Agent-based Network Security Simulator Nessi2

Yuntian Zhao, Youjun Wang, Hongqi Zhang, Chuanfu Zhang, Chao Yang

Zhengzhou Information Science and Technology Institute, Zhengzhou 450000, China;

manyer@yeah.net, daxia1wang@163.com

Keywords: Nessi2; distributed simulation; agent technology; JIAC

Abstract. With the development of the Internet, a variety of attack behaviors are emerging, which seriously threaten the security of cyberspace. In order to effectively defend these attacks, the researchers put forward a variety of research methods. During these methods, using simulation software to model the attack behavior and run simulation is one of the effective means. In this paper, it introduces the framework of Nessi2 simulation software, and analyzes the main functions of the software from two aspects of distributed simulation and security simulation.

Introduction

With the popularization of the Internet and the increase in network complexity, network security is becoming more and more seriously. It will cause a serious loss of property as long as a worm, Trojan or virus occurs every time. In order to study the behavior of these attacks through the method of modeling and simulation, researchers have developed a variety of simulation software. But among the existing simulation software, some of them are only simulate the network initial, which is expanded to simulate attack behavior. The others were built for the specific attack entity. They can be used for security simulation and analysis, but also have limitations: 1) they only can be used to model a specific attack, cannot simulator others; 2) the purpose of the simulation is to provide effective defense measures, but these software cannot be used to evaluate the effectiveness of the defensive measures. So the network security researchers developed a new network security simulation software —Nessi2 (Network Security Simulator).

Nessi2 Architecture

Nessi2 is developed and designed by DAI-Labor laboratory in Berlin, it is a network simulator that support distributed simulation. Nessi2 is a packet-level discrete-event simulator. Distinguished other network simulators, Nessi2 incorporates a variety of features which are relate to network security, such as profile-based automated attack generation, traffic analysis and detection algorithm plug-ins , these functions allow this software that can be used for security research and evaluation.

In order to resolve the complexity, Nessi2 is divided into three separate parts, the Graphical User Interface, the simulation backend and the result database 0. It shows in Figure 1. Each of these parts can be run on separate machines depending on the computational requirements.

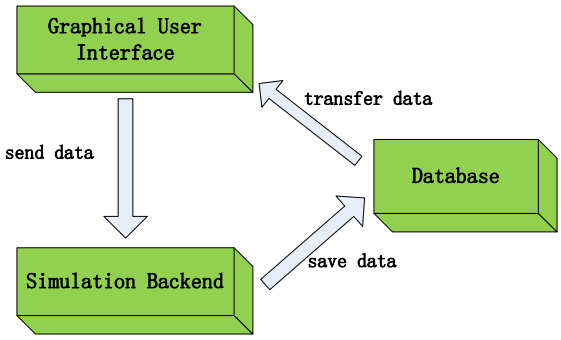

Fig. 1 Nessi2 Architecture

Graphical User Interface (GUI) is consisting of two perspectives, Network Editor Perspective and Network Simulation Perspective. Network Editor Perspective is used to establish the network topology, configuration network nodes, scenario scheduling and so on. Network Simulation Perspective can show the simulation result and the result data is stored in database.

Simulation backend is the simulation core of Nessi2. It is used to run simulation and store simulation data into MySQL. NeSSi2 is built upon the JIAC (Java Intelligent Agent Component-ware) framework.

MySQL database can store network topology data, node configuration information and simulation result.

Nessi2 Simulation Framework

Nessi2 is developed on Eclipse, Maven is used to build artifacts that support the extending of Nessi2 features. It can extend the software without changing the Nessi2 simulation core.

Figure 2 provides a conceptual view on NeSSi2 the Network Security Simulator. The bottom layer shows some of the important technologies that NeSSi2 uses. This includes the JIAC agent framework used for representation of the different entities in the simulation, EMF for the network data model and Eclipse RCP as the platform-independent, plugin-based execution environment. On top of the bottom layer, several concepts realized in NeSSi2 can be found.0

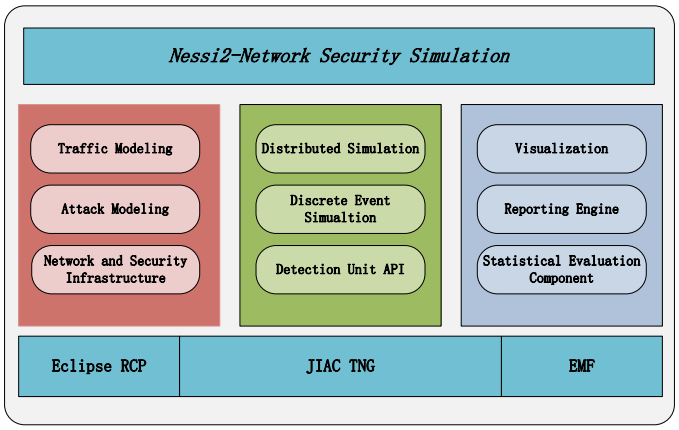

Fig. 2 Conceptual view on NeSSi2

As Figure 2 shows, Nessi2 supports distributed simulation, and can simulate in security field, such as attack modeling (DDOS, worm propagation model) and Detection Unit API to implement a combination of intrusion detection and assessment.

