Лабораторная работа №3. Устранение защиты по регистрации оконного приложения с использованием сторонних инструментов

<u>Цель</u>: ознакомиться с возможностями дизассемблеров для восстановления, анализа и модификации исходных кодов .Net-приложений

<u>Инструменты и дополнительные файлы</u>: дизассемблер <u>ILSpy</u> с плагином <u>Reflexil</u>, учебное приложение <u>HackOfCode.exe</u> и файл лицензии к нему e1f35abgds537dhe59yud.lic.

Задание 1. Знакомство с учебным примером защищенного приложения

1. Загрузите файл учебного приложения **HackOfCode.exe** и запустите его. В результате откроется окно приложения с надписью "Unregistered", указывающее на то, что используется незарегистрированное приложение:



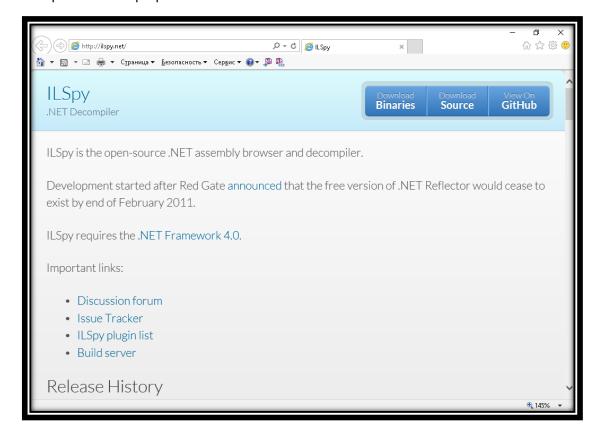
Функциональность приложения минимальна: оно отображает рисунок и завершает свою работу после нажатия кнопки «**Ок**». Пользователю предлагается зарегистрироваться и получить файл лицензии на право пользования приложением. При размещении этого файла в каталоге размещения приложения метка незарегистрированного приложения пропадает.

- 2. Загрузите файл лицензии на право пользования учебным приложением e1f35abgds537dhe59yud.lic и разместите его в каталоге размещения приложения HackOfCode.exe.
- 3. Запустите приложение **HackOfCode**.**exe** и убедитесь, что метка "Unregistered" не отображается:



Задание 2. Дизассемблирование защищенного приложения с помощью ILSpy

1. На сайте программы <u>ILSpy .NET Decompiler</u> ознакомьтесь с требованиями к установке, функциональными особенностями, списком плагинов, демонстрационным роликом и скриншотами программы:



- 2. Загрузите последнюю версию дистрибутива **ILSpy**, поставляемую в виде zip-архива (**ILSpy_Master_2.4.0.1963_Binaries.zip**).
- 3. Распакуйте загруженный архив и запустите на выполнение файл **ILSpy.exe**.

