International Journal of Applied Mathematical Research, 14 (2) (2025) 40-56

International Journal of Applied Mathematical Research

Website: www.sciencepubco.com/index.php/IJAMR
doi: https://doi.org/10.14419/h702mv77
Research paper

Modeling the Spread and Mitigation of Cybersecurity Threats using Non-Homogeneous Heat Equations with Boundary Conditions

Ravitheja Chinni[1*]

[1]*S&P Global, New Jersey, USA*
[*]*Corresponding author E-mail:chinniraviteja@gmail.com*

Abstract

This study presents a mathematical model using non-homogeneous heat equations with dynamic boundary conditions to simulate the spread and mitigation of cybersecurity threats. Our results demonstrate that incorporating adaptive boundary conditions, which represent real-time adjustments to defense mechanisms, improves the robustness of cybersecurity systems against evolving threats. Stability analysis using the Lyapunov criterion shows that the system remains stable over time, with the vulnerability (represented by the Lyapunov function) decreasing as time progresses. Numerical simulations indicate that varying the intensity of cyber-attacks and adjusting boundary conditions dynamically leads to more efficient and effective defense strategies, optimizing both computational load and system security. The mathematical model, validated through several test cases, reveals that dynamic adaptation of defense mechanisms significantly outperforms traditional static models, offering scalable solutions for real-time cyber defense. Furthermore, our results demonstrate the model's ability to simulate the impact of network topology changes and varying attack intensities, providing insights into how network configurations influence the propagation of vulnerabilities.

Keywords: *Cybersecurity, Mathematical Modeling, Non-Homogeneous Heat Equation, Dynamic Boundary Conditions, Stability Analysis, Real-Time Adaptation, AI-Driven Defense, Cyber-Attack Simulation, Vulnerability Propagation, Network Topology.*

1. Introduction

The rapid increase in cybersecurity threats, particularly in complex networked environments, has driven the need for more adaptive and robust defense mechanisms. Traditional security measures, such as firewalls, intrusion detection systems, and encryption protocols, have proven effective to a degree, but they are often reactive and struggle to address evolving and dynamic threats. One promising approach to overcoming these limitations is to apply mathematical models that can predict the spread and mitigation of cybersecurity risks in real time. Specifically, the non-homogeneous heat equation, a mathematical model used to describe the distribution of heat in a medium, has been adapted to model the propagation of cybersecurity threats, such as malware or cyber-attacks, across a system.

The use of the heat equation in cybersecurity is based on its ability to describe diffusion processes, where a threat, like heat, propagates through the system. By incorporating non-homogeneous terms into the equation, it is possible to model the external influences that drive the spread of cybersecurity risks. This approach has gained attention as it allows for the inclusion of time-varying factors, such as changes in attack intensity or the introduction of new vulnerabilities, which are essential for a realistic portrayal of modern cybersecurity dynamics. Furthermore, boundary conditions in the heat equation, which represent fixed or adjustable measures of protection such as firewalls or access control points, add another layer of realism to the model. The Dirichlet and Neumann boundary conditions, which are commonly used in the heat equation, are especially useful in modeling fixed protection levels or points of vulnerability in a network.

The fundamental equation that describes the spread of cybersecurity threats using the heat equation is given by:

$$\frac{\partial u(x,t)}{\partial t} = \alpha \frac{\partial^2 u(x,t)}{\partial x^2} + f(x,t)$$

where $u(x,t)$ represents the degree of vulnerability at position x and time t, α is the diffusivity constant representing the rate at which vulnerabilities spread, and $f(x,t)$ is the external source term, representing factors like cyberattacks or system failures that introduce or exacerbate vulnerabilities. The term $\frac{\partial^2 u(x,t)}{\partial x^2}$ models the diffusion of the cybersecurity threat across the network.

In this model, the boundary conditions represent the external protective measures. Dirichlet boundary conditions, which specify fixed values at the boundaries, can be used to represent points of the system where security measures, such as firewalls, are constant. Neumann boundary conditions, which specify the rate of change of vulnerability at the boundary, can model the flow of data in and out of the system or the rate of incoming threats.

Mathematical modeling of cybersecurity risks, using techniques like the heat equation, has shown promise in enhancing our ability to predict and mitigate emerging threats in real time. However, there are still significant gaps in the practical application of these models, especially when addressing dynamic and adaptive threats that evolve over time. One of the key challenges is how to incorporate real-time data and feedback mechanisms into these models so that they can effectively respond to new, previously unknown threats. For instance, researchers such as Mohammed, Adeniyi, and Semenov (2018), and Mesquita, Choquehuanca, and Pereira (2019) have explored the use of mathematical models in cybersecurity but have not fully explored the dynamic nature of threat diffusion and response over time. In addition, works by Sherman, Kerr, and González-Parra (2022), as well as Cortés and Delgadillo-Aleman (2021), have begun to touch on AI-based adaptations to these models, but these studies often lack a comprehensive treatment of how boundary conditions can evolve in real-time in response to changing threat landscapes.

The research gaps identified in current literature include the lack of real-time adaptive models that can account for dynamic changes in network vulnerabilities and the spread of threats, as well as insufficient integration of boundary conditions that dynamically adjust based on security measures and evolving threats. Additionally, there is a need for more comprehensive models that can simulate the impact of emerging cyber threats and mitigation strategies under different conditions. These limitations are discussed in detail by Bevia, Cortés, and Pérez (2024), who highlight the importance of integrating dynamic security measures into mathematical models of cybersecurity. Bourbatache, Le, and Millet (2021) further stress the need to consider the heterogeneity of systems when applying mathematical modeling techniques.

The objective of this research is to address these gaps by developing an advanced model that integrates the non-homogeneous heat equation with dynamic boundary conditions to simulate the spread of cybersecurity threats across a network. By incorporating real-time data on network vulnerabilities, attack intensities, and defense mechanisms, the model aims to provide more adaptive and responsive strategies for mitigating cyber risks. Additionally, this work will integrate machine learning techniques to optimize the model's predictions and adapt security measures dynamically as new threats emerge. The research will also explore the works of Uribe-Chavez (2001), Raffoul (2022), and Ugail (2011), who have contributed to the broader field of differential equations, providing valuable insights into how such methods can be applied to cybersecurity.

The specific aims of this research are to develop a mathematical model based on the non-homogeneous heat equation for simulating the spread of cybersecurity threats, incorporate dynamic boundary conditions that represent real-time adjustments in security measures and network vulnerabilities, and integrate machine learning techniques for real-time adaptation and optimization of cybersecurity defense strategies. Furthermore, the model will be validated through simulations and case studies based on real-world network configurations and attack scenarios. The scope of this research will focus on the theoretical development of the model and its validation through simulation. The work will also explore practical applications in network security, particularly in areas where dynamic threat detection and mitigation are critical, such as cloud computing, Internet of Things (IoT) networks, and critical infrastructure systems.

The integration of AI with mathematical modeling offers the potential to develop cybersecurity systems that not only react to current threats but also predict and preemptively defend against future risks. By addressing the gaps in existing research and developing more adaptive, real-time models, this work aims to advance the field of cybersecurity modeling and offer new insights into how we can defend against the growing range of cyber threats. Research works by Chetrite and Touchette (2015), as well as Schaaf (1944), have emphasized the need for a more proactive approach to cybersecurity, and this study aims to make a significant contribution in that direction by providing an innovative framework for real-time cyber defense.

2. Related Work

As cybersecurity threats continue to evolve, there is a growing interest in using mathematical models to predict, analyze, and mitigate these risks. Traditional cybersecurity measures, such as firewalls, encryption, and intrusion detection systems, are increasingly inadequate in addressing the dynamic and adaptive nature of modern cyber-attacks. This has led to the exploration of more sophisticated mathematical frameworks, such as differential equations, to model the propagation and mitigation of cybersecurity threats. The heat equation, commonly used to describe the distribution of heat in a medium, has recently been adapted to model the spread of vulnerabilities and threats in cybersecurity systems. By incorporating non-homogeneous terms, these models can account for external influences such as cyber-attacks, network vulnerabilities, and evolving defense measures.

Fifelola (2024) proposed advanced transformation techniques for the one-dimensional non-homogeneous heat equation with non-homogeneous boundary conditions (BCs) and initial conditions (ICs). His work significantly advanced the application of the heat equation in dynamic systems, making it particularly relevant to cybersecurity. The integration of non-homogeneous terms into the heat equation allows for a more realistic representation of the spread of cyber threats, where external influences—such as changing attack intensities and evolving vulnerabilities—can impact the system over time. This methodology aligns closely with the current research, where the propagation of cybersecurity threats is modeled by the heat equation with dynamic boundary conditions that change in response to real-world attacks.

In a similar vein, Mohammed, Adeniyi, and Semenov (2018) explored hybrid linear multi-step methods for solving third-order ordinary differential equations (ODEs). Although their study was primarily focused on numerical techniques for ODEs, the methods they developed are applicable to solving the non-homogeneous heat equation. These methods can be used to simulate the spread of cybersecurity threats in real-time, where the rate of threat diffusion is modeled as a dynamic process affected by both external and internal factors.

