Future Generation Computer Systems 127 (2022) 297–308

Contents lists available at ScienceDirect

Future Generation Computer Systems

journal homepage: www.elsevier.com/locate/fgcs

AI-assisted Computer Network Operations testbed for Nature-Inspired Cyber Security based adaptive defense simulation and analysis

Shishir Kumar Shandilya [a,*], Saket Upadhyay [a], Ajit Kumar [b], Atulya K. Nagar [c]

[a] Cyber Security and Digital Forensics Division, Vellore Institute of Technology, VIT Bhopal University, Bhopal, India
[b] Department of Computer Science and Engineering, Soongsil University, Seoul, South Korea
[c] School of Mathematics, Computer Science and Engineering, Liverpool Hope University, United Kingdom

ARTICLE INFO

Article history:
Received 10 January 2021
Received in revised form 16 August 2021
Accepted 3 September 2021
Available online 21 September 2021

Keywords:
Nature-Inspired Cyber Security
Computer Network Operations
Cyber range
Adaptive cyber defense
Network simulation
Performance tuning

ABSTRACT

In the current ever-changing cybersecurity scenario, active cyber defense strategies are imperative. In this work, we present a standard testbed to measure the efficacy and efficiency of customized networks while analyzing various parameters during the active attack. The presented testbed can be used for analyzing the network behavior in presence of various types of attacks and can help in fine-tuning the proposed algorithm under observation. The proposed testbed will allow users to design, implement, and evaluate the active cyber defense mechanisms with good library support of nature-inspired and AI-based techniques. Network loads, number of clusters, types of home networks, and number of nodes in each cluster and network can be customized. While using the presented testbed and incorporating active-defense strategies on existing network architectures, users can also design and propose new network architectures for effective and safe operation. In this paper, we propose a unified and standard testbed for cyber defense strategy simulation and bench-marking, which would allow the users to investigate current approaches and compare them with others, while ultimately aiding in the selection of the best approach for a given network security situation. We have compared the network performance in difference scenarios namely, normal, under attack and under attack in presence of NICS-based adaptive defense mechanism and achieved stable experimental results. The experimental results clearly show that the proposed testbed is able to simulate the network conditions effectively with minimum efforts in network configuration. The simulation results of defense mechanisms verified on the proposed testbed got the improvement on almost 80 percent while increasing the turnaround time to 1–2 percent. The applicability of proposed testbed in modern technologies like Fog Computing and Edge Computing is also discussed in this paper.

1. Introduction

Nature-Inspired algorithms represent a set of all methodologies and approaches which key idea is derived from nature like animal behaviors (cukcoo search algorithm, ant colony optimization), and other natural behaviors (genetic algorithm, particle swarm optimization) Nature-Inspired Cyber Security (NICS) techniques are gaining the attention of researchers, students, and practitioners due to their capability of providing robust and intelligent defense [1–4]. There are many network simulation environments already available to deploy, test, and observe how the network responds [5,6]. However, due to the rapid technological advancements in Cloud services, Artificial Intelligence (AI), and 5G/6G networks, cyber threats and attacks are presenting new challenges in automation of network response in the laboratory [7,8]. Cyber Ranges, a controlled and interactive cyber adversary simulation environment where professionals can test and learn how the deployed strategies might react to an actual threat in real world scenario. and other virtual environments fail to mimic the critical infrastructures quickly and takes a lot of time to get configured properly. Apart from that, as the attacks are also becoming more sophisticated day-by-day, these environments require more time to formalize and simulate the appropriate attack and usually suffer from the lack of scalability, flexibility, and interoperability. Thus, a cohesive system is required which can model the infrastructure accurately, model the attack scenario, and gives the flexibility to observe the network (defense) response in detail. In the case of the nature-inspired defense system, achieving the fore-mentioned points are more difficult as such kind of defense systems are not reactive. So, they proactively detect and mitigates the threats and attacks which may cause a sudden decline in throughput, status misinformation, or excessive bandwidth usage. Certainly, the adaptive defensive techniques, defense strategies and technologies that

* Corresponding author.
E-mail address: shishir.sam@gmail.com (S.K. Shandilya).

https://doi.org/10.1016/j.future.2021.09.018
0167-739X/

adapt to the changing threat surface and attack vectors. Such systems take real time information from the network assets and dynamically, require special arrangements in a cyber range or virtual environment, to monitor the network and explore the full potential of these defensive techniques [9–11]. In this paper, we have introduced an open-source bench-marking and analysis tool to provide a complete solution while specifically focusing on the requirements of Nature-inspired cybersecurity techniques. NICS techniques ranges from advanced multi-objective optimization to camouflaging of network architectures [12], and many more. The proposed testbed is also well-supported with the various pre-coded subroutines and libraries for implementing the AI and NICS techniques on few clicks, which facilitates the user to remain focus on the prime objective of research rather on fundamental issues of network setup and configuration. Primarily, the proposed testbed has two primary advantages,

1. It provides easy network setup based on selection (rather than design) of the number of nodes, clusters, protocols, and communications and
2. It provides a rich (ever-growing) library for implementing the AI and NICS techniques.

1.1. AI-assisted Computer Network Operations [AI-CNO]

Computer Network Operations (CNO) protects from advanced cyber-attacks and ever-evolving threats, while carefully examining the overall network on various parameters. Computer Network Operations typically consists of Computer Network Attack (CNA), Computer Network Defense (CND) and Computer Network Exploitation (CNE). Artificial Intelligence can enhance the procedures involved in CNA, CND and CNE as well [13]. In CNA, AI can be used for self-mutation, machine-on-machine attacks and automating Distributed Denial of Service (DDoS) attacks. In CND, AI can assist the defensive mechanism by effectively designing the cyber-deception and decoys. In CNE, AI can be used to exploit gathered in reconnaissance to generate intelligent decision making. Like in any other domain, AI treatment introduces the facility of prediction based on historical data, and quick decision making through learning in the domain of network simulations too [14,15]. Amalgamating AI with CNO capabilities leads to two major benefits: a. A more closer and effective observation on various parameters with more control. b. A better analysis of cause–effect to decide quality defensive methods. If it further combines with NICS, the resultant study could lead to a realistic, flexible and robust defensive method. But, to perform such experiments, an effective testbed is required to be developed along with rich libraries for both AI and NICS.

1.2. Nature-inspired Cyber Security [NICS]

Cyber attacks are becoming advanced and utilizing intelligent algorithms for the data breach. This situation is untenable and getting worse day-by-day. Therefore, the defensive mechanisms are expected to provide intelligent, and robust security against advanced cyber attacks [16,17]. Nature-inspired Cyber Security comes up with such solutions that not only provide intelligent and adaptive security but will also guarantee cyber immunity and resilience[1] in near future. NICS treatment to cybersecurity methods provides adaptability, self-organization, resilience, and robustness by default, along with the possibility of having offensive security up to a certain extent. However, such treatment also requires a high level of performance tuning and optimization, and therefore like any other nature-inspired systems, NICS also possesses many limitations like network latency and formulation of multi-objective functions [18,19].

NICS can provide many novel features like autonomous threat detection and resolution, artificial immune system (a computationally intelligent self learning algorithm modeled after mammalian immune system's characteristics of learning and memory for use in problem-solving), camouflaging of network architecture, and self-healing programs. NICS also fundamentally supports and having huge potential to achieve cyber immunity is a point where the cost of attacking is very high to make it achievable for attackers. Therefore, many current researchers and organizations like Kaspersky [20] are working extensively on this.

Further paper is organized as follows, Section 2 provides detailed background on the testbed and Section 3 discussed about the proposed testbed. Section 4 presents experimental details and results of various test-cases and in Section 5 the conclusion and future scope are discussed.

2. Background

Selecting a new/candidate defensive system for an organization is a critical decision, which requires a lot of observation, research, sensitivity analysis, and performance-tuning on various operational parameters. This cannot be done on existing operational networks due to the risk of losing data and integrity and also because their cause–response mapping is not discrete. Therefore, these defense systems are often analyzed on testbeds with an emphasis on making the overall observations realistic as much as possible. And this requires accuracy and full-control of network setup, configuration, and attack-and-defense simulation [6,21]. Test cases or scenarios generation is critical for testing security mechanisms; Li et al. [22] have proposed deep learning-based watermarking generation and testing the security of images.

