Virtually Pwned
Pentesting VMware

Claudio Criscione
@paradoxengine – c.criscione@securenetwork.it
Breaking virtualization means...

...hacking the underlying layer
...accessing systems locally
...bypassing access and network controls
...hitting multiple targets at once
...likely being able to find old vulnerabilities

96% of the Fortune 1000 *
Small number of different solutions deployed

* http://www.vmware.com/company/customers/
The elephant in the room
Escaping the VM

Yes, it can be done
Yes, exploiting is usually hard
Yes, it can be patched
Yes, it will happen again

No, it is not something you can easily audit

We’re going to attack virtualization infrastructure
Tools Of The Trade
No Tool was Available
The Virtualization ASsessment TOolkit

It is an “exploit pack” for Metasploit focusing on virtualization and cloud security.

Beta 0.4 will be released shortly after HITB.

Tnx to Luca Carettoni, Paolo Canaletti, drk1wi for helping with modules!
Desktop Products: Workstation, Player.
Not interesting during PenTest (for now ;))

Server Products: ESX, ESXi (now Hypervisor), Server
The “hypervisor layer”

Management Node: Virtual Center (vCenter), running on Windows Server

Client: binary, running on Windows
The Plan
Easy enough, Hannibal would be proud

1 - Recon

2 - Attack

3 - Pwn
How do you notice?

It’s virtual!
Recon

Local – are you in a VM?
   Easy – Check MAC address, processes
   Not so easy – Hardware access
      Thanks to Nahuel Grisolia's memory-based detection tool, it's easy as well

Remote – where’s Virtualization?
   Fingerprinting network services
   Helpful to discover “hidden” virtualization installations
Handy SOAP API to call
Works on most VMware products
Module leverages standard Metasploit scanner features (e.g. IP range scanning)

[...]
Every little bit* helps :-)
A multi layered attack
Client

Internal

Hypervisor

Management

Support
In the beginning was the command line

We used to have binary clients

Then everyone moved to web applications

Now, back to binary clients, like XEN Center, or VMware VI client

Can we exploit these clients? Let’s see…
VI Client Auto Update feature

1. GET /client/clients.xml
   - Autoupdate URL
2. RetrieveServiceInstance
   - ServiceInstance
3. RetrieveServiceStatus
   - Status
4. GET /client/clients.xml
   - Autoupdate URL

Login
<ConfigRoot>

<clientConnection id="0000">
  <authdPort>902</authdPort>
  <version>3</version>
  <patchVersion>3.0.0</patchVersion>
  <apiVersion>3.1.0</apiVersion>
  <downloadUrl>https://*/client/VMware-viclient.exe</downloadUrl>

</clientConnection>

</ConfigRoot>
vmware_vilurker

The Vilurker module can perform user-assisted code execution provided you can do MITM on a client.

Almost no one is using trusted certificates.

No code signing on updates, but user gets a certificate warning.

BONUS INFO: no SSL check on VMware Server 1.x.
Direct Hit
This path traversal was discovered by Flick and Morehouse and presented last year. Exploit was released as a perl script and it has been ported to VASTO.

It can be used to retrieve any file as the root user, including non-running guests. Works on outdated ESX, ESXi, Server.
vmware_updatemanager_traversal

JETTY-1004
VMware Update Manager includes Jetty 6.1.16
Runs on the vCenter (management) Server

Jetty 6.1.16 is vulnerable to path traversal (again)

Here is the magic string

/vci/downloads/health.xml/%3F/../../../../../../../$FILE
Ok, we can read files on the vCenter (Windows) as System. So what?

Follow me!
Introducing vpxd-profiler-*

It is a “debug” file written by vCenter.

Lots of information inside. Let’s go for low-hanging fruits for now. More to come 😊

/ SessionStats/SessionPool/Session/Id='06B90BCB-A0A4-4B9C-B680-FB72656A1DCB'/Username='FakeDomain\FakeUser'/SoapSession/Id='AD45B176-63F3-4421-BBF0-FE1603E543F4'/Count/total 1
So where do I write the SOAP ID?
Ask for advice to your monkey.
vmware_session_rider

Using the session is non-trivial: VI client has tight timeouts

The module acts as a proxy to access vCenter using the stolen session.

