

Forum:[Hack Forums](#)

Thread: [Understanding decompiled code \(.net\)](#)

peterX99 - March 20, 2019, 1:08 pm

Hi there,

I'm reverse engineering an program which updates the firmware of a PIC microcontroller. The program is downloading a "main file" and a couple of language files. All files seem to be encrypted. In my opinion, the file gets read by the updater, decrypted and then sent to the microcontroller.

I've opened the updater in a .net disassembler and indeed there is a conspicuous line which is calling a function "uncrypt".

Now I'm stuck because of a few lines which I've difficulties to understand:

First question

```
Code // inside {} <CppImplementationDetails>
namespace <CppImplementationDetails>
{
    // Token: 0x02000386 RID: 902
    [DebugInfoInPDB]
    [NativeCppClass]
    [UnsafeValueType]
    [MiscellaneousBits(65)]
    internal struct cKeySv$$BY0A@E
    {
    }
}
```

```
// inside "<module>"
[FixedAddressValueType]
internal static cKeySv$$BY0A@E cKeySv;
and
Code <Module>.cKeySv= 130;
*(ref <Module>.cKeySv+ 1) = 249;
*(ref <Module>.cKeySv+ 2) = 156;
.....
*(ref <Module>.cKeySv+ 15) = 93;
.....
```

Question: Is this an array being initialized? If so, of what type? (int?)
The last 4 lines of code indicating to me that we're writing 4 numbers to that array (?)

Second question

```
Code sbyte b2 = (sbyte)*(ref <Module>.cKey + 6));
b2 ^= *(ref <Module>.cKey + 11);
Question:
What is going on here?
```

My guess:

L1: We're taking the value at the 6th position of the array(?) cKey and storing it to b2

L2: b2 XOR with the value at the 11th position of the array(?) cKey

Thanks and best regards