International Journal of Multidisciplinary Engineering in Current Research
ISSN: 2456-4265, Volume 7, Issue 2, February 2022, http://ijmec.com/

MALWARE ATTACK MODELING AND SIMULATION TOOL

Ravikumar Kandru[1], E. Akhil Bab2[1], Dr. B.Naveen Kumar[3]
[1] Assistant Professor, Department of CSE, RISE Krishna Sai Gandhi Group of Institutions, Ongole,[2] Assistant Professor, Department of CSE, RISE Krishna Sai Prakasam Group of Institutions, Ongole,[3]Associate Professor, Department of CSE, RISE Krishna Sai Gandhi Group of Institutions, Ongole.

Abstract— Malware threats have become a pervasive and continuously evolving challenge in the realm of cyber security. To effectively defend against these threats, it is imperative to develop and employ advanced tools and strategies that can simulate and analyze malware attacks. This paper introduces a comprehensive Malware Attack Modeling and Simulation Tool (MAMST) designed to address this need. MAMST is built on a foundation of Discrete Event System Specification (DEVS), enabling highly accurate and scalable simulations of diverse malware attack scenarios. The primary objective of MAMST is to provide cyber security professionals, researchers, and organizations with a versatile platform for assessing and enhancing their cyber security measures. The tool allows users to create, customize, and execute malware attack simulations within a controlled and safe environment. By modeling various malware types, propagation techniques, and attack vectors, MAMST empowers users to assess the resilience of their network infrastructure, security protocols, and incident response strategies.

KEY WORDS: MODELLING AND SIMULATION, DISCRETE EVENT SIMULATION, CYBER SECURITY, CYBER-ATTACK EXPERIMENTS, NETWORK TESTING ENVIRONMENTS

I. INTRODUCTION

In today's digital landscape, malware attacks are an ever-present threat to individuals, businesses, and governments. Developing effective defenses against malware is a critical aspect of cyber security, and the "Malware Attack Modeling and Simulation Tool" serves as a valuable resource in achieving this goal.

Key Features:

Realistic Malware Scenarios: The tool allows users to create and simulate a wide range of realistic malware scenarios, including viruses, ransomware, spyware, and more. This enables users to better understand the various tactics, techniques, and procedures employed by different types of malware.

Vulnerability Assessment: It helps identify vulnerabilities in a network, system, or application by modeling potential attack vectors that malware might exploit. This assists in proactively strengthening security measures.

Incident Response Training: The tool can be used for training and improving the incident response capabilities of cyber security teams. It allows for realistic drills and exercises to simulate malware outbreaks and assess the efficiency of response procedures.

Behavioral Analysis: Users can analyze the behavior of different malware samples in a controlled environment to understand their propagation, persistence, and potential damage.

Integration with Security Solutions: It can be integrated with existing cyber security solutions and tools to assess their effectiveness in detecting, preventing, or mitigating malware attacks.

Customization: Users can customize the tool to replicate their specific network infrastructure, applications, and systems, making it highly adaptable to different environments.

Reporting and Analysis: The tool provides comprehensive reports and analysis of simulated attacks, helping users gain insights into potential weaknesses and areas for improvement.

Compliance Testing: It can assist organizations in testing their compliance with industry standards and regulations related to cyber security and data protection.

Benefits:

- Improved cyber security readiness by identifying weaknesses and vulnerabilities before real-world attacks occur.
- Enhanced incident response capabilities through realistic training and simulations.
- Reduced security risks and potential damage from malware attacks.

International Journal of Multidisciplinary Engineering in Current Research
ISSN: 2456-4265, Volume 7, Issue 2, February 2022, http://ijmec.com/

- Informed decision-making for cyber security strategy and resource allocation.
- Compliance with industry standards and regulations.

2. RELATED WORK

Related work in the field of "Malware Attack Modeling and Simulation" typically involves research, tools, and methodologies that address various aspects of malware, their behavior, and the simulation of cyber threats for testing and research purposes. Here are some areas of related work in this field:

Malware Analysis Tools: There are various tools designed for analyzing and dissecting malware, such as IDA Pro, OllyDbg, and Ghidra. These tools help security researchers understand how malware operates and can be used as a basis for developing simulations.

