Architecture of Network Environment for High-Risk Security Experimentation

Xiaohui Kuang, Xiang Li, Jinjing Zhao

Architecture of Network Environment for High-risk Security Experimentation

Xiaohui Kuang [1,2] Xiang Li [1,2] Jinjing Zhao [1,2]

[1] National Key Laboratory of Science and Technology on Information System Security, Beijing
[2] Beijing Institute of System Engineering, China

Xiaohui_kuang@163.com lixiang8358@hotmail.com
misszhaojinjing@hotmail.com

Abstract. Adequate Environment for conducting security experiments and test under controlled, safe, repeatable and as-realistic-as-possible conditions, are a key element for the research and development of adequate security solutions and the training of security personnel and researchers. In this paper, a new large-scale network experimental environment for high-risk security research was put forward. The main idea was using isolated computing clusters to obtain high levels of scale, manageability and safety by heavily leveraging virtualization technology, separating experiment and control network and multilayer sanitization.

Keywords: security experimental environment, architecture, virtualization, characteristic analysis

1 Introduction

Experimentation is a keystone of scientific method and of technology innovation and development. The success of Internet can be attributed to many factors. However, experimentation environment played an important role. Today more and more researchers involved in designing network services, protocols and security mechanism rely on results from emulation environments to evaluate correctness, performance and scalability [1][2][3][4][5][6][7].

To better understand the behavior of these applications and to predict their performance when deployed across the Internet, the emulation environments must closely match real network characteristics. But, these emulation environments pay attention to scalability, fidelity and flexibility, don't take into account about risk and sanitization. The high-risk experiments such as network worms, botnets, and viruses and so on, are difficult to conduct in these network emulation environments.

In this paper we describe a new architecture of network environment for conducting high-risk security experimentation. The key idea behind our environment is the use of isolated, special purpose computing clusters, which heavily use virtualization technology, physical and logical separation, and cluster management tools in order to

adfa, p. 1, 2011.

attain a high level of scalability, flexibility, isolation, and sanitization. The main contribution of this paper is 1) to analyze the requirement of high-risk security experiment, 2) to describe the architecture of a large-scale network environment that match the requirement, 3) to analysis the characteristic of the environment, 4) to describe the test on the environment.

The rest of the paper is organized as follows: Section II presents the requirement of security research facilities should match. Section III describes the architecture of our experimental environment named LNESET (Large-scale Network Emulation environment for high-risk Security ExperimenTation), and the technical details of the actual testbed. We then give the analysis in Section IV. Finally, our conclusions are in Section V.

2 Requirements for Security ExperimenTation environment

2.1 Risk Analysis and Risk Management

Depending on the kinds of research activities performed, security research can involve risks associated to the existing information infrastructure and ultimately to the public at large. From a confidentiality point of view, it is important to protect data from software vulnerabilities and malware collections from use for nefarious purposes. In addition, certain data collections that are used in security research, such as network traffic traces, could potentially contain private information. Similarly, details about network architecture or configuration of defensive mechanisms from real networks that are being emulated in a testbed facility should be protected from unauthorized access, as knowledge of these details could jeopardize their security. Most importantly, experiments involving live malware could "spill" into a real computing environment, affecting it from an integrity and availability point of view, for example, in the case of accidental releases of malware samples on previously uninfected machines. Alternatively, the effects of such actions could be indirect, such as those caused by researchers interacting with infected and criminal-controlled systems. In that case, this action could potentially trigger a premature and unnecessary arms race between the criminals and security researchers and security product vendors, by alerting cybercriminals that someone is "onto them", or on weaknesses in their tools and approaches.

2.2 Key Requirements Analysis

The construction of network experimental environment can be essentially conducted in four fashions: mathematical modeling, stochastic simulation, in laboratory emulations, or test-bed. The fidelity and scalability of the experimental environments built with different fashion are significantly different, and the laboratory emulation has often been the preferred method in security research, especially with regards to malware analysis.

Laboratory experimentation offers an interesting compromise. On the one hand, the controlled conditions in which they are ideally conducted provide 1) the ability to

validate previous experimental results obtained by others, i.e. repeatability, and 2) the ability to vary the parameters and characteristics of security solutions or threats being studied, i.e. experimental control. On the other hand, the use of the same or similar pieces of hardware (whether real or emulated) and the same software that is present in the real world, whether offensive or defensive, adds a level of a priori realism that is hard to attain just by modeling and simulation. However, in order to conduct security experiment, the emulation environment must obey the following criteria:

Scale. The number of elements (machines, subnets, processes, etc.) that are being emulated or recreated in the experiment should be large enough to approach the numbers in the real world or at the very least large enough so that statistically significant results can be obtained.

