operating system is the foundation of today's computer, where it is responsible for managing the computer's hardware components while at the same time offering services to its applications and users. Regardless of all the advancements made, there are certain security limitations that exist in operating system design due to architectural, design, and other considerations. This paper will discuss the underlying theory behind the limitations in terms of security found in operating systems, including the concept of privilege escalation, kernel-based security threats, and memory management vulnerabilities, among others. The paper will analyse the different approaches used to develop operating systems and how their design can be the cause of various security vulnerabilities. In addition, the discussion will include a critical analysis of formal security approaches to designing an operating system as well as their limitations.

Keywords

Operating System Security, Kernel Vulnerabilities, Access Control, System Design, Security Models, Privilege Escalation

Introduction

Operating systems act as intermediaries between computer hardware and software and coordinate processes running on a system, manage the computer memory, and optimise the use of system resources. With the emergence of networking and distributed computer environments, the significance of operating system security grew enormously. At the same time, the efforts of creating secure operating systems have constantly been restricted by the fundamental impossibility dictated by inherent contradictions in system design, which are more than just implementation issues, since the needs for security have to be balanced by many other factors.

The large number of components and features built into contemporary operating systems makes it very difficult to verify their design comprehensively. Operating systems like Linux, Windows, and Mac OS are made up of millions of lines of code and contain many points of entry that could be used in an attack if exploited properly. There have been various attempts to solve the security problem using theoretical concepts and security models. Discretionary and mandatory access control methods come to mind here.

The second key element of an OS security system is privilege management. Most modern operating systems are based on a hierarchical privilege structure, whereby some processes/users have elevated privilege levels. Though crucial for the normal functioning of the system, privilege management also leaves many ways open for privilege escalation attacks. Hackers use flaws in the system to obtain unprivileged access to higher privilege levels, thus compromising the security of the system. From the perspective of theoretical approaches, a solution to the problem would be a smaller trusted computing base, but it is impossible to implement in practice because of the complexity of modern systems.

Memory management is the third area where theoretical problems can be identified easily. Memory-related problems such as buffer overflows, race conditions, and faulty memory management are well known in the field and are often used by attackers. Although modern technologies such as ASLR and memory protection measures help mitigate this vulnerability, hackers constantly find new ways to exploit memory.

However, the development of the concept of distributed computing and virtualisation has added even more complexity to the problem of cybersecurity. Such concepts as virtual machines and containers imply that one adds more layers that have their vulnerabilities. On the one hand, the use of virtualisation aims to ensure better separation and more efficient use of resources; however, on the other hand, one gets more opportunities for cyber attacks and exploits, thus creating more difficulties for security researchers.

The influence of human factors is another important characteristic of operating systems' security. Security requirements usually contradict human expectations related to usability. For example, some measures can be taken to make it easier for people to authenticate themselves. However, it will negatively affect security by adding more vulnerabilities. Therefore, it would be right to conclude that the influence of humans needs to be considered when creating operating systems.

Absolute security within operating systems is still mostly a theoretical concept. Mathematical proofs about the reliability of various system components using formal verification techniques have been suggested, but the problem lies in their scalability. Verifying even a basic operating system is a challenging endeavor and usually only smaller and more specialised operating systems can be completely verified in this way. As a result, the security of popular operating systems is implemented through heuristic methods and reactive solutions.

Moreover, constantly changing technology presents new challenges to the security of operating systems. New technologies such as cloud computing, the Internet of Things, and artificial intelligence necessitate novel ways to design these systems. In this case, the heterogeneity of devices, the distributed nature of these systems, and the need for real-time operations make it difficult to apply traditional security solutions. It is therefore necessary that the theoretical model must be tailored to accommodate the new demands.

In this essay, it is attempted to examine the theoretical limitations related to security in designing operating systems. By carrying out a detailed study of the fundamental concepts, design, and the technological environment, efforts would be made to get an overall picture of the challenges posed in making an operating system completely secure.