Network Simulation Tools: Tools like Wireshark and tcpdump are used to captureand analyze network traffic. Researchers can use these tools to understand how malware communicates and spreads within a network, which can inform the design of simulation models.

Honeypots and Honeynets: Honeypots and honeynets are used to attract and study malicious activity. They can be integrated into simulations to create realisticattackscenarios and test the effectiveness of security measures.

Cyber Range Platforms: Cyber ranges provide controlled environments for simulating cyber attacks and testing defenses. These platforms often include malware simulation components to mimic real-world cyber threats.

Threat Intelligence and Indicators of Compromise (IoC) Feeds: These sources provide data on known malware and their behavior. Researchers can use this information to inform the design of realistic malware simulations.

Machine Learning and AI for Malware Detection: Researchers often employ machine learning and AI algorithms to detect and model malware behavior. These models can be used in simulations to mimic malware actions.

Adversarial Emulation Tools: Tools like Metasploit and Cobalt Strike are used for adversarial emulation and penetration testing. While not simulations in the traditional sense, they can be used to replicate the behavior of real-world attackers and their malware.

Cyber security Training and CTFs: Capture The Flag (CTF) competitions often involve simulating various cyber security challenges, including malware-related scenarios. These platforms help individuals learn about and practice defending against malware attacks.

Academic Research on Malware Behavior: Numerous academic studies focus on understanding and modeling the behavior of specific malware strains. This research can provide insights for simulation development.

Open-Source Malware Simulation Projects: Some open-source projects are specifically designed to simulate malware attacks and assess security measures. These projects may be used as references or integrated into other simulation tools.

When conducting related work in malware attack modeling and simulation, it's important to stay up-to-date with the latest research and tools in the field, as the cyber security landscape is constantly evolving.

Table I: Comparison of cyber-attack simulators.

Cyber Attack simulators	Language used	Simulator used	Number of scenarios	Number of nodes that can be modelled	Network type used
Igor Kotenko [8]	C++	OMNET++	N/A	1000	General
Park et al. [10]	Visual C ++	SECUSIM	20	1000	General
Kotenko and Man'kov [7]	Visual C ++	MASDK	N/A	1000	General
Kuhl et al.[5]	Java	Arena	37	1500	Enterprise
Dennis Lee Bergin [9]	Java	QualNet	6	1000	Mobile
DEVS-CAS	Java	DEVS-Suite	6	3500	Enterprise

2. NETWORK ARCHITECTURE AND DEVS-SUITE SIMULATIONENVIRONMENT

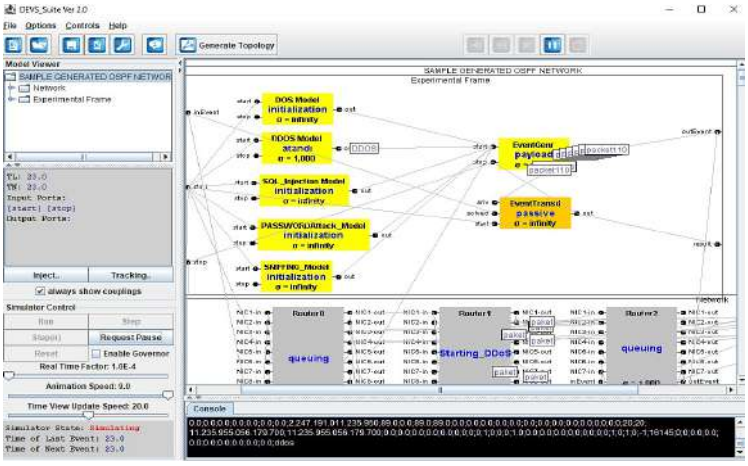
Simulation modelling of the general structure of a network, network traffic and cyber-attacks with an object-oriented programming approach provides great convenience for developers due to its ability to reuse object classes and easy development. Since the objects are modular, it is easy to add and reuse the codes in another project [22]. With a simulation where each function is abstracted as objects, calculations and modelling functions are split between objects, providing a more organized structure

