Global Information Assurance Certification Paper

Secure Browsing Environment

GIAC (GSEC) Gold Certification

Author: Robert Sorensen, rssoren@gmail.com
Advisor: Egan Hadsell

Accepted: September 20, 2011
Updated: May 11, 2012

Abstract

This paper will describe the current browsing environment and potential security issues. This is a very relevant topic as one reads about zero-day exploits or vulnerabilities in current browsers and/or applications like Adobe Flash or Acrobat Reader. Also, this paper will describe and detail the technical implementation of a Linux-based VM browsing environment with a step-by-step install and configuration guide. As people use web browsers on a daily basis, an interest in how one can securely and confidently conduct financial business using Windows is critical in today's computing environment.

1. Introduction

Today's computing environment is fraught with much treachery. It used to be that one could surf the web without any thought of infection or loss of private information. Those times have changed! One might argue that the safest connection to the web is no connection at all. However, this is not feasible in today's social networked world (Powell, 2011). The target has only increased for hackers.

The challenge is how to be an interactive player in a scary playground called the web. From image searches that are poisoned by cybercriminals (Hecht, 2011), to high-profile websites with booby-trapped ads (Smith, 2011), one can only sense a fear of an impending cybergeddon attack on a much larger and disruptive scale (Davis, 2011).

With the recent report from McAfee described as "Operation Shady RAT," that hackers have been lurking in the systems of major organizations for up to five years, accessing and reviewing top-secret data, is a scary thought (Lau, 2011). While this attack is sophisticated and advanced, it is one of many similar attacks taking place daily.

Additionally, older exploits are continually being updated and morphed into more serious threats. A recent example is the Ramnit worm variant that was transformed into financial-focused malware capable of draining bank accounts, which has reportedly incorporated bits and pieces of the publicly available Zeus malcode to make it more effective (Westervelt, 2011).

The question one must seriously consider is, "How can one defend themselves against such seemly impossible odds?" To this end, this research paper attempts to present a Secure Browsing Environment Virtual Machine (SBE VM) in which to provide a fighting chance against today's sophisticated and determined hacker.

2. Current Trends

One of the best sources to depict the current trends in IT security is the annual report produced by Sophos, *Security Threat Report 2011*. A highlight from the report really depicts the strategy of the cybercriminal: "By preying on our curiosity,

cybercriminals are able to use psychological traps to profit from unsuspecting users of technology. Malware scams and exploits are targeting social networking websites, applications, devices, and users proliferate" (Sophos, 2011, p. 2).

The report goes on to indicate that 95,000 malware pieces were analyzed by SophosLabs every day in 2010, nearly doubling the number of malware pieces tracked in 2009 (Sophos, 2011, p. 4). This is an extraordinary rate of growth.

Symantec publishes an annual report, *Symantic Internet Security Threat Report – Trends for 2010*, that presents some very alarming numbers. As summarized in this report, "Polymorphism and new delivery mechanisms such as Web-attack toolkits continued to drive up the number of malware variants in common circulation. In 2010, Symantec encountered more than 286 million unique variants of malware (Fossl, 2011, p. 6). The 2010 trend showed attacks through spear phishing, focusing on compromising specific organizations or individuals through social engineering techniques. There is a growing prevalence of Web-based attacks and the increased use of attack toolkits. "The Phoenix toolkit was responsible for the largest amount of Web-based attack activity in 2010. This kit, as well as many others, also incorporates exploits for Java vulnerabilities" (Fossl, 2011, p. 12).

Another report that epitomizes the current trend is from Blue Coat, *2011 Mid-Year Security Report*, and detailed in the Executive Overview, "The majority of web threats are now delivered from trusted and popular web sites that have been hacked for use by cybercrime. For this reason, reputation defenses become less effective. The once obscure link farm for search engine poisoning now resides within popular web sites. Phishing attacks overwhelmingly come from popular and trusted web sites hacked by cybercrime. Search engine Poisoning (SEP) ranks as the number one web threat delivery method at this point in the year" (Clare, 2011, p. 2). Also from their press release, "Web-based malware has become so dynamic that it is nearly impossible to protect every user from every new attack with traditional defenses." Continuing on, the leading malware delivery network, Shnakule, "had 2,000 unique host names per day with a peak of more than 4,300 per day. It is a broad-based malware delivery network whose malicious activities include drive-by downloads, fake anti-virus and codecs, fake flash and Firefox updates, fake warez, and botnet/command and controls" (Blue Coat, 2011).

Robert Sorensen, rssoren@gmail.com

In addition, a report from McAfee Labs, *McAfee Threats Report: Second Quarter 2011*, summarizes, "Malware continued its overall growth during the quarter as did rootkit malware. Rootkits, used primarily for stealth and resilience, makes malware more effective and persistent; its popularity is rising. The amount of malware that attacks vulnerabilities in Adobe products continues to overwhelm those in Microsoft products" (Dirro, Greve, Kashyap, Marcus, Paget, Schmugar, Shah, Wosotowsky, 2011, p. 2).

A common denominator to any malware delivery system is the human element. A quote from the book, *Information Security Management Handbook, Sixth Edition*, "It is well recognized that the greatest information security danger to any organization is not a particular process, technology, or equipment, rather it is the people who work within the "system" that hide the inherent danger" (Tipon, Krause, 2007, Ch. 43). No matter how secure we make our browsing environment, it still depends heavily on the human factor. Common sense must be prevalent in browsing habits and tendencies.

3. Introduction of VM

As current research has confirmed, the need of a secure browsing environment is very evident. Oracle VM VirtualBox, originally created by software company innotek GmbH, purchased by Sun Microsystems, and now owned by Oracle, is a general-purpose virtualization product for x86 and AMD64/intel64 hardware was selected as the virtualization environment (innotek GmbH). It is also the only professional solution that is freely available as Open Source Software under the terms of the GNU General Public License version 2 (VirtualBox).

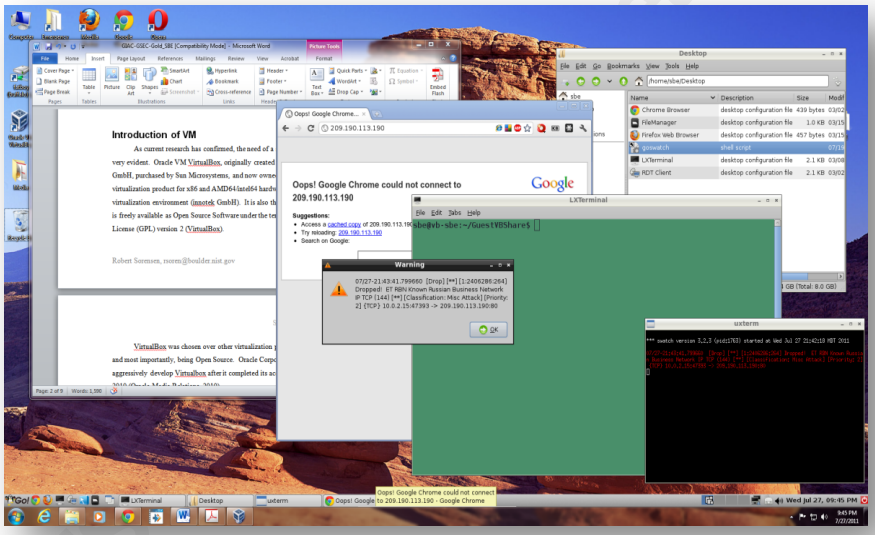
VirtualBox was chosen over other virtualization products due to its feature set, and most importantly, being Open Source. Oracle Corporation has continued to aggressively develop Virtualbox after it completed its acquisition of Sun Microsystems in 2010 (Oracle Media Relations, 2010).

VirtualBox provides a special software package, Guest Additions, which is designed to be installed inside the VM to improve performance of the guest OS and to add extra features. Guest Additions offers the following features:

- Mouse pointer integration which provides seamless mouse support.
- Shared folders provides an easy way to exchange files between the host and the

guest. VirtualBox makes this available to the guest operating system as a network share, even without networking support enabled.

- Better video support provides extra high and non-standard video modes as well as accelerated video performance. To further enhance video support, one can resize the virtual machine's window.
- Seamless windows suppresses the display of the Desktop background of the guest, allowing to run the windows of your guest operating system seamlessly next to the windows of your host. As shown in the screenshot below, SBE VM is running within Windows 7 environment with the LXPanel displayed just above the Windows task bar.



- Generic host/guest communication channels enables one to control and monitor guest execution.
- Time synchronization ensures the guest's system time is better synchronized with that of the host.
- Shared clipboard can be optionally shared with the host OS as well. (Introduction to Guest Additions, 2011, Chpt. 4).

3.1. SBE VM Objectives

The main objectives in researching and developing SBE VM are twofold: 1) Minimize, or ultimately, eliminate malware and/or virus infection in Windows operating systems by leveraging a Linux-based VM, and 2) Provide a bridge between the native OS and the virtual environment by the means of a shared directory. Introducing Windows users to a Linux-based browsing framework built using all Open-Source tools and custom scripts, is a critical element of this research paper.

4. Development of SBE VM

4.1. Linux wattOS Distribution Base

In conducting research for this project, many Linux distributions were considered. These included Aptosid, XPud, TinyLinux, CromeOS, Puppy Linux, Lubuntu, CrunchBang, Damn Small Linux, and SalixOS. Distrowatch.com was my main source of evaluating potential distribution where the top 100 Linux distributions are tracked and ranked (Distrowatch). Even though wattOS is ranked in the bottom third of the distributions according to Distrowatch, the description found on its web site is what set it apart from the many other distributions that were evaluated. "wattOS is a lightweight Linux operating system re-mastered from the core Ubuntu Linux build. It is a free operating system that focuses on a small footprint, low power, and a simple quick interface. Bring your old computer back to life again with a fresh install of wattOS!" (Baxter, 2011). In addition to meeting the performance objectives, the look and feel of wattOS felt right with the simple and efficient interface based on OpenBox (OpenBox) and LXDE (LXDE). When wattOS was initially installed on VirtualBox, it rocked! The only distribution found to have a faster boot time was ChromeOS, which lacked the ability to add custom security enhancements. Mainstream distributions like Ubuntu or Fedora were not even considered due to the bloated nature of them. All of the other light-weight distributions were eliminated for various aesthetic and/or performance issues.

A detailed step-by-step installation and configuration guide is included in Appendix A – SBE VM Step-by-Step Guide. It was felt that this detailed approach can

allow setup of their own SBE VM from scratch using the tools and techniques outlined in this guide. The following screenshot shows the desktop of the fully installed and configured SBE VM. Key applications have desktop shortcuts as well as panel shortcuts to facilitate ease of use and functionality of the environment.

4.2. Browser Security in Depth

Just like architecting a solution for the enterprise, one designs security in depth. This would include perimeter router with appropriate ACLs, firewalls, IDS/IPS, and anti-virus/anti-spam enterprise solution. Quoting an article concerning this, "No matter how good any single security application is, there is someone out there smarter than the people who designed it with more time on his hands than scruples who will eventually get past it. It is for this reason that common security practice suggests multiple lines of defense,

or in-depth security" (Bradley, 2009). This would allow the enterprise to attempt to protect and defend more vulnerable hosts. This is where SBE VM comes into play. It adds an additional layer on the client side, because we all know, no matter how good the enterprise defenses are, with targeted spear phishing or social engineering attacks, the host is still vulnerable.

In their book entitled, *Counter Hack Reloaded*, Ed Skoudis and Tom Liston points out browser vulnerabilities are discovered on a regular basis, especially (but not exclusively) in Internet Explorer. "Various types of browser holes, including buffer overflows, flaws that let an attacker escape the security restrictions on scripts or other active Web content (such as Java runtime environment), exploits that let malicious code bypass cryptographic signature checks, and problems that let malicious code execute in a different security zone than it should. All of these problem could be triggered if the victim surfs to the wrong Web site with a vulnerable browser" (Skoudis, and Liston, 2006, p. 432). The majority of these vulnerabilities exposes the Windows operating system, and thus, using SBE VM Linux-based OS, reduces the risk.

OpenDNS will be configured to provide an initial defense against phishing websites. Shared folders are established between host OS and SBE VM and all downloads are scanned for viruses using clamAV. In addition, Snort has been configured to act as a Host-based Intrusion Prevention System (HIPS). Finally, browser "add-ons" or "extensions" have been installed and configured to provide an additional front-line defense. These security features will now be discussed in more detail.

4.3. OpenDNS

OpenDNS is a free DNS server that typically outperforms local ISP DNS services due to the many millions of IP addresses of websites stored in their cache, thus taking less time to resolve a request. Another big advantage of using OpenDNS is that it blocks phishing websites from loading onto your computer. It uses data from Phishtank, a collaborative clearing house for data and information about phishing on the Internet (Phishtank). Another advantage of OpenDNS is that it addresses typos that one might make while entering a URL. For example, one might type "gogle.com." OpenDNS will open the main "google.com" web site automatically (Agarwal, 2008).

To configure OpenDNS in SBE VM, we first need to introduce a new network manager. The default manager that is installed in wattOS is 'wicd.' In order to configure OpenDNS, as well as have the network settings compatible with our later configuration of Snort in-line, it necessitates the change to 'network-manager' that is standard with a typical Ubuntu install. The details can be found in Appendix A – SBE VM Step-by-Step Guide.

4.4. VM Shared Folder - ClamAV – goinotify Script

To meet one of the main objectives of the project, there has to be a bridge tying the native OS with the virtual environment. This bridge is provided by the VirtualBox Shared Folders feature and can be established even without any network established. Much thought was given on how best to allow downloads from SBE VM back to the native OS in a safe manner. This was accomplished using a custom script and clamAV GPL open source project owned by SourceFire (clamAV).

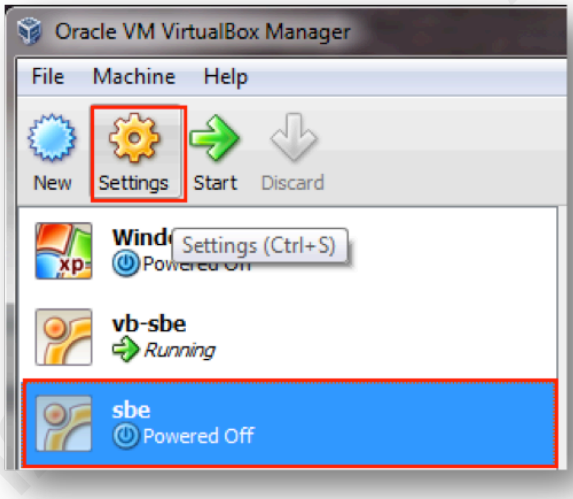
A script, goinotify, was developed to monitor and scan those files that are expressly downloaded. The core component of the script is the use of the event-monitoring built-in starting with the 2.6 Linux kernel. Inotifywait is part of the inotify-tools package, which efficiently waits for changes to files using Linux's inotify interface (inotify-tools). This was found to be a very efficient manner of validating downloads before moving them to the shared folder between the native OS and SBE VM.

We first need to establish the shared folder environment. A specific directory structure is created in the home directory of the SBE VM user, sbe. Next, clamAV and inotify-tools are installed.

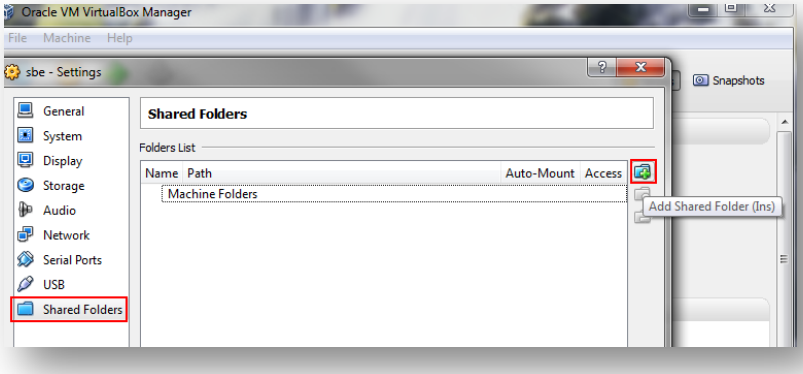
```
sbe@vb-sbe:~$ mkdir GuestVBShare
sbe@vb-sbe:~$ mkdir .infected
sbe@vb-sbe:~$ mkdir clamscan
sbe@vb-sbe:~$ touch clamscan/clamscan.log
sbe@vb-sbe:~$ sudo apt-get install inotify-tools clamav
[sudo] password for sbe:
Reading package lists... Done
Building dependency tree
Reading state information... Done
The following extra packages will be installed:
  clamav-base clamav-freshclam libclamav6 libtommath0
```

```
Suggested packages:
  clamav-docs libclamunrar6
The following NEW packages will be installed:
  clamav clamav-base clamav-freshclam inotify-tools libclamav6 libtommath0
0 upgraded, 6 newly installed, 0 to remove and 2 not upgraded.
Need to get 4,302 kB of archives.
After this operation, 12.1 MB of additional disk space will be used.
Do you want to continue [Y/n]? y
Setting up libclamav6 (0.97+dfsg-2ubuntu1.1) ...
Setting up clamav-base (0.97+dfsg-2ubuntu1.1) ...
Setting up clamav-freshclam (0.97+dfsg-ubuntu1.1) ...
 * Starting ClamAV virus database updater freshclam                [ OK ]
Setting up clamav (0.97+dfsg-2ubuntu1.1) ...
Setting up inotify-tools (3.13-3) ...
```

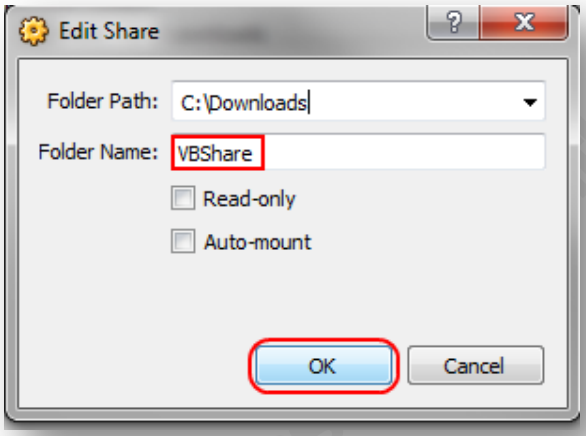
Now that clamAV and inotify-tools are installed and a directory structure created, VirtualBox shared folders will now be configured. As mentioned earlier, VirtualBox Guest Additions must be installed in order to establish this shared folder structure. With SBE VM shutdown, shared folders are configured.



- From Oracle VM VirtualBox Manager, highlight '**sbe**' VM and click '**Settings**'.

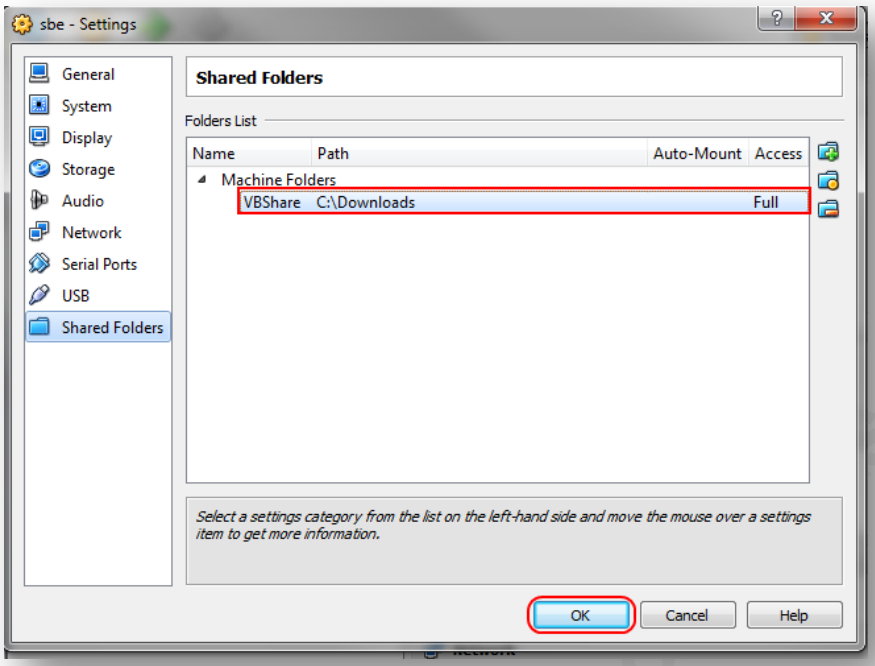


- Select '**Shared Folders**' on left, then **c**lick on '**Add Shared Folder**' icon.



- Select desired folder on host OS, typically, the '**Downloads**' folder is selected but any folder can be shared. For the Folder Name, it must be '**VBShare**' when the share is mounted at boot up.

Robert Sorensen, rssoren@gmail.com



- With a shared folder created, click '**OK**' to close settings dialogue.

In order to automatically mount the share upon boot up, the following entry must be made in /etc/rc.local, which runs at the end of all the multi-user boot levels. The mount type '-t vboxsf" is enabled by Guest Additions. Once the mount command is added, reboot the VM. The 'VBShare' will now be available and mounted as /home/sbe/GuestVBShare.

```
sbe@vb-sbe:~$ sudo vi /etc/rc.local
[sudo] password for sbe:

#!/bin/sh -e
#
# /etc/rc.local
#
# This script is executed at the end of each multiuser runlevel.
# Make sure that the script will "exit 0" on success or any other
# value on error.
#
# In order to enable or disable this script just change the execution
# bits.
#
# By default this script does nothing.

mount -t vboxsf -o uid=1000,gid=1000 VBShare /home/sbe/GuestVBShare/
```

Robert Sorensen, rssoren@gmail.com

```
/usr/local/bin/goinotify >/dev/null 2>&1

exit 0
```

```
sbe@vb-sbe:~$ sudo reboot

sbe@vb-sbe:~$ df –h
Filesystem      Size  Used Avail Use% Mounted on
/dev/sda1       6.0G  2.6G  3.1G  45% /
udev            997M  4.0K  997M   1% /dev
tmpfs           403M  760K  403M   1% /run
none            5.0M  4.0K  5.0M   1% /run/lock
none            1007M  4.0K 1007M   1% /run/shm
VBShare         922G  668G  255G  73% /home/sbe/GuestVBShare/
```

With the share now established, the 'goinotify' script will be created in /usr/local/bin.

```
sbe@vb-sbe:~$ sudo vi /usr/local/bin/goinotify
[Paste contents of /usr/local/bin/goinotify script]
sbe@vb-sbe:~$ sudo chmod 755 /usr/local/bin/goinotify
```

```
#!/bin/bash
# /usr/local/bin/goinotify

IFS=$'\n'
dl_dir=/home/sbe/Downloads
infect_dir=/home/sbe/.infected
share_dir=/home/sbe/GuestVBShare
log_file=/home/sbe/clamscan/clamscan.log

# Verify after copy/paste echo -e command is one line
dline () {
echo -e
"======================================================================" \ |
tee -a $log_file
}

# inotifywait command below is one line!!
inotifywait --monitor --exclude=".com.google.com.|.crdownload|.part" --
event moved_to,close_write --format '%:e@%f' "$dl_dir" | while read file;
do

file=`echo $file | awk -F@ ' { print $2 }'`

if [ "$file" = "" ]
then
```

Robert Sorensen, rssoren@gmail.com

```
   echo -e "Blank file..exiting...\n"
   exit 0
else
   if [ -f $dl_dir/$file ]
   then
      file_count=`ls $dl_dir/$file | egrep -v
'.com.google|.crdownload|.part' | wc -l`

      if [ "$file_count" = "0" ]
      then
         echo -e "Partial download..ignoring file $file...\n"
      else
         echo -e "Starting scan at `date`..." | tee -a $log_file
         echo -e "Verifying file '$file'..."

         clamscan --move=$infect_dir --no-summary --infected --bell --
log=$log_file $dl_dir/$file

         if [ -f $dl_dir/$file ]
         then
            echo -e "File '$file' was clean!  Moving to $share_dir..." |
tee -a $log_file
            mv $dl_dir/$file $share_dir
            dline
         else
            if [ -f $infect_dir/$file ]
            then
               echo -e "File '$file' was infected!  Moved to
$infect_dir..." | tee -a $log_file
            else
               echo -e "False alarm...\n"
            fi
            dline
         fi
      fi
   fi
fi
done
```

The line in the code, inotifywait, will be explained based on the parameters associated with the command.

```
inotifywait --monitor --exclude=".com.google.com.|.crdownload|.part" --event
moved_to,close_write --format '%:e@%f' "$dl_dir" | while read file; do
```

--monitor: Instead of exiting after receiving a single event, execute indefinitely.
--exclude <pattern>: Do Not process any events whose filename matches the specified POSI extended regular expressions, case sensitive.
--event <event>: Listen for specific event(s) only. Included here are

moved_to, close_write.
--format <fmt>: '%:e@%f'
%e – Replaced with the Event(s) which occurred, comma-separated.
@ - Delimiter
%f – When an event occurs within a directory, this will be replaced with the name of the File which caused the event to occur.

When a file download triggers inotifywait, the file name event is passed in the do loop of the script. Each file is then verified to ensure that it is a valid file name as both Chrome and Firefox uses temporary file names during download. Once a download is deemed complete, it is scanned by clamAV using the command that is described below:

```
clamscan --move=$infect_dir --no-summary --infected --bell --log=$log_file $dl_dir/$file
```

--move=Directory: Move any infected files to $infect_dir
--no-summary: Do not display summary at the end of scanning.
--infected: Only print infected files.
--bell: Sound bell on virus detection.
--log: Log to file defined in $log_file

$dl_dir/$file – file passed to clamscan from inotifywait event

After the file is processed by 'clamscan', if the file is still in the download directory, it was found clean and will then be moved to the shared folder to be available by the host OS.

4.5. Snort In-Line/Oinkmaster/Swatch alert scripts

The goal for SBE VM is to provide security in depth. The already obvious advantage of running in a Linux environment is only enhanced when a host-based intrusion prevention system like Snort in-line is incorporated. An excellent resource to build Snort in-line as provided by Phillip Bailey was used as a guide (Bailey, 2010).

The installation and configuration of Snort in-line will now be outlined. Some dependent packages are installed. Libdnet is downloaded from code.google.com. It provides a simplified, portable interface to several low-level networking routines, including network address manipulation, kernel arp cache, network firewalling, network interface lookup and manipulation, IP tunneling, and raw IP packet and Ethernet frame transmission (Libdnet).

Snort Data AcQuisition (DAQ) library is downloaded and compiled. The DAQ replaces direct calls into packet capture libraries like PCAT with an abstraction layer that make it easy to add additional software or hardware packet capture implementations (DAQ, 2010).

Finally, Snort source code is downloaded and compiled to support inline mode. Specific switches are included when running the './configure' command. One additional package, libzip-dev, was required to support compression.

```
sbe@vb-sbe:~$ mkdir temp; cd temp
sbe@vb-sbe:~/temp$ sudo apt-get install flex bison checkinstall libpcap0.8-dev
libnet1-dev libpcre3-dev libnetfilter-queue-dev iptables-dev
Reading package lists... Done
Building dependency tree
Reading state information... Done
The following extra packages will be installed:
  libnet1 libnetfilter-queue1 libnfnetlink-dev libpcrecpp0 m4 pkg-config
Suggested packages:
  bison-doc gettext
The following NEW packages will be installed:
  bison checkinstall flex iptables-dev libnet1 libnet1-dev libnetfilter-queue-dev
libnetfilter-queue1 libnfnetlink-dev
  libpcap0.8-dev libpcre3-dev libpcrecpp0 m4 pkg-config
0 upgraded, 14 newly installed, 0 to remove and 6 not upgraded.
Need to get 1,812 kB of archives.
After this operation, 6,722 kB of additional disk space will be used.
Do you want to continue [Y/n]? y

sbe@vb-sbe:~/temp$ wget http://libdnet.googlecode.com/files/libdnet-1.12.tgz
sbe@vb-sbe:~/temp$ tar xvfz libdnet-1.12.tgz
sbe@vb-sbe:~/temp/libdnet-1.12$ cd libdnet-1.12/
sbe@vb-sbe:~/temp/libdnet-1.12$ ./configure CFLAGS="-fPIC -g –O2"
sbe@vb-sbe:~/temp/libdnet-1.12$ make
sbe@vb-sbe:~/temp/libdnet-1.12$ sudo checkinstall
[sudo] password for sbe:

checkinstall 1.6.2, Copyright 2009 Felipe Eduardo Sanchez Diaz Duran
        This software is released under the GNU GPL.

The package documentation directory ./doc-pak does not exist.
Should I create a default set of package docs?  [y]: ENTER

Preparing package documentation...OK
```

Robert Sorensen, rssoren@gmail.com

```
Please write a description for the package.
End your description with an empty line or EOF.
>> ENTER
*****************************************
**** Debian package creation selected ***
*****************************************

This package will be built according to these values:

0 -  Maintainer: [ root@vb-sbe ]
1 -  Summary: [ Package created with checkinstall 1.6.2 ]
2 -  Name:    [ libdnet ]
3 -  Version: [ 1.12 ]
4 -  Release: [ 1 ]
5 -  License: [ GPL ]
6 -  Group:   [ checkinstall ]
7 -  Architecture: [ i386 ]
8 -  Source location: [ libdnet-1.12 ]
9 -  Alternate source location: [  ]
10 - Requires: [  ]
11 - Provides: [ libdnet ]
12 - Conflicts: [  ]
13 - Replaces: [  ]

Enter a number to change any of them or press ENTER to continue: ENTER

Installing with make install...

*********************************************************************
 Done. The new package has been installed and saved to
 /home/sbe/temp/libdnet-1.12/libdnet_1.12-1_i386.deb

 You can remove it from your system anytime using:

     dpkg -r libdnet
*********************************************************************
sbe@vb-sbe:~/temp/libdnet-1.12$ sudo dpkg -i libdnet_1.12-1_i386.deb
sbe@vb-sbe:~/temp/libdnet-1.12$ sudo ln -s /usr/local/lib/libdnet.1.0.1
/usr/lib/libdnet.1
sbe@vb-sbe:~/libdnet-1.12$ cd ..
sbe@vb-sbe:~/temp$ wget http://www.snort.org/downloads/1525
sbe@vb-sbe:~/temp$ tar xvfz 1525
sbe@vb-sbe:~/temp$ cd daq-0.6.2
sbe@vb-sbe:~/daq-0.6.2$ ./configure

Build AFPacket DAQ module.. : yes
```

Robert Sorensen, rssoren@gmail.com

```
Build Dump DAQ module..... : yes
Build IPFW DAQ module..... : yes
Build IPQ DAQ module..... : yes
Build NFQ DAQ module..... : yes
Build PCAP DAQ module..... : yes

sbe@vb-sbe:~/temp/daq-0.5$ make
sbe@vb-sbe:~/temp/daq-0.5$ sudo checkinstall
[sudo] password for sbe:

checkinstall 1.6.2, Copyright 2009 Felipe Eduardo Sanchez Diaz Duran
        This software is released under the GNU GPL.

The package documentation directory ./doc-pak does not exist.
Should I create a default set of package docs?  [y]: ENTER

Preparing package documentation...OK

Please write a description for the package.
End your description with an empty line or EOF.
>> ENTER
*****************************************
**** Debian package creation selected ***
*****************************************

This package will be built according to these values:

0 -  Maintainer: [ root@vb-sbe ]
1 -  Summary: [ Package created with checkinstall 1.6.2 ]
2 -  Name:    [ daq ]
3 -  Version: [ 0.6.2 ]
4 -  Release: [ 1 ]
5 -  License: [ GPL ]
6 -  Group:   [ checkinstall ]
7 -  Architecture: [ i386 ]
8 -  Source location: [ daq-0.6.2 ]
9 -  Alternate source location: [  ]
10 - Requires: [  ]
11 - Provides: [ daq ]
12 - Conflicts: [  ]
13 - Replaces: [  ]

Enter a number to change any of them or press ENTER to continue: ENTER

Installing with make install...
```

Robert Sorensen, rssoren@gmail.com

```
sbe@vb-sbe:~/temp/daq-0.6.2$ sudo dpkg -i daq_0.6.2-1_i386.deb
(Reading database ... 132264 files and directories currently installed.)
Preparing to replace daq 0.6.2-1 (using daq_0.6.2-1_i386.deb) ...
Unpacking replacement daq ...
Setting up daq (0.6.2-1) ...

sbe@vb-sbe:~/temp$  cd ..
sbe@vb-sbe:~/temp$  sudo apt-get install libzip-dev

sbe@vb-sbe:~/temp$ wget http://www.snort.org/downloads/1538
sbe@vb-sbe:~/temp$ tar xvfz 1538
sbe@vb-sbe:~/temp$ cd snort-2.9.2.2
sbe@vb-sbe:~/temp/snort-2.9.0.5$ ./configure --enable-build-dynamic-examples --
enable-gre --enable-reload --enable-linux-smp-stats --enable-zlib --enable-active-
response --enable-react --enable-flexresp3 --enable-ipv6
[Note:  Best to type in the ./configure options!  Copy/paste may not work properly.]
sbe@vb-sbe:~/temp/snort-2.9.2.2$ make
sbe@vb-sbe:~/temp/snort-2.9.2.2$ sudo make install

sbe@vb-sbe:~/temp/snort-2.9.2.2$ cd ..
sbe@vb-sbe:~/temp$ sudo mv snort-2.9.2.2/ /root
sbe@vb-sbe:~/temp$ sudo bash
root@vb-sbe:~/temp# cd /root/snort-2.9.2.2/
root@vb-sbe:/root/snort-2.9.2.2# checkinstall
checkinstall 1.6.2, Copyright 2009 Felipe Eduardo Sanchez Diaz Duran
        This software is released under the GNU GPL.

The package documentation directory ./doc-pak does not exist.
Should I create a default set of package docs?  [y]: ENTER

Preparing package documentation...OK

Please write a description for the package.
End your description with an empty line or EOF.
>> ENTER

*****************************************
**** Debian package creation selected ***
*****************************************

This package will be built according to these values:

0 -  Maintainer: [ root@vb-sbe ]
1 -  Summary: [ Package created with checkinstall 1.6.2 ]
2 -  Name:    [ snort ]
3 -  Version: [ 2.9.2.2 ]
```

Robert Sorensen, rssoren@gmail.com

```
4 -  Release: [ 1 ]
5 -  License: [ GPL ]
6 -  Group:   [ checkinstall ]
7 -  Architecture: [ i386 ]
8 -  Source location: [ snort-2.9.2.2 ]
9 -  Alternate source location: [  ]
10 - Requires: [  ]
11 - Provides: [ snort ]
12 - Conflicts: [  ]
13 - Replaces: [  ]

Enter a number to change any of them or press ENTER to continue:  ENTER

Installing with make install...

****************************************************************************
 Done. The new package has been installed and saved to
 /root/snort-2.9.2.2/snort_2.9.2.2-1_i386.deb

 You can remove it from your system anytime using:

      dpkg -r snort
****************************************************************************
root@vb-sbe:/root/snort-2.9.2.2# make uninstall
root@vb-sbe:/root/snort-2.9.2.2# dpkg -i snort_2.9.2.2-1_i386.deb
root@vb-sbe:/root/snort-2.9.2.2# mkdir –p /var/log/snort
root@vb-sbe:/root/snort-2.9.2.2# mkdir –p /etc/snort/rules
root@vb-sbe:/root/snort-2.9.2.2# cp etc/* /etc/snort
root@vb-sbe:/root/snort-2.9.2.2# touch /var/log/snort/alert

root@vb-sbe:/root/snort-2.9.2.2# snort -V

   ,,_     -*> Snort! <*-
  o"  )~   Version 2.9.2.2 GRE (Build 121)
   ""      By Martin Roesch & The Snort Team: http://www.snort.org/snort/snort-team
           Copyright (C) 1998-2012 Sourcefire, Inc., et al.
           Using libpcap version 1.1.1
           Using PCRE version: 8.12 2011-01-15
           Using ZLIB version: 1.2.3.4
```

Due to the fact that the libdnet source was downloaded and compiled specifically for Snort, we must guarantee that no Ubuntu updates for this library are applied. If libdnet is ever updated via apt-get, it will break Snort. In addition, 'libdbus-1-3' will be withheld as well to prevent a similar situation.

