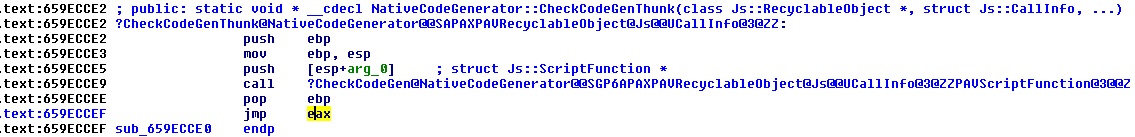
**CRASH IE 11 – MUTATION OBSERVER**

The vulnerability is in the mutationObserver function. It happens when this function tries to keep track of an element that has been already destroyed before.

From the code of the Proof of Concept (PoCie11.html) we can see different elements getting created (5 in total). At line 12, 28,44,48 and 53.

The program will crash at a **JMP EAX** Instruction in jscript9.dll



At this point the eax register will contain an invalid address in the form of: XXXX0000 where the 0 are the constant and the X changes at every crash. Thus the Internet Explorer will crash when trying to jump at those addresses.

We don’t have control over EIP.

Ideally we would need to spray the heap, so that when the program will try to jump into the address contained in the eax register, will instead jump right into our shellcode. The problem with this, is the constant end of the address (the 0’s) since jumping right there wouldn’t be doable due to the headers present at the beginning of an heap page which we can’t overwrite.

The file attached to this document are: PoCie11.html (the trigger)

Spray11.html (the standalone heap spray working on ie11)

PoC\_Spray.html (the trigger and heap spray combined)