A Game-Theoretic Approach for Optimal Multi-Target Defense Strategies in Programmable Networking

Jamil Ahmad Kassem[1,2][a], Helena Rifà-Pous[1,2][b] and Joaquin Garcia-Alfaro[3][c]

[1]*UOC-TECH Research Center, Computer Science, Multimedia and Telecommunication Studies, Barcelona 08018, Spain*

[2]*CYBERCAT - Center for Cybersecurity Research of Catalonia, Spain*

[3]*SAMOVAR, Télécom SudParis, Institut Polytechnique de Paris, Palaiseau 91120, France*

Keywords: Cyberdefense, Programmable Network, Software Defined Networking, Moving Target Defense, Game Theory.

Abstract: Traditional network defense strategies, which follow a linear sequence of vulnerability discovery, defense selection, and attack mitigation, often struggle to adapt to emerging and unpredictable cyber threats. This paper presents a novel strategic framework designed to optimize defense costs, addressing both security concerns and cost-effectiveness. Drawing inspiration from security games and Stackelberg-style leader–follower interactions, our approach introduces a resource management strategy that optimizes a defender-side cost function under rational attacker behavior. We test our approach within the programmable networking paradigm, which is expected to expand traditional network architectures. Our numerical evaluation in SDN-like scenarios shows that the proposed strategy significantly reduces total defense costs compared to representative Moving Target Defense (MTD) baselines, while providing defenders with the flexibility to trade off attack impact against reconfiguration costs.

1 Introduction

Cybersecurity remains a complex and ever-evolving challenge. Traditional defense mechanisms, which rely on reactive detection and mitigation, struggle to combat increasingly sophisticated adversaries. These adversaries, equipped with partial knowledge of the system, can potentially bypass existing defenses. To address these advanced threats, the research community has explored innovative security approaches such as *Moving Target Defense* (MTD) that disrupt attacks by dynamically altering system configurations [Sengupta et al., 2020, Ghosh et al., 2009]. This proactive strategy does not require prior knowledge of the adversaries or their methods, offering a distinct advantage [Zscaler, 2023]. MTD operates by introducing uncertainty into the system. Examples of this strategy include dynamically changing IP addresses or network routing configurations [Upeksha et al., 2024]. Adequate randomization can hinder the adversary's ability to gather knowledge, ultimately reducing the impact of potential cyberattacks [Gonzalez-Granadillo et al., 2018].

Despite growing interest, MTD faces notable limitations, as some methods lack the adaptability or efficiency required to counter persistent, rational adversaries. To address these challenges, this paper proposes a mathematical framework based on game theory and optimization to model the interaction between a defender and an attacker [A Petrosyan, 2016]. Game theory formalizes such interactions through defined actions and outcomes [Osborne et al., 2004], where players act according to utility functions that quantify gains and losses. Assuming both players behave rationally, each seeks to minimize losses; in contrast, against non-rational adversaries (*e.g.*, inexperienced attackers), low-risk defensive strategies can maintain comparable security. Our approach is implemented within the context of the programmable networking paradigm. As a practical example, we apply the approach to *Software Defined Networking* (SDN) [Kreutz et al., 2014] to accommodate our MTD strategies [Rubio-Hernan et al., 2018, Yoon et al., 2020]. The primary contributions of our paper focus on the timing and frequency aspects of MTD.
The paper is organized as follows. Section 2 surveys related work. Section 3 presents our proposed method and analytic results. Section 4 provides a baseline scenario for the adversary and the defender. Section 5 provides numerical results. Section 6 concludes.

[a] https://orcid.org/0000-0002-1040-8187
[b] https://orcid.org/0000-0003-0923-0235
[c] https://orcid.org/0000-0002-7453-4393

Takeaway

- Cost reduction using resource correlation.
- A closed-form strategy that avoids explicitly solving the NP-hard problem and can be computed in polynomial time.
- The effects of system configuration on performance metrics.

2 Related Work

The evolving landscape of cybersecurity has spurred research into proactive defense strategies that deter or redirect attacks rather than solely mitigating them. Early efforts explored deception-based techniques by deploying fake network terminals, using synthetic data, or decoy network interfaces to divert attacks from critical components [Borders et al., 2007, Lazarov et al., 2016, Rrushi, 2016]. These approaches highlight the potential for deception, but often lack adaptability to persistent and sophisticated threats. MTD has emerged as a prominent proactive strategy, leveraging techniques such as game theory, heuristics, and machine learning to dynamically adjust system configurations [Cho et al., 2020]. MTD activation can be time-based, event-driven, or a hybrid mixture of both [Colbaugh and Glass, 2012, Zhuang et al., 2012, Keromytis et al., 2012]. Although these methods enhance resilience, their one-dimensional nature leaves them vulnerable to advanced adversaries employing novel attack strategies.

