**AIOPro Visualization Manual**

1. Ubuntu (14.04 recommended)
   1. 사전 설치
      1. openssh-server

**$sudo apt-get install openssh-server**

* + 1. GTKWave

**$sudo apt-get install gtkwave**

* 1. 실행방법

aiopro\_visualizer\_ubuntu.7z 압축을 풀면 aiopro\_visualizer\_ubuntu 폴더가 생성된다. 해당 폴더에서 aiopro\_visualizer.py 파이썬 스크립트를 로그파일들이 들어있는 폴더 경로를 인자로 함께 실행시킨다.

**$cd aiopro\_visualizer\_ubuntu**

**$python visualizer.py [log\_dir]**

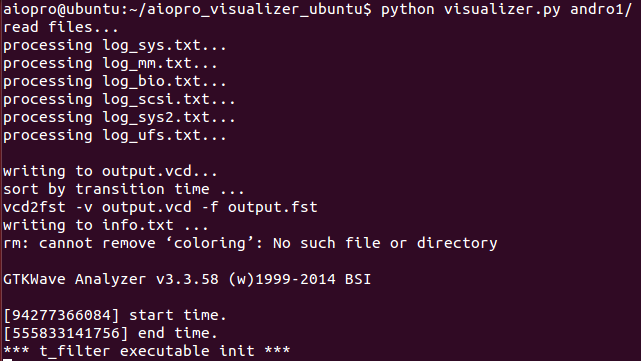


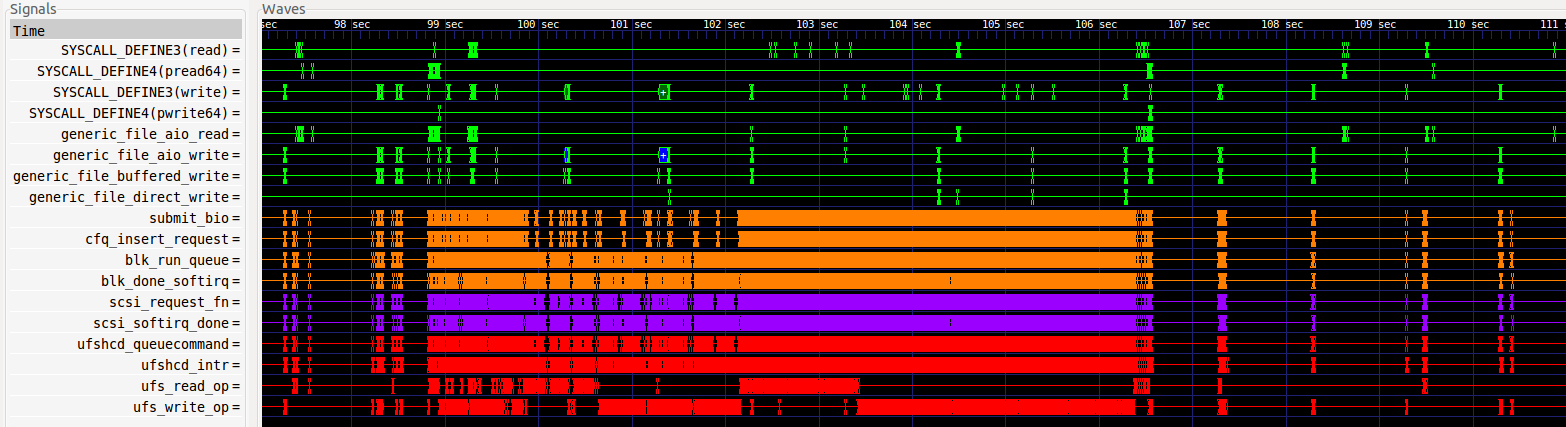
그림 . aiopro\_visualizer 실행 화면

그림 . aiopro\_visualizer 결과화면

1. Windows

디바이스의 로그파일들을 호스트에 저장한 후, 서버에 로그파일을 보내 visualization을 자동적으로 하도록 구현하였다.

