```
    // in default case, text region is read-only, so we need to change the page
to writable
    int mprotectRes = mprotect(pagePointer, PAGESIZE, PROT_READ | PROT_WRITE |
PROT_EXEC);
```

上面代码中，首先计算页对齐地址，然后修改了相应页面的访问权限，如果返回值为 0，则说明成功。只有这种情况下我们才能修改代码段的数据。

4. 修改指令

当确定了指令地址并且将该地址设置为可写以后，修改指令的工作仅仅剩下是一个很简单的赋值了。

需要特别注意的是，在修改指令前，必须先确认要修改的指令确实就是目标指令，因为如果二进制库更新以后，指令偏移量可能发生变化，如果不加判断地直接修改，其结果多半就是崩溃。

另外，在判断目标指令时还需要了解的是，包含地址信息的指令通常不适合作为判断依据，因为地址可能会在重定位时或者 ***prelink*** 时被修改。

本例中，首先对目标指令以及其后续指令进行了确认，确认无误的情况下，再修改目标指令。

```
        // We must check target instruction carefully to avoid any crash
        // If seeked instruction is target, change language ID in the instruction.
The first byte of the
        // instruction is language id.
        if((*(instruction + 1) == targetInstruction[1]) && (*(instruction + 2)
== targetInstruction[2])
                && (*(instruction + 3) == targetInstruction[3])) {

            //Check next instruction, length of ARM instruction is 4 bytes
            nextInstructionPointer = instruction + 4;
            if((*(nextInstructionPointer + 0) == nextInstruction[0]) &&
(*(nextInstructionPointer + 1) == nextInstruction[1])
                && (*(nextInstructionPointer + 2) == nextInstruction[2]) &&
(*(nextInstructionPointer + 3) == nextInstruction[3])) {

                // Both target instruction and next instruction is OK, update
target instruction here
                // Change the instruction by language
                if(region == IDictionary::CHINESE_MAINLAND) {
```