4. В окне приложения **ILSpy** откройте файл **HackOfCode.exe** (с помощью команды **File-Open...(Ctrl+O)**). Открываемый файл анализируется и помещается в древовидную структуру в левой части окна. При активизации узла в правой части окна можно ознакомиться с рядом свойств¹ исследуемой сборки, показывающие описание, правообладателя, организацию разработчика и т.п.:

```
媈 ILSpy
                                                                                                        П
<u>File View H</u>elp
Ġ 🕞 🔑 🌌 뤔 C#
                                 // C:\Users\Adm\Desktop\Lab3\HackOf Code.exe

    mscorlib (4.0.0.0)

                                // HackOf Code, Version=1.0.0.0, Culture=neutral, PublicKeyToken=null

    System (4.0.0.0)

    System.Core (4.0.0.0)

                                // Global type: <Module>

    System.Xml (4.0.0.0)

                                // Entry point: HackOfCode.Program.Main

    System.Xaml (4.0.0.0)

                                // Architecture: AnyCPU (32-bit preferred)

    WindowsBase (4.0.0.0)

                                // Runtime: .NET 4.0

    PresentationCore (4.0.0.0)

  PresentationFramework (4.1  using ...

    ICSharpCode.TreeView (4.2.

                                [assembly: AssemblyVersion("1.0.0.0")]

    Mono.Cecil (0.9.5.0)

                                [assembly: Debuggable(DebuggableAttribute.DebuggingModes.IgnoreSymbolStor

    ICSharpCode,AvalonEdit (4)

                                [assembly: AssemblyCompany("APEPS")]
  🖜 ICSharpCode.Decompiler 🔞
                                [assembly: AssemblyConfiguration("")]
  ILSpy (2.2.0.1737)
                                [assembly: AssemblyCopyright("Copyright @Yu.Tarnavski 2016")]
                                [assembly: AssemblyDescription("Training example")]
[assembly: AssemblyFileVersion("1.0.0.0")]

    System.Configuration (4.0.0)

  System.ComponentModel.

    ICSharpCode.NRefactory.VI

                                [assembly: AssemblyProduct("HackOf Code")]
                                [assembly: AssemblyTitle("HackOfCode")]

    System.Xml.Ling (4.0.0.0)

                                [assembly: AssemblyTrademark("KPI")]
  ICSharpCode.NRefactory.C
                                [assembly: CompilationRelaxations(8)]
  System.Windows.Forms (4.
                                [assembly: RuntimeCompatibility(WrapNonExceptionThrows = true)]

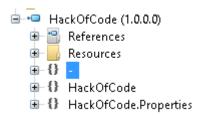
    ICSharpCode.NRefactory (5)

                                [assembly: ComVisible(false)]

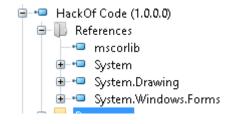
    Mono.Cecil.Pdb (0.9.5.0)

                                [assembly: Guid("17bebc32-2029-4d0a-b945-da95c63290ed")]
🕩 🕶 System.Drawing (4.0.0.0)
                                [assembly: TangetFramework(".NETFramework, Version=v4.5.2", FrameworkDisp.
  ■ HackOf Code (1.0.0.0)
```

5. Разверните узел анализируемого приложения, щелкнув по значку «+». В нем содержатся узлы ссылок (Referenses), ресурсов (Resourses) и кода приложения:

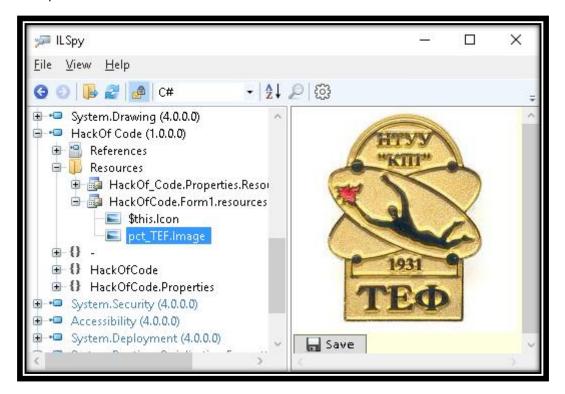


6. Разверните узел ссылок и ознакомьтесь со списком внешних сборок, которые используются приложением:

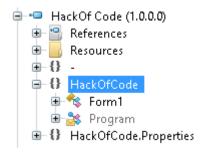


 $^{^{1}}$ Все эти свойства могут задаваться разработчиком в файле AssembyInfo.cs проекта Visual Studio.

7. Разверните узел ресурсов и ознакомьтесь с тем, какие ресурсы используются приложением:



8. Разверните узел кода приложения **HackOfCode** и ознакомьтесь с тем, какие классы используются в приложении:



Здесь отображение классов с областью видения сборки (internal) можно включить/отключить с помощью команды View-Show internal types and members или с помощью кнопки 🔒 в панели инструментов.