Specific MTD implementations have addressed scalability and adaptability. Yoon *et al.* [Yoon et al., 2020] proposed an asset-aware MTD using SDN and a three-tier attack graph, reducing system costs by factoring in attack paths and asset criticality. In contrast, Feng *et al.* [Feng et al., 2017] combined MTD with deception in a game-theoretic model, where defenders relocate resources and use deceptive signals to mislead adversaries. Other studies focus on specific threats, such as *Denial of Service* (DoS) attacks. Jia *et al.* [Jia et al., 2013] introduced a proxy-switching MTD, shuffling non-malicious nodes across secure proxies to counter insider threats. Wright *et al.* [Wright et al., 2016] extended this with a refined payoff model, emphasizing proactive defenses against adversaries targeting high-node proxies. In a separate work, Charpentier *et al.* [Charpentier et al., 2023] advanced MTD adaptability using deep reinforcement learning. These models often treat resources as a monolithic entity, neglecting correlations that could enhance efficiency.

Recent work by Kassem *et al.* [Ahmad Kassem et al., 2024, Ahmad Kassem et al., 2025] proposed an MTD framework for managing multiple resources; however, it inadequately addresses the entire lifespan of the system. Expanding on the work of Kassem *et al.* [Ahmad Kassem et al., 2024, Ahmad Kassem et al., 2025], this paper introduces a novel model designed to optimize the management of multiple resources. We consider the system's various components and their interrelationships, thereby reducing defense costs and mitigating the effects of attacks. Costs are quantified using established methodologies, such as the return-on-investment concept by Jeffrey *et al.* [Jeffrey, 2004], which evaluates component significance in a network through a cost-centric framework.

3 Proposed Approach

Inspired by related work on security games [Shukla et al., 2023, Ait Temghart et al., 2023], we design a defender-centric optimization framework. In this framework, the defender commits to a resource placement, and the adversary best responds by attacking the node with the maximum expected impact. The methodology is structured as follows: (i) construction of a system model with multiple nodes and resources (Section 3.3), (ii) integration of a game-theoretic approach and utility function to guide defender actions (Section 3.4), (iii) derivation of a general optimization solution (Section 4), and (iv) validation against existing methods (Section 5).

Our model adopts a leader–follower structure where the defender first selects a probabilistic placement of resources. A rational attacker follows and best responds by attacking the node that maximizes the expected impact. We focus on a single rational attacker and directly model their best response.

3.1 Practical Framework

The application is inspired by the work of Abdelkhalek *et al.* [Abdelkhalek et al., 2022] regarding mathematical modeling in real-world distributed systems. We consider an SDN controller that utilizes OpenFlow[1] to instruct switches to reroute traffic. The rerouting effectively changes the network topology and resource locations at regular intervals [Kampanakis et al., 2014]. Notable examples of such resources include databases and HTTP handlers. Figure 1 shows a sample network that is maintained and managed by the following key components:

[1] An open-standard protocol that enables network controllers to interact directly with network devices.

- Centralized SDN controller using OpenDaylight.[2]
- OpenFlow protocol for network management.
- Programmable switches and routers.

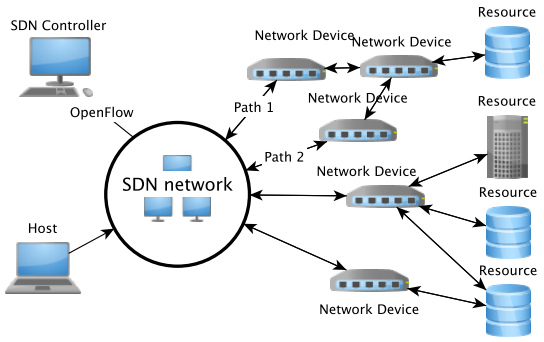

Figure 1: SDN model with multiple resources and a host. An OpenFlow controller manages the network

3.2 Adversarial Attack Vectors

The previous setup is vulnerable to cyberattacks ranging from DoS to data breaches. The adversarial model assumes that adversaries are persistent, adaptive, and resourceful while operating under uncertain conditions. Adversaries begin with reconnaissance, gathering intelligence on system architecture and vulnerabilities through monitoring, scanning, or social engineering. They then identify weaknesses during vulnerability discovery and exploit them using techniques such as DoS or data exfiltration attacks.

