WEF - Web Exploit Finder
Installation Manual

Automatic Drive-By-Download detection
in a virtualized environment

This documentation is a brief installation guide to set up a working environment for the Web Exploit Finder 2.0 architecture as it was used in Summer 2007.

It shows how to set up one computer that is used for both the Ruby on Rails web interface as the management console and the VMware Server with the virtualized clients as unpatched Microsoft XP versions with Internet Explorer 6.0.

This manual is quite advanced and does not cover the basics such as Linux server installation. It also requires you to understand how virtualization works and how to set up a VMware Server.

For more information about the architecture and its functionality, see www.xnos.org.

Contents

1. Requirements / Operating Systems

First of all, you need at least one very powerful computer which acts as the host for the virtual machines. Every virtual machines uses 256 MB of RAM and about 1 GB for the virtual hard disk.

In our example we use a 2xAthlon XP 1800+ with 4GB RAM and a 30GB Hard Disk total which was gracefully provided by the Internet Security department at the Hochschule der Medien in Stuttgart, Germany.

For the operating systems you need a CD image of an unpatched Windows XP Professional without any service packs. Since we use a Value User License we don't need a specific license key for every virtual machine. For the host operating system we use Debian Linux in the stable (4.0, Etch) flavor.

Please download Debian Linux from www.debian.org and install it with the basic configuration. Please set up the operation system so you can access the internet. All data files (VMware images, Ruby on Rails application) will be stored in the /data/ directory. Please make sure that there will be enough space left for the images. You will need at least 10 GB, depending on the number of virtual machines you want to run simultaneously.

2. Host Network Configuration

On the host machine, you need a working DHCP server to dynamically assign IP addresses from a subnet. Fortunately, Debian Linux provides this almost out-of-the-box. Run the following command as root to install the DHCP server (and some other userful tools):

```
apt-get install dhcp vim bzip2
```

Now we need to configure the DHCP-Server. First, we need a free subnet to use. In our case, the Debian machine uses the IP-address 141.62.88.250/24 (usually you can find out by typing /usr/sbin/ifconfig). Now we need to edit the file /etc/dhcpd.conf. Beside the routers, subnet (+ mask) and DNS servers, we map IP addresses to MAC addresses here, which is very important. VMware needs MAC-Addresses to talk, although WEF communication doesn't run as deep and uses IP addresses. Therefore we decide to keep the IP- and MAC-Hardware address fixed

here. As you see, we only create the prototype and three clients for the sake of keeping the overview. You should, of course add more clients. Just add up the ethernet address by one as well as the MAC address (notice that you need to up in HEX numbers). Our VMware scripts will take care of mapping inside VM later on. This is how the sample dhcpd.conf should like (+ adding more clients, like 20 or so to be shure).

```
subnet 141.62.88.0 netmask 255.255.255.0 {

          option routers       141.62.88.254;
          option subnet-mask   255.255.255.0;

          option domain-name-servers 141.62.1.5, 141.62.64.100;

          range dynamic-bootp  141.62.88.50 141.62.88.150;
          default-lease-time   21600;
          max-lease-time       43200;

          host prototype {
                    hardware ethernet 00:50:56:01:48:32;
                    fixed-address 141.62.88.50;
          }

          host client01 {
                    hardware ethernet 00:50:56:01:48:33;
                    fixed-address 141.62.88.51;
          }

          host client02 {
                    hardware ethernet 00:50:56:01:48:34;
                    fixed-address 141.62.88.52;
          }

          host client03 {
                    hardware ethernet 00:50:56:01:48:35;
                    fixed-address 141.62.88.53;
          }
}
```

After that, please restart the DHCP server via:

```
/etc/init.d/dhcpd restart
```

3. Virtualization Layer & Windows XP Image

This section describes how to install a ready-to-go VMware Server (at the time of writing this document, the latest release was version 1.0.3) and a non-modified Windows XP version in a virtualized environment. First, download the VMware Server 1.0.3 from vmware.com[2]. Please make sure that you download the „Linux tar.gz version“ and a console for your client system (either Windows or Linux). Install some prerequisites and then the VMware server package via

```
          apt-get install psmisc libx11-dev libice-dev libxtst-dev libxt-dev libxrender-
dev gcc make
          # also make sure to apt-get install the latest linux-headers-KERNELVER package
matching your running kernel
          tar zvxf Vmware-server-1.0.3-44356.tar.gz
          cd vmware-server-distrib/
          ./vmware-install.pl
```

and answer every question with yes (= enter), except for the following topics: While configuring the network make sure not to configure NAT or host-only networking, since we are using the virtual machines as bridged network devices. Please select „/data/vmware“ as the image directory. Now you can connect to the VMware Server with your client console (please install it on your workstation), and connect to the machine on port 902 (make sure, the firewall does not block connections on that port). You will also need a serial number. You can register on the VMware website to get the serial number you need to enter.

Create a new virtual machine by following the instructions of the VMware wizard. Among other default

configuration options, make sure to set the following:

- Naming: Call the VM configuration „winxp“, save it in the directory /data/vmware/winxp_prototype
- Hard drive: Set the hard drive to „increasing“ (so not all the space of the harddisk will be taken up on creation) and set the maximum hard drive size to 1.5 GB which should be enough for a default Windows XP machine.
- Memory: 256 MB or maybe even 192 MB could be enough
- Network device: Bridged

Remove the floppy device since you probably won't need it. Let the virtual CD-ROM drive point to your Windows XP installation CD image that needs to reside on the server for that.

