-l, --list <type> 列出指定模块的所有可用资源. 模块类型包括: payloads, encoders, nops,......all

-p, --payload < payload> 指定需要使用的payload(攻击荷载)。也可以使用自定义payload,几乎是支持全平台的

-f, --format < format> 指定输出格式

-e, --encoder <encoder> 指定需要使用的encoder（编码器），指定需要使用的编码，如果既没用-e选项也没用-b选项，则输出raw payload

-a, --arch < architecture> 指定payload的目标架构，例如x86 还是 x64 还是 x86\_64

-o, --out < path> 指定创建好的payload的存放位置

-b, --bad-chars < list> 设定规避字符集，指定需要过滤的坏字符。例如：不使用 '\x0f'、'\x00'

-n, --nopsled < length> 为payload预先指定一个NOP滑动长度

-s, --space < length> 设定有效攻击荷载的最大长度，就是文件大小

-i, --iterations < count> 指定payload的编码次数

-c, --add-code < path> 指定一个附加的win32 shellcode文件

-x, --template < path> 指定一个自定义的可执行文件作为模板,并将payload嵌入其中

-k, --keep 保护模板程序的动作，注入的payload作为一个新的进程运行

-v, --var-name < value> 指定一个自定义的变量，以确定输出格式

-t, --timeout <second> 从stdin读取有效负载时等待的秒数（默认为30，0表示禁用）

-h,--help 查看帮助选项

--platform < platform> 指定payload的目标平台