Team Car RamRod

Project Proposal

An Analytic Honeypot for Virtualized Environments

Professor Andrew Bennett

Team Members:

Matthew McLeod
Patrick McDonald
Evan Leky
David Garrett May

Executive Summary Page

Aware of the challenges of detecting emerging malware threats Team Car RamRod has begun researching and building a virtualized honeypot platform. The principle idea being to leverage the benefits of a virtualized environment to aid the discovery process, in that after a piece of malware is encountered and analysed the affected system can be dropped while another is created within moments to continue scanning. Using this method we hope to raise the efficiency in creating profiles of the different types of malware by comparing newly encountered examples to known cases and building relationship links.

In order for our project to be successful our services must ensure that the platform be robust and able to be scaled at a moments notice. One of our priorities will be to establish a database of the malware that has already been encountered in order to allow repeats to be immediately disregarded and the resources recycled into a fresh deployment of our vanilla installation.

We believe that this tool will be a helpful addition to the tools that security professionals rely upon by creating a repository that individually identifies malware applications and their derivatives. With this database in place studies documenting the evolution of the different types of malware will become a more straightforward process.

Team Car RamRod plans to utilize as much open source material in this endeavor as possible in an effort to keep operational costs to a minimum. Any additions to the open source platforms will be shared with the open source communities responsible for the host platforms.

Scope and Limitations

This project is intended to develop a process for capturing and examining Malware through the use of a honeypot. Upon detecting a piece of malware the system in place will immediately isolate the affected virtual machine transitioning it from a “live” status to that of a sandbox. As this is happening a new instance of a virtualized machine will be created to take the place of the machine removed from live service. We will then be able to compare the two different states of the virtual machine and be able to record the differences of the two. This will allow for the effects of the malware on the system to be observed. Once the malware has been logged the sandbox can be deleted removing the infected virtual machine with it.

Project Goal

This project's focus is primarily the creation of a way of capturing and collecting known and new malware examples in order to catalog them and aid in making the detection process more efficient.

Objectives

Deliverables:

A scalable architecture that is able to detect the presence of malware on a clean virtual machine and divert it to an analytic sandbox without interrupting the system.

Approach:

We will first set up a virtualized server and deploy virtualized copies of operating systems. Once we have the virtualized environment setup we can deploy our honeypot and record our findings to further improve security.

Timeline:

Our team will first research the necessary items to complete the project, and from there get familiar with the different software that will be used. Once the team is familiar with the specific software that is being used we will begin to set up our virtual environment and implement the software that will be used. Once we have the virtual environment setup we can then begin testing the different attacking and recording softwares that will provide us with our necessary information. Once we have begun the testing phase we will each be able to record which types of attacks have occurred on the virtual environment and begin logging that necessary information. Once all documentation and logging is completed we will then begin to formulate the final project presentation into the appropriate documentation and then present final project.

Milestones:

- Week1: Project Planned.
- Week2: Initial Project Setup.(Servers and virtual machine)
- Week3: Tuning and Testing.
- Week4: Running and Presenting Findings.

Benefits

- Capture and analyze malware while protecting the virtual environment
- Aid in the efficiency of detecting new malware strains
- Develop an in depth catalog of known variants
- Creation of new plugins for existing open source projects

Assumptions

- Log attacks
- Discover origin of attack
- What software is being used to attack
- What it is attacking
- How is it attacking

Closing

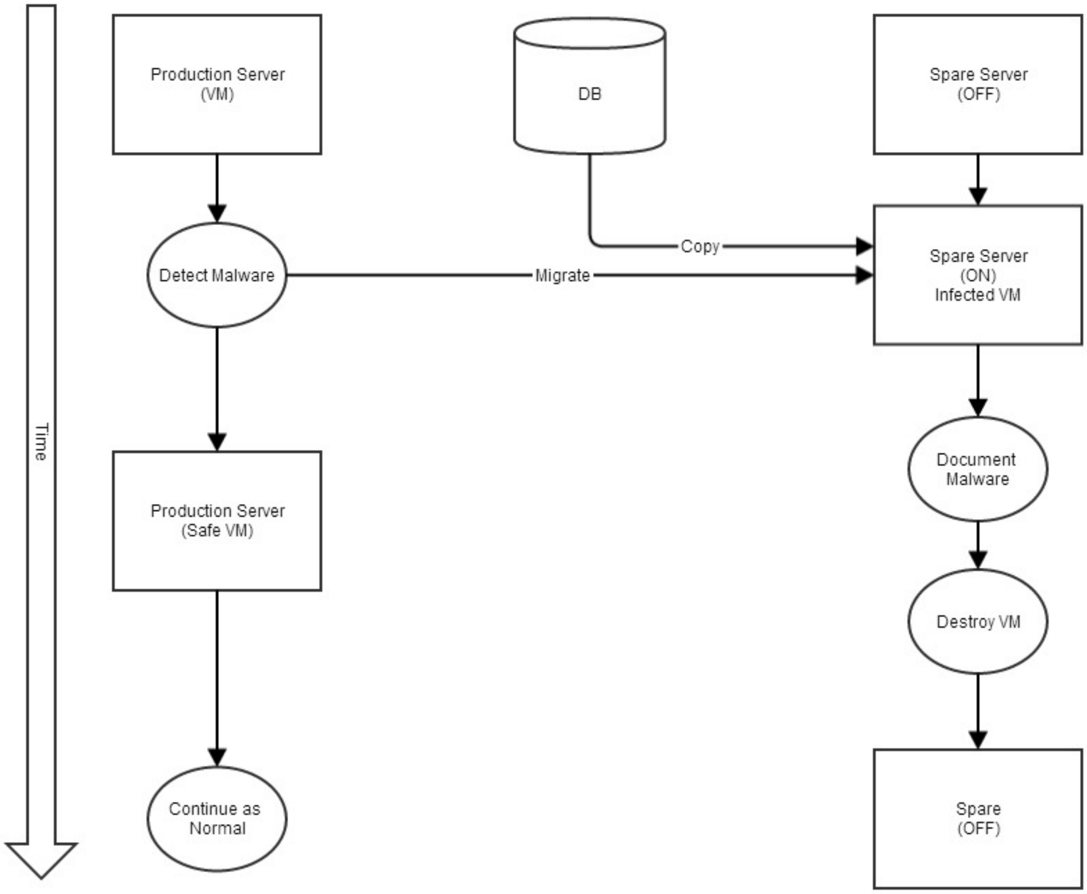
Team Car RamRod believes that this product will be able to expedite and increase efficiency when dealing with emerging malware threats. By being able to isolate and gather information about the different types of malware we encounter our project will eventually be able to distinguish the relationship between the newer versions of malware and their predecessors. By doing this we hope to develop the knowledge needed to speed up the identification and patching process. Using our approach after the pertinent data has been gathered from the infected virtual machine we will then be able to purge the malware in the sandbox by either reverting to an established snapshot or deleting and reinstalling the box entirely. After we have encountered a certain piece of malware we would then have the ability to mitigate the attack instead of analysing it for a second time by comparing it to our known samples.

Appendix

A. Resources

- VMware Workstation
- VMware vSphere
- VMware vCenter
- Cuckoo Sandbox
- Metasploit
- Windows Server 2008
- Windows 7
- Visualization software such as Gephi

B. Diagrams


Time
Production Server
(VM)
DB
Spare Server
(OFF)
Detect Malware
Copy
Migrate
Spare Server
(ON)
Infected VM
Production Server
(Safe VM)
Document
Malware
Destroy VM
Continue as
Normal
Spare
(OFF)