* 1. 사전 설치
     1. GTKWave가 설치된 Ubuntu 서버 (또는 가상머신)
     2. Python 2.7 설치

다운로드 URL: <https://www.python.org/downloads/release/python-2711/>

Windows x86 MSI installer (python-2.7.11) 또는 Windows x86-64 MSI installer (python-2.7.11.amd64) 다운로드하여 설치

설치방법:

- 설치 파일 (python-2.7.11 또는 python-2.7.11.amd64) 더블클릭

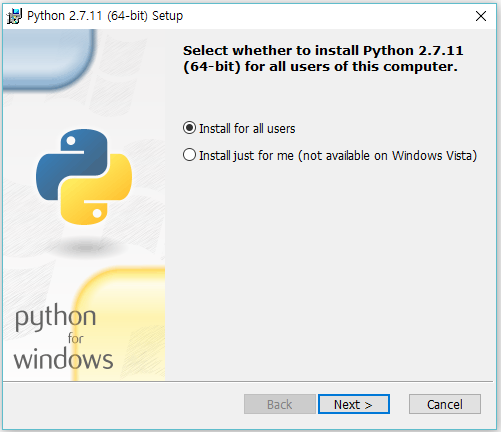
-“Next”클릭

그림 . Python 2.7 설치 화면

- 시스템 환경변수 “Path”에 Python2.7 설치 디렉토리 추가

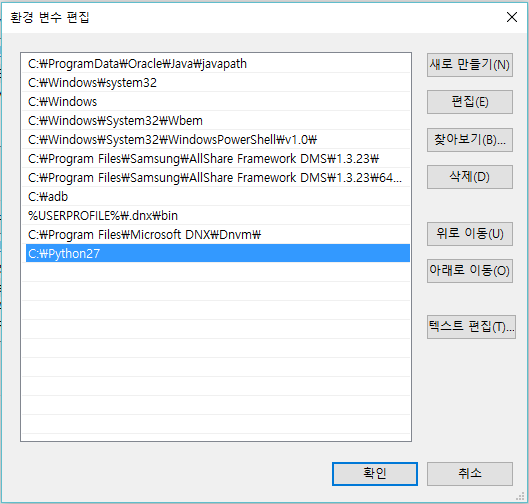
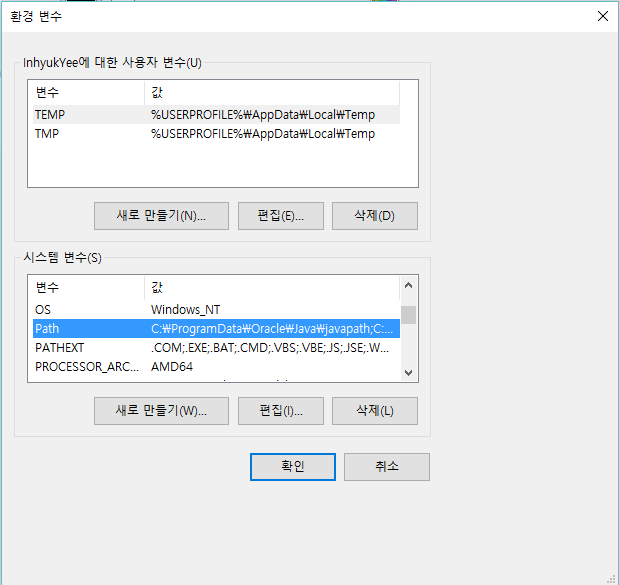


그림 . 시스템환경 변수 추가 화면

* + 1. Python 모듈 설치
       1. paramiko

다운로드 URL: <https://pypi.python.org/pypi/paramiko/1.16.0>

paramiko-1.16.0.tar.gz 다운로드

설치방법:

- paramiko-1.16.0.tar.gz 압축풀기 (paramiko-1.16.0 폴더 생성)