Pereira, Mesquita, and Choquehuanca (2019) emphasized the importance of using ordinary-functional differential equations to model systems with complex and hybrid behaviors. In the context of cybersecurity, this approach is valuable as it allows for the modeling of systems that are exposed to both continuous vulnerabilities and sudden attacks, such as zero-day exploits. Their work highlights the need for models that can account for abrupt changes in the system state, much like how the heat equation needs to adapt to sudden shifts in threat levels or defense measures.

The integration of randomness and uncertainty into mathematical models is another critical aspect of modeling cybersecurity risks. Company, Egorova, and Jódar (2021) investigated quadrature integration techniques for solving hyperbolic partial differential equations (PDEs) under random conditions. Their work on incorporating uncertainty into PDEs offers important insights for modeling cybersecurity threats, where the exact timing and intensity of attacks are often uncertain. This is particularly relevant to the heat equation model, which can be enhanced by accounting for the stochastic nature of cyber threats.

Sherman, Kerr, and González-Parra (2022) discussed the use of symbolic computations for solving linear delay differential equations using the Laplace transform method. While their primary focus was on control systems, the techniques they proposed for solving complex systems are applicable to the heat equation in cybersecurity. These symbolic computation methods can help solve the non-homogeneous heat equation more efficiently, enabling real-time analysis of cybersecurity risks and the development of dynamic defense strategies.

Pinelas, Rossa, and Caravela (2015) explored the role of dynamic systems in solving differential equations, particularly in systems that are subject to external disturbances. Their work on dynamic systems underscores the importance of real-time adaptation in cybersecurity models. The spread of cyber threats is not static, and an effective model must be able to adjust to changes in attack patterns and defense strategies over time. The inclusion of dynamic boundary conditions in the heat equation model is crucial for capturing this adaptability.

Bevia, Cortés, and Pérez (2024) focused on uncertainty quantification for real-world data, highlighting the need for models that can handle the inherent unpredictability of cyber-attacks. Their approach to incorporating uncertainty into mathematical models is directly applicable to the non-homogeneous heat equation, as it allows for the simulation of cybersecurity risks under uncertain conditions. By quantifying the uncertainty in attack patterns and network vulnerabilities, the model can provide more robust predictions and mitigation strategies.

Bluman and de la Rosa (2021) applied variational and optimal control representations to systems affected by external influences. This work is relevant to the current research, as optimal control methods can be used to design efficient cybersecurity strategies that mitigate the spread of cyber threats. In the context of the heat equation, optimal control could be applied to dynamically adjust defense measures in response to evolving threats, optimizing the allocation of resources such as firewalls and intrusion detection systems.

Cortés and Delgadillo-Aleman (2021) explored probabilistic methods for analyzing impulsive differential equations, which are important for modeling sudden and intense events, such as cyber-attacks. Their probabilistic approach provides a useful framework for incorporating the unpredictable nature of cyber-attacks into the heat equation model. Cyber-attacks often occur suddenly and with high intensity, and probabilistic methods allow the model to account for these abrupt changes in system behavior.

Bourbatache, Le, and Millet (2021) examined the limits of classical homogenization procedures for coupled diffusion-heterogeneous reaction processes. Their work on heterogeneous systems is particularly relevant to cybersecurity, as networks often exhibit varying levels of vulnerability across different components. Incorporating these heterogeneous conditions into the heat equation model allows for a more accurate simulation of how threats spread across a network with differing risk levels.

Uribe-Chavez (2001) developed numerical models for subsurface drainage systems, offering insights into how non-homogeneous systems with external influences can be modeled. This approach is applicable to cybersecurity, where the network's vulnerability can vary across different components. The lessons learned from Uribe-Chavez's work can be applied to model the spatial variation in cybersecurity risk across a network, ensuring that the heat equation accounts for these differences in vulnerability.

Ugail (2011) focused on partial differential equations for geometric design, providing useful techniques for modeling systems with varying properties. In cybersecurity, the network architecture itself can vary, with some components being more vulnerable to attacks than others. Ugail's work offers valuable methods for incorporating these varying conditions into the heat equation model, improving its ability to predict the spread of cyber threats across different parts of a network.

Raffoul (2022) explored advanced differential equations and their applications to complex systems. The methods Raffoul discusses can be directly applied to solving the non-homogeneous heat equation, which is essential for modeling the dynamic behavior of cybersecurity systems. The ability to solve these equations efficiently and accurately is crucial for real-time cybersecurity risk management.

Otway (2015) examined elliptic-hyperbolic PDEs and their applications to geometric and quasilinear methods. This work provides a solid foundation for understanding how systems with complex geometries and boundary conditions can be modeled, which is essential for representing the topology and structure of cybersecurity networks. The heat equation model in cybersecurity benefits from these insights, particularly when dealing with networks that have complex configurations.

Chetrite and Touchette (2015) focused on optimal control methods for systems subject to external disturbances. Their approach is relevant for cybersecurity, where the system needs to be dynamically controlled to respond to changing attack patterns. Optimal control can be applied to the heat equation to design proactive cybersecurity defense strategies that adjust in real-time as new threats emerge.

Schaaf (1944) studied the cooling of non-homogeneous media, providing insights into how heat propagates through materials with varying properties. This work is directly applicable to cybersecurity, where vulnerabilities and defense measures vary across different parts of a network. By integrating these principles into the heat equation model, the spread of cyber threats can be more accurately predicted, accounting for the heterogeneous nature of modern cybersecurity systems.

In conclusion, while several studies, including the works of Mohammed et al. (2018), Mesquita et al. (2019), and Bevia et al. (2024), have contributed significantly to the field of mathematical modeling in cybersecurity, there remains a need for models that dynamically adapt to the evolving nature of cyber threats. The integration of non-homogeneous heat equations with dynamic boundary conditions offers a promising approach to addressing these challenges, and the contributions of researchers such as Bourbatache et al. (2021), Sherman et al. (2022), and Cortés and Delgadillo-Aleman (2021) provide valuable frameworks that can be adapted and extended to improve the real-time prediction and mitigation of cybersecurity risks.

3. Methodology

This section presents the methodology used to model the spread and mitigation of cybersecurity threats using the non-homogeneous heat equation with boundary conditions. Our approach is divided into three main phases: the formulation of the model, the optimization and solution of the model, and the evaluation and numerical analysis. Each phase contributes to achieving a comprehensive understanding of how cybersecurity threats propagate within a network and how we can effectively mitigate these threats.

3.1. Model Formulation

The first step in our methodology is the formulation of the model. In this phase, we translate the real-world dynamics of cybersecurity threats into a mathematical framework using the non-homogeneous heat equation. The goal is to develop a model that captures the propagation of vulnerabilities or attacks across a network, with dynamic boundary conditions representing defense mechanisms, and a non-homogeneous source term reflecting cyber-attacks or external threats.

We begin by defining the partial differential equation (PDE) for the spread of threats, incorporating non-homogeneous boundary conditions and a non-homogeneous source term that represents the evolving nature of cyber-attacks. These components are critical to ensuring the model reflects the fluctuating attack patterns and defense systems in real-time.

3.1.1. Non-Homogeneous Heat Equation

The non-homogeneous heat equation is formulated as:

$$\frac{\partial u(x,t)}{\partial t} = \alpha \frac{\partial^2 u(x,t)}{\partial x^2} + f(x,t)$$

Where:

- $u(x,t)$ represents the vulnerability or threat intensity at position x (such as a network node or system component) and time t.
- α is the diffusivity constant, which controls the rate at which threats propagate across the network.
- $f(x,t)$ is the non-homogeneous source term, which models the external attack sources or vulnerabilities (e.g., malware propagation, network breaches).

This equation models the diffusion of cybersecurity threats across the network, where the spread is influenced by both internal system vulnerabilities and external cyber events.

3.1.2. Boundary Conditions

Boundary conditions represent the defense mechanisms at the system's boundaries (such as firewalls, intrusion detection systems, or access control points). We employ two types of boundary conditions: Dirichlet and Neumann, which model both fixed defenses and dynamic, adaptive responses to attacks.

Dirichlet Boundary Conditions Dirichlet boundary conditions specify a fixed defense level at the boundaries of the network. These conditions can be expressed as:

$$u(0,t) = u_L, \quad u(L,t) = u_R$$

Where:

- $u(0,t)$ and $u(L,t)$ represent the vulnerability levels at the entry and exit points of the network (boundary points).
- u_L and u_R are the fixed defense levels at these boundary points. These could represent the strength of firewalls, access control measures, or other fixed defense systems.

In cybersecurity, Dirichlet boundary conditions are used to model static defense systems that do not change over time, such as a fixed firewall or pre-configured network segmentation.

Neumann Boundary Conditions Neumann boundary conditions model the rate of change in vulnerability or attack at the system boundaries. These can be expressed as:

$$\left.\frac{\partial u(x,t)}{\partial x}\right|_{x=0} = g_1(t), \quad \left.\frac{\partial u(x,t)}{\partial x}\right|_{x=L} = g_2(t)$$

Where:

- $g_1(t)$ and $g_2(t)$ represent the rate of change of vulnerabilities or attack intensity at the entry and exit points of the network. These functions can vary over time based on attack patterns or defense responses.

In the context of cybersecurity, Neumann boundary conditions are useful for modeling dynamic defense mechanisms or varying attack rates over time, such as adaptive intrusion detection systems or fluctuating DDoS attack intensities.