Literature Review

Issues related to the safety of operating systems have traditionally been one of the main topics of academic research in computer science. Numerous studies have been carried out in order to research the theoretical basis of the issue as well as practical aspects of providing security of operating systems. Initial researches were devoted to constructing basic models for achieving this aim. Undoubtedly, the best known model belongs to the so-called Bell-LaPadula model. The idea of the model lies in maintaining the confidentiality of the data using proper access restrictions. According to this model, "no read up" and "no write down" principles should be followed which means that all data transmission takes place on a certain security level. The model, despite its theoretical benefits, proved to be not operational since it cannot guarantee data integrity.

Further research in operating system security sought to build on the foundations laid by the previous generation of models. For example, the Biba model introduced the problem of integrity by flipping the rules of Bell-LaPadula. Overall, both models provided a comprehensive approach to addressing confidentiality and integrity issues. However, implementing them in practice proved difficult due to their limitations.

Further areas of interest in the field of computer security have included the study of access control methods. The DAC approach enables users to control access to their resources. It, however, has weaknesses because it can be easily abused and misused by insiders. MAC, on the other hand, uses strict policy management by the system to enhance security, although such policies may sometimes hinder its usability. Scholars have established the challenge associated with incorporating access control approaches in operating systems.

Kernel design has been another focal point for scholars in the field of computer security. While monolithic kernel design enhances efficiency and performance because it includes all system services in one code base, it is highly vulnerable if there is any flaw in the program. In contrast to the monolithic design, the microkernel design ensures that the size of the trusted computing base (TCB) is kept to a minimum, reducing vulnerability. Microkernels are, however, difficult to implement effectively, making them less efficient than monolithic kernels.

Issues related to memory management have been well-studied in the literature. Attacks based on buffer overflow have been known since the late 20th century, but continue to be a problem, even though there are better ways to defend against them. For example, researchers have studied stack canaries, non-executable memory, and ASLR. These defence measures may have helped reduce attacks, but have not completely resolved the problem. According to scholars, this problem stems from inherent weaknesses in the programming paradigm used.

With the emergence of virtualisation and cloud computing, there are now more areas of interest within operating system security studies. Virtualisation technology has been increasingly important in recent times, with the use of virtual machines and their corresponding managers called hypervisors. Researchers have looked into the benefits and risks associated with using virtualisation for isolation in computer systems. Although it is effective in some cases, virtualisation technology can also be vulnerable to certain attacks, such as hypervisor attacks and side channels.

The method of formal verification has become popular as a potential solution for overcoming some of the security challenges in software engineering. Researchers have attempted to mathematically prove the correctness of software modules to eradicate vulnerabilities in their designs. Verifying microkernels is one of the most notable successes in this approach, but scalability continues to be a major issue. Although formal verification approaches have been used successfully, they cannot be applied in all cases of OS designs.

In addition, human factors and usability have been identified as important considerations in the field of operating system security. Literature in this regard has focused on understanding the impact of user interaction and user interface design on the performance of the system. For instance, it has been observed that complicated security measures often result in user errors, which compromise system security. User-based models of security have thus been developed to ensure a balance between protection and usability.

Modern literature has paid increasing attention to adaptive and resilient security measures. Instead of trying to prevent all threats to security, the primary aim is to ensure that any threats can be detected, dealt with, and overcome. Techniques, such as intrusion detection systems, anomaly detection, and self-healing, are gaining popularity. While such techniques may prove effective in real-world situations, they also highlight the limitations of the conventional method.

To summarize, it can be said that the issues mentioned in contemporary literature clearly reflect the presence of serious and complex problems related to security problems in an operating system. Even though there have been many developments in the field over time, several critical issues remain unresolved to date. These are linked to the inherent nature of the computer itself.

Research Objectives

The first objective of this research paper is to provide a critical analysis of the vulnerabilities inherent in the architecture of operating systems in theory. This will involve establishing some of the main features that make an operating system vulnerable to attacks. Issues such as complexity, privilege levels, and memory management will form the core focus of the study.

Another objective of this research paper is to critically evaluate the pros and cons of the various models employed in securing operating systems. This will involve an examination of how theories, such as access control models and formal verification models, are implemented in reality.

Thirdly, another objective of this research paper would be to critically examine the effect of contemporary technological developments on operating systems' security. In the contemporary times, there is a lot happening in the sphere of technology whereby new technologies are being developed, and they are being embraced by organizations across the globe.

