FOR518 – Laptop Setup (Pre-class)

Objectives

- Install required software for FOR518 – Mac Forensic Analysis

Class Preparation

This process should take approximately 1 hour, including download time. Xcode is **very** large will take a long time to download; depending on your connection this process could take longer.

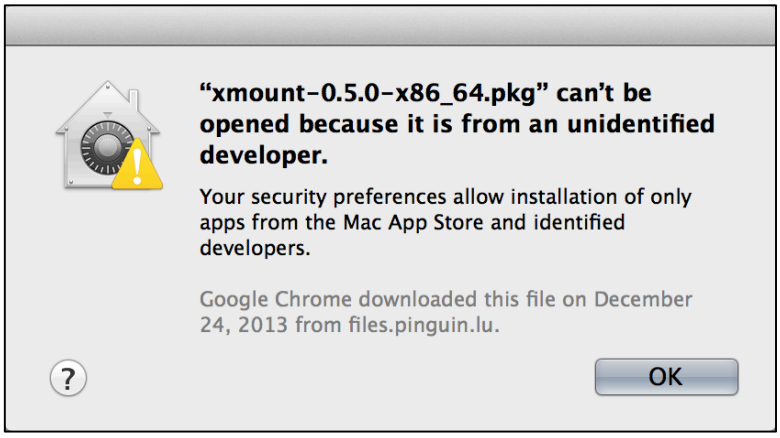
*You may use your host system **or** a virtual machine; however this setup has not been fully tested in a VM. If you choose to go this route, please be aware that not all tools may work as intended.*

******NOTE**: It is **very** important that steps 1-5 are following in order to ensure proper software installation.****

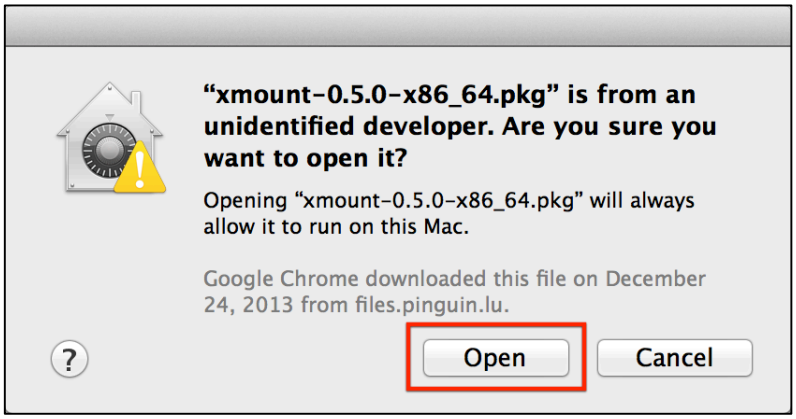
You may download the files at their respective websites listed **or** you may download an archive of these files here: http://for518.com/tools (excludes tools that are too large or needs to be done online). **If you are in class**, the Tools directory on your thumb drives will provide these tools. Please use the application 'The Unarchiver' to extract the 7zip files (included on thumb drive).

Gatekeeper Settings:

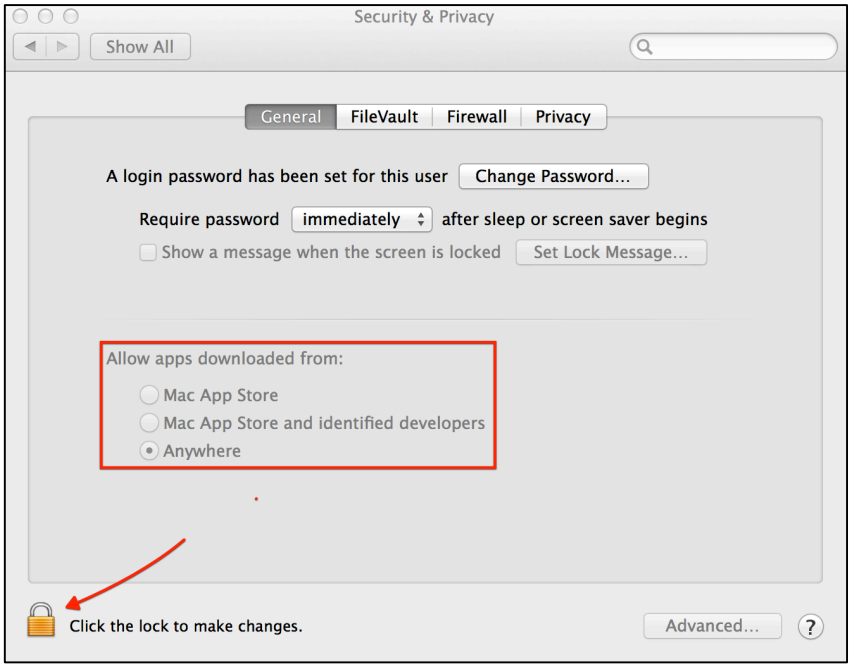
- Some installer files are from "Unidentified Developers".



