Secure .NET programming

ante.gulam[at]ri-ing.hr

Skype: ante.gulam

Twitter: h44rp (L4uf3r)

http://www.phearless.org

Ri ing d.o.o.

fsec

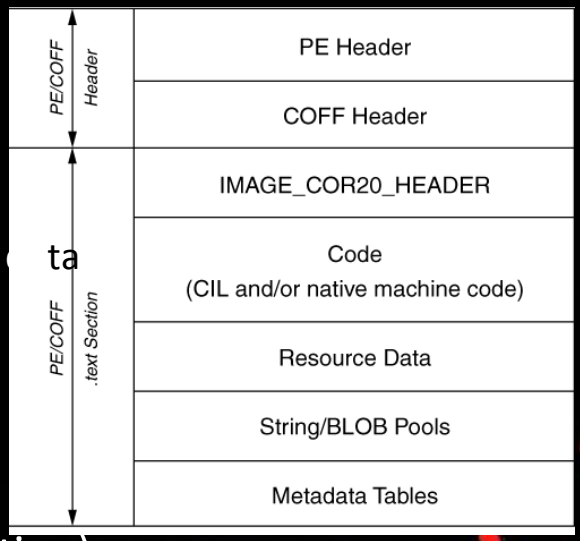
Agenda [Overview]

- .NET platform [intro (CLR, MSIL, JIT, PE/COFF..)]
- MSIL decompile/recompile (ILDAsm/ILAsm).exe
- Guidelines for safer code[SecureString class, unsafe, checked keywords, ‘foreign’ delegates..]
- Underneath the ice: vulnerabilities
- WPF/WCF configuration files[(*.config), debug=“true”, deployment retail=“true”…]
- WCF communication (SOAP/xml, REST/json) – (HTTP/TCP vs. WS) vs. SSL, hashing, auth etc.
- Runtime security – CAS vs. RBS, stack walking, reflection, advance framework configuration
- Outro [conclusion]

Ri ing d.o.o.

.NET platform [intro (CLR, MSIL, JIT, PE/COFF..)]

- About .NET framework playground (1.0 - 4.0)
  - Bottom to top: from BCL (FCL) to Linq
- Common Language Runtime engine details
- .NET assembly structure
  - PE header, CLR header, CLR metadata, CLR IL code, Native
- MSIL compiling and metalanguage basics
- Metadata (.param, .assembly, .method…)
- .NET development technologies
  - (WPF/WinForms, WCF, ASP.NET, Silverlight…)
- Default .NET security measures (OOTB)
  - Buffer overruns (System.IndexOutOfRangeException)
  - Format strings StringBuilder.AppendFormat (%n prevention)
  - Arithmetic overflow (unsigned/signed mix error (cast ☺))
  - CS0123 error (f() pointer type check), InvalidCastException (base/derived)
- Source code manipulation?
- MSIL reversing (dll hijacking), IL ASM – speaking CLR's language..
  - Change the logic, add backdoors, kraak and smaak



MSIL decompile/recompile (ILDAsm/ILAsm)

- Attacking executable files (!= attack on structures in memory / at runtime)
- Tools for decompilation(JetBrain dotPeek, Dis#, Salamander, .NET Reflector, IL Spy, Gray Wolf)
- ILAsm.exe/ILDasm.exe
- ILAsm basics (instruction set - Inside Microsoft .NET IL Assembler)
- Program logic → object control → access and value manipulation!!!
- Obfuscation/Deobfuscation (Salamander, Skater, Dotfuscator, Eazfuscator .NET...)
  - symbol renaming, overload and more...

Ri ing d.o.o.

