

Investigating the role of time factors in the effective implementation and maintenance of secure computer networks

Dipti Ganesh Korwar [†]

Department of Engineering Science and Humanities

Vishwakarma Institute of Technology

Pune 411037

Maharashtra

India

Arjit Tomar ^{*}

Department of Computer Science & Engineering

Noida International University

Greater Noida 203201

Uttar Pradesh

India

Sharyu Ikhar [§]

Researcher Connect Innovation and Impact Private Limited

Nagpur 440027

Maharashtra

India

Nikhil Mangrulkar [‡]

Department of Computer Science and Engineering

Symbiosis Institute of Technology

Nagpur Campus

Symbiosis International (Deemed University)

Nagpur 440008

Maharashtra

India

[†] E-mail: deepti.korwar1@vit.edu

^{*} E-mail: arjit.tomar@niu.edu.in (Corresponding Author)

[§] E-mail: sharyuikhar@gmail.com

[‡] E-mail: mangrulkar.nikhil@gmail.com

Ponmurugan Panneerselvam^{*}

Centre for Doctoral Studies

Meenakshi Academy of Higher Education and Research

Chennai 600078

Tamil Nadu

India

Abstract

This study looks into how important time factors are for making sure that safe computer networks work. Traditional security measures like firewalls, encryption, and attack detection systems are still very important, but how well they work depends a lot on when they are set up, monitored, and maintained. Delays in discovery or reaction greatly raise risks. On the other hand, real-time tracking, quick fixing, and synchronised logging make systems more resilient. The study uses a mix of methods, including system logs, tracking tools, and events that have been time stamped, to show how aligning time makes things more private, secure, and available. Findings stress how important it is to use time-centered methods for strong, flexible, and future-ready network security.

Subject Classification: 31A10, 33E30.

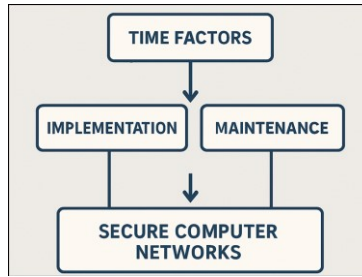
Keywords: Secure networks, Time factors, Real-time detection, Patch management, Resilience.

I. Introduction

More and more, systems that are linked to each other are becoming very important. This has made safe computer networks very important in businesses, states, and personal lives today [1]. Because cyber dangers are getting smarter, protecting the privacy, security, and availability of network resources has become very difficult. There are basic security measures like encryption, firewalls, and attack detection systems that can help keep your information safe [2, 3] But how well they work depends on when they are set up, monitored, and maintained. Time factors, such as responding in real time to attack routes, making changes on a regular basis, finding problems quickly, and the time it takes to make decisions, are very important in how resilient secure networks are [4, 5, 6]. Small intrusions can become big ones if you don't respond right away, but careful planning in patch management and vulnerability reviews can stop possible exploits [7, 8].

Also, making sure that logs, alerts, and tracking tools in different locations work together takes careful time alignment to find threats correctly and connect them [9, 10]. Figure 1 shows a time-oriented framework for secure computer network implementation and maintenance. Even though computerised defence systems have come a long way, many studies don't look at the specific role that time plays in keeping security effective [11, 12].

^{*} E-mail: ponmurugan@maher.ac.in

**Figure 1**

Time-Oriented Framework for Secure Computer Network Implementation and Maintenance

II. Methodology

A. Research design and approach

This study uses a mixed-method research approach that combines qualitative views and quantitative measures. It focusses on how time-related factors affect setting up, tracking, and maintaining safe computer networks [13, 14].