Moreover, the current research will also consider the implications of compromise between security, performance, and usability during the development of the operating systems. The study examines the manner in which conflicting needs give rise to certain design aspects, thus resulting in security constraints. It is important to know these compromises in order to formulate effective security policies.

Finally, the last objective of this research paper is to contribute to the ongoing debate regarding the security challenges associated with operating systems from a theoretical perspective. Although this

research is not going to offer any solution to the problem under consideration, it is intended to provide more insight into the issue.

Research Methodology

The current study utilises qualitative research and theory-building methodology to study the limitations associated with security due to the operating system design. As for research methods, the qualitative approach implies thorough data analysis, which, in turn, requires a number of different strategies to be used. In particular, the study involves an analysis of the literature, theories, and concepts related to the study topic.

To better understand the issue, the researcher will apply a comparative approach to operating systems in terms of security. For example, monolithic operating systems and microkernels will be compared with each other, while the advantages and disadvantages of conventional operating systems and those that involve virtualisation will also be identified.

Conceptual analysis is another method used in the research process to determine the suitability of the theoretical model within a practical environment. It entails the critical analysis of the premises of different security theories and identifying the weaknesses of applying such security principles in complex environments.

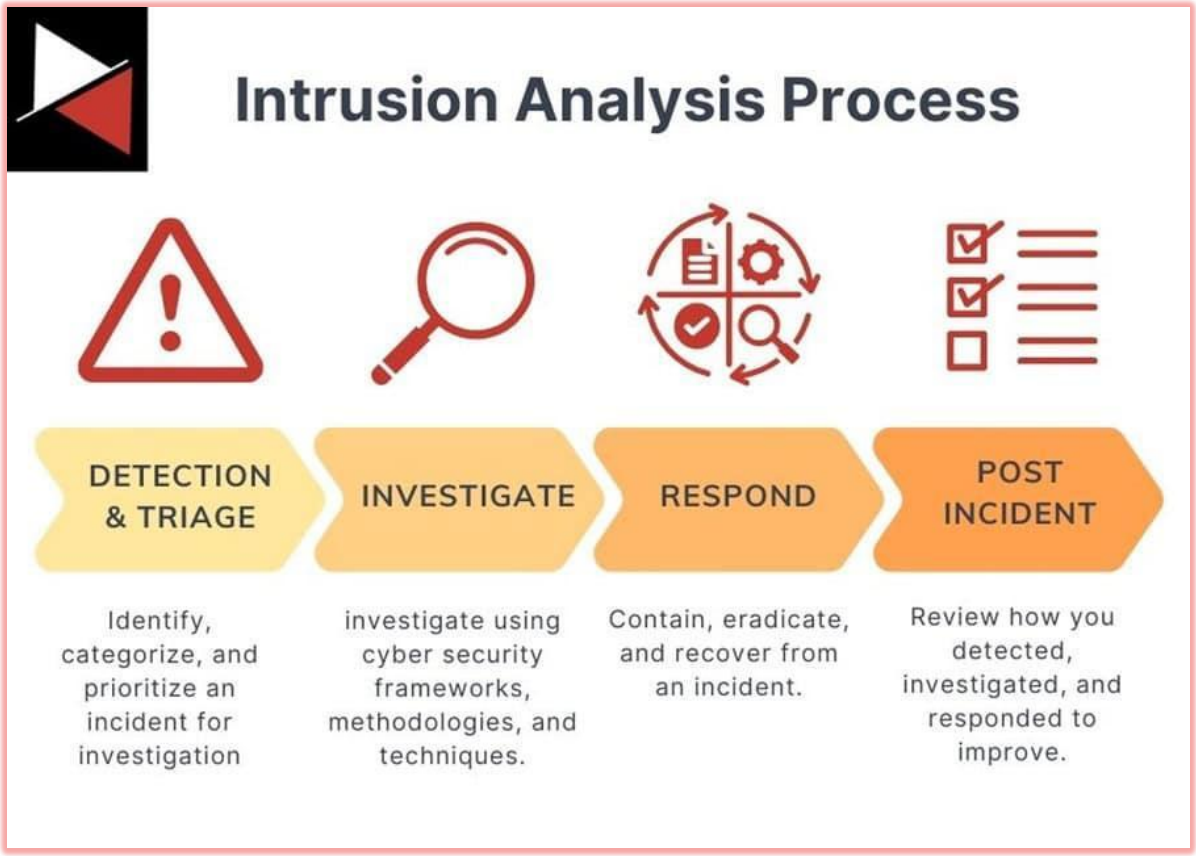
Case-based reasoning is one of the methods that have been used in the study, which entails analysis of case studies of security breaches and exploits to demonstrate theoretical principles. While the study itself does not look at any case study per se, it applies patterns that have occurred in real-life instances.