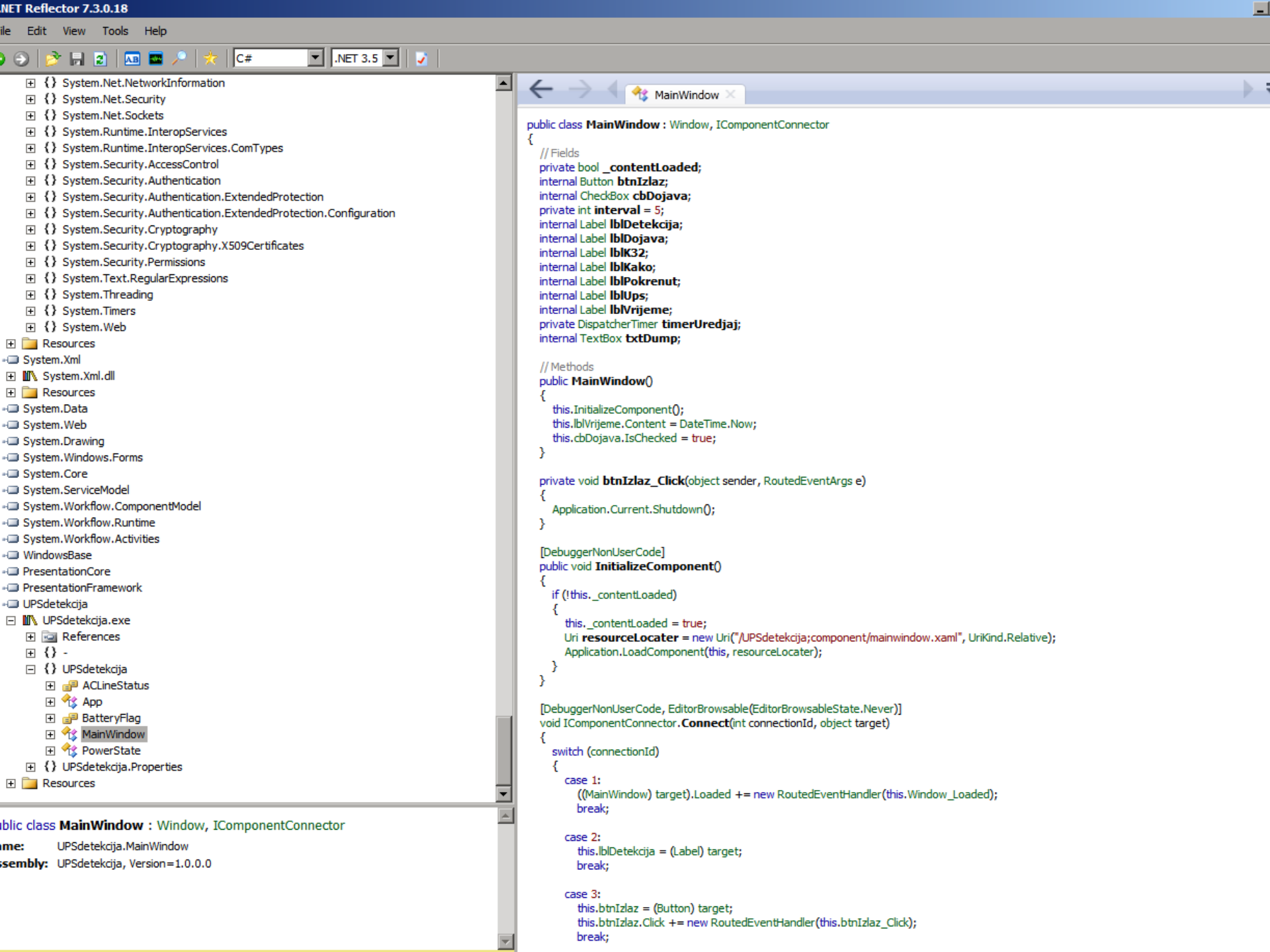
.NET obfuscation example

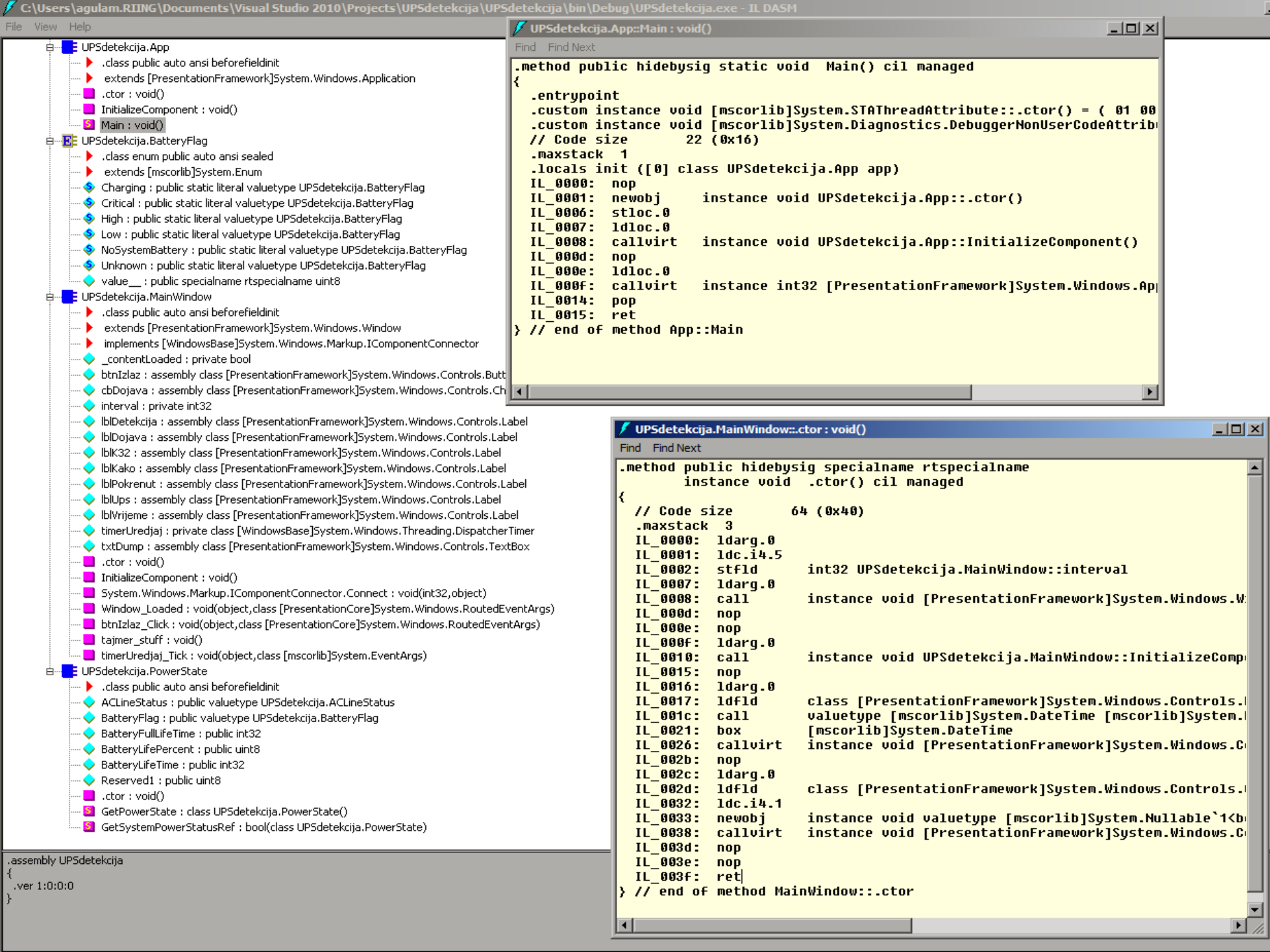
symbol renaming with overload induction

- ```
  private void IncreaseSalaries(EmployeeInfoCollection employees) {
  while (employees.HasMore()) {
  employee = employees.GetNext(true);
  employee.IncreaseSalary();
  NotifyEmployee(employee);
  }
  }
  ```
- ```
  private void a(a b) {
  while (b.a()) {
  a = b.a(true);
  a.a();
  a(a);
  }
  }
  ```