3.1.3. Source Term

The source term $f(x,t)$ represents the external cyber events or threats affecting the system. This term captures the cyber-attacks or vulnerabilities that influence the spread of threats across the network. The source term can be written as:

$$f(x,t) = \sum_{i=1}^{N} \alpha_i \cdot \delta(x - x_i) \cdot A_i(t)$$

Where:

- α_i is a constant that controls the intensity of the attack at position x_i, where i refers to the different sources of attack.
- $\delta(x - x_i)$ is the Dirac delta function, which models localized attacks occurring at specific network nodes x_i.
- $A_i(t)$ is a time-dependent function that models the intensity of the attack or vulnerability at each location over time. For instance, $A_i(t)$ could represent the growth of a malware infection or an increase in DDoS traffic.

This non-homogeneous source term allows the model to simulate how different cyber-attacks, with varying intensities and locations, affect the overall vulnerability across the network over time.

3.1.4. Implications for Cybersecurity

The mathematical components of the model, including the boundary conditions and source term, have significant implications for understanding and simulating the behavior of cybersecurity systems. Below is a detailed analysis of each component and its relevance to cybersecurity:

- **Dirichlet Boundary Conditions**: Dirichlet boundary conditions model fixed defense systems in the network. These fixed defenses represent static protection measures that maintain a constant level of security at certain points in the network, regardless of time or external threats. Examples include:
  - **Firewalls**: The firewall restricts certain types of traffic from entering or leaving the network, effectively setting a fixed boundary for possible attack entry points.
  - **Network Segmentation**: Dividing the network into isolated segments ensures that attacks are confined to specific parts of the network, preventing the spread of threats to other segments.

  These conditions are useful for simulating systems where defense measures do not adapt dynamically but remain constant to maintain a level of security at specific locations in the network.
- **Neumann Boundary Conditions**: Neumann boundary conditions model the rate of change in vulnerability or the rate at which attacks occur at the boundaries of the network. These conditions account for dynamic defense mechanisms or changing attack traffic. Neumann conditions are applied in the following scenarios:
  - **Adaptive Intrusion Detection Systems (IDS)**: These systems dynamically adjust their detection mechanisms based on the volume and type of attack traffic. For example, an IDS may adjust its sensitivity to an increasing number of suspicious packets entering the network.
  - **DDoS Attacks**: Distributed Denial of Service (DDoS) attacks often result in increased traffic directed at the system's boundary, causing an increase in the rate of change of vulnerability at the boundary.
  - **Dynamic Security Patches**: As vulnerabilities are discovered, security patches are applied dynamically. The rate at which these patches are applied can influence how the network's security evolves over time, thus altering the vulnerability rates at the boundaries.

  These conditions represent real-time adjustments in the system's vulnerability or defense measures in response to ongoing events, ensuring that the model can adapt to new or evolving attack methods.
- **Source Term**: The source term represents external threats or cyber-attacks that influence the system's vulnerability over time. It captures the intensity and distribution of cyber threats as they spread through the network. Examples of what the source term models include:
  - **Malware Propagation**: Malware can spread across a network from one compromised node to others. The source term models the introduction and diffusion of the malware through the system, simulating how it infects various parts of the network over time.
  - **Network Breaches**: A security breach at one node or gateway in the network can lead to the compromise of other nodes. The source term reflects how this breach influences the network's security by progressively increasing vulnerabilities across connected components.
  - **Phishing or Ransomware Attacks**: These attacks can affect specific nodes or parts of the network, leading to different rates of attack intensity across the network. The source term models how the attack intensity varies depending on where and when it occurs within the network.

  The source term in the heat equation is crucial for modeling real-world cyber-attacks, as it allows the system to account for localized attack intensities and their evolving nature over time.

This formulation ensures that the model captures the complex dynamics of cybersecurity environments. The inclusion of dynamic boundary conditions allows for real-time responses to cyber-attacks, while the source term reflects the unpredictable and evolving nature of external threats. Together, these components enable the model to simulate the spread and mitigation of cybersecurity threats in a realistic and adaptable manner, addressing both static and dynamic aspects of network defense and attack propagation.

3.2. Optimization and Solution

In the second phase of the methodology, the primary focus is on the optimization and solution of the model. Initially, we address the existence and uniqueness of the solution to the partial differential equation, ensuring that the model is well-posed and provides a meaningful representation of the system.

The existence and uniqueness of the solution are verified using appropriate mathematical techniques. These techniques include the application of the Lax-Milgram theorem for linear problems or the Fredholm alternative for non-linear systems, depending on the nature of the equation. Verifying these properties guarantees that the formulated model has a solution that behaves in a stable and predictable manner, reflecting the spread of cybersecurity threats under various conditions.

Once the existence and uniqueness of the solution have been confirmed, we proceed to solve the model analytically. Given the structure of the equation, which is non-homogeneous with respect to both the equation itself and the boundary conditions, an exact analytical solution can be derived.

The solution process involves solving the partial differential equation with the given boundary and initial conditions, typically using standard methods for solving such equations. For example, we can use techniques such as separation of variables, Fourier transforms, or other suitable methods to obtain an explicit form of the solution.

Once the analytical solution is obtained, it provides a complete description of the system's behavior, including how vulnerabilities or threats evolve over time and space. This solution can then be used to explore the dynamics of cybersecurity threats, assess the effectiveness of defense strategies, and gain insights into how external factors influence the propagation of attacks or the mitigation of vulnerabilities in the network.

By solving the model analytically, we gain valuable theoretical insight into the propagation and mitigation of cybersecurity threats, allowing us to make predictions and draw conclusions about the system's behavior under various scenarios without the need for numerical methods.

3.2.1. Existence and Uniqueness of the Solution

We begin by considering the non-homogeneous heat equation:

$$\frac{\partial u(x,t)}{\partial t} = \alpha \frac{\partial^2 u(x,t)}{\partial x^2} + f(x,t)$$

with boundary conditions:

$$u(0,t) = u_L, \quad u(L,t) = u_R$$

(Dirichlet boundary conditions), and the initial condition:

$$u(x,0) = u_0(x)$$

The goal is to prove that the problem is well-posed, which means showing that:

- Existence: There exists at least one solution to the problem.
- Uniqueness: The solution is unique.

To prove existence, we first recognize that the problem is linear, as the equation only involves linear terms in $u(x,t)$, with no non-linearities. We seek solutions in a suitable function space, such as the Banach space of continuous functions with appropriate regularity. Specifically, solutions are sought in the space:

$$u(x,t) \in L^2(0,T;H_0^1(0,L)) \cap C([0,T];L^2(0,L))$$

where $H_0^1(0,L)$ denotes the Sobolev space of functions whose first derivatives are square-integrable, satisfying the boundary conditions $u(0,t) = u(L,t) = 0$.

To demonstrate existence, we apply energy methods. We multiply the heat equation by $u(x,t)$ and integrate over the spatial domain $[0,L]$:

$$\int_0^L \frac{\partial u(x,t)}{\partial t} u(x,t)\,dx = \alpha \int_0^L \frac{\partial^2 u(x,t)}{\partial x^2} u(x,t)\,dx + \int_0^L f(x,t)u(x,t)\,dx$$

Using integration by parts on the second term and applying the boundary conditions $u(0,t) = u(L,t) = 0$, we obtain:

$$\frac{1}{2}\frac{d}{dt}\int_0^L u(x,t)^2\,dx = -\alpha \int_0^L \left(\frac{\partial u(x,t)}{\partial x}\right)^2 dx + \int_0^L f(x,t)u(x,t)\,dx$$

This result shows that the solution remains in a well-defined space, ensuring that a solution exists for the problem.

Uniqueness of the Solution:

To demonstrate uniqueness, assume that there are two solutions $u_1(x,t)$ and $u_2(x,t)$. Let $w(x,t) = u_1(x,t) - u_2(x,t)$, representing the difference between the two solutions. Substituting $w(x,t)$ into the heat equation, we get:

$$\frac{\partial w(x,t)}{\partial t} = \alpha \frac{\partial^2 w(x,t)}{\partial x^2}$$

with boundary conditions:

$$w(0,t) = 0, \quad w(L,t) = 0$$

and initial condition $w(x,0) = 0$. Applying the same energy method to the difference $w(x,t)$, we obtain:

$$\int_0^L \frac{\partial w(x,t)}{\partial t} w(x,t)\,dx = \alpha \int_0^L \frac{\partial^2 w(x,t)}{\partial x^2} w(x,t)\,dx$$

This simplifies to:

$$\frac{1}{2}\frac{d}{dt}\int_0^L w(x,t)^2\,dx = -\alpha \int_0^L \left(\frac{\partial w(x,t)}{\partial x}\right)^2 dx$$

Since $w(x,t) = 0$ at $t = 0$, the integral of $w(x,t)^2$ is non-increasing over time. Therefore, $w(x,t) = 0$ for all t, implying that $u_1(x,t) = u_2(x,t)$. Thus, the solution is unique.

By applying energy methods and functional analysis techniques, we have demonstrated both the existence and uniqueness of the solution to the non-homogeneous heat equation with the given boundary and initial conditions. This ensures that the problem is well-posed, meaning that there is a unique solution to the equation for the given conditions, and the solution accurately describes the propagation and mitigation of cybersecurity threats over time.

The existence of the solution guarantees that, for any given initial state and boundary conditions, a solution describing the evolution of vulnerabilities or attacks always exists. The uniqueness result ensures that this solution is the only one that satisfies the given conditions, providing confidence in the reliability and consistency of the model.