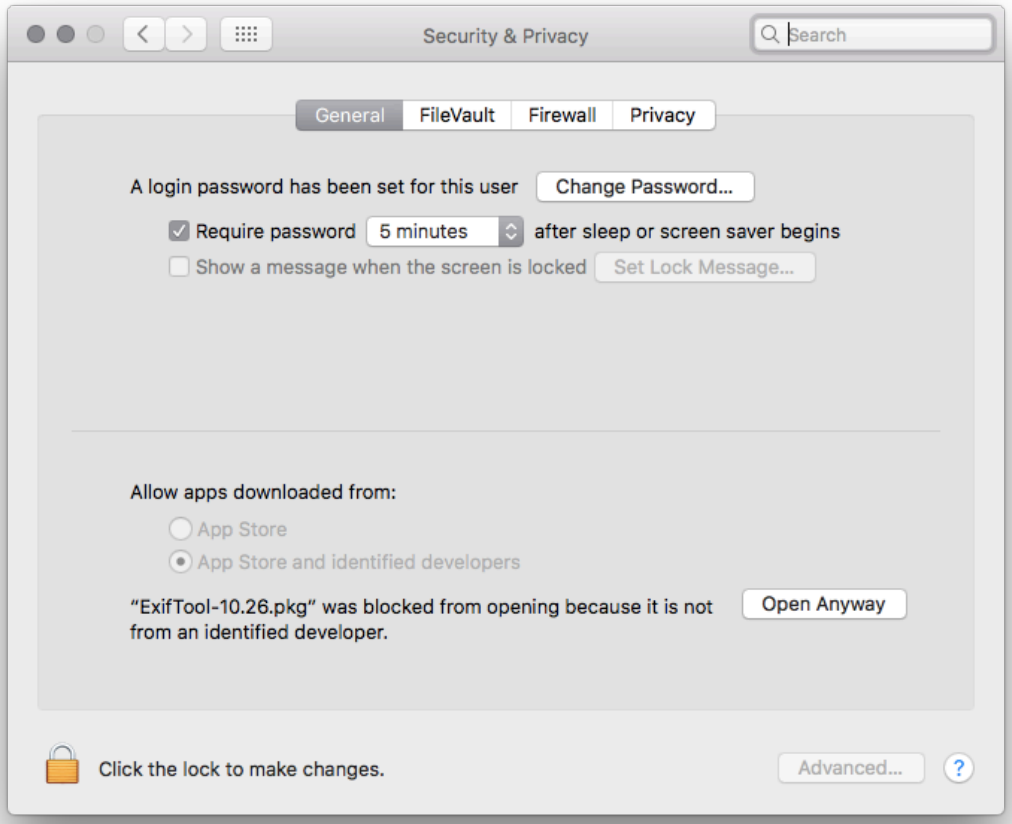
- Users may allow these files to be installed by Control+clicking the installer file and choosing "Open". A window will pop-up, select "Open".



- To permanently change this setting, navigate to the Security & Privacy Preferences Panel (`Apple Menu | System Preferences | Security & Privacy | General Tab`). Select "`Anywhere`" under "`Allow apps downloaded from:`".
    - This may require Administrative privileges.



Users using **macOS Sierra** or who **do not want to use the "Anywhere"** option will need to use the "Open Anyway", shown below, each time they get the "Unidentified Developers" error.



1. **Xcode & Xcode Command Line Tools**
   - If you have not already done so, register for an Apple Developer Account here. It requires an Apple ID, if you do not have one you may also register for one at `https://developer.apple.com/register/`
   - Determine your OS version by going to Apple Menu | About This Mac, you will need to download Xcode and Command Line Tools specific for this OS version. This chart may help determine this: `https://en.wikipedia.org/wiki/Xcode#Version_comparison_table`

   1. Please download the latest **Xcode** available for your operating system from the App Store or `https://developer.apple.com/downloads/`
      i. You may have to go click "More Downloads" to access older versions.