- aArticle Source: http://EzineArticles.com/6383394

Bi ing d.o.o.


.NET Reflector 7.3.0.18
ile Edit View Tools Help
C#
.NET 3.5
System.Net.NetworkInformation
System.Net.Security
System.Net.Sockets
System.Runtime.InteropServices
System.Runtime.InteropServices.ComTypes
System.Security.AccessControl
System.Security.Authentication
System.Security.Authentication.ExtendedProtection
System.Security.Authentication.ExtendedProtection.Configuration
System.Security.Cryptography
System.Security.Cryptography.X509Certificates
System.Security.Permissions
System.Text.RegularExpressions
System.Threading
System.Timers
System.Web
Resources
System.Xml
System.Xml.dll
Resources
System.Data
System.Web
System.Drawing
System.Windows.Forms
System.Core
System.ServiceModel
System.Workflow.ComponentModel
System.Workflow.Runtime
System.Workflow.Activities
WindowsBase
PresentationCore
PresentationFramework
UPSdetekcija
UPSdetekcija.exe
References
-
UPSdetekcija
ACLineStatus
App
BatteryFlag
MainWindow
PowerState
UPSdetekcija.Properties
Resources
blic class MainWindow : Window, IComponentConnector
me: UPSdetekcija.MainWindow
ssembly: UPSdetekcija, Version=1.0.0.0
MainWindow
public class MainWindow : Window, IComponentConnector
{
// Fields
private bool _contentLoaded;
internal Button btnIzlaz;
internal CheckBox cbDojava;
private int interval = 5;
internal Label lblDetekcija;
internal Label lblDojava;
internal Label lblK32;
internal Label lblKako;
internal Label lblPokrenut;
internal Label lblUps;
internal Label lblVrijeme;
private DispatcherTimer timerUredjaj;
internal TextBox txtDump;
// Methods
public MainWindow()
{
this.InitializeComponent();
this.lblVrijeme.Content = DateTime.Now;
this.cbDojava.IsChecked = true;
}
private void btnIzlaz_Click(object sender, RoutedEventArgs e)
{
Application.Current.Shutdown();
}
[DebuggerNonUserCode]
public void InitializeComponent()
{
if (!this._contentLoaded)
{
this._contentLoaded = true;
Uri resourceLocater = new Uri("/UPSdetekcija;component/mainwindow.xaml", UriKind.Relative);
Application.LoadComponent(this, resourceLocater);
}
}
[DebuggerNonUserCode, EditorBrowsable(EditorBrowsableState.Never)]
void IComponentConnector.Connect(int connectionId, object target)
{
switch (connectionId)
{
case 1:
((MainWindow) target).Loaded += new RoutedEventHandler(this.Window_Loaded);
break;
case 2:
this.lblDetekcija = (Label) target;
break;
case 3:
this.btnIzlaz = (Button) target;
this.btnIzlaz.Click += new RoutedEventHandler(this.btnIzlaz_Click);
break;


