

# SHCTF 2024 Excel CPU 题目说明

你们喜闻乐见的 Excel CPU 逆向（简单版）来啦🐱

这是一个 flag 解密机，你只需要昼夜不停地按 F9 键，本周结束前 flag 就会自动出现在第 142 行了！

.....或者，你也可以用 LilRan 写的反汇编器，获得解密机的汇编程序。

## 文件结构

- `shctf-challenge` 文件夹是**题目本体**，里面的 `ROM.xlsx` 就是**唯一一个**与 flag 有关的文件。使用 `excelCPU/CPU.xlsx` 模拟运行 `ROM.xlsx`，将自动解密出 flag
- `excelCPU` 文件夹是从 <https://github.com/lnkboxSoftware/excelCPU> 直接下载的原项目，未作任何修改，可以用来玩用来实际运行题目的 `ROM.xlsx`，但是**在预期解获得 flag 的过程中不会用到**
  - `CPU.xlsx` 是软件模拟的“CPU”本体
  - `ROM.xlsx` 是软件模拟的固件，是程序源码在汇编后的产物，每个程序在汇编后得到一个 `ROM.xlsx`，在 CPU 实际运行前载入到模拟的“内存”；**原项目的 `ROM.xlsx` 仅作参考**
  - `instructionSet.xlsx` 为指令集速查手册
  - `compileExcelASM16.py` 为汇编器
- `gift-from-LilRan` 文件夹是从 <https://github.com/Lil-Ran/Excel-CPU-Disassembler> 直接下载的反汇编脚本，未作任何修改，可供参考或使用。**在预期解获得 flag 的过程中无需关注其代码实现，直接用就行**
  - 通过 `python Excel-CPU-Disassembler.py -h` 查看用法
  - 示例：`python Excel-CPU-Disassembler.py ROM.xlsx -a -o dis.s`

总的来说，直接使用反汇编脚本得到题目的汇编代码，分析汇编代码，你就可以知道密文和解密 flag 的过程，进而解出 flag。

## Excel-ASM16 汇编注意点

完整的语法可查看 `excelCPU/README.md`、`excelCPU/instructionSet.xlsx` 及汇编器实现 `excelCPU/compileExcelASM16.py`，但它与 ARM 或 x86 中最常见的一些指令很相似，这里快速介绍要点：

1. 不同于 x86 每个地址对应的内存单元为一个字节，Excel-ASM16 每个地址对应的内存单元为 16 位。可以不准确地理解为：一个字节（字）= 16 位。
2. 共有  $2^{16}$  个内存单元。每个地址用 16 位表示。
3. 一条指令的长度可能为 32 位 或 16 位。
4. 有 16 个通用寄存器，名称分别是 R0, R1, ..., R15。
5. 没有函数调用，没有栈，没有 call 指令。
6. CPU 启动、程序加载后，PC（程序计数器）初始化为 0，读 0 号地址对应的指令。它通常是跳转指令，跳转到代码段第一条指令处。跳转指令占两个字，数据段通常从 2 号地址开始。后面紧跟代码段。
7. @XXXX 如 @DEAD 表示一个内存地址。
8. 立即数可用十进制（如 #12345）或十六进制（如 \$C0DE）表示。
9. LOAD 指令总是把某个地方的数复制到寄存器中，STORE 指令总是把寄存器中的数复制到某个地方。举例如下：
  - LOAD R1 \$1234 ; 把 0x1234 写到 R1
  - LOAD R2 @003F ; 把内存地址 0x3F 处的值写到 R2
  - LOAD R3 R4 ; R4 存着一个内存地址，从这个内存地址取对应值写到 R3
  - STORE R5 R6 ; R6 存着一个内存地址，把 R5 的数写到这个地址处
  - STORE R7 @0048 ; 把 R7 的数写到内存地址 0x48 处
  - TRAN R8 R9 ; 直接把 R8 的数写到 R9，这一步后 R8 和 R9 值一样了
10. ROL 和 ROR 为循环左移、循环右移。

## 如何运行起来

预期解获得 flag 的过程不需要运行。

1. 将 excelCPU/CPU.xlsx 复制到 shctf-challenge 下
2. 先用 Microsoft Excel 打开 ROM.xlsx，再用同一软件打开 CPU.xlsx，注意先后顺序
3. 如出现安全警告，需要启用内容
4. 在 Excel 软件的文件 -> 选项 -> 公式 中，打开“启用迭代计算”，“最多迭代次数”设为 1
5. 手动修改 CPU.xlsx S2 单元格（READ ROM 文字旁边）的值为 1，这将加载 ROM.xlsx
6. 手动修改 CPU.xlsx S2 单元格（READ ROM 文字旁边）的值为 0
7. 按 F9 进行一步重新计算，你将看到 CLOCK 变为 0，PC 变为 55
8. 必须在一次计算完成后、Excel 软件左下角显示“就绪”，再按下一次 F9，否则会竞争冒险、导致数据错乱。大约每秒能按三次 F9。每按两次 F9 是一个时钟周期
9. 数十个时钟周期后，PC 从 105 减小到 63，可以看到 A142 单元格出现 flag 的第一个字符 'S' 的 ASCII 码 83