IEEE Access
Multidisciplinary | Rapid Review | Open Access Journal

Received June 22, 2021, accepted July 2, 2021, date of publication July 6, 2021, date of current version July 15, 2021.

Digital Object Identifier 10.1109/ACCESS.2021.3095149

Maximizing Network Reliability to 0-Day Exploits Through a Heterogeneous Node Migration Strategy

YASMANY PRIETO, MIGUEL FIGUEROA, (Member, IEEE), AND JORGE E. PEZOA, (Member, IEEE)
Electrical Engineering Department, Universidad de Concepción, Concepción 4030000, Chile

Corresponding author: Yasmany Prieto (yprietoh@udec.cl)

This work was supported by the Agencia Nacional de Investigación y Desarrollo (ANID), FONDECYT POSTDOCTORADO 2020, under Grant 3200858.

ABSTRACT A recurrent problem currently affecting network reliability is the simultaneous exploitation of 0-day vulnerabilities shared between several node implementations across the network. When such 0-day vulnerabilities are exploited, large portions of the network may get compromised as a result. In this work, we propose a network node migration strategy to minimize the impact of 0-day attacks on network reliability. The migration method proposes replacing homogeneous node implementations with diverse alternatives to yield a heterogeneous network. The migration method allocates heterogeneous nodes within the network by minimizing the product between the average and the maximum number of network partitions, which may emerge after the simultaneous exploitation of 0-day risks on shared network resources. As we show, our migration strategy maximizes network connectivity in the event of a simultaneous 0-day attack. Our work's significant findings are the following: First, increasing the heterogeneity in node technologies reduces the attacker's ability to break down the entire network. Second, given a set of available network technologies that partially share risks, a network design implemented using several heterogeneous technologies sharing a small number of 0-day risks is more reliable than one with a small number of technologies whose 0-day risks are disjoint. Third, we observed that in a node-heterogeneous network topology, clustering nodes by technology improves network reliability.

INDEX TERMS Network diversity, network reliability, 0-day vulnerabilities, connected components.

I. INTRODUCTION

Reliability is nowadays a key criterion when designing a communication network [1]–[8]. The relevance of data networks in almost every socioeconomic aspect implies that the availability and quality of service are critical for the service providers and users. That ubiquitousness also has called the attention of attackers that seek to invalidate or capture network resources, and for that purpose, they develop new attack methodologies every day.

In that scenario, network administrators protect their networks with several mechanisms based on the information gathered to the day on attack patterns and disclosed critical vulnerabilities reported in databases such as [9]–[13]. However, even an updated system is still at risk to unknown 0-day exploits, which could be delivered in a massive and correlated manner. The term 0-day vulnerability refers to any software, hardware or firmware exposure unknown to vendors, administrators, and users, which an attacker could exploit until it is mitigated [14]–[18]. Such an attack can affect several network devices on the premise that they share the exploited vulnerability. This paper deals with shared vulnerabilities in firmware, operating systems, software applications, and protocol stacks implemented on network nodes. Vulnerabilities are compiled in professional lists of records such as The CVE List and The U.S. National Vulnerability Database. These record lists assign to each vulnerability an identification number, a description, and at least one known cybersecurity vulnerability [12]. For instance, the record CVE-2021-0275 is described as a cross-site scripting vulnerability affecting several versions of Juniper Networks Junos OS. An attack to such vulnerability could end up in complete control of

The associate editor coordinating the review of this manuscript and approving it for publication was Yu Liu.

the device [12]. The record CVE-2013-6026 is described as a firmware backdoor allowing remote attackers to several D-Link, Alpha Networks, and Planex routers. Lastly, the record CVE-2021-1520 corresponds to a privilege escalation (gaining root access) affecting four different Cisco VPN routers [10].

Although no one can prevent the damage inflicted by an unknown attack, there are methods to mitigate its extent. Two such mechanisms are software diversity and network diversity. Several works have been published on software diversity as a defense mechanism [19]–[22]. The main idea behind these works is to create different software instances with the same functionality but with a low probability of presenting the same faults. In this paper, we do not focus on developing different software instances but on benefiting from the existing ''natural'' heterogeneity in the technology market, as pointed out in [23]–[25].

On the other hand, most of the work on network diversity relies on the heterogeneous networking paradigm proposed in [26]. Zhang *et al.* observed that network technologies could present different network resources' implementations at each functional capability level. The more significant the heterogeneity between implementations, the smaller the number of shared vulnerabilities, and the higher the reliability to 0-day attacks. Implementing diverse technologies in the network prevents every node from being affected by exploiting a 0-day vulnerability. Thereby, a fraction of the network will keep working, albeit with a lower quality of service.

For a long time, introducing network diversity was an unapproachable problem due to compatibility, management, and economic constraints. Nevertheless, in the last decade or so, networking developments made it possible to coexist multivendor implementations in technologies such as Software-defined Networking (SDN) and Network Function Virtualization (NFV). Practical examples are The CloudNFV platform [27], the SDN-based Packet Transport Network operated by China Mobile [28], and the NFV-based Service Orchestration implemented by Anuta Networks [29]. These are suitable technologies to seize network diversity defense mechanisms against 0-day exploits.

In this work, we aim to improve the network's reliability in the face of simultaneous 0-day attacks to its nodes. Our main objective is to derive a methodology for maximizing network reliability in the face of 0-day exploits through the proper migration of heterogeneous nodes. More precisely, given a network topology, we propose to replace existing nodes with new ones, which are optimally selected from a set of heterogeneous technologies and placed onto the network to reduce the effect of network partitioning after a simultaneous 0-day attack. To do so, we model node technologies as a set of interchangeable resources, which allow the network to function correctly. Each resource could be shared by more than one node technology and could house a potential 0-day vulnerability.

The main contributions of this paper are the following:

- We develop a methodology to select and allocate diverse (heterogeneous) network node implementations by maximizing the network reliability in the face of 0-day simultaneous attacks.
- We propose an optimization function for network reliability that combines the average number and the maximum number of network partitions arising after 0-day attacks. Such combination effectively trades off the diversity in the number of migrated technologies and the number of mutually disjoint risks between them.
- We show how a higher degree of heterogeneity in network node implementation improves the reliability of migrated networks.
- We observe that in a node heterogeneous network topology, clustering nodes by technology improves network reliability.

The rest of the paper is organized as follows. Section II reviews the related work. Section III presents our model, formulates optimization problems, and discusses metrics and numerical evaluation conditions. Section IV discusses our results and Section V summarizes the paper.

II. RELATED WORK

In the literature, two network problems are tackled through network diversity, as observed in [30]. The first problem is malware propagation. Common vulnerabilities in neighbor nodes allow an attacker to employ the same tool to gain control of both nodes, easing how the attacker acquires targets in the network by propagation. To avoid the exploitation of 0-day shared vulnerabilities in such fashion, works like [14], [31]–[33] propose to diversify the network resources implemented in neighbor nodes. The models and problem statements presented in the works above improved our vision on network diversity methods.

Our work deals with the second problem, which is the simultaneous activation of vulnerabilities shared by some resources in network nodes. This event results in the impairment of nodes sharing the vulnerability and, potentially, the disconnection of the remaining functioning network. Several works focus on increasing network connectivity and robustness through diversity to tackle this problem, like [34]–[41]. In [34], Alleg *et al.* proposed a combination of redundancy and diversity mechanisms to meet a target Service Function Chain availability in an NFV framework. In the network, critical NFV instances were substituted by N thinner, diverse replicas that perform the same function and collectively process the same traffic as the original NFV. If some of the replicas fail, the rest will provide the network function, although with a lower service level. In [35]–[37], the authors proposed to maximize pairwise connectivity through the optimum placement of diverse variants to the network nodes. In [35], Newell *et al.* maximized the expected pairwise connectivity. They achieved a more resilient network by employing three variants, which fail independently and with different probabilities, instead of only the most reliable variant. In [36], Abbas *et al.* combined the optimum

placement of diverse node variants with the implementation of trusted nodes. They did not follow a probabilistic approach but sought to maximize pairwise connectivity in the worst-case attack scenario. They showed that employing the combination of trusted nodes and two variants was better than only using either or none of these defense mechanisms. Both works [35], [36] considered that each variant contains disjoint vulnerabilities. Ai *et al.* in [37] extended this model to include the case where specific variants shared vulnerabilities.

Besides connectivity metrics, the authors of [38]–[41] employed connected components metrics to analyze how well connected the network remained after failures. In [39], the authors proposed to design the network from scratch. They evenly divide the number of network nodes among the number of variants. Next, an edge is created from each node to another one from a different variant. So, if the nodes of one variant fail, the rest of the network remains in a single connected component. Caballero *et al.* [38] proposed two methods for allocating diverse variants to the nodes of an existing network. The best allocations resulted in clusters of nodes of the same variant. Such a pattern maximizes the connectivity among surviving nodes once a variant fails. Both works [38], [39] considered no difference in variant's reliability. Prieto *et al.* [40], [41] introduced a *vulnerability index*, which accounts for the different reliability of each variant. The network diversification problem in [40] was decomposed into two subproblems. The first subproblem optimally computes the number of nodes of each variant, according to the vulnerability index. The second subproblem allocates such nodes to the topology maximizing the network connectivity. Results showed a clusterization pattern similar to the one achieved in [38]. The work in [41] is a considerable extension of [40], where the authors take into account shared vulnerabilities among available technologies. They raised the following problem: Considering a pool of technologies that might share vulnerabilities, what is the largest number of technologies that do not share vulnerabilities? An important observation of this work was the joint influence of the average network degree and node diversity on the resulting post-failure network connectivity.