Boot up the virtual machine and configure the BIOS so network boot is disabled by default and the VM boots from the CD drive as the primary boot device.

You should now see the Windows XP installation process, where you need to set up the partition of the hard drive, and later configure your keyboard and localization settings. Also, enter your valid serial key for Windows XP. After the copying of the files, name the new computer „WEFCLIENT“, set the admin password to „project“ and create a user called „wef“. You also need to set up your internet connection. Do this by selecting the option that you are „connected to the internet through a LAN router“. After that, you should be able to boot into the regular Windows XP desktop.

First, install the „VMware Tools“ by clicking on the „VM“ menu item in your console and selecting the „Install VMware Tools“ option. This will greatly enhance the performance of the console. After this point, you can remove the „CD Drive“ from your virtual machine peripherals to speed up boot time.

Now you need to configure your default Windows user since the WEF scripts are going to connect through the „wef“ user you created before and you are logged in right now. Select the user through the Windows Control Panel and set the password to „project“. While you're in the control panel, also turn off the automatic updates under the System icon. In addition to that, we need to have the automatic login still preserved although a password is required (no, it is not possible to set no password since the VMware API is very restrictive to that). Please see [3] for instructions with the Registry. Further on, please uninstall unneeded things such as the Windows Messenger [4], also reset the IE homepage (e.g. to www.xnos.org or to about:blank). Then we suggest for you to do some initial web surfing to get rid of some of the default questions („Do you want to send this form...?“). In our example, we make search for „WEF“ on Google and hit https://www.amazon.com/ to hit a secure website and then click on a category to leave the secure website again.

The BrowserControl component needs a .NET runtime to work properly. You can get this from the Microsoft website[5]. Download and install the runtime in the VM prototype. It is possible that you need to install the „Windows Installer 3.0“ on to your system before you can install the .NET runtime. Please follow the instructions on the screen to install this requirement as well.

The last part is to install the vmcontrol files. Unpack the WEF package and run the installer like this:

```
cd /tmp/
wget http://www.xnos.org/fileadmin/labs/wef/wef-2.0.tar.gz
tar zvxf wef-2.0.tar.gz
cd wef-2.0
./install.sh vmcontrol
```

The vmcontrol files will be copied into /data/vmcontrol. Please see and edit the file /data/vmcontrol/vmcommon.sh to see if the variables reflect your configuration, especially when you use a different subnet and IP addresses.

You probably only need to configure the prototype MAC and IP address. The rest will be taken care of automatically. To try out the scripts, please shut down the prototype, set the verbose option to 1 and run

```
/data/vmcontrol/vmclone.sh
```

Make sure to set the verbose option to 0 again when you are running the scripts in production mode.

Also, the file „runBC.bat“ in /data/browsercontrol/ needs to updated to the SOAP path of the Management Console. You need to modify the IP address and set it to the IP address of the Management Console.

Now, the Windows part is done and the prototype client is ready to deploy into the other machines.

4. Management Console

The management console is a web-based Ruby on Rails application with a MySQL backend. First, you need to install the needed packages via apt-get:

```
apt-get install mysql-server-5.0 mysql-client-5.0 libmysqlclient15-dev
libmysql-ruby ruby1.8 ruby1.8-dev irb1.8 rdoc1.8 rails rubygems
```

Please make sure that the ruby version is at least 1.8.2, otherwise follow the installation steps for „Ruby Installation“ on the Rails Wiki[6] .

If everything went well, you can now use the command-line tool „gem“ to install Rails, the Mongrel webserver and some MySQL bindings.

```
gem install mongrel -y
gem install -v=1.2.3 rails -y
gem install -y mysql -- --with-mysql-config=/usr/bin/mysql_config
```

If the installer asks you about a version for mongrel, choose the highest version with no „win32“ suffix. Install the database called „wef“ and some default database entries with the following command.

```
cd /tmp/wef-2.0
./install.sh wefwww
```

Now you can start the server for the first time.

```
cd /data/wefwww
script/server -e production -p 80
```

You can now use the WEF web interface to check websites if they are malicious. You can connect to the web interface via

http://localhost/

or if you have a qualified name for your server, we prefer to use that instead of „localhost“, which is also useful for connecting to the web interface via a remote machine.

5. Further Instructions

Enjoy your installation now. You can use the web interface to fill in URLs that should be checked. While testing we also enjoyed using the VMware Server Console to see how the clients are surfing to the website. Please see the about page in the management console for more information and problems. Thank you for using WEF.

Benjamin Mack, 2007-10-27

[2] http://download3.vmware.com/software/vmserver/VMware-server-1.0.3-44356.tar.gz
[3] http://support.microsoft.com/default.aspx?scid=kb;en-us;315231
[4] http://www.pchell.com/support/removemessenger.shtml
[5] http://www.microsoft.com/downloads/details.aspx?displaylang=en&FamilyID=0856eacb-4362-4b0d-8edd-aab15c5e04f5
[6] http://wiki.rubyonrails.org/rails/pages/RailsOnCentos