

Targets

`rustc` is a cross-compiler by default. This means that you can use any compiler to build for any architecture. The list of *targets* are the possible architectures that you can build for.

To see all the options that you can set with a target, see the docs [here](#).

To compile to a particular target, use the `--target` flag:

```
$ rustc src/main.rs --target=wasm32-unknown-unknown
```

Target Features

`x86`, and `ARMv8` are two popular CPU architectures. Their instruction sets form a common baseline across most CPUs. However, some CPUs extend these with custom instruction sets, e.g. vector (`AVX`), bitwise manipulation (`BMI`) or cryptographic (`AES`).

Developers, who know on which CPUs their compiled code is going to run can choose to add (or remove) CPU specific instruction sets via the `-C target-feature=val` flag.

Please note, that this flag is generally considered as unsafe. More details can be found in [this section](#).