Cyber Regulatory Networks: Towards a Bio-inspired Auto-resilient Framework for Cyber-Defense

Usman Rauf[1(✉)], Mujahid Mohsin[2], and Wojciech Mazurczyk[3]

[1] Department of Software and Information Systems, University of North Carolina at Charlotte, Charlotte, NC, USA
urauf@uncc.edu, usman.cyberdna@gmail.com
[2] National University of Sciences and Technology, Islamabad, Pakistan
mmohsin@cae.nust.edu.pk
[3] Institute of Telecommunications, Warsaw University of Technology, Warsaw, Poland
wmazurczyk@tele.pw.edu.pl

Abstract. After decades of deploying cyber-security systems, it has become a well-known fact that the existing cyber-security architecture has numerous inherent limitations that make the maintenance of the current network security devices unscalable and provide the adversary with asymmetric advantages. These limitations include: (1) difficulty in obtaining the global network picture due to lack of mutual interactions among heterogeneous network devices, (2) poor device self-awareness in current architectures, (3) error-prone and time consuming manual configuration which is not effective in real-time attack mitigation, (4) inability to diagnose misconfiguration and conflict resolution due to multi-party management of security infrastructure. In this paper, as an initial step to deal with these issues, we present a novel bio-inspired auto-resilient *security* architecture. The main contribution of this paper includes: (1) investigation of laws governing the dynamics of correct feedback control in Biological Regulatory Networks (BRNs), (2) studying their applicability for synthesizing correct models for bio-inspired communication networks, i.e. Firewall Regulatory Networks (FRNs), (3) verification of the formal models of real network scenarios, to prove the correctness of the proposed approach through model checking techniques.

1 Introduction

With the ever increasing number of data breaches and security incidents it is evident that the traditional manual models of cyber-security are unable to defend complex and large cyber-networks. The new models of defence need to focus on auto-resiliency, integration and fast response-time. To meet these objectives, even after decades of development of cyber security systems, still there exist

A. Compagnoni et al. (Eds.): BICT 2019, LNICST 289, pp. 156–174, 2019.
https://doi.org/10.1007/978-3-030-24202-2_12

inherent limitations in the current cyber-security architecture that allow adversaries to not only plan and launch attacks effectively but also learn and evade detection quite easily. These limitations include: (1) difficulty to obtain the global knowledge (about security policies) of all security devices in the network, e.g., different firewalls are often managed by independent administrators, who have no incentive for conducting coordinated exercises to mitigate the global risk, (2) manual reconfiguration, which is time consuming, error-prone and ineffective in real-time attack mitigation, (3) lack of mutual interactions among network devices as most security devices such as firewalls and IDSs are configured and managed individually for a subset of assets being directly protected, without any realization of the global impact caused by reconfiguration of a single device on the entire enterprize, (4) multi-party management makes the diagnosis (for misconfiguration) and conflict resolution difficult and often leads to un-optimized policy. Current cyber-security architecture does not have any notion of coordination via interaction among security devices. The limited sensing mechanisms that exist work mostly offline, require human assistance and do not converge towards global optimal solution. Our proposed work is inspired by interaction at the cellular level between the entities of a Biological Regulatory Network (BRN), which is a very fundamental and crucial phenomenon in the dynamics of biological systems [25].

Existing cyber-security architectures also have no self-awareness of the risk as the security policies usually take into account only usability, reachability and demand requirements; whereas, the risk realization is either partially or completely ignored. Although the state of the art security risk assessment frameworks provide a general overview of assessing and mitigating threats in different phases of a kill-chain [1,28], these frameworks do not provide any means for aligning local goals with global objectives. Interaction and self-awareness are the fundamental ingredients for self-organization and adaptivity. Note that sensing and self-awareness are two completely different processes, as sensing is related to observing one's environment or neighbors through interaction, and self-awareness is the realization of one's internal state. The sensing process without the aim of optimizing a global objective (via feedback notion and automation) is nearly useless, as by the time humans take action, the damage could be already done, or becomes uncontrollable. The hierarchy in current cyber architectures lacks these functionalities. Therefore, there is an immense need to integrate feedback mechanism in current architectures to allow continuous and dynamic risk mitigation and real time response (in case of any perturbation).

On the other hand, biological systems have built-in feedback mechanisms, through which they adapt and survive unknown threats in the surrounding environment. We intend to redesign cyber-security architecture, based upon such regulatory, and feedback control mechanisms, in which even if one router, device or machine is compromised, the neighboring devices should have tendency to alter their behavior by allowing or restricting it from performing malicious activities. The resultant architecture should have the tendency to survive under abnormal conditions and to reduce (global) risk factor by maintaining progressive cycle to avoid a deadlock/malicious state where risk is higher/above a specific threshold.

1.1 Challenges

The incentive of every security device in a cyber infrastructure is to reduce the risk and to increase the usability and demand of the assets, for which it is responsible. In large scale networks many security devices are intertwined, and security policies of any device are not designed to reinforce neighboring security devices, rather they are more focused towards the interests (usability and demand) of the important assets that they are protecting. Consequently, a single erroneous action performed by an operator (to fulfill demand) at a local level in any security device, might have catastrophic (infrastructure level) impact (by increasing risk on the other devices), which is hard (or impossible) to comprehend via manual configuration. To the best of our knowledge, no existing technique implements correct feedback control mechanism, for reconciliation among security devices, to automate the global risk mitigation in an infrastructure. Designing optimal policies to mitigate risk at global level is a Distributed Constraint Optimization Problem (DCOP) [20], for which time complexity is exponential in the worst case scenario, and managing such sensitive system manually is nearly impossible.

1.2 Contributions

As a first step towards creating an auto-resilient cyber architecture, in this paper we present a three fold contribution: (1) first, we investigate the laws governing the dynamics of correct feedback control in BRNs, (2) then we apply these laws to synthesize correct model for bio-inspired networks, (3) finally we verify the synthesized models for real communication networks, through model checking techniques, to prove the correctness of the proposed approach.