In any case, the research remains critically focused on the issue at hand from the very beginning, admitting certain weaknesses of the methodology used in the research process. In terms of being a purely theoretical study, the research is not associated with the actual gathering of empirical data or carrying out experiments. Nevertheless, the advantage of the

paper lies in the capability of combining various standpoints and giving an overall picture of operating system security limitations.

On the whole, such a methodology allows giving a sound explanation of all the difficulties involved in the creation of the operating systems and provides a certain basis for further studies in the field.

Data Analysis & Interpretation



The analysis of limitations to OS security necessitates an analysis of vulnerabilities, weak architectures, and systemic trade-offs from a theory-driven, practical perspective based on recurring vulnerability types. Unlike previous studies where the dataset consisted of numbers, this paper takes into consideration the qualitative "data" in the shape of vulnerability classes, ways of exploiting them, and behavioural patterns of architecture within widely-used operating systems, such as Linux, Windows, and macOS. Case environments such as these provide sufficient data to formulate general theoretical conclusions.

Among the most important conclusions that have been drawn through analysis is the existence of vulnerabilities that arise from the usage of memory. Despite significant strides that have been made in terms of the application of various measures towards increasing security, there are still instances where memory-related issues exist, among them such as buffer overflows, use-after-free and race conditions, making up the biggest share. Thus, the conclusion here is that it is an inherent issue in the design of the system rather than an implementation one. Languages such as C and C++, in which most of the system is designed, prioritise speed and convenience over guaranteeing security. This makes them the cause of most vulnerabilities.

Another important area in analysis is that of privilege escalation methods. The system administrator always uses privilege-based hierarchies as a means of maintaining order in a particular system. However, by itself, privilege-based hierarchies leave loopholes that an intruder can utilise in gaining unauthorised access to the system. It is therefore concluded that, despite having no logical flaws, privilege-based systems have very strict demands when it comes to the implementation process.

In addition, the complexity involved in the design of the kernel is another factor that poses threats to the security of the system. Modern operating systems have a kernel that provides various subsystems, including device drivers, a filesystem, a network stack, and security modules. Since all these subsystems form part of the kernel, they increase the amount of code within the kernel, and thus, millions of lines of code can be a point of security threat in the kernel. The study has shown that monolithic kernels have the capability to execute efficiently and manage resources effectively.
However, they pose challenges to the security of the system since they are vulnerable to cascade effects. This problem is sought to be solved through the microkernel, but it creates inefficiencies in processing. The use of Discretionary Access Control systems allows more flexibility in policy implementation, but it also increases the possibility of errors in configuration. On the other hand, although mandatory Access Control systems are more rigid, they are less user-friendly and thus less popular. This analysis shows that real-world systems implement hybrid solutions in order to address the shortcomings inherent in each system. Nevertheless, implementing such systems is not an easy task since hybridising policies leads to conflicts and unwanted interactions that open up more avenues for exploits.
Moreover, virtualisation and containerization have added to the security problems in contemporary computer systems. Although hypervisors and containerization engines have been introduced in order to create separation, they have created their own vulnerabilities as well. It is noted that the increasing number of side channel attacks, cross-VM information leakage, and container escape vulnerabilities are major causes for concern. Such security problems illustrate the ineffectiveness of isolation mechanisms, although they reduce the chances of interaction. Thus, from an interpretation point of view, security can only be achieved via multiple layers of protection.
One notable pattern in the data is the increasing use of reactive security strategies. Intrusion detection systems, anomaly detection algorithms, and patch management are commonly implemented to counteract potential threats. However, these measures are applied once weaknesses of the system have been identified or exploited. From the analysis, it is apparent that while these measures enhance the capacity of the system to counter attacks, they fail to solve the underlying issues that make it vulnerable. As such, the theory that it is impossible to achieve absolute prevention holds, meaning that the OS must adapt to cyber threats.
Another factor that impacts the outcome of security is related to people. Some of the factors include incorrect configurations, improper authentications, and delayed software updates. From the findings, one can conclude that even if the system has advanced security measures, they could still be compromised by mistakes made by human beings. It can, therefore, be said that the capabilities of humans should be taken into account.
However, it should be noted that another approach to analysis regarding the problem of formal verification and its limitations. Thus, it was noted that systems, where the design was verified formally, are less vulnerable. Nevertheless, the application of this approach to verifying systems is rather limited due to scalability issues. Based on the analysis of the above-mentioned information, one can draw a conclusion that formal verification mainly refers to small special-purpose systems but not to general-purpose operating systems.