Computer Network Operations (CNO) are not new to the domain of network simulation in the presence of advanced cyber attacks like Distributed Denial of Service (DDoS), Low-Rate Transmission Control Protocol Denial of Service (LRTCPDoS) [23], and Evil Handshake. In a recent work, the authors proposed a DDoS[2] detection mechanism using entropy variations between attack and regular traffic [24]. The experimental simulation was done using a mininet emulator with POX controller and open flow switches. Many researchers have already accomplished the state-of-art CNO practices over a variety of Cyber Ranges (CR) [25]. There are many network testbeds are available for specific purposes [25], however unfortunately none of them provides a comprehensive platform and 'one-fits-all-types' of requirements. Now, new computation infrastructure and technologies are being proposed with different architecture and need a custom made testbed. For example, Christos et al. [26] have presented a novel and secure cache decision system (CDS) that used IoT and 6G. This forces the researchers to develop new testbed as per their own need, which is a time-consuming, deviating, and costly process. However, this problem cannot be completely mitigated but can be controlled up to an extent. Also, in the case of Cyber Ranges, a term which comes from the military domain, the accurate simulation of realistic scenario is often a difficult task [27–30].

There are several solutions which performs well like University of Utah's Emulab [31], DETER [32], Virtualized CR [33], and Purdue University's Reassure.[3] These solutions are open-source and offer a high-level of scalability along with the flexibility to customize the testbed as per the requirement. However, in

[1] https://cyberstartupobservatory.com/wp-content/uploads/REGISTRATION/Cyber-Resilience_Definition_Related_Disciplines&Frameworks_1.pdf

[2] Distributed Denial of Service.

[3] https://www.cerias.purdue.edu/research/projects/home/detail/52/a_safe_virtual_imaging_instrument_for_logically_destructive_experiments_reassure

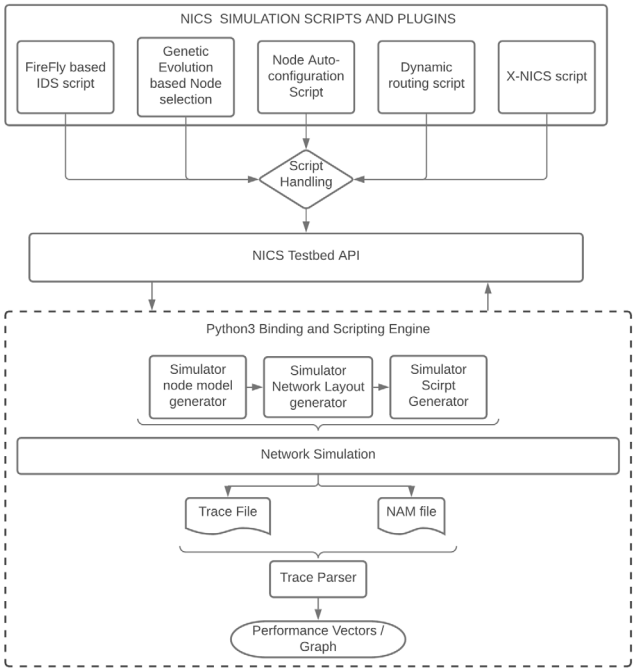


Fig. 1. Communication structure and module layout of the proposed testbed.

the case of NICS, the add-on focus is on optimizing the defense along with routine performance measurement and tuning. Therefore, an AI-assisted testbed specifically designed for experimenting and observing the adaptive defense mechanisms based on NICS is a major requirement for testing the next-generation cyber-resilience and cyber-immunity algorithms.

A comparison of features of proposed testbed with existing similar technologies is shown in Table 1

3. Proposed testbed for NICS-based defense

The block diagram of the proposed testbed is shown in Fig. 1 that highlight the various components and their interactions. From input to output i.e. operation flow of the proposed testbed is depicted in Fig. 2.

The user provides input to the constraints and parameters via the testbed interface or in the testbed script and submits it for simulation. Then parameters are passed to the scaling module that arranges the network structure based on *NpC* and *M* parameters. It will also integrate provided Attack Parameters and NICS/Adaptive defensive algorithms in the dynamic network and generate *tcl* script for NS2.

The generated script is passed for simulation and all registered key parameters are saved as 'post-simulation data', this data is used to generate a result with help of preset methods and AI-assisted data handling.

The generated results are then normalized and arranged for further observation of the network under observation.

3.1. Architecture

The proposed testbed's architecture follows a dynamic cluster-based network communication model, with an aim to give maximum flexibility to the user on a number of parameters including but not limited to:

- Number of nodes on each cluster
- Cluster-level communication protocol parameters
- Cluster topology, malicious node position, state and communication procedure
- Defensive algorithm deployment
- Cross-cluster communication procedure.

3.1.1. Network layout

The entire network is divided into M clusters denoted by Cx where $x \in \{1, 2, 3, \ldots, M\}$ and each cluster has n dynamic nodes denoted with reference to its parent cluster as $CxNy$ where $x \in \{1, 2, 3, \ldots, M\}$ and $y \in \{1, 2, \ldots, n\}$. For example, $C2N5$ denotes 5th Node of 2nd Cluster.

Each cluster is connected to 3 interconnected routers denoted by $R1$, $R2$ and $R3$ via single switch node denoted by $CxSW$ where $x \in \{1, 2, 3, \ldots, M\}$, for example, $C5SW$ denotes switch of 5th cluster.

In *Figs. 3 and 4*, we have 5 Clusters with 5 Node per Cluster (NpC) and 10 Nodes per Cluster.

3.1.2. Routers

The three main routers of the base network are interconnected as in Fig. 5(a), this setup is static and connects all clusters together.

3.1.3. Switches

Switches are the single point of contact that connects the clusters to routers, which ultimately enables them to perform inter-cluster communication. Each router is connected via a static number of switches.

In our current arrangement $R1$ is connected with $C1SW$&$C2SW$, $R2$ is connected with $C3SW$ and $R3$ is connected with $C4SW$ and $C5SW$ as shown in Figs. 5(b), 5(c), 5(d).

3.1.4. Clusters

Clusters are dynamically arranged entities of simulation, they are structured according to n parameter.

Cluster 1

In Cluster 1 (Fig. 6) $C1SW$ is connected to $C1N0$ and all the nodes from $C1N1$ to $C1Nn$ are connected directly to $C1N0$.

Cluster 2

Cluster 2 (Fig. 7) follows 'ring topology' and all the nodes are connected to the next node in numeric order, the last node is connected back to $C2N0$ to complete the ring formation. $C2N0$ is connected to $C2SW$ and $C2N1$.

Cluster 3

All the nodes in Cluster 3 are directly connected to $C3SW$ as shown in Fig. 8

Cluster 4

Cluster 4 uses two pivot nodes $C4N0$ and $C4N(\lfloor(n \div 2)\rfloor)$. All the nodes from $C4N1$ to $C4N(\lfloor(n \div 2) - 1\rfloor)$ are connected to $C4N0$, and all the nodes from $C4N(\lfloor(n \div 2) + 1\rfloor)$ to $C4N[n]$ are connected to $C4N(\lfloor(n \div 2)\rfloor)$ as in Fig. 9.

Cluster 5

Cluster 5's arrangement is inspired by mesh topology, every node in this cluster are interconnected with every other node in the cluster, as shown in Fig. 10. In real life implementation this might increase cost, but provides best recovery as one bad link does not disturb whole cluster.

3.2. Simulation component

The proposed testbed utilizes NS2 as the back-end to simulate the events via *.tcl* scripts. These scripts are generated using Python3 API.[4] The python script takes input from the user such as node per cluster (*NpC*), number of clusters (*M*), malicious node

[4] Application Programming Interface.

Table 1
Feature Comparison

Features	NICS Testbed	Cisco Packet Tracer	OMNet++	OPNET	NetSim	QualNet	TOSSIM
Preserves OS execution model	YES	YES	YES	YES	YES	YES	YES
Enables real-time simulation	YES	YES	YES	YES	YES	YES	NO
Hardware emulation	YES, via NS3	NO	NO	NO	NO	NO	NO
Can be used with multiple OS	YES	YES	YES	YES	YES	YES	NO
Customizable simulation detail	YES	YES	YES	YES	YES	YES	NO
Updated to improve performance regularly	YES	YES	YES	YES	YES	YES	NO
Incorporate Energy models	YES	YES	NO	NO	YES	YES	YES
Incorporate NICS	YES	NO	NO	NO	NO	NO	NO
Provide API to plug user NICS based script	YES	NO	NO	NO	NO	NO	NO

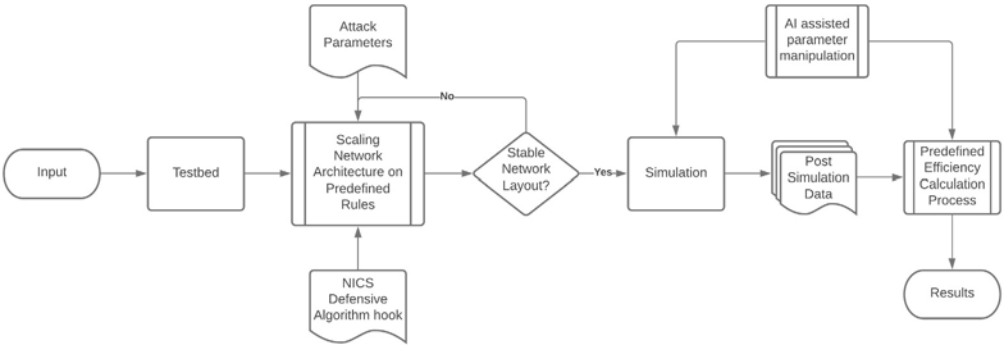


Fig. 2. Flow diagram of testbed.

