Malware Analysis in the ToMaTo Testbed

Dennis Schwerdel, Bernd Reuther, Paul Mueller
Integrated Communication Systems Lab, University of Kaiserslautern, Germany
{schwerdel, reuther, pmueller}@informatik.uni-kl.de

I. Introduction

In the last years a lot of holistic research efforts investigate concepts and technologies for future networks. All of these research projects need ways to evaluate their ideas and results. Experimental facilities aim to provide a realistic environment for experiments using emulation techniques.

Distributed research projects often result in distributed research hardware like in the German-Lab project. Experimental facility software must be able to handle the restrictions and features of distributed resources.

Networking experiments often need increased control over the network environments. This includes configurable link characteristics and network topologies.

In the German-Lab project, the Topology Management Tool [1] (ToMaTo) has been developed as an experimental facility software to run networking experiments on.

II. ToMaTo

ToMaTo allows users to build networking topologies containing *devices* and *connectors*. Devices are active components like computers that run the software of the experiment and are the only sources and sinks of data. Connectors are network components that connect devices and transport their data exhibiting certain configurable characteristics.

Different types of virtual machines can be selected as devices each exposing other features and resource consumption. This diversity allows both lightweight virtual machines for running Linux, and full-featured machines for running any operating system including Linux and Windows.

Four types of connectors allow the users to select hubs, switches, routers and to connect the topology to external network adapters. Topologies that do not use external networks are completely isolated.

ToMaTo features an easy-to-use graphical user interface for creating and configuring topologies as well as for accessing the devices. The user interface is web-based and thus is cross-platform and can be used without software installation. ToMaTo also allows to capture network traffic on connections and analyze them using well-known tools like Wireshark.

III. Scenario

To demonstrate the usage and benefits of ToMaTo in protocol analysis the scenario of malware analysis has been selected. Malware poses a huge security thread on Internet users as it has access to all data on the computer, can record user actions without the knowledge of the user and send this data over the Internet. The most common kind of malware allows the attacker to control the computer remotely and use it to launch other attacks and send spam mails. This way malware is currently responsible for most attacks and spam mails in the Internet.

An analysis of the communication protocol between the malware on the victims computer and the attacker can lead to methods to detect infected computers and quarantine them. Flaws in the communication protocol might offer a way to destroy the overlay network of the infected computers and thus break the control of the attacker. Although only a disinfection of the infected computer can completely remove the malware containment and attacks on the communication infrastructure of the malware network can prevent the disclosure of private user data as well of attacks and spam mails sent by the infected computer.

Handling malware and more so executing it raises some security considerations. After a computer has been infected with the malware it cannot be trusted anymore. That means that any protocol analysis must be independent of the infected computer. Also the computer must be reliably contained so the malware is not able to launch attacks or send spam mails or infect other computers.

IV. Malware Analysis with ToMaTo

ToMaTo has some unique features that are very useful is this scenario. ToMaTo supports multiple types of virtual machines and one of them can run Windows which is needed to run malware.

The fact that virtual machines can only use configured connections to communicate can be used to build a contained environment for an infected computer and to prevent the malware from spreading across the network. Since ToMaTo allows changes to running topologies, connections can be added and removed at runtime.

In ToMaTo, images of virtual machines can be downloaded and uploaded thus enabling to snapshot them. This allows to save the state before the infection and replay the infection without having to reinstall the operating system and setup the machine from scratch.

ToMaTo allows to capture networking traffic on the virtual connections without the help of the operating system. This especially helps when the operating system is infected with malware and cannot be trusted anymore.

V. Demonstration

The demonstration will show the analysis of malware using the ToMaTo testbed. For this analysis the topology in figure 1 is used.

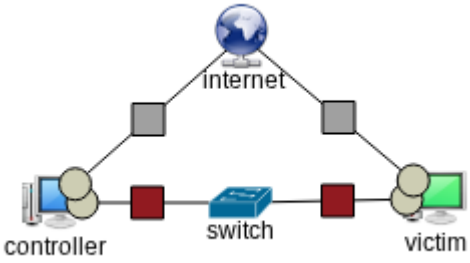

Figure 1. Protocol analysis topology

The topology consists of two devices, *controller* and *victim* and two connectors, *internet* and *switch*. The victim is a machine running Window that will be infected with the malware. The controller is a secured machine running Linux and thus cannot be affected by the malware. The controller will be used to emulate a counterpart for the malware to communicate with. The Internet is used by the victim machine to download the malware and by the experimentator to access the controller. The switch connects the victim with the controller and allows to capture the traffic between them.

In the first step of the analysis the topology is created and all devices are configured. Then the victim and the Internet are started and the malware is downloaded to the victim without executing it. Before the malware is executed, the connection between the Internet and the victim is removed so that the victim is completely contained. The controller runs a DNS server that is able resolve any name to its own IP address.

After the malware is started it will try to contact its malware server. Using the capture feature of ToMaTo the network data can be downloaded and analyzed with Wireshark. This way the host and the port of the malware server can be determined. Using this information the controller can programmed to run a simple application on that port so the malware can start its communication. Stepwise the protocol can now be reconstructed by capturing the data that the malware sends, sending a copy to the real malware server, and receiving the proper reply.

Since the analysis of malware is only a scenario to show the capabilities of the ToMaTo testbed for protocol analysis no complete analysis of the malware protocol will be done.

VI. Future Work

The tools used for protocol analysis will be bundled as a template for future usage in the ToMaTo testbed.

VII. Acknowledgements

This work has been done as part of the German Lab research project, funded by the German Federal Ministry of Education and Research (BMBF).

References

[1] Dennis Schwerdel, David Hock, Daniel Günther, Bernd Reuther, Paul Müller, and Phuoc Tran-Gia. ToMaTo - a network experimentation tool. In *7th International ICST Conference on Testbeds and Research Infrastructures for the Development of Networks and Communities (TridentCom 2011)*, Shanghai, China, April 2011.