Shared-risk-aware Design for Survivable Migration in SDN Environments

Conference Paper · January 2019
DOI: 10.1109/CCNC.2019.8651765

CITATIONS
0

READS
53

4 authors:

Yasmany Prieto
University of Concepción
9 PUBLICATIONS  3 CITATIONS

Christian Fernando Vega
Institución Universitaria CESMAG
2 PUBLICATIONS  0 CITATIONS

Jorge E. Pezoa
University of Concepción
73 PUBLICATIONS  600 CITATIONS

Jorge Crichigno
University of South Carolina
67 PUBLICATIONS  354 CITATIONS

Some of the authors of this publication are also working on these related projects:

Building a Cybersecurity Pipeline through Experiential Virtual Labs and Workforce Alliances View project

An Enhancement to the Constant Range Method for Nonuniformity Correction of Infrared Image Sequences View project

All content following this page was uploaded by Jorge E. Pezoa on 11 March 2019.

The user has requested enhancement of the downloaded file.
Abstract—In this paper, we propose a heterogeneous risk-aware Software-Defined Networking (SDN) migration method for designing survivable networks in the face of multiple correlated failures. The migration method, which is implemented in one shot, specifies how many nodes of each SDN implementation are needed, and where such nodes must be located, in order to yield an SDN migrated network with maximal survivability, when multiple correlated failures impact the entire network connectivity. We formulated the survivable SDN migration problem through integer optimization, where the proposed cost function assesses the survivability of the migrated network in terms of the number of connected components after a failure. The numerical results calculated over test networks show the capability of migration method to provide survivable SDN topologies, which trade-off the heterogeneity in the SDN implementations and the number of shared risks.

Keywords—survivability, SDN migration, shared group risks, heterogeneity.

I. INTRODUCTION

In this decade, SDN has emerged as a new and flexible technology for managing telecommunications networks. SDN is a new paradigm to programmable networks that allows for their dynamical configuration and management. SDN separates the control plane from the data plane and shifts the control plane to a centralized controller, thereby enabling network operators to implement and rapidly prototype network policies such as routing and Quality-of-Service (QoS) [1], [2].

The migration from Traditional Networks (TNs) to SDNs has been analyzed technically and economically, and the significant conclusions stated in [3], [4] are that the migration to SDNs is already occurring and both SDNs and TNs should coexist for decades. Today, SDNs offer to network operators an excellent opportunity for either totally or incrementally migrate from TNs to SDN environments. We can name, for example, the SDN-based Packet Transport Network operated by China Mobile, [5], and the Software Defined Optical Network designed by China Telecom, [5], as successful migration cases. Besides, we can also include the case of Huawei’s successful deployment of one of the first SDN hybrid deployments [6].

The flexibility, compatibility, and low cost administration of SDN also offers the possibility of migrating a network using multivendor technologies [7]. This is an interesting and appealing feature, since TNs are highly homogeneous because they are designed using commercial nodes that are supplied by a reduced number of vendors or, in most of the cases, they are supplied by a single vendor. In general terms, we can claim that there has been a tendency in network design to specify almost all the routers and switches from the same vendor, using a few different node models. This homogeneity in the nodes of TNs is not desirable because it introduces a large correlation, or shared risk, among the nodes. Examples of such risks are exploits (0-day vulnerabilities), hardware parts that can cause a network component to fail [8] and a huge list of denial-of-service vulnerabilities reported by Common Vulnerabilities and Exposures [9]. The effect of exploiting shared node risks on a network is the impairment of large portions of a net [10].