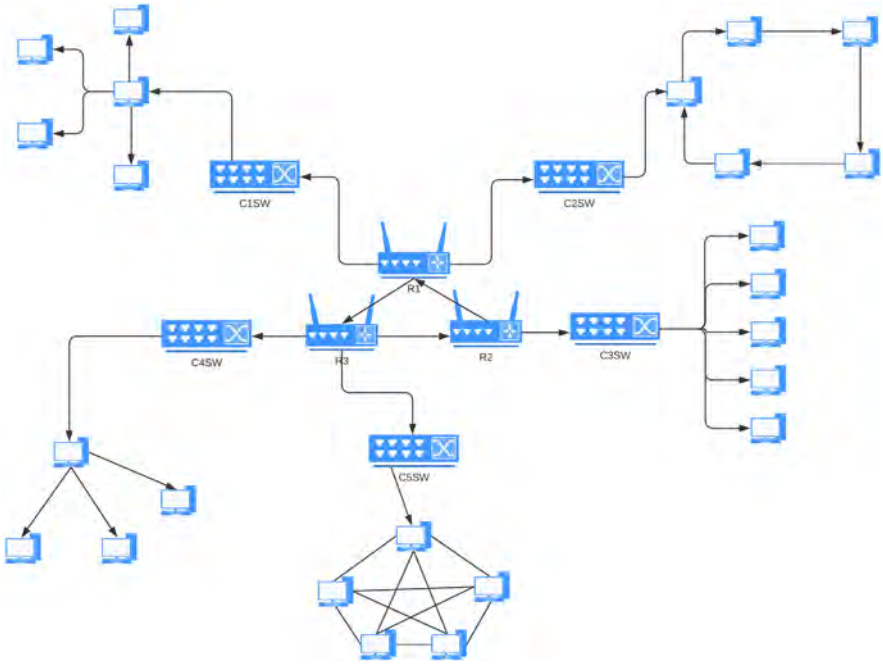


Fig. 3. Network with 5 Nodes per Cluster configuration.

properties, and simulation time (t). These parameters are used in the testbed to scale up/down the network.

The proposed testbed is implemented in the previous version, i.e. NS2, while the current version is NS3. The use of NS2 is one of the limitations of the proposed work. We implemented the proposed testbed in NS2 because of the wide availability of scripts and testbed in .tcl scripts, which helped fast development. Although we have planned to extend the support of the proposed testbed to other network simulators like OMNET++ and MATLAB, we will also add support for NS3.

3.3. Attack component

The attack component in the proposed testbed is user-defined malicious node $MALN(x)$, which can be configured to attack any node in the given network architecture.

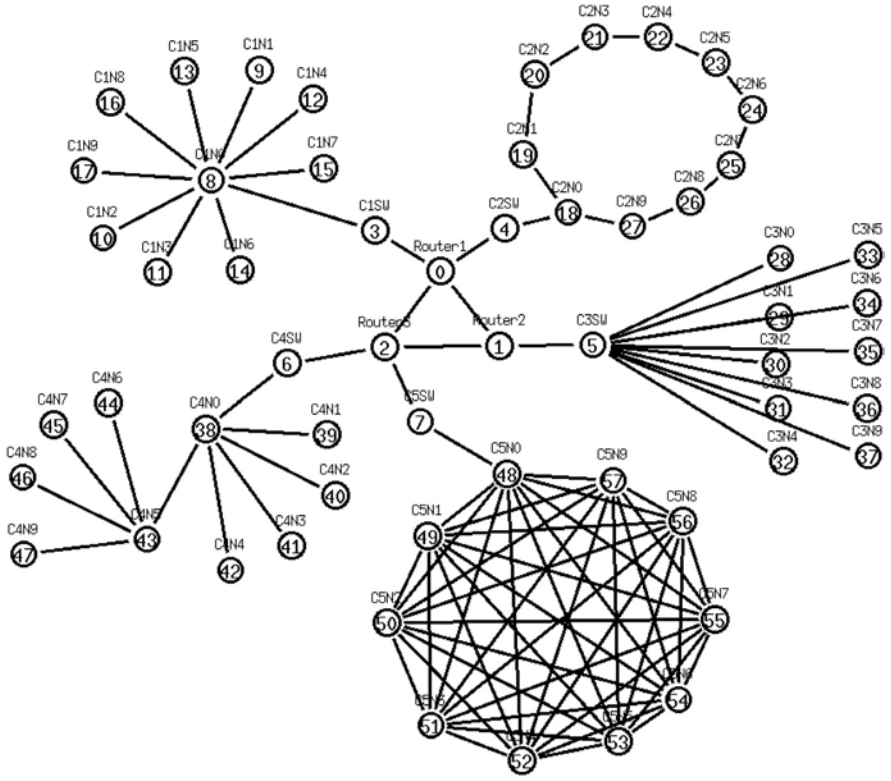
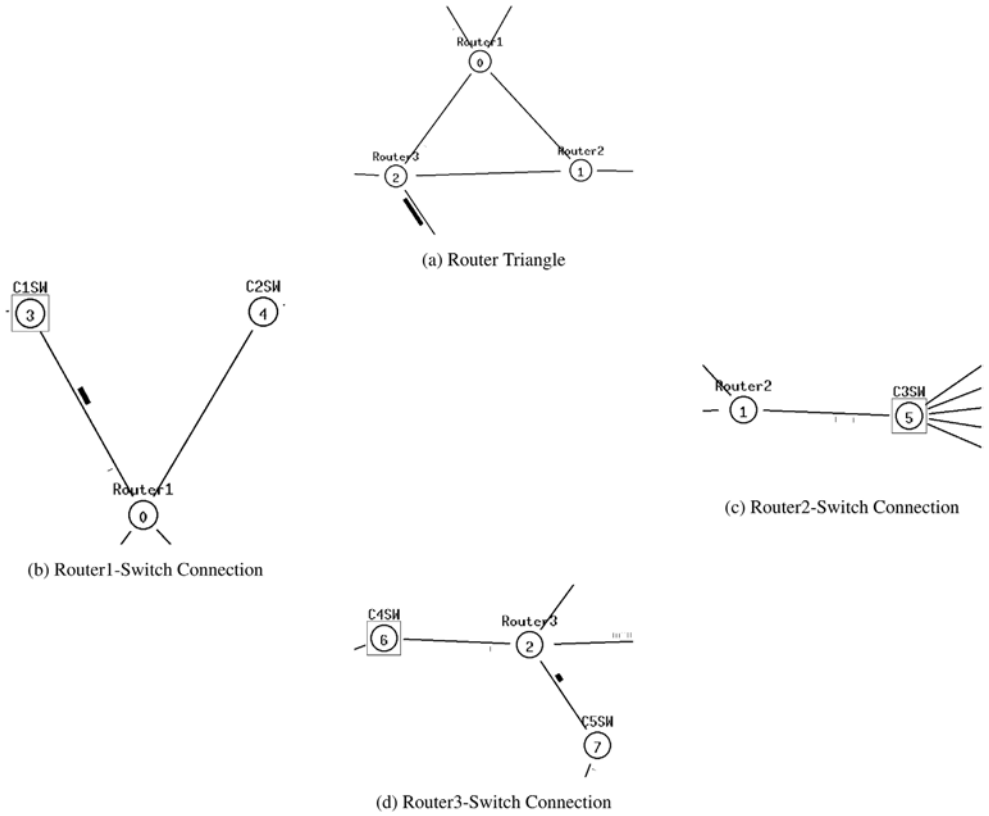


Fig. 4. Network with 10 Node per Cluster configuration.



(a) Router Triangle

(b) Router1-Switch Connection

(c) Router2-Switch Connection

(d) Router3-Switch Connection

Fig. 5. R*x*-R*y* and R*x* - C(*M*)SW Connections.

