“The more laws and order are made prominent, the more thieves and robbers there will be.” Lao Tzu

How To Steal a Virtual Machine

System and Network Security

Professors: Bill Lidinsky and Kevin Vaccaro

Assistants: Jeremy Hajek and Dawid Broda

Burim Bakalli
Frederick Eichhorst
Madeleine England

Virtualization

- Software used to create virtual resources from physical resources
- Run multiple virtual machines on a single physical machine
- Sharing of resources
- Rapid rate of adoption across IT field
- Implemented through the use of Virtual Infrastructures

Virtual Infrastructure

- Hypervisor (also known as the Virtual Machine Monitor)
- Provides the platform that virtual machines sit on
- Can be either:
  - Software run on top of another operating system
  - Software installed on the physical system without an operating system (also known as bare-metal hypervisor)

VI Security

- Unique threats to VI
- Many systems directly tied to one single environment
- Virtual Infrastructures created without a plan, with best practices ignored or unknown
- Administrators uninformed about security issues and known vulnerabilities
- Improve security through demonstration of vulnerabilities

Project Goal

- Steal a Virtual Machine from an un-protected host in a "typical" network scenario
- Use VASTO, a suite of Metasploit plug-ins created by Claudio Criscione, for doing VI penetration tests
- Exploit a known vulnerability in older VMware environments
- Replicate Claudio's demonstration at the Troopers conference

“Typical” Network Scenario

- Virtual Infrastructure is not secured properly
  - Weak passwords
  - Updates and patches not current
  - Weak network configuration
  - More common then most would like to admit

The Vulnerability

- A malicious user can bypass authenticating to a host and download any files they want, including entire Virtual Machines
- First reported by Justin Morehouse and Jason Kratzer
- Described in VMware Security Advisory VMSA-2009-0015 section 3b
- "A directory traversal vulnerability allows for remote retrieval of any file from the host system."