Since the current trend in networking seems to be migrating TNs to either SDNs or hybrid SDNs, we claim that such tendency can be exploited to design shared-risk–aware networks. Furthermore, inspired by the property that the heterogeneity in biological systems is a valuable commodity for survivability, [11], we claim that migrating TNs to SDNs environments by means of a network design that specifies multivendor nodes, and considers their shared node risks, should improve the survivability of the migrated network in the presence of
In this paper, we propose a heterogeneous risk-aware SDN migration method for designing survivable networks in the face of multiple correlated failures. We define survivability as the capability of the network to maintain a certain degree of connectivity, in the presence of attacks/failures. The migration method is implemented as a one-shot strategy that specifies how many nodes of each SDN implementation are needed, and where such nodes must be located, in order to yield an SDN network with maximal survivability, when multiple correlated failures impact the entire network connectivity. We formulated the survivable SDN migration problem through integer optimization, where the cost function considers the number of connected components arising after a correlated failure. As such, we may think of the cost function as a survivability metric for the entire network. We propose also to solve the integer optimization problem using Genetic Algorithms (GAs). The numerical results calculated over test networks show the capability of migration method to provide survivable SDN topologies, which trade-off the heterogeneity in the SDN implementations and the number of shared risks. We note that the major contribution of this paper is to tackle the SDN migration problem from a very different perspective. To the best of our knowledge, this is the first work focusing on SDN node migration for improving the survivability of the network, in the presence of correlated node failures. Moreover, the approach we follow in our work focuses more on the survivability of the entire network, rather than improving the reliability of network paths.

The rest of this paper is organized as follows. In Section II the related work on the area is summarized. In Section III we present the migration model and pose the integer optimization problem for migrating a network topology to a survivable SDN network, in the presence of correlated attacks. In this section we also introduce a heterogeneity metric for the SDN migrated networks. Next, in Section IV we present the GA technique used to solve the optimization problem and the numerical results of our work. Finally, in Section V we draw the conclusions of the paper and our future work.

II. RELATED WORK

In the literature, there are several works related to the migration from TNs to SDN environments. Some articles present solutions to the migration of services such as containers [12], Virtual Machines (VMs) [13], and architectures to support SDN [14]. In the survey by Amin et. al [15], they propose a taxonomy of migration based on network deployment strategies, SDN controllers, network management techniques, Traffic Engineering (TE) mechanisms and testing/verification and security mechanisms for hybrid SDN networks. Interestingly, the idea of migrating nodes to SDN has also been exploited to create green networks, which correspond to Virtual Networks (VNs), configured using SDN routers, that accommodate traffic demands simultaneously reducing energy consumption [16].

The line of work on network deployment strategies is of interest to this paper. From [15], [17], [18], we can observe that the study of node migration in network topologies has focused on how to carry out the migration in time. Caria et. al proposed a two-stage algorithm that optimizes the migration of nodes to SDN, regarding TE over primary and alternative paths [17]. In [19], Levin et. al presented the Panopticon architecture for migrating traditional and SDN switches. In addition, most of the studies available in the literature focus the migration work on conducting TE. For instance, Das et. al proposed a formal migration mechanism to replace nodes, in a single domain, over time while maximizing the TE gains [20]. Wang et. al introduced a migration model that efficiently trades off between migration costs and the load balance variation [21]. More recently, Tanha et. al proposed an algorithm that defines an upgrade policy for SDNs considering budget constraints [22]. Remarkably, most of the works have in common that the migration procedure is formalized through optimization problems, such as mixed integer programming or linear programming, which happen to be NP-hard.

In this work, however, we have taken a different path. First, we propose a one-shot migration process where all the nodes are replaced. Since this approach may appear as an expensive process, we have considered small network sizes to where few nodes must be migrated. Second, the goal of the SDN migration process is to improve the network survivability, in the face of correlated failures, instead of focusing on TE issues. Third, the approach presented here focuses on the survivability of the entire network, rather than on improving network path reliability. To the best of our knowledge, this is the first work on SDN node migration tackling the problem of improving network survivability in such a way.

III. SYSTEM MODEL AND PROBLEM FORMULATION

A. Definitions

The topology of a network is modeled here by the undirected graph $G = (V, L)$, where $V = \{1, 2, \ldots, n\}$ is the set of nodes and $L = \{(i, j) : \text{nodes } i \text{ and } j \text{ are connected}\}$ is the set of bidirectional links between the nodes. Let us assume that, for the SDN migration process we have at our disposal several vendors providing network nodes with SDN implementations. We denote here by $K = \{1, 2, \ldots, N\}$ the set of all the different SDN implementations available for migration. Also, for the failure model, we assume that each implementation considered here is prone to be attacked. More precisely, each SDN implementation may be exploited in a correlated manner, and the consequence of such correlated attacks is that all the SDN migrated nodes will simultaneously fail. Moreover, such failures induce failures in all their associated links.