Robert Sorensen, rssoren@gmail.com

```
sbe@vb-sbe:~$ sudo bash
[sudo] password for sbe:
root@vb-sbe:~# echo "libdnet hold" | dpkg --set-selections
root@vb-sbe:~# echo "libdbus-1-3 hold" | dpkg --set-selections
root@vb-sbe:~# apt-get upgrade
Reading package lists... Done
Building dependency tree
Reading state information... Done
The following packages have been kept back:
  libdnet
The following packages will be upgraded:
  ecryptfs-utils firefox firefox-globalmenu flashplugin-installer
  foomatic-filters gnome-keyring google-chrome-stable gvfs gvfs-backends
  gvfs-fuse libecryptfs0 libgcr0 libgp11-0 libgvfscommon0
  libpam-gnome-keyring linux-libc-dev udisks xserver-common xserver-xorg-core
19 upgraded, 0 newly installed, 0 to remove and 1 not upgraded.
Need to get 48.9 MB of archives.
After this operation, 152 kB of additional disk space will be used.
Do you want to continue [Y/n]? y
...
Setting up firefox-globalmenu (6.0.1+build1+nobinonly-0ubuntu0.11.04.1) ...
sbe@vb-sbe:~$
```

Once Snort is installed and verified, configuration is the next critical step. Also, oinkmaster will be installed and configured to support the ease of updating the emerging-threat rule base that is used in SBE VM.

sbe@vb-sbe:~$ **sudo vi /etc/snort/snort.conf**

Key Variables to Modify

Line	Original Variable	Modified Variable
45	ipvar HOME_NET any	ipvar HOME_NET **10.0.2.15/32**
48	Ipvar EXTERNAL_NET any	ipvar EXTERNAL_NET **!$HOME_NET**
80	var RULE_PATH ../rules	var RULE_PATH **rules**
159-161	# config daq: <type> # config daq_dir: <dir> # config daq_mode: <mode>	**config daq: nfq** **config daq_dir: /usr/local/lib/daq** **config daq_mode: inline**
265-269	preprocessor normalize_ip4 preprocessor normalize_tcp: ips ecn stream preprocessor normalize_icmp4 preprocessor normalize_ip6 preprocessor	#preprocessor normalize_ip4 #preprocessor normalize_tcp: ips ecn stream #preprocessor normalize_icmp4 #preprocessor normalize_ip6 #preprocessor normalize_icmp6

	normalize_icmp6	
273	preprocessor frag3_engine: policy windows detect_anomalies overlap_limit 10 min_fragment_length 100 timeout 180	preprocessor frag3_engine: policy **linux** detect_anomalies overlap_limit 10 min_fragment_length 100 timeout 180
283	preprocessor stream5_tcp: policy windows, detect_anomalies, require_3whs 180, \	preprocessor stream5_tcp: policy **linux**, detect_anomalies, require_3whs 180, \
297	preprocessor http_inspect: global iis_unicode_map unicode.map 1252 compress_depth 65535 decompress_depth 65535	preprocessor http_inspect: global iis_unicode_map **./rules/**unicode.map 1252 compress_depth 65535 decompress_depth 65535
506-511 [Comment out]	preprocessor reputation: \ memcap 500, \ priority whitelist, \ nested_ip inner, \ whitelist $WHITE_LIST_PATH/white_list.rules, \ blacklist $BLACK_LIST_PATH/black_list.rules	#preprocessor reputation: \ # memcap 500, \ # priority whitelist, \ # nested_ip inner, \ whitelist #$WHITE_LIST_PATH/white_list.rules, \ blacklist #$BLACK_LIST_PATH/black_list.rules
540	include classification.config	include **rules/**classification.config
541	include reference.conf	include **rules/**reference.conf
552	include $RULE_PATH/local.rules	include $RULE_PATH/local.rules **include emerging.conf**
554-607 [Comment out rules]	include $RULE_PATH/attack-responses.rules … include $RULE_PATH/x11.rules	#include $RULE_PATH/attack-responses.rules … #include $RULE_PATH/x11.rules

Refer to 'Appendix B – Configuration Files', for full version of /etc/snort.conf.

We next need to download and create the /etc/snort/rules/emerging.conf file.

```
sbe@vb-sbe:~$ sudo bash
root@vb-sbe:~# cd /etc/snort
root@vb-sbe:/etc/snort# wget http://rules.emergingthreats.net/open/snort-
2.9.0/emerging.conf
--2011-09-13 20:26:37--  http://rules.emergingthreats.net/open/snort-2.9.0/emerging.conf
Resolving rules.emergingthreats.net... 69.195.137.28, 216.40.222.19
Connecting to rules.emergingthreats.net|69.195.137.28|:80... connected.
HTTP request sent, awaiting response... 200 OK
Length: 3543 (3.5K) [text/plain]
Saving to: `emerging.conf'

100%[======================================>] 3,543       --.-K/s   in 0.04s

2011-09-13 20:26:37 (85.9 KB/s) - `emerging.conf' saved [3543/3543]

root@vb-sbe:/etc/snort# vi emerging.conf
```

Uncomment the following rules:

```
include $RULE_PATH/emerging-policy.rules
include $RULE_PATH/emerging-trojan.rules
include $RULE_PATH/emerging-games.rules
include $RULE_PATH/emerging-user_agents.rules
include $RULE_PATH/emerging-activex.rules
include $RULE_PATH/emerging-virus.rules
include $RULE_PATH/emerging-attack_response.rules
include $RULE_PATH/emerging-icmp.rules
include $RULE_PATH/emerging-icmp_info.rules
include $RULE_PATH/emerging-shellcode.rules
include $RULE_PATH/emerging-web_client.rules
include $RULE_PATH/emerging-current_events.rules
include $RULE_PATH/emerging-inappropriate.rules
include $RULE_PATH/emerging-deleted.rules
include $RULE_PATH/emerging-malware.rules
include $RULE_PATH/emerging-worm.rules
include $RULE_PATH/emerging-dns.rules
include $RULE_PATH/emerging-misc.rules
include $RULE_PATH/emerging-dos.rules
include $RULE_PATH/emerging-telnet.rules
include $RULE_PATH/emerging-exploit.rules
include $RULE_PATH/emerging-p2p.rules
include $RULE_PATH/emerging-tftp.rules
include $RULE_PATH/emerging-mobile_malware.rules
include $RULE_PATH/emerging-botcc.rules
include $RULE_PATH/emerging-compromised.rules
include $RULE_PATH/emerging-drop.rules
include $RULE_PATH/emerging-dshield.rules
include $RULE_PATH/emerging-rbn.rules
```

Robert Sorensen, rssoren@gmail.com

```
include $RULE_PATH/emerging-rbn-malvertisers.rules
include $RULE_PATH/emerging-tor.rules
include $RULE_PATH/emerging-ciarmy.rules
```

Refer to 'Appendix B – Configuration Files', for full version of /etc/snort/emerging.conf.

Create local.rules by copy/pasting into file as shown below.

```
sbe@vb-sbe:~$ sudo vi /etc/snort/rules/local.rules
[Copy/paste rules below.]

#/etc/snort/rules/local.rules
#drop icmp any any -> any any (msg:"ICMP Packet Dropped!"; sid:100001; rev:3;)
alert icmp any any -> any any (msg:"ICMP Packet Allowed"; sid:100001; rev:3;)
```

```
sbe@vb-sbe:~$ sudo apt-get install oinkmaster
Reading package lists... Done
Building dependency tree
Reading state information... Done
The following NEW packages will be installed:
  oinkmaster
0 upgraded, 1 newly installed, 0 to remove and 3 not upgraded.
Need to get 0 B/93.1 kB of archives.
After this operation, 340 kB of additional disk space will be used.
Selecting previously deselected package oinkmaster.
(Reading database ... 115527 files and directories currently installed.)
Unpacking oinkmaster (from .../oinkmaster_2.0-3_all.deb) ...
Processing triggers for man-db ...
Setting up oinkmaster (2.0-3) ...
```

Oinkmaster is configured to pull and configure the Emerging Threats open community ruleset. They are considered the fastest moving and most diverse Snort rulesets and firewall rules available (Emerging Threats). Emerging Threats exists because of the contributors of intelligence and signatures by the community. Updates are provided on a daily basis. Oinkmaster runs every time the system is booted to ensure that the rules are current and up to date. A script, gooinkmaster, was written to facilitate this process and can be run at any point to update the rulesets. Another script, pop, is used to capture the process ID of the active Snort process.

Robert Sorensen, rssoren@gmail.com

```
#!/bin/bash
#/usr/local/bin/gooinkmaster

echo -e "Updating Snort rule set...\n"

oinkmaster -o /etc/snort/rules/
snort_pid=`/usr/local/bin/pop snort.conf | awk ' { print $2 }'`

echo -e "Bumping Snort to reload updated rule set...\n"
echo -e "kill -HUP $snort_pid...\n"

kill -HUP $snort_pid
```

```
#!/bin/bash
#/usr/local/bin/pop

ps aux | egrep -i $1 | egrep -v "pop|grep"
```

Create the scripts by copy/pasting into files as shown below.

```
sbe@vb-sbe:~$ sudo vi /usr/local/bin/gooinkmaster
[sudo] password for sbe:
sbe@vb-sbe:~$ sudo chmod 755 /usr/local/bin/gooinkmaster
sbe@vb-sbe:~$ sudo vi /usr/local/bin/pop
sbe@vb-sbe:~$ sudo chmod 755 /usr/local/bin/pop
```

One of the key features enabled in the /etc/oinkmaster.conf file is the ability to modify rules after they have been downloaded. An example of the syntax is taken from the configuration file, modifysid * “replacethis” | “withthis.” Two directives have been implemented in SBE VM:

- modifysid * "^alert" | "reject"
- modifysid * "msg:\"" | "msg:\"Rejected! "
- modifysid emerging-rbn.rules "(.+) \[(.+)\] (.+) \$HOME_NET (.+)" | "${1} \$HOME_NET ${3} \[${2}\] ${4}"
- modifysid emerging-rbn-malvertisers.rules "(.+) tcp (.+) any \-\> \$HOME_NET (.+)" | "${1} tcp \$HOME_NET any \-\> ${2} ${3}"
- modifysid emerging-rbn-malvertisers.rules "(.+) udp (.+) any \-\> \$HOME_NET (.+)" | "${1} udp \$HOME_NET any \-\> ${2} ${3}"

The first directive, changes each rule to have the action "reject" instead of just "alert". 'Reject – remove the packet from the wire and return an ICMP "Communication administratively prohibited" (ICMP type 3, code 13) error packet' was chosen over 'drop – remove the packet from the wire and generate no error packet' to reduce retransmission time and provide the user with immediate results. An excellent comparison of reject verses drop is provided by Peter Benie and his conclusions summaries the reason why 'reject' was chosen, "Drop offers no effective barrier to hostile forces but can dramatically slow down applications run by legitimate users. Drop should not normally be used" (Benie).

The second directive changes the Snort alert to annotate in the alert that the packet was rejected. Swatch is configured to watch for 'Rejected!' and alert accordingly.

The third directive changes the emerging-rbn and emerging-rbn-malvertisers rules to switch the src/dst to trigger any connections to known Russian Business Networks.

In addition to the directive changes, two SIDs will be disabled. They are associated with the Ubuntu apt-get update/upgrade cycle and are SIDs 2013504 and 1390. Snort was preventing access to the updates due to a potential policy violation. As this is a functionality that must exist in SBE VM, steps were implemented in the /etc/oinkmaster.conf file to ensure these SIDs are disabled.

It is straightforward to disable specific SIDs in oinkmaster. Based on the guidance provided in the /etc/oinkmaster.conf file, SIDs to comment out, i.e. disable, after each update is done by placing a '#' in front of the rule. Syntax: disablesid SID. The following entries were added at the end of this section in order to disable these specific rules.

- disablesid 1390 [Generic GPL SHELLCODE NOOP (False Positive)]
- disabledsid 2013504 [Allow apt-get updates for Ubuntu]

sbe@vb-sbe:~$ **sudo vi /etc/oinkmaster.conf**

Key Variables to Modify

Line	Original Value	Modified Value
46	#Snort site in your registered user profile.	#Snort site in your registered user profile. **url =** **/http://rules.emergingthreats.net/open/snort-2.9.0/emerging.rules.tar.gz**
235	skipfile snort.conf	skipfile snort.conf **skipfile emerging.conf**
244	# skipfile threshold.conf	**skipfile threshold.conf**
383	# modifysid 100 “foo” | “bar”	# modifysid 100 “foo” | “bar” **modifysid * "^alert" | "reject"** **modifysid * "msg:\"" | "msg:\"Rejected! "** **modifysid emerging-rbn.rules ** **"(.+) \[(.+)\] (.+) \$HOME_NET (.+)" \ ** **"${1} \$HOME_NET ${3} \[${2}\] ${4}"** **modifysid emerging-rbn-malvertisers.rules ** **"(.+) tcp (.+) any \-\> \$HOME_NET (.+)" \ ** **"${1} tcp \$HOME_NET any \-\> ${2} ${3}"** **modifysid emerging-rbn-malvertisers.rules ** **"(.+) udp (.+) any \-\> \$HOME_NET (.+)" \ ** **"${1} udp \$HOME_NET any \-\> ${2} ${3}"**
453	# disablesid 4,5,6	# disablesid 4,5,6 **disablesid 1390,2013504**

Refer to ‘Appendix B – Configuration Files’, for full version of /etc/oinkmaster.conf.

```
sbe@vb-sbe:~$ sudo gooinkmaster
Updating Snort rule set...

Loading /etc/oinkmaster.conf
Downloading file from http://rules.emergingthreats.net/open/snort-
 2.9.0/emerging.rules.tar.gz... done.
Archive successfully downloaded, unpacking... done.
Setting up rules structures... done.
Processing downloaded rules... disablesid 2, enablesid 0, modifysid 29022, localsid 0,
 total rules 15831
Setting up rules structures... done.
Comparing new files to the old ones... done.
```

Robert Sorensen, rssoren@gmail.com

```
Checking flowbits dependencies... problems found:

WARNING: SID 2012816 depends on flowbit "ET.http.binary" which is set in
 INACTIVE SID 2000427 (SID 2012816 is broken unless you also enable SID 2000427).
WARNING: SID 2001984 depends on flowbit "is_proto_ssh" which is set in INACTIVE
 SID 2001983 (SID 2001984 is broken unless you also enable SID 2001983).

[***] Results from Oinkmaster started 20110902 10:00:58 [***]

[*] Rules modifications: [*]
   None.

[*] Non-rule line modifications: [*]
   None.

[*] Added files: [*]
   None.

Bumping Snort to reload updated rule set...
kill -HUP 1422...
```

The way Snort and oinkmaster starts upon reboot is slightly different due to the fact that a plumbed interface is required before Snort can be activated. With the way Snort is configured to be in-line, it must be running in order to pass any traffic. After much trial and error in attempting to get Snort and oinkmaster to start at boot time, it was discovered the best way was to incorporate the startup script in /etc/network/if-up.d/upstart configuration file.

```
sbe@vb-sbe:~$ sudo vi /etc/network/if-up.d/upstart [Modify file]
sbe@vb-sbe:~$ sudo vi /usr/local/bin/goSNORT [Paste in contents script below]
sbe@vb-sbe:~$ sudo vi /usr/local/bin/gosnort [Paste in contents script below]
sbe@vb-sbe:~$ sudo chmod 755 /usr/local/bin/gosnort /usr/loal/bin/goSNORT
sbe@vb-sbe:~$
```

```
#!/bin/sh
#/etc/network/if-up.d/upstart
MARK_DEV_PREFIX="/run/network/ifup."
MARK_STATIC_NETWORK_EMITTED="/run/network/static-network-up-emitted"

set -e

# lo emission handled by /etc/init/network-interface.conf
if [ "$IFACE" != lo ]; then
```

Robert Sorensen, rssoren@gmail.com

```
    initctl emit -n net-device-up \
        "IFACE=$IFACE" \
        "LOGICAL=$LOGICAL" \
        "ADDRFAM=$ADDRFAM" \
        "METHOD=$METHOD"

     /usr/local/bin/gosnort >/dev/null 2>&1

#Run oinkmaster once we know that network and Snort are running
     /usr/local/bin/gooinkmaster >/dev/null 2>&1
fi
```

```
#!/bin/bash
#/usr/local/bin/goSNORT

sudo /usr/local/bin/gosnort
```

```
#!/bin/bash
#/usr/local/bin/gosnort

#Check to see if snort is started
count=`/usr/local/bin/pop snort.conf | wc -l`

if [ $count -eq 0 ]
then
   iptables -A INPUT -i lo -j ACCEPT
   iptables -A INPUT -j QUEUE
   iptables -A OUTPUT -j QUEUE
   iptables -A FORWARD -j QUEUE

   #Update current IP address for HOME_NET in /etc/snort/snort.conf
   tmpip=`ifconfig | egrep -A1 "eth|wlan" | grep "inet addr" | awk ' {
print $2 }' | awk -F: ' { print $2 }'| head -1`

   if [ "$tmpip" = "" ]
   then
      echo -e "Sorry, no interfaces are up...\n Please configure
interface and run 'sudo gosnort'"
   else
      echo -e "Yes!  Interface is up.  Using $tmpip...\n"

      sed -i -e "s/ipvar HOME_NET [0-9]\{1,3\}\.[0-9]\{1,3\}\.[0-
9]\{1,3\}\.[0-9]\{1,3\}/ipvar HOME_NET ${tmpip}/" /etc/snort/snort.conf

      #Snort must be running in order to pass traffic.
      snort -D -N -c /etc/snort/snort.conf -A fast
   fi
else
   zenity --timeout=3 --info --text="Snort is already running...\n"
   echo -e "Snort is already running.  Exiting...\n"
   exit 1
fi
```

Breaking down the iptables entries above will provide an explanation of how iptables are configured. This pushes all traffic going in, out, and through the machine into IP queue from which Snort in-line will read its packet instead of using libpcap. Andrew Lockhart in his book, "Network Security Hacks, Second Edition", describes this concept in excellent detail (Lockhart, 2007, p. 380).

```
iptables -A INPUT -i lo -j ACCEPT
iptables -A INPUT -j QUEUE
iptables -A OUTPUT -j QUEUE
iptables –A FORWARD –j QUEUE
```

-A: Append one or more rules to end of the selected chain.
INPUT: For packets coming into the box itself
OUTPUT: Packets generated by local processes, i.e. Snort.
FORWARD: Packets being routed through the box.
-i lo –j ACCEPT: Pass (ACCEPT) all traffic form local interface
-j: Jump to target. This specifies the target of the rule, what to do if the packet matches it.
QUEUE: Means to pass the packet to userspace, i.e., Snort.

Breaking down the snort command and switches.

```
snort -D -N -c /etc/snort/snort.conf -A fast
```

-D: Run Snort in daemon mode. Alerts are sent to /var/log/snort/alert unless otherwise specified.
-N: Turn off packet logging. The program still generates alerts normally.
-c /etc/snort/snort.conf
-A fast: Alert mode. Fast writes alerts to the default "alert" file in a single-line, syslog style alert message.

Additional memory needs to be allocated for certain memory buffers to avoid errors like "*packet recv contents failure: No buffer space available.*" The following settings increase the buffer that NFQ uses for its queue (Ristic, 2008).

```
sbe@vb-sbe:~$ sudo bash
[sudo] password for sbe:
root@vb-sbe:~# echo -e "net.core.rmem_default = 4194304\nnet.core.wmem_default
 = 4194304\n\nnet.ipv4.tcp_wmem = 1048576 4194304 16777216\nnet.ipv4.tcp_rmem
 = 1048576 4194304 16777216" >>/etc/sysctl.conf
root@vb-sbe:~# sysctl -p /etc/sysctl.conf
```

Robert Sorensen, rssoren@gmail.com

```
net.core.rmem_default = 4194304
net.core.wmem_default = 4194304
net.ipv4.tcp_wmem = 1048576 4194304 16777216
net.ipv4.tcp_rmem = 1048576 4194304 16777216
```

The last piece of the Snort puzzle will be to implement Swatch, which is designed to monitor system activity. There is so much flexibility associated with configuring Swatch that we can easily monitor Snort alert logs for the predefined 'Rejected!' message. Swatch and wmctrl, which is a command that can be used to interact with an X Window manager, are installed as a prerequisite.

```
sbe@vb-sbe:~$ sudo apt-get install swatch wmctrl
Reading package lists... Done
Building dependency tree
Reading state information... Done
The following extra packages will be installed:
  libbit-vector-perl libcarp-clan-perl libdate-calc-perl libdate-manip-perl
  libfile-tail-perl libyaml-syck-perl
The following NEW packages will be installed:
  libbit-vector-perl libcarp-clan-perl libdate-calc-perl libdate-manip-perl
  libfile-tail-perl libyaml-syck-perl swatch wmctrl
0 upgraded, 8 newly installed, 0 to remove and 6 not upgraded.
Need to get 3,302 kB of archives.
After this operation, 17.6 MB of additional disk space will be used.
Do you want to continue [Y/n]? y
Setting up libcarp-clan-perl (6.04-1) ...
Setting up libbit-vector-perl (7.1-1build1) ...
Setting up libdate-calc-perl (6.0-2build1) ...
Setting up libyaml-syck-perl (1.17-1build1) ...
Setting up libdate-manip-perl (6.24-1) ...
Setting up libfile-tail-perl (0.99.3-4) ...
Setting up swatch (3.2.3-1) ...
Setting up wmctrl (1.07-6) ...
sbe@vb-sbe:~$ sudo apt-get install roxterm
Reading package lists... Done
Building dependency tree
Reading state information... Done
The following NEW packages will be installed:
  roxterm
0 upgraded, 1 newly installed, 0 to remove and 6 not upgraded.
Need to get 0 B/210 kB of archives.
After this operation, 975 kB of additional disk space will be used.
Setting up roxterm (1.22.2-1) ...
```

Robert Sorensen, rssoren@gmail.com

The '/etc/swatchrc' configuration file contains a pattern to watch for and an action to perform when the pattern is found. Much effort went into developing the right pattern and syntax to fire off a zenity alert pop-up message.

#/etc/swatchrc

```
#Snort Alerts
watchfor /Rejected!/
echo=red
exec echo "$_ | /usr/bin/sed 's/(/\\\(/' | /usr/bin/sed 's/)/\\\)/'" | /usr/bin/zenity --warning --
text "$_"

#ClamAV Virus Alerts
watchfor /FOUND/
echo=red
exec echo "$_ | /usr/bin/sed 's/(/\\\(/' | /usr/bin/sed 's/)/\\\)/'" | /usr/bin/zenity --warning --
text "$_"
```

watchfor - regex pattern
echo=red – Echo the matched line. The text mode is set for red.
exec – Command to execute when /Rejected!/ is found. The command may contain variables which are substituted with fields from the matched line. A $_ will be replaced by the entire line. The sed command cleans up extra () that aren't compatible with the zenity --warning command.
zenity --warning – Used to display a warning dialogue pop-up.

A script is set to activate the real-time alerting of Snort.

```
#!/bin/bash
#/usr/local/bin/goswatch
count=0
roxsession=0  #Default=0
#0=kill roxterm sessions (Alerts will display in current Workspace).
#1=roxterm sessions opened in Workspace 2 (Alerts always displayed in
Workspace 2).

#Verify that inotifywait is started

icount=`/usr/local/bin/pop inotifywait | wc -l`
if [ $icount -eq 0 ]
then
   /usr/local/bin/goinotify >/dev/null 2>&1
else
   echo -e "inotifywait watch already running...\n"
fi

count=`/usr/local/bin/pop swatch_script | wc -l`

if [ $count -eq 0 ]
```

Robert Sorensen, rssoren@gmail.com

```
then
   echo -e "Starting Swatch in Workspace 2...\n"
   wmctrl -s1
# roxterm command must be one line!
   roxterm --geometry=100x30 --color-scheme=Tango --hide-menubar -T
"Snort - Monitoring /var/log/snort/alert..." -e swatch -c /etc/swatchrc
-t /var/log/snort/alert  &
   roxterm --geometry=100x30 --color-scheme=Tango --hide-menubar -T
"Clamscan - Monitoring /home/sbe/clamscan/clamscan.log..." -e swatch -c
/etc/swatchrc -t /home/sbe/clamscan/clamscan.log  &
   sleep 1
   wmctrl -s0
   if [ $roxsession -eq 0 ]
   then
      zenity --window-icon=/home/sbe/Pictures/images/snort_swatch.png -
-title="Swatch Watch" --info --text="Swatch launched and is monitoring
for Snort and ClamAV virus alerts.  \nA Pop-up warning will be
displayed when a file is infected or when traffic is rejected\!"
      pkill roxterm
   else
      zenity --window-icon=/home/sbe/Pictures/images/snort_swatch.png -
-title="Swatch Watch" --info --text="Swatch launched in Workspace 2
monitoring for Snort and ClamAV virus alerts.  \nA Pop-up warning will
be displayed when a file is infected or when traffic is rejected\!"
   fi
else
   zenity --timeout=3 --info --text="Swatch is already running...\n"
   echo -e "Swatch is already running.  Exiting...\n"
   exit 1
fi
```

Create the scripts by copy/pasting into files as shown below. Verify that the longer lines in the scripts not broken up into separate lines when performing the copy/paste function. Also, create Desktop shortcuts to launch 'goSWATCH, and 'GOsnort'.

sbe@vb-sbe:~$ **sudo vi /etc/swatchrc**
sbe@vb-sbe:~$ **sudo vi /usr/local/bin/goswatch**
sbe@vb-sbe:~$ **sudo chmod 755 /usr/local/bin/goswatch**
sbe@vb-sbe:~$ **vi Desktop/goswatch** [paste from below]
sbe@vb-sbe:~$ **vi Desktop/gosnort** [paste from below]
sbe@vb-sbe:~$ **vi Desktop/gosnort** [paste from below]

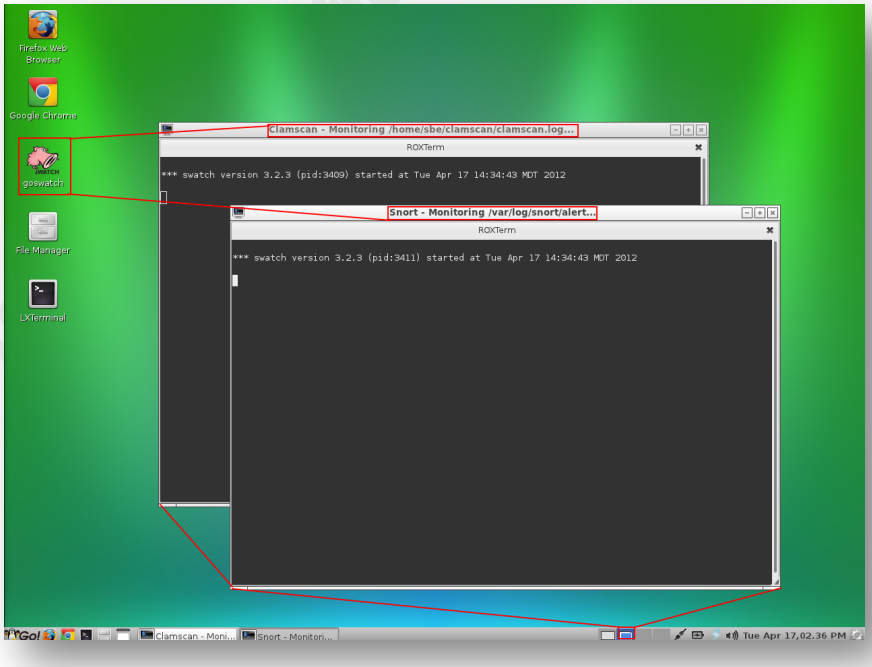
```
#goswatch Desktop Entry
[Desktop Entry]
Encoding=UTF-8
Name=goswatch
Exec=goswatch
Icon=/home/sbe/Pictures/images/snort_swatch.png
```

Robert Sorensen, rssoren@gmail.com

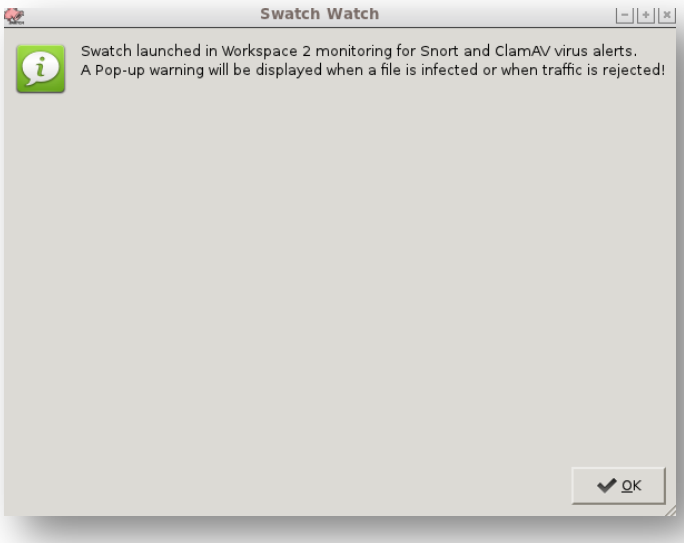
```
Type=Application
Categories=GTK;Utility;TerminalEmulator;

#gosnort Desktop Entry
[Desktop Entry]
Encoding=UTF-8
Name=GOsnort
Exec=/usr/local/bin/goSNORT
Icon=/home/sbe/Pictures/images/gosnort.png
Type=Application
Categories=GTK;Utility;TerminalEmulator;
```

Since 'GOsnort' requires sudo privilege, we'll add the following line to /etc/sudoers. This allows this script to run in privileged mode without having to enter the sudo password every time. Notice, it is limited to the two scripts associated with running Snort.

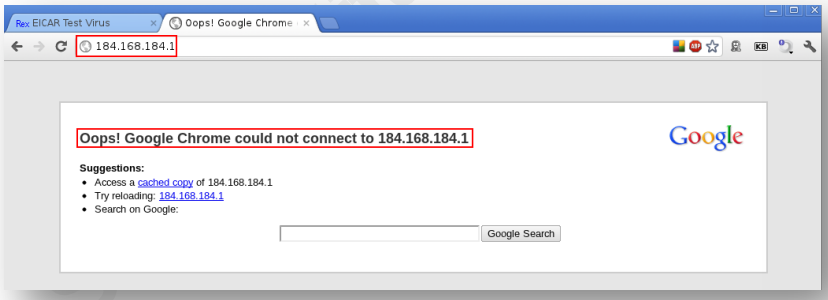
```
sbe@ubuntu:~/Desktop$ sudo visudo

%sbe ALL=NOPASSWD: /usr/local/bin/goSNORT, /usr/local/bin/gosnort
```



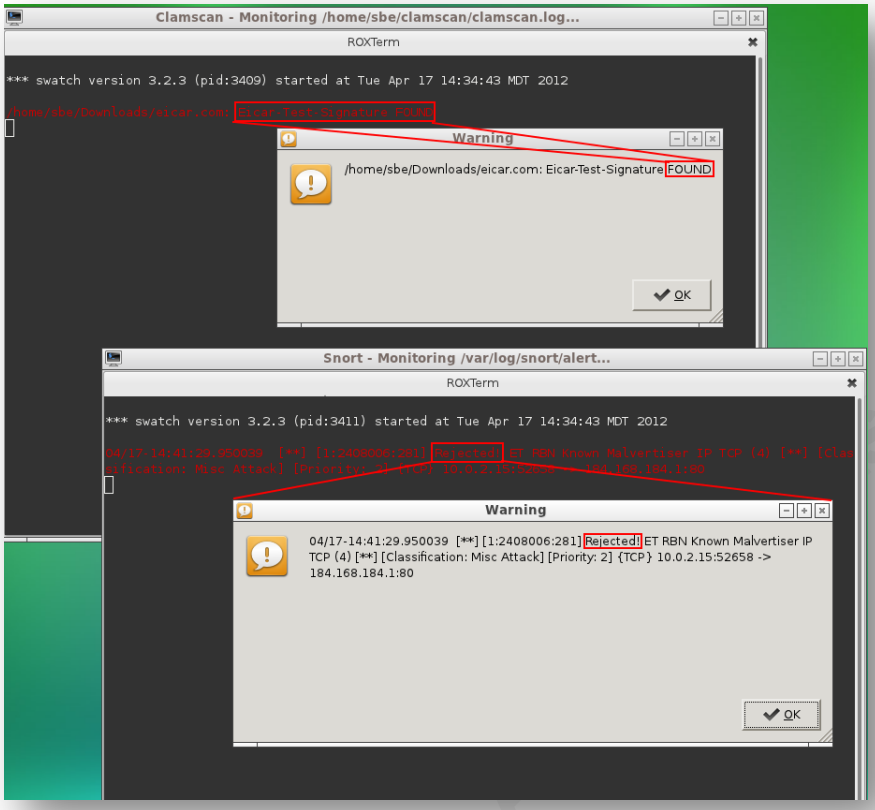
Robert Sorensen, rssoren@gmail.com



When the 'goswatch' script runs from the desktop (double-click the Snort icon), it launches two ROXTerm windows in Workspace 2 (one monitoring Clamscan virus alerts; other monitoring Snort alerts), switches back to Workspace 1, and presents the informational pop-up dialogue. It then monitors /home/sbe/clamscan/clamscan.log for any infected files and /var/log/snort/alert logs for any traffic that was rejected by Snort.

