|  |  |
| --- | --- |
|  | EUROPEAN COMMISSION  DIRECTORATE-GENERAL INFORMATICS  Directorate S - IT Security  **DIGIT S2 - IT Security Operations** |

CSIRC Forensic Lab  
Linux Template VM

|  |  |
| --- | --- |
| Date: |  |
| Version: | .02 |
| Authors: |  |
| Revised by: | Bertrand Varlet |
| Approved by: |  |
| Public: |  |
| Reference Number: |  |

Document History

|  |  |  |  |
| --- | --- | --- | --- |
| Version | Date | Comment | Modified Pages |
| 0.01 | 01/09/2017 | Document created by Bertrand Varlet |  |
| 0.02 | 08/12/2017 | Document updated by Bertrand Varlet |  |

Contact:  
Bertrand Varlet, bertrand.varlet@ext.ec.europa.eu

Table of Contents

[1. Introduction 4](#_Toc500492738)

[2. Template Creation & Configuration 5](#_Toc500492739)

[2.1. Hardware configuration 5](#_Toc500492740)

[2.2. Configuration 5](#_Toc500492741)

[2.2.1. CreateCustomUbuntuLiveCDScript.sh 5](#_Toc500492742)

[3. Adding new applications to the installation script 8](#_Toc500492743)

[4. Future Development 9](#_Toc500492744)

[4.1. VM Configuration 9](#_Toc500492745)

[4.2. Application installation 9](#_Toc500492746)

[5. List of Installed Applications 10](#_Toc500492747)

# Introduction

This document will cover all the steps taken to create, configure and maintain the CSIRC Forensic Lab Linux Template Virtual Machine (VM).

# Template Creation & Configuration

## Hardware configuration

The CSIRC Linux Template VM will be running on an Ubuntu 16.04 LTS operating system.

The "hardware" specifications are as followed:

* CPU: 1 (4 cores)
* RAM: 8 GB
* HDD: 100 GB
  + Single File
  + Used for the System

A standard installation of Ubuntu 16.04 LTS will serve as the base OS for the CSIRC Linux Template VM.

## Configuration

The CSIRC Linux Template VM is generated via a live CD. This Live CD is in turn generated via a bash script. To use the bash script, we need to run it from a Ubuntu 16.04 system. This can be a temporary VM, or any other system running with Ubuntu 16.04 and which has access to the CSIRC Repository Server.

The scrip used to generate the Live CD is name: CreateCustomUbuntuLiveCDScript.sh

### CreateCustomUbuntuLiveCDScript.sh

#### Abstract

This is bash script which will execute a series of commands which will lead to the creation of a custom Ubuntu ISO.

The script requires we use a base ISO file from Ubuntu and that it is ran from a Ubuntu 16.04 system.

Once the ISO file is created, we can use it to create a new VM, or "burn" it to a CD/DVD/USB and use it as a live response "Live CD"

#### Usage

1. Boot a Ubuntu 16.04 System (VM or Barebone)
   1. The system must have access to the Repository Server
2. Download the script from the CSIRC Repository Server:
   1. <http://RepoForIP/manual/script/CreateCustomUbuntuLiveCDScript.sh>
3. Save the script on the system (e.g: in the download folder)
4. Open a Terminal window
5. Type the following commands:
   1. Sudo chmod +x CreateCustomUbuntuLiveCDScript.sh
   2. Sudo ./CreateCustomUbuntuLiveCDScript.sh
6. Wait for the script to finish (~10 min)
7. Copy the ISO to the ESXI Storage
8. Create a new VM, select the hardware components, and select the ISO we just created in the CD/DVD reader of the VM.
9. Boot the VM
   1. Select to boot on the ISO
   2. Choose "Install Ubuntu"
10. Edit the file /etc/apt/sources.list
    1. Change the values from archive.ubuntu.be to RepoForIP
11. Run the commands:
    1. Sudo apt update
    2. Sudo apt upgrade -y & dist-upgrade -y
12. The VM is ready.

#### Script In's & Out's

The script is composed of 3 elements:

1. Variable definition (Preparation work).
2. Customising the ISO.
3. Generating the ISO.

In the first element, there are 2 variables we must update in the script:

* A few dependencies need to be installed before we can continue:
  + Unzip
  + xorriso
* CDIMAGENAME: the name of the ISO file we will use as a basis for the creation of our custom ISO.
* IMAGE\_NAME: the name of the output custom ISO we will generate.

Starting the section "#Customisation" is where we will the script to customise the ISO. (2nd element)

* The "#Update and Upgrade (Distribution)" section is modifying the sources.list in /etc/apt to have the script use our repository server for the download and installation of applications.
* The "#Download the signing key of repo's and add it to apt-key list" section is where we will download the signing keys for private PPA's and add them to the system key list to authorize download and installation of tools from these PPA's.
* The "#Purge Unwanted applications" section is where we will remove applications from the custom ISO.
  + For instance, we removed the games from the gnome desktop environment, the mail client, etc…
* The "#Install of Tool" sections are section with the specific bash commands to install a specific tool.
  + These sections are for tools who cannot be installed in a single command.
* The "#APT install" section is where we specify the commands to install applications via "apt install"
* The "#pip installs" section is where we specify the commands to install applications via "pip install"
* The "Customise gnome settings" section is where we change some specifics parameters of gnome.
  + For instance the configuration to avoid auto mount of drives when using the ISO as a live CD
* The "#Clean up" section is where we specify the commands we will use to remove any installation files from the final customised ISO.