3.3 Mathematical Model

We represent the framework in a mathematical model, which we use to formulate the defender strategy and optimize costs. We represent routes as nodes that contain multiple resources, where resources are free to move between these nodes. The system is formalized as consisting of n nodes ($n > 1$) and m resources ($m > 0$). The system is under the control of a defender agent whose primary objective is to minimize overall costs. The model assumptions for the defender and adversary are summarized as follows. The adversary knows n and m but does not know αs. The defender knows the system state but does not know about the attack. Table 1 summarizes the symbols used throughout the paper.

[2] An open-source centralized SDN controller.

Symbol	Meaning
n	Number of nodes
m	Number of resources
k	Index for node
i	Index for resources
R	Set of resources r_i
c_m	Cost of moving resource
N_m	Number of moves of R
$\alpha(i,k)$	Probability of moving r_i to k
k_a	Index for attacked node
$T_c(k_a)$	Expected impact if attacking node k_a
$\mathbb{C}(k_a)$	Defense cost when node k_a is attacked

Table 1: Notations and Definitions

3.4 Resource Moving Strategy

Matrix $A \in [0,1]^{n\times m}$ represents the defender's placement of resources across nodes. Row k and column i correspond to node k and resource r_i, respectively, so entry $A_{k,i} = \alpha(i,k)$ is the probability that resource r_i resides at node k. The matrix shows the percentage of time spent by r_i across nodes. For example, $\alpha(1,1)$ signifies that r_1 spends half the time in the first node.

$$A = \begin{pmatrix} \alpha(1,1) & \alpha(2,1) & \dots & \alpha(m,1) \\ \alpha(1,2) & \alpha(2,2) & \dots & \alpha(m,2) \\ \vdots & \vdots & \ddots & \vdots \\ \alpha(1,n) & \alpha(2,n) & \dots & \alpha(m,n) \end{pmatrix} \quad (1)$$

For each node k, the expected impact $T_c(k)$ is defined as the expected number of resources located at k. Assuming each resource r_i has weight w_i (with $w_i = 1$ in this work), we have $T_c(k) = \sum_{i=1}^{m} w_i \alpha(i,k)$. Intuitively, $T_c(k)$ captures how likely and how many resources are concentrated at a node, which determines the adversary's expected gain when targeting it.

Defender's Cost: $\mathbb{C}(k_a) = T_c(k_a) + c_m \cdot N_m \sum_{i=1}^{m} (1 - \max \alpha(i))^2$. Here, c_m is the normalized cost of moving r_i, and N_m is the expected number of movement rounds of r_i. Thus, $c_m N_m$ represents the total cost of moving r_i. For each r_i, $\max \alpha(i)$ is the maximum probability of it staying in a single node. $1 - \max \alpha(i)$ is the fraction of time that r_i is moving.

Adversary's Utility: $U_a(k_a) = T_c(k_a)$. We model the adversary as a myopic but rational attacker that aims to maximize expected damage in a one-shot interaction. Extending the model to heterogeneous adversary types or multi-step learning adversaries is left for future work.

4 Optimum Scenario

This section elaborates on the methodology behind the defender strategy. Section 4.1 starts by introducing the general method and the basis of the defender's approach. The defender employs the previous methodology to develop a general strategy, as outlined in Section 4.2.

4.1 Matrix Derivation

We derive the defender's matrix by balancing two objectives: reducing resource movement and minimizing the expected impact on any targeted node. This problem is related to subset sum-style partitioning, where a set of elements is divided to achieve a desired aggregate value [Kleinberg and Tardos, 2003]. Here, however, the matrix A is the decision variable, and the target quantity is $T_c(k)$.

To illustrate the construction, consider $m = 5$ resources and $n = 3$ nodes. The initial matrix assigns equal probability to every resource-node pair, and the defender then redistributes probability mass so that the expected impact remains constant while movement cost is reduced. In this example, the optimized matrix is

$$A = \begin{pmatrix} \frac{1}{3} & \frac{1}{3} & \frac{1}{3} & \frac{1}{3} & \frac{1}{3} \\ \frac{1}{3} & \frac{1}{3} & \frac{1}{3} & \frac{1}{3} & \frac{1}{3} \\ \frac{1}{3} & \frac{1}{3} & \frac{1}{3} & \frac{1}{3} & \frac{1}{3} \end{pmatrix} \rightarrow \begin{pmatrix} \frac{5}{6} & 0 & 0 & \frac{5}{6} & 0 \\ 0 & \frac{5}{6} & 0 & 0 & \frac{5}{6} \\ \frac{1}{6} & \frac{1}{6} & 1 & \frac{1}{6} & \frac{1}{6} \end{pmatrix}$$

This construction preserves the desired expected impact while concentrating movement on a smaller subset of resources, thereby reducing cost.