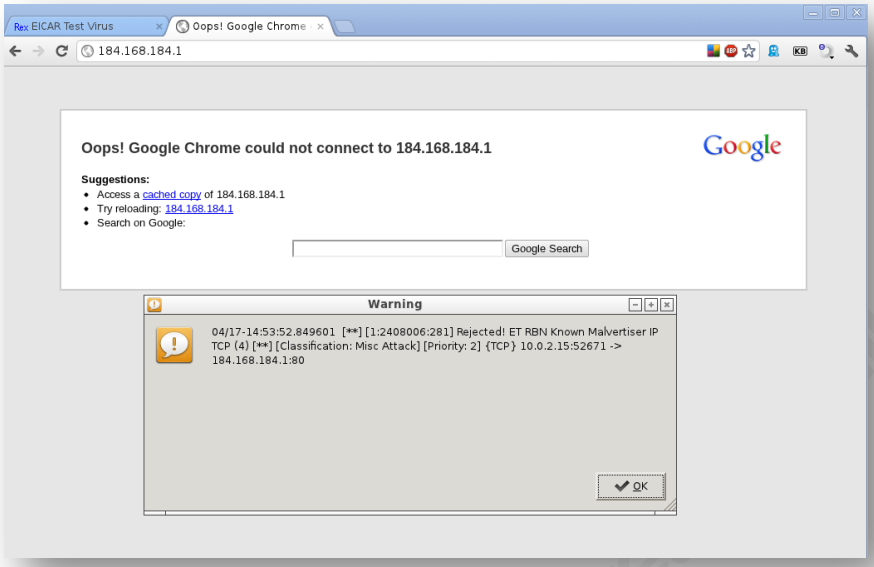


Robert Sorensen, rssoren@gmail.com



Going to a known Russian Business Network IP, 184[dot]168[dot]184[dot]1, Snort instantly blocks the traffic and generates an alert. As this example demonstrated, Swatch displayed the alert in red text and generated the 'zenity' warning pop-up. Clicking '**OK**', will close the warning dialogue. The script will continually monitor for alerts. To close Swatch, just hit '**CTRL-C**' within the terminal windows.

Robert Sorensen, rssoren@gmail.com



To display the alerts within the current Workspace, edit the /usr/local/bin/goswatch script and change the variable 'roxsession=0' as detailed in the script comments. This is the default setting.

4.6. Browser Security

Firefox is the web browser of choice when it comes to Linux distributions. Google Chrome is gaining more and more of the market share for Linux as well as Windows. For this reason, both Firefox and Google Chrome are part of SBE VM. The current web browser market share shows Internet Explorer garnering 36.3%, with Firefox second at 28.2%, and Chrome a solid third with 18.7%. Firefox and Chrome together make up almost half of the current web browser market share (Web Browser Market Share, 2011).

One of Firefox's main attractions is the abundance of available add-ons. Exploring the category, 'Privacy & Security' turned up a staggering 584 add-ons (Mozilla Security Add-Ons). One could become completely overwhelmed trying to determine what add-ons are truly beneficial for enhancing security. Knowing that the expertise of many is far superior to the expertise of one, input was solicited from the SANS Advisory Board Open.

Robert Sorensen, rssoren@gmail.com

The following add-ons were deemed to provide the best security while not requiring a technical expertise. Or as Lance Spitzner asked in the discussion, "Which addons/extensions would you recommend for the Ordinary Computer User (OCU), people with absolutely no technical skills (Spitzner, 2011)? The concept is to provide a level of protection without overwhelming a regular user. Based on the feedback from this collective group, in addition to my own research and experience, the following Firefox add-ons were included in SBE VM as outlined in Table below:

Firefox Add-ons		
Add-on	**Version**	**Description**
Adblock Plus	2.0.3	Adblock Plus allows you to regain control of the internet and view the web the way you want to. The add-on is supported by over forty filter subscriptions in dozens of languages which automatically configure it for purposes ranging from removing online advertising to blocking all known malware domains. (Adblock Plus)
BetterPrivacy	1.68	'Super-Cookie Safeguard' – Better Privacy serves to protect against special longterm cookies, which is prevalent in the Internet. This new cookie generation offers unlimited user tracking to industry and market research. This add-on makes users aware of those hidden, never expiring objects and to offer an easier way to view and to manage them (BetterPrivacy).
Ghostery	2.7.2	Ghostery tracks the trackers and gives a roll-call of the ad networks, behavioral data providers, web publishers, and other companies interested in browsing activity (Ghostery).
HTTPS Everywhere	2.0 Stable	HTTPS Everywhere is a Firefox extension that encrypts communications with a number of major websites. NOTE: Must download and install from referenced website (HTTPS Everywhere).
HTTPS Finder	0.85	HTTPS Finder automatically detects and enforces valid HTTPS connections as you browse, as well as automating the rule creation process for HTTPS-Everywhere (HTTPS Finder)
Locationbar	1.0.6	Formats and linkifies addresses in location bar. Puts emphasis on the domain to reduce spoofing risk (Locationbar).

ShareMeNot	1.11	Designed to prevent third-party buttons (such as the Facebook "Like" button or the Tweiter "tweet" button) embedded by sites across the Internet from tracking you until you actually interact with them (ShareMeNot)
WOT (Web of Trust)	20120302	A leading website reputation rating tool. The safe surfing tool uses an intuitive traffic-light style rating system to help determine which websites are trusted when they are searched. WOT ratings are powered by a global community of millions of trustworthy users, who have rated millions of websites based on their experiences (WOT).
Speed Dial	0.9.6.6	Direct access to your most visited websites (Speed Dial).

Some of the above add-ons require some additional configuration in order to get the most desired benefit. It also shows the settings incorporated in SBE VM. Detailed screenshots and configuration settings are listed in Appendix C – Browser Add-on/Extensions Guide.

Google Chrome is a web browser developed by Google and released on September 2, 2008. Google released the Chrome source code as an open source project called 'Chromium' (Paul, 2008). Chrome was built from the ground up with security in mind. It has built-in features to protect one from malicious websites using technology such as Safe Browsing, sandboxing, and auto-updates to protect against phishing and malware attacks (Google Chrome and Browser Security).

Again, Chrome has flourished with the many extensions available. Pruning through the list yielded the following extensions that were installed in SBE VM.

Google Chrome Extensions		
Add-on	Version	Description
Adblock Plus for Google Chrome (Beta)	1.2	Adblock Plus prevents the display of ads, and is a community-driven open source project which aims to make the Internet better for everyone. It will block most banners, popups and layer advertisements, disable video-ads on Youtube, and hide annoying images on any website (Adblock Plus for Google Chrome).

Ghostery	3.0.0	Ghostery helps protect your privacy by detecting trackers, web bugs, pixels, and beacons placed on web pages by Facebook, Google analytics, and over 500 other add networks (Ghostery Chrome).
Google Safe Browsing	1.1	A simple extension that verifies the reliability of a site with a simple redirect to the Google Safe Browsing (Google Safe Browsing).
KB SSL Enforcer	1.0.20	Extension enforces encryption for websites that support it as much as currently possible in Chrome. Automatically detects if a site supports SSL (TLS) and redirects to it (KB SSL Enforcer).
KB SSL Enforcer Browser Button	0.0.5	Provides a browser button for easy access to KB SSL Enforcer. Automatic security, browse encrypted. Requires KB SSL Enforcer extension (KB SSL Enforcer Browser Button).
WOT (Web of Trust)	1.2.12	The Web of Trust extension shows you which websites people trust for safe surfing, shopping and searching on the web. WOT ratings are powered by a global community of millions of trustworthy users, who have rated millions of websites based on their experiences (WOT Chrome).
Speed Dial	2.1	Direct access to your most visited websites (Speed Dial Chrome).

5. Validation Testing of SBE VM

How do we know if the security concepts built into SBE VM are effective? The best method will be to test the different levels of protection that have been incorporated in the VM. The first test will be validating the effectiveness of the shared folder between the host OS and the guest VM by downloading a known infected file and seeing if clamAV will quarantine the file. A known good file will also be downloaded to show how it gets shared back to the host OS.

Another interesting test will be doing some basic web browsing to a known news website, www.msnbc.com. Will there be aspects of a typical web site that might prove to be a means of possible infections? Our test will attempt to prove this.

Finally, an active site will be visited that is known to infect a Windows system to validate the effectiveness of the Linux-based browsing environment.

Robert Sorensen, rssoren@gmail.com

5.1. Shared Folder/clamAV Test

To test downloads from the Internet from SBE VM to our host OS, the eicar test virus will be downloaded. The "EICAR Standard Anti-Virus Test File" is a legitimate DOS program used by all vendors to test their products ability to detect a known virus signature. Four different versions of the test file will be downloaded. The first, eicar.com, contains the ASCII string, "X5O!P%@AP[4\PZX54(P^)7CC)7}$EICAR-STANDARD-ANTIVIRUS-TEST-FILE!$H+H*" which makes up the test file. The second file, eicar.com.txt, is a copy of this file with a different name. The third version contains the test file inside a zip archive. A good anti-virus scanner will spot a 'virus' inside an archive file. The last version is a zip archive containing the third file. This file can be used to see whether the virus scanner checks archives more than only one level deep (Eicar).

- Eicar.com: http://www.eicar.org/download/eicar.com
- Eicar.com.txt: http://www.eicar.org/download/eicar.com.txt
- Eicar_com.zip: http://www.eicar.org/download/eicar_com.zip
- Eicarcom2.zip: http://www.eicar.org/download/eicarcom2.zip
- Adobe Reader: http://ardownload.adobe.com/pub/adobe/reader/win/10.x/10.1.0/en_US/AdbeRdr1010_en_US.exe
- Trojan-Banker.Win32.Banker2.n.zip: http://vx.org.ua/dl/vir/Trojan-Banker.Win32.Banker2.n.zip
- Trojan-Spy.Win32.AdvKeyLogger.zip: http://vx.org.ua/dl/vir/Trojan-Spy.Win32.AdvKeyLogger.zip

The log file written out by the 'goinotify' script will be tailed as these four test files are downloaded. Then a legitimate download of Adobe Reader will be downloaded. Finally, two virus-laden files will be downloaded from VX Heavens website (VX Heavens).

```
==========
Starting scan at Wed Aug 24 21:37:01 MDT 2011...
----------
/home/sbe/Downloads/eicar.com: Eicar-Test-Signature FOUND
/home/sbe/Downloads/eicar.com: moved to '/home/sbe/.infected/eicar.com'
File 'eicar.com' was infected!  Moved to /home/sbe/.infected...
==========
Starting scan at Wed Aug 24 21:37:06 MDT 2011...
----------
/home/sbe/Downloads/eicar.com.txt: Eicar-Test-Signature FOUND
/home/sbe/Downloads/eicar.com.txt: moved to '/home/sbe/.infected/eicar.com.txt'
File 'eicar.com.txt' was infected!  Moved to /home/sbe/.infected...
==========
Starting scan at Wed Aug 24 21:37:11 MDT 2011...
----------
/home/sbe/Downloads/eicar_com.zip: Eicar-Test-Signature FOUND
/home/sbe/Downloads/eicar_com.zip: moved to '/home/sbe/.infected/eicar_com.zip'
File 'eicar_com.zip' was infected!  Moved to /home/sbe/.infected...
==========
Starting scan at Wed Aug 24 21:37:15 MDT 2011...
----------
/home/sbe/Downloads/eicarcom2.zip: Eicar-Test-Signature FOUND
/home/sbe/Downloads/eicarcom2.zip: moved to '/home/sbe/.infected/eicarcom2.zip'
File 'eicarcom2.zip' was infected!  Moved to /home/sbe/.infected...
==========
Starting scan at Wed Aug 24 21:41:07 MDT 2011...
----------
File 'AdbeRdr1010_en_US.exe' was clean!  Moving to /home/sbe/GuestVBShare...
==========
Starting scan at Wed Aug 24 21:54:34 MDT 2011...
----------
/home/sbe/Downloads/Trojan-Banker.Win32.Banker2.n.zip: Trojan.Agent-119144
 FOUND
/home/sbe/Downloads/Trojan-Banker.Win32.Banker2.n.zip: moved to
 '/home/sbe/.infected/Trojan-Banker.Win32.Banker2.n.zip.001'
File 'Trojan-Banker.Win32.Banker2.n.zip' was infected!  Moved to /home/sbe/.infected...
==========
Starting scan at Wed Aug 24 21:59:26 MDT 2011...
----------
/home/sbe/Downloads/Trojan-Spy.Win32.AdvKeyLogger.zip: Trojan.Spy.Keylogger-14
 FOUND
/home/sbe/Downloads/Trojan-Spy.Win32.AdvKeyLogger.zip: moved to
 '/home/sbe/.infected/Trojan-Spy.Win32.AdvKeyLogger.zip'
File 'Trojan-Spy.Win32.AdvKeyLogger.zip' was infected!  Moved to
 /home/sbe/.infected...
==========
```

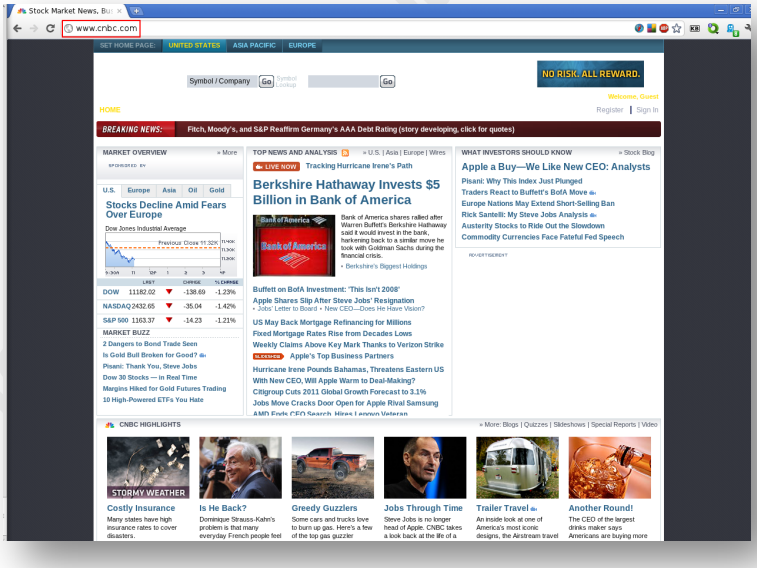
Robert Sorensen, rssoren@gmail.com

Our test provided the exact results one would expect. Infected files were identified and moved to the quarantine folder, '/home/sbe/.infected', while the clean download was moved to the shared folder, '/home/sbe/GuestVBShare' that is visible by the host OS.

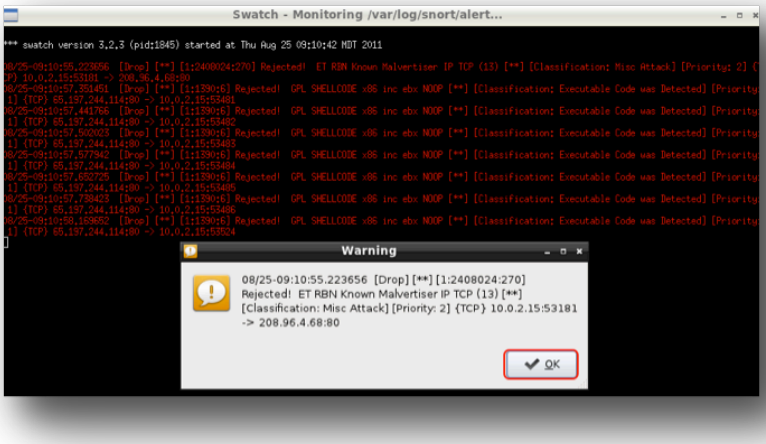
5.2. Snort in-line/Swatch Alert Test

Snort in-line mode using the latest Emerging Threat ruleset provides an immediate front line defense against known malware and other threat vectors. With the associated Swatch alert script in place, one will have immediate awareness of any threats that were blocked.

An excellent example proving the unseen potential threats that can and should be blocked was from the www.cnbc.com web site (cnbc.com). Immediately eight Swatch alerts popped up. A known Russian Business Network Malvertiser IP address was blocked in addition to insecure executable code was detected and blocked. As seen by the screenshot of the web page, no significant content was blocked except for the potentially malicious content.



Robert Sorensen, rssoren@gmail.com

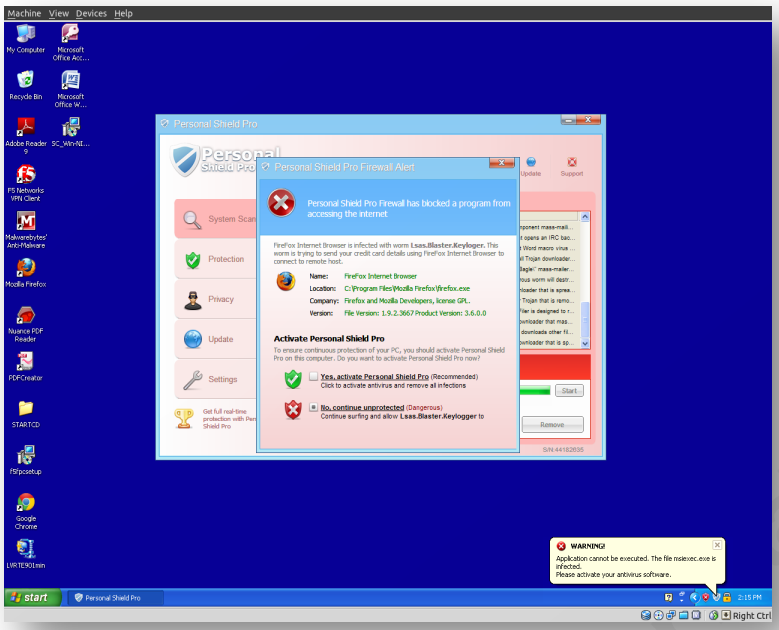


One can track the effectiveness of Snort in-line by watching the Swatch alerts as web sites are accessed. Is it not comforting to know that this functionality is built into SBE VM?

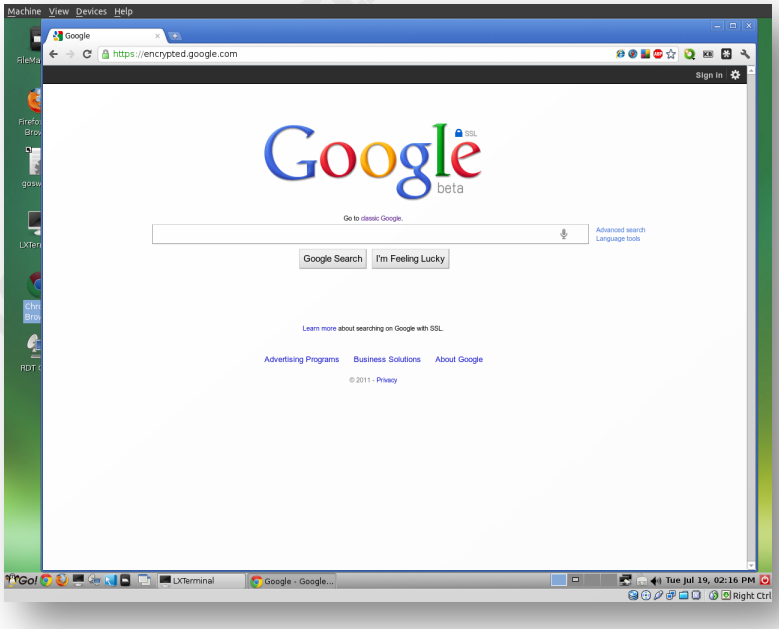
5.3. wattOSR5 Linux-based OS Environment

My organization is running FireEye Malware Analysis appliance that is very effective at detecting malware callbacks that are a 100% indication that the host was infected (FireEye). FireEye provides a pre-configured sandbox or live-mode analysis for unknown code and suspicious Web objects as well as identifies outbound malware transmissions across multiple protocols.

Another case study that demonstrates the effectiveness of SBE VM at preventing infections occurred on July 19, 2011. FireEye picked up a malware callback from a Windows host that downloaded the Backdoor.Cycbot (Backdoor.Cycbot). The URL that contained the infected java code was one that did not require the user to even click on a link but just by going to the site, triggered the infection. The URL being 188[dot]65[dot]208[dot]132[dot]/cgi-bin/counter.



A malware test environment (Windows XP VM) was used to browse to the infected site, and to make it appear the website was down, the script redirected the browser to the default google.com search page. Shortly after, the system was infected and malware downloaded and ran.

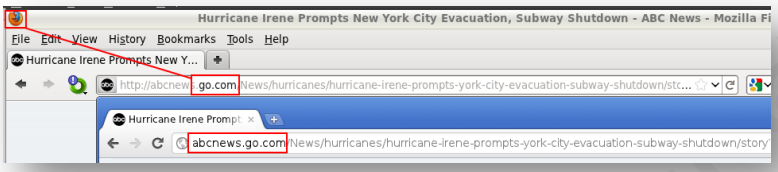


Robert Sorensen, rssoren@gmail.com

The exact same URL was immediately accessed from SBE VM. Nothing other than the redirect to google.com occurred. Due to the nature of this environment being immune to Windows-based vulnerabilities, it was not infected.

5.4. Browser Security Tests

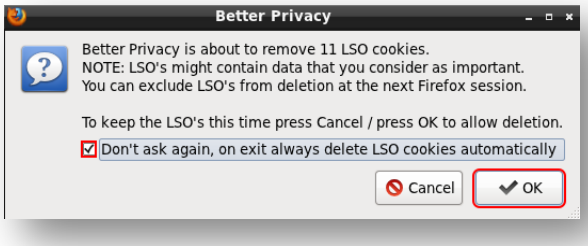
We need to next look at how the browser "add-ons" or "extensions" are helping to secure our privacy and browsing sessions. Not every one will be covered in our tests, but rest assured, they will play their role in securing our web browsing.



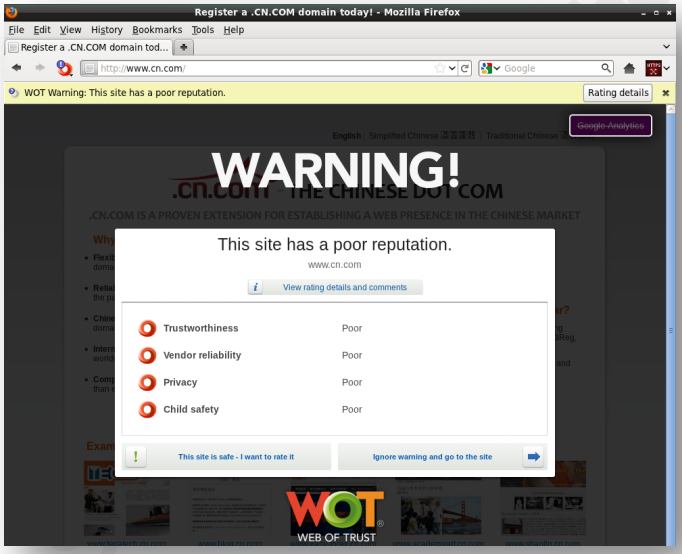
Locationbar in Firefox highlights the base domain of the website visited. The current version of Chrome has this capability built in the browser. It places the emphasis on the domain to reduce spoofing risk.

Robert Sorensen, rssoren@gmail.com

In the example of opening the denverpost.com page, Ghostery, supported on both Firefox and Chrome browsers, helped protect privacy by blocking 13 different trackers, beacons such as Facebook Social Plugins, Adlexus, DoubleClick, Twitter Button, etc.



The Firefox "add-on" BetterPrivacy checks every time the browser is closed for LSO cookies and removes them if found. Check the '**Don't ask again, on exit always delete LSO cookies automatically**' box, then '**OK**'.



The Web of Trust (WOT) "add-on/extension" for Firefox and Chrome again warned when browsing to a site that has a poor reputation based on the global community of millions of trustworthy users who have rated their browsing experience. This plugin prevented us from exposing our browser to a potentially risky website.

All of the fore mentioned browser "add-ons/extensions" add another layer of security and enhances the SBE VM.

Robert Sorensen, rssoren@gmail.com

6. Conclusions

Exploring many aspects of providing a safe browsing environment, this paper provided a means whereby Windows users can leverage a virtual Linux-based operating system that has incorporated enhanced security features. A step-by-step guide provides an eager learner the ability to create and experience the benefits of this approach.

This has been an incredible learning experience and with this, the community can benefit as well from this research. Many layers of security have been built into the SBE VM to include anti-virus scanning of downloaded files, host-based intrusion prevention system build around Snort. Also, browser "add-ons/extensions" have been installed to further layer and protect from the pitfalls that lurk on the Internet.

The last piece of the puzzle, of course, is the human element. Awareness of browsing tendencies and habits, and avoiding the potentially dangerous social engineering traps, one can truly dive head first into this wonderful innovation called the "World Wide Web!"

Robert Sorensen, rssoren@gmail.com

7. References

Adblock Plus. (n.d.) Adblock Plus – Firefox add-on to block undesired and/or malicious content. Website retrieved August 1, 2011, from https://addblockplus.org/en

Adblock Plus for Google Chrome. (n.d.) Prevents the display of ads. Website retrieved August 10, 2011, from https://chrome.google.com/webstore/detail/cfhdojbkjhnklbpkdaibdccddilifddb

Agarwal, A. (2008, March 16). OpenDNS – What is OpenDNS and why you absolutely need it" [Web log message]. Retrieved from http://www.labnol.org/internet/tools/opendsn-what-is-opendns-why-required-2/2587/

Backdoor.Cycbot. (n.d.). Retrieved August 25, 2011, from website: https://mil.fireeye.com/edp.php?sname=Bot.GBot

Bailey, P. (2010, December 21). Compiling Snort 2.9.0.3 on Ubuntu [Web log message]. Retrieved from http://bailey.st/blog/2010/12/21/compiling-snort-2-9-0-3-on-ubuntu/

Baxter, B. (2011, January 09). wattOS light-fast-now. Retrieved from http://planetwatt.com

Benie, P. (n.d.) Drop versus Reject [Web log message]. Retrieved August 23, 2011, from http://www.chiark.greenend.org.uk/~peterb/network/drop-vs-reject

BetterPrivacy. (n.d.) BetterPrivacy - Firefox add-on for viewing and managing longterm trackable cookies. Website retrieved August 1, 2011, from http://netticat.ath.cx/extensions.html

Blue Coat. (2011, July 6). Blue Coat Examines Malware Ecosystems in 2011 Mid-Year Web Security Report, retrieved from https://www.bluecoat.com/company/press-releases/blue-coat-examines-malware-ecosystems-2011-mid-year-web-security-report-0

Bradley, T. (2009, February 27). In Depth Security. Retrieved from http://netsecurity.about.com/od/newsandeditorial1/a/indepth.htm

clamAV (n.d). Open source (GPL) antivirus engine. Retrieved July 27, 2011, from clamAV website: http://www.clamav.net

Robert Sorensen, rssoren@gmail.com

Clare, T. (2011, July 6). Blue Coat – 2011 Mid-Year Security Report. Retrieved from http://www.bluecoat.com/doc/16622

cnbc.com (2011, August 25). Website retrieved August 25, 2011, from http://www.cnbc.com

Davis, K. (2011, August 3). Cybergeddon as Usual [Web log message]. Retrieved from http://www.internetevolution.com/author.asp?section_id=679&doc_id=232034

DAQ. (2010, August 12) Data AcQuisition Library. Retrieved August 23, 2011, from http://vrt-blog.snort.org/2010/08/snort-29-essentials-daq.html

Dirro, T., & Greve, P. & Kashyap, R., & Marcus, D., & Paget, F., & Schmugar, C., & Shah, J., & Wosotowsky, A. (2011, July) McAfee Threats Report: Second Quarter 2011, retrieved from http://www.mcafee.com/us/resources/reports/rp-quarterly-threat-q2-2011.pdf

Distrowatch. (n.d.) Retrieved July 25, 2011, from Distrowatch website: http://distrowatch.com

Eicar. (n.d.) Anti-Malware Testfile. Retrieved August 24, 2011, from website: http://www.eicar.org/85-0-Download.html

Emerging Threats. (n.d.) Open Source community Snort rule project. Retrieved August 23, 2011, from website: http://www.emergingthreats.net

FireEye. (n.d.) Malware Protection Systems. Retrieved August 25, 2011, from website: http://www.fireeye.com

Fossl, M. (Ed.) (2011, April) . *Symantec Internet Security Threat Report – Trends for 2010 Volume 16*, retrieved from https://www4.symantec.com/mktginfo/downloads/21182883_GA_REPORT_ISTR_Main-Report_04-11_HI-RES.pdf

Ghostery. (n.d.) Add-on that identifies and allows one to block web bugs. Retrieved August 1, 2011, from http://www.ghostery.com

Ghostery Chrome. (n.d.) Website retrieved August 10, 2011, from https://chrome.google.com/webstore/detail/mlomiejdfkolichcflejclcbmpeaniij

Google Chrome and Browser Security. (n.d.) Retrieved August 9, 2011, from Google website: http://www.google.com/chrome/intl/en/more/security.html

Google Safe Browsing. (n.d.) Retrieved August 10, 2011, from Chrome Web Store: https://chrome.google.com/webstore/detail/kcghpcmaemminjmoifneclajoomafben

Hecht. J. (2011, August 26). Image searches 'poisoned' by cybercriminals [Web log message]. Retrieved from http://www.newscientist.com/article/mg21128276.500-image-searches-poisoned-by-cybercriminals.html

HTTP-Everywhere. (n.d.) Firefox add-on that encrypts communication with major websites. Retrieved August 1, 2011, from https://www.eff.org/https-everywhere

HTTPS Finder. (n.d.) Firefox add-on that automatically detect and enforces valid HTTPS connections. Retrieved August 2, 2011, from https://code.google.com/p/https-finder

Innotek GmbH. (n.d.) Retrieved July 26, 2011, from VirtualBox website: http://virtualbox.org/wiki/innotek

Inotify-tools. (n.d.). Set of command-line programs for Linux providing a simple interface to inotify. Retrieved August 17, 2011, from https://github.com/rvoicilas/inotify-tools/wiki/#wiki-getting

Introduction to Guest Additions (2011). Retrieved July 27, 2011, from Oracle VirtualBox VM Documentation web site: http://www.virtualbox.org/manual/ch04.html#idp8356864

KB SSL Enforcer. (n.d.) Retrieved August 10, 2011, from Chrome Web Store: https://chrome.google.com/webstore/detail/flcpelgcagfhfoegekianiofphddckof

KB SSL Enforcer Browser Button. (n.d.) Retrieved August 10, 2011, from Chrome Web Store: https://chrome.google.com/webstore/detail/offhddbjgcdlkhfoolhpooeapdninhfp

Lau, H. (2011, August 04). The Truth Behind the Shady Rat [Web log message]. Retrieved from http://www.symantec.com/connect/blogs/truth-behind-shady-rat

Libdnet. (n.d.) Provides a simplified, portable interface to low-level networking routines. Retrieved August, 23, 2011, from https://code.gogle.com/p/libdnet

Locationbar. (n.d.) Linkifies URL segments – puts emphasis on the domain. Retrieved August 2, 2011, from http://en.design-noir.de/mozilla/locationbar2

Lockhart, A. (2007). *Network Security Hacks, Second Edition*. Sebastopol, CA: O'Reilly Media, Inc.

Robert Sorensen, rssoren@gmail.com

LXDE. (n.d.) Retrieved July 25, 2011, from LXDE website: http://lxde.org

Mozilla Security Add-ons (n.d.) Retrieved August 1, 2011 from Mozilla website: https://addons.mozilla.org/en-US/firefox/extensions/privacy-security/

OpenBox. (n.d) Retrieved July 25, 2011, from OpenBox website: http://openbox.org

Oracle Media Relations. (2010, January 27) Oracle Press Release – Oracle Completes Acquisition of Sun. Retrieved July 26, 2011, from Oracle website: http://www.oracle.com/us/corporate/press/044428

Phistank. (n.d.) Retrieved August 17, 2011, from website: http://www.phishtank.com

Powell, S. (2011, April 25). Welcome to the Socially Networked World: Now What? [Web log message]. Retrieved from http://sendsonline.org/2011/04/25/welcome-to-the-socially-networked-world-now-what

Paul, R. (2008, September 2) Google unveils Chrome source code and Linux port. Ars Technica. Retrieved August 9, 2011. http://arstechnica.com/open-source/news/2008/09/google-unveils-chrome-source-code-and-linux-port.ars

Ristic, I. (2008, January 23). Improving Snort_inline's NFQ Performance [Web log message]. Retrieved from http://www.inliniac.net/blog/2008/01/23/improving-snort_inlines-nfq-performance.html

ShareMeNot. (n.d.) Protecting against tracking from third-party social media buttons. Retrieved August 2, 2011, from http://sharemenot.cs.washington.edu

Skoudis, E., & Liston T. (2006). *Counter Hack Reloaded.* (2nd ed.). Upper Saddle River, NG. Prentice Hall.