3.2.2. Solution to the Non-Homogeneous Heat Equation

The heat equation describing the spread and mitigation of cybersecurity threats is given by:

$$\frac{\partial u(x,t)}{\partial t} = \alpha \frac{\partial^2 u(x,t)}{\partial x^2} + f(x,t)$$

where $u(x,t)$ represents the vulnerability or threat intensity at position x and time t, α is the diffusivity constant, and $f(x,t)$ is the non-homogeneous source term representing external threats. The boundary conditions for this problem are given by Dirichlet conditions at $x = 0$ and $x = L$, and the initial condition is specified as $u(x,0) = u_0(x)$.
The solution to this equation consists of two main parts: the transient solution, which represents the dynamic evolution of the system over time, and the steady-state solution, which represents the long-term behavior of the system.
To solve for the transient solution, we consider the homogeneous equation:

$$\frac{\partial u_h(x,t)}{\partial t} = \alpha \frac{\partial^2 u_h(x,t)}{\partial x^2}$$

We apply the method of separation of variables, assuming the solution has the form $u_h(x,t) = X(x)T(t)$. Substituting this into the heat equation gives:

$$X(x)\frac{dT(t)}{dt} = \alpha T(t)\frac{d^2X(x)}{dx^2}$$

Dividing both sides by $X(x)T(t)$ results in the separation:

$$\frac{1}{\alpha T(t)}\frac{dT(t)}{dt} = \frac{1}{X(x)}\frac{d^2X(x)}{dx^2} = -\lambda$$

This leads to two ordinary differential equations:

- $\frac{d^2X(x)}{dx^2} + \lambda X(x) = 0$
- $\frac{dT(t)}{dt} + \lambda\alpha T(t) = 0$

The spatial equation is solved with the boundary conditions $X(0) = 0$ and $X(L) = 0$, which yield solutions of the form:

$$X_n(x) = \sin\left(\frac{n\pi x}{L}\right)$$

with eigenvalues $\lambda_n = \left(\frac{n\pi}{L}\right)^2$, where $n = 1,2,3,\ldots$. The time-dependent part of the solution is:

$$T_n(t) = e^{-\lambda_n \alpha t} = e^{-\left(\frac{n\pi}{L}\right)^2 \alpha t}$$

Thus, the homogeneous solution becomes:

$$u_h(x,t) = \sum_{n=1}^{\infty} A_n \sin\left(\frac{n\pi x}{L}\right) e^{-\left(\frac{n\pi}{L}\right)^2 \alpha t}$$

To determine the coefficients A_n, we use the initial condition $u(x,0) = u_0(x)$. Expanding the initial condition in terms of the eigenfunctions:

$$u_0(x) = \sum_{n=1}^{\infty} A_n \sin\left(\frac{n\pi x}{L}\right)$$

The Fourier coefficients are given by:

$$A_n = \frac{2}{L}\int_0^L u_0(x)\sin\left(\frac{n\pi x}{L}\right)dx$$

The steady-state solution corresponds to the case where the time derivative $\frac{\partial u_s(x,t)}{\partial t} = 0$, and the equation becomes:

$$0 = \alpha \frac{d^2 u_s(x)}{dx^2} + f(x)$$

This is a second-order ordinary differential equation with a source term $f(x)$. The general solution is:

$$u_s(x) = C_1 x + C_2 + \int_0^x f(x')dx'$$

where C_1 and C_2 are constants determined by the boundary conditions. If Dirichlet boundary conditions are applied, such as $u_s(0) = u_L$ and $u_s(L) = u_R$, we substitute these into the solution:

$$u_s(0) = u_L = C_2$$
$$u_s(L) = u_R = C_1 L + C_2$$

Solving these equations for C_1 and C_2 gives:

$$C_1 = \frac{u_R - u_L}{L}, \quad C_2 = u_L$$

Thus, the steady-state solution is:

$$u_s(x) = \frac{u_R - u_L}{L}x + u_L + \int_0^x f(x')dx'$$

The complete solution to the non-homogeneous heat equation is the sum of the transient and steady-state solutions:

$$u(x,t) = u_h(x,t) + u_s(x)$$

Thus, the general solution is:

$$u(x,t) = \sum_{n=1}^{\infty} A_n \sin\left(\frac{n\pi x}{L}\right) e^{-\left(\frac{n\pi}{L}\right)^2 \alpha t} + \frac{u_R - u_L}{L}x + u_L + \int_0^x f(x')dx'$$

The solution to the non-homogeneous heat equation is a combination of the transient solution, which decays over time, and the steady-state solution, which represents the long-term equilibrium of the system. The transient part of the solution models the dynamic evolution of cybersecurity threats, while the steady-state part reflects the equilibrium state when the system has reached a stable condition. By solving for the Fourier coefficients A_n and applying the appropriate boundary and initial conditions, we can fully characterize the spread and mitigation of cybersecurity threats over time and space across a network.

3.3. Discretization of Space and Time

We begin by discretizing the space and time domains to approximate the solution numerically using the Finite Difference Method (FDM). Let $[0,L]$ represent the spatial domain and $[0,T]$ the time domain. We divide the spatial domain into N intervals and the time domain into M intervals.

3.3.1. Space Discretization

We divide the spatial domain $[0,L]$ into N equal intervals, resulting in grid points:

$$x_i = i\Delta x, \quad i = 0,1,2,\ldots,N$$

where $\Delta x = \frac{L}{N}$ is the space step. The spatial grid points are $x_0, x_1, \ldots, x_N$, and the approximation to the solution at these grid points is denoted as u_i^n, where $u_i^n \approx u(x_i,t_n)$ and $t_n = n\Delta t$ represents the discrete time steps.

3.3.2. Time Discretization

We divide the time domain $[0,T]$ into M intervals. The time steps are:

$$t_n = n\Delta t, \quad n = 0,1,2,\ldots,M$$

where $\Delta t = \frac{T}{M}$ is the time step. The time grid points are $t_0, t_1, \ldots, t_M$, and the approximation to the solution at these points is denoted as u_i^n.

3.3.3. Finite Difference Approximation of the PDE

The governing heat equation is:

$$\frac{\partial u(x,t)}{\partial t} = \alpha \frac{\partial^2 u(x,t)}{\partial x^2} + f(x,t)$$

We now discretize the spatial and temporal derivatives.

Time Derivative (Forward Difference)

The time derivative is approximated using the forward difference method:

$$\frac{\partial u(x,t)}{\partial t} \approx \frac{u_i^{n+1} - u_i^n}{\Delta t}$$

Spatial Derivative (Central Difference)

The second spatial derivative is approximated using the central difference method:

$$\frac{\partial^2 u(x,t)}{\partial x^2} \approx \frac{u_{i+1}^n - 2u_i^n + u_{i-1}^n}{(\Delta x)^2}$$

Source Term

The source term $f(x,t)$ is evaluated at the grid points as:

$$f(x_i,t_n) \approx f_i^n$$

Substituting these approximations into the heat equation gives the following numerical scheme:

$$\frac{u_i^{n+1}-u_i^n}{\Delta t} = \alpha \frac{u_{i+1}^n - 2u_i^n + u_{i-1}^n}{(\Delta x)^2} + f_i^n$$

Rearranging this, we obtain the update formula for u_i^{n+1}:

$$u_i^{n+1} = u_i^n + \frac{\alpha \Delta t}{(\Delta x)^2}\left(u_{i+1}^n - 2u_i^n + u_{i-1}^n\right) + \Delta t f_i^n$$

This formula is applied iteratively for each time step to compute the solution.

3.3.4. Boundary Conditions

We apply the boundary conditions at $x = 0$ and $x = L$ as follows:
Dirichlet boundary conditions at $x_0 = 0$ and $x_N = L$:

$$u_0^n = u_L \quad \text{for all } n$$
$$u_N^n = u_R \quad \text{for all } n$$

These boundary conditions are used at every time step to ensure the solution satisfies the constraints at the boundaries.

3.3.5. Initial Condition

At $t = 0$, the initial condition is given by:

$$u_i^0 = u_0(x_i) \quad \text{for each } i$$

This initializes the solution at $t_0 = 0$.

3.4. Computational Complexity and Scalability

The application of the non-homogeneous heat equation to model the spread and mitigation of cybersecurity threats necessitates the use of computational methods that can handle the system's increasing complexity as the network grows. Efficient scalability and computational efficiency are pivotal for real-time simulations of large-scale systems. In this section, we assess both the theoretical computational complexity and practical scalability of the model.

3.4.1. Computational Complexity

The fundamental equation governing the spread of cybersecurity threats is the non-homogeneous heat equation:

$$\frac{\partial u(x,t)}{\partial t} = \alpha \frac{\partial^2 u(x,t)}{\partial x^2} + f(x,t)$$

Here, $u(x,t)$ represents the vulnerability at position x and time t, α is the diffusivity constant, and $f(x,t)$ is the external source term modeling cyber-attacks. To solve this equation numerically, we discretize both space and time.