In this work, we employ a connected-components–based metric to describe the effect of diversity on network reliability. Historically, the study of network reliability has considered link failures, with utterly reliable nodes [7]. As a result, many connectivity metrics have been developed based on remaining paths between specific network nodes after failure events, as in [35]–[37]. Because network size changes according to the number of nodes that remain active after 0-day attacks, using such metrics could lead to deceiving results. The connected components metric employed in this work allows us to consider both the number of attacked nodes and the remaining functioning network connectivity after each 0-day attack event.

A common approach in addressing the network diversity problem has been defining those devices with distinct software versions, operating system (OS), or manufacturers, as vulnerability-disjoint implementations or variants [34]–[36], [38]–[40]. Such a definition implies that a specific 0-day vulnerability can only reach a particular implementation. However, works like [24], [25], and the vulnerability disclosure databases [9]–[13] reveal that apparently unrelated implementations could share common risks due to code reutilization, third-party software applications, etc. In the literature, other authors have considered shared vulnerabilities, such as [14], [31]–[33], [37], [41]–[43]. Some authors take into account shared vulnerabilities assessing the number of common vulnerabilities disclosed between the available technologies. However, when dealing with 0-day exploits, the model changes to consider common network resources or processes between node implementations as possible roots of unknown vulnerabilities.

We point out that average and worst-case scenario metrics are two common ways of assessing reliability. Examples of average metrics are the average pairwise connectivity [35], [37], the average attacking effort [14], [31], and the average normalized size of the largest component [38]. On the other hand, examples of worst-case scenario metrics are the worst-case pairwise connectivity [36], the least attacking effort [14], [31], and the minimum normalized size of the largest component [38]. As observed in [38], each of the metrics taken alone in assessing reliability is incomplete; that is why in our work, we consider a joint metric that incorporates both. This idea of carrying out a network design that looks after reliability, simultaneously optimizing the average and worst-case scenarios, can be found in Barábasi's book [44], which presents a network design that optimizes robustness to both random failures and targeted attacks.

Lastly, we comment that this work is a substantial improvement and extension of the preliminary version in [42]. The most significant extensions are: First, we change our definition of the Risk matrix. In [42], we based the risks on known shared-vulnerabilities. In this work, we associate 0-day risks to common resources or processes implemented in the network nodes. Second, we provide a suitable example to support the migration problem formulation with an objective function that considers both the average and the largest number of network partitions that arise after 0-day attacks. Third, we compare this approach to considering only the average or the worst-case term, and we find out our approach outperforms the other two. Fourth, we compare our shared-risk aware design to the common assumption in the state-of-the-art of using only technologies that present mutually disjoint risks in the design. Finally, we assess network robustness on the entire interval of heterogeneity metric values, from $h = 0$ (a monoculture network) to $h = 1$ (a risk-disjoint multiculture network). We showed that a more extensive heterogeneity of the available technologies contributes to a more reliable network.

III. METHODOLOGY

From the literature, we conclude that implementing a diverse set of technologies avoids the exploitation of a 0-day risk

impairs the entire network. Consequently, network reliability can be improved by migrating or redesigning its topology, including heterogeneous 0-day risk-aware nodes that do not lean on a unique technology.

Next, we introduce definitions and the theoretical framework used to define our methodology for network migration. Our methodology proposes a combinatorial optimization problem to select and locate current network node technologies onto a given network topology. The rationale behind the optimization function is explained using a simple yet informative case study.

A. PROBLEM STATEMENT

Network node migration in a data network is defined as the replacement of one or more nodes by other devices providing equivalent resources to the data communication topology, while maintaining the same number of nodes and the same links. The data network is modeled as the undirected graph $G = (V, E)$, where $V = \{1, 2, \ldots, n\}$ is the set of communication nodes and $E = \{(u, v) : \text{nodes } u \text{ and } v \text{ are connected}\}$ is the set of communication links between nodes. The ith network node is implemented by integrating R_i resources or processes, which run at each functional network level, from a pool of R different resources available in the market. Such a combination of resources is termed as a node implementation technology or, simply, a technology. Following the heterogeneous networking paradigm in [26], we assume that each network node can be implemented using one out of K different, yet equivalent, technologies. Thus, G is a monoculture or homogeneous data network if its n nodes are all implemented using a single technology; otherwise, G is a multiculture or heterogeneous data network. In practice, all the nodes in a monoculture network integrate the same OS, firmware version, protocol stack and are likely provided by the same vendor.

In this paper, we follow [45] to define data network reliability as the probability that the induced subgraph of surviving nodes is connected after 0-day risks are exploited. Regarding the failure model, each one of the R resources implementing a network node technology is prone to fail by exploiting a 0-day risk. The exploitation of a 0-day risk in a resource induces a correlated failure among all the network nodes sharing such resource. Consequently, a 0-day exploit impairs infrastructure connectivity by breaking down several network nodes simultaneously and for an extended period of time. Following [46], the dependency among technologies and their 0-day risks is represented through the K-by-R shared-risk matrix $\mathbf{X}$. For clarity, Table 1 shows an example of the shared-risk matrix $\mathbf{X}$ for the case of $K = 3$ different technologies and $R = 4$ resources, which can be used to implement the n network nodes. The element $X_{ir} = 1$ represents that a risk on the shared resource r affects the technology i and $X_{ir} = 0$ indicates otherwise. The example in Table 1 shows that Technology 3 can be impaired by exploiting 0-day risks R1 or R3. Consequently, all the nodes in G implemented using Technology 3 would fail in a correlated manner. We also note

TABLE 1. An example of the shared-risk matrix X describing the relationship between the shared resources (resources R1, R2, R3, and R4) and the available network node technologies (Technologies 1, 2, and 3).

	R1	R2	R3	R4
Tech 1	1	0	0	0
Tech 2	0	1	1	1
Tech 3	1	0	1	0

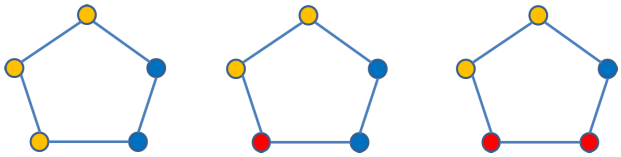


FIGURE 1. Sample configurations when the technologies in Table 1 are arbitrarily assigned to network nodes in a ring topology. From left to right: Configuration 1, Configuration 2, and Configuration 3. Node colors match the available technologies in Table 1.

that Technologies 1 and 3 share the 0-day risk R1, meaning that the exploitation of the R1 0-day will induce simultaneous failures in all the nodes implemented using either technology.

In practical terms, the 0-day vulnerability exploitation is a targeted attack since it is intended and directed to a group of network devices that share the same vulnerability. In this work, we consider two types of 0-day attacks: ''informed attacks'' and ''random attacks'' [47]–[50]. An informed attack has access to both a list of 0-day exploits and the network topology information. So, such attacks deliberately choose to exploit the vulnerability that maximally affects the network. Meanwhile, a random attack has access to an arbitrary exploit and executes it. Our migration strategy considers both attack types. A worst-case scenario metric is designed to assess reliability against informed attacks, and by contrast, an average metric is designed to deal with random attacks.

B. RATIONALE

Since exploiting 0-day risks impair network nodes and partitions the data network, we need to measure such impact. Our main idea is to assess the average and the maximal damage inflicted by 0-day attacks. Using a simple case study, we explain that both impacts must be regarded jointly to maximize network reliability through heterogeneous node migration.

Consider a small data network composed of five nodes connected in a ring topology as depicted in Fig 1. Suppose that, in a network migration process, nodes can be implemented using three different technologies and suppose that their resources (OS, firmware version, protocol stack, etc.) may be affected by four different 0-day risks. Figure 1 shows three possible technology-to-nodes mappings assigned to the five-node ring topology. For clarity, the node color identifies

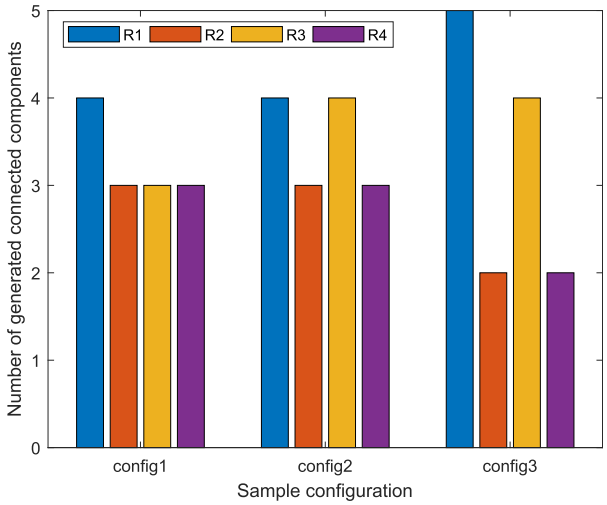


FIGURE 2. Number of network partitions that arise from the configurations from Fig. 1, as a 0-day vulnerability is exploited on each of the shared resources from Table 1. The C_{max} for Configurations 1, 2, and 3 are 4, 4, and 5, respectively. Also, the C_{avg} can be calculated for Configurations 1, 2, and 3 resulting in 3.25, 3.5, and 3.25, respectively. .

the technology used for its implementation, which matches the technology color in Table 1.