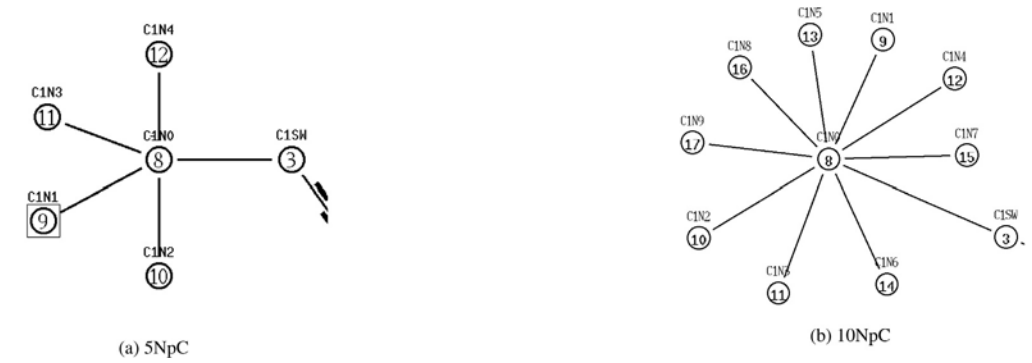


Fig. 6. Cluster 1 arrangement with 5NpCs & 10NpC.

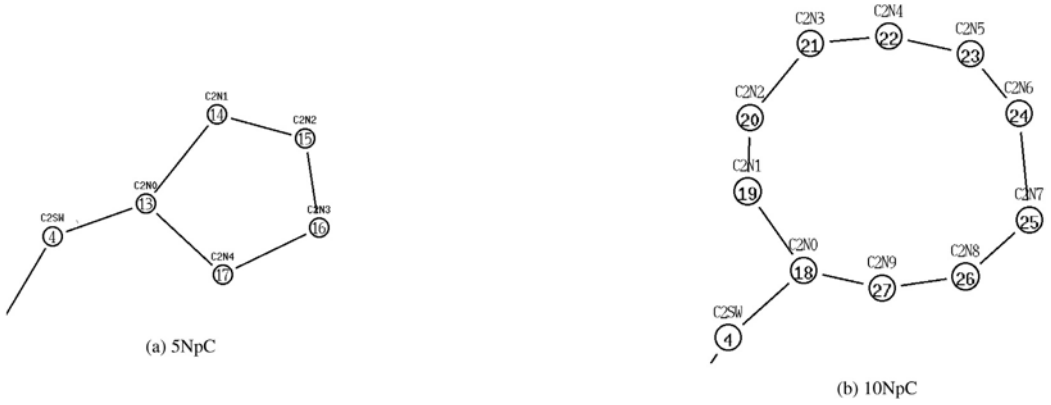
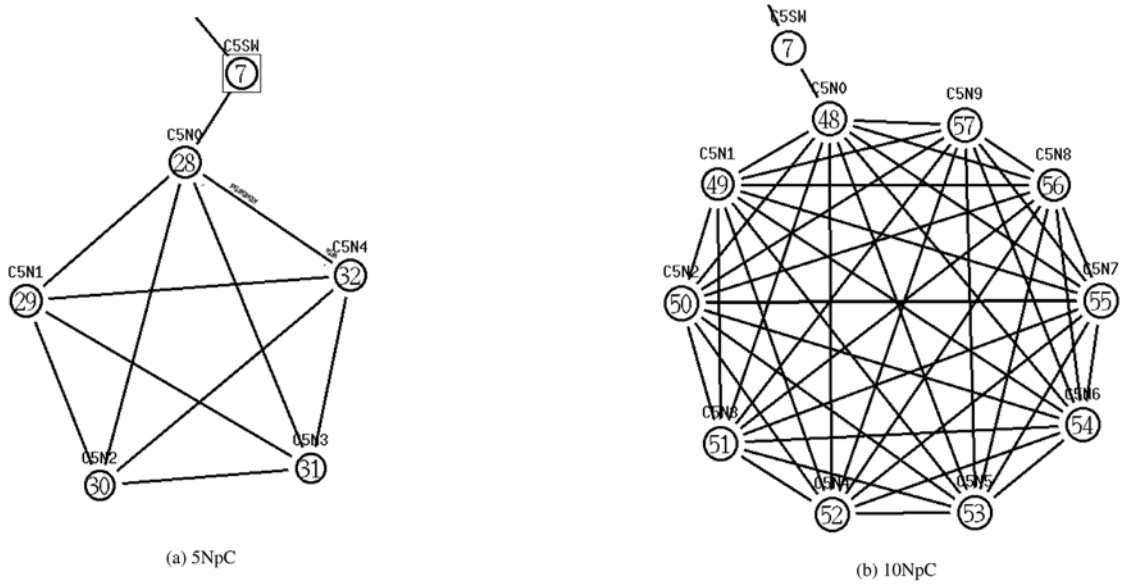


Fig. 7. Cluster 2 arrangement with 5NpCs & 10NpC.

Fig. 8. Cluster 3 arrangement with 5NpCs & 10NpC.

Fig. 9. Cluster 4 arrangement with 5NpCs & 10NpC.



(a) 5NpC (b) 10NpC

Fig. 10. Cluster 5 arrangement with 5NpCs & 10NpC.

Attack component can also be a new malicious cluster $MALC(y)$ defined by user where $MALN(x) \in MALC(y)$ and $x, y \in \{1, 2, 3, \ldots, \infty\}$.

3.4. User interaction and interface

The User Interaction module provides the interface of the testbed to the user. The user can interact with the testbed in two ways: a web-based GUI or a terminal-based component (command-line interface (CLI)). The GUI interface eliminates the technical difficulty of using the testbed, while CLI helps expert users fasten the experiment's configuration. Depending upon the requirement of the experiment, the user can use either method to provide parameters to the testbed and fetch results from it. Having a unified user interaction module will also help extend the testbed with other evolving technologies like IoT and edge computing without changing the interface of the testbed.

3.5. Reporting and visualization

The proposed testbed provides two major methods for result visualization,

- Network Animation Scripts (produced after each simulation)
- Report graphs (plots of performance metric and parameter calculation)

However, it can be further extended as per the requirements of experiment.

4. Experiment and result

4.1. General testbed setup and attack simulation

4.1.1. Experimental setup

We have simulated *Low-rate TCP attack*, which is a low-rate DoS[5] attacks that attempt to deny bandwidth to TCP flows while sending at sufficiently low average rate to elude detection by counter-DoS mechanism [23]. The *Low-Rate TCP Attack* was performed on Router2 of networks using $N = 5$ with $5NpC$, $10NpC$ and $50NpC$. where each simulation runs for 1 min of network activity time. This can be customized as per the requirement. The configuration details of each node along with other simulation parameters are listed in Table 2.

Table 2
Node configuration and simulation parameters.

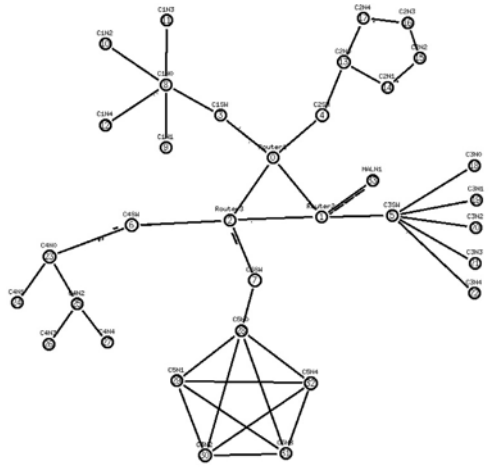
Node Configuration Attribute	Value
Link type	duplex-link
Link bandwidth	1000.0 Mb / 200 Mb
Link Latency	5ms
Network interface type	ETH
MAC type	Mac/802_11
Interface queue type	Queue/DropTail/PriQueue
Link layer type	LL
Queue Limit	50
Routing protocol	DAgent
Energy model	EnergyModel
Agent trace	ON
Router trace	ON
Movement trace	OFF

Network communication is established between clusters as following:

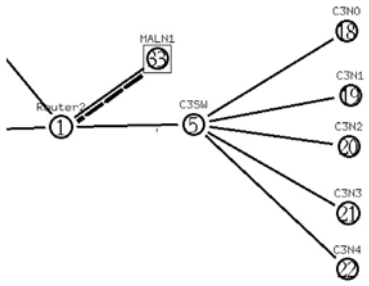
- $C1N(x)$ where $x \in \{1, \ldots, n\}$ is sending Telnet packets of 500 Mb to $C5N(n)$ at interval of 0.01 s
- $C2N(x)$ where $x \in \{1, \ldots, n\}$ is sending FTP packets of 500 Mb to $C4N(n)$ at interval of 0.01 s
- $C3N(x)$ where $x \in \{1, \ldots, n\}$ is sending SMTP packets of size $200b$ with $burst\_time = 50$ ms and $idle\_time = 50$ ms to $C1N(n)$ at rate of $100k$
- $C4N(x)$ where $x \in \{1, \ldots, n\}$ is sending HTTP packets of size $1000b$ at rate of 1.0 Mbps to $C3N(n)$
- $C5N(x)$ where $x \in \{1, \ldots, n\}$ is sending TCP CBR packets to $C1N(n)$
- $MALN1$ is sending $1000b$ TCP CBR packets to $C5N(x)$ via $R3$ and $R2$ (low rate)

The aforementioned parameters will simulate a 'Low-Rate TCP DoS' attack on $R2$ (Fig. 11(a)) which should decrease the performance of $R3$ and $R2$ significantly and might affect other clusters as well. We can then implement defensive algorithms on $R2$ and $R3$ and calculate their performance in comparison with *normal* and

[5] DoS:Denial of Service.