4.2 Defender Preferred Strategy

Building on Section 4.1, we now generalize the defender's strategy. The construction yields $T_c(k) = \frac{m}{n}$, which serves as the baseline expected impact. We first derive the strategy that minimizes attack impact, and then extend it to a more general cost-minimizing formulation.

4.2.1 Minimized Attack Impact

The minimized attack impact provides a general solution to the matrix derivation problem described in Section 4.1. Resources are partitioned into those that are stationary and those that are moving.

The total count of moving resources $L = (m \bmod n)\left\lceil \frac{m}{n} \right\rceil$. This number is obtained due to the resource exchange methodology described in Section 4.1. Since this exchange is done in pairs it depends on $(m \bmod n)\left\lceil \frac{m}{n} \right\rceil$, which gives the number of resources that can be paired. The remaining resources are stationary and as such are assigned $\alpha = 1$. Subtracting these resources from the matrix and while maintaining $T_c(k) = \frac{m}{n}$ gives a base of $\alpha = \frac{1}{n\left\lceil \frac{m}{n} \right\rceil}$ for each of the moving resources. Pairing up resources, gives $(m \bmod n)$ resources with $\alpha(i,k) = \frac{m}{n\left\lceil \frac{m}{n} \right\rceil}$, where $k \in \{1, \ldots, m \bmod n\}$. The remaining αs of these resources that cannot be paired remain at the base value $\alpha = \frac{1}{n\left\lceil \frac{m}{n} \right\rceil}$.

After implementing the previous logic, the defender finds the defense cost as shown in Equation (2).

$$\mathbb{C}(k_a) = \frac{m}{n} + c_m \cdot N_m \cdot L \cdot \left(1 - \frac{m}{n\left\lceil \frac{m}{n} \right\rceil}\right)^2 \quad (2)$$

4.2.2 Proof of Correctness

The cost in Eq. (2) has two strictly positive terms: the expected impact and the movement cost. Since $T_c(k)$ is already fixed at its minimum feasible average value, any further change in $\alpha(i,k)$ would increase either the attack impact or the movement cost. Therefore, the construction is optimal under the stated constraint.

4.2.3 Minimized Cost

We now relax the strict constraint on $T_c(k)$ to allow a small increase in expected impact if it yields a larger reduction in movement cost. Let δ denote this tradeoff. The resulting cost function is

$$\mathbb{C}'(k_a) = \frac{m}{n} + \delta \cdot L + c_m \cdot N_m \cdot L \cdot \left(1 - \frac{m}{n\left\lceil \frac{m}{n} \right\rceil} - \delta\right)^2. \quad (3)$$

Minimizing this expression with respect to δ gives $\delta = 1 - \frac{m}{n\left\lceil \frac{m}{n} \right\rceil} - \frac{1}{2c_m \cdot N_m}$, which yields the final defender cost in Eq. (4):

$$\mathbb{C}'(k_a) = \frac{m}{n} + L \cdot \left(1 - \frac{m}{n\left\lceil \frac{m}{n} \right\rceil} + \frac{1 - 2N_m}{4c_m \cdot N_m^2}\right). \quad (4)$$

4.2.4 Proof of Correctness

Equation (2) gives the minimum cost under a fixed $T_c(k)$ constraint, while Eq. (3) extends the model by allowing a controlled tradeoff through δ. Optimizing over δ therefore yields the lowest cost achievable under the generalized formulation.

5 Simulation

This section presents a numerical evaluation of the defender strategy derived in Section 4. The evaluation

uses the system configuration defined in Sections 3.3 and 3.1. All simulations are executed on a desktop machine equipped with an AMD Ryzen 9 7940HS processor (4.00 GHz), 32 GB of RAM, and a 64-bit operating system. The implementation and full set of results are available online.[3] The study estimates the movement cost parameter c_m and the relative resource costs by analyzing real downtime and configuration times. Average server downtime is approximately 10 minutes [Santoso and Sari, 2022], while configuration operations require between 40 and 60 minutes [Lee, 2021]. We set the reallocation cost to $\frac{1}{4}$, which reflects the ratio $\frac{10 \text{ minutes}}{40 \text{ minutes}}$, and we normalize all subsequent costs with respect to the resource cost. Section 5.1 compares the proposed model against related approaches from the literature, and Section 5.2 analyzes how different system parameters influence the overall design and performance of the defense strategy.