Under the exploitation of a given 0-day risk, the attacked nodes will stop operating, and the resulting network topology may either remain connected or break into several partitions. The number of network partitions emerging after triggering a 0-day attack can be measured using the number of connected components. The number of connected components, defined as the number of connected subgraphs in a network, is a standard way in network science to assess connectivity and robustness [44], [51]. For instance, under Configuration 1, an R1 0-day exploit results in four connected components (network partitions): the three yellow network nodes, which fail and become isolated, and the two blue network nodes, which remain functional and connected. Recall that a failed node is a singleton graph, and from network science, it is defined as a connected component [44], [51].

Figure 2 shows the number of network partitions that arise when 0-day risks are exploited on each shared technological resource for the network configurations in Fig. 1. Two statistics are used as metrics to analyze the shared risks' overall impact on the sample network: The maximum, $C_{\max}$, and the average, C_{avg}, number of network partitions arising after all 0-day attacks are exploited independently. The maximum number of network partitions indicates the worst-case scenario and the average number of network partitions measures the average damage inflicted on the network when 0-day exploits are triggered. We note that both $C_{\max}$ and C_{avg} belong to $[1, n]$ and the value 1 indicates that the network is connected, while larger values indicate otherwise.

From the results in Fig. 2, we can gain an insight into how a network migration process can generate network configurations coping against 0-day attacks. From the results, it is clear that Configuration 1 is the best because both metrics, the worst-case and the average number of network partitions, are the lowest. We also note that Configuration 2 exhibits a similar worst-case scenario as Configuration 1 in the face of 0-day exploits. However, Configuration 1 exhibits a lower average number of network partitions than Configuration 2 because, under the threat of an R3 0-day attack, Configuration 1 yields a more reliable topology. On the other hand, in the face of 0-day attacks, Configuration 3 yields the same average number of network partitions as Configuration 1. Nonetheless, such statistic hides the case that the network becomes entirely disconnected under the R1 0-day risk. Consequently, both metrics are required to effectively assess the impact of 0-day risks on the network topology.

We also note that broadly speaking, the more heterogeneous systems are the most reliable [26]. Thus, finding that Configuration 1 is the most reliable topology seems contradictory because using all the available technologies for node migration should lead to the most reliable configuration. However, technologies are composed of resources, which are the ones prone to fail under 0-day exploits. Moreover, one must also recall that different resources may share 0-day risks. Consequently, such dependencies must be taken into account when optimally migrating network nodes.

C. OPTIMAL NODE MIGRATION STRATEGY

In this work, we improve network reliability by minimizing the number of partitions induced by the surviving nodes after 0-day risks are exploited. Network reliability is maximal when the induced subgraph is connected, which means one partition or connected component emerges after failures. By minimizing the number of partitions arising after exploiting each 0-day risk, we increase the probability that post-failure network nodes are connected.

We propose now an optimal node migration strategy for maximizing network reliability. The strategy minimizes the product between the average and the largest number of network partitions arising after 0-day exploits impair communication nodes. We mathematically formulate the migration strategy in terms of the constrained integer optimization problem:

$$\begin{aligned}\mathbf{M}^* &= \underset{\mathbf{M}\in\mathcal{M}}{\operatorname{argmin}}\, C^2 \\ &= \underset{\mathbf{M}\in\mathcal{M}}{\operatorname{argmin}} \left(\frac{\sum_{r=1}^{R} f(G_r)}{R} \max_{r\in R} f(G_r) \right), \qquad (1) \\ &\quad \text{subject to: } \sum_{i=1}^{K} m_{ij} = 1 \quad j \in 1, 2, \ldots, n \qquad (2)\end{aligned}$$

where $\mathbf{M} = (m_{ij}) \in \{0, 1\}$ is the $K \times n$ migration matrix, which maps each network node to one out of the K available technologies, with $m_{ij} = 1$ mapping the jth node to the kth technology. $\mathcal{M}$ is the collection of all possible mappings for the nodes onto the K technologies. The subgraph $G_r(\mathbf{M}) \equiv G_r = (V_r, E_r) \subset (V, E)$ represents the migrated topology arising after the exploitation of the rth 0-day risk. $f(G_r) = C(G_r) - 1$, where $C(G_r)$ is the number of network

partitions (connected components) in G_r. We note that the set of equations in (2) imposes the constraint that a node has to be implemented with only one technology. Lastly, we note that the optimal migration allocation is given by the matrix $\mathbf{M}^*$.

We remark now how the optimization function considers the ideas in Section III-B. The function $f(G_r)$ aims to favor all the mapping yielding a single network partition after a 0-day exploit. The term $\frac{1}{R}\sum_{r=1}^{R} f(G_r)$ assesses the average number of network partitions arising after arbitrary 0-day risks exploitation by a random attack. Also, by minimizing the term $\max(f(G_r))$, we aim to favor those mappings inflicting the least amount of damage, in the worst-case scenario, after a 0-day risk is exploited by an informed attack. Besides, the optimization function can be thought of as an upper bound for the square of the average number of network partitions emerging after any shared 0-day risk is exploited.

D. BENCHMARKS, RELIABILITY METRICS, AND TEST NETWORKS

We compare our strategy with two methods that independently minimize the average (C_{avg}) and the largest number ($C_{\max}$) of network partitions arising after 0-day exploits impair communication nodes. More precisely, the optimization problems for the above mentioned strategies are given by:

$$\mathbf{M}^*_{avg} = \underset{\mathbf{M}\in\mathcal{M}}{\operatorname{argmin}} \left(\frac{\sum_{r=1}^{R} f(G_r)}{R} \right) \tag{3}$$

$$\text{subject to: } \sum_{i=1}^{K} m_{ij} = 1 \quad j \in 1, 2, \ldots, n \tag{4}$$

and,

$$\mathbf{M}^*_{max} = \underset{\mathbf{M}\in\mathcal{M}}{\operatorname{argmin}} \left(\max_{r\in R} f(G_r) \right) \tag{5}$$

$$\text{subject to: } \sum_{i=1}^{K} m_{ij} = 1 \quad j \in 1, 2, \ldots, n \tag{6}$$

We also compare our optimal node migration strategy with an approach usually followed in the heterogeneous network design literature [34]–[36], [38]–[41]. Such an approach supposes that heterogeneous networks should be implemented only with technologies exhibiting mutually disjoint risks. This assumption can be adopted by including additional constraints to (1) and (2) to obtain:

$$\mathbf{M}^* = \underset{\mathbf{M}\in\mathcal{M}}{\operatorname{argmin}} \left(\frac{\sum_{r=1}^{R} f(G_r)}{R} \max_{r\in R} f(G_r) \right), \tag{7}$$

$$\text{subject to: } \sum_{i=1}^{K} m_{ij} = 1 \quad j \in 1, 2, \ldots, n \tag{8}$$

$$\sum_{i=1}^{K} X_{ir} T_i \leq 1 \quad r \in 1, 2, \ldots, R \tag{9}$$

$$T_i - \sum_{j=1}^{n} m_{ij} \leq 0 \quad i \in 1, 2, \ldots, K \tag{10}$$

$$\sum_{j=1}^{n} m_{ij} - nT_i \leq 0 \quad i \in 1, 2, \ldots, K \tag{11}$$

where $T_i = 1$ indicates that the ith technology is used during the network migration and $T_i = 0$ indicates otherwise. The mutually disjoint 0-day–risk approach is formulated in the set of constraints (9), where for every resource only one migrated technology can integrate such resource. In addition, the set of constraints (10) and (11) ensures that $T_i = 1$ only if at least one of the nodes from the network is migrated with technology i, otherwise $T_i = 0$.

In addition, we compare our approach with a risk-unaware migration method. In such a method, network designers incorrectly assume that the K available technologies are risk-disjoint when actually they are not. This case, for instance, models the situation where the risk matrix is unknown to network designers. The risk-unaware migration method is formulated as in (1) and (2), with the dummy risk matrix $\mathbf{X} = \mathbf{I}$, where $\mathbf{I}$ is the identity matrix.

