

Forum:[Hack Forums](#)

Thread: [how to replace a string to call a function using dnlib?](#)

truco - January 2, 2022, 1:05 pm

hello,

i'm trying to replace all available strings inside a program to call a function but send the string as param.

for example:

```
Code Result.Append(b.ToString("String"));
```

i'm trying to replace to something like this (using dnlib):

```
Code Result.Append(b.ToString(Class1.Aes256Decrypt("EncryptedString")));
```

so basically what i'm trying to do is encrypt every string inside the program using aes256 and decrypt them when it's running

Here is what i have so far:

Code ...

```
...
foreach (var method in type.Methods)
{
    if (!method.HasBody)
        continue;
    if (method == decryptMethod)
        continue;

    method.Body.KeepOldMaxStack = true;

    for (var i = 0; i < method.Body.Instructions.Count; i++)
    {
        if (method.Body.Instructions[i].OpCode != OpCodes.Ldstr) continue;
        var oldString = method.Body.Instructions[i].Operand.ToString(); //Original String
        method.Body.Instructions[i].Operand = Algorithm.Aes256.EncryptString(oldString); //i
        encrypt the string here
        method.Body.Instructions.Insert(i + 1, new Instruction(OpCodes.Call, decryptMethod));
        //what do i use for decryptMethod?
    }

    method.Body.SimplifyBranches();
    method.Body.OptimizeBranches();
}
```

I just don't know how to replace the string with a function calling that specific string

0xDEAD10CC - January 2, 2022, 9:30 pm

The method would have to return a string to be suitable for the parameter replacement. You'd effectively be replacing the `Ldstr` instruction with a new set of instructions to load the argument onto the stack, load the string (i.e. the existing `Ldstr`), then an additional call instruction for the method return to be used as the new argument in the original method (presumably).

This:

Code ldstr "testing"

Becomes:

Code ldarg.0

ldstr "testing"

call instance string MyClass::MyMethod(string)

Or this depending:

Code ldarg.0

ldstr "testing"

call string MyClass::MyMethod(string)

What method are you trying to replace the string argument with? One pre-existing within the assembly? One within the .NET BCL? Or are you trying to dynamically generate one?

What is this ToString method in your example? That does not look like the correct use of the default string extension method.

truco - January 3, 2022, 3:07 am

(Jan 2, 2022 10:30 PM)0xDEAD10CC Wrote: [\[CITE\]](#) The method would have to return a string to be suitable for the parameter replacement. You'd effectively be replacing the ldstr instruction with a new set of instructions to load the argument onto the stack, load the string (i.e. the existing ldstr), then an additional call instruction for the method return to be used as the new argument in the original method (presumably).

This:

Code ldstr "testing"

Becomes:

Code ldarg.0

ldstr "testing"

call instance string MyClass::MyMethod(string)

Or this depending:

Code ldarg.0

ldstr "testing"

call string MyClass::MyMethod(string)

What method are you trying to replace the string argument with? One pre-existing within the assembly? One within the .NET BCL? Or are you trying to dynamically generate one?

What is this ToString method in your example? That does not look like the correct use of the default string extension method.

thank you for your answer!

so basically calling the function

Code Class1.Aes256Decrypt("EncryptedString"); will decrypt the encrypted string using aes and return it decrypted (so the program can run).

the aes decryption function already exists within the program, i don't know how to connect it with dnlib.

so for example, it will look something like this

Code MessageBox.Show("hello"); <- original line

replaced with :

Code MessageBox.Show(Class1.Aes256Decrypt("Rtei3gTSQakRdqQpN05jCA==")); <- "hello"

aes256 encrypted, the function will return the decrypted string

the sample code is taken here : <https://github.com/srn-g/EncryptStrings> but he encrypts the strings using base64 (plus it's not calling a pre-existing function), i'm trying to encrypt using aes256