Nessi2 Simulation Framework

Distribute Simulation. During large-scale simulation, Nessi2 uses distributed simulation method to reduce the simulation time and memory consumption. Distributed Simulation separates the task into several sub-tasks and assigns them to some nodes. One of these nodes controls the distribution of sub-tasks.

Nessi2 uses the agent technology to achieve the distributed simulation. The subdivision of task can run on different computers and process in a parallel-execution model. Network entities (routers, clients, servers) are simulated with the aid of agents. According to parameter configuration and hardware characteristics, each agent can simulate one or more nodes. Different agents can run different tasks at the same time.

Simulation nodes are divided into two categories, Master Simulation Nodes and Slave Simulation Nodes (Fig 3). Master Simulation Nodes manage and assign tasks to Slave Simulation Nodes, Slave Simulation Nodes run sub-tasks, and return the simulation result.

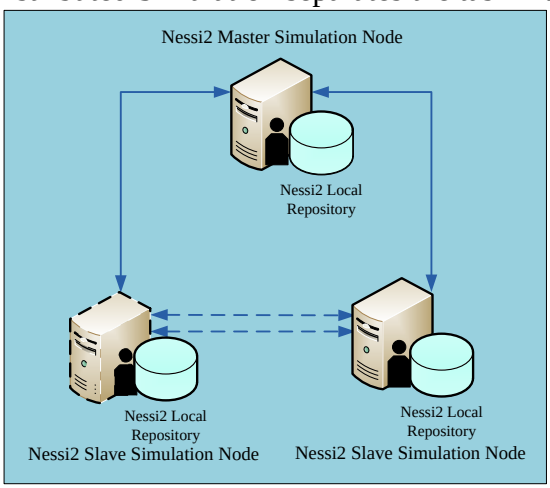

Fig. 3 Distributed Simulation Architecture

In simulation backend, Nessi2 has three agent roles, Subnet Agent（SA）, Platform Coordination Agent (PCA), Network Coordination Agent (NCA). One SA is used to manage an individual subnet, and receive events from the PCA. Once all events have been handled, the SA reports the result to the PCA. One PCA is used to create, configure and control the Subnet Agents in the same platform, the number of Subnet Agents is dependent on the available computing resources. PCA receives events

from NCA and returns the data to NCA after tasks have been accomplished. NCA receives events from GUI, and controls the entire network simulation. The NCA coordinates the distribution of the subnets at the beginning of a simulation.0

Security Simulation. Nessi2 provides different attack models, such as worm propagation model. During worm propagation, packets are sent by nodes have been simplified, the unnecessary portions of packet are removed, such as: survival time, checksum and payload. The simplified packet only contains the destination address, source address and packet size. Thereby, it has lowered the memory consumption.

The SQL Slammer worm and Blaster worm are set as the references to establish a model similar to the SIR. It defines the three states of nodes: Susceptible, Infective and Removal. SIR worm propagation model is a statistical model, which to analyze the speed and range of worm propagation by counting the number of susceptible hosts, infective hosts. Before the simulation started, the initial number of infected hosts should be set, and defined the worm type of initiative or not initiative (at least one is an active type).

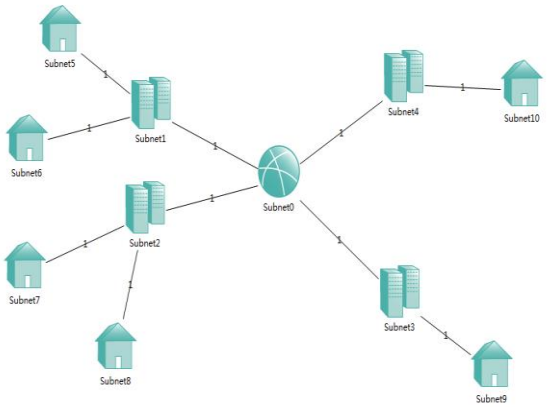

Fig. 4 Network Topology

In this experiment, since the limitation of creating network topology manually, it is difficult to create a large-scale network, so we use a small network as the test environment. Figure 4 is the network topology. In order to facilitate the distributed simulation, Nessi2 separates network into multiple subnets. In each subnet, it also can set up network topology. As shown in Figure5, it is the topology of Subnet 4. The existing applications in network are: Client, Server, Firewall, Router and so on. Bot is the host that contains attack applications, such as worm application and DDOS application.