5.1 Comparing With Similar Models

We compare the proposed model against two representative baseline models from the literature: Wang *et al.* [Wang et al., 2023] and Shukla *et al.* [Shukla et al., 2023]. The model by Wang *et al.* uses an energy-based formulation. In this formulation, the defender's budget is modeled as an energy level. This energy level is dynamically allocated across defense actions using game-theoretic optimization. The model by Shukla *et al.* combines greedy and defend-first heuristics with an equilibrium-based defense formulation. This combination leads to static or quasi-static placement strategies. These strategies emphasize protecting a subset of nodes. We utilize both baselines on the same network and threat setting as the proposed model. Their parameters are calibrated to preserve their core contributions. We also align these parameters with our cost normalization. For Shukla's approach, we consider three representative placement heuristics within this framework. One heuristic uses uniform placement of resources across nodes. A second heuristic prioritizes high-criticality resources. A third heuristic concentrates resources on a limited set of highly defendable nodes. These three heuristics capture a range of realistic defender behaviors.

All three models depend on parameters that are difficult to estimate precisely in highly uncertain environments. We therefore evaluate them over a range of parameter values. In this range, the defender still enjoys a reasonable margin of estimation error. We do not rely on a single calibrated point. Figure 3 illustrates the resulting total defense costs for different numbers of resources m. The figure highlights the cases $m = 40$ and $m = 200$. For small to moderate system sizes, such as $m = 40$, the proposed model yields slightly lower total cost than both Wang's and Shukla's models. This behavior reflects the benefit of explicitly balancing attack impact and movement overhead. As the system scales, such as at $m = 200$, the advantage of the proposed model becomes more pronounced. The costs of the two baseline models grow more steeply. The proposed model maintains a significantly lower total cost. This pattern indicates that static or energy-threshold-based policies tend to over-invest in defense actions in large systems. These policies also tend to concentrate resources suboptimally when the number of resources increases. The proposed strategy preserves cost-efficiency by systematically distributing movement load and impact. Overall, the results show that the proposed model consistently dominates the considered baselines in terms of total defense cost. This dominance is especially clear in larger networks.

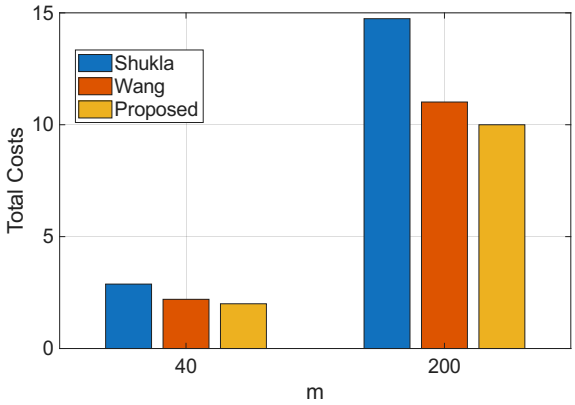

Figure 2: Defense cost $\mathbb{C}(k_a)$ when varying m. Two scenarios are considered, where m is either high or low.

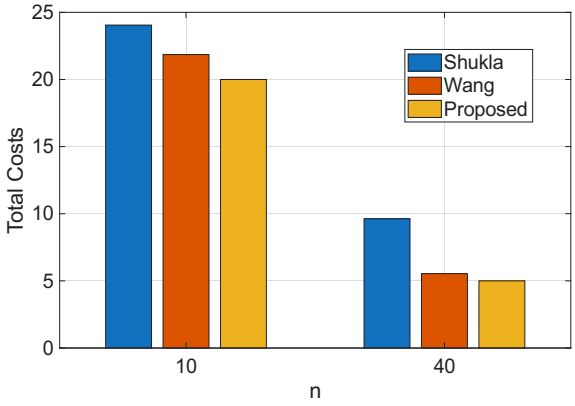

Figure 3: Defense cost $\mathbb{C}(k_a)$ when varying n. Two scenarios are considered, where n is either high or low.