Smith, G. (2011, March 3). Thousands of home computers infiltrated after hackers infect high-profile websites with booby-trapped ads. Retrieved from http://www.dailymail.co.uk/sciencetech/article-1362205/Thousands-home-computers-infiltrated-hackers-infect-high-profile-websites-booby-trapped-ads.html

Snort. (n.d.) Open source IDS/IPS. Retrieved August 17, 2011, from http://snort.org

Sophos. (2011). Security threat report 2011 [White paper]. Retrieved from http://www.sophos.com/medialibrary/Gated Assets/white papers/sophossecuritythreatreport2011wpna.pdf

Robert Sorensen, rssoren@gmail.com

Speed Dial. (n.d.) Retrieved April 9, 2012, from http://speeddial.uworks.net/speed_dial-0.9.6.6-fx.xpi

Speed Dial Chrome. (n.d.) Retrieved April 9, 2012, from Chrome Web store: https://chrome.google.com/webstore/detail/dgpdioedihjhncjafcpgbbjdpbbkikmi

Spitzner, L. (2011, July 28) Security Addons/Extensions for OCU. Retrieved from advisory-board-open@lists.sans.org

Tipon, H. & Krause, M. (2007). *Information security management handbook, sixth edition*. [Books24x7 version] Available from http://common.books24x7.com/toc.aspx?bookid=26438

VirtualBox. (n.d.) VirtualBox – A Powerful x86 and AMD64/Intel64 virtualization product for enterprise as well as home use. Website retrieved December 20, 2010, from http://Virtualbox.org

VX Heavens. (n.d.) Website retrieved August 24, 2011, from http://vxheavens.com

Web Browser Market Share (2011, June). Retrieved July 29, 2011 from W3Counter Global Web Stats website: https://www.w3counter.com/globalstats.php

Westervelt, R. (2011, August 25). Ramnit worm variant now dangerous banking malware [Web log message]. Retrieved from http://searchfinancialsecurity.techtarget.com/news/2240067214/Ramnit-worm-variant-now-dangerous-banking-malware?asrc=EM_NLN_14736496&track=NL-102&ad=846100

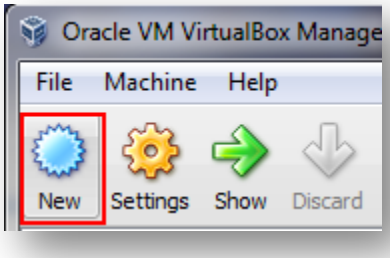
WOT. (n.d.) Website reputation rating tool. Retrieved August 2, 2011, from https://www.mywot.com

WOT Chrome. (n.d.) Website reputation rating tool. Retrieved August 10, 2011, from Chrome Web Store: https://chrome.google.com/webstore/detail/bhmmomiinigofkjcapegjjndpbikblnp

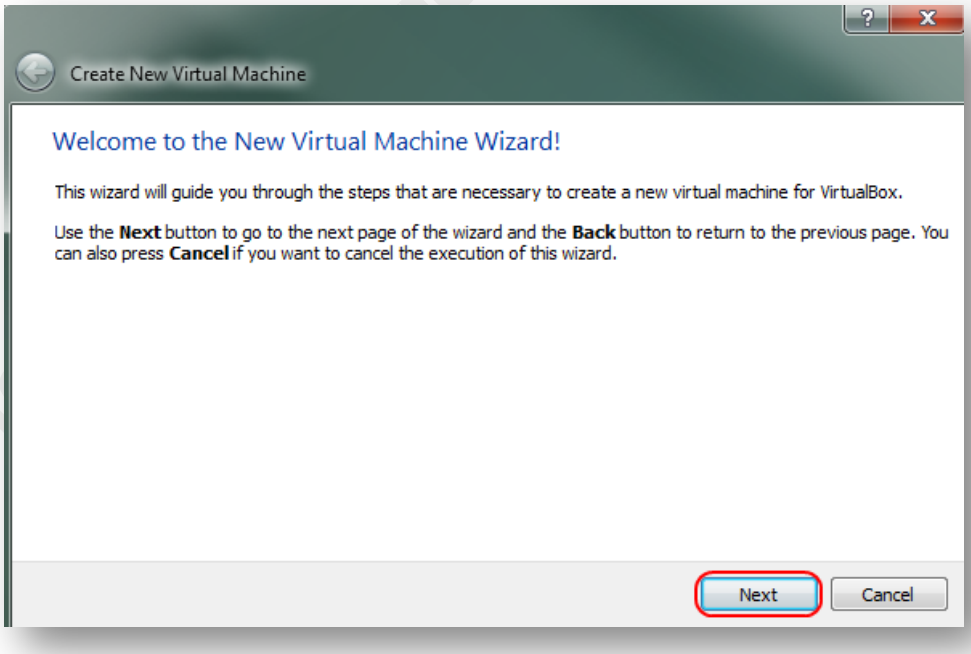
Appendix A – SBE VM Step-by-Step Guide

SBE VirtualBox VM – Create

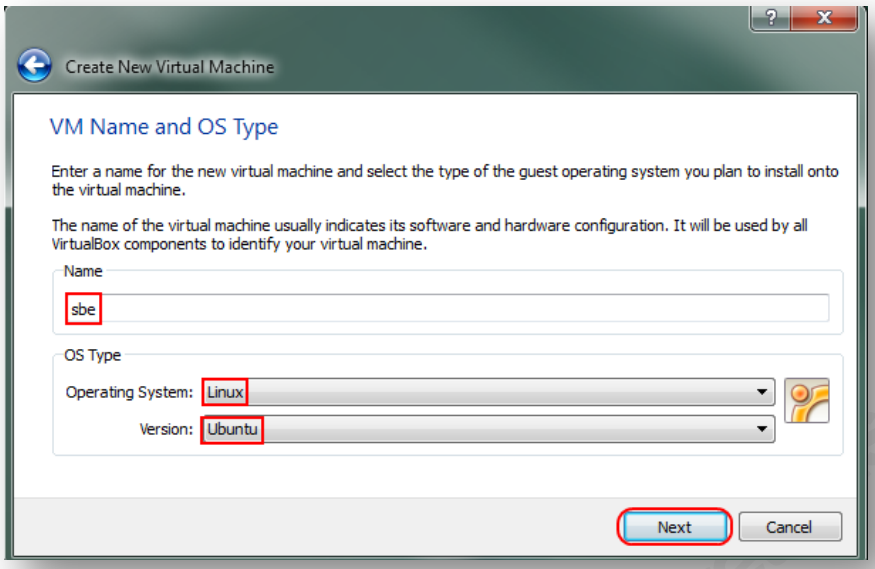
Appendix A will detail the step-by-step installation and configuration of SBE VM. There are sections for the creation of the VM using VirtualBox, VirtualBox settings, wattOSR5 installation, wattOSR5 configuration and customization, and finally, maintenance of the wattOSR5-based SBE VM. This guide assumes VirtualBox is currently installed and ready to create new VM. Let's get started!



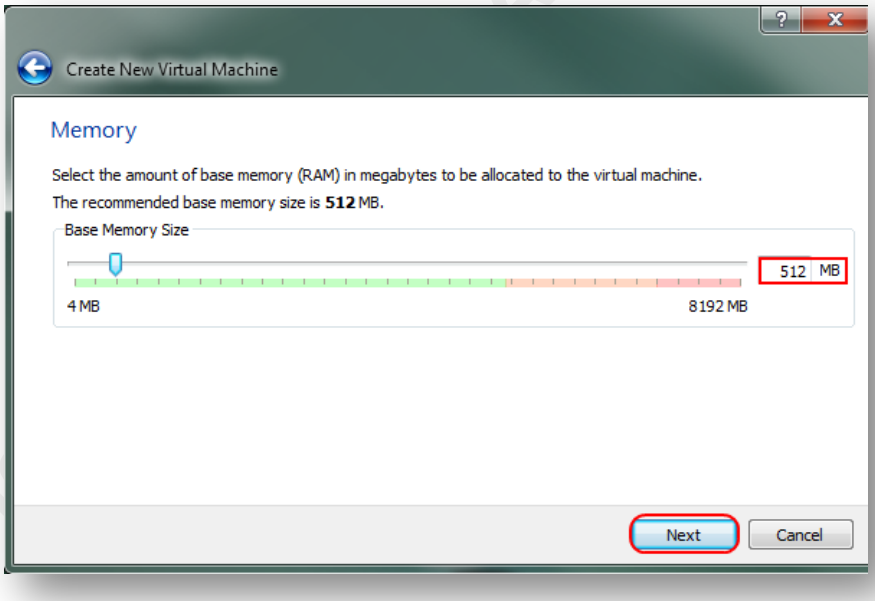
- From Oracle VM VirtualBox Manager, click '**New**'



- Welcome to the New Virtual Machine Wizard! Click '**Next**'

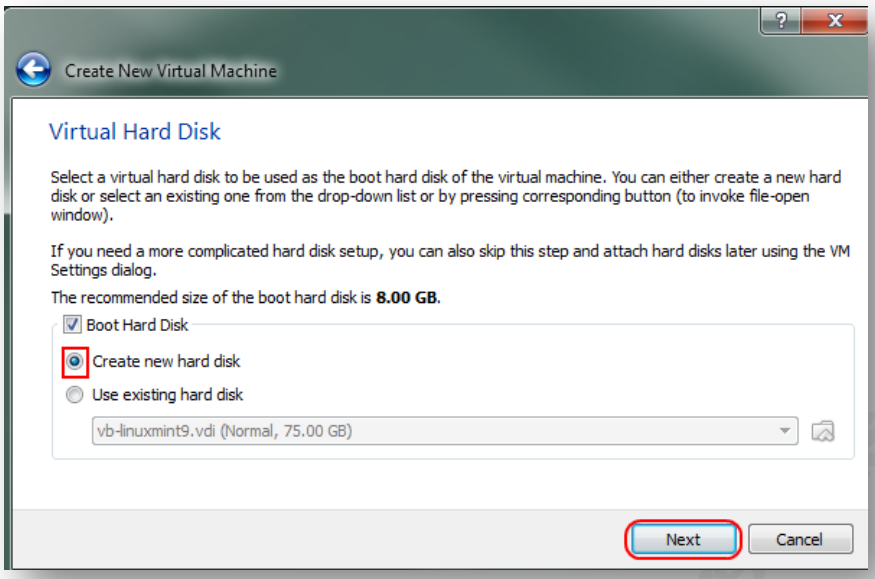


- Create New Virtual Machine - VM Name and OS Type. Enter '**sbe**' for Name, select '**Linux**'/**Ubuntu**' for Operating System/Version. Click '**Next**'.

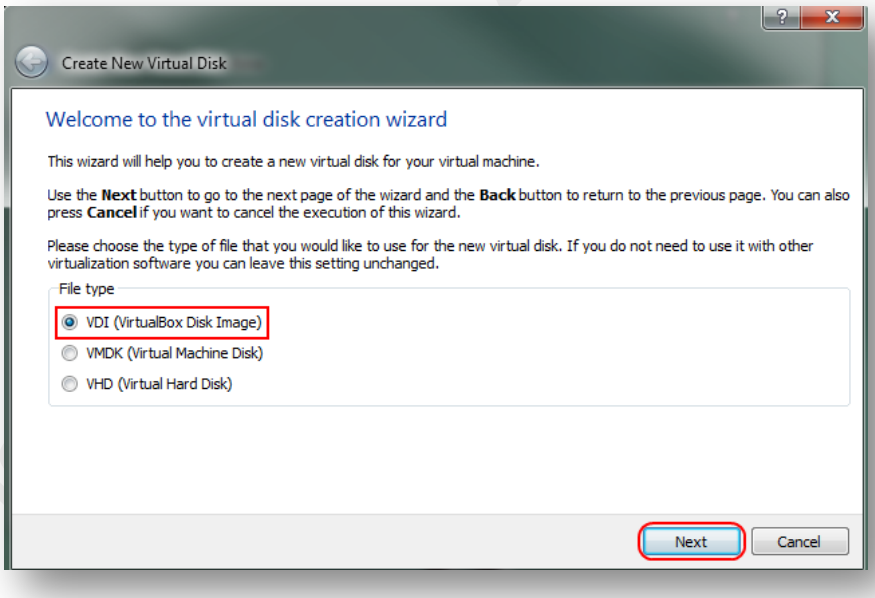


- Create New Virtual Machine – Memory. Select '**512 MB**' for Base Memory Size. If your host system has additional memory, recommend '**1024 MB**'. Click '**Next**'.

Robert Sorensen, rssoren@gmail.com

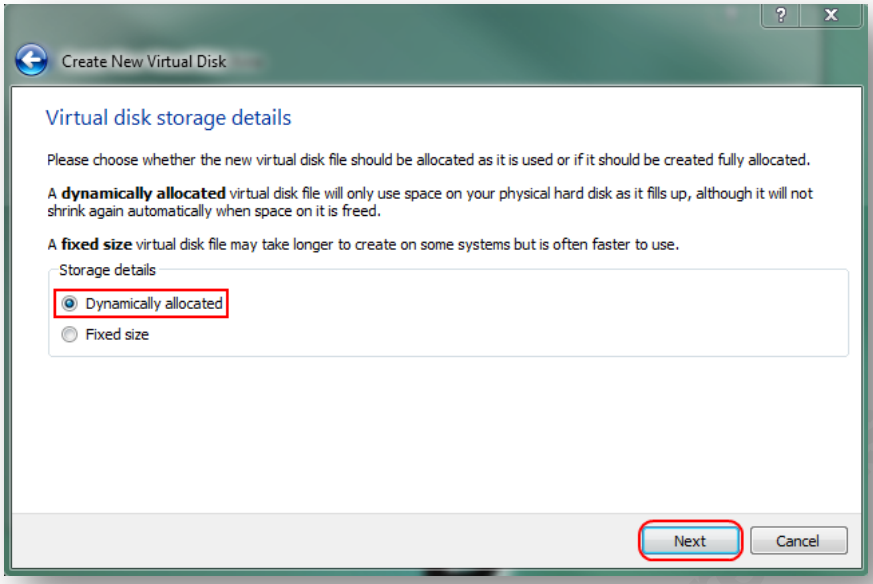


- Create New Virtual Machine – Virtual Hard Disk. Select '**Boot Hard Disk – Create new hard disk**'. Click '**Next**'.

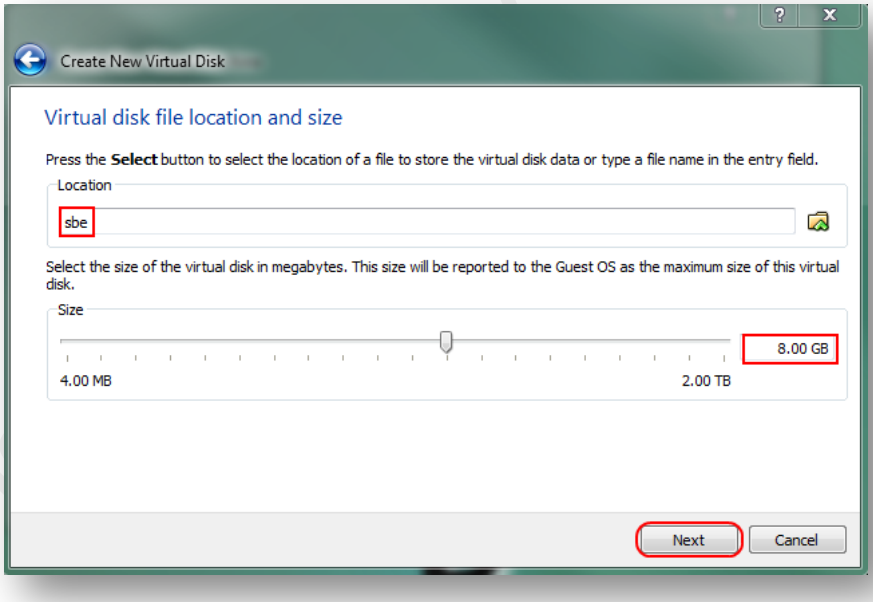


- Create New Virtual Disk – Welcome to the virtual disk creation wizard. **Select 'VDI (VirtualBox Disk Image)**. Click '**Next**'.

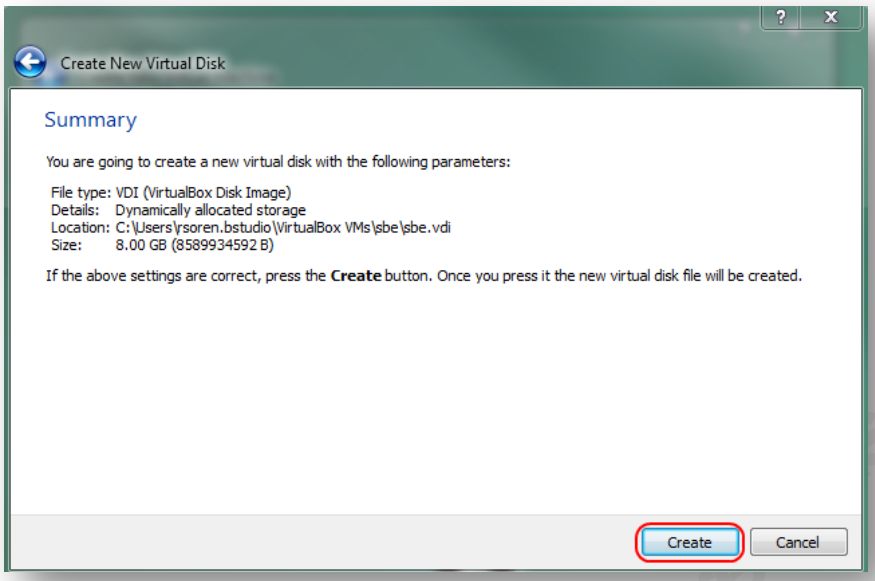
Robert Sorensen, rssoren@gmail.com



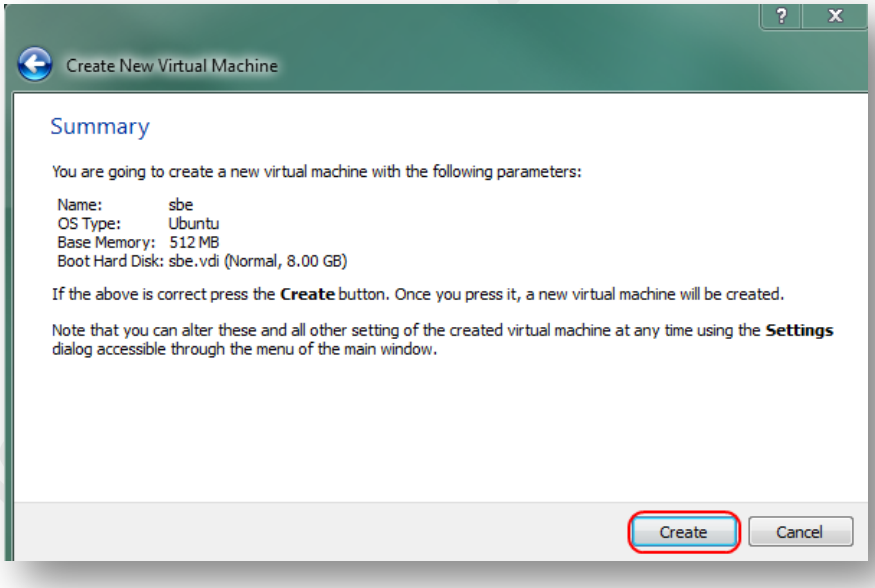
- Create New Virtual Disk – Virtual disk storage details. Select '**Dynamically allocated**'. Click '**Next**'.



- Create New Virtual Disk – Virtual disk file location and size. Select '**sbe**' for location and '**8.00 GB**' for size. Click '**Next**'.

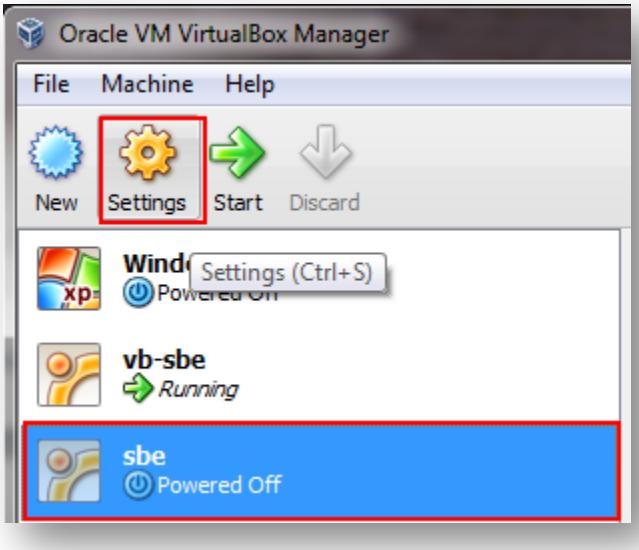


- Create New Virtual Disk – Summary. Review new virtual disk parameters. Click '**Create**'.

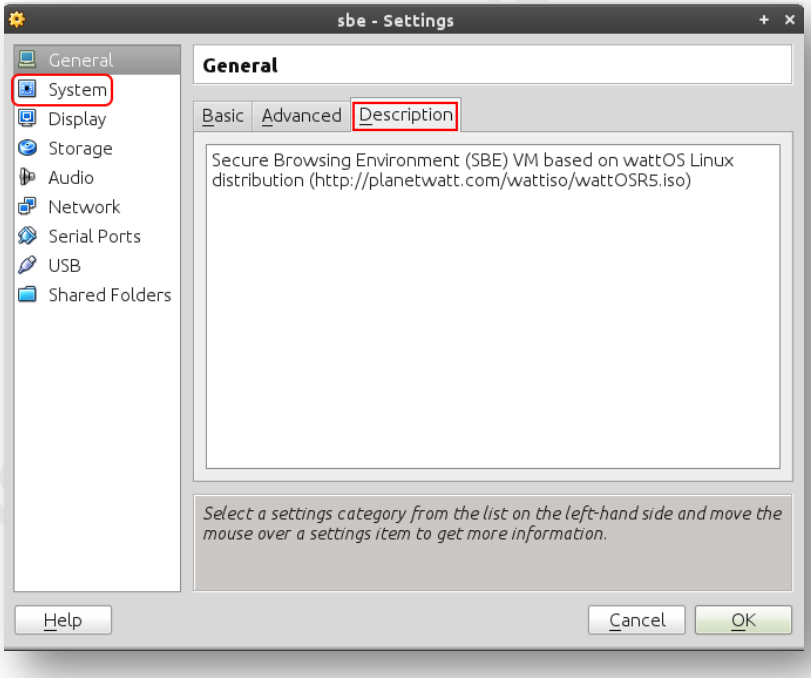


- Create New Virtual Machine – Summary. Review new virtual macnine parameters. Click '**Create**'.

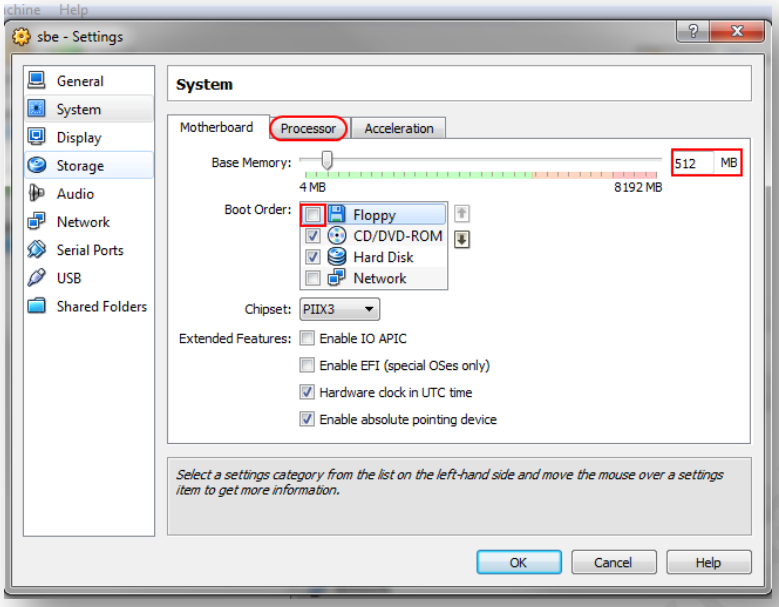
VirtualBox SBE VM - Settings



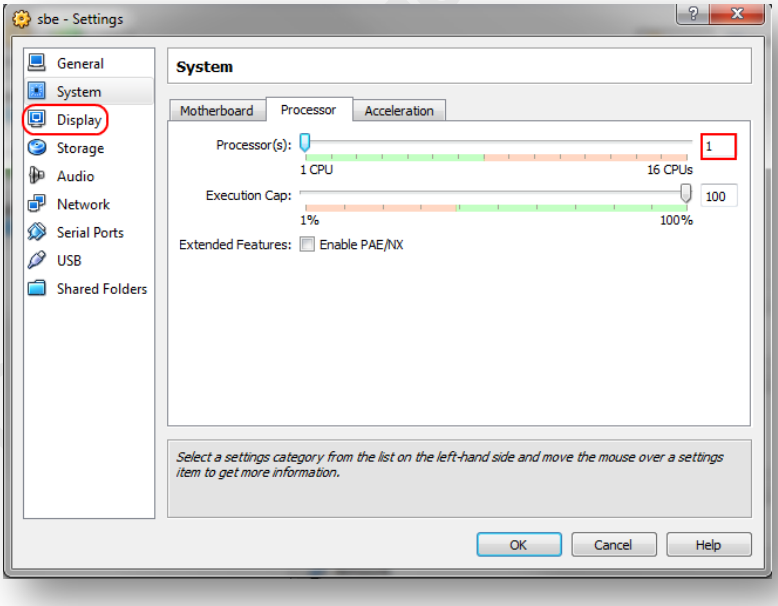
- Highlight '**sbe**', click '**Settings**'.



- Under '**General**' tab, accept the defaults for Basic/Advanced. Add an appropriate description under 'Description' tab. Click '**System**' tab on left.

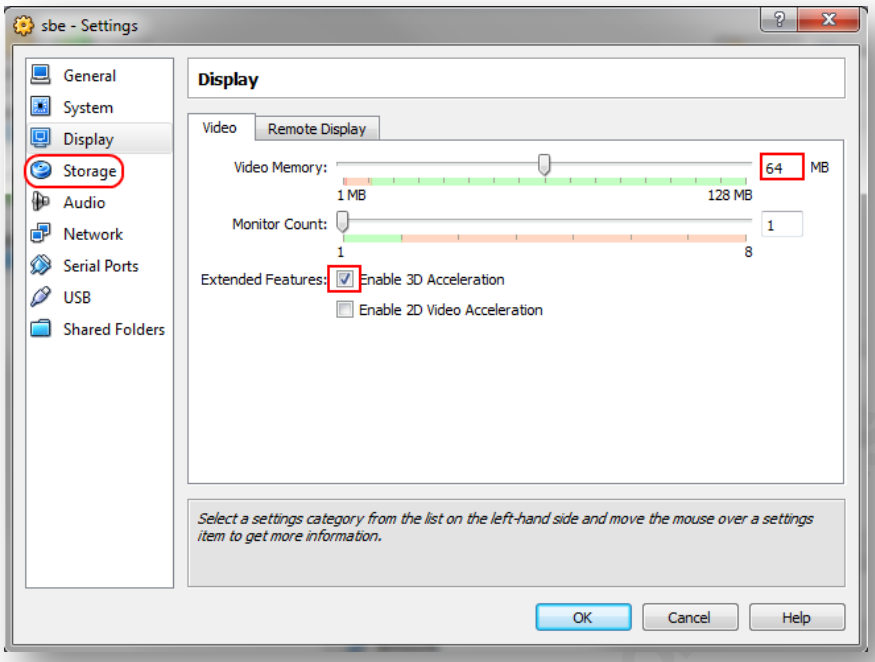


- Under '**System – Motherboard**' tab, uncheck '**Floppy**' from Boot Order. Set Base Memory to '**512 MB**' as a minimum. Note: If your system has at least 4 GB of memory, change this to 1024 MB. Click '**System - Processor**' tab.

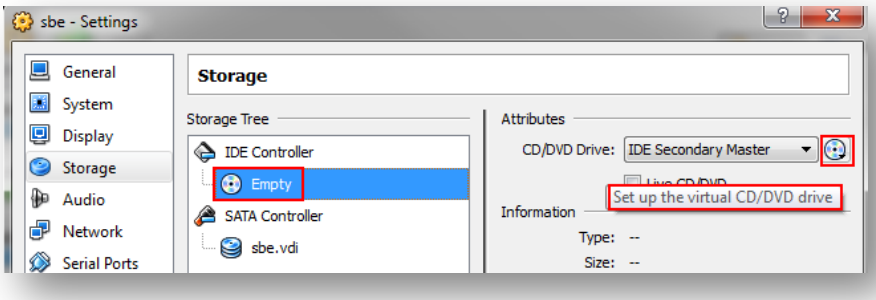


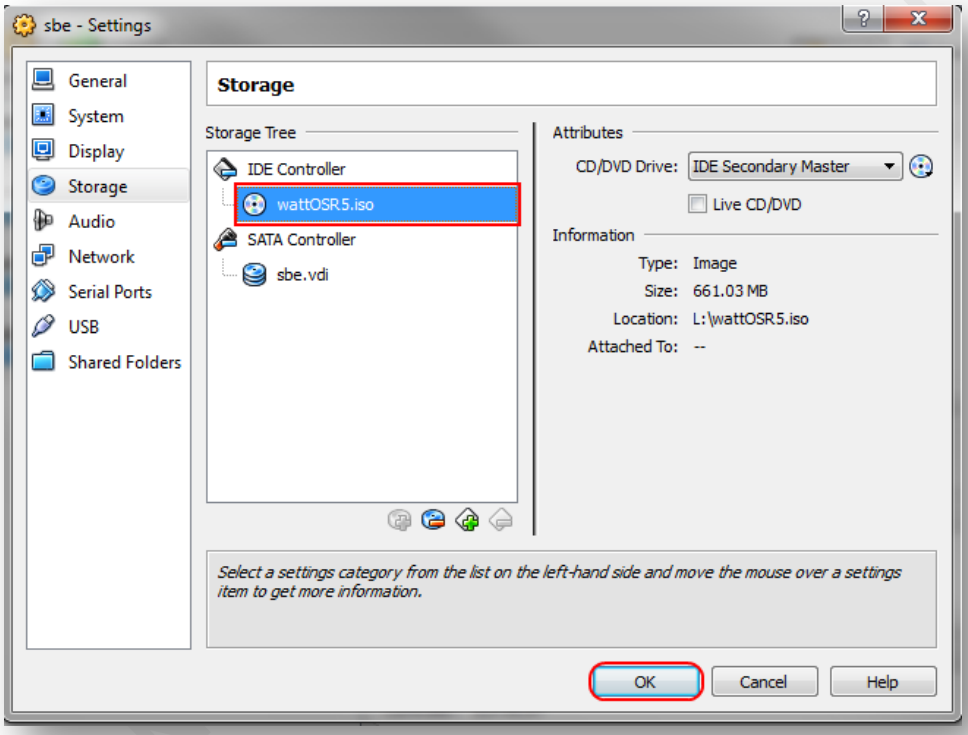
- Under '**System – Processor**' tab, select number of processor(s). Again, if your system supports it, select an appropriate number of processors. Click '**Display**' tab on left.

Robert Sorensen, rssoren@gmail.com



- Under '**Display – Video**' tab, select '**64 MB**' for Video Memory. Check '**Enable 3D Acceleration**' under Extended Features. Click '**Storage**' tab on left.

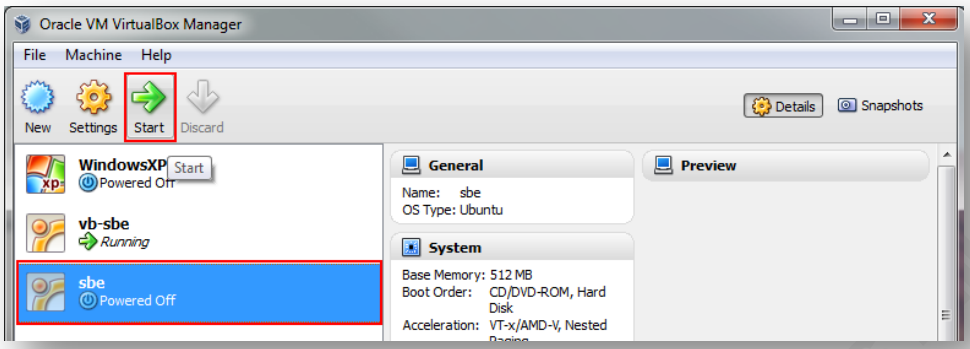




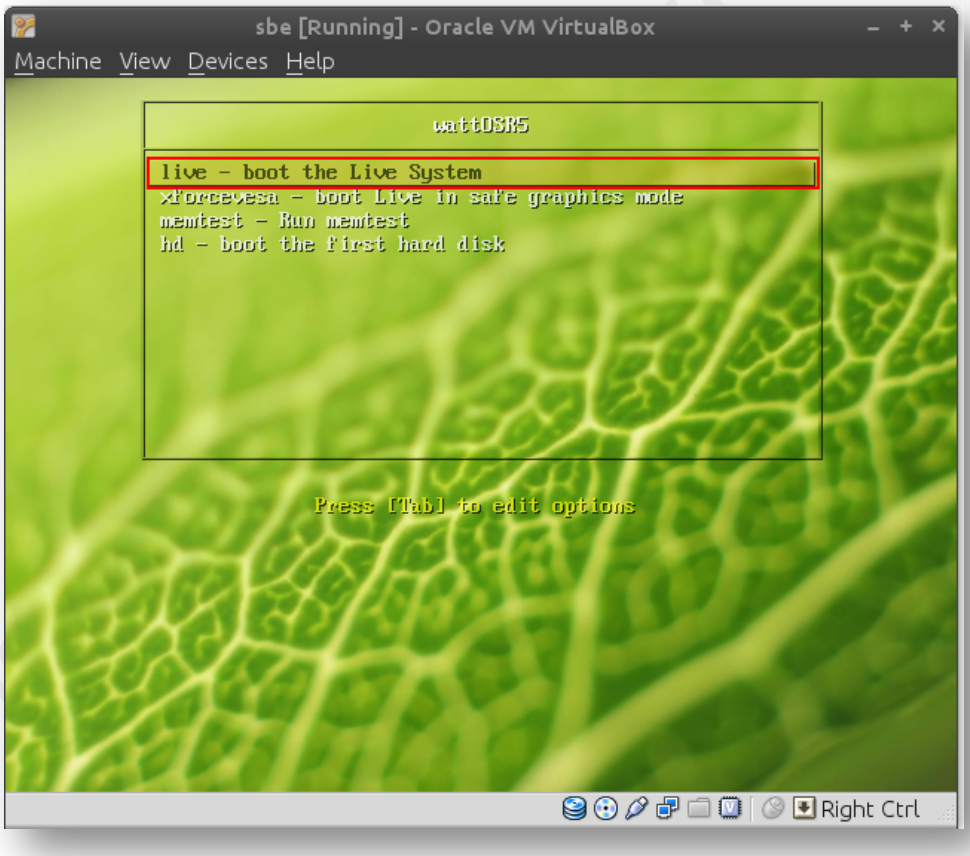
- Under '**Storage**', click drop-down CD icon next to CD/DVD Drive, and choose '**Choose a Virtual CD/DVD Disk file**' and select '**wattOSR5.iso**' that has been downloaded and saved (http://www.planetwatt.com/wattiso/wattOSR5.iso). This will complete the configuration of the VM at this stage. Click '**OK**'.