- 커맨드 창을 열어 paramiko-1.16.0 폴더로 진입

- 모듈 설치 (명령어: python setup.py install)

**$cd paramiko-1.16.0**

**$python setup.py install**

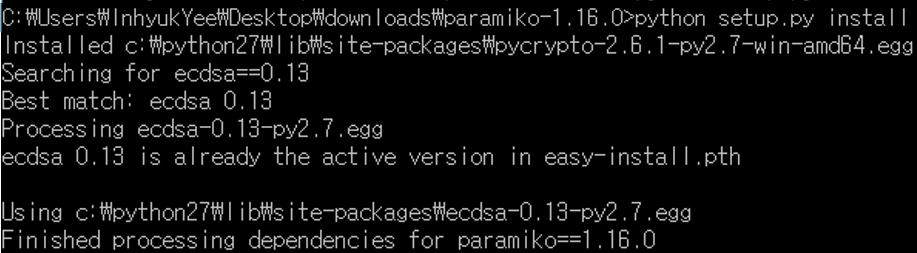


그림 . paramiko 모듈 설치화면

**\*아래와 같은 에러 발생시,** [**http://aka.ms/vcpython27**](http://aka.ms/vcpython27로%20접속하여%20VCForPython27.msi)**로 접속하여 다운로드 및 설치**



* + - 1. scp

다운로드 URL: <https://pypi.python.org/pypi/scp#downloads>

scp-0.10.2.tar.gz 다운로드

설치방법:

- scp-0.10.2.tar 압축풀기 (scp-0.10.2폴더 생성)

- 커맨드 창을 열어 scp-0.10.2 폴더로 진입

- 모듈 설치 (명령어: python setup.py install)

**$cd scp-0.10.2**

**$python setup.py install**



그림 . scp 모듈 설치화면

* + 1. putty 다운로드 및 환경변수 추가

다운로드URL: <http://www.chiark.greenend.org.uk/~sgtatham/putty/download.html>

putty.exe 다운로드

시스템 환경변수 “Path”에 putty.exe 파일이 들어있는 디렉토리 추가

* + 1. Xming 다운로드 및 설치

다운로드 URL: <https://sourceforge.net/projects/xming/>

다운로드 및 설치

* 1. 실행방법

Xming 실행 (Ubuntu 서버의 X window를 받는 역할)

aiopro\_visualizer\_win.7z 압축을 풀면 aiopro\_visualizer\_win 폴더가 생성된다.

1. server\_info.txt 파일에 서버 정보를 입력해준다.

remote\_aiopro\_visualizer\_location은 서버 내 aiopro\_visualizer\_ubuntu 폴더의 절대경로이다.

1. run.py 파일을 더블클릭하여 실행한다..

putty가 실행되고 마지막으로 Xming이 서버의 GTKWave GUI를 받아온다.

1. 이미 한번 visualization 처리된 데이터를 다시 visualization 하기 위해선 remote\_aiopro\_visualizer\_location에서 “$gtkwave output.gtkw” 명령어를 실행시킨다. (PuTTY 계정의 Configuration에서 Enable X11 forwarding 체크 및 X display location을 localhost:0로 설정, 그림 7 참조)