Fidelity. The static conditions describing the environmental setup should be as close as possible to those of typical equivalent environments in the real world. This includes network topology, server configurations, proportion of machines and equipment in various roles, and security mechanism.

Isolation. Security experiments in emulation environment are often involving live malware, such as worm and virus, which could destroy the security of the other network, such as management network or data-collection network. So the environment should isolate the live malware from the other network.

Sanitization. Even if the live malware could be isolated in experimental network, it can destroy subsequent experiment conducted in the environment. So, the experimental environment should sanitize machines and network nodes quickly and entirely.

3 Architecture of security experimental environment

3.1 Main idea

According to the requirement of security experiments, the LNESET are built on clustering, virtualization, and software routing technology. The architecture of LNESET is detailed in Fig. 1, and is basically composed of three layers; they are physical infrastructure layer, Meta resource layer, and Experiment layer.

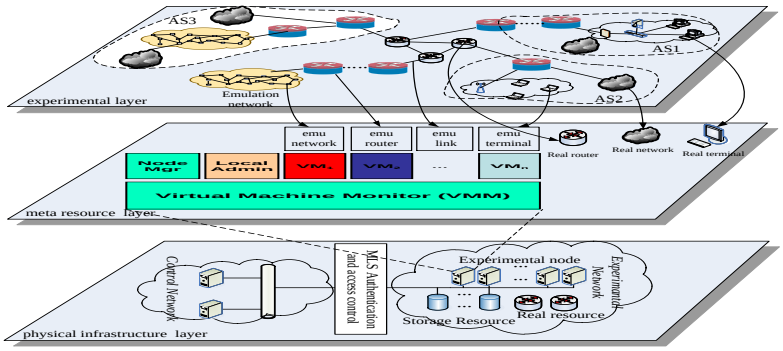

Fig. 1. The Architecture of Large-scale Network Emulation Environment for High-risk Security Experimentation

3.2 physical infrastructure layer

At the lowest layer is the physical infrastructure layer, which is composed of compute nodes, switches, and other physical resources that is similar to the Emulab suite. The physical infrastructure of LNESET consists of two components: the control network and experimental network.

The control network consists of a management server and the data server, the former is responsible for the centralized management, configuration and monitoring of the hardware and software resources in the experimental environment. Besides, it provides a user interface to the researchers to configure, run, monitor the environment, furthermore collect and analyze the experiment results. The data server is used to store a variety of virtual image files and experimental data.

The experimental network is made up of various types of experiment nodes which are interconnected with the programmable network. The programmable network is stacked with some high-speed Ethernet switches which support VLAN partitioning. The experiment nodes include computer cluster, real router, real terminal and real network.

The control network and experimental network are isolated by MLS authentication and access control mechanism, which is very import for security experiment. It allows only a set of predefined information to flow between the control network and experimental network, which is needed to manage, configure, and sanitize the experimental network. Any other information will be denied to translate from experimental network to control network.

3.3 Meta resource layer

The meta resource layer is composed of a subset of resources allocated from the physical resource layer as needed by the experiment. After the requested resources have been successfully allocated by the control network, LNESET will bootstrap each compute cluster node by running a customized OS image that supports VMs.

The meta resources include not only real terminal, router and network connectivity between them, but also emulation resources which are constructed by virtualization on computer cluster node. These emulation resources include emulation network, router, terminal and link. LNESET realize the emulation network by extending NS2[8], realize emulation router by extending Quagga[9], and realize emulation Link by extending TC in Linux kernel[10]. In order to improve the scalability of computer cluster, LNESET use Xen[11] and OpenVZ[12], which could emulate one hundred nodes on one computer.

Meta resources are managed by LNESET. LNESET translates the experiment specification into meta resource requirements and instructs to allocate the resources for the experiment. After experiment, LNESET could retrieve resources by management server. In order to clear the live malware, the computer cluster node will reboot remotely on initial stage of each experiment, which will format the local disk. After that, LNESET will reload different emulation image from data server according experimental requirement, which will emulate network, router, terminal or link.