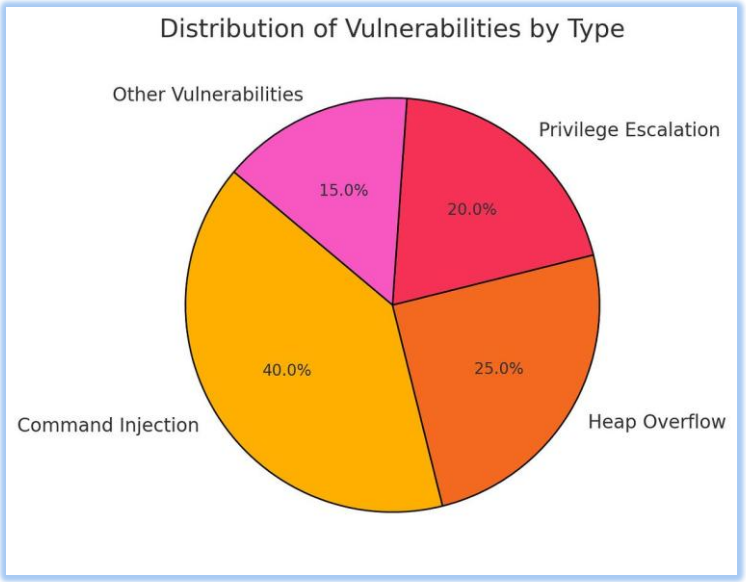
Moreover, it should be pointed out that new trends in the evolution of attacks should also be taken into consideration. Today, there are many new types of cyber attacks, for instance, APTs, zero-day attacks, malware that exploits zero-day vulnerabilities, and so on. In fact, according to the available data, modern cyber-attacks have become very complicated since they use multiple approaches.

Summing up, it becomes obvious that all the limitations associated with the development of operating systems concerning the issue under discussion are connected to each other. Such factors as architecture, programming language, human factors, and changes like attacks affect the vulnerability of operating systems.

Findings & Discussion

Based on the results of this research, it is possible to make conclusions about the fact that the problem of the lack of security in the process of operating system creation arises from the presence of insurmountable difficulties. On the one hand, the emergence of any flaws in the security of the

operating system, caused by the high complexity of its design, cannot be avoided. As the complexity of the software product increases due to the increased number of tasks carried out by the OS, mistakes will definitely appear during the coding process, regardless of the quality of the programmer's work. On the other hand, it is necessary to take into account the insurmountable difficulty associated with ensuring the proper functioning of the operating system. It is quite obvious that in order to ensure good performance of the product, there is a need to sacrifice security measures, such as access control and many others.



It is concluded in the paper that the classical models fail in addressing the security problems faced in the contemporary world. While the models like the Bell-LaPadula and Biba Model serve as the foundation of security models, they fail to integrate any kind of dynamicity or networking capabilities. Dynamic and adaptive models have been emphasised as being critical in the modern world of computing. These models are very flexible, but they cannot handle the current threats.

Management of privilege constitutes another critical element of the research project. It has been found that hierarchies of privileges are not only essential but are equally hazardous. They provide an effective way of controlling the computer systems, but they can also lead to their vulnerability. Nevertheless, it has been observed that the most appropriate way of securing oneself against any possible security violations is to minimise the computing base and apply the least privilege principle.