In this study, a network simulation tool prepared using the

International Journal of Multidisciplinary Engineering in Current Research
ISSN: 2456-4265, Volume 7, Issue 2, February 2022, http://ijmec.com/

DEVS approach was used to perform cyber-attacks. Supporting a hierarchical / modular structure and distributed operation of the DEVS (Discrete Event System Definition) approach provides convenience in modelling complex large-scale systems (consisting of atomic and coupled models). In the parallel and distributed simulation algorithm developed using the DEVS modelling approach, parallelism is provided by using the parallel DEVS atomic and coupled model definition, while the distributed approach is provided with client server-based architecture and this algorithm is used in the development of a DEVS-based network simulation tool [23].

The Discrete Event System Specification (DEVS) formalism/approach is a means of describing a mathematical object called a system. The DEVS approach was first introduced by Dr. Zeigler in his book 'Theory of Modelling and Simulation' in 1976 for the modelling and analysis of discrete event systems [24]. DEVS is a discrete



event-based, modular and hierarchical simulation approach, and it has recently become more prominent than other approaches [25].

Computer networks need to be modelled because it is both risky and troublesome to experiment on these systems for different purposes due to the size of computer networks, the difficulty of management and high installation costs. Modelling represents a real system. On the other hand, computer modelling is the making of an existing system in a computer environment through a computer. Thus, any desired work on the existing system will be possible without disturbing the system.

A network has many entry points. These entry points include the network hardware and software that make up the network, in addition to devices that can be considered as gateways to the network. It is imperative to consider these entry points to defend the network. For this purpose, security devices are needed to monitor network traffic and prevent unauthorized access to the network. With these devices, the network should be divided into different levels and external threats should be minimized as much as possible. Objects are used to represent devices that attackers are trying to exploit in the simulation environment. These devices used in the network can have many features according to their tasks. Not all of these features need to be modelled. Key features related to network and cyber-attack alerts are taken into account whencreating the model.

There are numerous software implementations of the DEVS approach. DEVS-Suite and DEVSJAVA are object-oriented implementations of parallel DEVS and its associated technologies [24]. Using the advanced features of the Java programming language and object- oriented programming techniques, it displays the behaviour of complex systems and network systems using the DEVS approach. DEVSJAVA is a modelling and simulation environment consisting entirely of Java classes and packages, using the DEVS approach, which enables the modular design and reuse of nodes, software assets and experimental frameworks that form a network with its object-oriented structure. DEVS-Suite is a general modelling and simulation tool developed with the Java programming language and is a new version of the DEVSJAVA simulation tool as seen in Fig. 1. The fact that it is designed with the Java programming language lies in the background of many features that make the DEVS-Suite simulator stand out from other tools. In this study, we integrated the BRITE [26] topology generation tool into the application

International Journal of Multidisciplinary Engineering in Current Research
ISSN: 2456-4265, Volume 7, Issue 2, February 2022, http://ijmec.com/

Figure 1: Network framework and attack models interface in DEVS-Suite environment.

4.MODELLING THE ATTACKS

The development of an attack simulation model is prepared by going through certain stages as shown in Fig. 2. In the first stage, a network structure on which attack models will be run should be modelled. For the required network structure, a network topology should be created or a topology generator should be used. Another step is to develop attack models according to their own characteristics and to design appropriate interfaces where configuration settings of these attack scenarios can be made. The other phases are the development of the simulation

and monitoring framework in which the effects and results of the attack steps are observed and evaluated.

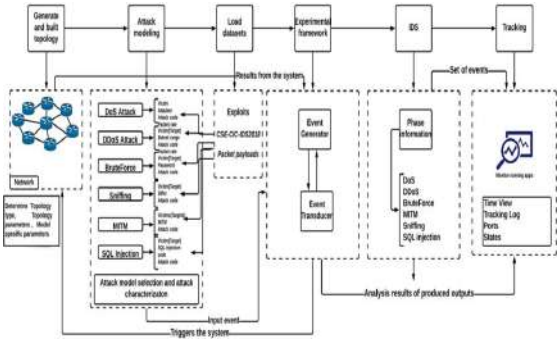


Figure 2: Cyber-attack development environment and components.