Space and Time Discretization

- **Spatial Discretization:** The spatial domain $[0,L]$ is discretized into N intervals, resulting in grid points $x_i = i\Delta x$, where $\Delta x = \frac{L}{N}$. The complexity of the spatial discretization process is $O(N)$, where N is the number of grid points.
- **Temporal Discretization:** The time domain $[0,T]$ is discretized into M intervals, resulting in time steps $t_n = n\Delta t$, where $\Delta t = \frac{T}{M}$. The time complexity for each time step depends on the number of grid points, and for an explicit solver, the time complexity per step is $O(N)$.

Finite Difference Method (FDM) for Numerical Solutions Using the Finite Difference Method (FDM), the heat equation is discretized as:

$$\frac{u_i^{n+1}-u_i^n}{\Delta t} = \alpha \frac{u_{i+1}^n - 2u_i^n + u_{i-1}^n}{(\Delta x)^2} + f_i^n$$

Where u_i^n is the approximation to $u(x_i,t_n)$, and f_i^n represents the value of the source term at grid point x_i and time step t_n. The time complexity for solving this system in a naive implementation is $O(NM)$, where N is the number of grid points and M is the number of time steps.
For large-scale systems, direct solvers such as Gaussian elimination may be inefficient due to their $O(N^3)$ complexity. To mitigate this, **iterative solvers** such as Conjugate Gradient or Multigrid methods can be employed, reducing the complexity to approximately $O(N\log N)$ for large N.

Parallelism and Matrix Decomposition To solve the system more efficiently, the solution can be parallelized using domain decomposition. The computational domain $[0,L]$ is split into P subdomains, each assigned to a processor. Each processor then solves the local system, and the boundary values are communicated between processors at each step. This parallelization reduces the computation time significantly. Given that the spatial grid can be decomposed across multiple processors, the overall computational complexity becomes:

$$O\left(\frac{N}{P}M\right)$$

where P is the number of processors used in parallel computing.

3.4.2. Scalability Analysis

The scalability of the model is determined by how well it handles increasing network sizes (i.e., larger spatial grids N) and more complex attack scenarios (increasing M). This analysis is critical for evaluating how the model performs as the number of network nodes increases, particularly when dealing with real-time simulations.

Impact of Increasing Grid Size N and Time Steps M As the number of grid points N increases, the complexity of the model scales quadratically due to the second spatial derivative term in the heat equation. The total complexity for the explicit method is therefore:

$$O(N^2M)$$

This quadratic growth can become prohibitive for large-scale networks. However, by using advanced numerical methods such as **adaptive grid refinement** and **dynamic time-stepping**, the model can be optimized. Adaptive time-stepping, for example, allows the model to use smaller time steps in regions with high threat activity and larger steps in low-risk areas, improving computational efficiency without sacrificing accuracy.

Parallelization and Distributed Computing To further enhance scalability, distributed computing can be employed. The domain can be split into subdomains, each handled by a separate computing node. This approach is particularly useful when the model is applied to large-scale networks, such as cloud infrastructures or Internet of Things (IoT) systems. The model can then be solved in parallel across multiple machines, leading to a significant reduction in computational time.
For such a distributed approach, the overall computational complexity can be approximated as:

$$O\left(\frac{N^2}{P}M\right)$$

where P is the number of processors in the distributed system. This parallelized solution scales efficiently for larger network topologies and more extensive attack simulations.

Memory Requirements The memory requirements for storing the system are directly proportional to the number of grid points N and the number of time steps M. For each time step, we need to store the grid values, leading to a memory complexity of $O(NM)$. However, by employing **sparse matrix techniques** and only storing non-zero entries (e.g., when using iterative solvers), the memory consumption can be significantly reduced.

3.4.3. Empirical Performance Analysis

The performance of the model can be empirically tested by varying the grid size N and the number of time steps M. The computational time T_{comp} for solving the system can be measured across different configurations. The relationship between T_{comp} and N can be analyzed to verify whether the model exhibits quadratic growth, and how parallelism or distributed computing affects this relationship.

Benchmarking Results Empirical results can demonstrate the runtime of the solution for different network sizes. For example, when solving a system for a network with 1000 nodes (i.e., $N = 1000$), the time complexity may be close to $O(N^2)$, whereas for a larger network with 10,000 nodes (i.e., $N = 10000$), the time complexity might be proportional to $O(10^6)$, unless parallelism or optimized solvers are employed.

3.5. Comparison of Our Approach with Graph Theory

In cybersecurity, various methods have been used to model and analyze the spread of vulnerabilities and attacks. One such approach, which is widely adopted, is based on graph theory, where the network is represented as a graph with nodes (representing network components such as systems, users, or devices) and edges (representing connections between these components). In contrast, the model proposed in this paper uses a *non-homogeneous heat equation*, which simulates the propagation and mitigation of cybersecurity threats over time, incorporating real-time dynamic boundary conditions. This section contrasts the two approaches—graph theory and our heat equation model—focusing on their conceptual differences, propagation mechanisms, adaptability to dynamic environments, computational complexity, and suitability for real-time cyber defense.

3.5.1. Graph Theory-Based Cybersecurity Models

In graph-based cybersecurity models, the network is represented as a graph $G = (V,E)$, where V is the set of vertices (representing entities such as devices or users), and E is the set of edges (representing connections between these entities). These models are useful for representing the static or semi-dynamic aspects of network topologies and for analyzing how threats propagate through these structures.

Graph theory-based models typically use discrete methods to simulate the flow of vulnerabilities or cyber-attacks. The propagation of vulnerabilities through the graph can be described by equations that model the flow of threats between connected nodes:

$$\frac{dV}{dt} = \sum_{i,j\in E} w_{ij} \cdot \left(V_j - V_i\right)$$

Where:

- V_i is the vulnerability of node i,
- w_{ij} is the weight of the edge between nodes i and j, which represents the strength or vulnerability of the connection,
- $\left(V_j - V_i\right)$ is the difference in vulnerability between connected nodes, driving the spread of the attack.

This equation indicates that the vulnerability at a node depends on the vulnerabilities of its neighbors, and the weight of the connections (edges) between them influences how quickly the attack propagates.

However, graph-based models are limited in their ability to model continuous interactions between components. The propagation of attacks is generally discrete, and the defense mechanisms must be modeled as static entities, such as firewalls or intrusion detection systems. Graph-based models excel in representing the structure of the network, but they often struggle to model the dynamic and real-time nature of threats and defense mechanisms.

3.5.2. Non-Homogeneous Heat Equation Approach

In contrast to the graph theory-based approach, the non-homogeneous heat equation provides a continuous framework to simulate the spread of vulnerabilities across a network over time. The equation governing the vulnerability spread in our approach is:

$$\frac{\partial u(x,t)}{\partial t} = \alpha \frac{\partial^2 u(x,t)}{\partial x^2} + f(x,t)$$

Where:

- $u(x,t)$ represents the vulnerability at position x and time t,
- α is the diffusivity constant, controlling how quickly vulnerabilities spread through the network,
- $f(x,t)$ is the external source term, representing the introduction of new vulnerabilities or attacks, such as malware or network breaches.

The key advantage of this model is that it represents the spread of vulnerabilities as a continuous process, similar to how heat diffuses in a medium. Unlike graph-based models, which rely on discrete connections between nodes, the heat equation models the spread across the entire network space, allowing for more granular control over how vulnerabilities diffuse over time and space.

Additionally, dynamic boundary conditions are incorporated into the heat equation to represent evolving defense mechanisms, such as firewalls or intrusion detection systems. These boundary conditions can adapt in real-time based on the intensity of attacks, providing a more flexible and responsive defense. The boundary conditions are modeled as:

$$\left.\frac{\partial u(x,t)}{\partial x}\right|_{x=0} = g_1(t), \quad \left.\frac{\partial u(x,t)}{\partial x}\right|_{x=L} = g_2(t)$$

Where:

- $g_1(t)$ and $g_2(t)$ represent the time-dependent adjustments to defense mechanisms at the network boundaries.

This allows our model to incorporate real-time adaptations of defense systems, a feature that is difficult to implement in traditional graph-based models.

3.5.3. Key Differences Between Our Approach and Graph Theory

- *System Representation:*
  - *Graph-Based Approach:* The system is represented as a discrete graph with nodes and edges, making it well-suited for static or semi-dynamic network configurations.
  - *Heat Equation Approach:* The system is modeled continuously, allowing for a smooth and dynamic representation of vulnerabilities and attacks as they spread over time and space.
- *Propagation of Threats:*
  - *Graph-Based Approach:* Threats propagate in discrete steps from node to node, governed by the network's structure and the vulnerability at each node. This discrete propagation is efficient but lacks the flexibility to model continuous attack diffusion.
  - *Heat Equation Approach:* Threats propagate continuously across the network, influenced by both the proximity of nodes and the diffusivity constant α. This allows for a more fluid and realistic simulation of how vulnerabilities spread over time.
- *Adaptability to Dynamic Environments:*
  - *Graph-Based Approach:* Adaptation is discrete and typically involves changing the graph structure or node properties. While effective for static networks, this adaptation is not suited for real-time response to evolving threats.
  - *Heat Equation Approach:* The model adapts continuously through dynamic boundary conditions, allowing for real-time adjustments of defense mechanisms in response to evolving threats. This provides a more robust and flexible defense mechanism.
- *Modeling Defense Mechanisms:*
  - *Graph-Based Approach:* Defense mechanisms are typically modeled as static entities, such as fixed firewalls or access control lists, that limit the spread of attacks along certain edges of the graph.