2 Related Work

Over the past few years, bio-inspired computing has evolved as an active area of research. Different aspects of biological phenomenon give rise to bio-inspired mechanisms with applications to cyber-security, which include: (1) Swarm Intelligence (SI), (2) Artificial Immune system (AIS), (3) Genetic Mutation, and (4) Gene and Cell Regulation.

SI can be defined as an emergent collective behavior of non intelligent interacting entities that attempt to achieve self-organization and global objectives without any centralized control. Inspired by the behavior of social insects, the main focus of the domain is to design resilient and robust systems, which can efficiently and intelligently operate under the threat and catastrophic conditions without any centralized control [3]. Mostly, SI based approaches have been used for efficient routing, for identifying the source of an attack in the network, i.e. Intrusion Detection System (IDS), and to prevent the attack by localizing its origin, i.e. Intrusion Prevention System (IPS). Few recent approaches towards this direction are [12,14,19,21,27].

The research in the domain of AIS began in the mid-1980s with Farmer, Packard, and Perelson's study [11]. The biggest revolution in this domain was

the utilization of the concept of human immune system for computer security which proposed one to one mapping or analogy between the immune system and IDSes. With the development of the HIS principle, Negative Selection Algorithm (NSA) [13], Clonal Selection Algorithm (CSA) [5], Immune Network Algorithm (INA) [4], and Danger Theory Algorithm (DTA) [2] become the most representative algorithms in this domain. The most recent approaches which utilize these concepts to design or improve IDSes includes [17,18,30].

Recently, researchers from the domain of cyber security have mapped the concept of genetic mutation to cyber infrastructure with the aim of improving resiliency against active cyber threats e.g. Denial of Service (DoS) attacks. They propose to change different parameters of the network (e.g. Routes or IP addresses) proactively to avoid links under congestion and to avoid spreading of malware [10,16,23]. This domain is relatively new (and less explored) as compared to other bio-inspired cyber-security domains. Therefore, decentralized and more efficient algorithm are required to fill the gap.

The fourth and the most important area which has not been explored by the researchers so far, to its fullest potential, is the study of natural phenomenon of Cell and Gene regulation via signal transduction [22]. *Signal Transduction* is a mechanism in which (observed) exterior signals from a cell are transmitted into its interior against which numerous autonomous entities, i.e., genes regulate each other to generate an appropriate response and maintain homeostasis (by maintaining the optimal values of different parameters, e.g., blood pressure, body temperature, and sugar level). The first and the only practical contribution in this domain was proposed by Dressler [9]. The author proposed a refined and practical model for self-organization in a network facility (i.e., load management during packet inspection in intrusion detection systems) based on the concept of cell regulation [9]. The motivation behind the choice was to embed self-organization in a distributed detection system to regulate the amount of traffic rate between probes and detection system in variable situations, and in order to save the detection unit from becoming a potential target.

In the modern cyber networks, the usability of applications or services running on the end hosts are very important for an enterprize and hence cannot be ignored. Although the approach proposed by Falko Dressler, presents an auto-regulatory architecture for distributed IDS, the model does not incorporate the notion of usability associated with an end host and risk affiliated with a flow. The reactive strategy of the mentioned approach completely relies on the maximum throughput that a detection unit can handle without any regard to the usability of end host, reachability requirement, and potential risk imposed by a certain flow.

3 Biological Regulatory Networks (BRNs)

Every functionality in the human body and evolution of the morphological features is highly influenced or controlled at molecular level [6]. Genes and proteins are the main ingredient of this controlling mechanism, which cooperate together

in a programmed manner to perform multiple tasks in an organism. Genes are the informative subunits of the DNA and they decode instructions in the form of proteins. Some proteins have the function of regulating the expression of genes by turning them on or off. This process of interaction, between genes and protein regulatory elements, establishes a BRN.

BRNs are very unique in their functionality as they have tendency to operate in adverse conditions under extreme threats without any central control or external monitoring. The main strength and capability of exceptional operability come from the structure of interactions, which is a feedback control mechanism. It is only through feedback loops in a BRN that imposes a controlled mechanism in order to maintain an optimal concentration of proteins in a cell [29]. Such feedback loops give rise to the phenomenon of genetic oscillations, which play a main role in the activity of maintaining the cascade of internal biochemical reactions with the extracellular environment. Molecular alterations in the performance of such behavioral rhythms can lead to the severe pathological diseases, e.g., cancer. This biological phenomenon can be summarized in a simple way: the dynamics of the living system is controlled by the BRNs, and at any given time a BRN of a living organism should optimize the cell behavior by maintaining the concentrations of proteins to make it survive in its (often abnormal) environmental conditions.

The behavior of a single entity in a BRN can be classified into the sequence of three different functions which are repeated infinitely often [25]: Sensing (signals from the neighboring entities), Actuating (changing internal state) and Signaling/influencing (firing/triggering an output signal) neighboring entities. A biological entity (gene/cell) in a BRN can either influence its neighboring entities positively or negatively. The process of positively (conversely negatively) influencing others is referred as *Activation* (conversely *Inhibition*). The influence phase of this natural process (activating/inhibiting neighboring entities), forms a feedback loop which is very fundamental to the control mechanism in BRNs for maintaining the optimal value of different parameters. As a first step towards creating bio-inspired resilient architecture, we intend to understand how this feedback notion works and can help us accomplishing our objectives. In the next section, we present a real-life example of a BRN, which is responsible for respiratory mechanism in the human body and demonstrate how its malfunctioning can lead to a severe lung disease.