The post-failure network reliability will be assessed in terms of two metrics related to network connectivity: the number of connected components and the Average Two-Terminal Reliability (ATTR). The number of connected components arising after the exploitation of 0-day risks is typically used in network science to assess the effect of failing nodes. This metric takes into account the number of failed nodes and the disconnectedness of the remaining functioning network. The number of connected components is a coarse-grained metric because it provides a global idea of how connected a network remains after failure. The ATTR figure of merit quantifies the likelihood that a random pair of network nodes is connected [8], [52], [53]. The ATTR of a data network when the rth 0-day risk is exploited is computed as:

$$\text{ATTR}(r) = \binom{n}{2}^{-1} \sum_{u\neq v} Z^r_{uv}, \tag{12}$$

where $\binom{n}{2}$ is the binomial coefficient and the binary variable $Z^r_{uv} = 1$ if there is a path between the nodes u and v after the exploitation of the rth 0-day risk, and $Z^r_{uv} = 0$ otherwise. The average ATTR of the network is computed as the weighted average over all the 0-day risks failures:

$$\text{ATTR} = \sum_{r=1}^{R} \alpha_r \ \text{ATTR}(r), \tag{13}$$

where α_r is the likelihood of exploiting rth 0-day risk. We note that if the network is connected, the number of connected components is 1, and the ATTR is also 1. Unlike the number of connected components, the ATTR is a fine-grained metric because it assesses the likelihood of connecting every pair of nodes that remain functional after a failure. In this work, we compute the ATTR metric after the occurrence of 0-day exploits in two ways. The $ATTR_1$ figure considers all the network nodes (failed and working nodes) in the migrated topology after 0-day risks are exploited. Therefore,

to compute $ATTR_1$ we use (12) and (13). We note that when nodes fail, the $ATTR_1$ metric is severely degraded because such nodes are no longer reachable. Thus, to accurately assess the post-attack network connectivity, we calculate $ATTR_2$ among all the functioning nodes after 0-day attacks. We define $ATTR_2$ and $ATTR_2(r)$ as:

$$ATTR_2 = \sum_{r=1}^{R} \alpha_r \, ATTR_2(r), \tag{14}$$

$$ATTR_2(r) = \binom{n_r}{2}^{-1} \sum_{u \neq v} Z_{uv}^r, \tag{15}$$

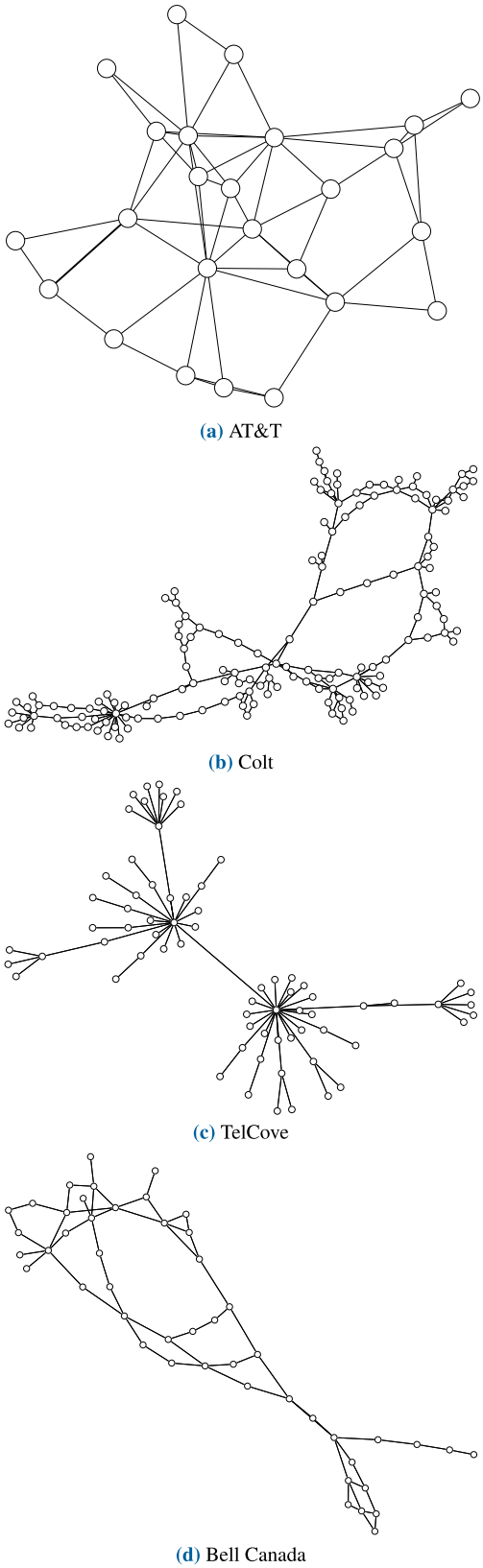
where n_r is the number of nodes that remain functional after the rth 0-day risk attack. To find n_r we compute the matrix $\mathbf{P} = \mathbf{X}'\mathbf{M}$, where $\mathbf{P} = (p_{ij}) \in \{0, 1\}$ is the $R \times n$ matrix composed of the 0-day risks that are present in each node, with $p_{ij} = 1$ means that the jth node contains the ith 0-day risk. Then, $n_r = n - \sum_{j=1}^{n} p_{ij}$. Consequently, $ATTR_2 = 1$ means that all the functioning nodes can be reached from other working nodes after the attacks.

Besides, to assess the degree of heterogeneity in the solutions found during the migration process, we employ the technology set risk heterogeneity metric [42]. This metric, which is denoted as h, computes the cumulative dissimilarity between each pair of 0-day risks associated to the technologies using:

$$h = 1 - \frac{1}{R\binom{K}{2}} \sum_{i=1}^{K} \sum_{j=i+1}^{K} \mathbf{r}_i \mathbf{r}_j^T, \tag{16}$$

where $\mathbf{r}_i$ and $\mathbf{r}_j$ are, respectively, the ith and jth rows of the risk matrix $\mathbf{R}$. The h metric takes values in (0, 1), where the value "0" (respectively, "1") indicates that all the technologies share every 0-day risk (respectively, do not share any 0-day risk) among the implementations. Note that when $h = 1$, a heterogeneous network is implemented with technologies exhibiting mutually disjoint risks.

Finally, we test the proposed design methodology using the four real-world IP network topologies depicted in Fig. 3. These networks correspond to backbone IP topologies that are regularly used as benchmarks in the research community. The topologies were extracted from Internet Topology Zoo [54]. These networks exhibit different sizes and degree distribution laws. More precisely, the AT&T network topology follows a random network degree distribution, the TelCove network follows a scale-free degree distribution law, while the Colt and the Bell Canada network are somewhere in between. The number of nodes, links, and graph density of each test network are summarized in Table 2. Graph density is defined as the ratio between the number of network links and the number of possible links, i.e., $\Delta = |E|/\binom{n}{2}$. Values of Δ closer to 1 represent better-connected and more robust network topologies.



(a) AT&T

(b) Colt

(c) TelCove

(d) Bell Canada

FIGURE 3. The four real-world, homogeneous network topologies evaluated using heterogeneous network migration strategies.

TABLE 2. Testing networks characteristics.

Network	nodes	links	Δ
AT&T	25	56	0.187
Bell Canada	48	64	0.057
TelCove	71	70	0.028
Colt	153	177	0.015

TABLE 3. The average risk heterogeneity h_{avg} of risk matrices, which are generated sampling Bernoulli random variables with different p parameter.

h_{avg}	0.1	0.2	0.3	0.4	0.5	0.6	0.7	0.8	0.9
p	0.98	0.94	0.86	0.78	0.71	0.63	0.54	0.46	0.31

TABLE 4. The number of technologies, K, available for network migration for the different risk heterogeneity values, h.

h	0	0.1	0.2	0.3	0.4	0.5	0.6	0.7	0.8	0.9	1
K	1	2	6	8	9	10	10	10	10	10	10

IV. NUMERICAL EVALUATION AND DISCUSSION

We run three evaluations to assess our approach's ability to maximize network reliability through heterogeneous node migration in the presence of 0-day exploits. The first evaluation compares three network design methodologies to cope with 0-day exploits. The second evaluation compares our migration strategy to those assuming that multiculture networks should only be migrated using technologies containing mutually disjoint 0-day risks. In the third evaluation, we assess the allocation of different technologies onto the network nodes shown in Fig. 3.

Our numerical evaluation employed the test topologies in Fig. 3 and regarded a set of $R = 10$ different network resources, which available technologies could implement. We considered eleven different classes of 0-day risks, which induce eleven shared-risk matrices with distinct risk heterogeneity levels. A different risk heterogeneity value characterizes each risk matrix, ranging from $h = 0$ (a single technology with a known 0-day risk) to $h = 1$ (K heterogeneous technologies sharing up to ten disjoint 0-day risks). In our evaluations, h varies uniformly in increments of 0.1. For $h \in [0.1, 0.9]$, we randomly generated risk matrices by sampling a Bernoulli random variable, with a p parameter controlling the occurrence of 0-day risk among the technologies. Table 3 lists the p parameter values generating, on average, risk matrices for each heterogeneity value required in the migration scenarios. We note that for $p = 0.45$, the average number of 0-day risks per technology is 4.5, a value consistent with those found in [10] and [11]. Additionally, Table 4 lists the number of available technologies for the eleven risk matrices. Also, in this work, we considered that the distribution of exploiting a 0-day is uniform, i.e., $\alpha_r = 1/R$.