Robert Sorensen, rssoren@gmail.com

wattOSR5 - Install

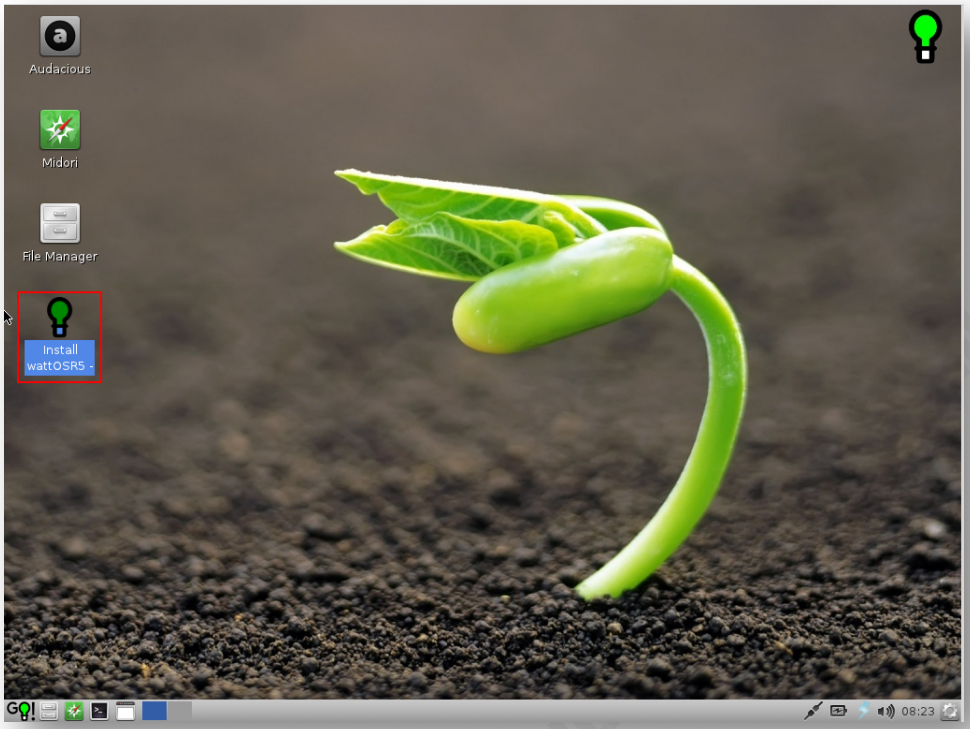


- Select '**sbe**' VM, then click '**Start**'.

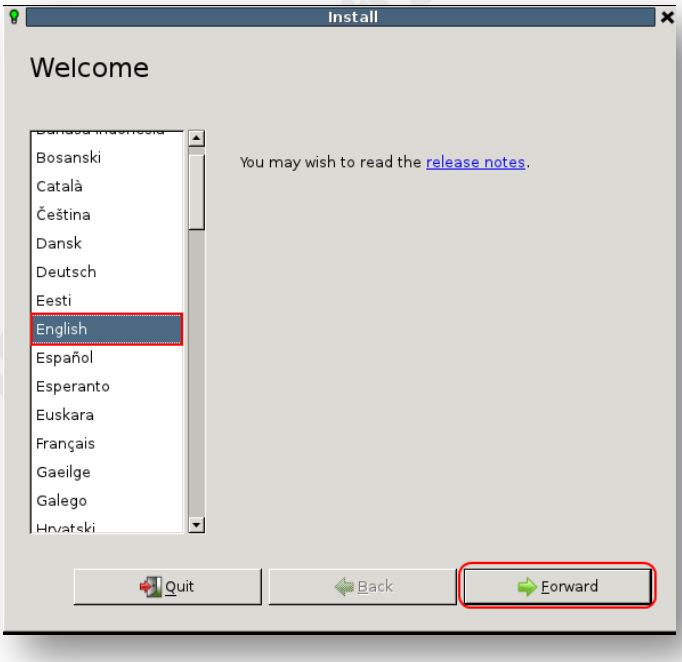


- Highlight '**live – boot the Live System**', then hit '**Enter**'.

Robert Sorensen, rssoren@gmail.com

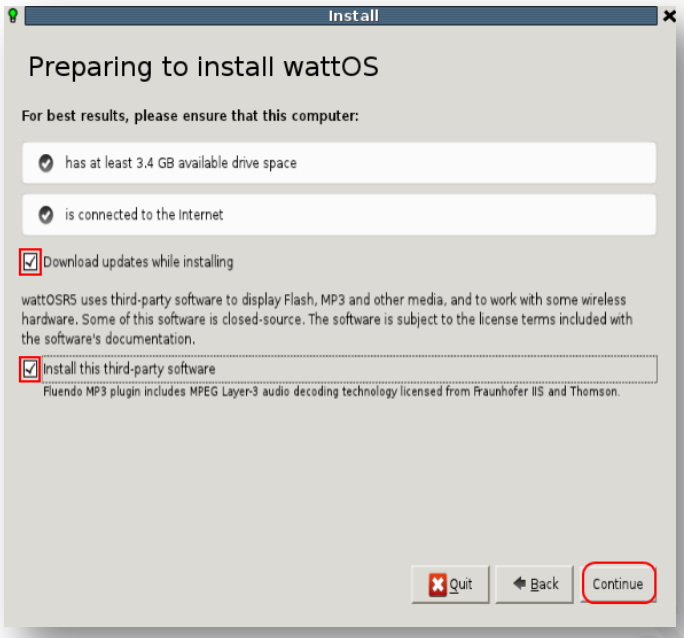


- Double-click '**Install wattOSR5**'

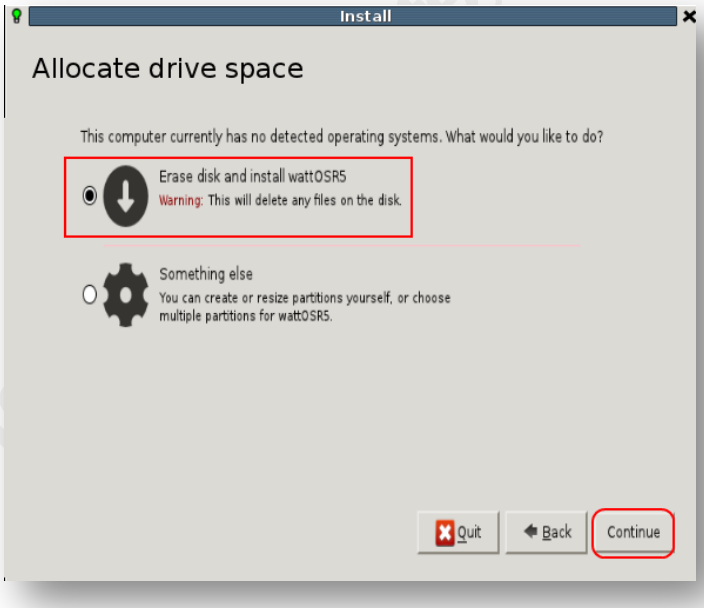


- Install – Welcome. Select '**English**'. Click '**Forward**'.

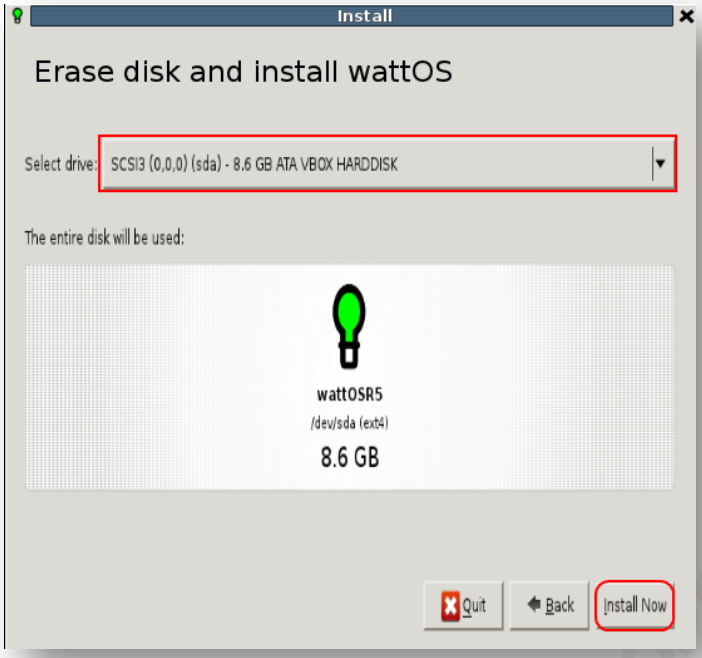
Robert Sorensen, rssoren@gmail.com



- Check '**Download updates while installing**' and '**Install this third-party software**'. Click '**Continue**'.

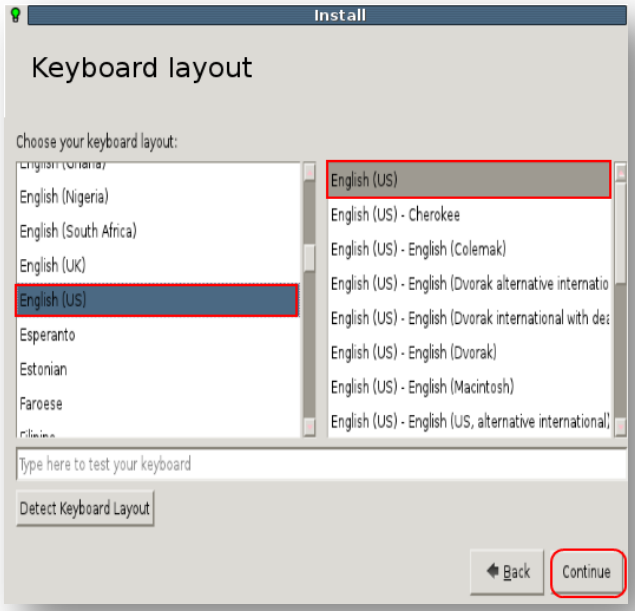


- Install – Allocate drive space. Select '**Erase disk and install wattOS5**'. Click '**Continue**'.

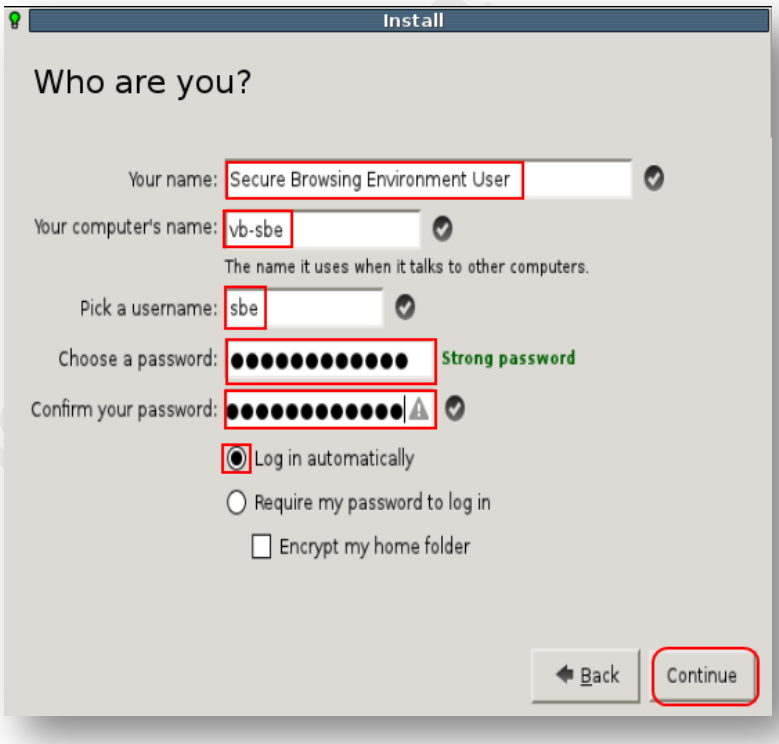


- Install – Erase disk and Install wattOS5. Select '**SCSI3 (0,0,0) (sda) – 8.6 ATA VBOX HARDDRIVE**'. Click '**Install Now**'.

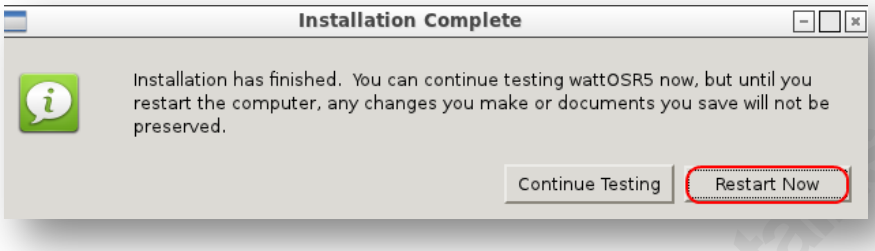
- Install – Where are you? Select '**Denver**' or appropriate time zone. Click '**Continue**'.



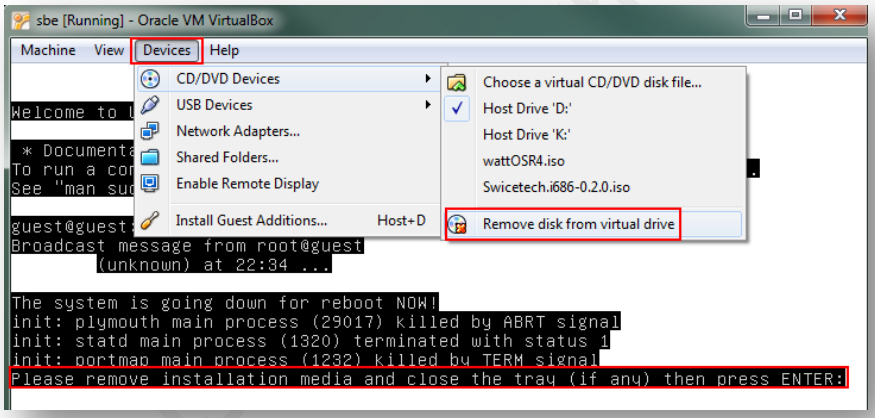
- Install – Keyboard layout. Select '**English (US)/English(US)**' or appropriate keyboard layout. Click '**Continue**'.



- Install – Who are you?
    - Enter Your name: '**Secure Browsing Environment User**'
    - Your computer's name: '**vb-sbe**'
    - Pick a username: '**sbe**'
    - Choose a password: '***enter passwd***'; Confirm your password: '***reenter passwd***'.
    - Check 'Log in automatically'. Click '**Continue**'.



- Installation Complete. Click '**Restart Now**'.



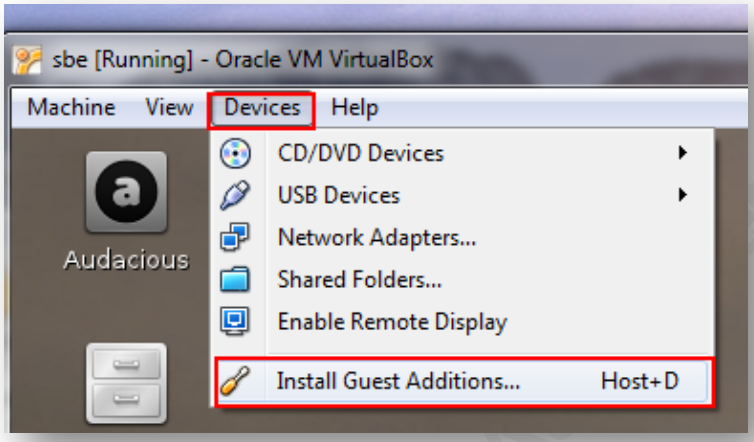
- Release 'wattOSR5.iso' from CD/DVD Devices by selecting '**Remove disk from virtual drive**'. Hit '**Enter**' to continue reboot.

Robert Sorensen, rssoren@gmail.com

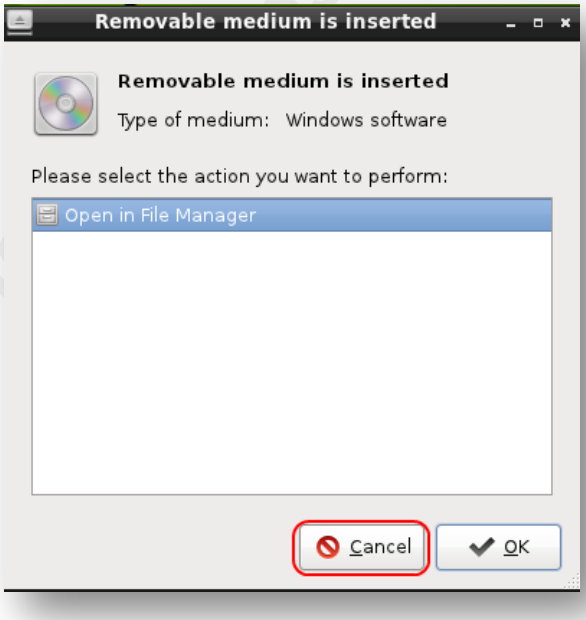
wattOS5 - Configure

Now that wattOS is installed, we now need to begin the real work of configuring the VM to provide a secure browsing environment. This will include installing the Guest Additions from Virtualbox, installing other key applications, and configuration of existing apps.

VirtualBox Guest Additions



- Select '**Devices → Install Guest Additions**' from VirtualBox pull down menu.



Robert Sorensen, rssoren@gmail.com

- Click 'Cancel' on Removable medium is inserted window. We will be installing this from a terminal window.

We must first install a compiler in order to build the kernel modules that are part of the Guest Additions. Below are the commands that were run in osurder to install Guest Additions. Open a LXterminal window and follow the steps below:

```
sbe@vb-sbe:~$ df –h
Filesystem      Size  Used Avail Use% Mounted on
/dev/sda1       6.9G  2.2G  4.4G  33% /
udev            491M  4.0K  491M   1% /dev
tmpfs           201M  764K  200M   1% /run
none            5.0M     0  5.0M   0% /run/lock
none            502M  4.0K  501M   1% /run/shm
/dev/sr0         49M   49M     0 100% /media/VBOXADDITIONS_4.1.10_76795
sbe@vb-sbe:~$ cd /media/VBOXADDITIONS_4.1.10_76795/

sbe@vb-sbe:/media/VBOXADDITIONS_4.1.10_76795$ sudo apt-get install build-
essential
[sudo] password for sbe:
Reading package lists... Done
Building dependency tree
Reading state information... Done
The following extra packages will be installed:
  binutils dpkg-dev fakeroot g++ g++-4.5 gcc gcc-4.5 libalgorithm-diff-perl libalgorithm-
diff-xs-perl
  libalgorithm-merge-perl libc-dev-bin libc6-dev libdpkg-perl libstdc++6-4.5-dev linux-
libc-dev make patch
Suggested packages:
  binutils-doc debian-keyring g++-multilib g++-4.5-multilib gcc-4.5-doc libstdc++6-4.5-
dbg gcc-multilib autoconf
  automake1.9 libtool flex bison gdb gcc-doc gcc-4.5-multilib libmudflap0-4.5-dev gcc-
4.5-locales libgcc1-dbg libgomp1-dbg
  libmudflap0-dbg binutils-gold glibc-doc libstdc++6-4.5-doc make-doc diffutils-doc
The following NEW packages will be installed:
  binutils build-essential dpkg-dev fakeroot g++ g++-4.5 gcc gcc-4.5 libalgorithm-diff-
perl libalgorithm-diff-xs-perl
  libalgorithm-merge-perl libc-dev-bin libc6-dev libdpkg-perl libstdc++6-4.5-dev linux-
libc-dev make patch
0 upgraded, 18 newly installed, 0 to remove and 1 not upgraded.
Need to get 22.9 MB of archives.
After this operation, 72.7 MB of additional disk space will be used.
Do you want to continue [Y/n]? y
Setting up build-essential (11.5ubuntu1) ...
Processing triggers for libc-bin ...
ldconfig deferred processing now taking place
```

Robert Sorensen, rssoren@gmail.com

```
sbe@vb-
sbe:/media/VBOXADDITIONS_4.1.10_76795$ sudo ./VBoxLinuxAdditions.run
Verifying archive integrity... All good.
Uncompressing VirtualBox 4.1.10 Guest Additions for Linux.....
VirtualBox Guest Additions installer
Removing existing VirtualBox DKMS kernel modules ...done.
Removing existing VirtualBox non-DKMS kernel modules ...done.
Building the VirtualBox Guest Additions kernel modules
The headers for the current running kernel were not found. If the following
module compilation fails then this could be the reason.

Building the main Guest Additions module ...done.
Building the shared folder support module ...done.
Building the OpenGL support module ...done.
Doing non-kernel setup of the Guest Additions ...done.
Starting the VirtualBox Guest Additions ...done.
Installing the Window System drivers
Installing X.Org Server 1.10 modules ...done.
Setting up the Window System to use the Guest Additions ...done.
You may need to restart the hal service and the Window System (or just restart
the guest system) to enable the Guest Additions.

Installing graphics libraries and desktop services components ...done.
sbe@vb-sbe:/media/VBOXADDITIONS_4.1.10_76795$ sudo reboot
```

Package Updates

By default wattOSR5 is installed with the Midori browser. SBE is configured and designed to support Firefox and Google Chrome browsers, and as such, Midori will be removed along with the default email program that wattOS installed. Also, java browser plugin support will be installed as added features.

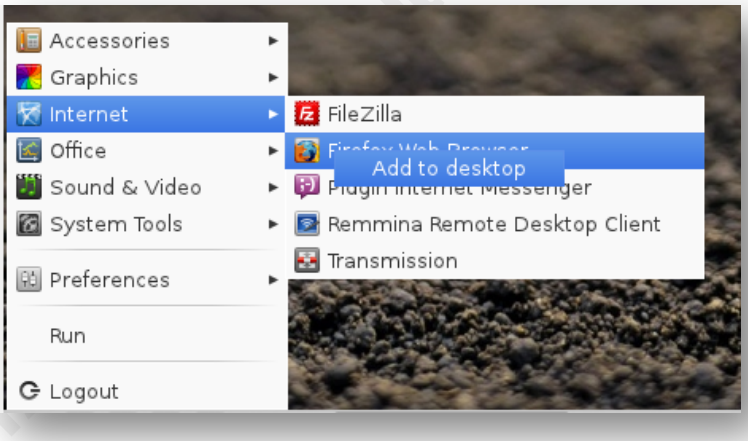
```
sbe@vb-sbe:~$ sudo apt-get install firefox
The following NEW packages will be installed:
  apturl apturl-common firefox firefox-globalmenu ubufox xul-ext-ubufox
0 upgraded, 6 newly installed, 0 to remove and 43 not upgraded.
Need to get 19.1 MB of archives.
After this operation, 40.1 MB of additional disk space will be used.
you want to continue [Y/n]? y
…
Setting up firefox (11.0+build1-0ubuntu0.11.10.1) ...
sbe@vb-sbe:~$ sudo apt-get install icedtea-6-jre-cacao icedtea6-plugin
[sudo] password for sbe:
Reading package lists... Done
Building dependency tree
```

Robert Sorensen, rssoren@gmail.com

```
Reading state information... Done
The following extra packages will be installed:
  ca-certificates-java icedtea-6-jre-jamvm icedtea-netx icedtea-plugin java-common
libaccess-bridge-java
  libaccess-bridge-java-jni libgif4 openjdk-6-jre openjdk-6-jre-headless openjdk-6-jre-lib
ttf-dejavu-extra tzdata-java
Suggested packages:
  default-jre equivs sun-java6-fonts ttf-baekmuk ttf-unfonts ttf-unfonts-core ttf-sazanami-
gothic ttf-kochi-gothic
  ttf-sazanami-mincho ttf-kochi-mincho ttf-wqy-microhei ttf-wqy-zenhei ttf-indic-fonts-
core ttf-telugu-fonts ttf-oriya-fonts
  ttf-kannada-fonts ttf-bengali-fonts
The following NEW packages will be installed:
  ca-certificates-java icedtea-6-jre-cacao icedtea-6-jre-jamvm icedtea-netx icedtea-plugin
icedtea6-plugin java-common
  libaccess-bridge-java libaccess-bridge-java-jni libgif4 openjdk-6-jre openjdk-6-jre-
headless openjdk-6-jre-lib
  ttf-dejavu-extra tzdata-java
1 upgraded, 14 newly installed, 0 to remove and 42 not upgraded.
Need to get 142 kB/39.6 MB of archives.
After this operation, 104 MB of additional disk space will be used.
Do you want to continue [Y/n]? y
Setting up icedtea-plugin (1.1.3-1ubuntu1.1) ...
Setting up icedtea6-plugin (6b21.1.3-1ubuntu1.1) ...
sbe@vb-sbe:~$ sudo apt-get install traceroute
Reading package lists... Done
Building dependency tree
Reading state information... Done
The following NEW packages will be installed:
  traceroute
0 upgraded, 1 newly installed, 0 to remove and 5 not upgraded.
Need to get 0 B/49.6 kB of archives.
After this operation, 176 kB of additional disk space will be used.
Selecting previously deselected package traceroute.
(Reading database ... 134454 files and directories currently installed.)
Unpacking traceroute (from .../traceroute_1%3a2.0.15-1_i386.deb) ...
Processing triggers for man-db ...
Setting up traceroute (1:2.0.15-1) ...
update-alternatives: using /usr/bin/traceroute.db to provide /usr/bin/traceroute (traceroute)
in auto mode.
update-alternatives: using /usr/bin/lft.db to provide /usr/bin/lft (lft) in auto mode.
update-alternatives: using /usr/bin/traceproto.db to provide /usr/bin/traceproto
(traceproto) in auto mode.
update-alternatives: using /usr/sbin/tcptraceroute.db to provide /usr/sbin/tcptraceroute
(tcptraceroute) in auto mode.
```

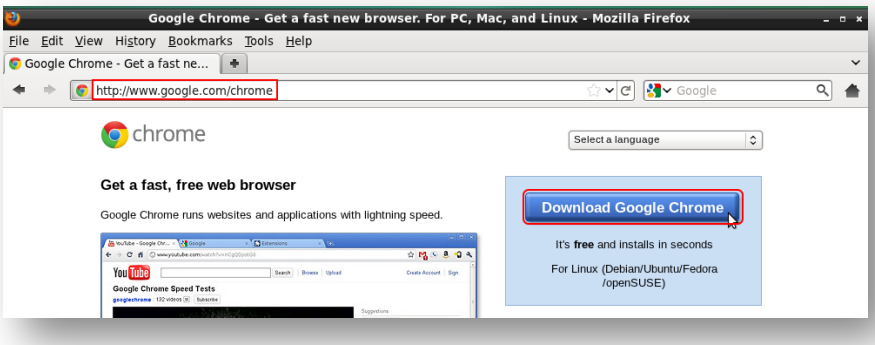
Robert Sorensen, rssoren@gmail.com

```
sbe@vb-sbe:~$ sudo apt-get remove midori
Removing midori ...
Processing triggers for hicolor-icon-theme ...
Processing triggers for man-db ...

sbe@vb-sbe:~$ sudo apt-get autoremove
Reading package lists... Done
Building dependency tree
Reading state information... Done
The following packages will be REMOVED:
  gir1.2-gstreamer-0.10 gir1.2-timezonemap-1.0 libtimezonemap1 libxklavier16 python-
appindicator python-xklavier
0 upgraded, 0 newly installed, 6 to remove and 41 not upgraded.
After this operation, 2,343 kB disk space will be freed.
Do you want to continue [Y/n]? y
(Reading database ... 130600 files and directories currently installed.)
Removing gir1.2-gstreamer-0.10 ...
Removing gir1.2-timezonemap-1.0 ...
Removing libtimezonemap1 ...
Removing python-xklavier ...
Removing libxklavier16 ...
Removing python-appindicator ...
Processing triggers for libc-bin ...
ldconfig deferred processing now taking place
```

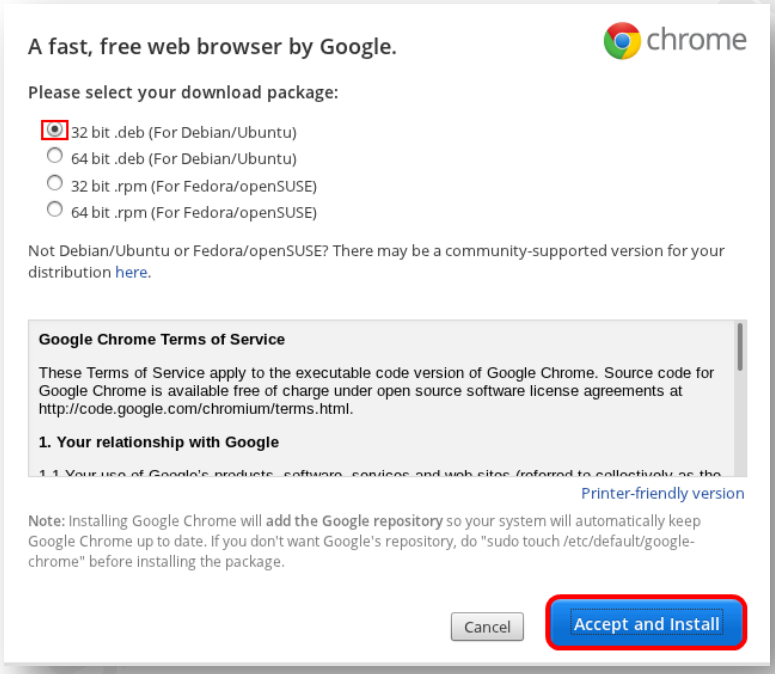


- Close Firefox and associated download/install windows.
- Add Firefox shortcut to desktop by right-clicking on '**Internet -> Firefox Web Browser**' then click '**Add to desktop**'.

We will now install Google-Chrome.

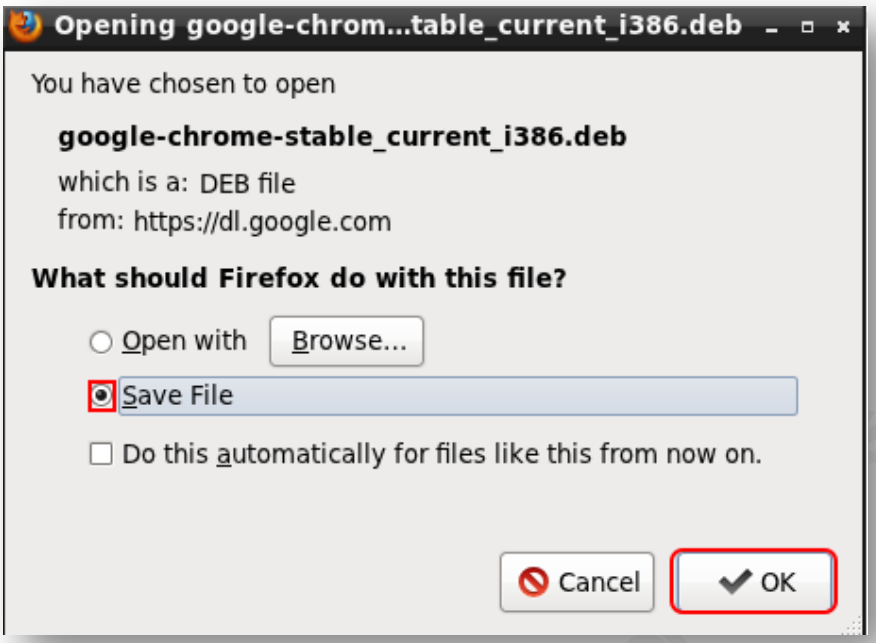


- From Firefox, go to URL http://www.google.com/chrome. Click on '**Download Google Chrome**'.



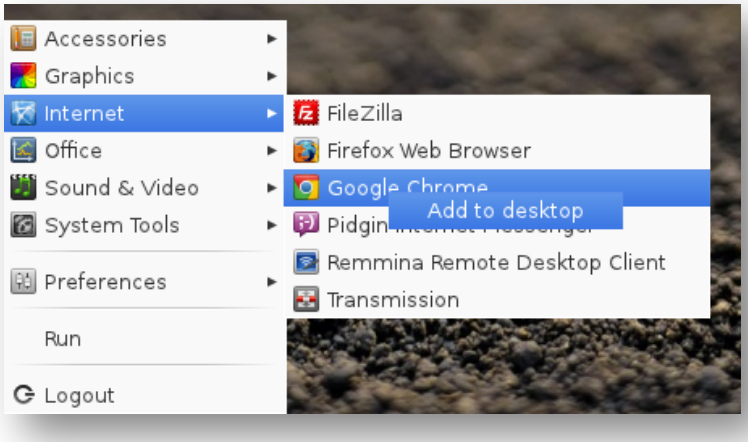
- Select '**32 bit .deb (For Debian/Ubuntu)**'. Click '**Accept and Install**'

Robert Sorensen, rssoren@gmail.com

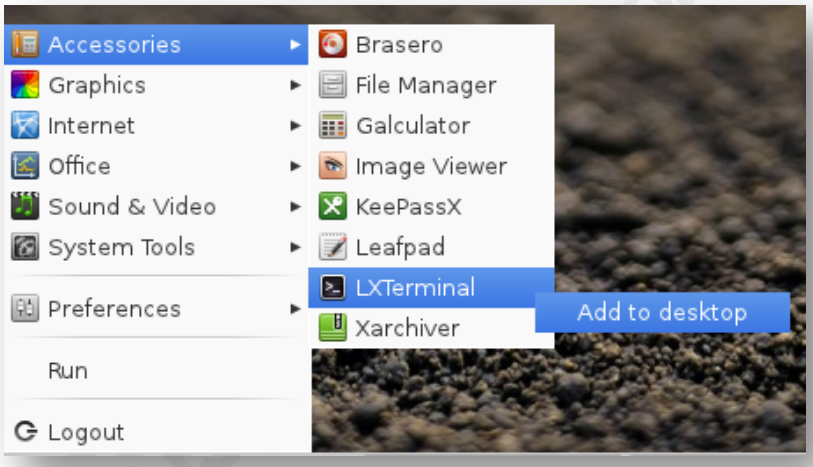


- Click '**Save File**', then '**OK**'.
- Open up LXTerminal window proceed to install google-chrome.

```
sbe@vb-sbe:~$ cd Downloads/
sbe@vb-sbe:~/Downloads$ sudo dpkg -i google-chrome-
stable_current_i386.deb
[sudo] password for sbe:
Selecting previously deselected package google-chrome-stable.
(Reading database ... 130495 files and directories currently installed.)
Unpacking google-chrome-stable (from google-chrome-stable_current_i386.deb) ...
Setting up google-chrome-stable (18.0.1025.142-r129054) ...
update-alternatives: using /usr/bin/google-chrome to provide /usr/bin/x-www-
browser (x-www-browser) in auto mode.
update-alternatives: using /usr/bin/google-chrome to provide /usr/bin/gnome-www-
browser (gnome-www-browser) in auto mode.
Processing triggers for man-db ...
sbe@vb-sbe:~/Downloads$
```



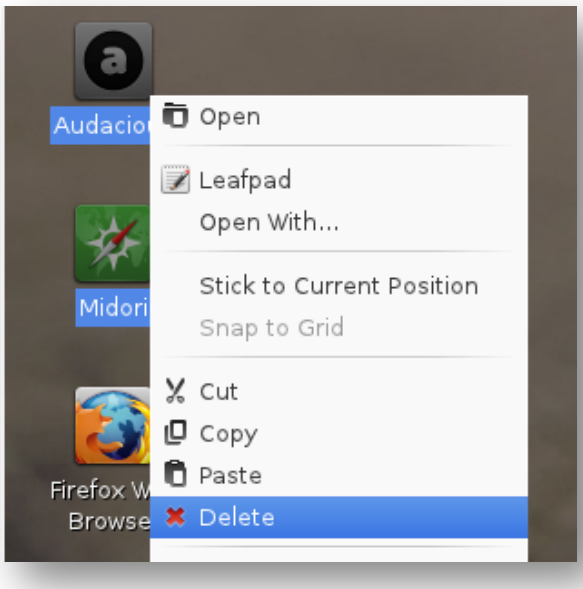
- Add Google Chrome shortcut to desktop by right-clicking on '**Internet -> Google Chrome**' then click '**Add to desktop**'.



