LLDB

LLDB Initialization

Example of a script to run LLDB on remote android server, receives an argument - process name (can also be package name)

```
@echo ========Pushing lldb-server to target========
@adb push <PATH_TO: lldb-server>
@adb shell "ls -al /data/local/tmp/lldb-server"
@adb shell su -c 'chmod 755 /data/local/tmp/lldb-server'
@adb shell "ls -al /data/local/tmp/lldb-server"
@
@echo ========Check if already running and kill========
@adb shell su -c "ps -A | grep -i lldb-server"
@echo ========Attempting to kill lldb-server========
@adb shell su -c pkill lldb-server
@adb shell su -c "ps -A | grep -i lldb-server"
@
@echo ========Forwarding ports from local 11966 to remote 11966========
@adb forward tcp:11966 tcp:11966
@
@echo ========Run program========
adb shell su -c /data/local/tmp/lldb-server g :11966 --attach $(ps -A ^| grep %1 ^| awk '{print $2}') 2>&1 1>nul
```

Once Android Studio NDK is installed, on windows we can run the following to execute the LLDB client:

```
C:\Users\<USER>\AppData\Local\Android\Sdk\ndk\<VERSION>\toolchains\llvm\prebuilt\windows-x86_64\bin\lldb.cmd %* --source <LLDB_SCRIPT>
```

Where <USER> is the username AndroidStudio NDK was installed at, <VERSION> Might change according to the latest version
<LLDB_SCRIPT> is a path to the LLDB script, for example:

```
```

```
command script import <PATH_TO_PYTHON_LLDB_SCRIPT>

platform select remote-android
gdb-remote 11966

b fopen
continue
```