The combinatorial optimization problems presented in (1), (7), (3), and (5) were solved in this work using a methodology based on Genetic Algorithms (GAs). This technique is a randomized, search-based, global optimization algorithm frequently employed to solve network or integer complex problems [55]–[59]. Section IV-D includes a case study to assess how closely a GA solution is to the optimal solution. For all the problems, we coded chromosomes in an integer-valued string of length n to match the number of network nodes; thereby, the jth position in a chromosome represents the jth node in the topology. The allocation of a particular technology, say k, to the jth node is determined by setting the value k in the jth position of a chromosome. Regarding the GA hyperparameters, we employed single-point crossover, executed with a probability of 0.8, as the crossover operator. For the mutation operator, a position in the chromosome is randomly selected, and its value is replaced by another technology, with a probability of 0.01. The fitness proportional selection was employed as a selection operator, implemented by a roulette wheel. The chromosome coding handles the sets of equality constraints (2), (8), (4), and (6). Lastly, inequality constraints (9), (10), and (11) were added as penalty functions to the objective function (7).

A. EVALUATION 1: COMPARING MIGRATION STRATEGIES

Figures 4 and 5 depict results from the first set of evaluations, which migrate nodes in the AT&T topology. Specifically, Fig. 4 compares the largest number of network partitions arising after 0-day risks are exploited, as a function of the risk heterogeneity metric, for the proposed network migration method (method 1) and those that independently minimize the average (method 2) and the largest number of network partitions (method 3). The first observation we can draw is that, regardless of the migration method, building a network with heterogeneous nodes reduces the worst-case scenario because the maximum number of network partitions decreases in the face of 0-day exploits. In other words, a heterogeneous network reduces the maximum amount of disconnectedness inflicted after 0-day risks are exploited. Second, as expected, the method minimizing the largest number of network partitions emerging after 0-day risks are exploited outperforms the other two methods. We note that the proposed network migration scheme achieves the second-best result. Third, as stated early in our rationale, migrating a network by minimizing the average number of partitions yields the worst results. Fourth, from Fig. 4 we can observe two different sets of results. When technologies available for migration are relatively risk homogeneous ($h \leq 0.4$) or highly risk heterogeneous ($h > 0.9$), no noticeable differences in the network reliability can be observed between the three methods. When a certain degree of risk heterogeneity is introduced ($0.6 \leq h \leq 0.9$), then differences in the network reliability can be noticed.

Figure 5 compares the average number of network partitions arising after 0-day risks are exploited, as a function of the risk heterogeneity metric, for the above-mentioned migration methods. We observe again that, as the network-node risk heterogeneity increases, so it does the performance of the migration method, achieving more reliable network designs. We also note that for fairly risk homogeneous ($h < 0.3$) or highly risk heterogeneous ($h > 0.9$) migrated networks, no noticeable differences in the network reliability can be

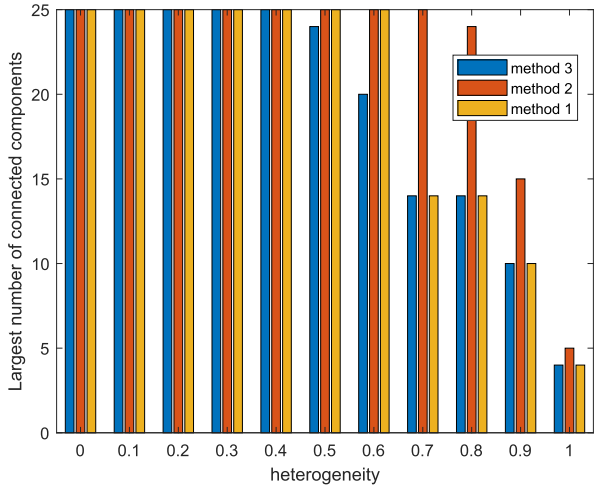


FIGURE 4. Network reliability assessed in terms of C_{max}, the largest number of network partitions (the largest network disconnectedness), as a function of the risk heterogeneity h, after the exploitation of 0-days risks in shared resources.

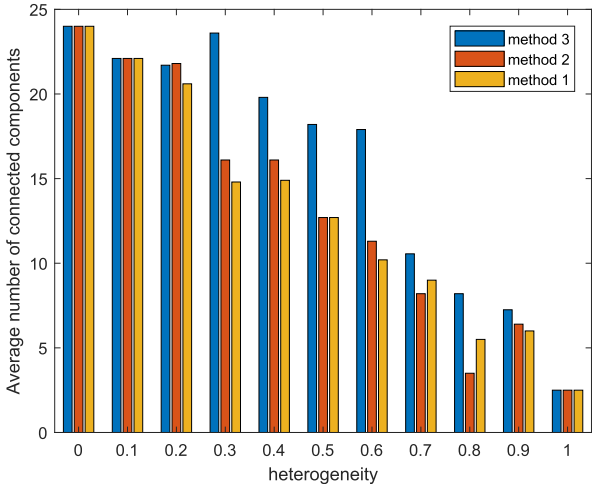


FIGURE 5. Network reliability assessed in terms of C_{avg}, the average number of network partitions (the average network disconnectedness), as a function of the risk heterogeneity h, after the exploitation of 0-days risks in shared resources.

observed between methods, and when a certain degree of risk heterogeneity is introduced ($0.3 \leq h \leq 0.9$), the migration methods produce noticeable differences in the network reliability. Unlike the previous case, we expect that migrated networks using the method in (3) should achieve the best results. However, it can be seen that, in several cases, networks migrated using the proposed method reach the maximal reliability. This result arises from the fact that GAs do not guarantee optimality. On the other hand, as expected, the lowest degree of reliability was achieved by the method minimizing the worst-case scenario.

We remark now that, as observed from Figs. 4 and 5, the proposed method effectively improves the AT&T network reliability in the presence of 0-day exploits. First, note that if an attacker gathers a priori information on which exploits would maximally damage network connectivity, our

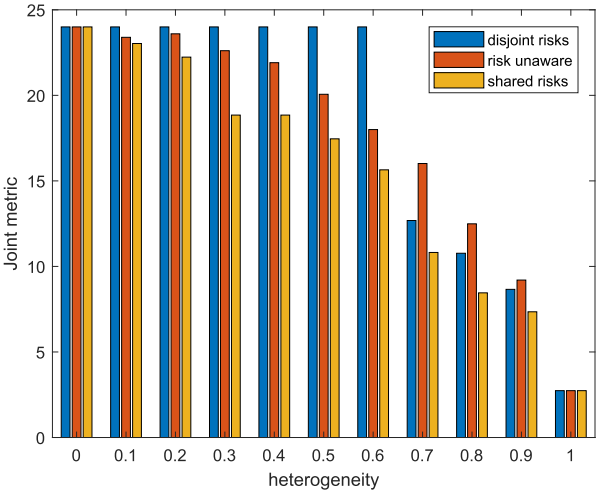


FIGURE 6. Network reliability assessed in terms of the joint metric $\sqrt{C_{max}C_{avg}}$, as a function of the risk heterogeneity h, after the exploitation of 0-days risks in shared resources.

migration method yields a small number of post-attack network partitions in such scenario. Second, in the case of "blind attacks" or random exploits, the network's average connectivity, based on the number of post-failure network partitions, reaches a good performance. For the other testing networks the migration results were consistent with those achieved in the AT&T network.

B. EVALUATION 2: IMPACT OF SHARED RISKS ON NETWORK RELIABILITY

In the second evaluation, we aim to understand whether relaxing the migration design constraint about technologies offering mutually disjoint 0-day risks would improve network reliability. Also, we want to comprehend the implications of overlooking the risk matrix and incorrectly assuming 0-day disjoint risks among the technologies. Figure 6 shows the optimal value for the objective function in (1), in terms of the risk heterogeneity for AT&T network. Recall that this optimization function is the product between the maximum and the average number of partitions emerging after 0-day risks are exploited and can be thought of as an upper bound for the square of the average number of such network partitions. The figure shows the optimal values, $\sqrt{(C^2)^*}$, for three methodologies: The proposed one (shared risks), another incorrectly disregarding the existence of shared 0-day risks, and another regarding only mutually disjoint 0-day risks.

As in the first evaluation, we note that increasing risk heterogeneity favors the increase of post-failure connectivity. Results show that our network migration method achieves topologies with maximal reliability in all cases. This is attributed to the fact that the approach acknowledges shared 0-day risks among technologies and employs this as leverage. The migration method constrained to employ only disjoint-risks technologies performs as well as the proposed method only under such scenario: $h = 1$. The difference

between the migrated topologies achieved by these two methods relies on the disjoint-risks constraint: the search space of solutions in the disjoint-risks method is a subspace of our proposed method. We note that, for $h \leq 0.6$, the disjoint-risks migration method did not achieve any solution other than using a single technology (homogeneous network), meaning that the optimal value for the objective function happens to be the worst: 24. This, in turn, leads to an entirely disconnected AT&T network. This statement is supported by the migration method incorrectly assuming a 0-day disjoint risk matrix. In such a case, by simply dropping the disjoint-risks constraint, the method yields heterogeneous topologies when the risk heterogeneity index is less than 0.7. For $h \geq 0.7$, the incorrect assumption about disjoint 0-day risks produces the smallest network reliability.

