8.03 IsGenericTableEmpty

Next up let's take a look at `RtlIsGenericTableEmpty`. This is yet another potentially easy function to reverse that may give us valuable information.

Once again, let's try to predict how it could work. The table may contain a member that is set or changed when it's not empty. Another possibility is that it uses the member we found previously that has to do with the number of table elements, what we're calling `NumOfElements`.

Disassembly of `RtlIsGenericTableEmpty`:

```
CMP QWORD PTR DS:[RCX], 0x0
SETE AL
RET
```

Once again, it's a very simple function.

- Compare the first element in the table (passed via RCX) to 0.
- `SETE` Sets the byte in the operand to 1 if the Zero Flag (ZF) is set, otherwise it sets the operand to 0. That means if RCX is zero, AL is set to 1 and the function returns AL.

> It's common to come across Assembly instructions you haven't seen before, so don't be afraid to look online for answers.

This function will check if the first member is zero, and if it is the function returns true. If not, then it returns false. This tells us that the first member in a generic table is not zero when there are elements in the table.

In the end, we didn't learn much, but any bit helps. We should add this information to our prediction.

```
struct Table{
    QWORD Member1;  //Nonzero when the table has elements.
    QWORD_PTR Member2;
    QWORD_PTR Member3;
    QWORD_PTR Member4;
    QWORD Member5;
    DWORD NumOfElements;
    QWORD Member7;
    QWORD Member8;
    QWORD Member9;
};
```

<- Previous Lesson
Next Lesson -> - WIP

Chapter Home