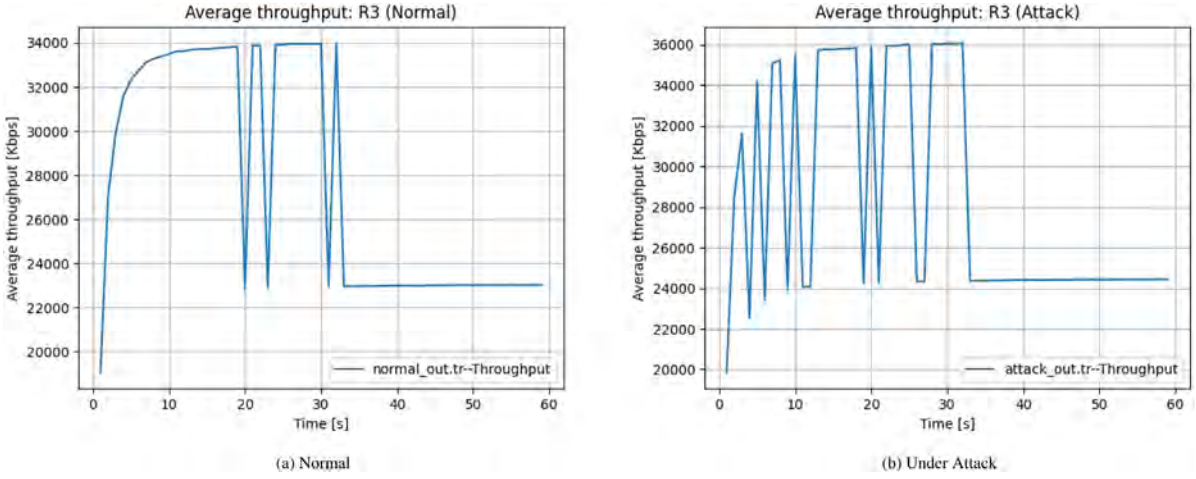


(a) Overall $5NpC$ with $MALN1$



(b) $MALN1$ position, $5NpC$

Fig. 11.



(a) Normal (b) Under Attack

Fig. 12. Avg. Throughput of R3 with respect to time, 5 Node per Cluster.

attack scenario to check if that improves the network stability and if it does then by what margin.

4.1.2. Performance evaluation

Every simulation in the proposed testbed provides a *trace file* as the output of various configuration parameters. The *trace file* has all the information required to calculate different metrics' values that can be used for the performance evaluation of methods and approaches. In the proposed work, we have carried out few experiments to showcase the functionality of the testbed, and we have used the *'Average Throughput v/s Time'* plot to determine each node performance. We expect to observe a significant performance drop of $R3$, impacting other network components when under attack. After deploying a defensive mechanism, a recovery for throughput is expected in the network.

4.1.3. 5 nodes per cluster

In this simulation we set $N = 5$, $n = 5$ and $t = 60.0$. In first execution we set *activate_mal* = *False* and calculated avg. throughput shown in Fig. 12(a). Then we set *activate_mal* = *True* to activate $MALN1$ and calculated avg. throughput of $R3$ shown in Fig. 12(b).

By observing Fig. 12 we can see that before the attack we see a smooth rise in throughput from 1 s to 19 s and then it drops but rises again from 23 s to 31 s mark.

While observing Fig. 12(b) we see that $R3$ struggles to stabilize and performance drops constantly in the initial 10 s frame and then stabilizes for a bit and drops again.

4.1.4. 10 nodes per cluster

In this simulation we set $N = 5$, $n = 10$ and $t = 60.0$. In first execution we set *activate_mal* = *False* and calculated avg. throughput shown in Fig. 13(a). Then we set *activate_mal* = *True* to activate $MALN1$ and calculated avg. throughput of $R3$ shown in Fig. 13(b).

From observation of Fig. 13 we can see that during normal execution $R3$ maintains high throughput from around 17 s to end of simulation, i.e. 60 s.

While looking at Fig. 13(b) we can observe that performance drops are regular as $R3$ struggles to maintain stability under activation of the malicious node.

4.1.5. 50 nodes per cluster

In this simulation we set $N = 5$, $n = 50$ and $t = 60.0$. In first execution we set *activate_mal* = *False* and calculated avg. throughput shown in Fig. 14(a). Then we set *activate_mal* = *True* to activate $MALN1$ and calculated avg. throughput of $R3$ shown in Fig. 14(b).

Here we can see that the impact of one malicious node decreases when we increase NpC significantly, and we only observe

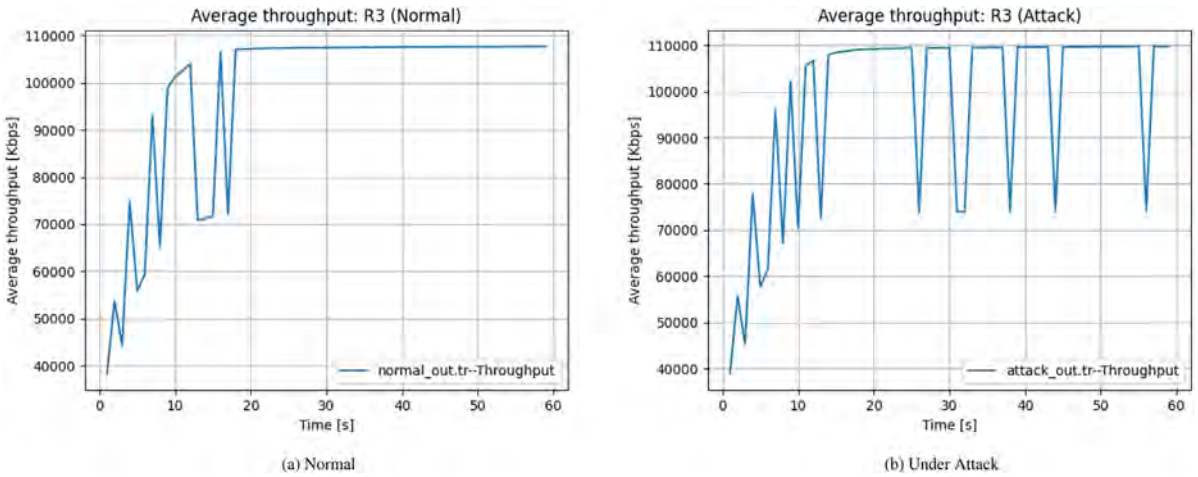


Fig. 13. Avg. Throughput of R3 with respect to time, 10NpC.

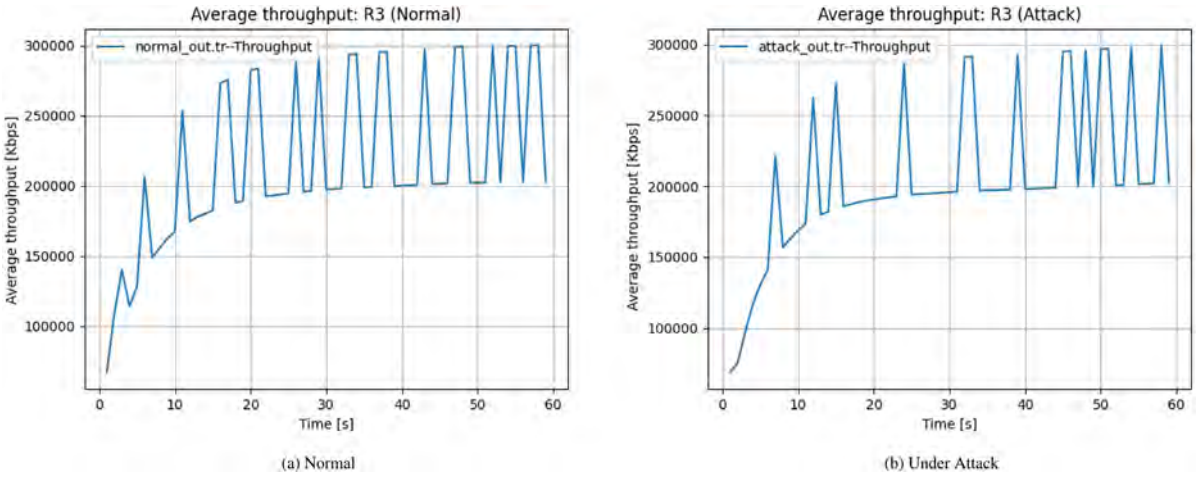


Fig. 14. Avg. Throughput of R3 with respect to time, 50NpC.

slight deviation from regular performance, but still influence of *MALN*1 can be seen as we have more valleys than peaks in Fig. 14(b) in comparison to Fig. 14(a).