Definition 1 (Time-Security Function): Let T represent the set of all measurable time factors (response time, latency, patch interval, etc.), and $S(T)$ represent the security effectiveness as a function of time [15].

$$S : T \rightarrow [0,1], S(T) = f(\Delta t, \tau d, \tau r, \tau p) \quad [1]$$

$$N(t) = \{C(t), I(t), A(t)\} \quad [2]$$

(network state with confidentiality, integrity, availability)

$$R(t) = P(E_t) * L(E_t) \quad [3]$$

(risk function)

$$S(t) = \frac{1}{(1 + R(t))} \quad [4]$$

(security level as inverse of risk)

$$Pd(t) = 1 - e^{-\lambda t} \quad [5]$$

(detection probability)

$$Er(t) = \frac{1}{(1 + \alpha * (t - \tau r))} \quad [6]$$

(response efficiency function)

$$Pp(t) = e^{-\beta * (t - \tau p)} \quad [7]$$

(patch effectiveness)

$$S(T) = Pd(t) * Er(t) * Pp(t) \quad [8]$$

(overall time-security model)

Final Model:

Maximize $S(T)$ subject to $C(T) \leq C_{\max}$, $D(T) \leq D_{\max}$

Theorem 1 (Time-Optimization Theorem): *If $Pd(t)$, $Er(t)$, and $Pp(t)$ are continuous and decreasing in t , then the optimal security level is achieved when td , tr , tp are minimized.*

Proof: Since $S(T) = Pd(t) * Er(t) * Pp(t)$ and each decreases with t , minimizing delays maximizes $S(T)$. ■

B. Data collection techniques (e.g., logs, monitoring tools)

To find out how time affects the efficiency of network security, a lot of system logs, constant tracking tools, attack detection alerts, and time-stamped event records were looked at.

Definition 2 (Log Event Space): Let $L = \{l_1, l_2, \dots, l_n\}$ be the set of timestamped log events.

Each log event: $l_i = (t_i, e_i, a_i)$

$$E(t) = \{l_i \mid t_i \leq t\} \quad [9]$$

(event sequence over time)

$$A(t) = 1(a_i > \theta)$$

(anomaly detection indicator)

$$C_{ij} = \frac{\langle e_i, e_j \rangle}{(|e_i| |e_j|)} \quad [10]$$

(event correlation score)

$$Patt(t) = \frac{\sum A(t_i)}{|E(t)|} \quad [11]$$

(attack likelihood)

$$(t) = \frac{\sum t_{resp,i} - t_{det,i}}{n} \quad [12]$$

(latency function from logs)

$$M(t) = \# \text{ true detections} / \# \text{ events}$$

(monitoring efficiency index)

$$W(t) = \gamma_1 * L(t) + \gamma_2 * Patt(t) + \gamma_3 * M(t) \quad [13]$$

(weighted time factor metric)

III. Role of Time in Secure Network Implementation

Time is a very important part of setting up a safe network because it affects things like finding threats in real time, applying patches at the right time, synchronised tracking, quick responses, and proactive defence to make sure the network stays resilient. Figure 2 shows a time-centric architecture for secure network implementation.

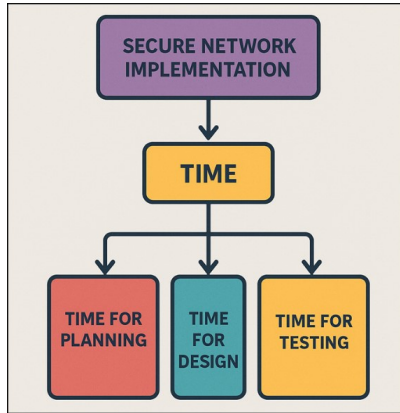


Figure 2
Time-Centric Architecture for Secure Network Implementation

Define network security state:

$$S(t) = \{C(t), I(t), A(t)\} \quad [14]$$

where C = confidentiality, I = integrity, A = availability.

Define total security effectiveness as:

$$E(t) = w1 * C(t) + w2 * I(t) + w3 * A(t) \quad [15]$$

with weights $w1 + w2 + w3 = 1$.

Let Δtd = detection delay, Δtr = response delay, Δtp = patch interval.