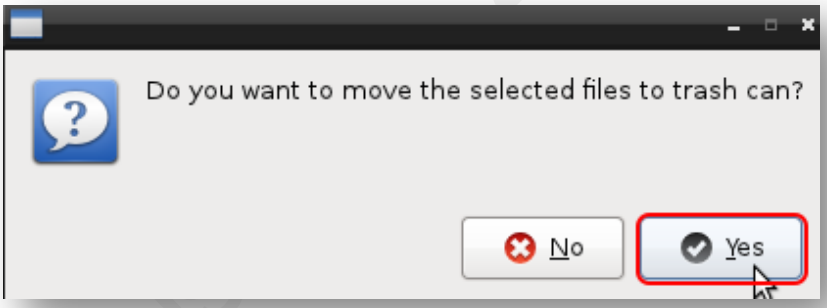
- Add LXTerminal shortcut to desktop by right-clicking on '**Accessories ->LXTerminal**' then click '**Add to desktop**'.

Next, clean up unused icons from Desktop.

Robert Sorensen, rssoren@gmail.com



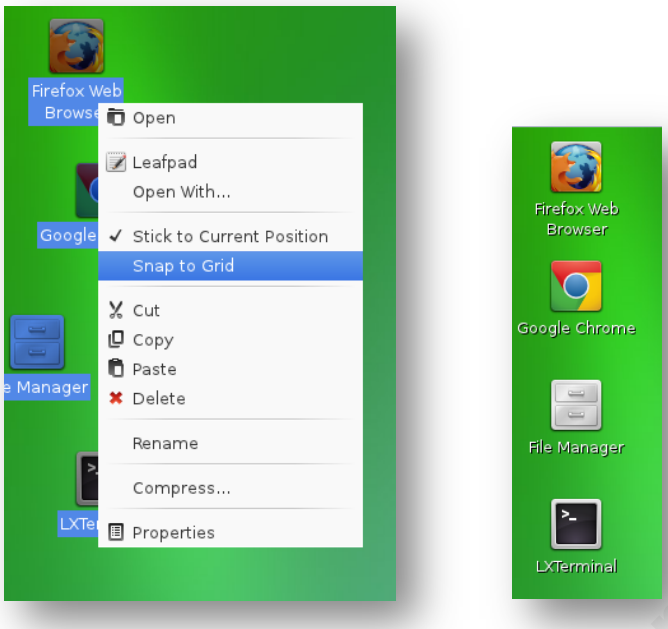
- Highlight 'Audacious/Midori' Icons on Desktop, right-click and select '**Delete**'.



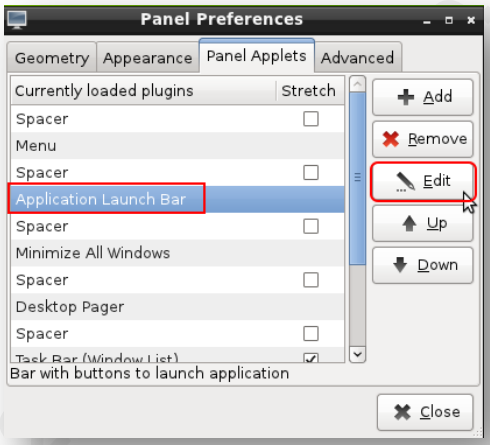
- Click '**Yes**' to confirm moving selected Icons to trash can.

Align icons on desktop by highlighting all icons, **Right-click - Snap to Grid**

Robert Sorensen, rssoren@gmail.com

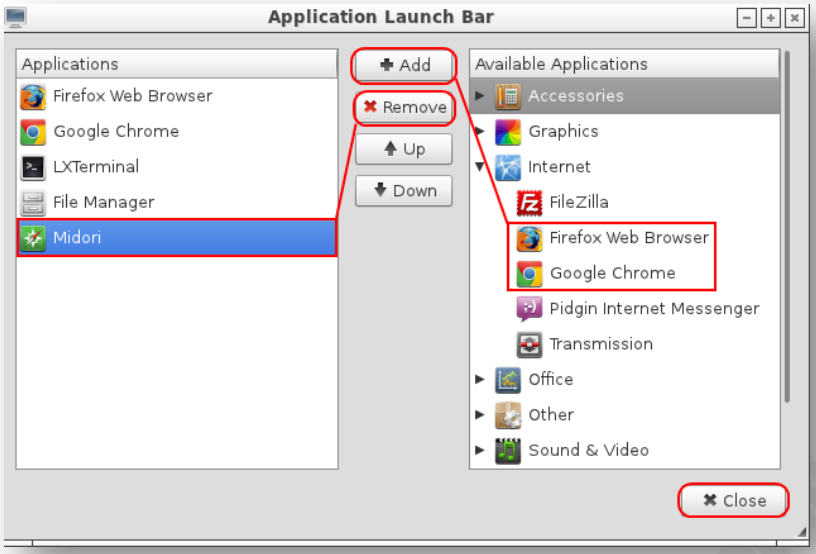


We will add shortcuts to the panel bar for quickly launching browsers and other applications.



- Right-click on LXpanel bar, select '**Add/Remove Panel Items**'. Highlight 'Application Launch Bar' just below Menu/Spacer entries. Click on '**Edit**' tab.

Robert Sorensen, rssoren@gmail.com



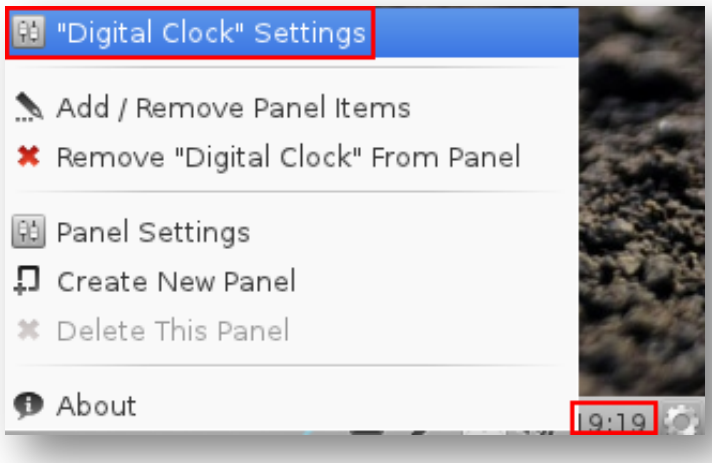
- Highlight and remove 'Midori Web Browser' Applications. Under Available Applications -> Internet, add 'Firefox Web Browser, and Google Chrome' to the list. Order by highlighting Application, click '**Up/Down**' tab accordingly - 'Firefox/Chrome/LXTerminal/File Manager'. Click '**Close**' to exit Application Launch Bar app, '**Close**' to exit Panel Preference app.

- Shortcuts for key apps are now added to LXPanel for quick access.

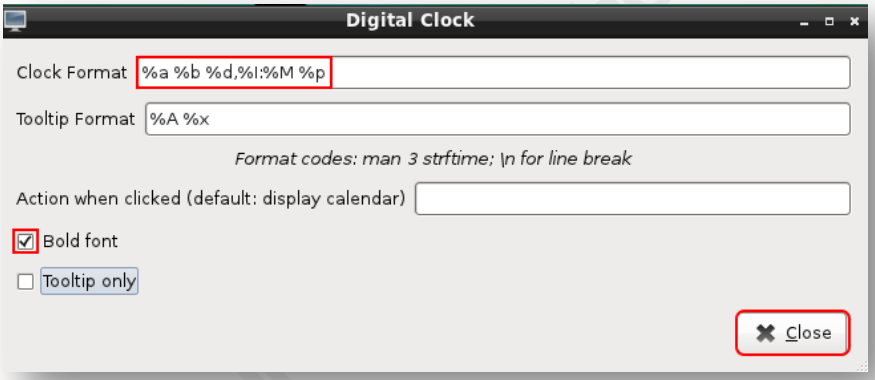
Date/Time Configuration

An update to the date and time display in the lower right side of the LXPanel will be modified. This will allow for the display of the date as well as the current time.

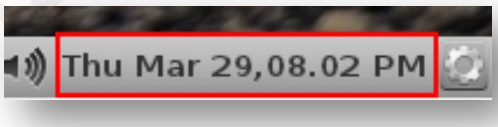
Robert Sorensen, rssoren@gmail.com



- Right-click on time in lower right hand corner of LXPanel. Select **'"Digital Clock" Settings**'



- Update Clock Format with **'%a %b %d,%I:%M %p**'. Click '**Bold font**' check box, then click '**Close**'.

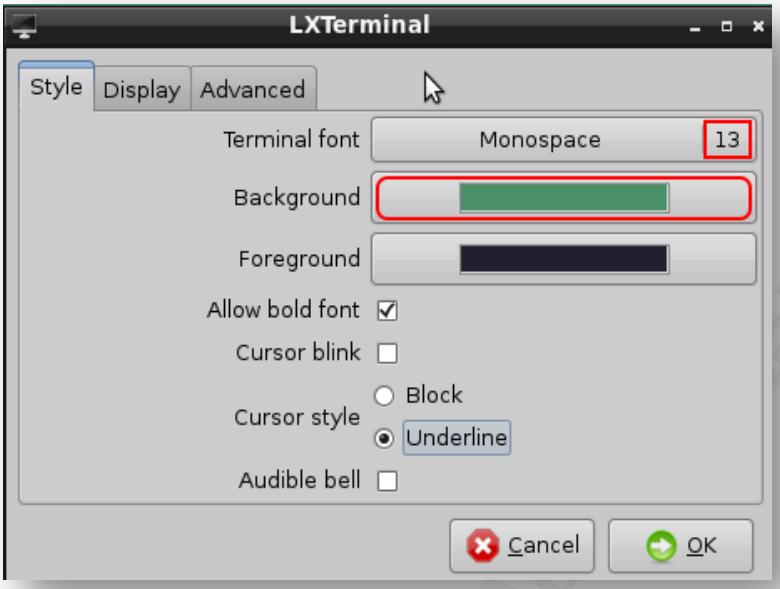


- The result is a nicely formatted date and time display.

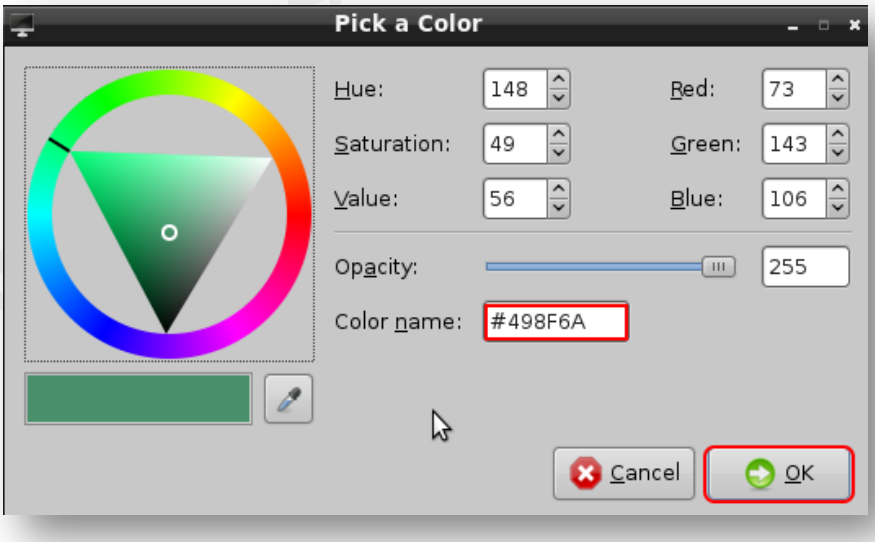
Robert Sorensen, rssoren@gmail.com

LXTerminal Configuration

A few tweaks are made to LXTerminal to fit into the new SBE theme. Open LXTerminal and select '**Edit → Preferences**'.

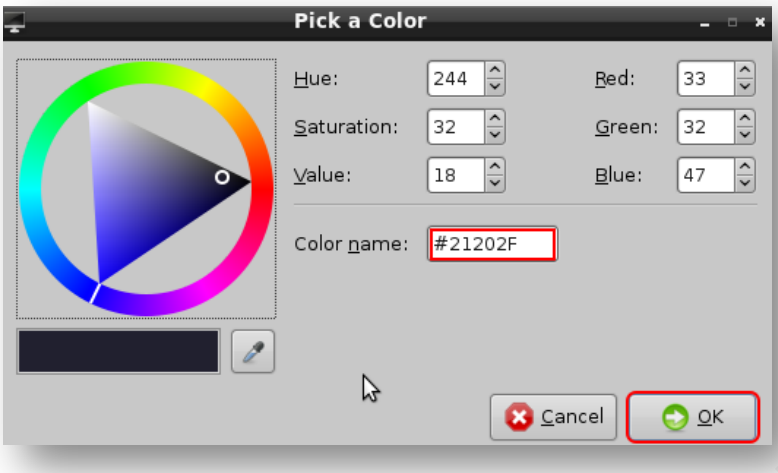


- LXTerminal – Style. Change Terminal Font to 'Monospace – 13'. Click 'Display'.



- LXTerminal – Background 'Pick a Color'. Change the Color name: to '**#498F6A**'. Click '**Ok**'. Next Select '**Foreground**'.

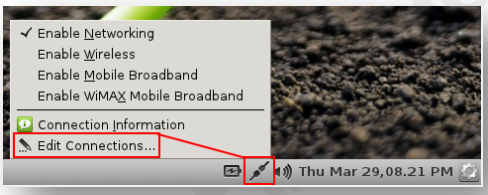
Robert Sorensen, rssoren@gmail.com



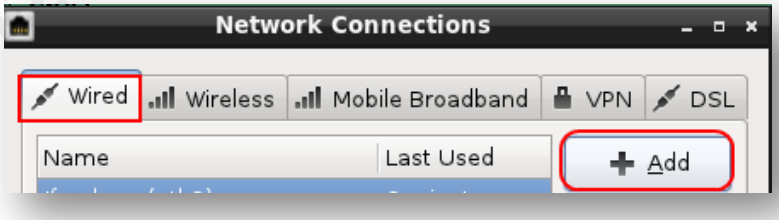
- LXTerminal – Foreground 'Pick a Color'. Change the Color name: to '**#21202F**'. Click '**OK**' to close color selection, then '**OK**' to exit preferences.

Network Configuration

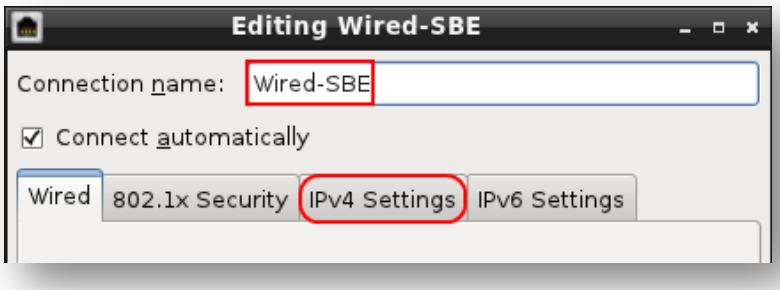
Configure an IPv4 static network address and OpenDNS servers via the 'nm-applet' that is shown in the LXPanel near far right next to the time applet.



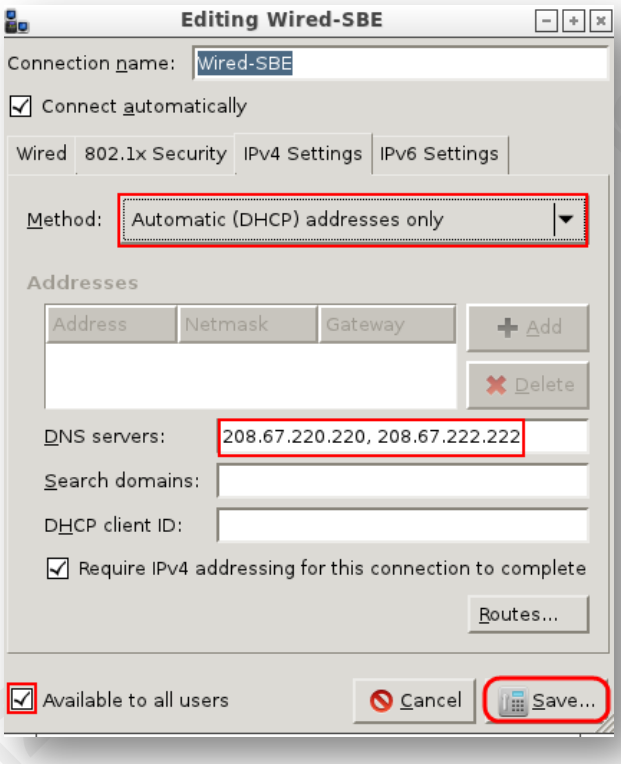
Click on 'nm-applet' icon in LXPanel, then select 'Edit Connections'.



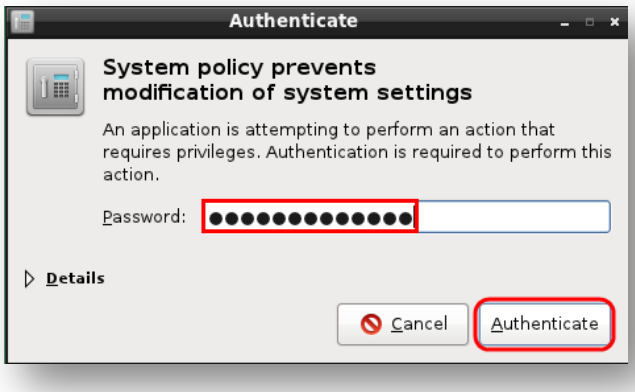
Robert Sorensen, rssoren@gmail.com



- Under 'Wired' Network Connections, Click '**Add'**, Enter Connection Name: '**Wired-SBE**'.



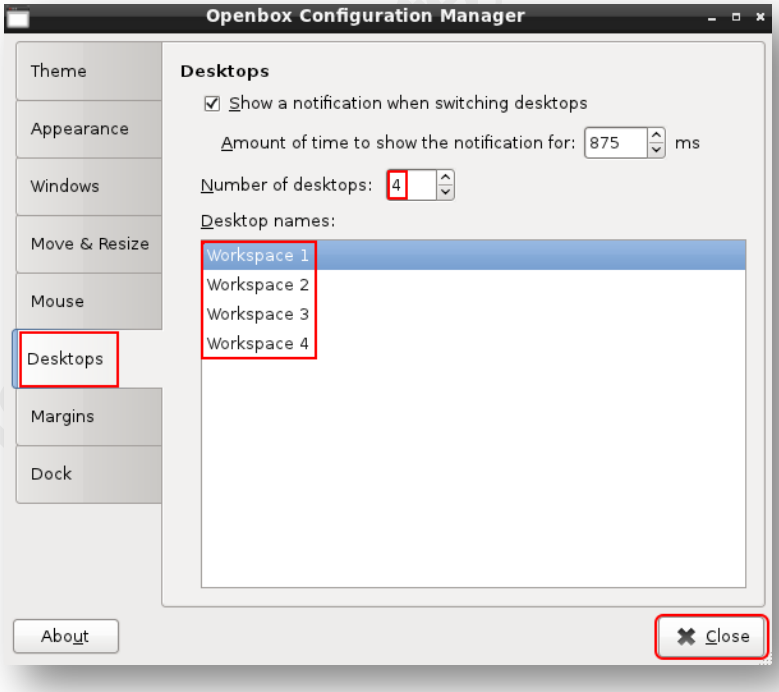
- Select '**IPv4 Settings**' tab. For Method type, Select '**Automatic (DHCP) addresses only'.** Enter DNS Servers: **208.67.220.220, 208.67.222.222**. Check '**Available to all users**' box, then click '**Save**'.



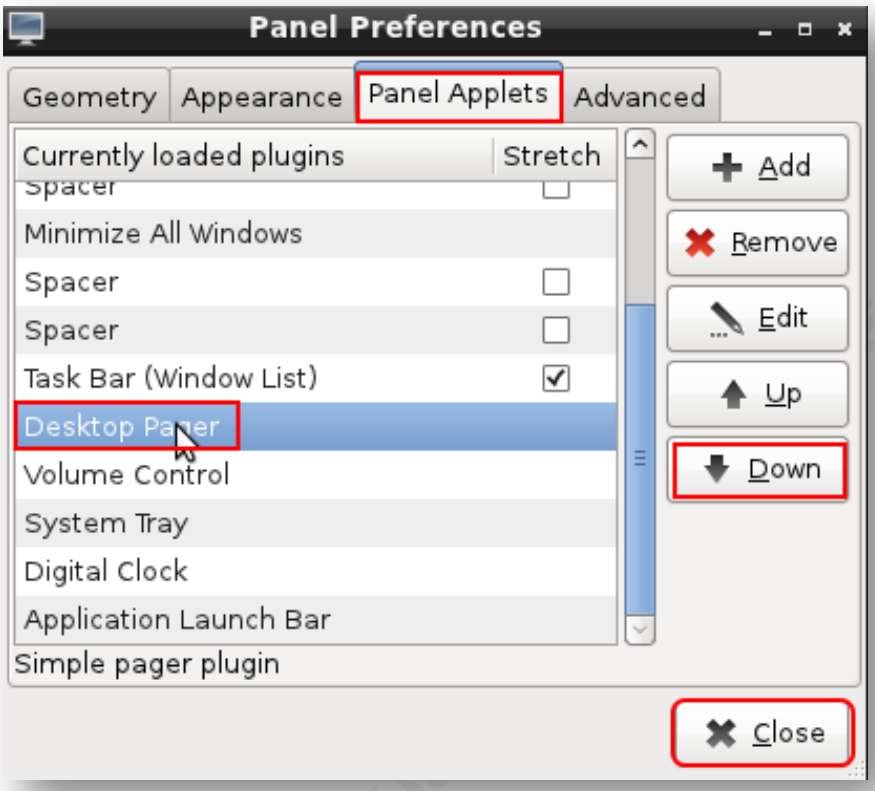
- Enter '**sudo**' Password, click '**Authenticate**', then 'Close'. Finally, click on network icon and select '**Wired-SBE**'. You might need to disconnect in order to activate new setting.

Panel Configuration

A few adjustments to the panel configuration will now be made. Increase workspaces from default of two to four under the Desktop "Pager" Settings. In order to adjust the settings, need to run, 'obconf,' which is a configuration utility for Openbox.



- Select '**Desktops'** tab on left. Change number of desktops to '**4'**. Update Desktop names to '**Workspace 1, Workspace 2, etc.**'. Click '**Close'**.

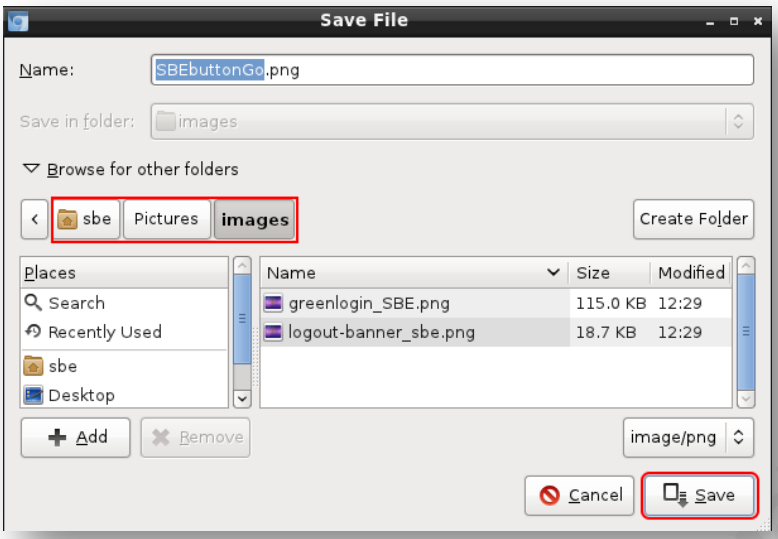


- Right-click on panel, select '**Panel Preferences**'. Click on '**Panel Applets**', highlight '**Desktop Pager**' and move down to right before 'Volume Control'. This will place it to the right side of the panel.

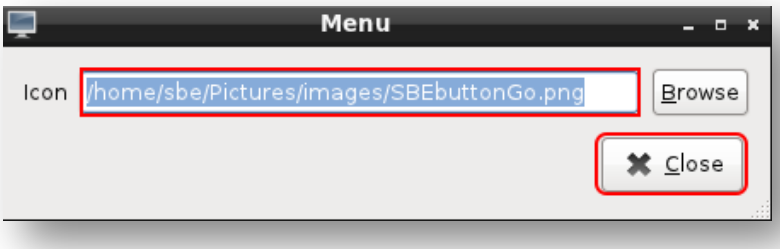
Customization

Customized background, login/logout banner will now be applied to SBE VM. One must first download the customized images from picasaweb.google.com. Download and save images to the '/home/sbe/Pictures/images' directory.

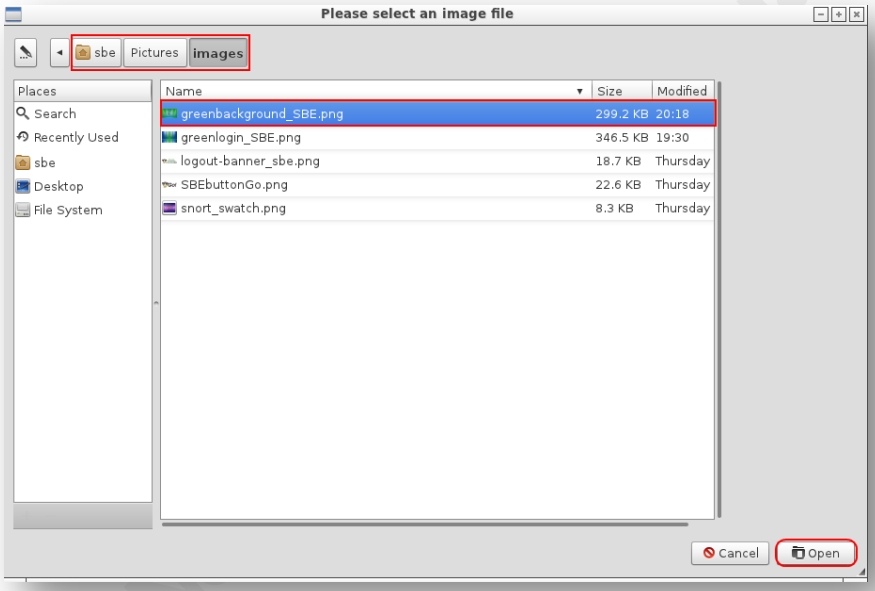
Robert Sorensen, rssoren@gmail.com



- greenbackground_SBE.png: https://picasaweb.google.com/rssoren/SBEVMCustomizedImages#5726621486876619442
- greenlogin_SBE.png: https://picasaweb.google.com/rssoren/SBEVMCustomizedImages#5726615865482355106
- logout-banner_sbe.png: https://picasaweb.google.com/rssoren/SBEVMCustomizedImages#5641147408434187506
- SBEbuttonGo.png: https://picasaweb.google.com/rssoren/SBEVMCustomizedImages#5641147425337741218
- snort_swatch.png: https://picasaweb.google.com/rssoren/SBEVMCustomizedImages#5652430566324630370
- gosnort.png: https://picasaweb.google.com/rssoren/SBEVMCustomizedImages#5737407492664935218



- Update the Start menu button by right-clicking the wattOS  button, select '**"Menu" Settings**'. Click '**Browse**' and select '**/home/sbe/Pictures/images/SBEbuttonGo.png**'. Click '**Close'**. The result will be the SBE customization button. 



- Update the background image by right-clicking anywhere on the desktop, select '**Desktop Preferences**'. Click on '**Appearance**' tab and select '**Wallpaper**' browse link. Browse to '**/home/sbe/Pictures/images/greenbackground_SBE.png**' and click '**Open**'. Click '**Close**' to exit the Desktop Preferences window.

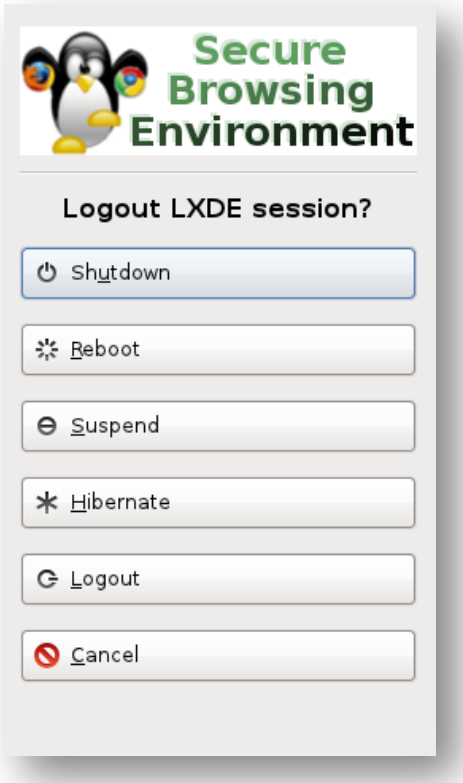
- Update the login background image by modifying '/etc/lxdm/lxdm.conf' file.

sbe@vb-sbe:~$ **sudo vi /etc/lxdm/lxdm.conf**

Change the bg= line as shown below

[display]
gtk_theme=Clearlooks
From:
bg=/usr/share/backgrounds/default.png
To:
bg=/home/sbe/Pictures/images/greenlogin_SBE.png

Robert Sorensen, rssoren@gmail.com



- Update logout banner by backing up and replacing the /usr/share/lxde/images/logout-banner.png' file.

```
sbe@vb-sbe:~$ cd Pictures/images/
sbe@vb-sbe:~/Pictures/images$ sudo mv /usr/share/lxde/images/logout-banner.png
/usr/share/lxde/images/logout-banner_wattos.png
[sudo] password for sbe:
sbe@vb-sbe:~/Pictures/images$ sudo cp logout-banner_sbe.png
/usr/share/lxde/images/logout-banner.png
```

As a final clean up step, a shell script, gozero, was developed to clean up old install files and to zero out the dead space so the VirtualBox .vdi file will allow to be compressed.

```
#!/bin/bash
#/usr/local/bin/gozero script – Clean up and zero out empty space.

if [ "$UID" -ne "0" ]
then
   echo -e "Usage: sudo $0 [must run as root!]\n"
```

Robert Sorensen, rssoren@gmail.com

```
  exit 0
else
  echo -e "Cleaning up and zeroing out empty space...\n"

  apt-get autoclean
  apt-get clean
  dd if=/dev/zero of=/0bit bs=20971520
  if [ -f /0bit ]
  then
    echo -e "0bit file created. `ls -l /0bit`...\n"
    echo -e "Removing file...\c"
    rm -f /0bit
    echo -e " Done\n"
  else
    echo -e "0bit file not created...\n"
  fi
echo -e "Make sure you compress the .vdi file outside of this VM!\n"
fi
```

Running the program performs an apt-get autoclean, which clears out the local repository of retrieved package files. Then all empty space is written with the "\0 character" which makes for very efficient compression.

```
sbe@vb-sbe:~$ sudo leafpad /usr/local/bin/gozero
[sudo] password for sbe:
[Paste in 'gozero' script.]
sbe@vb-sbe:~$ sudo chmod 755 /usr/local/bin/gozero
sbe@vb-sbe:~$ sudo gozero
Cleaning up and zeroing out empty space...

Reading package lists... Done
Building dependency tree
Reading state information... Done
dd: writing `/0bit': No space left on device
183+0 records in
182+0 records out
3828588544 bytes (3.8 GB) copied, 13.9324 s, 275 MB/s
0bit file created. -rw-r--r-- 1 root root 3828588544 2012-03-29 20:55 /0bit...
Removing file... Done

Make sure you compress the .vdi file outside of this VM!
```

wattOSR5 – Maintenance

As with any other Linux distribution, packages are constantly being updating with either security updates or enhancements. For this reason, it is prudent to periodically run an 'apt-get update;apt-get upgrade'. An example session is shown below:

```
sbe@vb-sbe:~$ sudo apt-get update
Ign http://ppa.launchpad.net oneiric InRelease
Ign http://ppa.launchpad.net oneiric InRelease
Ign http://ppa.launchpad.net oneiric InRelease
Ign http://dl.google.com stable InRelease
….
Hit http://us.archive.ubuntu.com oneiric-backports/restricted Translation-en
Hit http://us.archive.ubuntu.com oneiric-backports/universe Translation-en
Ign http://ppa.launchpad.net oneiric/main Translation-en_US
Get:1 http://dl.google.com stable Release.gpg [198 B]
Ign http://ppa.launchpad.net oneiric/main Translation-en
Ign http://ppa.launchpad.net oneiric/main Translation-en_US
Ign http://ppa.launchpad.net oneiric/main Translation-en
Ign http://ppa.launchpad.net oneiric/main Translation-en_US
Ign http://www.remastersys.com oneiric/main Translation-en_US
Ign http://ppa.launchpad.net oneiric/main Translation-en
Ign http://ppa.launchpad.net oneiric/main Translation-en_US
Ign http://ppa.launchpad.net oneiric/main Translation-en
Ign http://ppa.launchpad.net oneiric/main Translation-en_US
Ign http://ppa.launchpad.net oneiric/main Translation-en
Ign http://ppa.launchpad.net oneiric/main Translation-en_US
Ign http://www.remastersys.com oneiric/main Translation-en
Ign http://ppa.launchpad.net oneiric/main Translation-en
Get:2 http://dl.google.com stable Release [1,347 B]
Get:3 http://dl.google.com stable/main i386 Packages [1,237 B]
Ign http://dl.google.com stable/main TranslationIndex
Ign http://dl.google.com stable/main Translation-en_US
Ign http://dl.google.com stable/main Translation-en
Fetched 2,782 B in 1min 2s (44 B/s)
Reading package lists... Done
sbe@vb-sbe:~$ sudo apt-get upgrade
Reading package lists... Done
Building dependency tree
Reading state information... Done
The following packages have been kept back:
  jupiter linux-generic linux-headers-generic linux-image-generic smplayer
The following packages will be upgraded:
  apt apt-transport-https apt-utils flashplugin-installer
  gir1.2-javascriptcoregtk-3.0 gir1.2-webkit-3.0 gstreamer0.10-plugins-good
```

Robert Sorensen, rssoren@gmail.com

```
  gzip jockey-common jockey-gtk language-pack-en language-pack-gnome-en
  libapt-inst1.3 libapt-pkg4.11 libfreetype6 libgudev-1.0-0
  libjavascriptcoregtk-1.0-0 libjavascriptcoregtk-3.0-0 libmysqlclient16
  libnautilus-extension1 libpng12-0 libudev0 libwebkitgtk-1.0-0
  libwebkitgtk-1.0-common libwebkitgtk-3.0-0 libwebkitgtk-3.0-common libxml2
  linux-headers-3.0.0-16 linux-headers-3.0.0-16-generic
  linux-image-3.0.0-16-generic multiarch-support mysql-common python-libxml2
  python-pkg-resources udev xserver-xorg-video-openchrome
36 upgraded, 0 newly installed, 0 to remove and 5 not upgraded.
Need to get 75.3 MB of archives.
After this operation, 1,233 kB of additional disk space will be used.
Do you want to continue [Y/n]?  y
….
Setting up libjavascriptcoregtk-3.0-0 (1.6.3-1~oowkt3) ...
Setting up libwebkitgtk-3.0-common (1.6.3-1~oowkt3) ...
Setting up libwebkitgtk-3.0-0 (1.6.3-1~oowkt3) ...
Setting up gir1.2-javascriptcoregtk-3.0 (1.6.3-1~oowkt3) ...
Setting up gir1.2-webkit-3.0 (1.6.3-1~oowkt3) ...
Setting up gstreamer0.10-plugins-good (0.10.30-1ubuntu7.1) ...
Setting up jockey-common (0.9.4-0ubuntu10.1) ...
Setting up jockey-gtk (0.9.4-0ubuntu10.1) ...
Setting up libwebkitgtk-1.0-common (1.6.3-1~oowkt3) ...
Setting up libjavascriptcoregtk-1.0-0 (1.6.3-1~oowkt3) ...
Setting up libwebkitgtk-1.0-0 (1.6.3-1~oowkt3) ...
Setting up mysql-common (5.1.61-0ubuntu0.11.10.1) ...
Setting up libmysqlclient16 (5.1.61-0ubuntu0.11.10.1) ...
Setting up libnautilus-extension1 (1:3.2.1-0ubuntu4.2) ...
Setting up linux-headers-3.0.0-16 (3.0.0-16.29) ...
Setting up linux-headers-3.0.0-16-generic (3.0.0-16.29) ...
Setting up python-libxml2 (2.7.8.dfsg-4ubuntu0.2) ...
Setting up python-pkg-resources (0.6.16-1ubuntu0.1) ...
Setting up xserver-xorg-video-openchrome (1:0.2.904+svn920-1ubuntu0.2) ...
Processing triggers for libc-bin ...
ldconfig deferred processing now taking place
Processing triggers for initramfs-tools ...
update-initramfs: Generating /boot/initrd.img-3.0.0-16-generic
```

This completes the base installation, configuration and maintenance of wattOSR5 in our VirtualBox SBE VM.