Next, we suppose that for each SDN implementation, there exist a list of well-known risks and a list of communication protocols. Let $R$ denote the total number of risks that are known for the SDN implementations. Thus, for each SDN implementation, say, the $i$th implementation with $i \in K$, we associate the binary, $R$-dimensional row vector $x_i$. This vector is termed as the risk vector associated to the $i$th SDN implementation, say, the $i$th implementation.
implementation and is defined as $x_i = (X_{i1}, X_{i2}, \ldots, X_{iR})$, where $X_{ij} = 1$ specifies that the migrated node is affected by the $j$th risk and $X_{ij} = 0$ indicates otherwise. By stacking the $K$ risk vectors we introduce the $K$-by-$R$ migration risk matrix $X$. Also, let us denote by $K' = \{1, 2, \ldots, k\} \subseteq K$ the set of all the SDN implementations selected for the migration.

For simplicity, we assume here that the entire migrated network is orchestrated by a single SDN controller. Also, we suppose that all the implementations in $K'$ are able to communicate with each other by one or more communication protocols. Finally, we assume also that the SDN controller is not a network node and is also completely reliable; consequently, we exclude it from the analysis.

B. Rationale

To design a survivable network in the presence of correlated node failures, we must preserve network connectivity even though a part of the network has failed. From our failure model, when a risk in an SDN implementation is exploited, not only the attacked nodes fail but also their associated links. This later issue affects the connectivity of all the migrated nodes that remain working because two or more connected components may arise [23], [24]. Consequently, we claim that, to migrate from TNs to survivable SDNs, when a correlated failure exploits a shared risk in two or more SDN implementations, we must minimize the number of connected components arising after the failure.

To clarify our rationale please refer to the simple network topology in Fig. 1(a). Consider that $K$ SDN implementations are at hand, and are affected by $R = 4$ shared risks as specified by the risk matrix:

$$
X = \begin{pmatrix}
1 & 1 & 0 & 0 \\
1 & 0 & 1 & 0 \\
1 & 0 & 0 & 1 \\
0 & 1 & 1 & 0 \\
0 & 1 & 0 & 1 \\
\end{pmatrix}.
$$

(1)

For this configuration, we analyzed several SDN migration alternatives and the best node migration specification is given in Fig. 1(b), where only three out of the five implementations were selected. Namely, the selected SDN implementations are “Implementation 3” (green nodes), “Implementation 4” (purple nodes), and “Implementation 5” (red node). Note that SDN Implementation 3 and 4 do not share common risks between them and two nodes were selected from each implementation. Besides, Implementation 5 was selected for one node because it only shares a single risk with each Implementation 3 and 4. From Fig. 1(c), we observe that post-failure, the working nodes in the migrated SDN network remain connected.

The following key observations, which form the basis of our survival SDN migration strategy, were obtained analyzing the above mentioned results. First, the SDN implementations selected for migration must exhibit a reduced number of shared risks, to limit their impact on the migrated topology. Second, the number of SDN nodes selected for migration should not be leaned to one or few SDN implementations.

C. Survival SDN Migration Problem Statement

Consider a graph $G = (V, L)$ representing the topology of a network with $n$ nodes, and $K$ different SDN migration implementations affected by $R$ shared risks as specified in the risk matrix $X$. We tackle the problem of specifying how many nodes of each implementation are needed, and where such nodes must be located, to yield a network with maximal survivability, when multiple correlated failures impact the entire network connectivity. We mathematically formulate the survivable SDN migration problem as:

$$
(C^2)^* = \min_{T(V) \in M} \left( \frac{\sum_{r=1}^{R} C(G_r) - 1}{R} \max_{r \in R} (C(G_r) - 1) \right),
$$

(2)

subject to: $\sum_{k=1}^{K} \sum_{j=1}^{n} 1_{\{T(j)\}}(k) = n.$

(3)

where $T(V) : V \rightarrow K$ is a mapping from $V$ to $K$ such that $T(j) = k$ assigns the $j$th node to the $k$th SDN implementation, $M$ is the search space of all possible mappings for assigning the $K$ SDN implementations to the $n$ nodes, $G_r = (V_r, E_r) \subset (V, E)$ is the migrated topology resulting after a failure affecting all the SDN implementations exhibiting the $r$th risk, $C(G_r)$ is the number of connected components in $G_r$, and $1_{\{T(j)\}}(k) = 1$ is the indicator function of $i$th node executes the $k$th SDN implementation. Lastly, we note that (3)
imposes the constraint that a node cannot execute more than one SDN implementation.

