This repository Search | Pull requests | Issues | Gist

angr / **angr**

Watch 30 | Star 226 | Fork 27

The next-generation binary analysis platform from UC Santa Barbara's Seclab!

2,227 commits | **1** branch | **0** releases | **14** contributors

Branch: **master** | angr / +

Set an additional error property in path_hierarchy on finding an unre… …

rhelmot authored 2 days ago | latest commit 36c6c1aa61

angr	Set an additional error property in path_hierarchy on finding an unre…	2 days ago
tests	Merge branch 'cfg/get_paths' into 'master'	6 days ago
.gitignore	Shut the Heck Up about egg-info!!!	3 months ago
.gitlab-ci.yml	Test on master	6 days ago
LICENSE	license	23 days ago
README.md	readme update	19 days ago
requirements.txt	pull requirements straight from pip	23 days ago
setup.py	tick setup	7 days ago

Code

Issues 9

Pull requests 1

Wiki

Pulse

Graphs

HTTPS clone URL

`https://github.com`

You can clone with HTTPS, SSH, or Subversion.

Clone in Desktop

Download ZIP

README.md

angr

angr is a platform-agnostic binary analysis framework developed by the Computer Security Lab at UC Santa Barbara and their associated CTF team, Shellphish.

For information about how to use angr, consult the angr-doc repository.

Installation

Installing angr is quite simple!

Dependencies

angr is built for Python 2. Python 3 support is feasable somewhere out in the future, but we are a little hesitant to make that commitment right now (pull requests welcome!).

All of the python dependencies should be handled by pip and/or the setup.py scripts. You will, however, need to build some C to get from here to the end, so you'll need whatever base compiler package your OS wants to use, as well as the python development package (for the right headers). At some point in the dependency install process, you'll install the python library cffi, but it won't run unless you install libffi.

You will also need to use the python virtual environments in the build (and usage) process.

On Ubuntu, you will want:

```
sudo apt-get install python-dev libffi-dev build-essential virtualenvwrapper
```

Production install

angr is meant (and tested) to be installed in a virtualenv. `mkvirtualenv angr` will do the trick. To install, do:

```
mkvirtualenv angr
pip install angr
```

To switch to the virtualenv later (and use angr), do `workon angr`.

Development install

`pip` provides a nice "development installation" mode, allowing a developer to work on a git repo without having to constantly reinstall the package. To utilize this, perform the following:

```
mkvirtualenv angr
mkdir ~/angr; cd ~/angr

git clone https://github.com/angr/angr
git clone https://github.com/angr/angr-management
git clone https://github.com/angr/simuvex
git clone https://github.com/angr/claripy
git clone https://github.com/angr/cle
git clone https://github.com/angr/pyvex
git clone https://github.com/angr/vex
git clone https://github.com/angr/archinfo
git clone https://github.com/zardus/ana
git clone https://github.com/zardus/cooldict

pip install -e ./cooldict -e ./ana -e ./archinfo -e ./pyvex -e ./cle -e ./claripy -e ./simuvex
```

This will create a `~/angr` directory, into which all of the angr sub-components will be checked out. You can then branch/edit/recompile the various modules in-place, and it will automatically reflect in your virtual environment.

Troubleshooting

libgomp.so.1: version `GOMP_4.0' not found

This error represents an incompatibility between the pre-compiled version of `angr-z3` and the installed version of `libgomp`. A Z3 recompile is required. You can do this by executing:

```
pip install -I --no-use-wheel angr-z3
```

Can't import mulpyplexer

There are sometimes issues with installing mulpyplexer. Doing `pip install --upgrade 'git+https://github.com/zardus/mulpyplexer'` should fix this.

Windows and Capstone

On windows installing capstone can be a bit of a hassle. You might need to manually specify a wheel to install, but sometimes it installs under a name different from "capstone", so if that happens you want to just remove capstone from the requirements.txt files in angr and archinfo.

Claripy and z3

Z3 is a bit weird to compile. Sometimes it just completely fails to build for no reason, saying that it can't create some object file because some file or directory doesn't exist. Just retry the build.

Claripy and z3 on Windows

Z3 might compile on windows if you have a l33t enough build environment. If this isn't the case for you, you should download a wheel from somewhere on the internet. I found one once, but can't seem to find it again while writing this.

If you build z3 from source, make sure you're using the unstable branch of z3, which includes floating point support. In addition, make sure to have `Z3PATH=path/to/libz3.dll` in your environment.