Also, most of the time average throughput of Fig. 14(b) stays between 150000 Kbps–200000 Kbps range. On the other hand, we see a significant jump in the range of 250000 Kbps–300000 Kbps in Fig. 14(a)

4.2. NICS based adaptive defense and analysis

NICS consists of many nature-inspired approaches that can be used to build an adaptive defense. Like any other defense mechanism, NICS also introduces network load due to network traffic and requires additional procedures at all nodes. In the case of NICS, network latency can be a serious issue that could lead to more burden on the network. Therefore, careful selection of appropriate defensive method as per the target network, and its critical observation is necessary, to avoid the problems like poor throughput, and overload on traffic. Many researchers have trusted the Firefly algorithm for clustering and global optima [34]. Firefly algorithms are capable of solving multi-search problems [35,36]. These algorithms are based on the behavior of fireflies to flashing light. In this example, each node of the network can be considered as one firefly which can be further compared with other nodes on the basis of expected throughput on that point. In the presence of attack, specific nodes possess low throughput and thereby require remedies. Firefly algorithms can search the alternative of these nodes using multi-objective search to maintain the QoS of the network (throughput in this case). Implementation of AI can further enhance this adaptability of the network (generated by NICS), using learning classifiers [34]. As the algorithm will run for all nodes and every time when any update will happen in the network, it has to be propagated which will generate more load, therefore the accuracy and learning of classifiers are to be monitored rigorously which is possible by the proposed testbed.

4.2.1. 5 nodes per cluster

In this simulation we set $N = 5$, $n = 5$ and $t = 60.0$. In first execution we set *activate_mal* = *False* and calculated avg. throughput shown in Fig. 15. Then we set *activate_mal* = *True* to activate *MALN*1 and *defense_hook* = *True* to activate firefly inspired NICS defensive algorithm and calculated avg. throughput of *R*3 shown in Fig. 15 In Fig. 15 we can observe that after 6th second our adaptive defense kicks-in and prevents significant drops in performance after that and keeps network at acceptable stability while nearly eliminating effect of *MALN*1. It is also interesting to observe that while the avg. throughput of adaptive defense is still lower than that of *normal* operation, we do not suffer during event of active *attack*

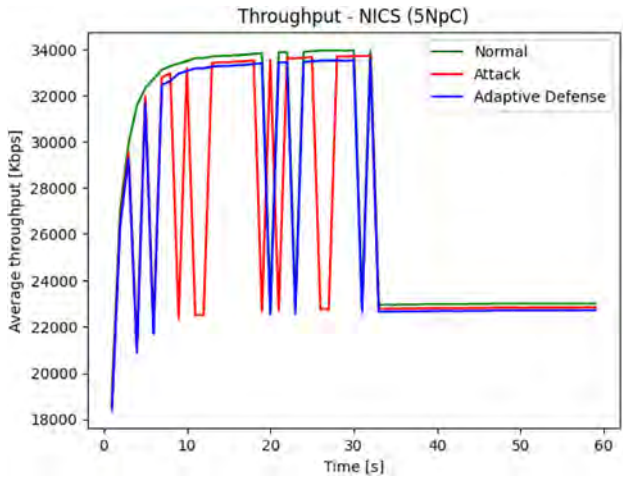


Fig. 15. Adaptive Defense Performance with 5NpC.

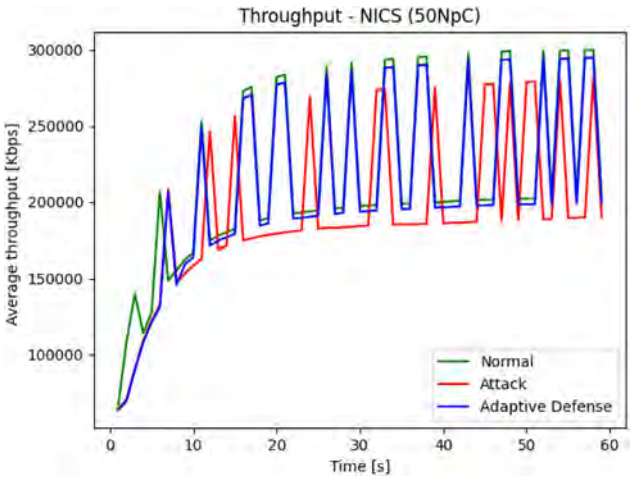


Fig. 17. Adaptive Defense Performance with 50 Nodes per Cluster.

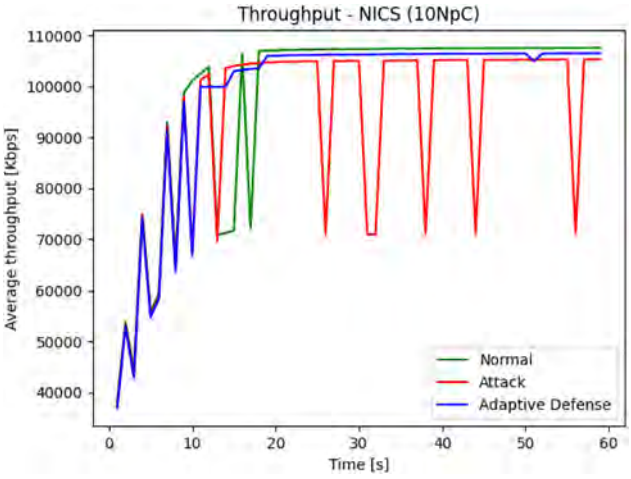


Fig. 16. Adaptive Defense Performance with 10 Node per Cluster.

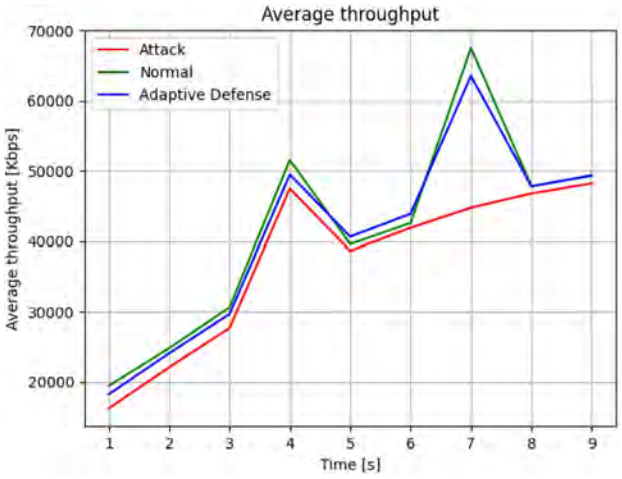


Fig. 18. Case2: Adaptive Defense Performance with 5 Nodes per Cluster.

4.2.2. 10 nodes per cluster

In this simulation we set $N = 5$, $n = 10$ and $t = 60.0$. In first execution we set *activate_mal* = *False* and calculated avg. throughput shown as *green* plot in Fig. 16. Then we set *activate_mal* = *True* to activate *MALN*1 and *defense_hook* = *True* to activate firefly inspired NICS defensive algorithm and calculated avg. throughput of *R*3 shown by *red* and *blue* plots in Fig. 16 In Fig. 16 we observe activation of adaptive defense at around 10th second and the network begins to stabilize at an acceptable performance. During the initial 10-second time frame, the performance is the same as the *attack* trend, this is due to the fact that most adaptive defense algorithms take some time to activate and find an optimal defensive solution. In Fig. 16 we can observe slight performance loss in comparison to *normal* operation which is due to the inevitable performance-security trade-off that comes with NICS strategies. From a 10–21 s time frame we can see the attempt of a defensive mechanism to stabilize network traffic.

4.2.3. 50 nodes per cluster

In this simulation we set $N = 5$, $n = 50$ and $t = 60.0$. In first execution we set *activate_mal* = *False* and calculated avg. throughput shown as *green* plot in Fig. 17. Then in next executions we set *activate_mal* = *True* to activate *MALN*1 and *defense_hook* = *True* to activate firefly inspired NICS defensive algorithm and calculated avg. throughput of *R*3 shown by *red* and *blue* plots in Fig. 17

In Fig. 17, we can see the defensive algorithms activates around 8th second and then almost traces the performance graph of normal execution. From our previous observation from 50NpC Attack [4.1.5] we know that the effect of Low-Rate TCP Denial of Service is a Denial of Service or DoS attack which exploits the vulnerability of TCP's congestion control mechanism by periodically sending attack packets continuously, launching attack packets at a constant low-rate. This attack is not much significant, hence in Fig. 17 we see close relation and small deviation in performance of both *attack* and *active_defense*. After initial 8 s, network stabilizes at acceptable performance. Apart from signifying the importance of an AI-assisted NICS-based defense system for the adaptive response, the presented experiment and results also show that network performance vs security trade-off should be kept in mind when implementing AI and NICS based techniques for network defense.