Now, we make the following remarks. The term $C(G_r) - 1$ is an attempt to favor the emergence of a single connected component after a correlated failure occurs. The term $\frac{1}{R} \sum_{r=1}^{R} C(G_r) - 1$ assesses the average number of connected components arising after a failure. Minimizing the term $\max(C(G_r) - 1)$ lessens the worst case in the number of connected components appearing after any shared risk is exploited. Consequently, $C = \sqrt{C^2}$ is a metric of survivability for a migrated SDN network: the closer to 1 the more survivable network. Lastly, The optimal migration allocation is given by the mapping $f(V)$ achieving $(C^2)^*$.  

D. Heterogeneity Metric for Migrated Networks

To assess the heterogeneity of the migrated networks we introduce here the SDN-implementation set heterogeneity metric $h$. This metric computes the cumulative dissimilarity between each pair of SDN implementations using the formula:

$$h = 1 - \frac{1}{R \binom{K}{2}} \sum_{i=1}^{K} \sum_{j=i+1}^{K} x_i x_j^T.$$  

(4)

Note that the metric is normalized by the number of risks, $R$, and the number of combinations for the SDN implementations $\binom{K}{2}$. Note also that the metric takes values in $[0, 1]$, where the value “0” (respectively, “1”) means that all the SDN implementations share every risk (respectively, do not share risks) among the implementations.

IV. RESULTS

A. Solving the Survivable SDN Migration Problem via GAs

We propose to solve the survivable SDN migration problem through the GA technique [25]. For the GA, we coded the chromosome in an integer-valued string of length $n$. The $j$th position in the chromosome denotes the $j$th node in the network. Each position in the chromosome contains a non-negative integer value, say $k$, which specifies to which SDN implementation belongs its corresponding node in the migrated network. For the crossover operator, we employ the single point crossover that was executed with a probability of 0.8. For the mutation operator, a position in the chromosome is randomly selected and its value is replaced by another SDN implementation, with a probability of 0.01. For the selection process, we use the fitness proportional selection, implemented by a roulette wheel and taking into consideration to address a minimization problem by the inverse function. The stopping criteria for the GA technique considers two options: (i) the number of iterations reaches a maximum, predefined number; and (ii) the absolute difference between the mean values of the cost function, in two consecutive generations, for the entire population is smaller than some predefined tolerance $\epsilon$. The solution to the problem, obtained in the final generation, is the chromosome from the population that achieves the minimum value for the objective function.

B. Numerical Results

The networks extracted from Topology Zoo, [26], and depicted in Fig. 2 were used in this work to show the capability of the proposed survivable SDN migration design. In our calculations, we have used the following three risk matrices: a 5-by-5 $X_1 = I$,

$$X_2 = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 1 & 1 \\ 1 & 0 & 1 & 0 \\ 0 & 1 & 0 & 0 \end{pmatrix}, \text{ and } X_3 = \begin{pmatrix} 1 & 1 & 0 & 0 \\ 0 & 1 & 1 & 0 \\ 1 & 0 & 1 & 0 \\ 0 & 1 & 0 & 1 \end{pmatrix}.$$  

The values in the above mentioned matrices were arbitrarily assigned to assess the network survivability in the presence of correlated failures triggered by shared risks. The risk matrix $X_1$ does not exhibit shared risks among the SDN implementations; consequently, the heterogeneity metric $h = 1$. For the risk matrix $X_2$, up to three implementation can be selected without exhibiting shared risks, and heterogeneity index is $h = 0.875$. The risk matrix $X_3$ shows a configuration with two or more shared risks, and its heterogeneity index is $h = 0.8$.