C:\Users\agulam.RIING\Documents\Visual Studio 2010\Projects\UPSdetekcija\UPSdetekcija\bin\Debug\UPSdetekcija.exe - IL DASM
File View Help
UPSdetekcija.App
.class public auto ansi beforefieldinit
extends [PresentationFramework]System.Windows.Application
.ctor : void()
InitializeComponent : void()
Main : void()
UPSdetekcija.BatteryFlag
.class enum public auto ansi sealed
extends [mscorlib]System.Enum
Charging : public static literal valuetype UPSdetekcija.BatteryFlag
Critical : public static literal valuetype UPSdetekcija.BatteryFlag
High : public static literal valuetype UPSdetekcija.BatteryFlag
Low : public static literal valuetype UPSdetekcija.BatteryFlag
NoSystemBattery : public static literal valuetype UPSdetekcija.BatteryFlag
Unknown : public static literal valuetype UPSdetekcija.BatteryFlag
value__ : public specialname rtspecialname uint8
UPSdetekcija.MainWindow
.class public auto ansi beforefieldinit
extends [PresentationFramework]System.Windows.Window
implements [WindowsBase]System.Windows.Markup.IComponentConnector
_contentLoaded : private bool
btnIzlaz : assembly class [PresentationFramework]System.Windows.Controls.Butt
cbDojava : assembly class [PresentationFramework]System.Windows.Controls.Ch
interval : private int32
lblDetekcija : assembly class [PresentationFramework]System.Windows.Controls.Label
lblDojava : assembly class [PresentationFramework]System.Windows.Controls.Label
lblK32 : assembly class [PresentationFramework]System.Windows.Controls.Label
lblKako : assembly class [PresentationFramework]System.Windows.Controls.Label
lblPokrenut : assembly class [PresentationFramework]System.Windows.Controls.Label
lblUps : assembly class [PresentationFramework]System.Windows.Controls.Label
lblVrijeme : assembly class [PresentationFramework]System.Windows.Controls.Label
timerUredjaj : private class [WindowsBase]System.Windows.Threading.DispatcherTimer
txtDump : assembly class [PresentationFramework]System.Windows.Controls.TextBox
.ctor : void()
InitializeComponent : void()
System.Windows.Markup.IComponentConnector.Connect : void(int32,object)
Window_Loaded : void(object,class [PresentationCore]System.Windows.RoutedEventArgs)
btnIzlaz_Click : void(object,class [PresentationCore]System.Windows.RoutedEventArgs)
tajmer_stuff : void()
timerUredjaj_Tick : void(object,class [mscorlib]System.EventArgs)
UPSdetekcija.PowerState
.class public auto ansi beforefieldinit
ACLineStatus : public valuetype UPSdetekcija.ACLineStatus
BatteryFlag : public valuetype UPSdetekcija.BatteryFlag
BatteryFullLifeTime : public int32
BatteryLifePercent : public uint8
BatteryLifeTime : public int32
Reserved1 : public uint8
.ctor : void()
GetPowerState : class UPSdetekcija.PowerState()
GetSystemPowerStatusRef : bool(class UPSdetekcija.PowerState)
.assembly UPSdetekcija
{
.ver 1:0:0:0
}
UPSdetekcija.App::Main : void()
Find Find Next
.method public hidebysig static void Main() cil managed
{
.entrypoint
.custom instance void [mscorlib]System.STAThreadAttribute::.ctor() = (01 00
.custom instance void [mscorlib]System.Diagnostics.DebuggerNonUserCodeAttrib
// Code size 22 (0x16)
.maxstack 1
.locals init ([0] class UPSdetekcija.App app)
IL_0000: nop
IL_0001: newobj instance void UPSdetekcija.App::.ctor()
IL_0006: stloc.0
IL_0007: ldloc.0
IL_0008: callvirt instance void UPSdetekcija.App::InitializeComponent()
IL_000d: nop
IL_000e: ldloc.0
IL_000f: callvirt instance int32 [PresentationFramework]System.Windows.Ap
IL_0014: pop
IL_0015: ret
} // end of method App::Main
UPSdetekcija.MainWindow::.ctor : void()
Find Find Next
.method public hidebysig specialname rtspecialname
instance void .ctor() cil managed
{
// Code size 64 (0x40)
.maxstack 3
IL_0000: ldarg.0
IL_0001: ldc.i4.5
IL_0002: stfld int32 UPSdetekcija.MainWindow::interval
IL_0007: ldarg.0
IL_0008: call instance void [PresentationFramework]System.Windows.W
IL_000d: nop
IL_000e: nop
IL_000f: ldarg.0
IL_0010: call instance void UPSdetekcija.MainWindow::InitializeComp
IL_0015: nop
IL_0016: ldarg.0
IL_0017: ldfld class [PresentationFramework]System.Windows.Controls.
IL_001c: call valuetype [mscorlib]System.DateTime [mscorlib]System.
IL_0021: box [mscorlib]System.DateTime
IL_0026: callvirt instance void [PresentationFramework]System.Windows.C
IL_002b: nop
IL_002c: ldarg.0
IL_002d: ldfld class [PresentationFramework]System.Windows.Controls.
IL_0032: ldc.i4.1
IL_0033: newobj instance void valuetype [mscorlib]System.Nullable`1<b
IL_0038: callvirt instance void [PresentationFramework]System.Windows.C
IL_003d: nop
IL_003e: nop
IL_003f: ret
} // end of method MainWindow::.ctor