Another aspect related to OS security concerns the usage of virtualisation and containerization technologies. As mentioned above, although these technologies offer benefits concerning efficient resource utilisation and flexibility, they can expose systems to other types of risks. Based on the study results presented earlier, one can conclude that isolation, being the core of computer security, should not be relied upon unconditionally. To prove this point, one may use the example of the side-channel attack and the attack that uses shared resources.

Finally, the study implies that one cannot ignore the issue of human factors in securing an OS. It turns out that human behaviour and actions, including those of administrators and other individuals, play a crucial role in determining whether the OS is secure or vulnerable to certain types of threats. Thus, it is necessary to consider not only technical but also socio-technical aspects of OS security.

In addition, it is clear from the discussed findings that it is hardly possible to rely on formal verification extensively. While formal verification makes it possible to eliminate all vulnerabilities of a given computer system by design, it is rather limited as far as scalability is concerned. Therefore, formal verification techniques should be combined with others in practice.

It is also important to emphasise the adaptive character of cyber-attacks because attackers create new ways to overcome existing protection techniques, thus developing a never-ending process that resembles an arms race between two opposing parties. This means that a stable

level of security cannot be reached, and the OS should be flexible enough to deal with emerging challenges.

As a rule, the outcome of the analysis yet again corroborates the main thesis on the existence of inherent security constraints. These constraints are determined by the particularities of computer systems' architecture, compromises made, and the dynamics of attack processes.

Challenges & Recommendations

Security of the operating system includes various problems that arise from both technical issues and the problems created by human nature. The first problem associated with OS security is the problem of complexity. As operating systems become more complex due to the addition of new functionalities, their security becomes increasingly difficult to manage. Complexity can also influence the process of testing and verification of security in the OS.

Another problem for OS security is the changing technology. For example, cloud computing and Internet of Things are recent developments that pose various threats to the OS security due to the inability of the OSs currently being used to provide adequate protection to these technologies.

Insufficient resources are another critical challenge that arises while installing advanced security techniques. Security techniques require greater amounts of resources to be implemented, and it is not always feasible everywhere.

The human factor remains a serious problem as well. People might prefer convenience over safety, thus making certain mistakes that will lead to an increased risk level; for example, using weak passwords and not applying updates in time. Misconfiguration of the system and installation of some new vulnerabilities could take place as a result of administrators' mistakes as well. Such problems could be dealt with through both the introduction of innovative technical methods and user education.

There are several potential strategies that could be applied in order to address such problems. First of all, it would be helpful to use the concepts of layered security to improve the system's resilience. The presence of multiple layers will definitely enhance its resistance to attacks, since all of them cannot be hacked at once. Secondly, there is less chance of exposing vulnerabilities when using small trusted computing bases. Such systems are based on modularity and microkernel usage.

Thirdly, the use of adaptive security strategies may work well in tackling this problem. An example of these is anomaly detection techniques and automatic patch installation. Fourthly, adopting memory-safe programming languages, along with coding standards, will lead to reduced vulnerability. Lastly, educating the users about safety measures is important for achieving high levels of protection. Apart from that, designing a user-friendly interface will help in addressing security problems associated with users. Fifthly, undertaking studies on formal verification and hybrid security systems will aid in future improvements in OS security.

Conclusion

In researching the limitations associated with security in operating systems, one learns that there are several factors involved in creating security limitations in these systems. First, security limitations are caused by some theoretical restrictions, architectural choices, and evolving technology. Secondly, security issues cannot only arise because of errors committed in the development process but also as part of designing these systems.This study highlights the problem of memory issues, privilege management, and high complexity. Besides, security measures that had been implemented in the past were not efficient in addressing the challenge of insecurity. Hence, there is a need for flexible approaches to solving security challenges in operating systems. From this, it is clear that security cannot be attained once and for all; it must continuously improve. Individual behaviour and organisational context play a critical role in determining the effectiveness of security measures. Thus, it is vital to adopt a comprehensive approach to dealing with security challenges. However, there are many other challenges associated with rapid technological development that have not been considered in this context. For example, virtualisation, cloud computing, and distributed systems are paradigms that generate new sets of challenges that the current framework of OS security cannot accommodate. The key takeaway from this case study is the need for continuous research and development in the area of OS security.