Robert Sorensen, rssoren@gmail.com

Appendix B – Configuration Files

/etc/snort/snort.conf

```
#--------------------------------------------------
#   VRT Rule Packages Snort.conf
#
#   For more information visit us at:
#     http://www.snort.org                   Snort Website
#     http://vrt-sourcefire.blogspot.com/    Sourcefire VRT Blog
#
#     Mailing list Contact:      snort-sigs@lists.sourceforge.net
#     False Positive reports:    fp@sourcefire.com
#     Snort bugs:                bugs@snort.org
#
#     Compatible with Snort Versions:
#     VERSIONS : 2.9.2.2
#
#     Snort build options:
#     OPTIONS : --enable-ipv6 --enable-gre --enable-mpls --enable-targetbased --enable-
    decoder-preprocessor-rules --enable-ppm --enable-perfprofiling --enable-zlib --
    enable-active-response --enable-normalizer --enable-reload --enable-react --enable-
    flexresp3
#
#     Additional information:
#     This configuration file enables active response, to run snort in
#     test mode -T you are required to supply an interface -i <interface>
#     or test mode will fail to fully validate the configuration and
#     exit with a FATAL error
#--------------------------------------------------

##################################################
# This file contains a sample snort configuration.
# You should take the following steps to create your own custom configuration:
#
#  1) Set the network variables.
#  2) Configure the decoder
#  3) Configure the base detection engine
#  4) Configure dynamic loaded libraries
#  5) Configure preprocessors
#  6) Configure output plugins
#  7) Customize your rule set
#  8) Customize preprocessor and decoder rule set
#  9) Customize shared object rule set
##################################################

##################################################
# Step #1: Set the network variables.  For more information, see README.variables
##################################################

# Setup the network addresses you are protecting
ipvar HOME_NET 10.0.2.15/32

# Set up the external network addresses. Leave as "any" in most situations
ipvar EXTERNAL_NET !$HOME_NET

# List of DNS servers on your network
ipvar DNS_SERVERS $HOME_NET

# List of SMTP servers on your network
ipvar SMTP_SERVERS $HOME_NET

# List of web servers on your network
ipvar HTTP_SERVERS $HOME_NET

# List of sql servers on your network
ipvar SQL_SERVERS $HOME_NET

# List of telnet servers on your network
```

```
ipvar TELNET_SERVERS $HOME_NET

# List of ssh servers on your network
ipvar SSH_SERVERS $HOME_NET

# List of ftp servers on your network
ipvar FTP_SERVERS $HOME_NET

# List of sip servers on your network
ipvar SIP_SERVERS $HOME_NET

# List of ports you run web servers on
portvar HTTP_PORTS
     [80,81,311,591,593,901,1220,1414,1830,2301,2381,2809,3128,3702,4343,5250,7001,7145,75
     10,7777,7779,8000,8008,8014,8028,8080,8088,8118,8123,8180,8181,8243,8280,8800,8888,88
     99,9080,9090,9091,9443,9999,11371,55555]

# List of ports you want to look for SHELLCODE on.
portvar SHELLCODE_PORTS !80

# List of ports you might see oracle attacks on
portvar ORACLE_PORTS 1024:

# List of ports you want to look for SSH connections on:
portvar SSH_PORTS 22

# List of ports you run ftp servers on
portvar FTP_PORTS [21,2100,3535]

# List of ports you run SIP servers on
portvar SIP_PORTS [5060,5061,5600]

# List of file data ports for file inspection
portvar FILE_DATA_PORTS [$HTTP_PORTS,110,143]

# List of GTP ports for GTP preprocessor
portvar GTP_PORTS [2123,2152,3386]

# other variables, these should not be modified
ipvar AIM_SERVERS
     [64.12.24.0/23,64.12.28.0/23,64.12.161.0/24,64.12.163.0/24,64.12.200.0/24,205.188.3.0
     /24,205.188.5.0/24,205.188.7.0/24,205.188.9.0/24,205.188.153.0/24,205.188.179.0/24,20
     5.188.248.0/24]

# Path to your rules files (this can be a relative path)
# Note for Windows users:  You are advised to make this an absolute path,
# such as:  c:\snort\rules
var RULE_PATH rules
var SO_RULE_PATH ../so_rules
var PREPROC_RULE_PATH ../preproc_rules

# If you are using reputation preprocessor set these
# Currently there is a bug with relative paths, they are relative to where snort is
# not relative to snort.conf like the above variables
# This is completely inconsistent with how other vars work, BUG 89986
# Set the absolute path appropriately
var WHITE_LIST_PATH ../rules
var BLACK_LIST_PATH ../rules

##################################################
# Step #2: Configure the decoder.  For more information, see README.decode
##################################################

# Stop generic decode events:
config disable_decode_alerts

# Stop Alerts on experimental TCP options
config disable_tcpopt_experimental_alerts

# Stop Alerts on obsolete TCP options
config disable_tcpopt_obsolete_alerts
```

```
# Stop Alerts on T/TCP alerts
config disable_tcpopt_ttcp_alerts

# Stop Alerts on all other TCPOption type events:
config disable_tcpopt_alerts

# Stop Alerts on invalid ip options
config disable_ipopt_alerts

# Alert if value in length field (IP, TCP, UDP) is greater th elength of the packet
# config enable_decode_oversized_alerts

# Same as above, but drop packet if in Inline mode (requires
    enable_decode_oversized_alerts)
# config enable_decode_oversized_drops

# Configure IP / TCP checksum mode
config checksum_mode: all

# Configure maximum number of flowbit references.  For more information, see
    README.flowbits
# config flowbits_size: 64

# Configure ports to ignore
# config ignore_ports: tcp 21 6667:6671 1356
# config ignore_ports: udp 1:17 53

# Configure active response for non inline operation. For more information, see
    REAMDE.active
# config response: eth0 attempts 2

# Configure DAQ related options for inline operation. For more information, see
    README.daq
#
config daq: nfq
config daq_dir: /usr/local/lib/daq
config daq_mode: inline
# config daq_var: <var>
#
# <type> ::= pcap | afpacket | dump | nfq | ipq | ipfw
# <mode> ::= read-file | passive | inline
# <var> ::= arbitrary <name>=<value passed to DAQ
# <dir> ::= path as to where to look for DAQ module so's

# Configure specific UID and GID to run snort as after dropping privs. For more
    information see snort -h command line options
#
# config set_gid:
# config set_uid:

# Configure default snaplen. Snort defaults to MTU of in use interface. For more
    information see README
#
# config snaplen:
#

# Configure default bpf_file to use for filtering what traffic reaches snort. For more
    information see snort -h command line options (-F)
#
# config bpf_file:
#

# Configure default log directory for snort to log to.  For more information see snort -h
    command line options (-l)
#
# config logdir:


##################################################
# Step #3: Configure the base detection engine.  For more information, see  README.decode
```

Robert Sorensen, rssoren@gmail.com

```
##################################################

# Configure PCRE match limitations
config pcre_match_limit: 3500
config pcre_match_limit_recursion: 1500

# Configure the detection engine  See the Snort Manual, Configuring Snort - Includes -
    Config
config detection: search-method ac-split search-optimize max-pattern-len 20

# Configure the event queue.  For more information, see README.event_queue
config event_queue: max_queue 8 log 3 order_events content_length

##################################################
## Configure GTP if it is to be used.
## For more information, see README.GTP
##################################################

# config enable_gtp

##################################################
# Per packet and rule latency enforcement
# For more information see README.ppm
##################################################

# Per Packet latency configuration
#config ppm: max-pkt-time 250, \
#   fastpath-expensive-packets, \
#   pkt-log

# Per Rule latency configuration
#config ppm: max-rule-time 200, \
#   threshold 3, \
#   suspend-expensive-rules, \
#   suspend-timeout 20, \
#   rule-log alert

##################################################
# Configure Perf Profiling for debugging
# For more information see README.PerfProfiling
##################################################

#config profile_rules: print all, sort avg_ticks
#config profile_preprocs: print all, sort avg_ticks

##################################################
# Configure protocol aware flushing
# For more information see README.stream5
##################################################
config paf_max: 16000

##################################################
# Step #4: Configure dynamic loaded libraries.
# For more information, see Snort Manual, Configuring Snort - Dynamic Modules
##################################################

# path to dynamic preprocessor libraries
dynamicpreprocessor directory /usr/local/lib/snort_dynamicpreprocessor/

# path to base preprocessor engine
dynamicengine /usr/local/lib/snort_dynamicengine/libsf_engine.so

# path to dynamic rules libraries
dynamicdetection directory /usr/local/lib/snort_dynamicrules

##################################################
# Step #5: Configure preprocessors
# For more information, see the Snort Manual, Configuring Snort - Preprocessors
##################################################

# GTP Control Channle Preprocessor. For more information, see README.GTP
```

```
# preprocessor gtp: ports { 2123 3386 2152 }

# Inline packet normalization. For more information, see README.normalize
# Does nothing in IDS mode
#preprocessor normalize_ip4
#preprocessor normalize_tcp: ips ecn stream
#preprocessor normalize_icmp4
#preprocessor normalize_ip6
#preprocessor normalize_icmp6

# Target-based IP defragmentation.  For more inforation, see README.frag3
preprocessor frag3_global: max_frags 65536
preprocessor frag3_engine: policy linux detect_anomalies overlap_limit 10
     min_fragment_length 100 timeout 180

# Target-Based stateful inspection/stream reassembly.  For more inforation, see
     README.stream5
preprocessor stream5_global: track_tcp yes, \
   track_udp yes, \
   track_icmp no, \
   max_tcp 262144, \
   max_udp 131072, \
   max_active_responses 1, \
   min_response_seconds 1
preprocessor stream5_tcp: policy linux, detect_anomalies, require_3whs 180, \
   overlap_limit 10, small_segments 3 bytes 150, timeout 180, \
    ports client 21 22 23 25 42 53 79 109 110 111 113 119 135 136 137 139 143 \
        161 445 513 514 587 593 691 1433 1521 2100 3306 6070 6665 6666 6667 6668 6669 \
        7000 8181 32770 32771 32772 32773 32774 32775 32776 32777 32778 32779, \
    ports both 80 81 311 443 465 563 591 593 636 901 989 992 993 994 995 1220 1414 1830
     2301 2381 2809 3128 3702 4343 5250 7907 7001 7145 7510 7802 7777 7779 \
        7801 7900 7901 7902 7903 7904 7905 7906 7908 7909 7910 7911 7912 7913 7914 7915
     7916 \
        7917 7918 7919 7920 8000 8008 8014 8028 8080 8088 8118 8123 8180 8243 8280 8800
     8888 8899 9080 9090 9091 9443 9999 11371 55555
preprocessor stream5_udp: timeout 180

# performance statistics.  For more information, see the Snort Manual, Configuring Snort
     - Preprocessors - Performance Monitor
# preprocessor perfmonitor: time 300 file /var/snort/snort.stats pktcnt 10000

# HTTP normalization and anomaly detection.  For more information, see
     README.http_inspect
preprocessor http_inspect: global iis_unicode_map ./rules/unicode.map 1252 compress_depth
     65535 decompress_depth 65535
preprocessor http_inspect_server: server default \
    http_methods { GET POST PUT SEARCH MKCOL COPY MOVE LOCK UNLOCK NOTIFY POLL BCOPY
     BDELETE BMOVE LINK UNLINK OPTIONS HEAD DELETE TRACE TRACK CONNECT SOURCE SUBSCRIBE
     UNSUBSCRIBE PROPFIND PROPPATCH BPROPFIND BPROPPATCH RPC_CONNECT PROXY_SUCCESS
     BITS_POST CCM_POST SMS_POST RPC_IN_DATA RPC_OUT_DATA RPC_ECHO_DATA } \
    chunk_length 500000 \
    server_flow_depth 0 \
    client_flow_depth 0 \
    post_depth 65495 \
    oversize_dir_length 500 \
    max_header_length 750 \
    max_headers 100 \
    max_spaces 0 \
    small_chunk_length { 10 5 } \
    ports { 80 81 311 591 593 901 1220 1414 1830 2301 2381 2809 3128 3702 4343 5250 7001
     7145 7510 7777 7779 8000 8008 8014 8028 8080 8088 8118 8123 8180 8181 8243 8280 8800
     8888 8899 9080 9090 9091 9443 9999 11371 55555 } \
    non_rfc_char { 0x00 0x01 0x02 0x03 0x04 0x05 0x06 0x07 } \
    enable_cookie \
    extended_response_inspection \
    inspect_gzip \
    normalize_utf \
    unlimited_decompress \
    normalize_javascript \
    apache_whitespace no \
    ascii no \
```

Robert Sorensen, rssoren@gmail.com

```
    bare_byte no \
    directory no \
    double_decode no \
    iis_backslash no \
    iis_delimiter no \
    iis_unicode no \
    multi_slash no \
    utf_8 no \
    u_encode yes \
    webroot no

# ONC-RPC normalization and anomaly detection.  For more information, see the Snort
    Manual, Configuring Snort - Preprocessors - RPC Decode
preprocessor rpc_decode: 111 32770 32771 32772 32773 32774 32775 32776 32777 32778 32779
    no_alert_multiple_requests no_alert_large_fragments no_alert_incomplete

# Back Orifice detection.
preprocessor bo

# FTP / Telnet normalization and anomaly detection.  For more information, see
    README.ftptelnet
preprocessor ftp_telnet: global inspection_type stateful encrypted_traffic no
preprocessor ftp_telnet_protocol: telnet \
    ayt_attack_thresh 20 \
    normalize ports { 23 } \
    detect_anomalies
preprocessor ftp_telnet_protocol: ftp server default \
    def_max_param_len 100 \
    ports { 21 2100 3535 } \
    telnet_cmds yes \
    ignore_telnet_erase_cmds yes \
    ftp_cmds { ABOR ACCT ADAT ALLO APPE AUTH CCC CDUP } \
    ftp_cmds { CEL CLNT CMD CONF CWD DELE ENC EPRT } \
    ftp_cmds { EPSV ESTA ESTP FEAT HELP LANG LIST LPRT } \
    ftp_cmds { LPSV MACB MAIL MDTM MIC MKD MLSD MLST } \
    ftp_cmds { MODE NLST NOOP OPTS PASS PASV PBSZ PORT } \
    ftp_cmds { PROT PWD QUIT REIN REST RETR RMD RNFR } \
    ftp_cmds { RNTO SDUP SITE SIZE SMNT STAT STOR STOU } \
    ftp_cmds { STRU SYST TEST TYPE USER XCUP XCRC XCWD } \
    ftp_cmds { XMAS XMD5 XMKD XPWD XRCP XRMD XRSQ XSEM } \
    ftp_cmds { XSEN XSHA1 XSHA256 } \
    alt_max_param_len 0 { ABOR CCC CDUP ESTA FEAT LPSV NOOP PASV PWD QUIT REIN STOU SYST
    XCUP XPWD } \
    alt_max_param_len 200 { ALLO APPE CMD HELP NLST RETR RNFR STOR STOU XMKD } \
    alt_max_param_len 256 { CWD RNTO } \
    alt_max_param_len 400 { PORT } \
    alt_max_param_len 512 { SIZE } \
    chk_str_fmt { ACCT ADAT ALLO APPE AUTH CEL CLNT CMD } \
    chk_str_fmt { CONF CWD DELE ENC EPRT EPSV ESTP HELP } \
    chk_str_fmt { LANG LIST LPRT MACB MAIL MDTM MIC MKD } \
    chk_str_fmt { MLSD MLST MODE NLST OPTS PASS PBSZ PORT } \
    chk_str_fmt { PROT REST RETR RMD RNFR RNTO SDUP SITE } \
    chk_str_fmt { SIZE SMNT STAT STOR STRU TEST TYPE USER } \
    chk_str_fmt { XCRC XCWD XMAS XMD5 XMKD XRCP XRMD XRSQ } \
    chk_str_fmt { XSEM XSEN XSHA1 XSHA256 } \
    cmd_validity ALLO < int [ char R int ] > \
    cmd_validity EPSV < [ { char 12 | char A char L char L } ] > \
    cmd_validity MACB < string > \
    cmd_validity MDTM < [ date nnnnnnnnnnnnnn[.n[n[n]]] ] string > \
    cmd_validity MODE < char ASBCZ > \
    cmd_validity PORT < host_port > \
    cmd_validity PROT < char CSEP > \
    cmd_validity STRU < char FRPO [ string ] > \
    cmd_validity TYPE < { char AE [ char NTC ] | char I | char L [ number ] } >
preprocessor ftp_telnet_protocol: ftp client default \
    max_resp_len 256 \
    bounce yes \
    ignore_telnet_erase_cmds yes \
    telnet_cmds yes
```

Robert Sorensen, rssoren@gmail.com

```
# SMTP normalization and anomaly detection.  For more information, see README.SMTP
preprocessor smtp: ports { 25 465 587 691 } \
    inspection_type stateful \
    b64_decode_depth 0 \
    qp_decode_depth 0 \
    bitenc_decode_depth 0 \
    uu_decode_depth 0 \
    log_mailfrom \
    log_rcptto \
    log_filename \
    log_email_hdrs \
    normalize_cmds \
    normalize_cmds { ATRN AUTH BDAT CHUNKING DATA DEBUG EHLO EMAL ESAM ESND ESOM ETRN
     EVFY } \
    normalize_cmds { EXPN HELO HELP IDENT MAIL NOOP ONEX QUEU QUIT RCPT RSET SAML SEND
     SOML } \
    normalize_cmds { STARTTLS TICK TIME TURN TURNME VERB VRFY X-ADAT X-DRCP X-ERCP X-
     EXCH50 } \
    normalize_cmds { X-EXPS X-LINK2STATE XADR XAUTH XCIR XEXCH50 XGEN XLICENSE XQUE XSTA
     XTRN XUSR } \
    max_command_line_len 512 \
    max_header_line_len 1000 \
    max_response_line_len 512 \
    alt_max_command_line_len 260 { MAIL } \
    alt_max_command_line_len 300 { RCPT } \
    alt_max_command_line_len 500 { HELP HELO ETRN EHLO } \
    alt_max_command_line_len 255 { EXPN VRFY ATRN SIZE BDAT DEBUG EMAL ESAM ESND ESOM
     EVFY IDENT NOOP RSET } \
    alt_max_command_line_len 246 { SEND SAML SOML AUTH TURN ETRN DATA RSET QUIT ONEX QUEU
     STARTTLS TICK TIME TURNME VERB X-EXPS X-LINK2STATE XADR XAUTH XCIR XEXCH50 XGEN
     XLICENSE XQUE XSTA XTRN XUSR } \
    valid_cmds { ATRN AUTH BDAT CHUNKING DATA DEBUG EHLO EMAL ESAM ESND ESOM ETRN EVFY }
     \
    valid_cmds { EXPN HELO HELP IDENT MAIL NOOP ONEX QUEU QUIT RCPT RSET SAML SEND SOML }
     \
    valid_cmds { STARTTLS TICK TIME TURN TURNME VERB VRFY X-ADAT X-DRCP X-ERCP X-EXCH50 }
     \
    valid_cmds { X-EXPS X-LINK2STATE XADR XAUTH XCIR XEXCH50 XGEN XLICENSE XQUE XSTA XTRN
     XUSR } \
    xlink2state { enabled }

# Portscan detection.  For more information, see README.sfportscan
# preprocessor sfportscan: proto  { all } memcap { 10000000 } sense_level { low }

# ARP spoof detection.  For more information, see the Snort Manual - Configuring Snort -
    Preprocessors - ARP Spoof Preprocessor
# preprocessor arpspoof
# preprocessor arpspoof_detect_host: 192.168.40.1 f0:0f:00:f0:0f:00

# SSH anomaly detection.  For more information, see README.ssh
preprocessor ssh: server_ports { 22 } \
                  autodetect \
                  max_client_bytes 19600 \
                  max_encrypted_packets 20 \
                  max_server_version_len 100 \
                  enable_respoverflow enable_ssh1crc32 \
                  enable_srvoverflow enable_protomismatch

# SMB / DCE-RPC normalization and anomaly detection.  For more information, see
    README.dcerpc2
preprocessor dcerpc2: memcap 102400, events [co ]
preprocessor dcerpc2_server: default, policy WinXP, \
    detect [smb [139,445], tcp 135, udp 135, rpc-over-http-server 593], \
    autodetect [tcp 1025:, udp 1025:, rpc-over-http-server 1025:], \
    smb_max_chain 3, smb_invalid_shares ["C$", "D$", "ADMIN$"]

# DNS anomaly detection.  For more information, see README.dns
preprocessor dns: ports { 53 } enable_rdata_overflow

# SSL anomaly detection and traffic bypass.  For more information, see README.ssl
preprocessor ssl: ports { 443 465 563 636 989 992 993 994 995 7801 7802 7900 7901 7902
```

Robert Sorensen, rssoren@gmail.com

```
     7903 7904 7905 7906 7907 7908 7909 7910 7911 7912 7913 7914 7915 7916 7917 7918 7919
     7920 }, trustservers, noinspect_encrypted

# SDF sensitive data preprocessor.  For more information see README.sensitive_data
preprocessor sensitive_data: alert_threshold 25

# SIP Session Initiation Protocol preprocessor.  For more information see README.sip
preprocessor sip: max_sessions 40000, \
   ports { 5060 5061 5600 }, \
   methods { invite \
             cancel \
             ack \
             bye \
             register \
             options \
             refer \
             subscribe \
             update \
             join \
             info \
             message \
             notify \
             benotify \
             do \
             qauth \
             sprack \
             publish \
             service \
             unsubscribe \
             prack }, \
   max_uri_len 512, \
   max_call_id_len 80, \
   max_requestName_len 20, \
   max_from_len 256, \
   max_to_len 256, \
   max_via_len 1024, \
   max_contact_len 512, \
   max_content_len 2048

# IMAP preprocessor.  For more information see README.imap
preprocessor imap: \
   ports { 143 } \
   b64_decode_depth 0 \
   qp_decode_depth 0 \
   bitenc_decode_depth 0 \
   uu_decode_depth 0

# POP preprocessor. For more information see README.pop
preprocessor pop: \
   ports { 110 } \
   b64_decode_depth 0 \
   qp_decode_depth 0 \
   bitenc_decode_depth 0 \
   uu_decode_depth 0

# Modbus preprocessor. For more information see README.modbus
preprocessor modbus: ports { 502 }

# DNP3 preprocessor. For more information see README.dnp3
preprocessor dnp3: ports { 20000 } \
   memcap 262144 \
   check_crc

# Reputation preprocessor. For more information see README.reputation
#preprocessor reputation: \
#   memcap 500, \
#   priority whitelist, \
#   nested_ip inner, \
#   whitelist $WHITE_LIST_PATH/white_list.rules, \
#   blacklist $BLACK_LIST_PATH/black_list.rules
```

Robert Sorensen, rssoren@gmail.com

```
##################################################
# Step #6: Configure output plugins
# For more information, see Snort Manual, Configuring Snort - Output Modules
##################################################

# unified2
# Recommended for most installs
# output unified2: filename merged.log, limit 128, nostamp, mpls_event_types,
    vlan_event_types

# Additional configuration for specific types of installs
# output alert_unified2: filename snort.alert, limit 128, nostamp
# output log_unified2: filename snort.log, limit 128, nostamp

# syslog
# output alert_syslog: LOG_AUTH LOG_ALERT

# pcap
# output log_tcpdump: tcpdump.log

# database
# output database: alert, <db_type>, user=<username> password=<password> test
    dbname=<name> host=<hostname>
# output database: log, <db_type>, user=<username> password=<password> test dbname=<name>
    host=<hostname>

# prelude
# output alert_prelude

# metadata reference data.  do not modify these lines
include rules/classification.config
include rules/reference.config


##################################################
# Step #7: Customize your rule set
# For more information, see Snort Manual, Writing Snort Rules
#
# NOTE: All categories are enabled in this conf file
##################################################

# site specific rules
include $RULE_PATH/local.rules
include emerging.conf

#include $RULE_PATH/attack-responses.rules
#include $RULE_PATH/backdoor.rules
#include $RULE_PATH/bad-traffic.rules
#include $RULE_PATH/blacklist.rules
#include $RULE_PATH/botnet-cnc.rules
#include $RULE_PATH/chat.rules
#include $RULE_PATH/content-replace.rules
#include $RULE_PATH/ddos.rules
#include $RULE_PATH/dns.rules
#include $RULE_PATH/dos.rules
#include $RULE_PATH/exploit.rules
#include $RULE_PATH/file-identify.rules
#include $RULE_PATH/finger.rules
#include $RULE_PATH/ftp.rules
#include $RULE_PATH/icmp.rules
#include $RULE_PATH/icmp-info.rules
#include $RULE_PATH/imap.rules
#include $RULE_PATH/info.rules
#include $RULE_PATH/misc.rules
#include $RULE_PATH/multimedia.rules
#include $RULE_PATH/mysql.rules
#include $RULE_PATH/netbios.rules
#include $RULE_PATH/nntp.rules
#include $RULE_PATH/oracle.rules
#include $RULE_PATH/other-ids.rules
#include $RULE_PATH/p2p.rules
```

Robert Sorensen, rssoren@gmail.com

```
#include $RULE_PATH/phishing-spam.rules
#include $RULE_PATH/policy.rules
#include $RULE_PATH/pop2.rules
#include $RULE_PATH/pop3.rules
#include $RULE_PATH/rpc.rules
#include $RULE_PATH/rservices.rules
#include $RULE_PATH/scada.rules
#include $RULE_PATH/scan.rules
#include $RULE_PATH/shellcode.rules
#include $RULE_PATH/smtp.rules
#include $RULE_PATH/snmp.rules
#include $RULE_PATH/specific-threats.rules
#include $RULE_PATH/spyware-put.rules
#include $RULE_PATH/sql.rules
#include $RULE_PATH/telnet.rules
#include $RULE_PATH/tftp.rules
#include $RULE_PATH/virus.rules
#include $RULE_PATH/voip.rules
#include $RULE_PATH/web-activex.rules
#include $RULE_PATH/web-attacks.rules
#include $RULE_PATH/web-cgi.rules
#include $RULE_PATH/web-client.rules
#include $RULE_PATH/web-coldfusion.rules
#include $RULE_PATH/web-frontpage.rules
#include $RULE_PATH/web-iis.rules
#include $RULE_PATH/web-misc.rules
#include $RULE_PATH/web-php.rules
#include $RULE_PATH/x11.rules

##################################################
# Step #8: Customize your preprocessor and decoder alerts
# For more information, see README.decoder_preproc_rules
##################################################

# decoder and preprocessor event rules
# include $PREPROC_RULE_PATH/preprocessor.rules
# include $PREPROC_RULE_PATH/decoder.rules
# include $PREPROC_RULE_PATH/sensitive-data.rules

##################################################
# Step #9: Customize your Shared Object Snort Rules
# For more information, see http://vrt-sourcefire.blogspot.com/2009/01/using-vrt-
    certified-shared-object-rules.html
##################################################

# dynamic library rules
# include $SO_RULE_PATH/bad-traffic.rules
# include $SO_RULE_PATH/chat.rules
# include $SO_RULE_PATH/dos.rules
# include $SO_RULE_PATH/exploit.rules
# include $SO_RULE_PATH/icmp.rules
# include $SO_RULE_PATH/imap.rules
# include $SO_RULE_PATH/misc.rules
# include $SO_RULE_PATH/multimedia.rules
# include $SO_RULE_PATH/netbios.rules
# include $SO_RULE_PATH/nntp.rules
# include $SO_RULE_PATH/p2p.rules
# include $SO_RULE_PATH/smtp.rules
# include $SO_RULE_PATH/snmp.rules
# include $SO_RULE_PATH/specific-threats.rules
# include $SO_RULE_PATH/web-activex.rules
# include $SO_RULE_PATH/web-client.rules
# include $SO_RULE_PATH/web-iis.rules
# include $SO_RULE_PATH/web-misc.rules

# Event thresholding or suppression commands. See threshold.conf
include threshold.conf
```

Robert Sorensen, rssoren@gmail.com

/etc/oinkmaster.conf

```
# $Id: oinkmaster.conf,v 1.134 2008/02/18 19:33:45 andreas_o Exp $ #

# This file is pretty big by default, but don't worry.
# Everything in here is completely optional and the defaults
# should work for most people. The download URL of the rules
# archive must be set either in here or on the command line.

# Remember not to let untrusted users edit Oinkmaster configuration
# files, as things like the PATH to use during execution is defined
# in here.

# Use "url = <url>" to specify the location of the rules archive to
# download. The url must begin with http://, https://, ftp://, file://
# or scp:// and end with .tar.gz or .tgz, and the file must be a
# gzipped tarball what contains a directory named "rules".
# You can also point to a local directory with dir://<directory>.
# Multiple "url = <url>" lines can be specified to grab multiple rules
# archives from different locations.
#
# Note: if URL is specified on the command line, it overrides all
#       possible URLs specified in the configuration file(s).
#
# The location of the official Snort rules you should use depends
# on which Snort version you run. Basically, you should go to
# http://www.snort.org/rules/ and follow the instructions
# there to pick the right URL for your version of Snort
# (and remember to update the URL when upgrading Snort in the
# future!). You can of course also specify locations to third party
# rules. You may specify multiple URLs.
#
# As of March 2005, you must register on the Snort site to get access
# to the official Snort rules. This will get you an "oinkcode".
# You then specify the URL as
# http://www.snort.org/pub-bin/oinkmaster.cgi/<oinkcode>/<filename>
# For example, if your code is 5a081649c06a277e1022e1284b and
# you use Snort 2.7, the url to use would be:
# http://www.snort.org/pub-
    bin/oinkmaster.cgi/5a081649c06a277e1022e1284bdc8fabda70e2a4/snortrules-snapshot-
    2.7.tar.gz
# See the Oinkmaster FAQ Q1 and http://www.snort.org/rules/ for
# more information.

# URL examples follows. Replace <oinkcode> with the code you get on the
# Snort site in your registered user profile.
url = http://rules.emergingthreats.net/open/snort-2.9.0/emerging.rules.tar.gz


# VRT certified rules for registered users, Snort 2.7.
# url = http://www.snort.org/pub-bin/oinkmaster.cgi/<oinkcode>/snortrules-snapshot-
    2.7.tar.gz

# VRT certified rules for registered users, Snort 2.8.
# url = http://www.snort.org/pub-bin/oinkmaster.cgi/<oinkcode>/snortrules-snapshot-
    2.8.tar.gz

# VRT certified rules for registered users, Snort-CURRENT
# ("CURRENT" here means experimental snapshots!).
# url = http://www.snort.org/pub-bin/oinkmaster.cgi/<oinkcode>/snortrules-snapshot-
    CURRENT.tar.gz


# Community rules and Snort 2.4.
# url = http://www.snort.org/pub-bin/downloads.cgi/Download/comm_rules/Community-Rules-
    2.4.tar.gz

# Community rules for snort-CURRENT
# url = http://www.snort.org/pub-bin/downloads.cgi/Download/comm_rules/Community-Rules-
    CURRENT.tar.gz
```

