HOW TO PLAY NICE IN THE SANDBOX

01

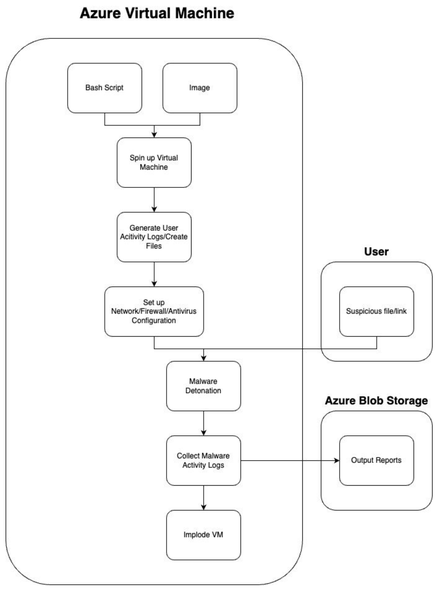
STUDENTS: JACOB LUSH, NATHAN BORN, KAI-JEN ZHENG, DANIEL CHEN, ZHAORUI ZHANG, HAYDEN GUO

Introduction

- BECU, being a credit union, is a valuable target for malware since they manage a lot of valuable personal information about their customers.
- BECU needs a way to stay one step ahead of malware creators so that they can protect their customers. If they are reactive instead of proactive in their approach, their customers can be harmed.
- A sandbox environment is the perfect tool for analyzing malware threats before they cause any damage. A sandbox environment is an isolated environment which can be used to safely detonate and analyze malware.
- Our project aims to provide BECU engineers with a sandbox environment tool which can be used to quickly and easily analyze malware on a variety of different types of systems.

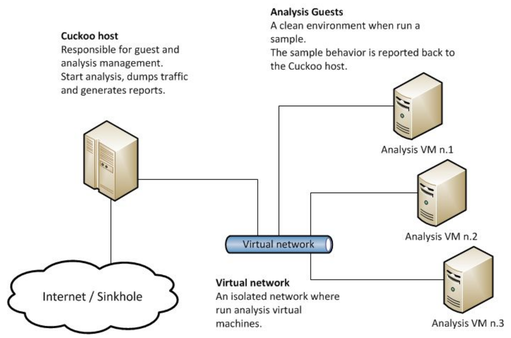
Requirements

- Perform malware detonations safely in an isolated environment
- Record file changes made by malware in the environment
- Record network activity that occurs in the environment
- Support a variety of Windows and Linux operating systems and versions for malware detonation
- Emulate natural file and network activity to trick sophisticated sandbox-detecting malware into detonating
- Automatically generate a report of file changes and network activity that occur during malware detonation
- Provide a user interface for users which allows for submission of malware



Cuckoo Sandbox

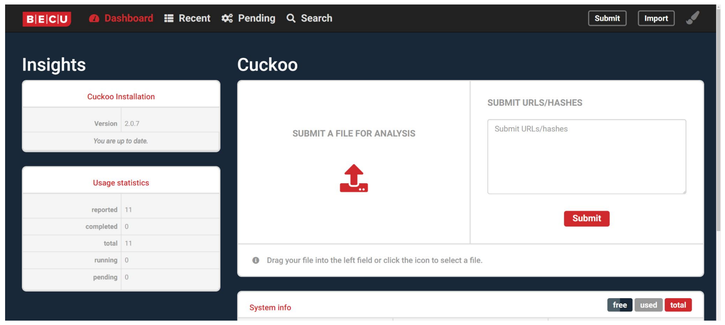
- We use Cuckoo as our primary tool for creating sandbox environments.
- Cuckoo comes loaded with useful tools for VM creation, malware analysis, report generation, and user interface
- Our Cuckoo server hosts an API which is used by our front end to perform a wide range of tasks from malware analysis task submission to generating a report.
- We add our own custom ISO files to add support for malware detonation on different types of machines.
- The sandbox environments deployed from our custom ISOs come loaded with files that regular users would have installed as to prevent sandbox detection by malware.
- Our sandbox environments use FakeNet to emulate network activity which also aids in preventing sandbox detection from malware.
- Some useful features of Cuckoo include taking screenshots of the sandbox environments during execution, a fully featured web portal interface, the ability to install custom plugins, and much more.



Cloud Deployment

- Our Cuckoo server host is deployed on an Azure virtual machine where Cuckoo has access to its resources for creating sandbox environments.
- We use a large Azure virtual machine with 4 core processor, 1Tb of disk space, and 16Gb of RAM in order to support the resource requirements for hosting several different machines with different operating systems.
- Our server uses nginx to host our web interface. Nginx is playing the role of being a web server/reverse proxy to our Azure hosted Cuckoo server host.
- Various networking changes on Azure are made to open up the server to users.
- Users have the option of using the web interface or directly connecting to the Cuckoo host machine using SSH to perform malware analysis.

Web Interface



- BECU engineers access the tool through a web portal which includes a variety of tools for submitting and viewing submitted analyses.
- Includes “dashboard” tab for submitting malware for analysis, viewing sandbox environment status, checking version and usage information, and listing the past few analyses.
- Includes “recent” tab for viewing individual analyses in greater detail. Includes Cuckoo’s suite of report analysis tools for viewing network and file behaviors.
- The web interface is a modified version of the Cuckoo Sandbox web interface. It follows the stylings listed in the official BECU Digital Pattern Library.

Development Process

- We began development by evaluating several different sandboxing tools, virtualization softwares, and analysis tools.
- We found that Cuckoo satisfied all of our requirements for a sandbox environment as well as being open source and free to use.
- We decided to use VirtualBox to create our sandbox environment VMs since it was also open source and worked well with Cuckoo.
- We agreed to use TCPDump installed onto Cuckoo to analyze network activity in our sandbox environments. Cuckoo uses Volatility to perform deep file analysis.
- We then worked on our own custom API for submitting analyses and downloading reports. At this stage we had our own fully customized UI.
- At the end of the project, we shifted toward using a modified version of the Cuckoo API and frontend since it came loaded with features we liked.

Future Improvements

- Fully featured custom API and user interface instead of modified Cuckoo
- Sandbox environments hosted directly on Azure instead of through Cuckoo
- More ISO files to support more types and versions of operating systems
- Integrated manual analysis tool with remote desktop to sandbox environments

ELECTRICAL & COMPUTER ENGINEERING
UNIVERSITY of WASHINGTON

ADVISORS: KELLY HO, PAYMAN ARABSHAHI

SPONSORS: BECU