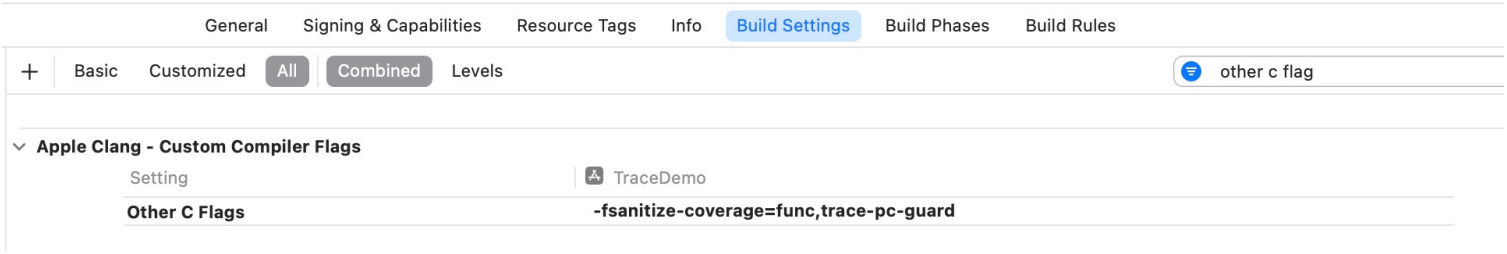
第19节课内容总结

Clang文档地址

https://clang.llvm.org/docs/SanitizerCoverage.html#tracing-pcs

给Clang编译器添加标记 `-fsanitize-coverage=func,trace-pc-guard`



__sanitizer_cov_trace_pc_guard_init

返回项目当中符号的个数。

__sanitizer_cov_trace_pc_guard

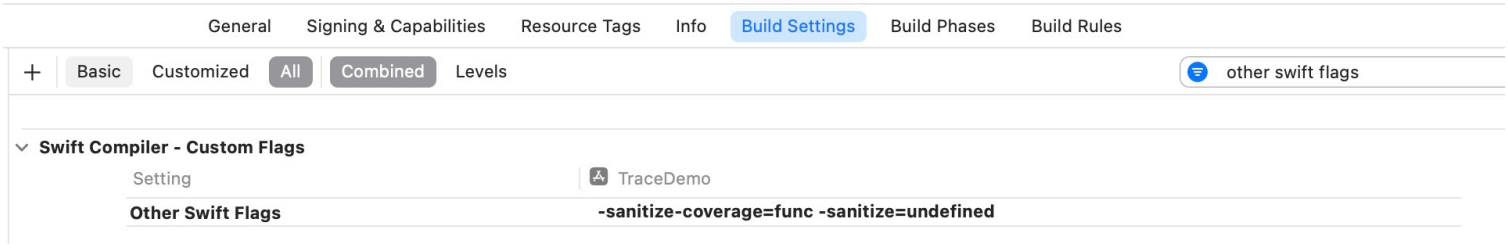
`__builtin_return_address`：返回当前函数的地址。
`dladdr`：可获得一个函数名称和地址。第一个参数：函数的地址。第二个参数：`Dl_info`结构体指针。

```
typedef struct dl_info {
        const char      *dli_fname;     /* Pathname of shared object */
        void            *dli_fbase;     /* Base address of shared object */
        const char      *dli_sname;     /* Name of nearest symbol */
        void            *dli_saddr;     /* Address of nearest symbol */
} Dl_info;
```

通过这俩个函数就能拿到当前已经被调用过的符号。

拦截swift符号

给swift编译器添加俩个标签`-sanitize-coverage=func`，`-sanitize=undefined`


General
Signing & Capabilities
Resource Tags
Info
Build Settings
Build Phases
Build Rules
Basic
Customized
All
Combined
Levels
other swift flags
Swift Compiler - Custom Flags
Setting
TraceDemo
Other Swift Flags
-sanitize-coverage=func -sanitize=undefined