Robert Sorensen, rssoren@gmail.com

```
# Example for rules from the Emerging Threats site (previously known as Bleeding Snort).
# url = http://www.emergingthreats.net/rules/emerging.rules.tar.gz
# Old url:
# url = http://www.emergingthreats.net/rules/bleeding.rules.tar.gz

# If you prefer to download the rules archive from outside Oinkmaster,
# you can then point to the file on your local filesystem by using
# file://<filename>, for example:
# url = file:///tmp/snortrules.tar.gz

# In rare cases you may want to grab the rules directly from a
# local directory (don't confuse this with the output directory).
# url = dir:///etc/snort/src/rules

# Example to use scp to copy the rules archive from another host.
# Only OpenSSH is tested. See the FAQ for more information.
# url = scp://user@somehost.example.com:/somedir/snortrules.tar.gz

# If you use -u scp://... and need to specify a private ssh key (passed
# as -i <key> to the scp command) you can specify it here or add an
# entry in ~/.ssh/config for the Oinkmaster user as described in the
# OpenSSH manual.
# scp_key = /home/oinkmaster/oinkmaster_privkey


# The PATH to use during execution. If you prefer to use external
# binaries (i.e. use_external_bins=1, see below), tar and gzip must be
# found, and also wget if downloading via ftp, http or https. All with
# optional .exe suffix. If you're on Cygwin, make sure that the path
# contains the Cygwin binaries and not the native Win32 binaries or
# you will get problems.
# The following UNIX style path is assumed by default:
# path = /bin:/usr/bin:/usr/local/bin

# Example if running native Win32 or standalone Cygwin:
# path = c:\oinkmaster;c:\oinkmaster\bin

# Example if running standalone Cygwin and you prefer Cygwin style path:
# path = /cygdrive/c/oinkmaster:/cygdrive/c/oinkmaster/bin

# We normally use external binaries (wget, tar and gzip) since they're
# already available on most systems and do a good job. If you have the
# Perl modules Archive::Tar, IO::Zlib and LWP::UserAgent, you can use
# those instead if you like. You can set use_external_bins below to
# choose which method you prefer. It's set to 0 by default on Win32
# (i.e. use Perl modules), and 1 on other systems (i.e. use external
# binaries). The reason for that is that the required Perl modules
# are included on Windows/ActivePerl 5.8.1+, so it's easier to use
# those than to install the ported Unix tools. (Note that if you're
# using scp to download the archive, external scp  binary is still
# used.)
# use_external_bins = 0


# Temporary directory to use. This directory must exist when starting and
# Oinkmaster will then create a temporary sub directory in here.
# Keep it as a #comment if you want to use the default.
# The default will be checked for in the environment variables TMP,
# TMPDIR or TEMPDIR, or otherwise use "/tmp" if none of them was set.

# Example for UNIX.
# tmpdir = /home/oinkmaster/tmp/

# Example if running native Win32 or Cygwin.
# tmpdir = c:\tmp

# Example if running Cygwin and you prefer Cygwin style path.
# tmpdir = /cygdrive/c/tmp

# The umask to use during execution if you want it to be something
```

```
# else than the current value when starting Oinkmaster.
# This will affect the mode bits when writing new files.
# Keep it commented out to keep your system's current umask.
# umask = 0027

# Files in the archive(s) matching this regular expression will be
# checked for changes, and then updated or added if needed.
# All other files will be ignored. You can then choose to skip
# individual files by specifying the "skipfile" keyword below.
# Normally you shouldn't need to change this one.
# update_files = \.rules$|\.config$|\.conf$|\.txt$|\.map$

# Regexp of keywords that starts a Snort rule.
# May be useful if you create your own ruletypes and want those
# lines to be regarded as rules as well.
# rule_actions = alert|drop|log|pass|reject|sdrop|activate|dynamic

# If the number of rules files in the downloaded archive matching the
# 'update_files' regexp is below min_files, or if the number
# of rules is below min_rules, the rules are regarded as broken
# and the update is aborted with an error message.
# Both are set to 1 by default (i.e. the archive is only regarded as
# broken if it's totally empty).
# If you download from multiple URLs, the count is the total number
# of files/rules across all archives.
# min_files = 1
# min_rules = 1

# By default, a basic sanity check is performed on most paths/filenames
# to see if they contain illegal characters that may screw things up.
# If this check is too strict for your system (e.g. you get bogus
# "illegal characters in filename" errors because of your local language
# etc) and you're sure you want to disable the checks completely,
# set use_path_checks to 0.
# use_path_checks = 1

# If you want Oinkmaster to send a User-Agent HTTP header string
# other than the default one for wget/LWP, set this variable.
# user_agent = Mozilla/4.0 (compatible; MSIE 6.0; Windows NT 5.1)

# The normal Snort rules usually resides in a directory called
# "rules" in the downloaded archive. You can tell Oinkmaster to
# look in another directory by setting the rules_dir statement.
# This allows you to update the shared object rules (so_rules) with
# Oinkmaster by creating a separate oinkmaster.conf containing
# "rules_dir = so_rules". Note that you can not set multiple
# directories so you have to run Oinkmaster separately for each
# directory. Remember to point to different output directories as
# the "rules" and "so_rules" directories contains files with
# identical filenames.
# To update the shared object rules, use:
# rules_dir = so_rules
# The default is to update the normal rules:
# rules_dir = rules

# You can include other files anywhere in here by using
# "include <file>". <file> will be parsed just like a regular
# oinkmaster.conf as soon as the include statement is seen, and then
# return and continue parsing the rest of the original file. If an
# option is redefined, it will override the previous value. You can use
# as many "include" statements as you wish, and also include even more
# files from included files. Example to load stuff from "/etc/foo.conf".
# include /etc/foo.conf



##############################################################################
# Files to totally skip (i.e. never update or check for changes)             #
#                                                                            #
# Syntax: skipfile filename                                                  #
# or:     skipfile filename1, filename2, filename3, ...                      #
```

Robert Sorensen, rssoren@gmail.com

```
#####################################################################

# Ignore local.rules from the rules archive by default since we might
# have put some local rules in our own local.rules and we don't want it
# to get overwritten by the empty one from the archive after each
# update.
skipfile local.rules

# The file deleted.rules contains rules that have been deleted from
# other files, so there is usually no point in updating it.
skipfile deleted.rules

# Also skip snort.conf by default since we don't want to overwrite our
# own snort.conf if we have it in the same directory as the rules. If
# you have your own production copy of snort.conf in another directory,
# it may be really nice to check for changes in this file though,
# especially since variables are sometimes added or modified and
# new/old files are included/excluded.
skipfile snort.conf
skipfile emerging.conf

# You may want to consider ignoring threshold.conf for the same reasons
# as for snort.conf, i.e. if you customize it locally and don't want it
# to become overwritten by the default one. It may be better to put
# local thresholding/suppressing in some local file and still update
# and use the official one though, in case important stuff is added to
# it some day. We do update it by default, but it's your call.
skipfile threshold.conf

# If you update from multiple URLs at the same time you may need to
# ignore the sid-msg.map (and generate it yourself if you need one) as
# it's usually included in each rules tarball. See the FAQ for more info.
# skipfile sid-msg.map

#####################################################################
# SIDs to modify after each update (only for the skilled/stupid/brave).  #
# Don't use it unless you have to. There is nothing that stops you from  #
# modifying rules in such ways that they become invalid or generally     #
# break things. You have been warned.                                    #
# If you just want to disable SIDs, please skip this section and have a  #
# look at the "disablesid" keyword below.                                #
#                                                                        #
# You may specify multiple modifysid directives for the same SID (they   #
# will be processed in order of appearance), and you may also specify a  #
# list of SIDs on which the substitution should be applied.              #
# If the argument is in the form something.something it's regarded       #
# as a filename and the substitution will apply on all rules in that     #
# file. The wildcard ("*") can be used to apply the substitution on all  #
# rules regardless of the SID or file. Please avoid using #comments      #
# at the end of modifysid lines, they may confuse the parser in some     #
# situations.                                                            #
#                                                                        #
# Syntax:                                                                #
#   modifysid SID "replacethis" | "withthis"                             #
# or:                                                                    #
#   modifysid SID1, SID2, SID3, ... "replacethis" | "withthis"           #
# or:                                                                    #
#   modifysid file "replacethis" | "withthis"                            #
# or:                                                                    #
#   modifysid * "replacethis" | "withthis"                               #
#                                                                        #
# The strings within the quotes will basically be passed to a            #
# s/replacethis/withthis/ statement in Perl, so they must be valid       #
# regular expressions. The strings are case-insensitive and only the     #
# first occurrence will be replaced. If there are multiple occurrences   #
# you want to replace, simply repeat the same modifysid line.            #
# As the strings are regular expressions, you MUST escape special        #
# characters like $ \ / ( ) | by prepending a "\" to them.               #
#                                                                        #
# If you specify a modifysid statement for a multi-line rule, Oinkmaster #
# will first translate the rule into a single-line version and then      #
```

```
# perform the substitution, so you don't have to care about the trailing #
# backslashes and newlines.                                              #
#                                                                        #
# If you use backreference variables in the substitution expression,     #
# it's strongly recommended to specify them as ${1} instead of $1 and so #
# on, to avoid parsing confusion with unexpected results in some         #
# situations. Note that modifysid statements will process both active    #
# and inactive (disabled) rules.                                         #
#                                                                        #
# You may want to check out README.templates and template-examples.conf  #
# to find how you can simplify the modifysid usage by using templates.   #
##########################################################################

# Example to enable a rule (in this case SID 1325) that is disabled by
# default, by simply replacing leading "#alert" with "alert".
# (You should really use 'enablesid' for this though.)
# Oinkmaster removes whitespaces next to the leading "#" so you don't
# have to worry about that, but be careful about possible whitespace in
# other places when writing the regexps.
# modifysid 1325 "^#alert" | "alert"

# You could also do this to enable it no matter what type of rule it is
# (alert, log, pass, etc).
# modifysid 1325 "^#" | ""

# Example to enable ALL rules in ALL files (usually not a good idea).
# modifysid * "^#" | ""

# Example to add "tag" stuff to SID 1325.
# modifysid 1325 "sid:1325;" | "sid:1325; tag: host, src, 300, seconds;"

# Example to make SID 1378 a 'drop' rule (valid if you're running
# Snort_inline).
# modifysid 1378 "^alert" | "drop"

# Example to replace first occurrence of $EXTERNAL_NET with $HOME_NET
# in SID 302.
# modifysid 302 "\$EXTERNAL_NET" | "\$HOME_NET"

# You can also specify that a substitution should apply on multiple SIDs.
# modifysid 302,429,1821 "\$EXTERNAL_NET" | "\$HOME_NET"

# You can take advantage of the fact that it's regular expressions and
# do more complex stuff. This example (for Snort_inline) adds a 'replace'
# statement to SID 1324 that replaces "/bin/sh" with "/foo/sh".
# modifysid 1324 "(content\s*:\s*"\/bin\/sh"\s*;)" | \
#                "${1} replace:"\/foo\/sh";"

# If you for some reason would like to add a comment inside the actual
# rules file, like the reason why you disabled this rule, you can do
# like this (you would normally add such comments in oinkmaster.conf
# though).
# modifysid 1324 "(.+)" | "# 20020101: disabled this rule just for fun:\n#${1}"

# Here is an example that is actually useful. Let's say you don't care
# about incoming welchia pings (detected by SID 483 at the time of
# writing) but you want to know when infected hosts on your network
# scans hosts on the outside. (Remember that watching for outgoing
# malicious packets is often just as important as watching for incoming
# ones, especially in this case.) The rule currently looks like
# "alert icmp $EXTERNAL_NET any -> $HOME_NET any ..."
# but we want to switch that so it becomes
# "alert icmp $HOME_NET any -> $EXTERNAL_NET any ...".
# Here is how it could be done.
# modifysid 483 \
# "(.+) \$EXTERNAL_NET (.+) \$HOME_NET (.+)" | \
# "${1} \$HOME_NET ${2} \$EXTERNAL_NET ${3}"

# The wildcard (modifysid * ...) can be used to do all kinds of
# interesting things. The substitution expression will be applied on all
# matching rules. First, a silly example to replace "foo" with "bar" in
```

```
# all rules (that have the string "foo" in them, that is.)
# modifysid * "foo" | "bar"

# If you for some reason don't want to use the stream preprocessor to
# match established streams, you may want to replace the 'flow'
# statement with 'flags:A+;' in all those rules.
# modifysid * "flow:[a-z,_ ]+;" | "flags:A+;"

# Example to convert all rules of classtype attempted-admin to 'drop'
# rules (for Snort_inline only, obviously).
# modifysid * "^alert (.*classtype\s*:\s*attempted-admin)" | "drop ${1}"

# This one will append some text to the 'msg' string for all rules that
# have the 'tag' keyword in them.
# modifysid * "(.*msg:\s*".+?)"(\s*;.+;\s*tag:.*)" | \
#             "${1}, going to tag this baby"${2}"

# There may be times when you want to replace multiple occurrences of a
# certain keyword/string in a rule and not just the first one. To
# replace the first two occurrences of "foo" with "bar" in SID 100,
# simply repeat the modifysid statement:
# modifysid 100 "foo" | "bar"
# modifysid 100 "foo" | "bar"
modifysid * "^alert" | "reject"
modifysid * "msg:\"" | "msg:\"Rejected! "

modifysid emerging-rbn.rules \
"(.+) \[(.+)\] (.+) \$HOME_NET (.+)" | \
"${1} \$HOME_NET ${3} \[${2}\] ${4}"

modifysid emerging-rbn-malvertisers.rules \
"(.+) tcp (.+) any \-\> \$HOME_NET (.+)" | \
"${1} tcp \$HOME_NET any \-\> ${2} ${3}"

modifysid emerging-rbn-malvertisers.rules \
"(.+) udp (.+) any \-\> \$HOME_NET (.+)" | \
"${1} udp \$HOME_NET any \-\> ${2} ${3}"

# Or you can even specify a SID list but repeat the same SID as many
# times as required, like:
# modifysid 100,100,100 "foo" | "bar"

# Enable all rules in the file exploit.rules.
# modifysid exploit.rules "^#" | ""

# Enable all rules in exploit.rules, icmp-info.rules and also SID 1171.
# modifysid exploit.rules, snmp.rules, 1171 "^#" | ""

###########################################################################
# SIDs that we don't want to update.                                      #
# If you for some reason don't want a specific rule to be updated         #
# (e.g. you made local modifications to it and you never want to          #
# update it and don't care about changes in the official version), you    #
# can specify a "localsid" statement for it. This means that the old      #
# version of the rule (i.e. the one in the rules file on your             #
# harddrive) is always kept, regardless if the official version has       #
# been updated. Please do not use this feature unless in special          #
# cases as it's easy to end up with many signatures that aren't           #
# maintained anymore. See the FAQ for details about this and hints        #
# about better solutions regarding customization of rules.                #
#                                                                         #
# Syntax:  localsid SID                                                   #
# or:      localsid SID1, SID2, SID3, ...                                 #
###########################################################################
# Example to never update SID 1325.
# localsid 1325


###########################################################################
# SIDs to enable after each update.                                       #
# Will simply remove all the leading '#' for a specified SID (if it's  #
```

Robert Sorensen, rssoren@gmail.com

```
# a multi-line rule, the leading '#' for all lines are removed.)       #
# These will be processed after all the modifysid and disablesid       #
# statements. Using 'enablesid' on a rule that is not disabled is a    #
# NOOP.                                                                #
#                                                                      #
# Syntax:  enablesid SID                                               #
# or:      enablesid SID1, SID2, SID3, ...                             #
########################################################################

# Example to enable SID 1325.
# enablesid 1325

########################################################################
# SIDs to comment out, i.e. disable, after each update by placing a    #
# '#' in front of the rule (if it's a multi-line rule, it will be put  #
# in front of all lines).                                              #
#                                                                      #
# Syntax:  disablesid SID                                              #
# or:      disablesid SID1, SID2, SID3, ...                            #
########################################################################

# You can specify one SID per line.
# disablesid 1
# disablesid 2
# disablesid 3

# And also as comma-separated lists.
# disablesid 4,5,6
disablesid 1390,2013504

# It's a good idea to also add comment about why you disable the sid:
# disablesid 1324    # 20020101: disabled this SID just because I can
```

/etc/snort/emerging.conf

```
#
# Emerging Threats Configuration Include
#
#       This file is intended to be added to your snort.conf as an include.
#       The intention is to make sure that any specific variables and the
#       like are included in your instance of snort.
#
#       Add a line like this to your snort.conf, or just use this file to
#       decide which variables to add to your own snort.conf:
#
#       include $RULE_PATH/emerging.conf
#
#      This file is valid for both Emerging Threats open and ET Pro rulesets
#
#
# More information available at www.emergingthreats.net or
#   www.emergingthreatspro.com
#
#
################################################################################


# This var is required for several sigs in the POLICY ruleset. It is plural because you
can do a range of ports
#var SSH_PORTS 22

#These vars are required if you're using the Digitalbond Scada signatures in the
scada.rules category
#var DNP3_SERVER $HOME_NET
#var DNP3_CLIENT $HOME_NET
#var DNP3_PORTS 20000
#var MODBUS_CLIENT $HOME_NET
#var MODBUS_SERVER $HOME_NET
```

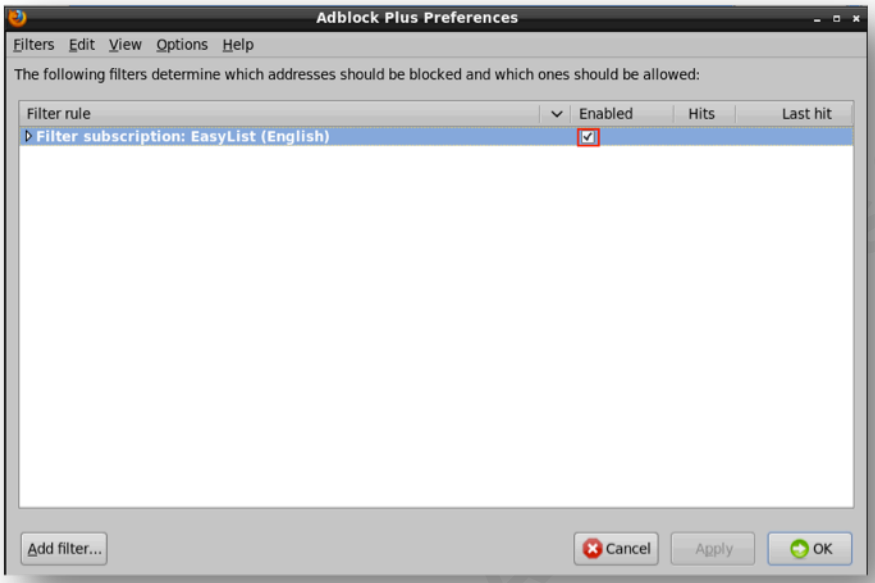
Robert Sorensen, rssoren@gmail.com

```
#var ENIP_CLIENT $HOME_NET
#var ENIP_SERVER $HOME_NET

#include $RULE_PATH/classification.config
#include $RULE_PATH/reference.config

#include $RULE_PATH/emerging-ftp.rules
include $RULE_PATH/emerging-policy.rules
include $RULE_PATH/emerging-trojan.rules
include $RULE_PATH/emerging-games.rules
#include $RULE_PATH/emerging-pop3.rules
include $RULE_PATH/emerging-user_agents.rules
#include $RULE_PATH/emerging-activex.rules
#include $RULE_PATH/emerging-rpc.rules
include $RULE_PATH/emerging-virus.rules
include $RULE_PATH/emerging-attack_response.rules
include $RULE_PATH/emerging-icmp.rules
#include $RULE_PATH/emerging-scan.rules
#include $RULE_PATH/emerging-scada.rules
#include $RULE_PATH/emerging-voip.rules
#include $RULE_PATH/emerging-chat.rules
include $RULE_PATH/emerging-icmp_info.rules
include $RULE_PATH/emerging-shellcode.rules
include $RULE_PATH/emerging-web_client.rules
#include $RULE_PATH/emerging-imap.rules
#include $RULE_PATH/emerging-web_server.rules
include $RULE_PATH/emerging-current_events.rules
include $RULE_PATH/emerging-inappropriate.rules
#include $RULE_PATH/emerging-smtp.rules
##include $RULE_PATH/emerging-web_specific_apps.rules
include $RULE_PATH/emerging-deleted.rules
include $RULE_PATH/emerging-malware.rules
#include $RULE_PATH/emerging-snmp.rules
include $RULE_PATH/emerging-worm.rules
include $RULE_PATH/emerging-dns.rules
include $RULE_PATH/emerging-misc.rules
#include $RULE_PATH/emerging-sql.rules
include $RULE_PATH/emerging-dos.rules
#include $RULE_PATH/emerging-netbios.rules
include $RULE_PATH/emerging-telnet.rules
include $RULE_PATH/emerging-exploit.rules
include $RULE_PATH/emerging-p2p.rules
include $RULE_PATH/emerging-tftp.rules
include $RULE_PATH/emerging-mobile_malware.rules

include $RULE_PATH/emerging-botcc.rules
#include $RULE_PATH/emerging-botcc-BLOCK.rules
include $RULE_PATH/emerging-compromised.rules
#include $RULE_PATH/emerging-compromised-BLOCK.rules
include $RULE_PATH/emerging-drop.rules
#include $RULE_PATH/emerging-drop-BLOCK.rules
include $RULE_PATH/emerging-dshield.rules
#include $RULE_PATH/emerging-dshield-BLOCK.rules
include $RULE_PATH/emerging-rbn.rules
include $RULE_PATH/emerging-rbn-malvertisers.rules
#include $RULE_PATH/emerging-rbn-BLOCK.rules
#include $RULE_PATH/emerging-rbn-malvertisers-BLOCK.rules
include $RULE_PATH/emerging-tor.rules
#include $RULE_PATH/emerging-tor-BLOCK.rules
include $RULE_PATH/emerging-ciarmy.rules
```

Appendix C – Browser Add-on/Extensions Guide

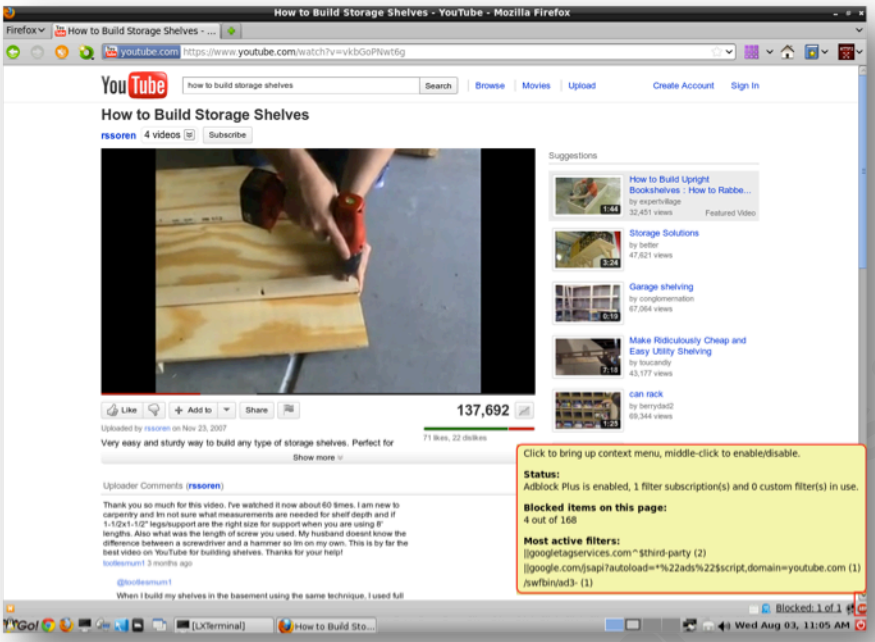
Firefox Add-ons Configuration

AdBlock Plus



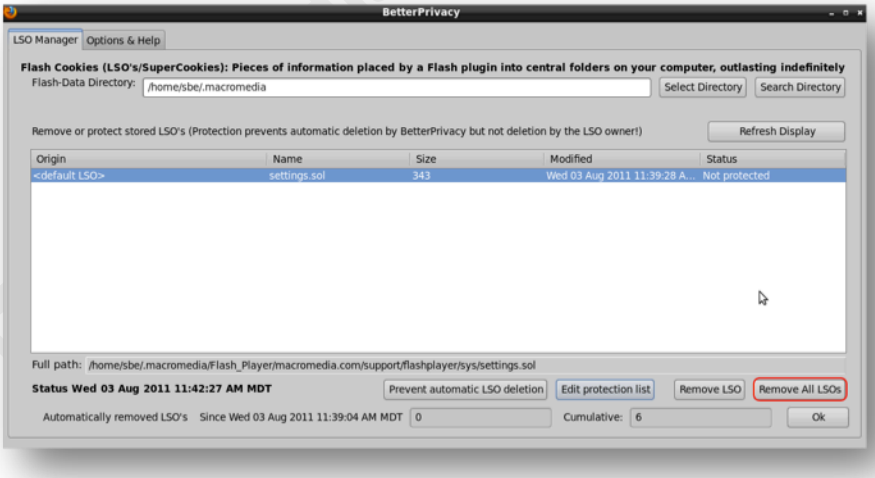
- Accept the default '**Filter subscription: EasyList (English) '**, Check '**Enabled**' box.

Robert Sorensen, rssoren@gmail.com



- An example of 'AdBlock Plus' in action of Youtube video being played. Notice the items blocked on this page.

Better Privacy

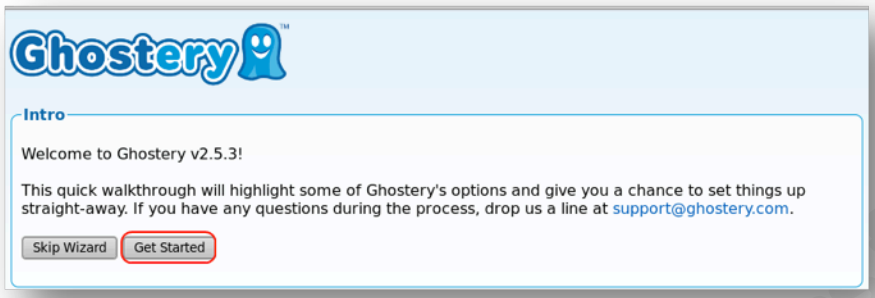


- BetterPrivacy settings to remove all LSO's/SuperCookies. Highlight the LSOs, then click '**Remove All LSO's**.

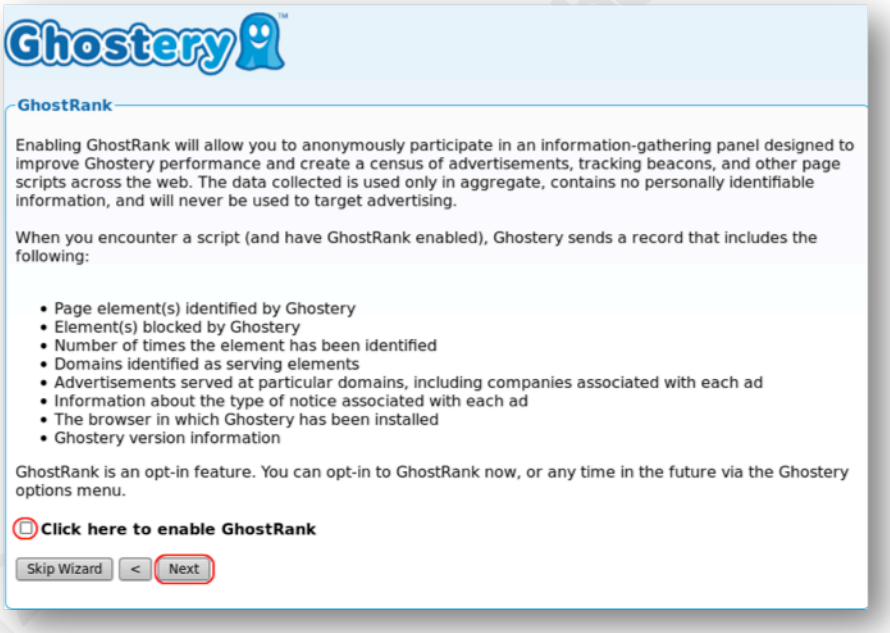
Robert Sorensen, rssoren@gmail.com

Ghostery

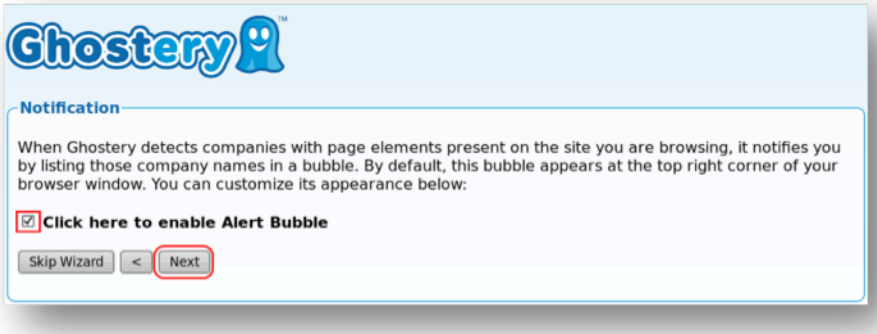
Best way to configure Ghostery is through the Configuration Wizard.



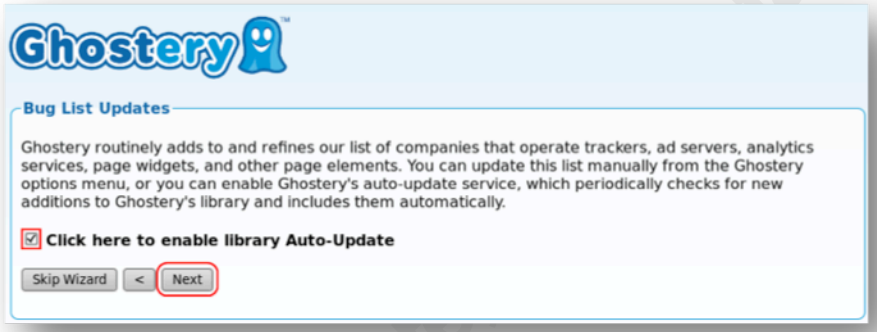
- Go to Firefox Add-ons -> Ghostery Preference -> Ghostery Configuration Wizard. Click '**Get Started**'.



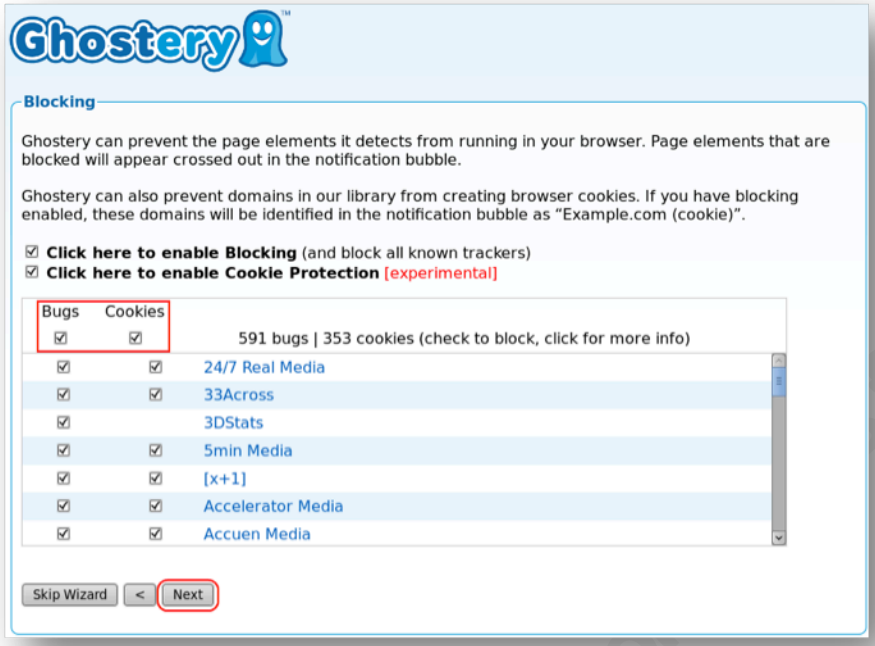
- Your preference to support GhostRank or not. Accept default settings of not enabling GhostRank by not selecting '**Click here to enable GhostRank**'. Click '**Next**'



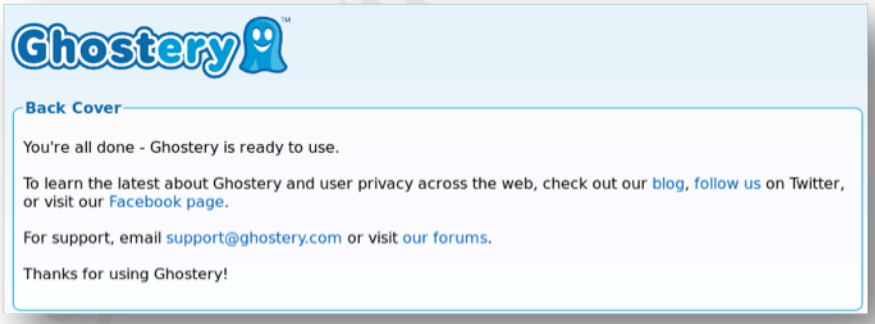
- Notification – Check '**Click here to enable Alert Bubble**'. Click '**Next**'.



- Bug List Updates – Check '**Click here to enable library Auto-Update**'. Click '**Next**'.



- Blocking – Check '**Bugs and Cookies**'. Click '**Next**'.



- Ghoster is ready to use.