Guidelines for safer code[SecureString class, unsafe, checked keywords, delegates…]

- Memory dumping (SecureString mandatory!!)
- Unmanaged code (when and where go ‘unsafe’?)
  - int* p = stackalloc int[32];
- Checked keyword in practice (OverflowException)
- Assert usage? (CAS classes, PermissionSet class)
- Security of delegates “from outside”
  - SecurityPermission(SecurityPermissionFlag.Execution).PermitOnly();
  - AllowPartiallyTrustedCallersAttribute
- Input validation (SQL..), hardcoding, exception throwing, assemblies, privileges, crypto…
- Code signing (GAC)
  - Authenticode (chaining certificates), Strong Names (PKI)
  - MakeCat , SignTool, Strong Name Tool (sn.exe)

Ri ing d.o.o.

Guidelines for safer code[SecureString class, unsafe, checked keywords, delegates…]

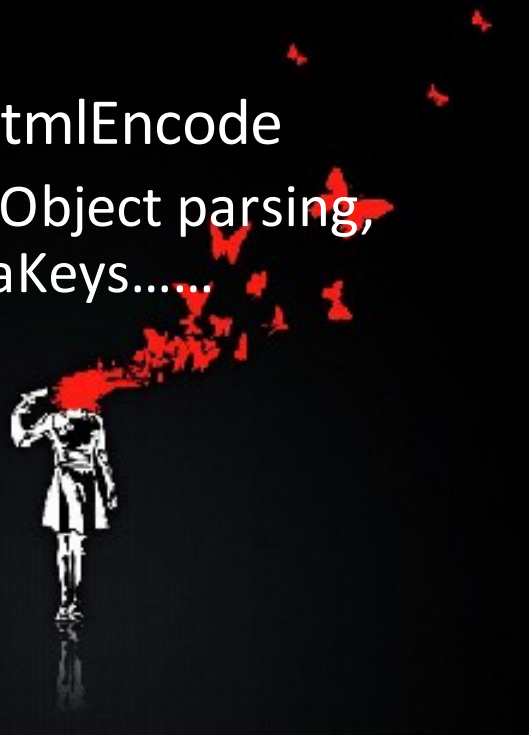
- Memory dumping (SecureString mandatory!!)
- Unmanaged code (when and where go 'unsafe'?)
  - int* p = stackalloc int[32];
- Checked keyword in practice (OverflowException)
- Assert usage? (CAS classes, PermissionSet class)
- Security of delegates "from outside"
  - SecurityPermission(SecurityPermissionFlag.Execution).PermitOnly();
  - AllowPartiallyTrustedCallersAttribute
- Input validation (SQL..), hardcoding, exception throwing, assemblies, privileges, crypto…
- Code signing (GAC)
  - Authenticode (chaining certificates), Strong Names (PKI)
  - MakeCat , SignTool, Strong Name Tool (sn.exe)

```
System.Security.SecureString X = new System.Security.SecureString();
secString.AppendChar(p);

IntPtr p = System.Runtime.InteropServices.Marshal.SecureStringToBSTR(X);
string  dekript = System.Runtime.InteropServices.Marshal.PtrToStringUni(p);

secString.Dispose();
```

Ri ing d.o.o.

Underneath the ice: vulnerabilities

- Web vulnerabilities as we know them (XSS, SQL injection, CSRF, Response splitting (CRLF injection), SOAP injection, HPP, Xpath injection, File upload/download, directory traversal …)
- Using technologies like LINQ/Entity framework as SQL query language for data sanitization
- ValidateRequest="true“ inside Machine.config
- Compare validator <asp:CompareValidator >, htmlEncode
- Games without frontiers: Oracle Padding (CBC), Object parsing, Direct Object Reference, Error handling, GV DataKeys……
- TFS – Check-in/Get latest version sniffing?
  - WYSIWYG on the wire/air (POST/gzip mess-up)
  - Always HTTPS for Team Explorer



Ri ing d.o.o.

```
Expect: 100-continue

HTTP/1.1 100 Continue

----------------------------8e5m2D6l5Q4h6
Content-Disposition: form-data; name="item"

[redacted]/Administracija/wPrijaviSe.xaml
----------------------------8e5m2D6l5Q4h6
Content-Disposition: form-data; name="wsname"

[redacted] <- domain computer name
----------------------------8e5m2D6l5Q4h6
Content-Disposition: form-data; name="wsowner"

[redacted] <- domain username
----------------------------8e5m2D6l5Q4h6
Content-Disposition: form-data; name="filelength"

3253
----------------------------8e5m2D6l5Q4h6
Content-Disposition: form-data; name="hash"

3mTbEmr7xSfRu0mYGZ65TQ==
----------------------------8e5m2D6l5Q4h6
Content-Disposition: form-data; name="range"

bytes=0-1198/1199

----------------------------8e5m2D6l5Q4h6
Content-Disposition: form-data; name="content"; filename="item"
Content-Type: application/gzip

[binary data]
----------------------------8e5m2D6l5Q4h6--

HTTP/1.1 200 OK
Date: Tue, 20 Sep 2011 12:
Server: Microsoft-IIS/6.0
X-Powered-By: ASP.NET
X-AspNet-Version: 2.0.5072
Cache-Control: private
Content-Length: 0

POST /VersionControl/v1.0/upload.asmx HTTP/1.1
User-Agent: Team Foundation (devenv.exe, 10.0.31118.1)
```

```
<Button Height="23" Margin="10,130,90,0“
 Name="btnPromjeni" VerticalAlignment=“
Top" Click="btnPromjeni_Click">Spremi</Button>
```