The DEVS-Suite cyber-attack application is built on top of the DEVS-Suite core. High level performance, scalability, theoretical system design and ease of use are provided by using DEVS formalism and advanced software engineering techniques. Events handled by nodes and links can be described as states charts, as shown in Fig. 3. To visualize the behaviour of simulation models, it is necessary to show state changes in response to internal and external events. A node atomic model occurs with an "initial" stage and does not have any information for other model components. After each node sends a hello message to its neighbours, it starts creating tables and learns what's going on in the network. Events in the system are selected according to the selected attack type. A sufficient number of cases are used to understand the attack logic. Increasing the number of events reduces performance and improves accuracy. Only external and internal transition functions can cause a node to change its context for new events.

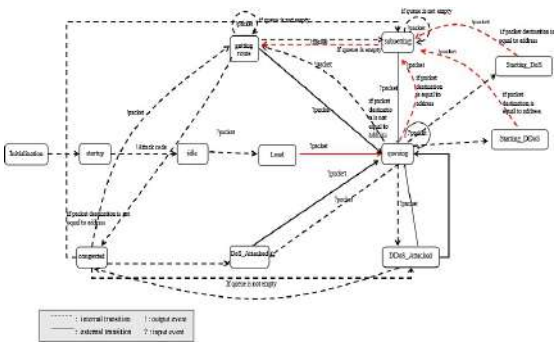


Figure 3: Target node states, state transitions.

After the network is modelled, it is necessary to create an attack scenario suitable for the attack purpose. Simulation models include user-defined methods of creating cyber-attacks. Each model network processes only one scenario at a time, even if there are many attack scenarios specifically defined for that network. The DEVS-Suite provides tools for identifying and simulating detailed attacks in the application.

There are many attack mechanisms that the attacker can use. The target entity generates different warnings against each attack, depending on its intrusion detection scheme. As a result, besides observing the interactions between an attacker and the target, a simulation model is required that can represent the characteristics of various attack mechanisms and targets.

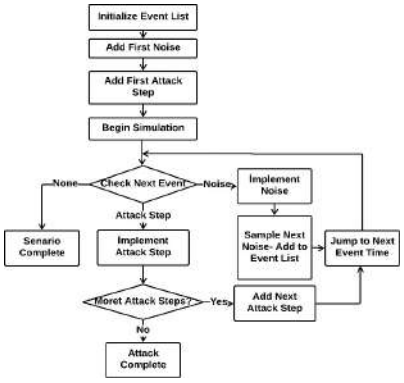
There is no need to model all network traffic, which represents all packets carried between devices, as in a physical network, so that the simulation performance is not degraded when modelling network traffic. Therefore, models mostly include network traffic involved in attack progress or intrusion detection processes. The DEVS network model is developed according to the network OSI standard with several abstractions. Since application level based upon the protocol implementation are in focus, first abstraction is to flatten seven segment OSI layers to three layers. These layers are data link, routing, and application. Another presumption is about socket representation in which only IPv4 implementation is modelled together with port name rather port number. Furthermore, very generic DDoS attack is implemented, various contemporary versions are ignored.

5.SIMULATED ATTACK TYPES AND METHODS

In this study, commonly used cyber-attack types are used to perform cyber-attacks against the virtual large-scale

International Journal of Multidisciplinary Engineering in Current Research
ISSN: 2456-4265, Volume 7, Issue 2, February 2022, http://ijmec.com/

network system configured and topologically designed under the DEVS- Suite. In this context, attack models have been integrated into the DEVS-based distributed large-scale network simulation model as seen in Fig. 4 a. These models are: DoS, DDoS, BruteForce, SQL injection, Man in the Middle and Sniffing are attack models. The developed attack simulator is configured to provide an infrastructure that can simulate many types of attacks. Simulating more attacks cause to take action against future threats. In this case, the probability of detecting attacks in a shorter time arises [27]. In this article, the attack methods and simulation stages of DoS and DDoS attack types are explained in detail.