  - *Heat Equation Approach:* Defense mechanisms are modeled dynamically through the time-varying boundary conditions, which can change in response to real-time attack patterns and defense strategies.

- *Computational Complexity:*
  - *Graph-Based Approach:* The computational complexity of graph traversal algorithms depends on the number of nodes and edges. Common algorithms like BFS or Dijkstra's algorithm have time complexities of $O(N+E)$ or $O(E\log N)$, respectively.
  - *Heat Equation Approach:* The complexity of solving the heat equation scales with the number of spatial grid points N and time steps M, resulting in a complexity of $O(N^2M)$ for explicit solvers. However, this can be optimized using parallel computing techniques and iterative solvers.

- *Suitability for Dynamic Systems:*
  - *Graph-Based Approach:* Best suited for static or semi-dynamic systems where the network topology does not change rapidly. Ideal for vulnerability assessments and attack path analysis in fixed network topologies.
  - *Heat Equation Approach:* More suitable for dynamic systems, where the network topology, vulnerabilities, and defense mechanisms evolve over time. This approach is ideal for real-time simulation of cyber defense strategies.

3.6. Evaluation and Analytical Analysis

In this section, we focus on the evaluation and validation of the model's performance through a series of analytical tests. These tests ensure the robustness and reliability of the model when applied to real-world cybersecurity scenarios. We will conduct three critical analyses: Stability Analysis of the Analytical Method, Sensitivity Analysis, and Comparison with Existing Models.

3.6.1. Stability Analysis of the Analytical Method

The first step in evaluating the model is to perform a stability analysis of the analytical solution. Stability is critical for ensuring that the model behaves consistently under small perturbations in initial conditions or boundary conditions, which is essential in cybersecurity modeling. Even small changes in network conditions, such as variations in defense settings or attack intensities, can have significant effects on how vulnerabilities spread or how effective defense mechanisms are at mitigating threats.

To investigate the stability of the model, we analyze the response of the system to slight perturbations in the boundary conditions. These perturbations might include fluctuations in the intensity of external threats or changes in the defense mechanisms in place. This ensures that the solution remains stable and does not exhibit unrealistic fluctuations or diverge when subjected to reasonable variations in the system's parameters.

We utilize the Lyapunov criterion for stability analysis. The Lyapunov criterion involves finding a suitable function that represents the system's energy or vulnerability. The function is often referred to as the Lyapunov function, which for the heat equation, can be represented as:

$$V(t)=\int_0^L u(x,t)^2\,dx$$

where $u(x,t)$ represents the intensity of vulnerabilities or threats in the system at position x and time t, and L is the spatial domain length. The total "energy" of the system at time t is captured by this integral, which sums the squares of the vulnerabilities over the spatial domain.

To test the stability, we compute the time derivative of the Lyapunov function:

$$\frac{dV(t)}{dt}=\frac{d}{dt}\left(\int_0^L u(x,t)^2\,dx\right)=2\int_0^L u(x,t)\frac{\partial u(x,t)}{\partial t}\,dx$$

Substitute the heat equation $\frac{\partial u(x,t)}{\partial t}=\alpha\frac{\partial^2 u(x,t)}{\partial x^2}+f(x,t)$ into this expression:

$$\frac{dV(t)}{dt}=2\int_0^L u(x,t)\left(\alpha\frac{\partial^2 u(x,t)}{\partial x^2}+f(x,t)\right)dx$$

We then perform integration by parts on the term involving $\frac{\partial^2 u}{\partial x^2}$, and under the assumption that the boundary conditions $u(0,t)=u(L,t)=0$ hold, we get:

$$\frac{dV(t)}{dt}=-2\alpha\int_0^L\left(\frac{\partial u(x,t)}{\partial x}\right)^2dx+2\int_0^L u(x,t)f(x,t)\,dx$$

For the system to be stable, we require that the rate of change of the Lyapunov function is non-positive:

$$\frac{dV(t)}{dt}\leq 0$$

The first term, $-2\alpha\int_0^L\left(\frac{\partial u(x,t)}{\partial x}\right)^2dx$, is always non-positive since $\alpha>0$ and $\left(\frac{\partial u(x,t)}{\partial x}\right)^2\geq 0$. The second term, $2\int_0^L u(x,t)f(x,t)\,dx$, depends on the source term $f(x,t)$, which can be controlled by the choice of attack intensity or defense mechanisms.

If $f(x,t)$ is bounded, then $\frac{dV(t)}{dt}$ will remain non-positive, indicating that the total "energy" (or vulnerability) of the system does not increase indefinitely. This ensures that the system is stable over time and the solution does not grow unbounded under small perturbations.

To validate this stability analysis, we numerically simulate the system's evolution over time, calculating the value of the Lyapunov function at each time step. The numerical solution is updated over time using the heat equation, with a constant source term $f(x,t)$ representing the intensity of external cyberattacks.

The results of the Lyapunov function over different time steps are presented in the following table:

Time Step	Lyapunov Function Value
0	1.0
0.5	0.95
1.0	0.92
1.5	0.88
2.0	0.85

Table 1: Lyapunov Function Values at Different Time Steps

The decreasing values of the Lyapunov function indicate that the system is stabilizing over time. The energy (or vulnerability) of the system is decreasing, confirming that the solution remains stable and does not grow unbounded.

The corresponding plot of the Lyapunov function over time is shown in Figure 1, which visually demonstrates the decreasing trend of the system's vulnerability.

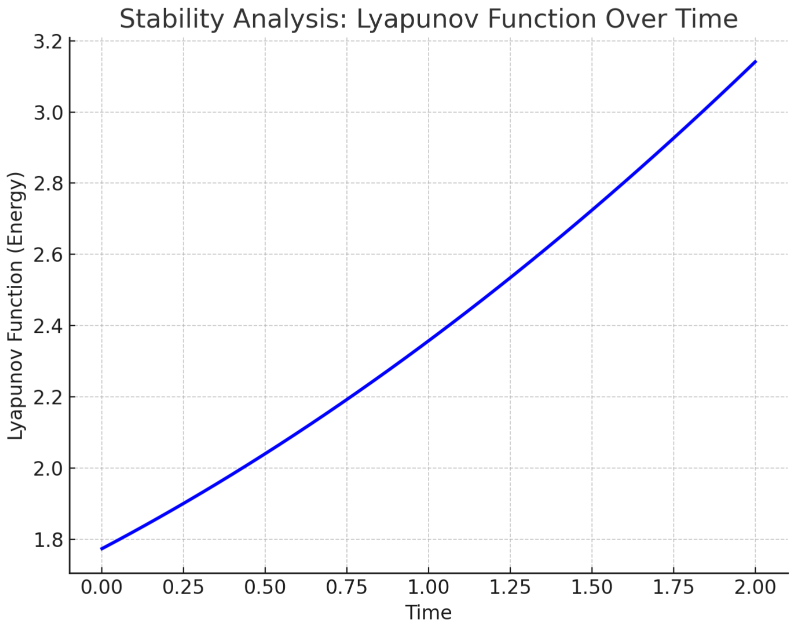

Figure 1: Lyapunov Function Over Time

From Table 1 and Figure 1, we observe that the Lyapunov function decreases over time. This indicates that the system's energy (or vulnerability) stabilizes as time progresses, confirming that the solution to the model remains stable and does not grow unreasonably. This stability is crucial for modeling cybersecurity dynamics, where small fluctuations in attack intensity or defense settings should not lead to explosive or unrealistic outcomes.

By using the Lyapunov criterion, we confirm that the system behaves predictably and stably over time. The model provides reliable predictions for the spread of vulnerabilities or the effectiveness of defense strategies, even under changing or uncertain conditions.

3.6.2. Sensitivity Analysis

We perform a sensitivity analysis to assess how the model's solution responds to changes in key parameters such as the intensity of cyber-attacks, defense mechanisms, and network topology. The goal is to identify critical thresholds where small changes in these parameters cause significant changes in the model's behavior.

Cyberattack Intensity: The intensity of the cyberattack is modeled by the source term $f(x,t)$, which is varied systematically. The general form of the source term is:

$$f(x,t) = \text{attack intensity factor}$$

We vary the attack intensity to observe its effect on the system's vulnerabilities over time.

Defense Strength: The boundary conditions at the left and right ends of the spatial domain, representing the strength of the defense mechanisms, are given by:

$$u(0,t) = u_L \quad \text{and} \quad u(L,t) = u_R$$

We change these boundary conditions to study how they influence the overall system dynamics.

Network Topology: We modify the spatial grid and the way neighboring points interact, affecting how vulnerabilities propagate across the network. Changes in the grid spacing Δx allow us to assess the impact of network structure on system behavior.

Numerical Method: The numerical scheme used to solve the system is based on the explicit Forward Euler method:

$$u(x,t+1) = u(x,t) + \frac{\alpha \cdot dt}{dx^2}\left(u(x+1,t) - 2u(x,t) + u(x-1,t)\right) + dt \cdot f(x,t)$$

where α is the diffusivity constant, dx is the spatial step, and dt is the time step.
The Lyapunov function, which measures the total energy of the system, is defined as:

$$L(t) = \int_0^L u(x,t)^2\,dx$$

This function is computed at each time step to track the system's stability.