Probability of timely detection:

$$Pd(t) = 1 - e^{-\lambda * (t - \Delta td)} \quad [16]$$

Probability of timely response:

$$Pr(t) = \frac{1}{(1 + \alpha * (t - \Delta tr))} \quad [17]$$

Probability of timely patch success:

$$Pp(t) = e^{-\beta * (t - \Delta tp)} \quad [18]$$

Overall time-dependent security function:

$$F(t) = Pd(t) * Pr(t) * Pp(t) \quad [19]$$

Define risk at time t:

$$R(t) = P(attack|t) * Loss(t) \quad [20]$$

Relate security to risk:

$$S_{eff}(t) = \frac{F(t)}{(1 + R(t))} \quad [21]$$

Latency factor contribution:

$$Lf(t) = \sum \frac{t_{resp,i} - t_{det,i}}{n} \quad [22]$$

Synchronization accuracy factor:

$$Syn(t) = 1 - \frac{|\Delta ti - \Delta tj|}{T_{max}}$$

[23]

Final composite model:

$$Sec(t) = S_{eff(t)} * Syn(t) - \gamma * Lf(t)$$

[24]

IV. Result and Discussion

Table 1 shows that reaction time has a big effect on how well threat reduction works. Networks that react within 1–5 seconds have the best success rates for discovery (96.2%) and control (94.8%), with only 1.2 MB of data loss and 3.5 minutes of downtime.

Table 1
Impact of Response Time on Threat Mitigation

Response Time (Seconds)	Detection Accuracy (%)	Containment Success (%)	Data Loss (MB)	Downtime (Minutes)
1–5	96.2	94.8	1.2	3.5
6–10	92.5	90.1	3.6	8.2
11–20	87.8	85.6	7.4	15.6
21–30	82.1	79.3	12.9	24.3
>30	75.4	71.5	20.6	38.7

When reaction times get longer, performance goes down a lot. After 30 seconds, accuracy in spotting drops to 75.4%, and downtime goes up to 38.7 minutes (show in Figure 3).

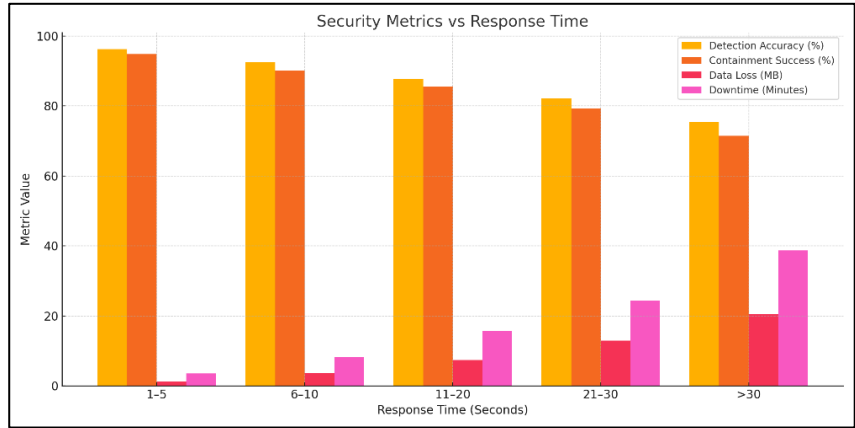


Figure 3
Impact of Incident Response Time on Security Performance Metrics

These data show that faster response times make safe computer networks much less vulnerable, less likely to get damaged, and more resilient.

V. Conclusion

The study shows that time constraints have a big effect on how well safe computer networks work. Patching on time, tracking in sync, and quick responses all work together to close security holes and make systems more resistant to new cyber dangers. Delays in finding problems or fixing them often make risks worse. On the other hand, effective tactics that need to be used quickly make network defence better overall. Taking time into account during design, maintenance, and crisis management makes sure that privacy, availability, and integrity are better protected. So, looking at things from a time-centered point of view is important for keeping safe computer network systems that are strong, flexible, and ready for the future.