3.1 BRN of the Cystic Fibrosis (Pseudomonas Aeruginosa)

Cystic fibrosis is a life threatening genetic disease that primarily effects the lungs and digestive systems [24]. The main cause of the respiratory deficiency in patients of cystic fibrosis is mucus production. The regulatory network which controls the mechanism of mucus production is shown in Fig. 1. $AlgU$ (x) is the main regulator of mucus production and it favors its own production while another gene inhibits it. The regulatory network of mucus production can be analyzed using different approaches (e.g. Linear Hybrid Automata, coupled Differential Equation, Regulatory Network Transition Systems, or Regulatory Graphs)

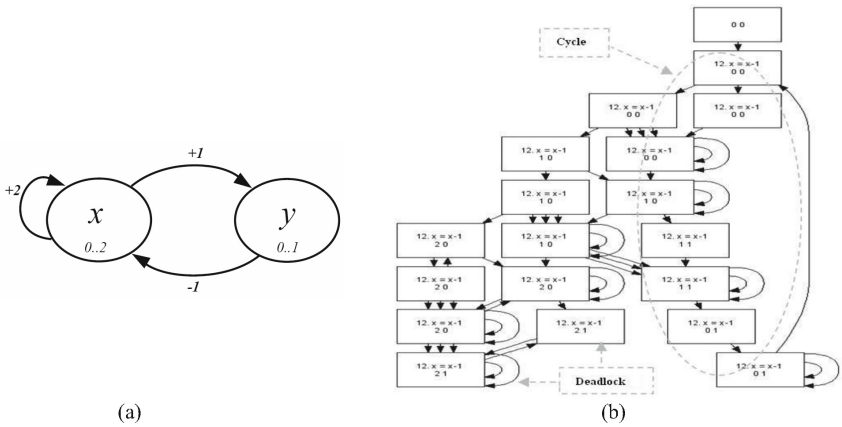

Fig. 1. (a) BRN of Pseudomonas aeruginosa, (b) Corresponding state space

[24], but for the convenience and simplicity we represent it by a regulatory graph in Fig. 1(a), where x represents gene $AlgU$ (or its protein) and y represents the inhibitor protein of $AlgU$. The concentration of x and y is defined over the qualitative levels. A positive sign (+) represents that x is the promotor of y or positively influences y when its (x's) concentration reaches level-1. As a result of positive influence (activation), y's concentration starts increasing. Once y's concentration reaches level 1, it inhibits x; as a result the concentration of x reduces towards the minimum (through a feedback mechanism). The regulatory interaction "+2" means that x becomes its own promoter/activator once its concentration reaches level 2. The concentration levels of each biological entity are represented qualitatively. Level '0' corresponds to the situation when the concentration for the protein of a certain gene/biological entity is absent. In the same manner, the higher levels, i.e., '1' or '2' refer to certain amount of concentration. By carefully analyzing the regulatory graph of the Cystic Fibrosis, Rauf et al. [25] show that it can govern two types of behaviors, oscillatory and deadlock. Oscillatory is considered as normal whereas the deadlock condition which is a hold and wait event sequence can be referred to as a malicious behavior. If the inhibitory entity (y) activates before a certain level (before which $AlgU$ favors its own production) then the system will not lead towards the malicious disease and will remain in progressive cycles, by maintaining the optimal values of the concentrations. Figure 1(b) shows the two possible behaviors of the (Pseudomonas aeruginosa) BRN, achieved by the concurrent model checking. Although there are two possible behaviors, the chances for malfunctioning of this BRN are very low as the probability of someone being infected by this disease is ≈0.00001. This is only because of the nature imposed feedback control mechanism that regulatory graph always avoids malicious behavior by remaining in progressive cycles/oscillations. To avoid deadlock, disease state or malicious state, there must be a realization of risk and there should be a feedback mechanism through

Table 1. Analogy between BRNs and cyber networks

Characteristics of biological entity	Characteristics of cyber entity
Sensing	
An entity receives signals from all the neighboring entities as a result it becomes aware of the current state of its neighboring entities	A security device can collect required information from its neighboring devices i.e. value of assets, desired reachability requirements, and risk evaluation against different policy rules
Actuation	
Biological entity increases/decreases its concentration under excitatory/inhibitory interaction from its neighboring entities	A security device can (autonomously) add/remove set of rules under the influence of certain activities/signals from its neighboring entities, which will result in increase/decrease of the attack coverage of a firewall/security device
Reaction	
Influencing neighboring entities through excitation or inhibition. However, nature of influence in biological systems is always static leading to disease states	A security device can also react against its actuation/updated (local) goals to influence/effect its neighboring entities, after evaluation of threat impact, risk payoff, or benefits associated with the assets, which it is responsible for the security and performance of the system

which y can be aware of the threats and can inhibit x. This shows that the understanding of theories and principles behind this self-organization through feedback mechanism is crucial.

4 BRN-Inspired Cyber-Security Architecture Mapping

To proceed further with the idea of integrating auto-resiliency characteristics of the biological systems in current cyber architecture, there must exist some analogy and mapping of actions among fundamental entities of both domains. Table 1 shows our effort towards mapping of BRNs dynamics to cyber networks.

4.1 Architecture of the Bio-inspired Firewall Regulatory Networks (FRNs)

We propose the idea of an architecture, through which security devices can regulate each other via inhibitory/excitatory orders to optimize the global objective. Figure 2 gives the realization of the proposed architecture, in which each security device has its own sensors and actuators for self-awareness and interaction with the neighboring elements. The most important part is the decision engine, which receives information from the sensors and decides which actions to take through actuators (according to global interest).

Nature of the Interaction can be of two types in terms of cyber networks. **Inhibition** order (from one device to another), which is driven by the risk associated to the assets for which a device is responsible and **Activation/Excitation** order, which is driven by pay-off associated with the reachability. In working principle of BRNs, nature of an interaction of a certain biological entity (gene) with its neighboring entities always remain static and does not change over time. The static (nature of) interaction leads towards disease states and malicious functionality of organs. We aim to avoid rather than eliminate the causes which may lead a system to bad/disease states.