Don’t do this at home (or work)

Real-world examples:

the bad ones

The bad, the bad and the ugly

Bad practice No.1
binding ConnString to Cb control

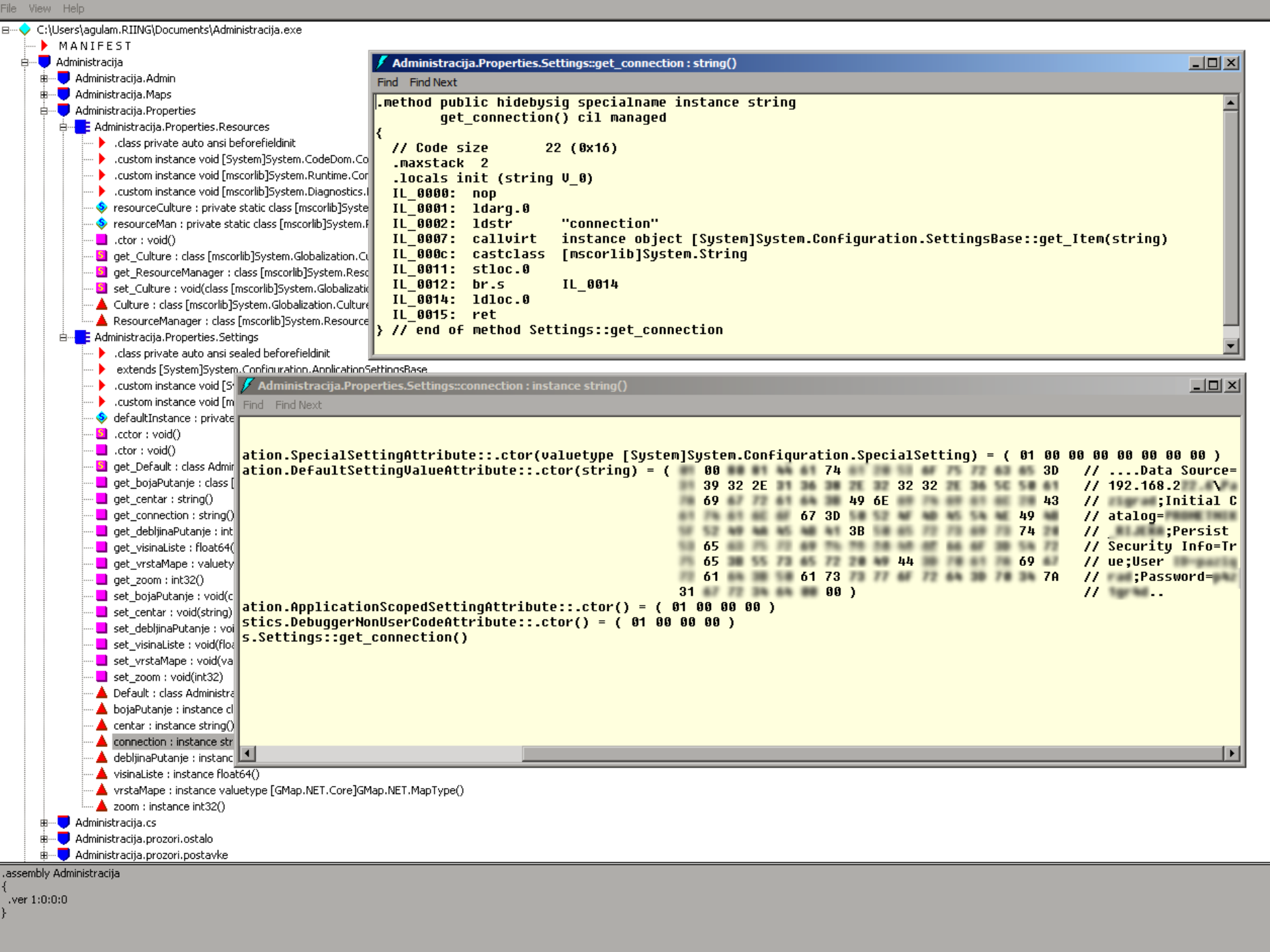
- ddlServer.DataSource = sc.DohvatiServere();
  ddlServer.DataTextField=ServerName";
  ddlServer.DataValueField=ConnString";
  ddlServer.DataBind();

ConnString = “DataSource=10.10.10.2\DB1;Initial Catalog=Database_1;Persist Security Info=True;User ID=korisnik1;Password=123#pass”

Bring d.o.o.