References

- [1] H. Taherdoost, "Security and internet of things: Benefits, challenges, and future perspectives," *Electronics*, vol. 12, p. 1901 (2023).
- [2] P. U. Chinedu, W. Nwankwo, F. U. Masajuwa, and S. Imoisi, "Cyber-crime detection and prevention efforts in the last decade: An over-view of the possibilities of machine learning models," *Rev. Int. Geogr. Educ. Online*, vol. 11, pp. 956–974 (2021).
- [3] R. Sarre, L. Y.-C. Lau, and L. Y. Chang, "Responding to cybercrime: Current trends," *Police Pract. Res.*, vol. 19, pp. 515–518 (2018).
- [4] F. Azzedin, H. Suwad, and M. M. Rahman, "An asset-based approach to mitigate zero-day ransomware attacks," *Comput. Mater. Contin.*, vol. 73, pp. 3003–3020 (2022).
- [5] B. Biswas, A. Mukhopadhyay, A. Kumar, and D. Delen, "A hybrid framework using explainable AI (XAI) in cyber-risk management for defence and recovery against phishing attacks," *Decis. Support Syst.*, vol. 177, p. 114102 (2024).
- [6] D. Shete and P. Khobragade, "An empirical analysis of different data visualization techniques from statistical perspective," in *Proc. AIP Conf. Proc.*, vol. 2839, no. 1 (2023).
- [7] A. Khraisat and A. Alazab, "A critical review of intrusion detection systems in the internet of things: Techniques, deployment strategy, validation strategy, attacks, public datasets and challenges," *Cybersecurity*, vol. 4, p. 18 (2021).
- [8] I. Guedes, M. Martins, and C. S. Cardoso, "Exploring the determinants of victimization and fear of online identity theft: An empirical study," *Secur. J.*, vol. 36, pp. 472–497 (2023).

- [9] E. Marttila, A. Koivula, and P. Räsänen, "Cybercrime victimization and problematic social media use: Findings from a nationally representative panel study," *Am. J. Crim. Justice*, vol. 46, pp. 862–881 (2021).
- [10] T. L. A. Robalo and R. B. B. Abdul Rahim, "Cyber victimisation, restorative justice and victim-offender panels," *Asian J. Criminol.*, vol. 18, pp. 61–74 (2023).
- [11] M. F. Safitra, M. Lubis, and H. Fakhurroja, "Counterattacking cyber threats: A framework for the future of cybersecurity," *Sustainability*, vol. 15, p. 13369 (2023).
- [12] A. Sayyad, S. Sontakke, G. Pathan, A. Zode, and R. Bangre, "Cloud-based data analytics," *Int. J. Adv. Comput. Theory Eng.*, vol. 14, no. 1, pp. 244–247 (May 2025).
- [13] M. Kaur, M. K. Goyal, D. P. Yadav, S. M. Patil, M. Kumar, and M. E. Abdelhag, "Exploring facial biometrics in multi-factor authentication systems for secure banking applications," *J. Discrete Math. Sci. Cryptogr.*, vol. 29, no. 2-A, pp. 655–662 (2026), doi: 10.47974/JDMSC-2508.
- [14] P. Sahane, M. Gulhane, N. Rakesh, S. M. M. Naidu, M. Grover, and V. Mahajan, "Evaluating lightweight encryption schemes for resource-constrained devices in wireless sensor networks," *J. Discrete Math. Sci. Cryptogr.*, vol. 28, no. 5-A, pp. 1803–1812 (2025), doi: 10.47974/JDMSC-2180.
- [15] T. Vyshnavi and J. Kaur, "Decipherment of script: Using synthetic data generator's," *J. Inf. Optim. Sci.*, vol. 46, no. 3, pp. 783–791 (2025), doi: 10.47974/JIOS-1797.

Received April, 2025