[3]Implementation available at https://github.com/JamilahKassem/Optimal-Multi-Target-Defense-Strategies/

In Figure 3, we evaluate the sensitivity of $\mathbb{C}(k_a)$ to the number of nodes n. In a small network ($n = 10$), our proposed model achieves a cost reduction of approximately 10% over the Wang model and 20% over the Shukla model. As the network scales to $n = 40$, the performance gap between our approach and the Wang model narrows, yet the Shukla model exhibits doubled costs. These outcomes are a product of the defense methodologies. The Shukla model dual heuristic approach allows for versatility, but suffers from higher complexity compared to our approach while yielding slightly worse results. Conversely, the Wang model relies on an energy-based budget for resource placement. As n increases, the uncertainty inherent in estimating the required energy level leads to a higher margin of error and suboptimal allocations. Our strategy maintains dominance by balancing movement and expected impact costs.

This improvement arises from the tradeoff between expected attack impact and movement costs, which prevents resource concentration and reduces movement costs. By comparison, the two baseline models depend more strongly on defender-side assumptions and may perform worse when these values are not estimated accurately. In the experiments reported here, we consider an optimistic setting in which the defense parameters are assumed to be known with high accuracy.

Model complexity: Let $r = m \bmod n$ and $q = \lceil m/n \rceil$. The total runtime is determined by the number of iterations for the loops of the algorithm. The two loop counts sum to $T(m,n) = r^2q + (n-r)^2(q-1)$. This yields three cases: if $m \ll n$ then $T = \Theta(m^2)$; if $m = \Theta(n)$ then $T = \Theta(n^2)$; and if $m \gg n$ then $T = \Theta(mn)$. In large networks where n and m grow proportionally, the dominating behavior is $\Theta(mn)$. Since the loops are independent, the loops can be easily run in parallel.

5.2 Insights

This section is dedicated to establishing the relationship between various system variables and overall performance. The ultimate objective is to present a strategic framework for the defender during the construction of the system network. Specifically, when the defender cannot alter certain variables during runtime, these values are dependent upon the initial configuration. The defender can manage three primary variables, c_m, m, and n. The study uses similar setup as the one used in Section 5.

Cost of Movement c_m: The defender lacks direct control over the cost of movement, but it can be affected by the system's configuration. The findings indicate that this parameter determines the total cost savings achieved by deploying the proposed game model. In particular, applying the proposed model becomes more advantageous as c_m increases. In an optimal configuration where c_m is significantly low, MTD no longer requires optimization and can be used more freely. However, this scenario is currently infeasible given the existing technologies and networks.

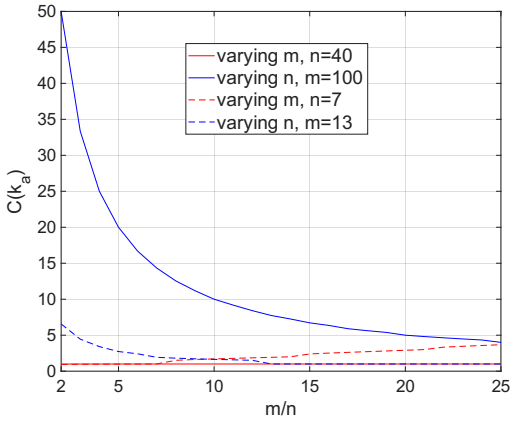

Figure 4: Defense cost $\mathbb{C}(k_a)$ when varying m and n. Two scenarios are considered for each, where m is either high or low, similarly for n.

System Configuration When configuring a network similar to the one depicted in Section 3.1, the primary challenge is determining the optimal values for n and m. The manager can be constrained by external physical or operational conditions that affect partitioning the system. Consequently, we provide a concise overview of how the overall costs of the systems can be optimized.

Number of nodes n: It is evident that increasing n gives advantages to the system manager. More importantly, costs eventually reach a limit where a further increase in n yields minimal gain. Specifically, as shown in Figure 4, the point at which the further increase in n becomes less advantageous depends on c_m. This occurs when $c_m = \frac{1}{4}$ increases n beyond 10 nodes, resulting in decreased benefits. However, for $c_m = \frac{1}{40}$, this value increases to the 15 nodes. As such, increasing c_m by tenfold leads to a 50% increase in n, after which increasing n becomes less beneficial.

Number of resources m: To examine the significance of m, we present Figure 4, where the costs are normalized by m to derive the cost per resource. This normalization facilitates understanding of the impact of increasing m. Although an increase in m increases total costs, the cost per resource exhibits negligible variation when m exceeds 5. Similarly to n, the configuration of m depends on c_m, where the lower values

of c_m demonstrate an improvement in costs when m increases further.