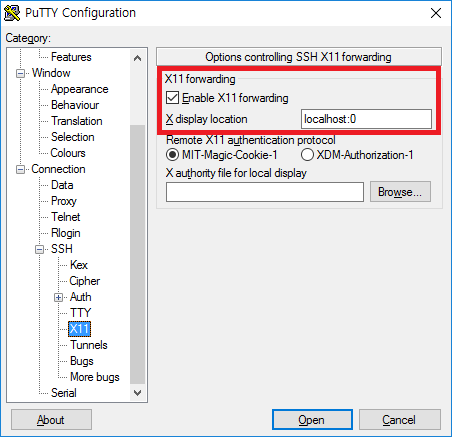


그림 . PuTTY X11 forwarding

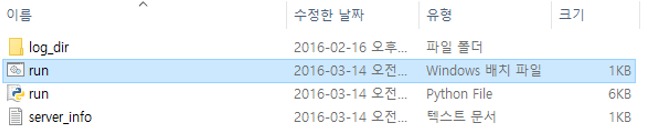


그림 . aiopro\_visualizer\_win 폴더

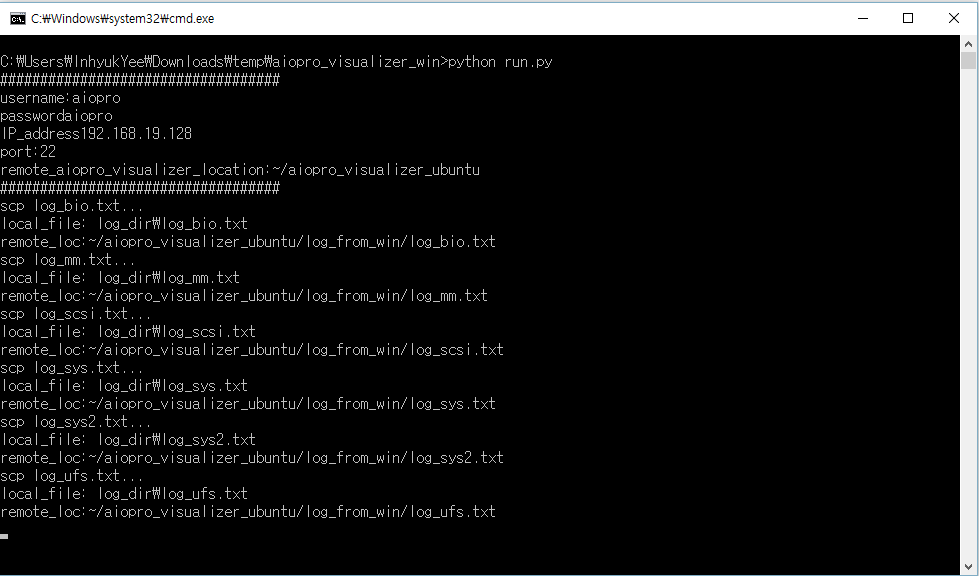


그림 . run.py 실행 시, 커맨드창 로그

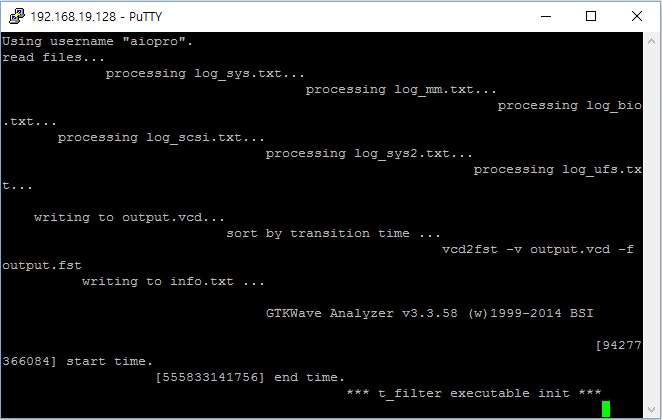


그림 . run.py 실행 시, putty 로그

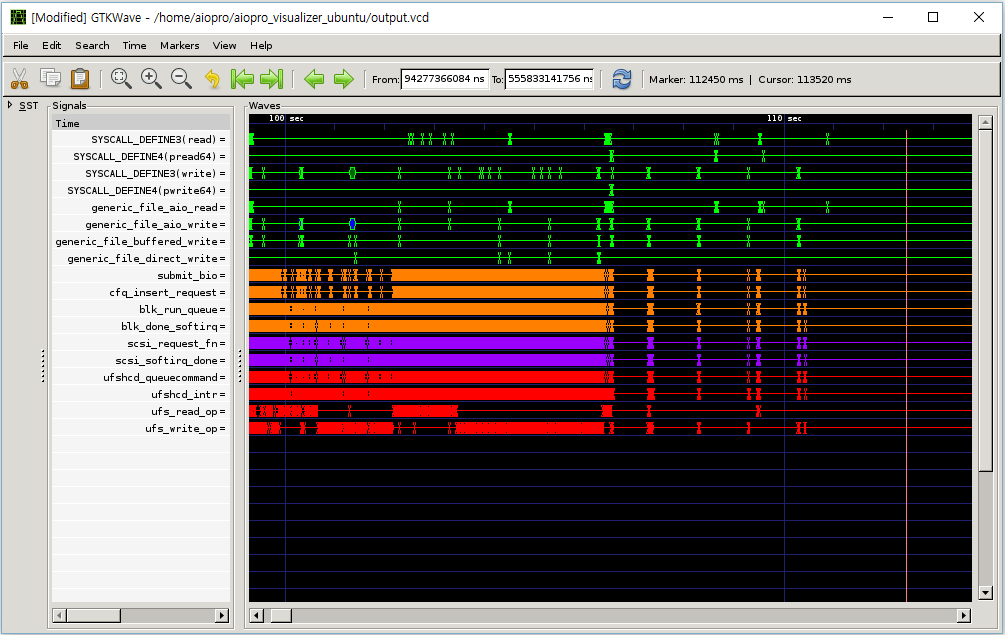


그림 . run.py 실행 시, GTKWave 결과화면

1. AIOPro Visualizer 구조