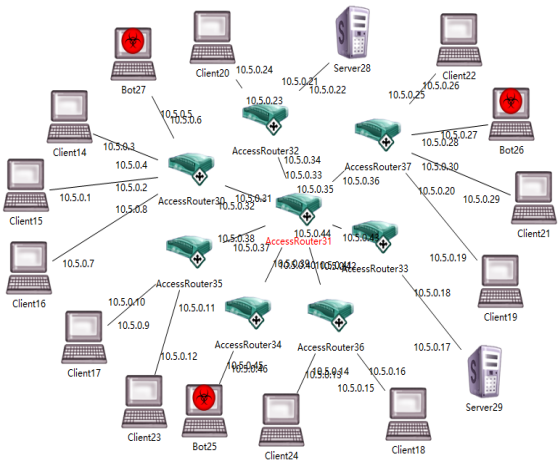

Fig. 5 Topology of Subnet4

After creating network topology, we need to apply roles to nodes. The application in these roles is defined by .profile file. This file not only can define the action of the node, it also can define the security application. The creation of the script is to map application to the network node. Nessi2 supports different application event running on the same network environment without affecting each other by creating different scripts. After the script generation, we set the simulation parameters, such as: simulation start and end conditions, and the database to store the results of the simulation data, the number of simulation repetitions and so on.

In Network Simulation Perspective, it can take the result data from database, and show in the Statistics window. In this window, we can see the behavior of each node and link. Figure 6 is the traffic of Subnet 4. Figure 7 is the simulation statistical results of Bot 27 in Subnet 4. In these two figures we can see the traffic sent by bot is significantly more than the other nodes and the packet sent and received of Bot 27during simulation.

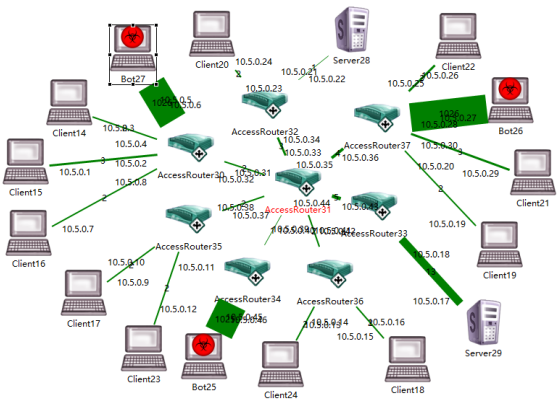

Fig. 6 Traffic of Subnet 4

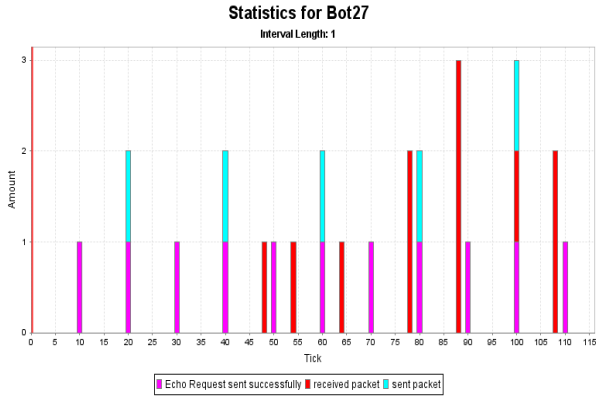

Fig. 7 Statistical results of Bot 27

Nessi2 Advantages

1) Scalability. The user can use JAVA language to write application, and use Maven to define and adds the dependencies of the application. Therefore, Nessi2 can add functionality or application needed.
2) Fidelity. Nessi2 creates a network topology based on network characteristics, it also can automatically generate a network topology depended on the download file, it has high fidelity. (It can download the AS topology file from CAIDA website, then import into Nessi2, automatically generated network topology).
3) Extensibility.Nessi2 can work with third-party software, such as Wireshark.

Summary

Nessi2 is a simulator that can do security simulation and distributed simulation. Nessi2 currently enables applications include: DDOS attacks, worm propagation, and distributed collaborative intrusion detection system testing. But the design of the distributed intrusion detection system is not completed, so we will continue to follow this research work.

References

[1] Dennis Grunewald Marco Lützenberger Joël Chinnow. Agent-based Network Security Simulation Proc. of 10th Int. Conf. on Autonomous. Agents and Multiagent Systems (AAMAS 2011), Tumer, Yolum, Sonenberg and Stone (eds.), May, 2–6, 2011, Taipei, Taiwan, pp.1325-1326.

[2] Stephan Schmidt, Rainer Bye, Joël Chinnow. Application-level simulation for network security . SIMUTools (Revised and extended version) March 03 - 07, 2008, Marseille, France.

[3] Joel Chinnow, Rainer Bye, Ahmet Camtepe, Karsten Bsufka, Sahin Albayrak. Evaluation of Attacks and Countermeasures in Large Scale Networks. Informatik schafft Communities 41. Jahrestagung der Gesellschaft für Informatik , 4.-7.10.2011, Berlin.

[4] Alessio Lomuscio, Paul Scerri, Ana Bazzan, and Michael Huhns (eds.). Engineering JIAC Multi-Agent Systems. Proceedings of the 13th International Conference on Autonomous Agents and Multi-agent Systems (AAMAS 2014), May 5-9, 2014, Paris, France.

[5] Marco Lützenberger Tobias Küster Thomas Konnerth Alexander Thiele Nils Masuch. JIAC V — A MAS Framework for Industrial Applications. Proceedings of the 12th International Confer-ence on Autonomous Agents and Multiagent Systems (AA-MAS 2013), Ito, Jonker, Gini, and Shehory (eds.), May, 6–10, 2013, Saint Paul, Minnesota, USA.