C. EVALUATION 3: IMPACT OF NODE MIGRATION PATTERNS ON NETWORK RELIABILITY

For the third evaluation, we choose a single 0-day risk matrix with a high degree of risk heterogeneity ($h = 0.9$) and migrate nodes to the network topologies in Fig. 3. The risk matrix is:

$$\mathbf{X} = \begin{pmatrix} 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 1 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 1 \\ 1 & 0 & 1 & 1 & 0 & 0 & 1 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 1 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 1 \\ 1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 & 1 & 0 & 1 & 1 & 1 & 0 \\ 0 & 0 & 0 & 1 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 1 & 1 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 & 1 & 1 & 0 & 1 & 1 \end{pmatrix}. \quad (17)$$

From $\mathbf{X}$, some technologies share resources, the number of resources per technology varies from 2 to 5, and the maximum number of 0-day risk-disjoint technologies is three (see, for instance, the set $\{Tech_5, Tech_6, Tech_8\}$).

The results achieved by the method are presented in Tables 5 and 6 as well as in Figs. 7 to 10. The network node migration results are analyzed next using two not independent perspectives: The technologies-to-deploy selection and the technology-to-node assignment.

From the technologies-to-deploy selection perspective, we can observe in Table 5 that most of the migrated nodes were implemented using the risk-disjoint technologies $Tech_5$, $Tech_6$, and $Tech_8$. However, for the Bell Canada and Colt networks, we noted an increase in the percentage of migrated nodes implemented using $Tech_1$ and $Tech_4$ compared to the other two networks. This behavior is partially explained by observing the 0-day risk matrix (17). $Tech_4$ implements only three resources with 0-day risks and shares only one with $Tech_5$, while $Tech_1$ exhibits only two 0-day exploitable resources. As a result of our migration methodology, we observe the preferential selection of a large number of technologies exhibiting a reduced number of shared exploitable resources. This result implies that the exploitation of a particular 0-day risk only would affect a small portion of the network.

From the technology-to-node assignment perspective, we base our discussion on Figs. 7 to 10. A different color

TABLE 5. Percentage of migrated nodes implemented using each available technology, for each testing network, regarding a 0-day risk matrix with $h = 0.9$.

Available technology	Nodes in AT&T (%)	Nodes in TelCove (%)	Nodes in Bell Canada (%)	Nodes in Colt (%)
$Tech_1$	4.0	4.2	8.3	10.5
$Tech_2$	0.0	0.0	4..2	3.9
$Tech_3$	0.0	1.4	2.1	0.7
$Tech_4$	4.0	1.4	8.3	13.7
$Tech_5$	24.0	23.9	18.8	18.3
$Tech_6$	32.0	35.2	31.3	28.8
$Tech_7$	4.0	1.4	2.1	0.7
$Tech_8$	32.0	32.4	22.9	20.9
$Tech_9$	0.0	0.0	0.0	0.7
$Tech_{10}$	0.0	0.0	2.1	2.0

TABLE 6. ATTR values for the migrated networks in Fig. 3, regarding a 0-day risk matrix with $h = 0.9$.

Network	$ATTR_1$	$ATTR_2$	Δ
AT&T	0.59	1.0	0.187
Bell Canada	0.58	0.94	0.057
TelCove	0.57	0.82	0.028
Colt	0.47	0.72	0.015

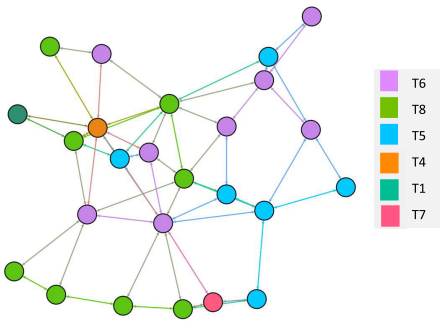


FIGURE 7. Technology allocation for the migrated AT&T network regarding a 0-day risk matrix with $h = 0.9$.

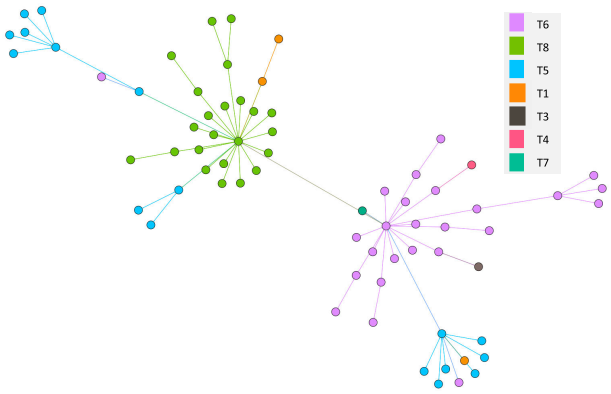


FIGURE 8. Technology allocation for the migrated Telcove network regarding a 0-day risk matrix with $h = 0.9$.

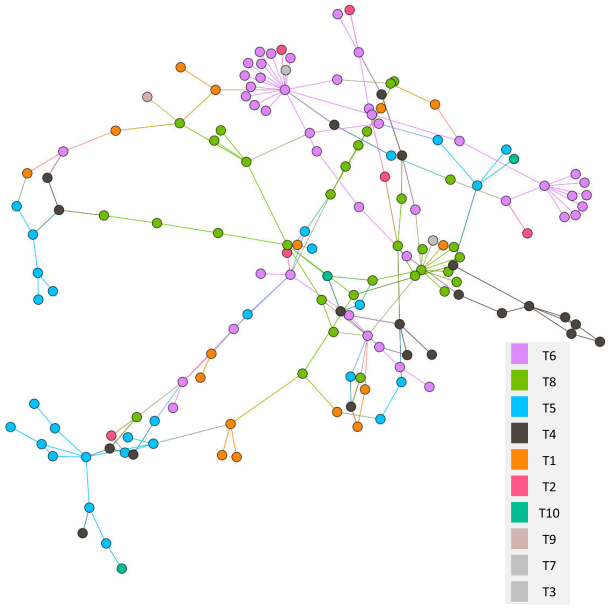


FIGURE 9. Technology allocation for the migrated Colt network regarding a 0-day risk matrix with $h = 0.9$.

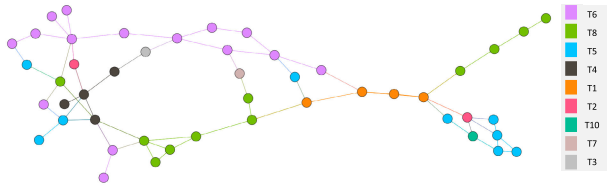


FIGURE 10. Technology allocation for the migrated Bell Canada network regarding a 0-day risk matrix with $h = 0.9$.

is associated with each technology of the risk matrix (17). For instance, blue represents technology 5, purple represents technology 6, and green represents technology 8. The most evident pattern observed in the migrated topologies is that technologies are allocated to nodes so that if a specific 0-day attack impairs a technology, it minimally disconnects the rest of the functioning network.

We note that a particular pattern arises in the case of the TelCove network. According to its degree distribution, the TelCove network follows a scale-free law. Thus, such a network contains hub nodes, which carry a significant responsibility on network connectivity. As shown in Fig. 8, the three most connected hubs in the migrated TelCove network are implemented with different, risk-disjoint technologies. As a consequence of this smart migration, a single 0-day attack could only impair one of them. We can observe another remarkable migration pattern: technological clustering, where migrated nodes connected to the hubs are implemented with the same hub technology. To understand this result, we must note that a hub node's impairment after exploiting a 0-day risk would minimally affect the other hubs. However, the migrated leaf nodes directly connected to a 0-day exploited hub would end up as a single network partition, whichever technology they implement. According to that and the available technologies in this evaluation, our migration methodology reproduces this clustering effect. The other three networks in Figs. 7, 9, and 10 also present a lighter but noticeable clustering effect.

To assess the allocation process in terms of node connectivity, we calculated the ATTR metric through $ATTR_1$ and $ATTR_2$. The results are listed in Table 6. We remark on the following ideas. First, denser networks (higher Δ) reach higher values for $ATTR_1$ and $ATTR_2$ because networks with more links can cope better with node removals. Second, 0-day attacks, while disrupting several nodes, minimally impair functioning-node connectivity. Third, the node technology allocation in the migrated AT&T network reduced each 0-day attack to impairing a group of nodes, while the rest of the functioning nodes remained connected. Thus, according to our results, a combination of a fair amount of network connectivity and technology heterogeneity is necessary to achieve higher network reliability.

D. ON THE ACCURACY OF GA SOLUTIONS

The optimization problem stated in (1) can be analytically solved if the network under analysis is fully connected and the risk matrix is composed of risk-disjoint technologies. In this scenario, after exploiting a 0-day risk, the network remains fully connected. Besides, the number of connected components is equal to the number of failed nodes plus one (the set of working nodes that remains fully connected). Thus, in this scenario, the optimal solution to the problem in (1) is to assign the available technologies onto the network nodes evenly. For example, the optimal solution for a fully connected network with ten nodes and two risk-disjoint technologies is implementing five nodes using the first technology and five with the second technology. To reach the above-stated conclusion, we can observe that in the objective function:

$$C^2 = \frac{\sum_{r=1}^{R} f(G_r)}{R} \max_{r \in R} f(G_r), \tag{18}$$

the average term will be a constant:

$$\frac{\sum_{r=1}^{R} f(G_r)}{R} = \frac{\sum_{r=1}^{R} \eta_r}{K} = \frac{n}{K}, \tag{19}$$

where η_r is the number of nodes implemented with the rth technology, and K is the number of risk-disjoint technologies. Then, to minimize (18), it suffices to solve: $\min(\max_{r \in R} f(G_r))$, which is achieved by assigning the same number of nodes to each technology. Thus, in the proposed scenario, the minimum value is $(C^2)^* = (\frac{n}{K})^2$.