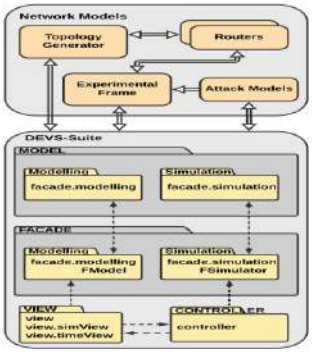


a)

b)

Figure 4: a) conceptual models and modelling methodology, b) DEVS-based attack scenario

The simulator uses DEVS method to execute the attack scenarios. Fig. 4 b presents the modelling methodology used to manage the simulation of an attack scenario. In order to test the model developed in the software environment, the experimental framework concept in DEVS-Suite should be created. Experimental frameworks are used in DEVS-based simulations to drive scenarios by injecting inputs and interpreting outputs. This design traditionally requires generator, receiver, and converter models with different roles. In certain controlled experiments, such as model testing, sequential programming offers a simpler design that has many benefits, especially code reduction, test case development output, and diagnostics for failed tests. This research presents a testing framework derived from atomic DEVS that facilitates testing through scripting [28]. The experimental framework consists of two main components with several attack models as seen in Fig. 1:



1 – Event Generator: It is a generator connected to the input terminals of the system to give a trigger signal to the system.

2 – Event Transducer: It is a transducer that is connected to the output ends of the model in order to evaluate the results coming from the system model. The event transducer is a tool used in the evaluation and analysis of the results of the simulation study.

DoS attack

In this study, after defining the network components and starting the network simulation at different scales with a topology generator, if the DoS model is selected as the attack model in the control interface, the DoS attack configuration window is opened. In the form where the DoS attack settings are made, the IP numbers of the victim computer and the computer that will perform the attack, the attack code and the number of packets to be sent in each step are set. With this data, the simulated DoS attack model is triggered. In the event generator atomic model in the experimental framework, events can be generated automatically or manually to input ports. An input event includes a port name, data value (packet), and elapsed time. Elapsed time is a timestamp of the associated event and is used to plan and inject a specific event. Elapsed time is provided in units of time associated with the simulator clock. In this study, the packages that make up the data value were prepared using the CSE-CIC-IDS2018 dataset shared by the Canadian Cyber Security Institute [29]. The CSE-CIC-IDS2018 dataset, which was prepared in a suitable test environment, was created by taking into account the deficiencies in the previously used datasets. This dataset has been produced by considering threat structures and safe behaviour traffic. The dataset was produced in PCAP file format and converted to CSV format. This data shared by the Canadian Cyber Security Institute has been made available to the public and researchers who want to work in the field of cyber security can access the PCAP and CSV format of this data

International Journal of Multidisciplinary Engineering in Current Research
ISSN: 2456-4265, Volume 7, Issue 2, February 2022, http://ijmec.com/

set.

Label	Number of samples
Benign	6000000
Bot	290000
BruteForce-Web	612
BruteForce-XSS	231
DDoS	690000
DoS attacks-Slowloris	11000
DoS attacks-Goldeneye	41500
DoS attacks-Hulk	462000
DoS attacks-SlowHTTPTests	140000
FTP BruteForce	196000
Infileration	61000
SQL Injection	90
SSH BruteForce	188000

Table II: Detail of the dataset.

In this study, the CSV format of this data set was used. The vectors in this dataset contain 79 features and different features can be selected and used according to the attack type. Depending on the type of attack, the properties of these samples can be pre-processed and transformed, if necessary, and different data can be obtained. In Table II, the sample numbers of the labels in the data set are seen.

In the DoS attack simulation, the input event is automatically generated by connecting the outputs of theDoS attack model to the input port of the event generator atomic model. At each step, packets whose target is the victim computer and whose number is set at the beginning of the attack are sent from the output ports of the attacking device.