Therefore, we propose that at any given time elements (security devices) of the network should synthesize a set of regulatory interactions (among them) which always leads towards progressive cycles (auto-regulation of the system) rather than deadlock/malicious states (where the risk is always high). In the next section we classify feedback mechanisms, which is fundamental to synthesize regulatory interactions at a given time; through which system can always remain in progressive cycles (self-organize or reconfigure itself if there is any perturbation in the external environment).

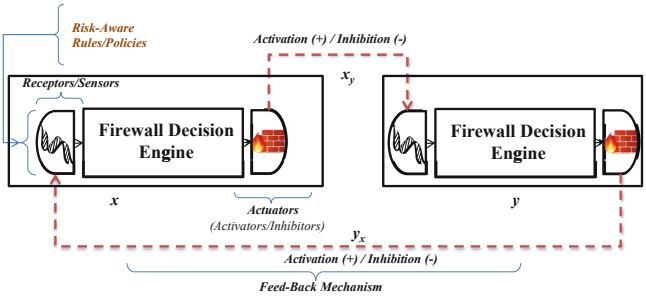

Fig. 2. Bio-inspired architecture of firewall regulatory networks for self organization/automated risk mitigation

4.2 Notion of the Feedback and Current Cyber Infrastructures

René Thomas and d'Ari [29] mathematically proved using Linear Stability Analysis that: "*A feedback loop is positive if it contains even number of negative interactions (Fig. 3(a)) and negative if it contains odd number of negative interactions (Fig. 3(b))*". When a system only contains negative feedback (conversely positive), which means it only has odd number of negatively regulated interactions, it tends to oscillate around a certain optimal value. The oscillatory behavior can also be referred to as a cyclic behavior, and such mechanism in biological sciences is called "homeostasis". If synthesis/evolution of a certain parameter starts, it triggers the evolution of the same parameter in the following linked elements (through positive regulations), unless a negatively regulated entity is activated, from that point onward (the negatively regulated entity), suppresses

the evolution of the same parameter in the following entities. This causes suppression in the synthesis of that entity by pushing it to the ground state where it is unable to regulate the neighboring entities. Hence the decay effect reaches to the original entity/element. This happens periodically and corresponding behavior is referred to as oscillatory behavior. Conversely, the systems having positive feedback loops tends to end up in (unique or multi) stable states/deadlocks. Stable states are those states in which the value of a certain parameter for all entities becomes stable and it remains the same, which can also be considered as a deadlock state, from where no progress becomes possible.

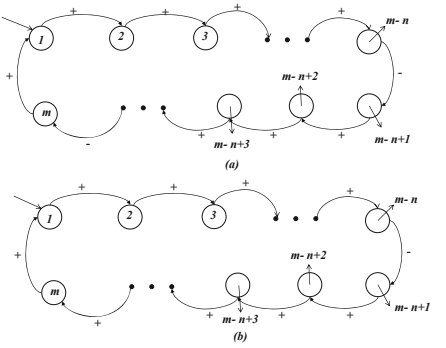

Fig. 3. Classification of feedback control mechanisms; (a) positive feedback loop, (b) negative feedback loop; where m and n represent entity indexing

In the context of current risk aware cyber infrastructures, deadlocks or stable-steady states can be viewed as states where the overall risk for an organization is above a certain bearable threshold. In the next section we describe our proposed framework in detail for synthesizing the correct feedback control models of Firewall Regulatory Networks (FRNs).

5 The Proposed Framework

This section describes our proposed BRN-inspired security architecture. Since, there are a variety of security devices for securing cyber networks, here we consider one particular type of security device, i.e., the network firewall (FW). A typical corporate network can use several firewalls to segregate the network according to the organizational needs, e.g., external, internal and demilitarized zone (DMZ). We propose a revolutionary new paradigm whereby all these FWs interact and form a network that we call FRNs. This model can be extended to any other network security devices in the future, such as, IRNs for intrusion detection regulatory networks where the host and the network IDS sensors interact.

5.1 FRN Synthesizer

Figure 4 sketches the details of our proposed framework, which uses FRN topology, connectivity requirements, and asset demands/responses as an input. The first step towards achieving a correct feedback control mechanism (for self-organization) is to synthesis the set of interactions at a certain point in time among all security devices (to resolve conflicting issues), so that bad/malicious states can be avoided. Describing the formal notion of cyber demand/response (between security firewalls) and the correct control logic are the fundamental ingredients of our proposed framework. For this purpose in the forthcoming section, we formalize cyber demand/response (regulatory interactions) features and control logic as Constraint Satisfaction Problem (CSP) [8,26].

As a next step in FRN synthesizing phase, we pass the formal model to the SMT solver z3 [7] to synthesize the FRN with the correct feedback control. Finally, we formally model the synthesized instance of the FRN, in concurrent model checking tool SPIN using PROcess MEta LAnguage (PROMELA) [15] to verify whether the synthesized instance achieves self-regulation or not. The details about each component of the framework are discussed in the forthcoming subsections.

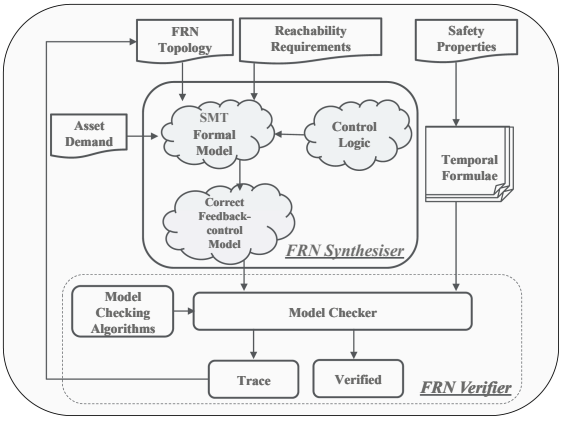

Fig. 4. Proposed framework

Formalization of the Request/Response and Reachability Requirements. The following set of constraints represents the nature of interactions (among two entities in a network) in the first order logic.

