# Generating the Application Image

This section describes generating an application image (.img) and its virtual image (.img.vm) using the ELF.

## In Windows

An application image and its virtual image must be generated over command line for programming the application over JTAG/SWD. From the SDK directory, execute the following steps to generate the application image:

|  |
| --- |
| python .\script\boot.py --output app.img <path to the generated elf> <boot argument 1> <boot argument 2> |

For Example:

|  |
| --- |
| python .\script\boot.py --output app.img .\examples\http\_client\bin\http\_client.elf host=httpbin.org path=/json port=443 secured=1 method=get ca\_cert=/data/httpbin\_ca.pem ssid=InnoPhase\_AE\_AP passphrase=innophaseae |

This command generates the app.img and app.img.vm in the current directory.

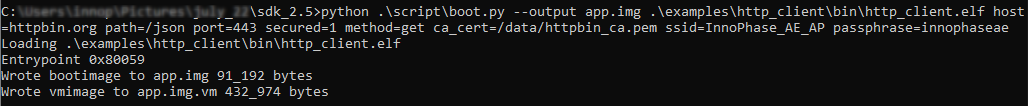


Figure 4: Generating the application image – Windows

A screenshot of a computer

Description automatically generated

Figure 5: Application image and its VM image – Windows

## In Linux

To generate the application image in Linux, execute the following command from the SDK directory

|  |
| --- |
| sudo python3 ./script/boot.py --output <application image name> <path of the application> |

For example:

|  |
| --- |
| sudo python3 ./script/boot.py --output app.img ./examples/http\_client/bin/http\_client.elf host=httpbin.org path=/json port=443 secured=1 method=get ca\_cert=/data/httpbin\_ca.pem ssid=InnoPhase\_AE\_AP passphrase=innophaseae |

This command generates the app.img and app.img.vm in the current directory.

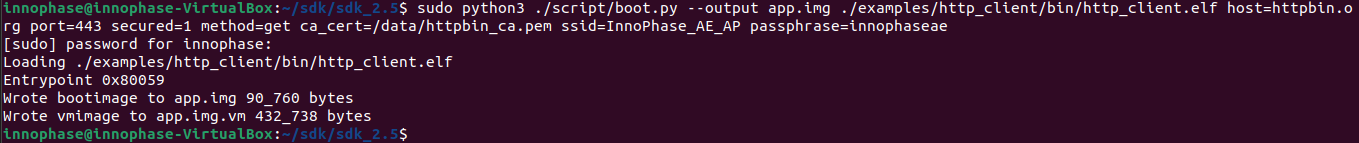


Figure 6: Generating the application image – Linux

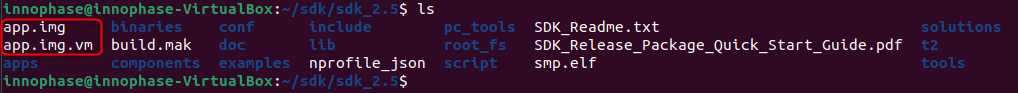


Figure 7: Application image and its VM image – Linux