Bad practice No.2
ad hoc SQL query – passing input

- protected void btn1_Click(object sender, EventArgs e)
  { string dbcon = "Server=(local);Database=Northwind;Integrated Security=SSPI";
  string cmdStr = "insert into Korisnici (ImePrez, Telefon) values ('" + txtIme.Text + "', '" + txtTel.Text + "')";
  using (SqlConnection konekcija = new SqlConnection(dbcon)) = new SqlCommand(cmdStr, conn)) { conn.Open(); cmd.ExecuteNonQuery(); }
  using (SqlCommand cmd = new SqlCommand(cmdStr, konekcija)) { konekcija.Open(); cmd.ExecuteNonQuery(); }
}


File View Help
C:\Users\agulam.RIING\Documents\Administracija.exe
MANIFEST
Administracija
Administracija.Admin
Administracija.Maps
Administracija.Properties
Administracija.Properties.Resources
.class private auto ansi beforefieldinit
.custom instance void [System]System.CodeDom.Co
.custom instance void [mscorlib]System.Runtime.Cor
.custom instance void [mscorlib]System.Diagnostics.
resourceCulture : private static class [mscorlib]Syste
resourceMan : private static class [mscorlib]System.
.ctor : void()
get_Culture : class [mscorlib]System.Globalization.Cu
get_ResourceManager : class [mscorlib]System.Resc
set_Culture : void(class [mscorlib]System.Globalizatio
Culture : class [mscorlib]System.Globalization.Culture
ResourceManager : class [mscorlib]System.Resource
Administracija.Properties.Settings
.class private auto ansi sealed beforefieldinit
extends [System]System.Configuration.ApplicationSettingsBase
.custom instance void [S
.custom instance void [m
defaultInstance : private
.cctor : void()
.ctor : void()
get_Default : class Admir
get_bojaPutanje : class [
get_centar : string()
get_connection : string()
get_debljinaPutanje : int
get_visinaListe : float64(
get_vrstaMape : valuety
get_zoom : int32()
set_bojaPutanje : void(c
set_centar : void(string)
set_debljinaPutanje : voi
set_visinaListe : void(flo
set_vrstaMape : void(va
set_zoom : void(int32)
Default : class Administra
bojaPutanje : instance cl
centar : instance string()
connection : instance str
debljinaPutanje : instanc
visinaListe : instance float64()
vrstaMape : instance valuetype [GMap.NET.Core]GMap.NET.MapType()
zoom : instance int32()
Administracija.cs
Administracija.prozori.ostalo
Administracija.prozori.postavke
Administracija.Properties.Settings::get_connection : string()
Find Find Next
.method public hidebysig specialname instance string
get_connection() cil managed
{
// Code size 22 (0x16)
.maxstack 2
.locals init (string V_0)
IL_0000: nop
IL_0001: ldarg.0
IL_0002: ldstr "connection"
IL_0007: callvirt instance object [System]System.Configuration.SettingsBase::get_Item(string)
IL_000c: castclass [mscorlib]System.String
IL_0011: stloc.0
IL_0012: br.s IL_0014
IL_0014: ldloc.0
IL_0015: ret
} // end of method Settings::get_connection
Administracija.Properties.Settings::connection : instance string()
Find Find Next
ation.SpecialSettingAttribute::.ctor(valuetype [System]System.Configuration.SpecialSetting) = (01 00 00 00 00 00 00 00)
ation.DefaultSettingValueAttribute::.ctor(string) = (01 00 00 01 44 61 74 61 20 53 6F 75 72 63 65 3D // ....Data Source=
39 32 2E 31 36 38 2E 32 32 2E 36 5C 50 61 // 192.168.2
69 67 72 61 64 3B 49 6E 69 74 69 61 6C 20 43 // ;Initial C
67 3D 50 52 4D 53 54 4E 49 4D // atalog=
3B 50 65 72 73 69 73 74 20 // ;Persist
65 20 53 65 63 75 72 69 74 79 20 49 6E 66 6F 3D 54 72 // Security Info=Tr
65 38 55 73 65 72 20 49 44 3D 70 61 73 73 69 67 // ue;User
61 64 3B 50 61 73 73 77 6F 72 64 3D 70 34 7A // ;Password=
31 67 72 34 64 00 00) // ..
ation.ApplicationScopedSettingAttribute::.ctor() = (01 00 00 00)
stics.DebuggerNonUserCodeAttribute::.ctor() = (01 00 00 00)
s.Settings::get_connection()
.assembly Administracija
{
.ver 1:0:0:0
}

WPF/WCF configuration files[(*.config), debug="true", deployment retail="true"…]

- Default .NET configuration files
  - **enterprisesec.config -** enterprise-level security policies
  - **security.config -** machine-level security
  - **machine.config -** .NET environment installation settings
  - **web.config / app.config –** default settings
- Compilation debug="true" problem
  - Runtime mem, batch optimization …
- Deployment retail="true" as a solution
- Max message size value ("2147483647")
- Encryption of configuration files
  - Command line tools / inside code

Ri ing d.o.o.