$$Inf_{i,j}^{(+,des)} : A_{i,des}^{req} \bigwedge (reach_{j,des} \bigvee j == des)$$

$$Inf_{i,j}^{(-,des)} : A_{i,des}^{resp} \bigwedge (reach_{j,des} \bigvee j == des)$$

$$\forall_{i,j} \left(Inf_{i,j}^{(+,des)} \bigvee Inf_{i,j}^{(-,des)} \right) \mapsto \mathcal{E};\ \text{where } i, j,\ \&\ des \in F$$

Where $Inf_{i,j}$ represents the nature of the influence entity i has on entity j, as a result of accessability request ($A^{req}_{i,des}$ between i and des) or as a result of response ($A^{resp}_{i,des}$), and F is a set of security devices in a regulatory network. The nature of influence can be + or −, depending upon if it is a request (req) or a response ($resp$). Note that we make a basic assumption here, i.e., positive influence (+) is derived by an accessibility demand ($A^{req}_{i,des}$) from a source (i) to a destination (des) and negative sign (−) is influenced by the risk affiliated with response ($A^{resp}_{i,des}$) of accessability demand. Therefore, influence of an entity (i) on a neighboring entity (j) against an accessibility demand/response can be calculated by evaluation of ($Inf^{(+,des)}_{i,j} \bigvee Inf^{(-,des)}_{i,j}$). In any case, all the edges across the network ($e \subseteq \mathcal{E}$) should be assigned a value against any request/response between i and des. These constraints must be evaluated against every accessability request from a source to a destination (des) over the whole FRN, to figure out the nature of the influence among security devices.

Formalization of the Feedback Control Logic. In the following constraints, we formalize the correct feed-back control logic, which must be consistent throughout the regulatory network.

$$func : \mathbb{N} \longmapsto \mathbb{B}$$

$$func(e_i) : \left(\sum_{i:1}^{n} e_i \right) \% 2$$

$$func(e^l_i) : \bigwedge_{l=1}^{m} \left(\left(\sum_{i:1}^{n} e^l_i \right) \% 2 \right)$$

$$\forall_l \exists_i \left(func(e_i) \neq 0 \right) \longmapsto \mathbb{B}$$

We describe regulatory interaction as a mathematical function ($func$) over the set of natural numbers, which evaluates to true/false. Where e_i is the set of interactions (edges) between entities/security devices in a closed feedback loop, and there can be n such interactions (in a closed feedback loop), i.e., e_i: {e_1, e_2,..., e_n}. We represent *+ve* (conversely −ve) interaction as a number *0* (conversely *1*) due to its positive parity. Therefore, the mathematical function representing the regulatory interactions (in a closed feedback loop) according to the formalization is the sum of all interactions in a closed feedback loop with modulo *2*. Finally there can be m multiple feedback loops in a system or a security device/component may be involved in multiple feedback mechanisms, thus, to achieve global objective we quantify over all closed feedback loops to find a satisfiable instance of the model. If there exists a satisfiable instance, we get an answer as true along with the configuration of the instance.

The above mentioned formalizations together provide us with the satisfiable instance of the system/model, which has an odd number of -ve interactions. As mathematically proved by René Thomas and Richard d'Ari, that an even

number of -ve interactions leads system to a deadlock or malicious state, our formalization tends to avoid the instances of the system/model which may lead the system towards malicious behavior or which may not have tendency to optimally regulate the value of risk (globally) in a network [29].

5.2 FRN Verifier

To prove the correctness of the proposed formalism, we synthesize FRNs model based on the real-life examples, and verify it using model checking to determine whether the synthesized models are able to reconfigure themselves under any external perturbation. If the control logic is correctly integrated the corresponding models should be able to recover from the high risk states. As measuring quantitative risk is not the focus of this research, we affiliate qualitative risk levels to each entity in the FRN (as a firewall is represented as a state machine, which can evolve over these levels which in reality represents the risk imposed by the active policy in a device, and can be calculated using any risk assessment metric). We allow risk levels to evolve over the qualitative values, to observe if the resultant state space of the FRN contains any deadlock or not. At the end, we update our topology, as user accessability demands change over time, which might lead the previously synthesized model to a deadlock. Therefore, whenever additional accessability demands arrive, new model must be synthesized. For the verification purposes, we present an abstract formal representation of a firewall/security device, regulatory interactions, and the parameters which govern the state of a security device.

In our abstraction, the state of a firewall is defined over qualitative levels. This means that a firewall is an entity which can assume any qualitative value from a given set. The firewalls can have an impact on each other and such impact is modeled as regulatory interactions. As utility of any active subset of firewall rules can be calculated and thresholds on regulatory interactions can also be assigned, therefore, our abstraction is practical and aligned with the real-life practice. In the following sections we discuss the modeling elements of the ***Verifier*** one by one in details.

Discrete Model of a Firewall. In this section we present the formal model of an entity in a regulatory network. A regulating entity (e.g. firewall) is defined as an automaton. It receives an input from interacting neighbors, changes its internal state in response to it, and produces an appropriate output, depending on threshold level (θ_{ij}), where i, j are the interacting entities.

Formally, a set of firewalls F can be expressed as a set of interacting automata and a firewall may assume any positive value in a range.

$$F = \{f_1, f_2.., f_m\};$$
$$f_k = \{0, \ldots, n_k\};\ where\, k \in \{1, \ldots, m\}$$

There can be m firewalls in a network and any firewall f_k can have any possible discrete qualitative levels. The possible states for the network $\mathcal{F}$ can then be

defined as the cartesian product:

$$\mathcal{F} = f_1 \times f_2 \times f_3 \times \ldots \times f_m$$

Regulatory Interactions Modeling. An excitatory (resp. inhibitory) interaction ($f_1 \xrightarrow{+} f_2$) (resp. $f_1 \xrightarrow{-} f_2$) is active when usability demand to access certain area is equal to or above a specific threshold level θ. Conversely, inhibitory interaction ($f_1 \xrightarrow{-} f_2$) is active or triggered when the risk imposed on a certain firewall is equal or above a certain threshold. We also associate a threshold (θ_{12}) to each interaction from ($f_1 \xrightarrow{\theta_{12}} f_2$).Where $\theta_{12} \in \{1, ..n\}$. f_1 is called the activator of f_2, if $f_1 \geq \theta_{12}$ (resp. $f_1 < \theta_{12}$) for the excitatory interaction (respectively inhibitory interaction).