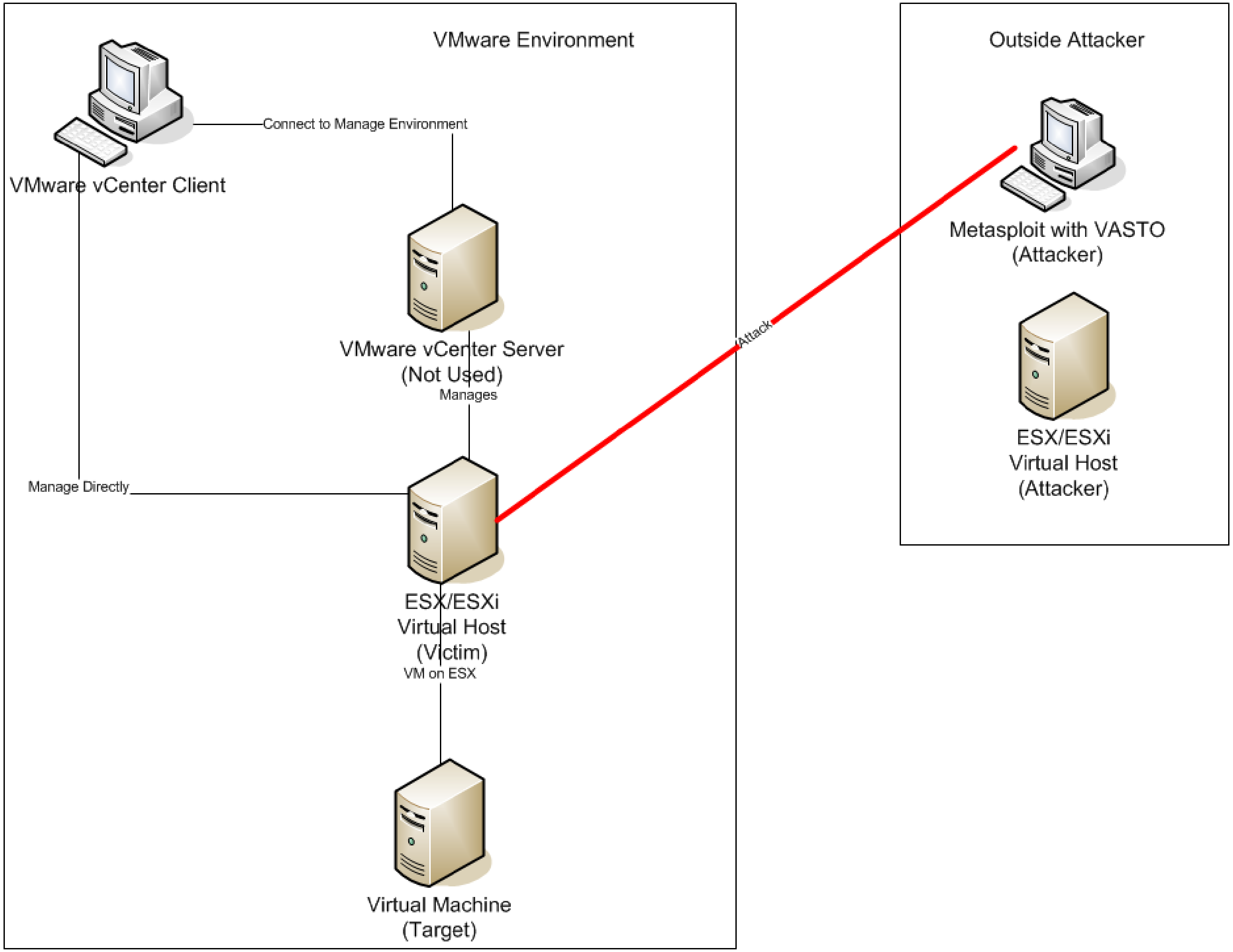
The Plan

- Create an environment based on the “typical” scenario described
- Create a VM inside the Victim ESXi Server
- Using an edited VASTO vmware_guest_stealer module, exploit the vulnerability, break into the ESXi Server, and steal the VM
- Transfer the stolen VM to the attacker ESXi Server
- Create a new VM from the stolen data and use it

Software and Operating Systems

- Client System
  - Windows XP, VI Client software installed
- Management Server
  - Windows Server 2003, VI Server installed
- Attacker System
  - Ubuntu, VASTO designed to work with Linux/Ubuntu
  - Metasploit, a popular application for penetration testing
  - VASTO, plug-ins for Metasploit, ruby scripts focusing on VIs
- Host Servers (Attacker and Victim)
  - ESXi 3.5, free, small, bare-metal hypervisor from VMware.

Project Environment



Attacker Environment

- Ubuntu 10.4
- Metasploit Framework 3.5.1
- Metasploit requires Ruby
  - Ubuntu does not provide Ruby by default
  - Ruby support must be installed on the system using the apt-get command
- VASTO 0.3 plug-ins
- Attacker ESXi Server will be used to access stolen data

Victim Environment

- Operating system and applications set up with default configurations
- Systems and servers located in the same network
- Target VM created

Exploitation Setup

- Attacker System
  - Load vmware_guest_stealer module
- Set parameters for Ruby script
  - RHOSTS (the Host Server IP address)
  - RPORT (443)
  - TARGET (ESXi)
- Exploit
  - Metasploit checks if Host is vulnerable to exploit
    - Provides a list of all vulnerable virtual machines

Exploitation

- Set parameters of FILE (the full path and name of vm)
- Set OPERATION (FILE, to download files)
- Exploit
  - Download the VM files off of the host

Failure

- Ruby script
  - SSL verification
  - “grep” command
  - Un-locatable information in the .vmdk file
- Power-on failure
  - Vmdk flat file can’t be downloaded when the VM is on

Success

- Files
  - *vmname*.vmx
  - *vmname*.vmdk
  - *vmname*-flat.vmdk
- May be more than one vmdk and flat.vmdk file
  - One of each will be created for each virtual hard disk on the virtual machine
- Files are on the attacker system

Accessing the VM

- Confirm stolen files can be used by the attacker
- Transfer stolen files to VI Client capable PC
- Upload files to Attacker ESXi Server
- Create new VM using stolen files
- Open VM and view contents

Prevent the Exploit From Working

- Update, Update, Update!
- Protect the network
- Follow best practices (patching, configuration, etc..)
- Educate admins on Virtualization security
- Protect the VM the same way you would a physical machine and mission-critical files – It is both!

Conclusions

- Virtualization security is a new area of development, and not well understood by many
- Vulnerabilities, when not dealt with, can have disastrous consequences
- Critical to employ the same security measures in a virtual environment as in a physical environment
- Education in virtualization security is key to protecting virtual environments

Demonstration

- Show the target VM
- Open Metasploit and load in VASTO
- Run the exploit to show vulnerable VMs
- Run the exploit to steal the VM
- Move stolen files to Attacker ESXi host
- Create a new VM using the stolen data
- Show the new VM to confirm the VM was stolen
- http://www.youtube.com/watch?v=s1HOShB1D6k

Thanks

- Claudio Criscione. Without his desire and expertise in VI exploitation, we would have not gained the in-depth understanding and knowledge that we now have.
- Bill Lidinsky, Kevin Vaccaro and Dawid Broda for their ideas and encouragement.
- Jeremy Hajek for helping us acquire the hardware and software.