The 3rd element is the commands to generate the actual customised ISO file.

* We do no modify this element of the script.

# Adding new applications to the installation script

To add a new application to install in the script, you will need to ensure the following is done:

1. On the CSIRC Repository Server
   1. Download the installation files of the application
      * For APT you can update apt-mirror
      * For PIP you will need to download it with pip download
      * Any other, you'll need to wget the source
   2. Ensure you save the files in the corresponding folder/sub folder tree structure
      * Refer to the document: "CSIRC Repository Server" for details.
2. In the createCustomIso.sh script:
   1. In the customize section, type the command line that you would normally type on a live linux system to install the application E.G:
      * For APT: apt install appName
      * For PIP: pip install appName –no-index –find-links=http://RepoForIP/pip –trusted-host RepoForIP
3. Execute the createCustomIso.sh script
   1. It'll take at least 10 min, and in the end produce a Live CD ISO.
      * You can use the ISO to boot as a live CD from an external media
      * Or install the customized Ubuntu 16.04 you just created. ☺

# Future Development

This section will cover the items that are still pending to be done or added to the CSIRC Windows Template VM.

## VM Configuration

We will cover here the list of changes in the VM configuration that still need to be applied.

1. Sync clock
2. Change Keyboard layout
3. Automate the source.list update with RepoForIP after install.
   1. Currently reverts back to default values when using the install option from the live CD.

## Application installation

1. Email Forensic Tools

# List of Installed Applications

This section will contain a detailed list of all the non-standard Linux applications that were installed on the CSIRC Linux Template VM (ordered alphabetically):

* Artifacts (v20171107)
* Autoconf (v2.69)
* Automake (v1.15)
* Autotools-dev (v20150820)
* Baseline-Analysis (v6b2e8012)
* Bencode (v1.0)
* Bless (v0.6.0)
* Build-essential (v12.1)
* Curl (v7.47.0)
* Dc3dd (v7.2.641)
* Dislocker (v0.7.1)
* Dos2unix (v6.0.4)
* Ewf-tools (v20140608)
* Exif (v0.6.21)
* Forensic Registry EDitor (v0.1.1)
* Guymarger-beta (v0.8.4)
* Hexedit (v1.2.13)
* Jq (v1.5)
* Libevtx-utils (v20160107)
* Libfuse-dev (v2.9.4)
* Libmbedtls-dev (v2.2.1)
* Libncurses5-dev (v6.0)
* Mono-complete (v4.2.1)
* Nfdump (v1.6.12)
* P7zip-full (v9.20)
* P7zip-rar (v9.20)
* Parallel (v20141022)
* Plaso (v1.4.0)
* Python pyopenssl (v0.15.1)
* Python python-evtx (v0.6.1)
* Python python-registry (v1.0.4)
* Python rekall (v1.6.0)
  + Rekall-profiles (v20170824)
* Python scapy (v2.3.3)
* Python virtualEnv (v15.1.0)
* Python-pip (v8.1.1)
* Radare2 (v0.9.6)
* Regripper (v2.8)
* Sleuthkit (v4.2.0)
* Sqlitebrowser (v3.7.0)
* Tcpdump (v4.9.0)
* Tlsh-tools (v3.4.4)
* Tshark (v2.2.6)
* Unzip (v6.0)
* Volatility (v2.5)
* Wireshark (v2.2.6)
* Xmount (v0.7.5)
* Yara (v3.4.0)

# Quick User Guide

This section will cover the basis on how to use the Linux VM once it has been installed/Live Booted.

## Regripper

CSIRC Team is using the regripper tool to perform offline registry analysis.

To use regripper in the Linux VM:

1. /opt/rip.pl -r Hive -f plugin > output.txt

## Volatility

Volatility is one of the tools CSIRC Team is using for memory analysis.

To use it in the Linux VM:

1. Volatility –f /path/to/memdump.dmp plugin

## Rekall

Rekall is another memory analysis tool used by the CSIRC Team.

However, to avoid any issues with other python tools, Rekall was installed in a separate Virtual Environment.

To use recall, the following must be done:

1. Source /opt/VErekall/bin/activate
   1. You should now see a prompt preseeded with (VERekall)
2. Recall –f /path/to/memdump.dmp
   1. You should now see a prompt starting with the memdump.dmp file name
3. In this prompt, type any plugin you want to execute, e.g.: pslist
4. Once you are done, you can exit the virtual environment by typing: deactivate

## Scapy

Scapy is a packet manipulator tool.

In the same maner as Rekall, when installed, a python virtual environment was used.

To use scapy, follow the instruction below:

1. Source /opt/VEscapy/bin/activate
   1. You should now see a prompt preseeded with (VEscapy)
2. scapy
3. To exit the scapy shell, type: exit()

Once you are done, you can exit the virtual environment by typing: deactivate