   2. Please **also** download the latest **Command Line Tools** (for your version of the OS) from `https://developer.apple.com/downloads/`
      i. You may have to go click "More Downloads" to access older versions.

   3. Install **Xcode** (**Note**: This will take a while, grab some coffee.)
      i. If installing via App Store, installation will be done for you.
      ii. If installing via DMG file, open the DMG file and drag the application to the `/Applications` directory.

4. Install **Command Line Tools**
    i. Open the DMG file, double-click the package installer and follow the default prompts.

2. **OS X FUSE**
    1. Download OSXFUSE from `http://osxfuse.github.io/`
    2. Open the DMG file, double-click the package installer and follow the default prompts.

3. **libewf**
    1. Download the newest `libewf-20140608.tar.gz` from `https://github.com/libyal/legacy/tree/master/libewf/.`
    2. Locate and open the `Terminal.app` from `/Applications/Utilities/`
    3. Use the `cd` command to open the default Downloads directory.
    4. Use the `tar` command to unpack the `libewf-20140608.tar.gz` file.
    5. Once unpacked, cd into the `libewf-20140608` directory.
    6. Configure and install `libewf` using the commands (shown below):
        i. `./configure`
        ii. `make`
        iii. `sudo make install`
    7. **A summary screen will be shown to you once complete. Ensure the "FUSE support" states something other than "no" – different versions will display "libosxfuse" or "libfuse".**

```
Building:
   libcstring support:                     local
   libcerror support:                      local
   libcthreads support:                    local
   libcdata support:                       local
   libcdatetime support:                   local
   libclocale support:                     local
   libcnotify support:                     local
   libcsplit support:                      local
   libuna support:                         local
   libcfile support:                       local
   libcpath support:                       local
   libbfio support:                        local
   libfcache support:                      local
   libfdata support:                       local
   libfvalue support:                      local
   libmfdata support:                      local
   ADLER32 checksum support:               zlib
   DEFLATE compression support:            zlib
   BZIP2 compression support:              bzip2
   libhmac support:                        local
   MD5 support:                            libcrypto
   SHA1 support:                           libcrypto
   SHA256 support:                         libcrypto
   libcaes support:                        local
   AES support:                            libcrypto
   libodraw support:                       local
   libsmdev support:                       local
   libsmraw support:                       local
   libcsystem support:                     local
   GUID/UUID support:                      native
   FUSE support:                           libosxfuse

Features:
   Multi-threading support:                pthread
   Wide character type support:            no
   ewftools are build as static executables: no
   Python (pyewf) support:                 no
   Verbose output:                         no
   Debug output:                           no
   Version 1 API compatibility:            no
```

i. Like in screenshot above, your output should say "libosxfuse" or "libfuse" rather than "no". If yours does not, try the following troubleshooting:
   1. Find your `fuse.h` file. In a terminal type `sudo find / -name fuse.h`
   2. Take note of where your fuse.h file is located. For example, the author's is installed in `/usr/local/include/osxfuse/fuse.h`
   3. Try the command `./configure --with-libfuse=/usr/local/`
   4. If all else fails, send the path to your `fuse.h` file and the `config.log` file created in the `libewf-YYYYMMDD` directory to your instructor.

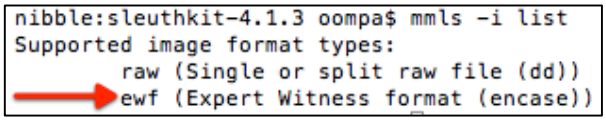
```
$ cd ~/Downloads

$ tar xvf libewf-20140608.tar.gz

$ cd libewf-20140608

$ ./configure

$ make

$ sudo make install
```

4. **xmount 64-bit Package**
   1. Download xmount-0.5.0-x86_64.pkg (or newer) from `http://www.pinguin.lu/`
      a. Click the `XMOUNT` link on the right side under "Projects".
      b. Download the package labeled, "`Mac OS X 64bit package`"
   2. Open the DMG file, double-click the package installer and follow the default prompts.
      a. **If you get the error "OS X Fuse Not Installed Error"** please run the following 'mkdir' command below then re-run the xmount package installer. *(Make sure you type out "osxfusefs.fs" in each case when using tab completion.)*

```
$ mkdir -p /Library/Filesystems/osxfusefs.fs/Support/osxfusefs.kext
```