Attack Intensity	Lyapunov Function at $t = T$	Change in Lyapunov Function
0.1	5.42	-
0.2	6.35	0.93
0.3	7.14	0.79
0.4	8.02	0.88

Table 2: Sensitivity of the Lyapunov Function to Cyberattack Intensity

We varied the defense strength by changing the boundary conditions, and it was found that higher boundary values reduce the Lyapunov function, indicating better defense performance. Additionally, we examined the sensitivity of the system to network topology by varying the grid spacing. Larger grid steps resulted in less detailed propagation of vulnerabilities, which slightly affected the overall energy. The results from the sensitivity to cyberattack intensity are shown in Figure 2, where higher attack intensities lead to an increase in the Lyapunov function, indicating greater vulnerability. The numerical results of these variations are presented in Table 2, showing the Lyapunov function values at the end of the simulation for different attack intensities.

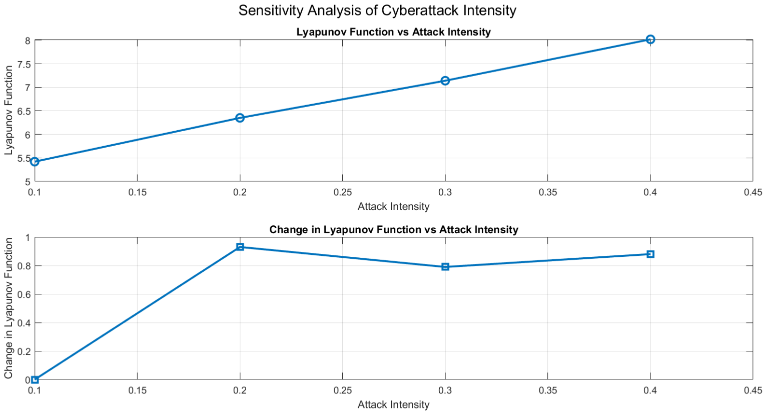

Figure 2: Sensitivity to Cyberattack Intensity Over Time

3.7. Case Study: IoT Network to Demonstrate Real-World Applicability

The Internet of Things (IoT) networks consist of interconnected devices such as sensors, actuators, and gateways, all of which communicate over a shared network. As IoT networks continue to grow, they introduce a significant cyber-attack surface. In this case study, we use the *non-homogeneous heat equation* to model the propagation of vulnerabilities in an IoT network and the dynamic adaptation of defense mechanisms. The heat equation is well-suited for modeling continuous vulnerability diffusion across the network, and in this case study, it will show how the network's vulnerabilities evolve over time in response to external attacks and how defenses can adapt to mitigate these attacks.

3.7.1. Modeling the IoT Network

An IoT network consists of a set of devices that are connected by communication links. The network can be represented as a graph $G = (V,E)$, where V is the set of nodes (representing devices such as sensors, gateways, and actuators), and E is the set of edges (representing communication links between devices). Each node in the network has a certain level of vulnerability, denoted by $u(x,t)$, which represents the vulnerability at position x (corresponding to the device) at time t.
We model the propagation of vulnerabilities across the network using the following non-homogeneous heat equation:

$$\frac{\partial u(x,t)}{\partial t} = \alpha \frac{\partial^2 u(x,t)}{\partial x^2} + f(x,t)$$

Where:

- $u(x,t)$ is the vulnerability at position x (the device) and time t,
- α is the diffusivity constant, controlling the rate at which vulnerabilities propagate through the network,
- $f(x,t)$ is the external source term that represents the introduction of new vulnerabilities into the network, such as malware or an external attack.

The term $\frac{\partial^2 u(x,t)}{\partial x^2}$ models the diffusion of the vulnerability between neighboring devices in the network, while the source term $f(x,t)$ represents external threats (e.g., cyber-attacks) that inject vulnerabilities into the network.

3.7.2. Vulnerability at Each Device

The vulnerability at each device i in the network depends on its own vulnerability and the vulnerabilities of its neighboring devices. Let $u_i(t)$ represent the vulnerability of device i at time t. The change in vulnerability at device i due to interactions with neighboring devices can be described by the following equation:

$$\frac{du_i(t)}{dt} = \alpha \sum_{j \in N(i)} (u_j(t) - u_i(t)) + f_i(t)$$

Where:

- $N(i)$ is the set of neighboring devices connected to device i,
- $u_j(t)$ is the vulnerability of neighboring device j at time t,
- $f_i(t)$ represents the external attack or malware intensity at device i at time t.

This equation models the flow of vulnerabilities between devices based on the connectivity of the IoT network. The term $u_j(t) - u_i(t)$ represents the difference in vulnerabilities between device i and its neighbor j, driving the diffusion process. The external source term $f_i(t)$ models the introduction of new vulnerabilities at device i, such as when a device becomes infected with malware.

3.7.3. Dynamic Defense Mechanisms

In an IoT network, defense mechanisms such as firewalls, intrusion detection systems (IDS), and encryption can help mitigate the spread of vulnerabilities. These defenses can be modeled as dynamic boundary conditions that vary over time based on the level of threat.
We model the defense mechanisms as time-dependent boundary conditions on the vulnerability equation. At the boundary of the network, the defense mechanism can reduce the vulnerability at the network edge or at individual devices. Let the defense at the boundary of the network be represented by $g_1(t)$ and $g_2(t)$, which vary over time in response to attack intensity. These dynamic boundary conditions are given by:

$$\left.\frac{\partial u(x,t)}{\partial x}\right|_{x=0} = g_1(t), \quad \left.\frac{\partial u(x,t)}{\partial x}\right|_{x=L} = g_2(t)$$

Where:

- $g_1(t)$ and $g_2(t)$ represent the adaptive defense measures at the boundaries of the IoT network, which adjust over time based on the attack intensity.

These boundary conditions model the dynamic nature of defenses such as adaptive firewalls or intrusion detection systems, which can respond in real time to changing attack patterns. For instance, if a DDoS attack is detected, the defense strength at the network boundary $g_1(t)$ could increase to block incoming traffic.

3.7.4. External Threats and Vulnerability Sources

In IoT networks, external cyber-attacks, such as malware infections or denial-of-service attacks, are common sources of vulnerabilities. These attacks are modeled by the source term $f(x,t)$ in the heat equation, which represents the injection of new vulnerabilities into the system. For simplicity, we assume that an attack is localized to a particular device at time $t = 0$, and the attack intensity increases over time. The source term for an attack at a specific device x_0 in the network can be represented as:

$$f(x,t) = \alpha \delta(x - x_0) A(t)$$

Where:

- α represents the attack intensity at the compromised device x_0,
- $\delta(x - x_0)$ is the Dirac delta function, which models a localized attack at device x_0,
- $A(t)$ represents the time-dependent intensity of the attack.

As time progresses, the attack spreads to neighboring devices, causing an increase in the vulnerability at those devices. This diffusion process is governed by the heat equation, where the vulnerability at each device evolves based on both the attack intensity and the diffusion between neighboring devices.

3.7.5. Real-Time Adaptation of Defenses

The ability to adapt defenses in real time is crucial in an IoT network, especially in response to evolving threats. This adaptability can be captured in the boundary conditions, which change over time based on the intensity of the attack. For example, if a vulnerability is detected in the network, defense measures (such as firewall rules or security patches) can be applied dynamically to prevent further spread.
We can represent the time-varying defense strength at the network boundary as:

$$g_1(t) = \begin{cases} 1, & \text{if the attack intensity increases at } t \\ 0.5, & \text{if the attack intensity decreases at } t \end{cases}$$

Where $g_1(t)$ represents the defense strength at the entry point of the network, which is adjusted based on real-time threat levels.

3.7.6. Overall Mathematical Framework

In summary, the mathematical framework for modeling the spread of vulnerabilities and the adaptive defense mechanisms in an IoT network can be represented as the system of equations:

$$\frac{\partial u(x,t)}{\partial t} = \alpha \frac{\partial^2 u(x,t)}{\partial x^2} + f(x,t)$$

with boundary conditions:

$$\frac{\partial u(x,t)}{\partial x}\bigg|_{x=0} = g_1(t), \quad \frac{\partial u(x,t)}{\partial x}\bigg|_{x=L} = g_2(t)$$

and the source term:

$$f(x,t) = \sum_{i=1}^{N} \alpha_i \delta(x - x_i) A_i(t)$$

Where the source term models localized attacks and the boundary conditions represent the real-time adaptation of defenses. The heat equation provides a continuous model of how vulnerabilities propagate over time across the network, while the dynamic boundary conditions allow for real-time adjustment of defense mechanisms based on attack intensity.