Modeling of Parameters. At any time instant the state of a firewall depends only on its set of attractors. Attractors are the other entities (firewalls) which can directly influence a firewall via inhibitory or excitatory interactions. We represent the set of attractors of an entity as $w(f_i^\alpha)$, where $\alpha \subset \{1, ..., m\}$. The residual effect of $w(f_i^\alpha)$ on the evolution of f_i^α can be given by the logical parameter.

$$\mathcal{K}(w(f_i^\alpha)) \in \{0, \ldots, n_\alpha\}$$

The logical parameter corresponds to the level towards which a firewall evolves:

1. if $f_i^\alpha < \mathcal{K}(w(f_i^\alpha))$ then f_i^α is increasing
2. if $f_i^\alpha > \mathcal{K}(w(f_i^\alpha))$ then f_i^α is decreasing
3. if $f_i^\alpha = \mathcal{K}(w(f_i^\alpha))$ then f_i^α is stable

The above mentioned behavior of evolution of a firewall over qualitative levels, as a response of interactions with neighboring entities is modeled as *Resource_allocation* process in SPIN model checker.

The Complete Model of the FRN. The main components of a FRN are:

- m security devices (e.g. firewalls) are modeled as m processes f_i where $i = \{1, 2, \ldots, m\}$ having their own identical thresholds for either usability or risk.
- Process *Resource_ allocation* which changes resources of f_i as its internal states are changed as a result of interactions, keeping in view its attractors $\mathcal{K}(w(f_i^\alpha))$.
- Process *Observer* which ensures at every step of the computation that f_i remains within the bound $\{0, ..., n\}$.

After modeling of all components of the system, namely f_i, *Resource_allocation*, and *Observer*, we make parallel composition of all the components and allow the system to evolve.

$$FRN: \; f_i \;||\; Resource\_allocation \;||\; Observer$$

6 Evaluation

For the synthesis of the regulatory interaction between security devices for control mechanism (FRN), we use Z3 SMT solver [7]. For verifying the correctness of the generated instances of FRN, we use PROMELA as modeling formalism and model checking tool SPIN [15]. All the experiments were conducted on Core i7 machine with 3.4 GHz processor, and 16 GB memory.

6.1 Case Studies

Regulation of Risk Between Two Security Devices. We consider a real-life scenario in which few assets in local area network are protected by a screening firewall and a specialized firewall. Figure 5(a) gives the description of the case study. Customer service department in the Demilitarized Zone (DMZ) is protected by an immediate screening firewall (Fw_1), whereas the more important (A_2, A_4) and critical (A_1, and A_3) assets are protected by a specialized firewall (Fw_2). Due to the desire for attracting customers/visitors, web servers have influence on screening firewall to increase its "allow" space. Lets assume that to facilitate the customers, Fw_1 (managing the customer service department) sends a service accessability request (Skype) for all assets behind Fw_2 (managing the engineering department). Since only Fw_2 has a complete visibility of the assets under its control, it is aware of a known Skype's elevation of privileges vulnerability (CVE-2017-11786), which resides at A_2 (a non-critical asset). Fw_2 evaluates the impact of the request and realizes that it imposes a threat (with CVSS v3.0 base score of 88%) to its neighboring critical assets (A_1, A_3) as well. As a consequence, it must inhibit Fw_1, if the risk imposed is above the pre-specified threshold. In Fig. 5, X represents Fw_1 and Y represents Fw_2.

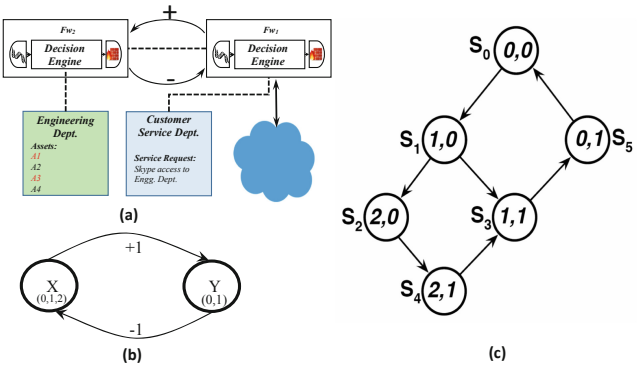

Fig. 5. (a) Case study: two firewalls securing assets, (b) Synthesized FRN model, (c) Corresponding State Space

Now to resolve this issue, we need reconciliation, and before that we need a control mechanism, representing: *"who can influence whom and in what sense"*. Using our formalism to synthesize the regulatory interaction, we obtain the model in Fig. 5(b). We associate qualitative numbers as threshold for verification. The resultant model states: when the demand of customer services X increases and reaches up to a qualitative level 1, it imposes a threat on the firewall Y. As a result of this situation the risk on Y starts increasing. When the risk of Y reaches an unbearable threshold, it transmits an inhibition order to X. The inhibition order informs X to reduce its allow space (or to only activate the set of flows/policies for which the residual risk is below the qualitative level 1), until the threat imposed on Y is reduced to the minimum. To prove correctness of the proposed approach we model the derived instance of the system in SPIN model checker. As both firewalls are concurrently evolving or dynamically changing entities which influence each other, therefore, PROMELA is the best suited formalism to analyze the behavior of such concurrent model.