Fig. 18 shows results of another use-case, from which it is evident that the proposed testbed is capable for quickly setting up the different network scenarios for NICS-based experiments with detailed configuration of various network parameters.

5. Conclusion and future scope

We have presented a testbed specifically for designing and experimenting with computer network operations while utilizing the benefits of AI and NICS to achieve an intelligent and adaptive defense system. The proposed testbed is fully capable of incorporating the peculiarities of both AI and NICS techniques for an effective network simulation. We have presented the general architecture of the proposed testbed and Low-Rate TCP attack scenario with experiment results on throughput. A prototype of AI-assisted NICS-based (firefly inspired) adaptive defense system is also demonstrated, where we have compared the network performance in different scenarios namely, 'normal', 'under attack', and 'under attack in presence of NICS-based adaptive defense mechanisms' and achieved stable experimental results. Selection and appropriate defensive system for an organization is a critical decision. The proposed testbed can be the solution that offers full-customization on various parameters under observation, a high degree of sensitivity analysis, and effective performance-tuning on various operational parameters, without the risk of losing data and integrity. The immediate future work can be a web application service of the proposed testbed and establishing the testbed as standard NICS-based testbed through rich libraries.

5.1. Applicability with evolving modern technologies

Recently, nature-inspired algorithms are being used in evolving modern technology like Internet-of-Things and Edge computing. SmartFog is a Fog architecture build upon nature-inspired algorithms that can be helpful in low decision making latency and adaptive resource management [37]. Similar to SmartFog, Samah et al. [38] have used genetic algorithms and queuing networks to proposed an effective offloading mechanism for mobile edge computing. As an application of IoT and nature-inspired algorithms, Emmanuel Freeman et al. [39] has proposed an algorithm (based on Kestrel bird behavior) that can detect water leakage and report the accurate location to the management. For the security of 5G-enabled IoT applications, authors have presented a detailed survey on the application and usage of bio-inspired algorithms for the network layer of the architecture [40]. In the fore-mentioned application areas, the nature-inspired cyber security algorithms can play significant role if developed, tested and deployed properly. It is important to provide a detailed modeling and simulation of attacks and defense for these evolving technologies, and the plug-and-play interface of the framework. In the future extension of the proposed work, we will add the specific interfaces to adopt IoT and edge related experimental simulation.

CRediT authorship contribution statement

Shishir Kumar Shandilya: Conceptualization, Methodology, Data curation, Writing – original draft preparation. **Saket Upadhyay:** Software, Development, Validation, Writing. **Ajit Kumar:** Writing – reviewing and editing, Development, Validation. **Atulya K. Nagar:** Project administration, Supervision.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

References

[1] S. Bitam, S. Zeadally, A. Mellouk, Bio-inspired cybersecurity for wireless sensor networks, IEEE Commun. Mag. 54 (6) (2016) 68–74.

[2] U. Rauf, A taxonomy of bio-inspired cyber security approaches: existing techniques and future directions, Arab. J. Sci. Eng. 43 (12) (2018) 6693–6708.

[3] K. Demertzis, L. Iliadis, A bio-inspired hybrid artificial intelligence framework for cyber security, in: Computation, Cryptography, and Network Security, Springer, 2015, pp. 161–193.

[4] S.N. Mthunzi, E. Benkhelifa, T. Bosakowski, S. Hariri, A bio-inspired approach to cyber security, in: Machine Learning for Computer and Cyber Security: Principle, Algorithms, and Practices, CRC Press, Boca Raton, FL, USA, 2019, p. 75.

[5] M.E. Kuhl, M. Sudit, J. Kistner, K. Costantini, Cyber attack modeling and simulation for network security analysis, in: 2007 Winter Simulation Conference, IEEE, 2007, pp. 1180–1188.

[6] D.S. Fowler, M. Cheah, S.A. Shaikh, J. Bryans, Towards a testbed for automotive cybersecurity, in: 2017 IEEE International Conference on Software Testing, Verification and Validation (ICST), IEEE, 2017, pp. 540–541.

[7] G. Apruzzese, M. Colajanni, L. Ferretti, A. Guido, M. Marchetti, On the effectiveness of machine and deep learning for cyber security, in: 2018 10th International Conference on Cyber Conflict (CyCon), IEEE, 2018, pp. 371–390.

[8] H. Jiang, T. Choi, R.K. Ko, Pandora: A cyber range environment for the safe testing and deployment of autonomous cyber attack tools, 2020, arXiv preprint arXiv:2009.11484.

[9] M. Atighetchi, P. Pal, F. Webber, C. Jones, Adaptive use of network-centric mechanisms in cyber-defense, in: Sixth IEEE International Symposium on Object-Oriented Real-Time Distributed Computing, 2003, IEEE, 2003, pp. 183–192.

[10] G. Cybenko, M. Wellman, P. Liu, M. Zhu, Overview of control and game theory in adaptive cyber defenses, in: Adversarial and Uncertain Reasoning for Adaptive Cyber Defense, Springer, 2019, pp. 1–11.

[11] Z. Hu, P. Chen, M. Zhu, P. Liu, Reinforcement learning for adaptive cyber defense against zero-day attacks, in: Adversarial and Uncertain Reasoning for Adaptive Cyber Defense, Springer, 2019, pp. 54–93.

[12] N. Wagner, C.S. Sahin, J. Pena, W.W. Streilein, A nature-inspired decision system for secure cyber network architecture, in: 2017 IEEE Symposium Series on Computational Intelligence (SSCI), IEEE, 2017, pp. 1–8.

[13] T.U.S. Joint Chiefs of Staff, Information operations, 2014, Joint Publication 3-13, URL Online;accessed10-January-2020.

[14] S.-D. Chi, J.S. Park, K.-C. Jung, J.-S. Lee, Network security modeling and cyber attack simulation methodology, in: Australasian Conference on Information Security and Privacy, Springer, 2001, pp. 320–333.

[15] R. Pal, L. Golubchik, K. Psounis, T. Bandyopadhyay, On robust estimates of correlated risk in cyber-insured IT firms: A first look at optimal AI-based estimates under "small" data, ACM Trans. Manag. Inf. Syst. 10 (3) (2019) 1–18.

[16] J.B. Fraley, J. Cannady, The promise of machine learning in cybersecurity, in: SoutheastCon 2017, IEEE, 2017, pp. 1–6.

[17] I. Ghafir, M. Hammoudeh, V. Prenosil, L. Han, R. Hegarty, K. Rabie, F.J. Aparicio-Navarro, Detection of advanced persistent threat using machine-learning correlation analysis, Future Gener. Comput. Syst. 89 (2018) 349–359.

[18] M. Breza, J. McCann, Lessons in implementing bio-inspired algorithms on wireless sensor networks, in: 2008 NASA/ESA Conference on Adaptive Hardware and Systems, 2008, pp. 271–276, http://dx.doi.org/10.1109/AHS.2008.72.

[19] S. Mthunzi, E. Benkhelifa, T. Bosakowski, S. Hariri, A Bio-inspired Approach To Cyber Security: Principles, Algorithms, and Practices, 2019, pp. 75–104, http://dx.doi.org/10.1201/9780429504044-4.

[20] N. Pankov, Applied cyberimmunity: What is it?, 2019, https://www.kaspersky.com/blog/applied-cyberimmunity/28772/, [Online; accessed 10-January-2020].

[21] V.D. Veksler, N. Buchler, B.E. Hoffman, D.N. Cassenti, C. Sample, S. Sugrim, Simulations in cyber-security: A review of cognitive modeling of network attackers, defenders, and users, Front. Psychol. 9 (2018) 691.

[22] D. Li, L. Deng, B.B. Gupta, H. Wang, C. Choi, A novel CNN based security guaranteed image watermarking generation scenario for smart city applications, Inform. Sci. 479 (2019) 432–447.

[23] A. Kuzmanovic, E.W. Knightly, Low-rate TCP-targeted denial of service attacks: the shrew vs. the mice and elephants, in: Proceedings of the 2003 Conference on Applications, Technologies, Architectures, and Protocols for Computer Communications, 2003, pp. 75–86.

[24] A. Mishra, N. Gupta, B. Gupta, Defense mechanisms against DDoS attack based on entropy in SDN-cloud using POX controller, Telecommun. Syst. 77 (1) (2021) 47–62.

[25] J. Davis, S. Magrath, A survey of cyber ranges and testbeds, Tech. rep., Defence Science and Technology Organisation, Edinburgh (Australia), 2013.