Table 7 compares the exact optimal solution for the problem in (1) with the results achieved by the proposed GA for fully connected networks of 10-200 nodes and ten risk-disjoint technologies ($h = 1$). We also calculated the relative error produced by the GA in finding the optimal solutions. The relative errors are calculated as the absolute error between the exact and the GA solutions, divided by the exact solution.

TABLE 7. Exact and GA approximate solutions to the problem in (1) with different sizes fully connected networks.

nodes	Exact result	GA result
10	1	1
20	4	4
50	25	30
100	100	110
150	225	240
200	400	400

Table 7 shows that, for network sizes between 50 and 150 nodes, the GA does not find the optimal solution by assigning one extra node to a particular technology instead of assigning an even distribution. For example, when $n = 50$, $(C^2)^* = 30 = 5 \cdot 6$. The solutions delivered by the GA technique have a maximum relative error equal to 17% compared to the exact solutions, and on average, the relative error was 5,3%. For larger network sizes, the relative error decreases. According to these results, we can expect that the solutions to our optimization problems achieved by the GA are sufficiently close to optimal.

V. CONCLUSION

In this work, we propose a network node migration method to increase network reliability through technology heterogeneity. We developed a formulation to translate available technology heterogeneity into network reliability by minimizing the impact of simultaneous 0-day attacks due to shared resources in network nodes. The network is migrated or redesigned through an optimization problem with a clear rationale: Choose several technologies from a diverse set and allocate them onto the network nodes in a way that prevents large portions of the network from failing simultaneously.

In our results, we observed that jointly minimizing the worst-case and the average number of post-failure network partitions leads to a migrated network that copes better with 0-day exploits than independently optimizing each case. We also observed that the more heterogeneous the available set of migration technologies, the lesser the effect of 0-day risk exploitation on network connectivity.

We compared our migration methodology with an approach usually followed in the literature that constrains migrated nodes from sharing any risk with other nodes. We showed that allowing technologies that partially share risks to be part of the node migration improves network reliability compared to implementations allowing only disjoint risk technologies. Also, we observed that migrating a network using a diverse set of technologies results in improved network reliability compared to a monoculture, even when 0-day shared risks are incorrectly estimated.

Lastly, we assessed our migration methodology using four real-world IP backbone network topologies with different sizes and connectivity degrees. In the four migrated networks, we noted two common topological features: they were implemented using a large number of available technologies, which present a small number of shared 0-day risks, and network nodes were allocated in technological clusters. Both features contribute to maximizing network connectivity in the event of simultaneous 0-day attacks.

REFERENCES

[1] S. Bijwe, F. Machida, S. Ishida, and S. Koizumi, "End-to-end reliability assurance of service chain embedding for network function virtualization," in *Proc. IEEE Conf. Netw. Function Virtualization Softw. Defined Netw. (NFV-SDN)*, Nov. 2017, pp. 1–4.
[2] X. Gao, Z. Ye, J. Fan, W. Zhong, Y. Zhao, X. Cao, H. Yu, and C. Qiao, "Virtual network mapping for multicast services with max-min fairness of reliability," *J. Opt. Commun. Netw.*, vol. 7, no. 9, pp. 942–951, Sep. 2015.
[3] L. Qu, C. Assi, K. Shaban, and M. J. Khabbaz, "A reliability-aware network service chain provisioning with delay guarantees in NFV-enabled enterprise datacenter networks," *IEEE Trans. Netw. Service Manage.*, vol. 14, no. 3, pp. 554–568, Sep. 2017.
[4] Z. Zhang, J. Zhang, and T. Huang, "The reliability mapping monitoring method of network function virtualization," in *Proc. IEEE 4th Int. Conf. Comput. Commun. (ICCC)*, Dec. 2018, pp. 555–559.
[5] S. Xiang and J. Yang, "K-terminal reliability of ad hoc networks considering the impacts of node failures and interference," *IEEE Trans. Rel.*, vol. 69, no. 2, pp. 725–739, Jun. 2020.
[6] I. B. Gertsbakh and Y. Shpungin, *Models Networking Reliability: Analysis, Combinatorics*. Boca Raton, FL, USA: CRC Press, 2016.
[7] H. Pérez-Rosés, "Sixty years of network reliability," *Math. Comput. Sci.*, vol. 12, no. 3, pp. 275–293, Sep. 2018.
[8] S. Neumayer and E. Modiano, "Network reliability under geographically correlated line and disk failure models," *Comput. Netw.*, vol. 94, pp. 14–28, Jan. 2016.
[9] Wired Business Media. *Security Week: Internet and Enterprise Security News, Insghts, and Analysis, 2020*. Accessed: Feb. 25, 2021. [Online]. Available: http://www.securityweek.com/
[10] M. Horowitz. *Router Security, 2020*. Accessed: Feb. 25, 2021. [Online]. Available: https://routersecurity.org/
[11] Rapid7. *Rapid7 Exploit Database, 2020*. Accessed: Feb. 25, 2021. [Online]. Available: https://www.rapid7.com/db/modules/
[12] CVE List. *CVE Details, 2020*. Accessed: Feb. 25, 2021. [Online]. Available: https://www.cvedetails.com/
[13] Carnegie Mellon University. *Software Engineering Institute The Cert Division, 2020*. Accessed: Feb. 25, 2021. [Online]. Available: https://www.cert.org/
[14] D. Borbor, L. Wang, S. Jajodia, and A. Singhal, "Optimizing the network diversity to improve the resilience of networks against unknown attacks," *Comput. Commun.*, vol. 145, pp. 96–112, Sep. 2019.
[15] M. Zhang, L. Wang, S. Jajodia, A. Singhal, and M. Albanese, "Network diversity: A security metric for evaluating the resilience of networks against zero-day attacks," *IEEE Trans. Inf. Forensics Security*, vol. 11, no. 5, pp. 1071–1086, May 2016.
[16] L. Bilge and T. Dumitras, "Before we knew it: An empirical study of zero-day attacks in the real world," in *Proc. ACM Conf. Comput. Commun. Secur.*, 2012, pp. 833–844.
[17] M. Zhang, L. Wang, S. Jajodia, and A. Singhal, "Evaluating the network diversity of networks against zero-day attacks," *Netw. Secur. Metrics*, pp. 117–140. Springer, 2017.
[18] L. Wang, M. Zhang, S. Jajodia, A. Singhal, and M. Albanese, "Modeling network diversity for evaluating the robustness of networks against zero-day attacks," in *Proc. Eur. Symp. Res. Comput. Secur.* Cham, Switzerland: Springer, 2014, pp. 494–511.
[19] A. Avizienis, "The N-version approach to fault-tolerant software," *IEEE Trans. Softw. Eng.*, vols. SE–11, no. 12, pp. 1491–1501, Dec. 1985.
[20] J. E. Just and M. Cornwell, "Review and analysis of synthetic diversity for breaking monocultures," in *Proc. ACM Workshop Rapid Malcode*, 2004, pp. 23–32.
[21] J. C. Knight, "Diversity," in *Dependable Historic Computing*. Berlin, Germany: Springer, 2011, pp. 298–312.
[22] P. Larsen, A. Homescu, S. Brunthaler, and M. Franz, "SoK: Automated software diversity," in *Proc. IEEE Symp. Secur. Privacy*, May 2014, pp. 276–291.
[23] B. Baudry and M. Monperrus, "The multiple facets of software diversity: Recent developments in year 2000 and beyond," *ACM Comput. Surveys*, vol. 48, no. 1, pp. 1–26, Sep. 2015.
[24] J. Han, D. Gao, and R. H. Deng, "On the effectiveness of software diversity: A systematic study on real-world vulnerabilities," in *Proc. Int. Conf. Detection Intrusions Malware, Vulnerability Assessment*. Berlin, Germany: Springer, 2009, pp. 127–146.

