Open Computer Forensics Architecture (OCFA)

User documentation

“Installing OCFA on Ubuntu”

Practical installation procedures,
Installing The Open Computer Forensics Architecture
on Ubuntu

October 2009

KLPD, Driebergen

Author: J. van der Wal

User documentation

“Installing OCFA on Ubuntu”

User documentation

“Installing OCFA on Ubuntu”

Table of contents

1 Introduction

This document intends to guide the ocfa-maintainer to install the Open Computer Forensics Architecture (OCFA) on an Ubuntu machine. Please use the newest version of ocfa (at this time version 2.2.0).

1.1 Remarks

All ubuntu-packages were installed using the synaptic package manager. All additional packages the package manager needs to install a given package are not mentioned in this manual. The user just has to follow up this hint.

If this document contains errors or if you think something is missing, please post this on the ocfa mailing list. We might be able to help and this will also help others which might encounter the same problems.

1.2 Tested versions

The following Ubuntu versions are tested:

- Ubuntu 8.10
- Ubuntu 9.04

2 Installing OS

The installation of the OS (Ubuntu) needs some attention. OCFA will use the following direcotiry structure:

Directory	file	Description	
/var/ocfa		Storage of all case specific data	
/var/ocfa/queues		Directory containing al persistent queues for all cases.	
/var/ocfa/windows		A directory used to samba-share files for processing by the MS-Windows modules.	
/var/ocfa/<casename>		The case specific directory. This maybe a symbolic link to external storage like a SAN.	
/var/ocfa/<casename>hashsets		The case specific hashsets	
/var/ocfa/<casename>index		The case specific full text index	
/var/ocfa/<casename>log		The case specific log (from log4j)	
/var/ocfa/<casename>repository		The case specific repository, containing all evidences from the case	
/var/ocfa/<casename>/thumbnails		The case specific thumbnails from all images	
/var/ocfa/<casename>/tmp_rar		Temporary working dir	
/var/ocfa/<casename>/work		Temporary working dir	
/usr/local/digiwash		The installation directory, where all ocfa software will be installed.	
/usr/local/digiwash/etc	casename.conf	Case specific configuration file and the rulelist.	
/usr/local/digiwash/static/hashsets		The overall system wide hashsets, to filter NIST content.	
/var/log/	ocfa.log	This file contains all logging from running a case	

2.1 Directories containing the data

Mounting the data filesystem with option 'noatime' will improve performance. For example all data in “/var/ocfa/<case>/repository” will be access frequently by ocfa

2.2 DONT'S

Don't use sourcedata stored on an external USB-drives.

Don't use samba or nfs network shares as working directory, repository or storage of the persistent queues.

Special attention is needed for the persistent queues. Store those on the local harddrive with a native linux filesystem like ext, xfs. We had some problems storing them on a SAN storage device.

3 Installing the postgresql database

3.1 Packages

libpq5

libpg-perl

postgresql

3.2 Configuration

The user has to change two different configuration files from the postgresql installation:

- /etc/postgresql/8.3/main/pg_hba.conf
- /etc/postgresql/8.3/main/postgresql.conf

Use the following commands to start the database server and also become the postgresql user, to have the rights to change these files.

```
sudo /etc/init.d/postgresql-8.3 start
sudo su - postgres
```

Edit the configuration file: /etc/postgresql/8.3/main/pg_hba.conf
Change the access rights of the localhost network to "trust". See codeblock underneath:

```
# IPv4 local connections:
host    all         all         127.0.0.1/32          trust
# IPv6 local connections:
host    all         all         ::1/128               trust
```

Edit the second configuration file: /etc/postgresql/8.3/main/postgresql.conf
Change "listen_address" to a wildcard character to listen to anyone.
See codeblock underneath:

```
Listen_addresses aanpassen
#------------------------------------------------------------------------
```

```
# CONNECTIONS AND AUTHENTICATION
#---------------------------------------------------------------------------
# - Connection Settings -

listen_addresses = '*'          # what IP address(es) to listen on;
```

To let these changes take effect, restart the postgresql server with following command:

```
sudo /etc/init.d/postgresql-8.3 restart
```

4 Installing development environment

Ocfa uses the default g++ and automake development environment.

4.1 Packages

autoconf
automake
autotools-dev
g++

libace-dev
libboost-dev
libssl-dev
libtool
libpq-dev
libxerces-c2-dev
libxerces-c28
autogen
cpp-doc
gcc-doc

Optional for debug use:
gdb-doc
valgrind

4.2 Extra packages needed for the ocfa Architecture (OcfaArch)

apache2
libcgicc5
libcgicc5-dev
libclucene-dev

4.3 Packages needed for the modules

uuid-dev
libdb-dev
libmagic-dev
samba
antiword
exiftags
p7zip-full
libspreadsheet-parseexcel-perl
libmail-mboxparser-perl
libmail-box-perl
libxml-dom-xpath-perl

python-devel
libcv-dev
libhighgui-dev
xpdf-utils

4.4 Packages needed for the NDA modules (optional)

Only if you have access to the NDA modules, additional packages are needed

netpbm
tesseract-ocr

4.5 Packages needed for OcfaRepFs

Remark: If you don't intend to use a a file-browser view on your multimedia content like 'video' and 'images', you can skip this paragraph.

OcfaRepFs is a perl based fuse module, so some extra perl modules are needed:

Perl modules, installed from cpan:

- Proc-DaemonLite
- Fuse

```
cpan>install Proc::DaemonLite
cpan>install Fuse
```

Other:

Make sure the fuse kernel base modules are installed.