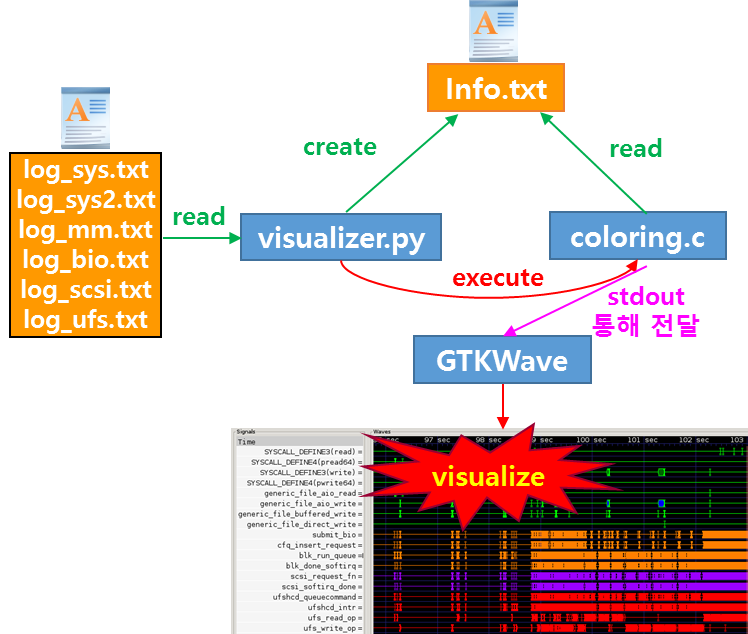


그림 1. Visualizer 전체 동작과정

그림 11은 aiopro visualizer의 전체적인 동작과정을 개괄적으로 나타내었다. 동작방식은 다음과 같다. 로그 파일들은 매 라인마다 각 함수의 시작시간, 실행시간, 레이어 이름, type, lba, size 등의정보를 가지고있다.

* 1. visualizer.py는 6개의 로그 파일들(log\_\*.txt)을 모두 라인별로 읽어 시작시간을 오름차순으로 정렬해 info.txt에 저장한다.
  2. coloring.c는 info.txt 파일을 읽어 포맷에 맞게 stdout으로 GTKWave에 전달해준다.
  3. GTKWave는 coloring.c가 전달해준 정보를 visualize 해준다.