We verify correctness of our postulate about automated risk mitigation using correct feedback control mechanism via self-regulation property written in Linear Temporal Logic (LTL):

$$(x = 0, y = 0) \implies \mathbf{X}(\neg(x = 0, y = 0)) \wedge \mathbf{G}(\mathbf{F}(x = 0, y = 0))$$

The property states that: *if the system starts from a normal state where* $x = 0$ & $y = 0$, *and the next state is not* $x = 0$ & $y = 0$, *then along the path of evolution, in future the system eventually goes back to the ground or normal state* $x = 0$ & $y = 0$.

Figure 5(c) illustrates the existence of self-regulation in the state space of the model with two regulatory firewalls, it also shows how system regulates itself or recovers once it reaches malicious/bad state ($x = 2$ & $y = 1$), where risk is maximum or above a normal value. The state space generated in the example is from the SPIN model checker.

Regulation of the Risk Between Three Security Devices. We present another case study which contains three firewalls in a feedback mechanism. Figure 6 gives the description of the case study. In this case we consider a bad state to be a situation where (qualitative levels of) residual risk always remains maximum ($X = 2$ & $Y = 1$ & $Z = 2$).

To verify if our proposed method works with the extended case study we again model our system using PROMELA [15]. The state space of the system (Fig. 7) shows that the system never gets stuck in a malicious state. Whenever it encounters a situation where the risk is above bearable threshold, it regulates (recovers) itself and remains in a progressive cycle. Careful analysis shows that there are multiple ways to avoid the high-risk state. The best possibility is when firewall Z inhibits firewall X before it (X) reaches a configuration where its risk becomes maximum, as a result the system switches from (1,1,1) to (0,1,1). In the worst case scenario, the system reaches the state (2,1,2), due to the delayed

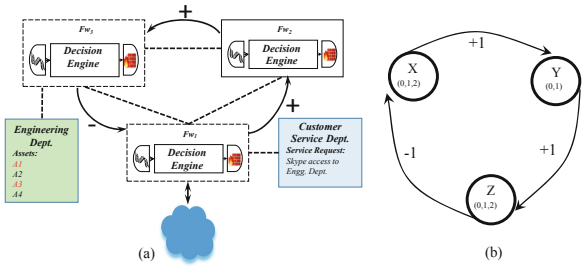

Fig. 6. (a) Case study: three firewalls securing assets, (b) Synthesized FRN model

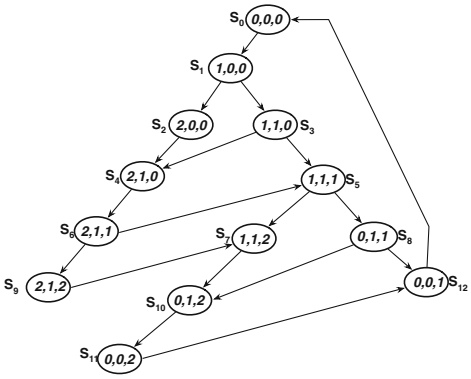

Fig. 7. State space of the FRN containing three regulatory firewalls

propagation of inhibition order from Z to X. Although it reaches malicious state, it recovers from it and goes back to the normal states after series of alterations in its configuration ((2,1,2)→(0,1,2)→(0,0,2) →(0,0,1)→(0,0,0)).

Overhead of the SMT Formalization and Formal Verification. We also perform overhead analysis of the SMT formalization by selecting different number of entities in a feedback loop (to synthesize regulatory interaction). Figure 8 illustrates that our formalization is capable of synthesizing a set of regulatory interactions for a large number of devices in milliseconds. As we increase the number of entities the required time increases linearly which makes this approach feasible and practical for the real-life implementation.

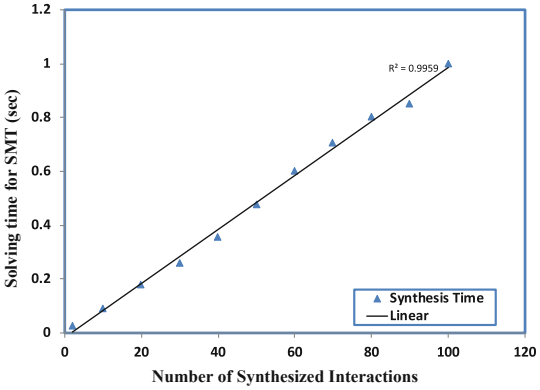

Fig. 8. Overhead analysis of synthesizer

7 Conclusion and Future Work

In this paper, we present a bio-inspired resilient architecture for cyber-security. We propose a novel framework for automated risk mitigation by embedding self-organizing features in the current infrastructure. By formalizing feedback control mechanism as a constraint satisfaction problem in SMT we allow for an automatic synthesis of cyber-regulatory networks which can reconfigure themselves, thereby, eliminating the need of manual reconfiguration by the administrators. To prove the correctness our proposed approach, we formally model two real-life scenarios and analyze their self-organizing behavior using model checking tool. The results show that the proposed architecture allows cyber network to be self-organizing, and dynamically adaptable to variable conditions. The low overhead for synthesizing correct feedback control makes our approach more practical for real-life implementation. In the future we plan to deploy this architecture on the medium scale networks and determine its resiliency against different scenarios. We also aim to explore cooperative game theory, to embed notion of conflict resolution in this architecture to deal with conflicting scenarios, in which a device might not be interested to cooperate with the other devices.

