**Prerequisites for Sovrin Experiments on a Workstation**

Daniel Hardman

Oct 7 2016

*(Our starting point for these instructions was a clean ubuntu 16.04 machine that has nothing special installed, except for the “build-essential” debian package. We have also tested on ubuntu 14 and on centos, though not quite as often. The windows steps are not recorded here, but should generally be derivable from their linux counterparts. We will update the doc with windows instructions soon.)*

**Python 3.5 and libsodium**

**Ubuntu:**

**Python3.5 installation**

|  |
| --- |
| sudo add-apt-repository ppa:fkrull/deadsnakes  sudo apt-get update |

**On Ubuntu 14** (python3.5 is pre-installed on most Ubuntu 16 systems; if not, do it there as well.):

|  |
| --- |
| sudo apt-get install python3.5 |

**Libsodium installation**

We need to install libsodium with the package manager. This typically requires a package repo that's not active by default. Inspect **/etc/apt/sources.list** file with your favorite editor (using sudo).

**On ubuntu 16**, you are looking for a line that says

deb http://us.archive.ubuntu.com/ubuntu xenial main universe

**On ubuntu 14**, look for or add:

deb http://ppa.launchpad.net/chris-lea/libsodium/ubuntu trusty main

deb-src http://ppa.launchpad.net/chris-lea/libsodium/ubuntu trusty main

|  |
| --- |
| sudo apt-get update |

**Note:** On ubuntu 14, if you get a GPG error about public key not available, run this:

|  |
| --- |
| sudo apt-key adv --keyserver keyserver.ubuntu.com --recv-keys B9316A7BC7917B12  sudo apt-get update |

Install libsodium; the version depends on your distro version.

**On Ubuntu 14**

|  |
| --- |
| sudo apt-get install libsodium13 |

**On Ubuntu 16**

|  |
| --- |
| sudo apt-get install libsodium18 |

If you still get the error *E: Unable to locate package libsodium13* then **add this to your /etc/apt/sources.list** file

deb http://ppa.launchpad.net/chris-lea/libsodium/ubuntu trusty main

deb-src http://ppa.launchpad.net/chris-lea/libsodium/ubuntu trusty main

**Now run**

|  |
| --- |
| sudo apt-get update  sudo apt-get install libsodium13 |

**CentOS/Redhat:**

|  |
| --- |
| sudo yum install python3.5  sudo yum install libsodium-devel |

**Mac:**

1. Go to [python.org](https://www.python.org/) and from the "Downloads" menu, download the Python 3.5.0 package (python-3.5.0-macosx10.6.pkg) or later.
2. Open the downloaded file to install it.
3. If you are a homebrew fan, you can install it using this brew command:

|  |
| --- |
| brew install python3 |

1. To install homebrew package manager, see: [brew.sh](http://brew.sh/)
2. Once you have homebrew installed, run

|  |
| --- |
| brew install libsodium |

**Windows:**

1. Go to <https://download.libsodium.org/libsodium/releases/> and download the latest libsodium package (libsodium-1.0.8-mingw.tar.gz is the latest version as of this writing)
2. When you extract the contents of the downloaded tar file, you will see 2 folders with the names libsodium-win32 and libsodium-win64.
3. As the name suggests, use the libsodium-win32 if you are using 32-bit machine or libsodium-win64 if you are using a 64-bit operating system.
4. Copy the libsodium-x.dll from libsodium-win32\bin or libsodium-win64\bin to C:\Windows\System or System32 and rename it to libsodium.dll.
5. Download the latest build (pywin32-220.win-amd64-py3.5.exe is the latest build as of this writing) from [here](https://sourceforge.net/projects/pywin32/files/pywin32/Build%20220/) and run the downloaded executable.

**Developer tools**

Install pip, pip3, and python3.5-dev; we’ll be using these tools to work with sovrin source and binary packages. **For Ubuntu**:

|  |
| --- |
| sudo apt-get install python-pip python3-pip python3.5-dev unzip make |

… or, **for Centos** and similar distros:

|  |
| --- |
| curl "https://bootstrap.pypa.io/get-pip.py" -o "get-pip.py"  sudo python get-pip.py  pip -V |

**Crypto libraries**

Now install some crypto dependencies. You will have to confirm actions several times. You can ignore errors about requirements.txt or updates to pip. (For mac, note that you’ll want to wget setup-charm-homebrew.sh instead.)

For Debian/Ubuntu

|  |
| --- |
| wget <https://github.com/evernym/indy-anoncreds/blob/master/setup-charm.sh>  chmod +x setup-charm.sh  ./setup-charm.sh  sudo ldconfig |
|  |

For OSX

|  |
| --- |
| wget <https://raw.githubusercontent.com/evernym/anoncreds/master/setup-charm-homebrew.sh>  chmod +x setup-charm-homebrew.sh  ./setup-charm-homebrew.sh |

**Python virtual environment**

You will probably want to use pip in a python virtual environment, unless you are a hard-core python developer who understands the pros and cons. On Ubuntu 14, this is particularly important, because sovrin uses python 3.5 but will have other versions of python on the system.

To install Python virtual environment tools:

|  |
| --- |
| sudo -H pip install virtualenvwrapper |

Now, append some lines to the .bashrc file to make virtualenvwrapper easy to use. In Ubuntu:

|  |
| --- |
| echo '' >> ~/.bashrc  echo '# Python virtual environment wrapper' >> ~/.bashrc  echo 'export VIRTUALENVWRAPPER\_PYTHON=/usr/bin/python3' >> ~/.bashrc  echo 'export WORKON\_HOME=$HOME/.virtualenvs' >> ~/.bashrc  echo 'source /usr/local/bin/virtualenvwrapper.sh' >> ~/.bashrc |

**In Centos** the commands are almost identical, but the path to virtualenwrapper.sh changes at the end:

|  |
| --- |
| echo '' >> ~/.bashrc  echo '# Python virtual environment wrapper' >> ~/.bashrc  echo 'export VIRTUALENVWRAPPER\_PYTHON=/usr/bin/python3' >> ~/.bashrc  echo 'export WORKON\_HOME=$HOME/.virtualenvs' >> ~/.bashrc  echo 'source /usr/bin/virtualenvwrapper.sh' >> ~/.bashrc |

Then run .bashrc to make changes take effect in the current shell:

|  |
| --- |
| source ~/.bashrc |

**Create sovrin virtual environment**

Now, create a virtual environment for yourself that uses python 3.5:

|  |
| --- |
| mkvirtualenv -p python3.5 sovrin  workon sovrin |

At this point you can switch back to the [Getting Started tutorial](https://github.com/sovrin-foundation/sovrin-client/blob/stable/getting-started.md#install-sovrin) and follow steps under “Install Sovrin”