5. **The Sleuth Kit**
   1. Download `sleuthkit-4.#.#.tar.gz` from `https://www.sleuthkit.org/sleuthkit/download.php`
   2. Locate and open the `Terminal.app` from `/Applications/Utilities/`
   3. Use the `cd` command to open the default `Downloads` directory.
   4. Use the `tar` command to unpack the `sleuthkit-4.#.#.tar.gz` file.
   5. Once unpacked, `cd` into the `sleuthkit-4.#.#.tar.gz` directory.
   6. Configure and install `sleuthkit` using the commands:
      a. `./configure --disable-java`
      b. `make`
      c. `sudo make install`

7. Ensure the install was successful and the `libewf` package was recognized by executing the `mmls -i list` command.



   a. If you do not see the string "`ewf (Expert Witness format (encase))`" in this list something went wrong. Please attempt to reinstall `libewf` and The Sleuth Kit.

```
$ cd ~/Downloads

$ tar -xvf sleuthkit-4.#.#.tar.gz

$ cd sleuthkit-4.2.0

$ ./configure

$ make

$ sudo make install

$ mmls -i list
```

6. **exiftool**
   1. Download `ExifTool-9.48.dmg` (or newer) from `http://www.sno.phy.queensu.ca/~phil/exiftool/`
   2. Open the DMG file, double-click the package installer and follow the default prompts.

7. **Synalyze It!**
   1. Download Synalyze It! Pro Trial from `http://www.synalysis.net/downloads/`. This is a 30-day trial; you may also purchase the non-pro version for from the App Store.
   2. If purchased from the App Store, it will install automatically.
   3. If you downloaded the trial, unzip the file and move the application to your `/Applications` directory.

8. **SQLite Browsers:**
   - You may choose your favorite, these are recommended:
   - **Firefox & SQLite Manager Add-on**

   1. Firefox
      a. Download the latest version of Firefox from `firefox.com`.
      b. Open the DMG file, drag the Firefox application to the `/Applications` directory.
   2. SQLite Manager Add-on
      a. From Firefox, download from `http://addons.mozilla.org/en-US/firefox/addon/sqlite-manager/`

b. Click the green "+ Add to Firefox" button, press the "Install Now" button, and restart Firefox.
    i. To ensure successful installation, go to Tools in the menu – you should see "SQLite Manager".

- **SQLite Database Browser**

3. Download the latest version of SQLite Database Browser from http://sqlitebrowser.org/.
4. Open the DMG file, drag the SQLite Database Browser application to the /Applications directory.

9. **Hex Editors**
    - You may choose your favorite, these are recommended:
        i. Hex Fiend
            1. Download from http://ridiculousfish.com/hexfiend/
            2. Unzip and move the application to the /Applications directory.
        ii. 0xED
            1. Download from http://www.suavetech.com/0xed/
            2. Open the BZip2 archive by double clicking, then move the application to the /Applications directory.

10. **The Unarchiver**
    1. Download The Unarchiver from the Mac App Store or from http://unarchiver.c3.cx/unarchiver, under the "Other Links" heading.
    2. Double-click to unzip.
    3. Drag the Unarchiver.app file to the /Applications directory.

11. **Homebrew**
    1. Download the Mac package manager Homebrew from https://brew.sh/.
    2. This webpage will contain a script that you need to copy and paste into your Terminal window.

12. **Volatility**
    1. Download and install Volatility using homebrew using Homebrew.
    2. Use the brew install command to do this.
        1. brew install volatility

```
$ brew install volatility
```

13. **John the Ripper**
    1. Download and install John the Ripper using Homebrew
    2. Use the brew install command to do this.
        1. brew install john-jumbo

```
$ brew install john-jumbo
```

14. **Create a FOR518 directory**
    - The exercises for this class will reference a FOR518 folder in the user's home directory to dump various files for use in other exercises. (`~/FOR518`)
    - Please create a directory named FOR518. You do not have to create it in your home directory, but be sure to remember where it is. The workbook used in class will reference this directory in your home directory.
    - The command below shows how to create this folder in your home directory. You may also use the GUI interface to do this.

```
$ mkdir ~/FOR518
```