References

1. International Standards Organization ISO/IEC 27005: 2008. Information technology-security techniques-information security risk management. International Standards Organization, Geneva, Switzerland (2008)
2. Aickelin, U., Bentley, P.J., Cayzer, S., Kim, J., McLeod, J.: Danger theory: the link between AIS and IDS. CoRR, abs/0803.1997 (2008)
3. Bonabeau, E., Dorigo, M., Theraulaz, G.: Swarm Intelligence: From Natural to Artificial Systems. Oxford University Press Inc., New York (1999)
4. de Castro, L.N.: Artificial Immune Systems: A New Computational Intelligence Approach. Springer, London (2002)

5. de Castro, L.N., Von Zuben, F.J.: The clonal selection algorithm with engineering applications. In: GECCO - Workshop Proceedings, pp. 36–37. Morgan Kaufman (2002)
6. Davidson, E.H., Erwin, D.H.: Gene regulatory networks and the evolution of animal body plans. Science **311**(5762), 796–800 (2006)
7. de Moura, L., Bjørner, N.: Z3: an efficient SMT solver. In: Ramakrishnan, C.R., Rehof, J. (eds.) TACAS 2008. LNCS, vol. 4963, pp. 337–340. Springer, Heidelberg (2008). https://doi.org/10.1007/978-3-540-78800-3_24
8. Dechter, R.: Constraint Processing. Morgan Kaufmann Publishers Inc., San Francisco (2003)
9. Dressler, F.: Self-organized network security facilities based on bio-inspired promoters and inhibitors. In: Dressler, F., Carreras, I. (eds.) Advances in Biologically Inspired Information Systems. Studies in Computational Intelligence, pp. 81–98. Springer, Heidelberg (2007). https://doi.org/10.1007/978-3-540-72693-7_5
10. Duan, Q., Al-Shaer, E., Jafarian, H.: Efficient random route mutation considering flow and network constraints. In: 2013 IEEE Conference on Communications and Network Security (CNS), pp. 260–268, October 2013
11. Farmer, J.D., Packard, N.H., Perelson, A.S.: The immune system, adaptation, and machine learning. Physica D **22**, 187–204 (1986). Proceedings of the Fifth Annual International Conference
12. Fink, G.A., Haack, J.N., McKinnon, A.D., Fulp, E.W.: Defense on the move: ant-based cyber defense. IEEE Secur. Priv. **12**(2), 36–43 (2014)
13. Forrest, S., Perelson, A.S., Allen, L., Cherukuri, R.: Self-nonself discrimination in a computer. In: Proceedings of 1994 IEEE Computer Society Symposium on Research in Security and Privacy, pp. 202–212, May 1994
14. Haack, J.N., Fink, G.A., Maiden, W.M., McKinnon, A.D., Templeton, S.J., Fulp, E.W.: Ant-based cyber security. In: 2011 Eighth International Conference on Information Technology: New Generations (ITNG), pp. 918–926, April 2011
15. Holzmann, G.J.: The SPIN Model Checker: Primer and Reference Manual. Addison-Wesley Professional, Boston (2003)
16. Jafarian, J.H., Al-Shaer, E., Duan, Q.: Openflow random host mutation: transparent moving target defense using software defined networking. In: Proceedings of the First Workshop on Hot Topics in Software Defined Networks, HotSDN 2012, pp. 127–132. ACM (2012)
17. Jinquan, Z., Xiaojie, L., Tao, L., Caiming, L., Lingxi, P., Feixian, S.: A self-adaptive negative selection algorithm used for anomaly detection. Prog. Nat. Sci. **19**(2), 261–266 (2009)
18. Li, G.Y., Guo, T.: Receptor editing-inspired negative selection algorithm. In: 2010 International Conference on Machine Learning and Cybernetics (ICMLC), vol. 6, pp. 3117–3122, July 2010
19. Liu, Z., Kwiatkowska, M., Constantinou, C.: A swarm intelligence routing algorithm for manets. In Proceedings of the 3rd IASTED International Conference on Communications, Internet and Information Technology (CIIT 2004), p. 1. ACTA Press (2004)
20. Modi, P.J., Shen, W.M., Tambe, M., Yokoo, M.: Adopt: asynchronous distributed constraint optimization with quality guarantees. Artif. Intell. **161**(1), 149–180 (2005)
21. Muraleedharan, R., Osadciw, L.A.: An intrusion detection framework for sensor networks using honeypot and swarm intelligence. In: 6th Annual International Mobile and Ubiquitous Systems: Networking Services, MobiQuitous 2009, pp. 1–2, July 2009

22. Rauf, U.: A taxonomy of bio-inspired cyber security approaches: existing techniques and future directions. Arab. J. Sci. Eng. **43**, 6693–6708 (2018)
23. Rauf, U., Gillani, F., Al-Shaer, E., Halappanavar, M., Chatterjee, S., Oehmen, C.: Formal approach for resilient reachability based on end-system route agility. In: Proceedings of the 2016 ACM Workshop on Moving Target Defense (MTD), pp. 117–127 (2016)
24. Rauf, U., Sameen, S., Cerone, A.: Formal analysis of oscillatory behaviors in biological regulatory networks: an alternative approach. Electron. Notes Theoret. Comput. Sci. **299**, 85–100 (2013)
25. Rauf, U., Siddique, U., Ahmad, J., Niazi, U.: Formal modeling and analysis of biological regulatory networks using spin. In: 2011 IEEE International Conference on Bioinformatics and Biomedicine (BIBM), pp. 304–308, November 2011
26. Rossi, F., van Beek, P., Walsh, T.: Handbook of Constraint Programming (Foundations of Artificial Intelligence). Elsevier Science Inc., New York (2006)
27. Sellami, K., Chelouah, R., Sellami, L., Ahmed Nacer, M.: Intrusion detection based on swarm intelligence using mobile agent. In: International Conference on Swarm Intelligence, June 2011
28. NIST SP800-30. Risk Management Guide for Information Technology Systems. National Institute of Standards and Technology, USA (2002)
29. Thomas, L.C., d'Ari, R.: Biological Feedback. CRC Press, Boca Raton (1990)
30. Zeng, J., Liu, X., Li, T., Li, G., Li, H., Zeng, J.: A novel intrusion detection approach learned from the change of antibody concentration in biological immune response. Appl. Intell. **35**(1), 41–62 (2011)