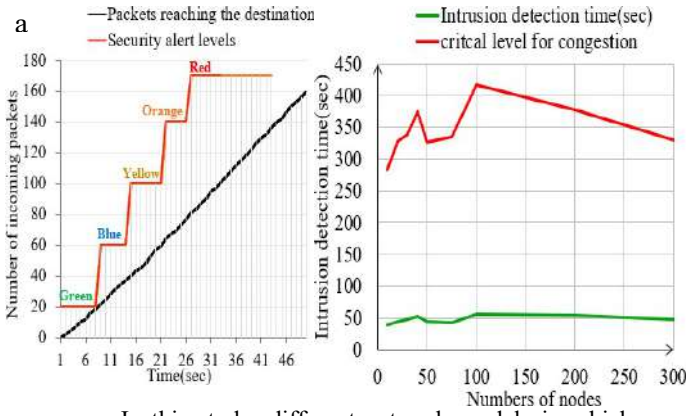
Figure 5: a) security alert levels, b) intrusion detection time.

Packet traffic initiated intensively by the attacker towards the victim device reaches the destination in different ways according to the routing tables and routing algorithms in the network. During the simulation, congestion may occur at some router nodes on the route due to the density of packet traffic. After the packets arriving at the input ports of the target computer selected as the victim reach a certain number, the computer switches to DoS-Attacked state and the status of the victim is indicated with a warning message by specifying the IP number of the victim in the console window where the DoS attack against the related device is made.

When the first DoS attack warning is made, it does not mean that the device is completely out of service. During the attack, this warning repeats periodically depending on the number of packets reaching the target. If the attack is not stopped after a while, a complete blockage will occur and the red level status will be entered and the service will be blocked then the attack achieves its purpose.

Packets sent from a certain source do not cause any abnormality up to a certain number. This situation is shown as the green level in Fig. 5 a. It is configured as 20 packets in 10 seconds with simulation time for acceptable level as normal network traffic in atomic model configuration. After this level is exceeded, it is considered as an abnormal situation and a DoS attack warning alarm is given according to this abnormal situation.

Different security risk levels are determined in Fig. 5 a according to the number of packages increasing over time. The red level, where the number of packets reaching the target in the specified unit time is 160, indicates the level of congestion. The colour of the target device in the simulation environment also changes according to the colours in the graphic. This makes it easy for the observer to notice the danger. Depending on the number of nodes in the network, the upper and lower levels of the warning alarm times are shown in Fig. 5 b. It is understood from the graph that intrusion detection times are not affected by the number of nodes.



In this study, different network models in which the attack simulation will be carried out are produced with a topology generator in order to test the DoS attack

International Journal of Multidisciplinary Engineering in Current Research
ISSN: 2456-4265, Volume 7, Issue 2, February 2022, http://ijmec.com/

simulation on networks of different sizes. With the topology generator integrated into the attack simulator, networks of different sizes can be simulated, from small to very large-scale networks. Simulation of large- scale networks uses high CPU and memory resources

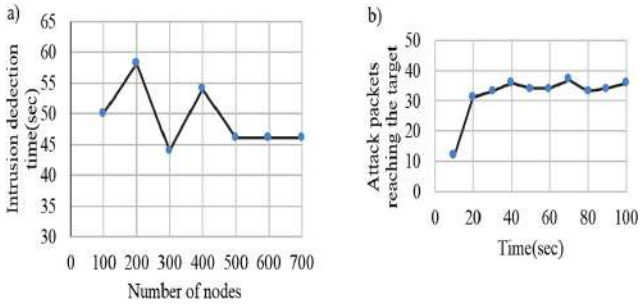


Figure 6: a) intrusion detection time graphs based on network size, b) the instant packet graph reachingthe target.

After the DoS attack started, the number of packets detected from the attacker source to the target node in fixed time slots are shown in Fig. 6 b. Intrusion detection times in DoS attack performed on networks with different number of router nodes are shown in Fig. 6 a. In a DoS attack, attack detection is calculated in proportion to the number of abnormal packets reaching the target in a given time frame. The heavy network traffic generated causes congestion on some routes, in which case the packets are diverted to different routes to reach the destination. Packet losses occur due to the queue overflow in the nodes, which leads to a decrease in the number of attack packets reaching the target in unit time and an increase in the attack alarm time. As the number of routers in the network increases, alternative routes to the target also increase and the time of attack packets to reach the target becomes balanced. This causes the attack alarm times to take close values in DoS attacks made from a single source in large-scalenetworks.