6 Conclusion

Conventional security methodologies are increasingly challenged by rapidly evolving threats within the contemporary cybersecurity landscape. This has precipitated a shift in scholarly research toward innovative, knowledge-agnostic strategies that diverge from the traditional framework of detection, mitigation, and recovery. Instead, new frameworks emphasize proactive and moving defense techniques that aim to minimize losses rather than halt attacks outright. However, the inherently active and dynamic nature of these strategies imposes additional operational costs on defense and requires careful optimization.

This paper proposes a novel strategy specifically designed to protect distributed resource-segmented networks. We utilize a leader-follower model together with an explicit defender cost function to optimize resource allocation and minimize defense costs. The simulations demonstrate that this strategy significantly reduces total defense cost compared to representative MTD baselines, particularly when dealing with multiple resources and larger networks. These results suggest that the proposed approach has strong potential to improve network security in programmable networking environments and provide insights into how various configurations and system variables impact overall performance.

Looking ahead, we plan to explore scenarios involving multiple intelligent adversaries and incorporate the concept of resource criticality into the model, aiming for a more nuanced and adaptive defense strategy.

Acknowledgments — This work was supported by the Spanish Ministry of Science and Innovation through the projects PID2021-125962OB-C31 "SECURING/CYBER" and PID2024-156914OB-C41 "SAFE/CYBER". Additional funding was provided by the ARTEMISA International Chair in Cybersecurity (C057/23) and the DANGER Strategic Project of Cybersecurity (C062/23), both funded by the Spanish National Institute of Cybersecurity through the European Union NextGenerationEU and the Recovery, Transformation, and Resilience Plan.

REFERENCES

A Petrosyan, L. a. (2016). *Recent advances in game theory and applications*. Springer.

Abdelkhalek, M., Hyder, B., Govindarasu, M., and Rieger, C. G. (2022). Moving target defense routing for sdn-enabled smart grid. In *2022 IEEE International Conference on Cyber Security and Resilience (CSR)*, pages 215–220.

Ahmad Kassem, J., Rifà-Pous, H., and Garcia-Alfaro, J. (2024). Diseño de una estrategia probabilística de defensa de objetivo móvil para manejar ataques contra nodos de red con múltiples recursos. In *XVIII Reunión Española de Criptología y Seguridad de la Información*, RECSI 24.

Ahmad Kassem, J., Rifà-Pous, H., and Garcia-Alfaro, J. (2025). Revisiting a probabilistic moving target defense strategy to handle attacks against network nodes with multiple resources. In *Advances in Information and Communications*, volume 1284 LNNS, page 536 – 554.

Ait Temghart, A., Marwan, M., and Baslam, M. (2023). Stackelberg security game for optimizing cybersecurity decisions in cloud computing. *Security and Communication Networks*, 2023(1):2811038.

Borders, K., Falk, L., and Prakash, A. (2007). Openfire: Using deception to reduce network attacks. In *2007 Third International Conference on Security and Privacy in Communications Networks and the Workshops - SecureComm 2007*, pages 224–233. IEEE.

Charpentier, A., Neal, C., Boulahia-Cuppens, N., Cuppens, F., and Yaich, R. (2023). Real-time defensive strategy selection via deep reinforcement learning. In *Proceedings of the 18th International Conference on Availability, Reliability and Security*, ARES '23. Association for Computing Machinery.

Cho, J.-H., Sharma, D. P., Alavizadeh, H., Yoon, S., Ben-Asher, N., Moore, T. J., Kim, D. S., Lim, H., and Nelson, F. F. (2020). Toward Proactive, Adaptive Defense: A Survey on Moving Target Defense. *IEEE Communications Surveys & Tutorials*, 22(1):709–745.

Colbaugh, R. and Glass, K. (2012). Predictability-oriented defense against adaptive adversaries. In *2012 IEEE international conference on systems, man, and cybernetics (SMC)*, pages 2721–2727. IEEE, IEEE.

Feng, X., Zheng, Z., Cansever, D., Swami, A., and Mohapatra, P. (2017). A signaling game model for moving target defense. In *IEEE INFOCOM 2017 - IEEE Conference on Computer Communications*, pages 1–9. IEEE.

Ghosh, A., Pendarakis, D., and Sanders, W. (2009). Moving target defense co-chair's report-national cyber leap year summit 2009. Technical report, Federal NITRD Program, Washington, DC, USA.