- fuse-utils
- libfuse-dev

5 Installation from source

There are remaining packages not included in the ubuntu repository, or with an older version. These packages the user has to install from source.

5.1 Rar

```
tar zxf rarlinux-3.x.x.tar.gz
cd rar
sudo make install
```

5.2 Photorec

Used Version: 6.11WIP

Extract photorec tar file (download from http://www.cgsecurity.org/wiki/TestDisk_Download)

```
tar -xjf testdisk-6.11-WIP.tar.bz2
```

Change to photorec directory and execute following commands:

```
./configure --without-ncurses
```

```
make
sudo make install
```

5.2.1 Extra patch

The user has to change the executable name, otherwise the ocfa modules configuration detection mechanism will fail. This is done by creating a softlink.

```
cd /usr/local/sbin
sudo ln -s photorec photorec_cli
```

5.3 Libewf

Download libewf from www.uitwisselplatform.nl and compile it:

```
tar xzf libewf-20080501.tar.gz
cd libewf-20080501
./configure
make
sudo make install
```

5.4 TSK, The sleuthkit

Used version: sleuthkit 3.0.1 (http://www.sleuthkit.org)

Extract the tar file and change to sleuthkit3 directory.

Do:

```
./configure
make
sudo make install
```

5.4.1 Manual patch

The user has to create a softlink to an executable to stay compatible with older versions.

```
cd /usr/local/bin
ln -s blkls dls
```

5.5 Perl modules from CPAN

You have to download two perl-modules from CPAN:

1. Mail-Box-2.088.tar.gz
2. Mail-Transport-Dbx-0.07.tar.gz

General procedure [xxxx is either Box-2.088 or Transport-Dbx-0.07]:

```
tar xzf Mail-xxxx.tar.gz
cd  Mail-xxxx
perl Makefile.pl
make
sudo make install
```

An alternative to the above method is to use the `cpan` command. Run this command as root and use “`install` *`packagename`*” to install the proper package. Packagenames are of the form Mail::Transport::Dbx (note the double colons).

5.6 Vinetto

The Vinetto package is used to build a module for dissecting Thumbs.db files.

```
tar zxf vinetto-beta-0.07.tar.gz
cd  vinetto-beta-0.07
sudo python setup.py install
```

6 Building OCFA

The previous chapters described which preliminary packages need to be installed before building OCFA.

Now you should be ready to build OCFA. Untar and unzip the ocfa source package (“tar xjf ocfa-2.2.0pl0gpl.tar.bz2”).

Now you will have three subdirectories (components) called

1. OcfaLib

2. OcfaArch
3. OcfaModules

Build the three components in the order listed above. So, first change directory to OcfaLib and issue the commands;

```
./configure
make
sudo make install
cd ..
```

Do the same for the other OCFA components. If configure gives you any errors, they are most likely about packages which are not installed. If this happens make sure you have all packages installed. Often you can make an educated guess using the synaptic search function to install the proper package (hint; always choose the -dev package when in doubt).

The configure script of OcfaModules might issue a warning about the java version. You may ignore this warning.

7 Configuration

7.1 Samba

Edit /etc/samba/smb.conf:

Add the following section to the end of the smb.conf file:

```
[ocfa]
        comment = Samba to ocfa
        path = /var/ocfa/windows
        valid users = @ocfa ocfa
        writable = yes
        inherit acls = yes
        create mask = 0775
```

Execute following commands on commandline:

```
smbpasswd -a ocfa
sudo /etc/init.d/samba restart ('samba' is 'smb' on Suse)
```

7.2 Java environment

For the NDA part of ocfa and the OcfaJavaLib, it is necessary to have java stuff installed. The following java components are needed:

- jdk-1.5.0
- ant-1.7
- tomcat-5.5.20

unpack those components in a directory. I have chosen for ***/opt/java*.**

Result of ls -l /opt/java

```
2009-09-17 07:46 ant -> apache-ant-1.7.0
2006-12-13 13:15 apache-ant-1.7.0
2009-09-17 07:50 apache-tomcat-5.5.20
2009-09-16 16:35 jdk -> jdk1.5.0_21
2009-08-24 21:55 jdk1.5.0_21
2009-09-16 16:35 jre -> jdk/jre
2009-09-16 16:27 test
2009-09-17 07:51 tomcat -> apache-tomcat-5.5.20
```

If you install those components, they have to be defined.

```
cd /etc/profile.d
vi java.sh
```

Example of java.sh:

```
#!/bin/bash
export JAVA_HOME=<path-to-your-jdk-directory.
export ANT_HOME=<path-to-your-ant-direcotry>
export CATALINA_HOME=<path-to-your-tomcat-direcoty>
```

I also made some symlinks to the binaries:

```
cd /usr/local/bin
ln -s /opt/java/jdk/bin/javac
ln -s /opt/java/ant/bin/ant
ln -s /opt/java/jre/bin/java
```

7.3 Digest hashsets

Copy your hashsets (adinfodb digestdb proddb) to /usr/local/digiwash/static/hashsets
The hashsets already present in this directory are empty/dummy.

8 The next document to read

The next step is to run a test. Please read:

“Gebruikersdocumentatie - Gebruik in de praktijk”
[Translation from Dutch: Userdocumentation – Practical use]
Don't worry, the content is English...