6.CONCLUSION AND FUTURE WORK

Realistic Threat Simulation: It enables the creation of realistic malware attack scenarios, which helps security teams understand the nature of threats they might face in the real world. This realistic threat simulation is essential for preparedness.

Evaluating Defenses:
By simulating various types of malware attacks, the tool helps in evaluating the effectiveness of existing security measures. This evaluation is crucial for identifying vulnerabilities and making improvements.

Training and Skill Development: It can be used for training security personnel, allowing them to practice responding to different malware incidents. This training helps build expertise and readiness.

Policy Development: The tool provides insights into potential weaknesses in security policies and procedures. It assists organizations in refining their cybersecurity policies to be more resilient against malware attacks.

Cost-Efficient Testing:

Malware Attack Modeling and Simulation Tool provides a cost-effective means to test and refine security strategies, reducing the likelihood of actual incidents and their associated costs.

Customization:

Users can tailor the simulations to specific malware types and scenarios, allowing for a focused and precise assessment of their security posture.

In conclusion, the "Malware Attack Modeling and Simulation Tool" is a crucial asset in the ongoing battle against cyber threats, offering a proactive and controlled environment for testing and strengthening cybersecurity defenses, ultimately enhancing an organization's resilience to malware attacks.

REFERENCES

[1] Anderson, R., & Mohay, G. (2007). Computer Security Threats Modeling and Simulation. In Proceedings of the 2007 Winter Simulation Conference (pp. 1562-1570). IEEE. - This paper discusses the modeling and simulation of computer security threats, which includes malware attacks.

[2] Dacier, M., & Deswarte, Y. (2006). Early Warning and Attack Tracing for Fast Worms. IEEE Security & Privacy, 4(1), 20-26. - This paper discusses techniques for modeling and simulating fast-spreading malware attacks, particularly worms.

[3] Toulouse, J. (2008). Malware simulation: Improving detection tools. In International Conference on Malicious and Unwanted Software (pp. 73-76). IEEE. - This conference paper discusses the use of malware simulation to enhance malware

detection tools.

[4] Sommer, R., & Paxson, V. (2010). Outside the closed world: On using machine learning for network intrusion detection. In Proceedings of the 2010 IEEE symposium on security and privacy (pp. 305-316). IEEE. - This paper explores intrusion detection systems, which often involve simulating and modeling various types of cyberattacks, including malware.

[5] Anderson, R., & Janicke, H. (2011). Advances in computer and information sciences and engineering. CRC Press. - This book discusses various aspects of computer security, including modeling and simulation of malware attacks.

[6] The MITRE Corporation. (n.d.). ATT&CK®: Adversarial Tactics, Techniques, and Common Knowledge. https://attack.mitre.org/ - MITRE's ATT&CK framework provides detailed information on adversary tactics and techniques, which can be useful for modeling and simulating malware attacks.

[7] Kotenko, I.; Ulanov, A. (2005). Agent-based simulation of DDOS attacks and defense mechanisms, *Journal of Computing*, Vol. 4, No. 2, 16-37

[8] OpenAI. (n.d.). GPT-3 (ChatGPT). https://openai.com/research/chatgpt - While not specific to malware, GPT-3 models like ChatGPT can be used to generate simulated content or assist in discussions related to malware attack scenarios.

[9] Park, J. S.; Lee, J. S.; Kim, H. K.; Jeong, J. R.; Yeom, D. B.; Chi, S. D. (2001). SECUSIM: A tool for the cyber-attack simulation, Qing, S.; Okamoto, T.; Zhou, J. (Eds.), *Information and Communications Security, Lecture Notes in Computer Science*, Springer, Berlin, 471-475, doi:10.1007/3-540-45600-7_53