Process Coloring

An Information Flow-Preserving Approach to Malware Investigation

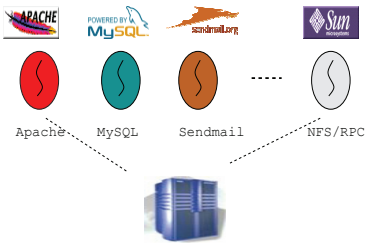
Eugene Spafford, Dongyan Xu, Xuxian Jiang, Ryan Riley
PC@cs.purdue.edu

Introduction

Cyberinfrastructures are facing increasingly stealthy and sophisticated malware threats. For example, recent reports have suggested that new computer worms and viruses deliberately avoid fast massive propagation. Instead, they lurk in infected machines and inflict contaminations over time, such as rootkit and backdoor installation, botnet creation, and private data theft. Current methods for detection and investigation do not fully exploit the use of *information flows* tracked at the operating system level. We argue that OS-level information flow is currently an under-utilized tool for malware investigation. We will use operating system information flows to propagate malware break-in *provenance* information to demonstrate that provenance preservation can help achieve more efficient and effective malware investigation. We will also show that this technique can be used to produce live alerts for malware that existing tools are unable to provide.

Preserving Provenance

One of the main goals of our technique is to quickly and easily determine the malware break-in point. To accomplish this task, we start by assigning each potential break-in point (usually a service, such as a web server or mail server) a unique, system-wide identifier that can be visualized as a *color*. See the figure to the left for an illustration of this. This color is then diffused to any objects or processes influenced by that service on the system.



For example, if a red web server process is hijacked by an attacker and creates a file named `trojan`, then that file will also be colored red. If that malicious file is later executed and changes some important system files, all of those files will also be colored red. In addition, all of the system level operations (such as process creation and file read and writes) are being tracked in a log file and are associated with the color red.

Using Provenance

Now that provenance information is being preserved using the convention of colors, there are two primary ways the color information can used in malware investigation.

First, a system administrator can use the color information to quickly determine the break-in point for a piece of malware. If the administrator discovers a rogue file or process on the system, she can simply check the object's color and know the *break-in point at a glance*.

Second, the color information and the logging information can be used to generate *runtime malware alerts* by looking for coloring anomalies such as color mixing, where a process bears multiple colors that should not occur together naturally. We see this capability of color-based live malware alerts as one of the most intriguing aspects of our approach.

Applications and Impacts

The impacts of process coloring are two fold:

1. Process coloring can be readily integrated into the *existing* log-based intrusion analysis tools to improve the timeliness, efficiency, and tamper-resistance of malware investigations in cyberinfrastructures.
2. Process coloring will enable *new* tools for system monitoring and malware alert. We envision the following application scenario: At runtime, the administrator non-intrusively monitors the OS-level information flows visualized with process and object colors. When the log color exhibits an abnormal pattern, she will notice that by visual inspection or by an alert generated by the tool.

The promise and practicality of process coloring have been demonstrated in our previous research in real-world, self-propagating worms, using our first-generation process coloring prototype. For every worm investigated, we are able to receive runtime alerts and identify the break-in point of the worm *before* detailed log analysis. Moreover, reduction of inspected log data is achieved in every worm experiment. Recently, we have begun to enhance and apply process coloring to investigate malware attacks against client-side software such as web browsers.

Process Coloring

An Information Flow-Preserving Approach to Malware Investigation


Eugene Spafford
Dongyan Xu
Xuxian Jiang (GMU)
Ryan Riley


1. Problem

Malware like worms, rootkits, and bots are an increasing threat to cyberinfrastructures. Current methods for detection and investigation do not fully exploit the *malware break-in provenance* information propagated along operating system information flows.

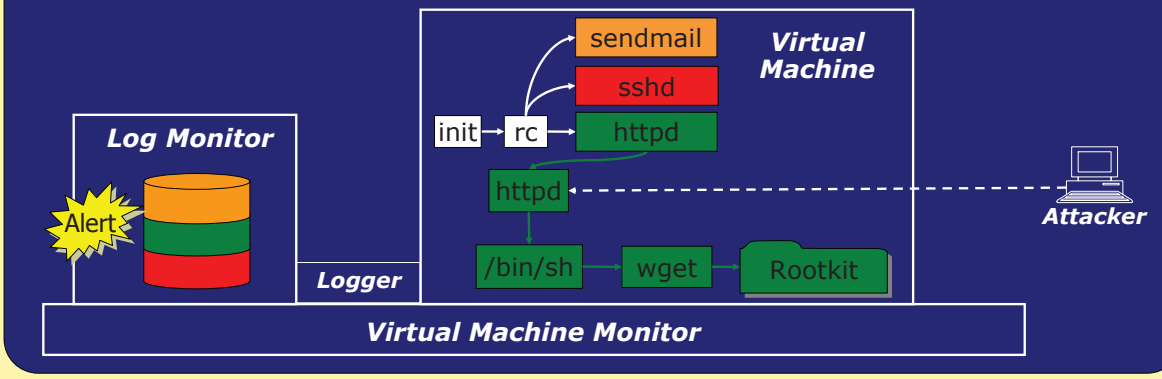
2. Goals

- Shorter malware infection-to-detection interval
- Fast, accurate determination of malware break-in point
- Complete account of malware contaminations
- No disruption to normal system operations

3. Our Technique

- Define break-in provenance by assigning each potential break-in point a *color*
- Diffuse colors along operating system information flows
- Detect and investigate malware based on color and coloring anomaly

4. System Overview



5. New Capabilities

- Color-based malware alert
- Color-based determination of malware break-in point
- Color-based log file partitioning

6. Applications

- Server and client side malware investigation
- Malware damage recovery
- Bot detection and profiling

7. Implementation Details

- Xen virtual machine monitor 3.0.2 hosts the protected guest OS.
- Linux 2.6.16 is enhanced with coloring capability in the guest VM.
- Color diffusion is triggered by system calls.
- Log entries and log monitor are located outside of the guest VM

DTO NICIAR NICECAP Kickoff Meeting
March 7, 2007
Chantilly, Virginia