9. Разверните узел кода класса **Form1** для просмотра его структуры. Код выбранного свойства или метода отображается в правой части окна:

```
🗩 ILSpy
<u>File View H</u>elp
                                🦻 🕞 🔑 🌌 뤔 C#
                                      // HackOfCode.Form1
  🗷 🖺 References
                                       private void Form1_Load(object sender, EventArgs e)
  •
        Resources
  ⊕-{}
                                           this.CheckRegStat();
  if (!this.isRegistered)
     🖨 🦠 Form1
        <image> 🁔 Base Types
                                                this.lbl_Status.Text = "Unregistered!";
           🦜 Derived Types
                                                this.lbl Status.ForeColor = Color.Red;

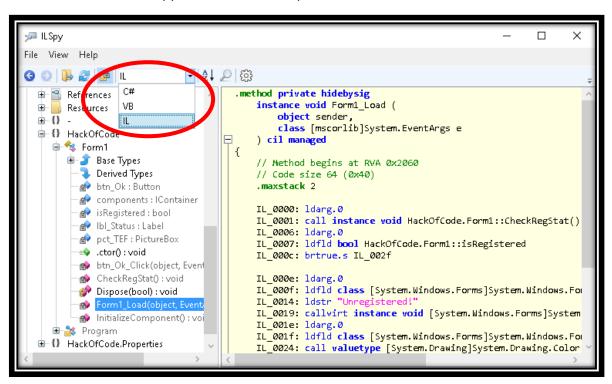
    btn_Ok : Button

                                                return:
          💣 components : IContainer
                                           this.lbl_Status.Text = "";
          🚁 isRegistered : bool
          🚁 | Ibl_Status : Label |

    pct_TEF : PictureBox

          =🔷 .ctor() : void
          🍻 btn_Ok_Click(object, Event
          酚 CheckRegStat() : void
          🧼 Dispose(bool) : void
          A Form1 Load(object, Ex
          酚 InitializeComponent() : vo
     🖶 💸 Program
  由 {} HackOfCode.Properties
```

10. По умолчанию для представления кода используется язык С#, но с помощью выпадающего списка в панели инструментов можно выбрать VB или IL:



11. В классе **Form1** выберите для просмотра метод **CheckRegStat** (), в котором, как можно предположить, реализуется логика защиты приложения:

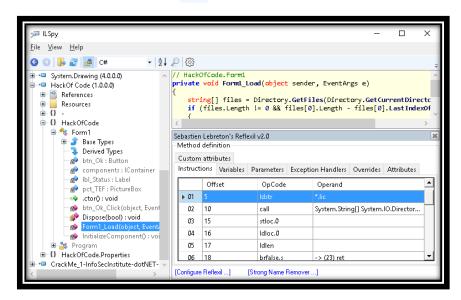
```
// HackOfCode.Form1
private void CheckRegStat()
{
   try
   {
    string[] files = Directory.GetFiles(Directory.GetCurrentDirectory(), "*.lic");
```

```
if (files.Length != 0)
      if (files[0].Length - files[0].LastIndexOf('h') == 11)
         this.isRegistered = true;
      }
      else
      {
         MessageBox.Show("License wrong!", "Error", MessageBoxButtons.OK,
MessageBoxIcon.Exclamation);
      }
    }
    else
      this.isRegistered = false;
    }
  catch (Exception arg_52_0)
    MessageBox.Show(arg 52 0.Message);
  }
}
```

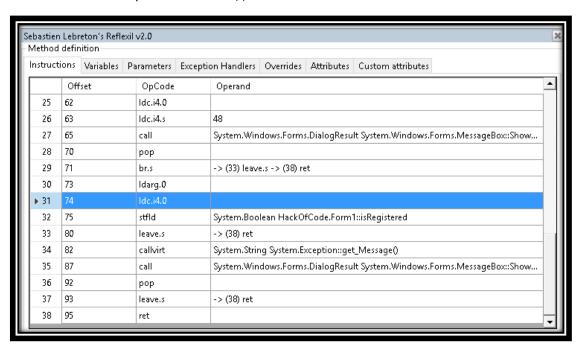
12. Проанализируйте логику работы метода CheckRegStat () и убедитесь, что возможным способом обхода защиты есть замена во внешнем операторе If-else значения переменной isRegistered с false на true. Однако выполнить редактирование метода в окне ILSpy невозможно, если не установлен соответствующий плагин – Reflexil.