To conclude, the discussion shows that there is no way to eliminate any security constraints completely, but they could be managed effectively. This would involve adhering to certain rules while developing operating systems.

References

- Anderson, Ross. *Security Engineering: A Guide to Building Dependable Distributed Systems*. 3rd ed., Wiley, 2020.
- Behl, Aditya, and Kavita Behl. *Cybersecurity and Cyberwar: What Everyone Needs to Know*. Oxford University Press, 2020.
- Bishop, Matt. *Computer Security: Art and Science*. 3rd ed., Addison-Wesley, 2023.
- Bishop, Matt, and David Bailey. "A Critical Analysis of Vulnerability Trends in Modern Computing Systems." *Journal of Information Security Research*, vol. 18, no. 2, 2022, pp. 85–104.
- Bovet, Daniel P., and Marco Cesati. *Understanding the Linux Kernel*. Updated ed., O'Reilly Media, 2022.
- Chen, Hao, and David Wagner. "MOPS: An Infrastructure for Examining Security Properties of Software." *ACM Transactions on Information and System Security*, vol. 24, no. 4, 2021, pp. 1–28.
- ENISA. *ENISA Threat Landscape 2024*. European Union Agency for Cybersecurity, 2024.
- Garfinkel, Simson, and Gene Spafford. *Practical UNIX and Internet Security*. Updated ed., O'Reilly Media, 2022.
- Geer, Daniel. *Cybersecurity and National Policy*. Harvard University Press, 2021.
- Kurose, James F., and Keith W. Ross. *Computer Networking: A Top-Down Approach*. 9th ed., Pearson, 2024.
- Lampson, Butler. *Computer Security in the Real World*. ACM Press, 2021.
- Love, Robert. *Linux Kernel Development*. 4th ed., Addison-Wesley, 2022.
- McKusick, Marshall Kirk, and George Neville-Neil. *The Design and Implementation of the FreeBSD Operating System*. 3rd ed., Addison-Wesley, 2023.
- MITRE Corporation. *Common Vulnerabilities and Exposures (CVE) Database*. MITRE, 2024.
- National Institute of Standards and Technology (NIST). *Security and Privacy Controls for Information Systems and Organizations (SP 800-53 Rev. 5)*. NIST, updated 2024.
- OWASP Foundation. *OWASP Top 10 Web Application Security Risks 2024*. OWASP, 2024.
- Russinovich, Mark E., David A. Solomon, Alex Ionescu, and Pavel Yosifovich. *Windows Internals*. 8th ed., Microsoft Press, 2023.
- Saltzer, Jerome H., and Michael D. Schroeder. "The Protection of Information in Computer Systems." *Proceedings of the IEEE*, reprinted edition, 2022.
- Schneier, Bruce. *A Hacker's Mind: How the Powerful Bend Society's Rules, and How to Bend Them Back*. W. W. Norton, 2023.
- Schneier, Bruce. *Click Here to Kill Everybody: Security and Survival in a Hyper-connected World*. W. W. Norton, 2020.
- Shostack, Adam. *Threat Modeling: Designing for Security*. 2nd ed., Wiley, 2023.
- Silberschatz, Abraham, Peter B. Galvin, and Greg Gagne. *Operating System Concepts*. 11th ed., Wiley, 2024.
- Smith, Sean W., and John Marchesini. *The Craft of System Security*. Updated ed., Addison-Wesley, 2022.
- Stallings, William. *Operating Systems: Internals and Design Principles*. 10th ed., Pearson, 2024.
- Stallings, William, and Lawrie Brown. *Computer Security: Principles and Practice*. 5th ed., Pearson, 2024.
- Tanenbaum, Andrew S., and Herbert Bos. *Modern Operating Systems*. 5th ed., Pearson, 2022.
- The Linux Foundation. *2024 State of Open Source Security Report*. Linux Foundation Research, 2024.
- Verizon. *2024 Data Breach Investigations Report (DBIR)*. Verizon Enterprise, 2024.
- ACM. *Computing Curricula 2020: Paradigms for Global Computing Education*. Association for Computing Machinery, 2020.
- IEEE Computer Society. *Cybersecurity Trends and Emerging Threats Report*. IEEE Press, 2024.