[25] M. Garcia, A. Bessani, I. Gashi, N. Neves, and R. Obelheiro, "OS diversity for intrusion tolerance: Myth or reality?" in *Proc. IEEE/IFIP 41st Int. Conf. Dependable Syst. Netw. (DSN)*, pp. 383–394. IEEE, 2011.
[26] Y. Zhang, H. Vin, L. Alvisi, W. Lee, and S. K. Dao, "Heterogeneous networking: A new survivability paradigm," in *Proc. Workshop New Secur. Paradigms*, 2001, pp. 33–39.
[27] CloudNFV. *Taking NFV to the Cloud, 2018*. Accessed: Aug. 25, 2018. [Online]. Available: http://cloudnfv.com/
[28] Open Networking Foundation. *Use Cases for Carrier Grade SDN*. Accessed: May 29, 2019. [Online]. Available: https://opennetworking.org/wp-content/uploads/2014/10/TR-538_Use_Cases_for_Carrier_Grade_SDN.pdf
[29] Anuta Networks. *Case Study Network Service Orchestration for Multi-Vendor NFV, 2016*. Accessed: May 29, 2019. [Online]. Available: https://anutanetworks.com/wp-content/uploads/2016/06/Case-Study-Network-Service- Orchestration-for-Multi-Vendor-NFVI.pdf
[30] B. E. Helvik, P. Vizarreta, P. E. Heegaard, K. Trivedi, and C. Mas-Machuca, "Modelling of software failures," in *Guide to Disaster-Resilient Communication Networks*. Cham, Switzerland: Springer, 2020, pp. 141–172.
[31] T. Li, C. Feng, and C. Hankin, "Scalable approach to enhancing ICS resilience by network diversity," in *Proc. 50th Annu. IEEE/IFIP Int. Conf. Dependable Syst. Netw. (DSN)*, Jun. 2020, pp. 398–410.
[32] I. Marsa-Maestre, J. M. Gimenez-Guzman, D. Orden, E. de la Hoz, and M. Klein, "REACT: Reactive resilience for critical infrastructures using graph-coloring techniques," *J. Netw. Comput. Appl.*, vol. 145, Nov. 2019, Art. no. 102402.
[33] C. Huang, S. Zhu, and R. Erbacher, "Toward software diversity in heterogeneous networked systems," in *Proc. Annu. Conf. Data Appl. Secur. Privacy*. Berlin, Germany: Springer, 2014, pp. 114–129.
[34] A. Alleg, T. Ahmed, M. Mosbah, and R. Boutaba, "Joint diversity and redundancy for resilient service chain provisioning," *IEEE J. Sel. Areas Commun.*, vol. 38, no. 7, pp. 1490–1504, Jul. 2020.
[35] A. Newell, D. Obenshain, T. Tantillo, C. Nita-Rotaru, and Y. Amir, "Increasing network resiliency by optimally assigning diverse variants to routing nodes," *IEEE Trans. Dependable Secure Comput.*, vol. 12, no. 6, pp. 602–614, Nov. 2015.
[36] W. Abbas, A. Laszka, and X. Koutsoukos, "Diversity and trust to increase structural robustness in networks," in *Proc. Amer. Control Conf. (ACC)*, Jul. 2019, pp. 4043–4048.
[37] J. Ai, H. Chen, and G. Cheng, "A correlation-aware diverse variant placement to increase network resilience," in *Proc. IEEE 4th Int. Conf. Comput. Commun. (ICCC)*, Dec. 2018, pp. 306–310.
[38] J. Caballero, T. Kampouris, D. Song, and J. Wang, "Would diversity really increase the robustness of the routing infrastructure against software defects?" in *Proc. NDSS*, 2008, pp. 1–5.
[39] Y. Zhu and X. Huang, "Node robust algorithm study based on graph theory," in *Proc. 8th Int. Conf. Fuzzy Syst. Knowl. Discovery (FSKD)*, Jul. 2011, pp. 2300–2303.
[40] Y. Prieto, J. E. Pezoa, N. Boettcher, and S. K. Sobarzo, "Increasing network reliability to correlated failures through optimal multiculture design," in *Proc. Conf. Electr., Electron. Eng., Inf. Commun. Technol. (CHILECON)*, Oct. 2017, pp. 1–6.
[41] Y. Prieto, N. Boettcher, S. E. Restrepo, and J. E. Pezoa, "Optimal multiculture network design for maximizing resilience in the face of multiple correlated failures," *Appl. Sci.*, vol. 9, no. 11, p. 2256, May 2019.
[42] Y. Prieto, C. Vega, J. E. Pezoa, and J. Crichigno, "Shared-risk-aware design for survivable migration in SDN environments," in *Proc. 16th IEEE Annu. Consum. Commun. Netw. Conf. (CCNC)*, Jan. 2019, pp. 1–6.
[43] O. Temizkan, S. Park, and C. Saydam, "Software diversity for improved network security: Optimal distribution of software-based shared vulnerabilities," *Inf. Syst. Res.*, vol. 28, no. 4, pp. 828–849, Dec. 2017.
[44] A.-L. Barabasi and M. Posfai, *Networking Science*. Cambridge, U.K.: Cambridge Univ. Press, 2016.
[45] O. Goldschmidt, P. Jaillet, and R. Lasota, "On reliability of graphs with node failures," *Networks*, vol. 24, no. 4, pp. 251–259, Jul. 1994.
[46] P. Y. Chen, G. Kataria, and R. Krishnan, "Correlated failures, diversification, and information security risk management," *MIS Quart.*, vol. 35, no. 2, pp. 397–422, 2011.
[47] V. M. Bier and M. N. Azaiez, *Game Theoretic Risk Analysis Security Threats*, vol. 128. New York, NY, USA: Springer, 2008.
[48] D. Rios Insua, A. Couce-Vieira, J. A. Rubio, W. Pieters, K. Labunets, and D. G. Rasines, "An adversarial risk analysis framework for cybersecurity," *Risk Anal.*, vol. 41, no. 1, pp. 16–36, Jan. 2021.
[49] D. L. Banks, J. M. R. Aliaga, and D. R. Insua, *Adversarial Risk Analysis*. Boca Raton, FL, USA: CRC Press, 2015.
[50] G. S. Parnell, C. M. Smith, and F. I. Moxley, "Intelligent adversary risk analysis: A bioterrorism risk management model," *Risk Anal.*, vol. 30, no. 1, pp. 32–48, Jan. 2010.
[51] M. E. J. Newman, *Networks: An Introduction*. London, U.K.: Oxford Univ. Press, 2010.
[52] H.-W. Lee, E. Modiano, and K. Lee, "Diverse routing in networks with probabilistic failures," *IEEE/ACM Trans. Netw.*, vol. 18, no. 6, pp. 1895–1907, Dec. 2010.
[53] S. Rai and D. P. Agrawal, *Distributed Computing Network Reliability*. IEEE Comput. Soc., 1990.
[54] S. Knight, H. X. Nguyen, N. Falkner, R. Bowden, and M. Roughan, "The Internet topology zoo," *IEEE J. Sel. Areas Commun.*, vol. 29, no. 9, pp. 1765–1775, Oct. 2011.
[55] W. Kosinski and D. Mikolajewski, "Genetic algorithms for network optimization," in *Proc. Int. Conf. Comput. Aspects Social Netw.*, 2009, pp. 171–176.
[56] K. Deep, K. P. Singh, M. L. Kansal, and C. Mohan, "A real coded genetic algorithm for solving integer and mixed integer optimization problems," *Appl. Math. Comput.*, vol. 212, no. 2, pp. 505–518, Jun. 2009.
[57] T. Yokota, M. Gen, and Y.-X. Li, "Genetic algorithm for non-linear mixed integer programming problems and its applications," *Comput. Ind. Eng.*, vol. 30, no. 4, pp. 905–917, Sep. 1996.
[58] E. J. Anderson and M. C. Ferris, "Genetic algorithms for combinatorial optimization: The assemble line balancing problem," *ORSA J. Comput.*, vol. 6, no. 2, pp. 161–173, May 1994.
[59] A. Norouzi and A. H. Zaim, "Genetic algorithm application in optimization of wireless sensor networks," *Sci. World J.*, vol. 2014, Feb. 2014, Art. no. 286575.

YASMANY PRIETO received the B.S. degree in electronics and telecommunications engineering from Universidad de Pinar del Río, Cuba, in 2012, and the Ph.D. degree in electrical engineering from Universidad de Concepción, Concepción, Chile, in 2019. He is currently a Postdoctoral Researcher and a part-time Lecturer with Universidad de Concepción. His research interests include network optimization, pattern recognition, and signal processing.

MIGUEL FIGUEROA (Member, IEEE) received the bachelor's degree in electronics engineering and the M.Sc. degree in electrical engineering from Universidad de Concepción, Chile, in 1990 and 1997, respectively, and the M.Sc. and Ph.D. degrees in computer science and engineering from the University of Washington, Seattle, WA, USA, in 1999 and 2005, respectively. He is currently a Professor of electrical engineering with Universidad de Concepción. His current research interests include hardware accelerators for scientific computing and high-performance embedded systems, VLSI circuits for low-power video processing and computer vision, and high-performance instrumentation for quantum cryptography and radioastronomy.

JORGE E. PEZOA (Member, IEEE) received the B.S. degree in electronics engineering and the M.S. degree in electrical engineering from Universidad de Concepción, Concepción, Chile, in 1999 and 2003, respectively, and the Ph.D. degree in electrical engineering from The University of New Mexico, Albuquerque, NM, USA, in 2010. He is currently an Associate Professor with the Departamento de Ingeniería Eléctrica, Universidad de Concepción. His research interests include distributed computing, pattern recognition, statistical signal processing, network optimization, and hyperspectral image and signal processing for industrial processes. He is a member of the Society of Photo-optical Instrumentation Engineers (SPIE), the Optical Society of America (OSA), and the Association for Computing Machinery (ACM).