Will fake the login to the client and can be easily tweaked to act as a password grabber (unlike VIlurker).

The last exploits combined: vmware_autopwn
Bonus!

Any read-only access to the vCenter file system means vCenter takeover!
The Interface is FUN

Web-based & Complex

XSS

URL Forwarding

BONUS: Shutdown keyword has not been changed, can shutdown local Tomcat
vmware_webaccess_portscan

CVE-2010-0686

“URL Forwarding” means performing POST requests on remote hosts.

Can be used to exploit IP-based trusts and reach internal networks.

Not just portscan!
Management is not just interface

vCenter connects to ESX server via SSL [SOAP]
Certificates are usually not valid, but stored.

Let's do a nice* diagram

*Diagram powered by ORACLE's Open Office
I'm sure it will look better with LibreOffice
Attacking it

MITM vCenter – ESX → Connection Broken

On reconnection, the vCenter will check the certificate. And fail, usually.

SSL MITM → Admin gets usual warning

Admin agrees → password sniffed

If you are a Metasploit guru, let's speak about MITM in Metasploit Later
Differences...

It seems 4.0 and 4.1 behave differently when it comes to reconnection.

4.0 won't reconnect on a different CN.

Just change the CN to match the right one.

Word of advice:
You really, really, don't want MITM on any VMware component.
If everything else failed...
vmware_login

If nothing works, you can always bruteforce!

Will do standard metasploit bruteforcing

Local Windows Administrator is admin of the vCenter by default, and it has no lockout.
For ESX, ESXi the local root user has no lockout. [watch out for AD integration though]
Which means a lot of bruteforcing fun.
What’s different?

Multiple local EscalationOfPriv in Virtual Machines
Will eventually include these as modules as well
Discovered by great researchers
Low level attacks, close to the CPU or OS

What else?
Our new Attack surface

Paravirtualization and support tools
vmware_sfcb_exec

CVE-2010-2667

A vulnerability in Virtual Appliance Management Infrastructure resulting in code exec as root

Requires authentication OR can be exploited locally without any authentication.
The attack

<?xml version="1.0" encoding="UTF-8"?>
<CIM CIMVERSION="2.0" DTDVERSION="2.0">
 <MESSAGE ID="13" PROTOCOLVERSION="1.0">
  <SIMPLEREQ>
   <METHODCALL NAME="SetServerName">
    <LOCALCLASSPATH> <LOCALNAMESPACEPATH>
     <NAMESPACE NAME="root"/>
    </LOCALNAMESPACEPATH>
    <NAMESPACE NAME="cimv2"/>
   </LOCALNAMESPACEPATH>
   <CLASSNAME NAME="VAMI_NetworkSetting"/>
  </LOCALCLASSPATH>
  
  <PARAMVALUE NAME="HostName" PARAMTYPE="string">121;$(echo${IFS}ls${IFS}-l)>/tmp/echo</PARAMVALUE>
  </METHODCALL>
  </SIMPLEREQ>
</MESSAGE>
</CIM>

Kudos to Marsh Ray and others for this Twitter-Powered payload ;-)
External references

If you didn't see Brossard yesterday you missed out :-)

Go back and see the “internal differences”!
So, can we attack virtualization?
Summing up

You can attack the admin client, **sniffing** the password or **owning** the administrator.

You can attack the hypervisor and its core modules (by **path traversal**).

You can hijack other user’s **sessions**.

You can attack the administration **web** interface.

You can attack supporting services on the virtual machine.

So, we have a problem!
Questions
What now?

Rethink about virtualization security
Check what you have
Keep on hacking!

Claudio Criscione
Twitter - @paradoxengine
Email - c.criscione@securenetwork.it
Blog & Download - vasto.nibblesec.org
Work & Virtualization PenTest – securenetwork.it