3.4 Experiment layer

The experiment layer is created according to the network model of the experiment. Each compute cluster node serves as a basic scaling unit and implements the operation of emulation network, emulation terminal or emulation router that is mapped to it. A scaling unit for a network consists of a simulator instance and zero or more emulated hosts each running as a VM. An emulated host can also run directly on a physical compute node. This is designed for cases where the emulated hosts have stronger resource requirements and thus cannot be run as VMs. LNESET provides an emulation infrastructure to connect the emulated hosts running on virtual or physical machines to the corresponding simulator instances.

After the experiment starts, the experimenter can interact with the emulated hosts and the simulated network entities. To access the emulated hosts, the experimenter can directly log onto the corresponding computing nodes or VMs and execute commands (e.g., pinging any virtual hosts). To access the dynamic state of the simulated network, LNESET provides an online monitoring and control framework. This allows the experimenter to query and visualize the state of the network.

4 analysis of architecture

According to the previous architecture description in detail, LNESET can match the requirement of high-risk security experiment.

Scale. In LNESET, meta resources include emulation network, router, and terminal, which are realized by virtualization technology. Each computer cluster node can emulate network including one hundred node, such as router, terminal or server. So, LNESET could provide a large network environment for security experiment based on cluster, hence it has a better scalability.

Fidelity. In LNESET, except of real network, router and server, the emulation resources have high consistency with real resources. Based on virtualization technology, emulation resource could provide a realistic operating environment for testing real implementations of network protocol, applications and services. The routers, terminals and servers, and the connectivity between them could be configured according the experiment requirement. The network layer and system layer topology and network flow of environment could configure dynamically, so LNESET also provide a high level fidelity than test-bed based on simulation or analysis model.

Isolation. Because the MLS authentication and access control mechanism only allows very specific requests to proceed, the control network has very little interaction with the experimental network, so that it is not generally susceptible to attacks or malicious activities carried out by the security experiment such as worm or virus.

Sanitization. In LNESET, every resource will be initialized at the beginning of experiment. The experimental node include real node and cluster node will be sanitized entirely, because they reboot over network, and the boot files in server cannot rewrite. After reboot, the local disks of these experimental nodes will be formatted clearly. Every real router will be reset at the initial stage. So LNESET should sanitize all resources entirely.

5 Conclusion

Based on the in-depth analysis of the requirements of environment for high-risk experiment, in this paper we propose an emulation experimental environment named LNESET. We describe the architecture of LNESET in detail. LNESET supports high-level isolate between control network and experimental network, provides the necessary tools for the experimenters to allocate and reclaim the resources. The resources may consist of the compute nodes in the cluster and real machines or routers. Each compute node is treated as a scaling unit in LNESET; it can be configured as an emulation network with a set of VMs, a router or terminal for emulated hosts. The latter provides a realistic operating environment for testing real implementations of network applications and services. The VMs are connected with the simulator instances using a flexible emulation infrastructure. The analysis shows that LNESET is effective for high-risk security experimentation.

Acknowledgment

This research is supported by National Natural Science Foundation of China (Grant No. 61100223).

References

1. Steffen Maier, Daniel Herrscher, Kurt Rothermel. Experiences with node virtualization for scalable network emulation [J]. Computer Communications, 2007(30): 943-956.
2. PrimoGENI for hybrid network simulation and emulation experiments in GENI. 2012
3. Design and Evaluation of a Virtual Experimental Environment for Distributed Systems.
4. Rob Simmonds, Brian W. Unger. Towards Scalable Network Emulation[J]. Computer Communications, 2003, 26(3): 264-277.
5. Network Emulator with Virtual Host and Packet Diversion. 2012
6. Design and Implementation of a Simulation Environment for Network Virtualization.
7. www.planet-lab.org
8. http://nanam.isi.edu/nanam/index.php/Main_Page
9. http://www.nongnu.org/quagga
10. http://www.kernel.org
11. http://xen.org
12. http://wiki.openvz.org/Main_Page
13. Shuyuan Mary Ho and Hwajung Lee. A Thief among Us: The Use of Finite-State Machines to Dissect Insider Threat in Cloud Communications. Journal of Wireless Mobile Networks, Ubiquitous Computing, and Dependable Applications (JoWUA). Volume 3, Number 1/2 (March 2012).
14. Yoshiaki Hori, William Claycomb, and Kangbin Yim. Guest Editorial: Frontiers in Insider Threats and Data Leakage Prevention. Journal of Wireless Mobile Networks, Ubiquitous Computing, and Dependable Applications (JoWUA). Volume 3, Number 1/2 (March 2012).