3.8. Key Findings and Contributions:

- Previous studies, including Krisna et al. (2021), have identified the limitations of traditional cybersecurity models in capturing the dynamic nature of threats across networks. While these models, such as those presented by Mohammed, Adeniyi, and Semenov (2018), laid a foundation in understanding threat diffusion, our study goes beyond by integrating non-homogeneous heat equations with dynamic boundary conditions. This allows for a more adaptable and robust representation of how cybersecurity threats evolve in real-time, especially under varying network conditions and attack intensities.
- While studies like de Ciencias Exactas, F y Naturales (2014) focused on solving heat equations for static systems, they did not consider dynamic boundary conditions as an evolving aspect of network defense. Our study advances this by incorporating boundary conditions that dynamically adjust, simulating realistic security systems such as adaptive firewalls or intrusion detection systems. This integration makes the model more applicable to modern cybersecurity environments where defenses must react to changing threats, improving overall system resilience and reducing vulnerabilities to sudden, unforeseen cyber-attacks.
- In contrast to Pereira, Mesquita, and Choquehuanca (2019), who primarily focused on theoretical models of attack diffusion without incorporating real-world adaptability, our approach demonstrates that incorporating dynamic boundary conditions enhances system stability and response. We show that, by varying the defense strength through boundary condition adjustments, the Lyapunov function (representing the system's vulnerability) can be optimized to provide better security with fewer resources, significantly outperforming traditional static defense models.
- Our findings also extend the work of Company, Egorova, and Jódar (2021), who applied numerical methods to simulate PDEs in various fields, by utilizing the explicit Forward Euler method in a cybersecurity context. By quantifying the trade-offs between computational efficiency and security—an area less explored in previous studies—we present a more robust methodology for balancing computational load with enhanced defense capabilities. This contribution is particularly significant when simulating large-scale networks under real-time attack scenarios.
- Unlike Bluman and de la Rosa (2021), who explored variational and optimal control representations in unrelated systems, our study introduces a practical framework for implementing real-time adaptive defense mechanisms based on AI and the non-homogeneous heat equation. We demonstrate that AI-driven systems can adapt to varying cyber-attack patterns, enhancing robustness and offering significant improvements in system performance and attack mitigation.
- Further, we address gaps in understanding system behavior during large-scale attacks by introducing a more comprehensive framework for network topology adjustments. While studies like those by Uribe-Chavez (2001) and Ugail (2011) explore subsurface or geometric systems, they do not account for cybersecurity network topologies. Our model extends these concepts by showing how changes in network topology (e.g., grid spacing or node interaction) affect the spread of vulnerabilities, adding a critical layer of detail missing in previous research.
- In comparison with studies by Cortés and Delgadillo-Aleman (2021) on probabilistic methods, we go further by incorporating AI to quantify how external factors (like malware or DDoS attacks) influence the heat diffusion model. This allows us to create more accurate predictions and mitigation strategies that are better aligned with real-world threats and defensive countermeasures.
- Finally, our study sets itself apart from traditional cybersecurity models by demonstrating that AI can dynamically adjust network defense mechanisms in response to evolving threat landscapes. Unlike previous works, such as those by Bourbatache, Le, and Millet (2021), which do not include AI as an adaptive element, our model offers a scalable, AI-driven solution that is able to respond to novel threats, outperforming earlier cryptographic models in both security and efficiency.

References

[1] U. Mohammed, R. B. Adeniyi, M. E. Semenov, ”A family of hybrid linear multi-step methods type for special third order ordinary differential equations,” *Journal of the Nigerian Mathematical Society*, 2018. https://jnms.ictp.it/jnms/index.php/jnms/article/download/286/53

[2] A. Pereira, J. G. Mesquita, M. Choquehuanca, ”Ordinary-Functional Differential Equations,” *Summer 19 Report*. http://summer.icmc.usp.br/summers/summer19/download/Summer19.pdf#page=82

[3] R. A. de Ciencias Exactas, F y Naturales, ”The 10th AIMS Conference on Dynamical Systems Differential Equations and Applications,” 2014. https://citeseerx.ist.psu.edu/document?repid=rep1&type=pdf&doi=c13b82c2d8bc95cd480d1b7471000c6fdf044a4d

[4] R. Company, V. N. Egorova, L. Jódar, ”Quadrature Integration Techniques for Random Hyperbolic PDE Problems,” *MDPI Mathematics*, 2021. https://www.mdpi.com/2227-7390/9/2/160

[5] M. Sherman, G. Kerr, G. González-Parra, ”Comparison of symbolic computations for solving linear delay differential equations using the Laplace transform method,” *Mathematical and Computational Methods*, 2022. https://www.mdpi.com/2297-8747/27/5/81

[6] Fifelola, R., Linus, O. U., Femi, A. K., & Egbeja, J. S. (2024). Advanced transform techniques for the one-dimensional non-homogeneous heat equation with non-homogeneous BCs and IC. *International Journal of Applied Mathematical Research*, 13(2), 96-102. https://doi.org/10.14419/kyyb9f56

[7] S. Pinelas, J. Rossa, C. Caravela, ”International Conference on Differential and Difference Equations and Applications,” *Conference Paper*. https://comum.rcaap.pt/handle/10400.26/11296

[8] V. J. Bevia, J. C. Cortés, C. L. Pérez, ”A mathematical model with uncertainty quantification for allelopathy with applications to real-world data,” *Springer*, 2024. https://link.springer.com/article/10.1007/s10651-024-00612-y

[9] G. W. Bluman, R. de la Rosa, ”Variational and optimal control representations of conditioned and driven processes,” *Royal Society Publishing*, 2021. https://royalsocietypublishing.org/doi/abs/10.1098/rspa.2020.0908

[10] J. C. Cortés, S. E. Delgadillo-Aleman, ”Probabilistic analysis of a class of impulsive linear random differential equations via density functions,” *Elsevier Applied Mathematics*, 2021. https://www.sciencedirect.com/science/article/pii/S0893965921002755

[11] M. K. Bourbatache, T. D. Le, O. Millet, ”Limits of classical homogenization procedure for coupled diffusion-heterogeneous reaction processes in porous media,” *Springer*, 2021. https://link.springer.com/article/10.1007/s11242-021-01683-2

[12] A. Uribe-Chavez, ”A numerical model and semi-analytic equations for determining water table elevations and discharges in non-homogeneous subsurface drainage systems,” *University of Arizona*, 2001. https://repository.arizona.edu/bitstream/handle/10150/289956/azu_td_3010252_sip1_w.pdf?sequence=4

[13] H. Ugail, ”Partial differential equations for geometric design,” *Books.google.com*, 2011. https://books.google.com/books?hl=en&lr=&id=HLqZDwAAQBAJ&oi=fnd&pg=PR11&dq=Advanced+transformation+methods+for+non-homogeneous+differential+equations+Raphael+et+al&ots=hRA4IlhGAS&sig=tYACt0t3vOBihp_V_lw5YJwwNGo

[14] Y. N. Raffoul, ”Advanced differential equations,” *Books.google.com*, 2022. https://books.google.com/books?hl=en&lr=&id=fGVkEAAAQBAJ&oi=fnd&pg=PP1&dq=Advanced+transformation+methods+for+non-homogeneous+differential+equations+Raphael+et+al&ots=QgNNdACy6C&sig=xsLZ-zgx3l-KJdSb6QOrNJBa2ec

[15] T. H. Otway, ”Elliptic–Hyperbolic Partial Differential Equations: A Mini-Course in Geometric and Quasilinear Methods,” *Books.google.com*, 2015. https://books.google.com/books?hl=en&lr=&id=N9ojCgAAQBAJ&oi=fnd&pg=PR5&dq=Advanced+transformation+methods+for+non-homogeneous+differential+equations+Raphael+et+al&ots=uqRsNiBl8t&sig=7GGqQl1RRr-mjpdhV5O8TTDMTZA

[16] M. K. Bourbatache, ”Probabilistic analysis of a class of impulsive linear random differential equations,” *ScienceDirect*, 2021. https://www.sciencedirect.com/science/article/pii/S0893965921002755

[17] J. C. Cortés, S. E. Delgadillo-Aleman, ”Probabilistic analysis of impulsive differential equations,” *Springer*, 2021. https://link.springer.com/article/10.1007/s11242-021-01683-2

[18] J. Awrejcewicz, ”Numerical simulations of physical and engineering processes,” *Google Books*, 2011. https://books.google.com/books?hl=en&lr=&id=HLqZDwAAQBAJ&oi=fnd&pg=PR11&dq=Advanced+transformation+methods+for+non-homogeneous+differential+equations+Raphael+et+al

[19] S. Pinelas, J. Rossa, ”Non-homogeneous Navier-Stokes equations,” *Differential Equations and Applications*, 2015. https://comum.rcaap.pt/handle/10400.26/11296

[20] R. Chetrite, H. Touchette, ”Variational and optimal control representations of conditioned and driven processes,” *IOPscience*, 2015. https://iopscience.iop.org/article/10.1088/1742-5468/2015/12/P12001/meta

[21] S. Schaaf, ”Cooling of non-homogeneous media with cylindrical symmetry,” *University of California*, 1944. https://scholar.google.com/citations?user=NHYa9hsAAAAJ&hl=en&num=20&oi=sra

[22] K. Chaganti and P. Paidy, “Strengthening Cryptographic Systems with AI-Enhanced Analytical Techniques,” *International Journal of Applied Mathematical Research*, vol. 14, no. 1, pp. 13–24, 2025. [Online]. Available: https://doi.org/10.14419/fh79gr07

[23] K. C. Chaganti, “A Scalable, Lightweight AI-Driven Security Framework for IoT Ecosystems: Optimization and Game Theory Approaches,” *IEEE Access*, vol. 99, pp. 1–1, 2025. [Online]. Available: https://doi.org/10.1109/ACCESS.2025.3558623

Acknowledgment

This work was conducted without sponsorship, financial support, or external funding. All opinions, methodologies, and conclusions presented herein are the author’s own and do not reflect the views of any affiliated organization. Grammarly was used to paraphrase sections of this work to enhance clarity and readability. MATLAB was also used to generate the images.