[26] C.L. Stergiou, K.E. Psannis, B.B. Gupta, Iot-based big data secure management in the fog over a 6G wireless network, IEEE Internet Things J. 8 (7) (2020) 5164–5171.
[27] E. Hildebrand, R. Flinterman, J. Mulder, A. Smit, Clusus: A cyber range for network attack simulations, 2019.
[28] M. Leitner, M. Frank, W. Hotwagner, G. Langner, O. Maurhart, T. Pahi, L. Reuter, F. Skopik, P. Smith, M. Warum, AIT cyber range: Flexible cyber security environment for exercises, training and research, in: European Interdisciplinary Cybersecurity Conference (EICC), 2020, pp. 18–19.
[29] J. Vykopal, R. Ošlejšek, P. Čeleda, M. Vizvary, D. Tovarňák, Kypo cyber range: Design and use cases, SciTePress, 2017.
[30] J. Vykopal, M. Vizváry, R. Oslejsek, P. Celeda, D. Tovarnak, Lessons learned from complex hands-on defence exercises in a cyber range, in: 2017 IEEE Frontiers in Education Conference (FIE), IEEE, 2017, pp. 1–8.
[31] C. Siaterlis, A.P. Garcia, B. Genge, On the use of emulab testbeds for scientifically rigorous experiments, IEEE Commun. Surv. Tutor. 15 (2) (2012) 929–942.
[32] T. Benzel, The science of cyber security experimentation: the DETER project, in: Proceedings of the 27th Annual Computer Security Applications Conference, 2011, pp. 137–148.
[33] J. Mayo, R. Minnich, D. Rudish, R. Armstrong, Approaches for scalable modeling and emulation of cyber systems: Ldrd final report, Sandia Report, SAND2009-6068, Sandia National Lab, Citeseer, 2009.
[34] K.M. Prasad, A.R.M. Reddy, K.V. Rao, Bartd: Bio-inspired anomaly based real time detection of under rated app-ddos attack on web, J. King Saud Univ. - Comput. Inf. Sci. 32 (1) (2020) 73–87.
[35] H. Wang, W. Wang, X. Zhou, H. Sun, J. Zhao, X. Yu, Z. Cui, Firefly algorithm with neighborhood attraction, Inform. Sci. 382 (2017) 374–387.
[36] H. Wang, W. Wang, L. Cui, H. Sun, J. Zhao, Y. Wang, Y. Xue, A hybrid multi-objective firefly algorithm for big data optimization, Appl. Soft Comput. 69 (2018) 806–815.
[37] D. Kimovski, H. Ijaz, N. Saurabh, R. Prodan, Adaptive nature-inspired fog architecture, in: 2018 IEEE 2nd International Conference on Fog and Edge Computing (ICFEC), IEEE, 2018, pp. 1–8.
[38] S.A. Zakaryia, S.A. Ahmed, M.K. Hussein, Evolutionary offloading in an edge environment, Egypt. Inform. J. (2020).
[39] E. Freeman, D.A. Quaye, I.E. Agbehadji, R.C. Millham, Nature-inspired search method for IoT-based water leakage location detection system, in: 2019 International Conference on Mechatronics, Remote Sensing, Information Systems and Industrial Information Technologies (ICMRSISIIT), Vol. 1, IEEE, 2019, pp. 1–8.
[40] K. Saleem, G.M. Alabduljabbar, N. Alrowais, J. Al-Muhtadi, M. Imran, J.J. Rodrigues, Bio-inspired network security for 5G-enabled IoT applications, IEEE Access 8 (2020) 229152–229160.

Shishir Kumar Shandilya is Deputy Director of SECURE - Centre of Excellence in Cyber Security and Division Head of Cyber Security & Digital Forensics at Vellore Institute of Technology, VIT Bhopal University, India, and Visiting Researcher at Liverpool Hope University, United Kingdom. He is also a Cambridge University Certified Professional Teacher & Trainer, and a Senior Member of IEEE. He is an Academic Advisor to National Cyber Safety & Security Standards, New Delhi. He has received IDA Teaching Excellence Award for distinctive use of technology in Teaching by Indian Didactics Association, Bangalore (2016) and Young Scientist Award for two consecutive years, 2005 and 2006, by Indian Science Congress & MP Council of Science & Technology. He has seven books published by Springer Nature, IGI-USA, River-Denmark, and Prentice Hall of India. His recently published book is on Advances in Cyber Security Analytics and Decision Systems by Springer. His research interests include important aspects of Advanced Cyber Security, with a focus on Advanced Persistent Threats (APT). He has made significant contributions to related areas such as Camouflage of network architectures, and Proactive Cyber Defense.

Saket Upadhyay is Cybersecurity and Digital Forensics student at VIT Bhopal University. He has worked in the field of penetration testing and vulnerability assessment with reputed firms, won multiple international awards for his participation in security competitions and holds "Best Paper Award" from IEEE Big Data Conference 2019 for his research contribution on the framework for malware detection with multiple ML models. He is interested in reverse engineering, malware analysis and cyber defense. He likes learning new technologies and approach in the field of cybersecurity. Current research area includes malware detection using machine learning, Nature Inspired Cyber Security and Adaptive Cyber Defense strategies.

Ajit Kumar is a Post-doctoral research fellow at Soongsil University, Seoul, South Korea. He has completed his Ph.D. in Computer Science and Engineering from the Department of Computer Science, Pondicherry University in May 2018. He has received his Bachelor of Computer Application (BCA) from IGNOU in the year 2009 and a Master of Computer Science in the year 2011, from Pondicherry University. His area of interest includes Information Security, Malware detection using Machine learning. He has published 6 research articles in peer-reviewed International Journal and has also presented and published his research in International IEEE and Elsevier conferences.

Prof. Atulya K Nagar is Pro-Vice Chancellor (Research) and the Foundation Professor of Computer and Mathematical Sciences at Liverpool Hope University. He is also Head of Department of Computer Science Department. A mathematician by training, Prof. Nagar brings multi-disciplinary expertise in computational science, bioinformatics, operations research and systems engineering to the Faculty of Business & Computer Sciences. He received a prestigious Commonwealth Fellowship for pursuing his Doctorate in applied non-linear mathematics, which he received from the University of York in 1996. Prof. Nagar is an internationally recognized scholar working at the cutting edge of theoretical computer science, applied mathematical analysis, operations research, and industrial systems engineering. The center of his research expertise lies in his IDS group, which pursues strategic and applied research into advancing applications of engineering, computational and biological systems. The research of the group seeks to contribute to the general body of knowledge and to influence IT practice in systems modeling and planning, scheduling, optimization, and informatics. One such innovative theme is DNA sequence analysis using sophisticated computational techniques. The work of the group is highly theoretical, and primarily benefits the scientific community, with demonstrable potential for practical applications and relevance to society as a whole. Prof. Nagar has published a substantial number of research papers in reputed outlets such as the IEE and IEEE publications. He has co-edited a volume on Intelligent Systems area and serves on editorial boards for a number of prestigious journals including the International Journal of Artificial Intelligence and Soft Computing, and the Journal of Universal Computer Science. Prof. Nagar was a Conference Chair for the European Modelling Symposium (EMS 2008); currently he is a Conference and TPC Chair for the Developments in E-Systems Engineering (DeSE'09) Conference (www.dese.org.uk); and he serves on International Programme Committees (IPC) for several international conferences. He has been an expert reviewer for the Biotechnology and Biological Sciences Research Council (BBSRC) grants peer-review committee for Bioinformatics Panel and has been selected to serve on the prestigious Peer-Review College of the Arts and Humanities Research Council (AHRC) as a Scientific/Technical expert member. He is a member of numerous professional organizations including the IEE; a fellow of the Higher Education Academy (FHEA); he is a member of the Council of Professors and Heads of Computing (CPHC); and has been listed in the invaluable reference Marquis' Who's Who in Science and Engineering. Prof. Nagar supervises Ph.D. research projects in Computer Science and serves on Ph.D. external examiner panels. He holds a Visiting Professorship at the University of Madras; and Adjunct Professorship at the Mathematics department at the Indian Institute of Technology (IIT), Roorkee. He is a member of the Board of Studies at Stella Maris College, India; external examiner for M.Sc. Computer Science programme at Staffordshire University. His teaching expertise is in Applied Analysis, Systems Engineering and Computational Biology. Prof. Nagar earned his Ph.D. in Applied Nonlinear Mathematics from the University of York (UK) in 1996. He holds B.Sc. (Hons.), M.Sc. and MPhil (with distinction) degrees, in Mathematical Sciences, from the MDS University of Ajmer, India. Prior to joining Liverpool Hope University, Prof. Nagar has worked for several years as a Senior Research Scientist, on various EPSRC sponsored research projects, in the department of Mathematical Sciences, and later in the department of Systems Engineering, at Brunel University. In the work at Brunel he has contributed to the development of new techniques based on mathematical control systems theory for modeling and analysis of uncertainty in complex decision making systems.