Gonzalez-Granadillo, G., Dubus, S., Motzek, A., Garcia-Alfaro, J., Alvarez, E., Merialdo, M., Papillon, S., and Debar, H. (2018). Dynamic risk management response system to handle cyber threats. *Future Generation Computer Systems*, 83:535–552.

Jeffrey, M. (2004). Return on investment analysis for e-business projects. *Internet Encyclopedia*, 3:211–236.

Jia, Q., Sun, K., and Stavrou, A. (2013). MOTAG: Moving Target Defense against Internet Denial of Service Attacks. In *2013 22nd International Conference on Computer Communication and Networks (ICCCN)*, pages 1–9. IEEE.

Kampanakis, P., Perros, H., and Beyene, T. (2014). Sdn-based solutions for moving target defense network protection. In *Proceeding of IEEE International Symposium on a World of Wireless, Mobile and Multimedia Networks 2014*, pages 1–6.

Keromytis, A. D., Geambasu, R., Sethumadhavan, S., Stolfo, S. J., Yang, J., Benameur, A., Dacier, M., Elder, M., Kienzle, D., and Stavrou, A. (2012). The meerkats cloud security architecture. In *2012 32nd International Conference on Distributed Computing Systems Workshops*, pages 446–450. IEEE, IEEE.

Kleinberg, J. and Tardos, E. (2003). Algorithm design.

Kreutz, D., Ramos, F. M., Verissimo, P. E., Rothenberg, C. E., Azodolmolky, S., and Uhlig, S. (2014). Software-defined networking: A comprehensive survey. *Proceedings of the IEEE*, 103(1):14–76.

Lazarov, M., Onaolapo, J., and Stringhini, G. (2016). Honey sheets: What happens to leaked google spreadsheets? In *9th USENIX Workshop on Cyber Security Experimentation and Test*. ResearchGate.

Lee, S. (2021). Reducing complexity of server configuration through public cloud storage. *Electronics*, 10(11).

Osborne, M. J. et al. (2004). *An introduction to game theory*, volume 3. Springer.

Rrushi, J. L. (2016). Nic displays to thwart malware attacks mounted from within the os. *Computers & Security*, 61:59–71.

Rubio-Hernan, J., Sahay, R., De Cicco, L., and Garcia-Alfaro, J. (2018). Cyber-physical architecture assisted by programmable networking. *Internet Technology Letters*, 1(4):e44.

Santoso, B. and Sari, M. W. (2022). Improvement of setup time on server infrastructure automation using ansible framework. *Journal of Engineering Science and Technology*, 17(5):3660–3671.

Sengupta, S., Chowdhary, A., Sabur, A., Alshamrani, A., Huang, D., and Kambhampati, S. (2020). A survey of moving target defenses for network security. *IEEE Communications Surveys & Tutorials*, 22(3):1909–1941.

Shukla, P., An, L., Chakrabortty, A., and Duel-Hallen, A. (2023). A robust stackelberg game for cybersecurity investment in networked control systems. *IEEE Transactions on Control Systems Technology*, 31(2):856–871.

Upeksha, R., Maduranga, M., Chanka Lasantha, N., and Aminda, N. (2024). Hybrid machine learning and moving target defense (MTD) for comprehensive switchport attack detection. In *2024 8th SLAAI International Conference on Artificial Intelligence (SLAAI-ICAI)*, pages 1–6.

Wang, Z., Shen, H., Zhang, H., Gao, S., and Yan, H. (2023). Optimal dos attack strategy for cyber-physical systems: A stackelberg game-theoretical approach. *Information Sciences*, 642:119134.

Wright, M., Venkatesan, S., Albanese, M., and Wellman, M. P. (2016). Moving Target Defense against DDoS Attacks: An Empirical Game-Theoretic Analysis. In *3rd ACM Workshop on Moving Target Defense*, pages 93–104. ResearchGate.

Yoon, S., Cho, J.-H., Kim, D. S., Moore, T. J., Free-Nelson, F., and Lim, H. (2020). Attack Graph-Based Moving Target Defense in Software-Defined Networks. *IEEE Transactions on Network and Service Management*, 17(3):1653–1668.

Zhuang, R., Zhang, S., DeLoach, S. A., Ou, X., Singhal, A., et al. (2012). Simulation-based approaches to studying effectiveness of moving-target network defense. In *National symposium on moving target research*, volume 246, pages 1–12. Citeseer, Citeseer.

Zscaler (2023). What is Deception Technology? Importance & Benefits| Zscaler.