Задание 3. Установка и применение плагина Reflexil

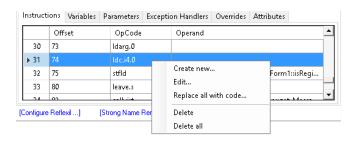
- 1. Зайдите на сайт плагина Reflexil и ознакомьтесь с его возможностями.
- 2. Загрузите zip-архив дистрибутива плагина.
- 3. Выполните установку плагина. Для этого извлеките из скачанного zip-архива файл библиотеки **Reflexil.ILSpy.Plugin.dll** и поместите его в папку установки **ILSpy**.
- 4. Вызовите установленный плагин, выбрав в главном меню **ILSpy** команду **View- Reflexil v2.0** или в панели инструментов кнопку .



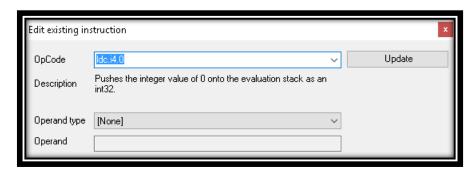
5. Выберите в дереве класса **Form1** метод **CheckRegStat** () - в панели плагина на вкладке «**Instructions**» отобразятся II-команды.



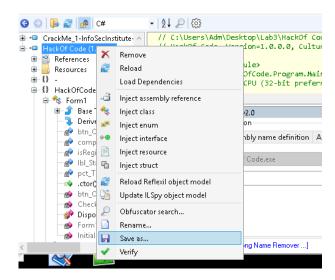
- 6. Найдите строку кода **31**, в которой инструкция **Idc.i4.0** задает значение **false** для переменной **isRegistered**.
- 7. Отредактируйте строку кода 31, заменив инструкцию **ldc.i4.0** на **ldc.i4.1**. Для перехода к редактированию строки воспользуйтесь командой **Edit...** контекстного меню:



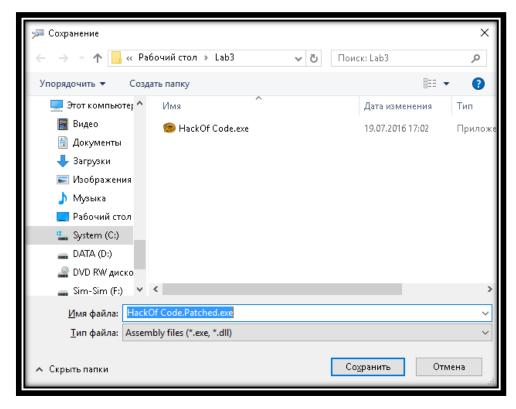
8. В окне редактирования инструкции выполните редактирование поля «**OpCode**» (заменив **Idc.i4.0** на **Idc.i4.1**) и обновите его, нажав кнопку «**Update**»



9. Сохраните внесенные изменения в новом файле *.exe, воспользовавшись командой Save as ... контекстного меню:



10. Согласитесь с предлагаемым по умолчанию именем HackOfCode.Patched.exe:



11. Запустите файл **HackOfCode.Patched.exe** и убедитесь, что защита устранена.

Требование к отчету по лабораторной работе

Отчет должен содержать титульный лист с указанием темы и цели работы, результаты выполнения п.п.1,6-8,12 задания 2, п.п.1,1 задания 3, а также выводы по работе.

Контрольные вопросы

- 1. Каковы функциональные особенности программы ILSpy?
- 2. Какие требования к установке у программы ILSpy?
- 3. Как выполняется установка программы ILSpy?
- 4. Как просмотреть кода метода класса в **ILSpy**? Какие языки программирования могут использоваться для представления кода?
- 5. Допускается ли модификация кода в **ILSpy**?
- 6. Как устанавливается плагин **Reflexil** для **ILSpy**? Как и для чего он используется?