As a first result, we show in Figs. 3 and 4 the SDN migrated topologies resulting after using our survivable migration approach, for the three risk matrices considered here. Fig. 3 shows the topologies for the ATT network and Fig. 4 shows the topologies for the Arpanet. Note that all the five SDN implementations have been used in the migrated designs. Also note that as the heterogeneity index $h$ decreases, the SDN migrated implementations reflects more node clustering. Further, note that for the ATT network, given its node connectivity, when
When $h = 1$ the SDN migrated topology does not require a lot of clustering to avoid the arising of a large number of connected components. However, when $h = 0.8$ it can be observed that migrated nodes with the same implementation tend to be connected. Unlike ATT network, the lowest connectivity of the Arpanet network demands more SDN node clustering for the three migrated topologies.

In Fig. 5 we show the relationship, for each network, between the optimum value of the cost function, achieved via GAs, and the heterogeneity index. It can be observed that, as the $h$ index increases the value of the cost function decreases. This means that as the heterogeneity in the SDN implementations augments, the estimate of the average number of connected components decreases, upon the occurrence of a correlated failure. In other words, as the heterogeneity in the SDN implementations increases, the damage inflicted on the network topology by shared risks decreases on the average.

We now compare the estimate of the average number of connected components of different networks, for the same value of the $h$ index. We observe that the higher the number of nodes, the higher the estimate of the average number of connected components. This behavior can be explained in a simple manner considering a fully connected network and a completely heterogeneous set of SDN implementations ($h =$...
1. Under this scenario, after a failure triggered by any risk, the number of connected components will be given by $\frac{n}{h} + 1$. Thus, we expect that for any value of the heterogeneity index, the estimate of the average number of connected components in the migrated network should be dictated by the ratio $\frac{n}{h}$.

To establish a fair comparison among the different networks, we include in the analysis the number of nodes and the node average degree of each network. We divide $C^*$ by $n$ to obtain values in the range $(0, 1)$. Besides, we weight the heterogeneity index by the node average degree to introduce the effect of connectivity in the analysis. Figures 6(a) and (b) show the normalized estimate of the average number of connected components as a function of the degree-weighted heterogeneity index. On the one hand, Fig. 6(a) compares, for each SDN migrated network the effect of the risk matrix. It can be observed that, for each migrated network, increasing the $h$ index, that is introducing heterogeneity, decreases the normalized estimate of the average number of connected components, meaning that the migrated network shows a survivable behavior. On the other hand, Fig. 6(b) compares the migrated networks for each risk matrix. It can be observed that now it is the combined effect of higher heterogeneity and higher average node degree that reduces the average number of connected components, thereby yielding a more survivable SDN migrated network.

V. CONCLUSION

We have proposed a one-shot method for migrating from TNS to SDNs exhibiting both heterogeneous SDN implementations and survivability to correlated failures triggered by shared risks. The proposed migration method maps SDN implementations onto nodes and such mapping aims to create topologies with a small number of connected components after a failure occurs. The most noticeable result of the mapping is that SDN nodes cluster according to their shared risks, thereby improving the survivability of the SDN migrated network post-failure. Besides, the cost function proposed as the basis for the migration scheme has been thought of as a survivability metric because it aims to estimate the number of connected components arising after a failure. We also introduced here a metric for the heterogeneity in the SDN implementations that quantifies the existence of shared risks within the available implementations to carry out the migration. In our numerical calculations, we migrated four TNS using three risk matrices with different heterogeneity index. We observed that, for each network, when the heterogeneity in the risk matrix increases, the estimate of the number of connected components appearing after a failure decreases. When we normalized the estimate of the number of connected components arising after a failure by the number of nodes in the network, we observed that the combined effect of higher heterogeneity and higher average node degree reduces the estimated average number of connected components. Thus, both the heterogeneity in the set of SDN implementations and the network connectivity can be related to an increase in the network survivability when facing correlated node failures.

As a future work, we will focus on designing multiple SDN migration methods to cope with the economical burden of a one-shot migration. Besides, we will study other cost functions for creating non-singleton connected components of large size.

ACKNOWLEDGMENT

This work was supported by CONICYT: FONDECYT Regular 2016 Folio 1160559, PCHA/Doctorado Nacional Folio 2015-21150313 and PCHA/Doctorado Nacional Folio 2018-21180418.

REFERENCES