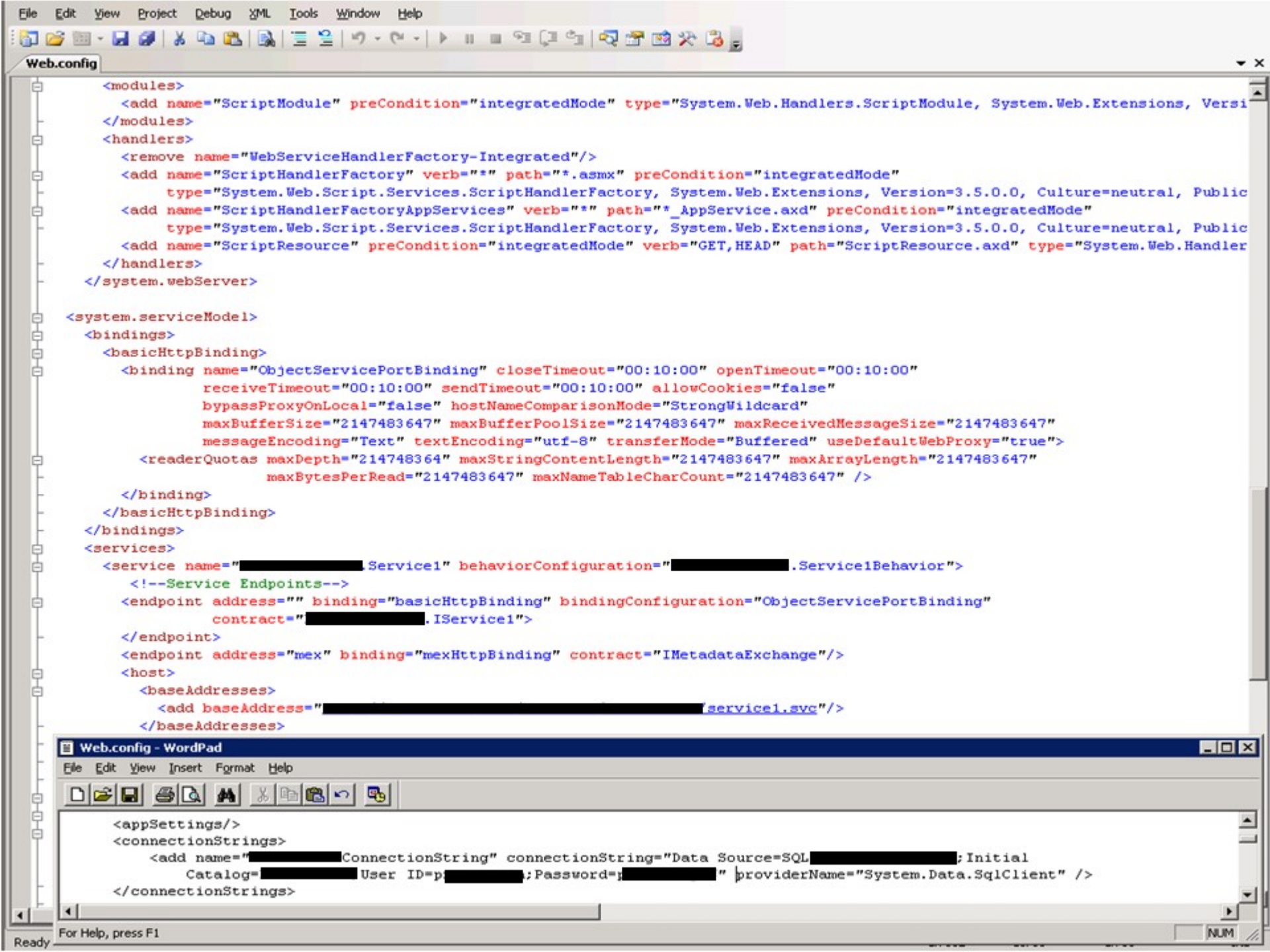
WPF/WCF configuration files[(*.config), debug="true", deployment retail="true"…]

- Default .NET configuration files
  - **enterprisesec.config -** enterprise-level security policies
  - **security.config -** machi
  - **machine.config -** .NET
  - **web.config / app.config –** default settings
- Compilation debug="true" problem
  - Runtime mer
- Deployment re
- Max message
- Encryption of
  - Command lin

[DriveLetter]:\Windows\Microsoft.Net\Framework\[.NET version number]\config

```
protected void Page_Load(object sender, EventArgs e)
{
Configuration config; ConfigurationSection configSection;
config = WebConfigurationManager.OpenWebConfiguration
(Request.ApplicationPath); configSection = config.GetSection("
connectionStrings");
if (configSection != null)
{ if (!(configSection.SectionInformation.IsLocked))
{ configSection.SectionInformation.ProtectSection
("DataProtectionConfigurationProvider"); config.Save(); } } }
```

Bi ing d.o.o.


File Edit View Project Debug XML Tools Window Help
Web.config
<modules>
<add name="ScriptModule" preCondition="integratedMode" type="System.Web.Handlers.ScriptModule, System.Web.Extensions, Versi
</modules>
<handlers>
<remove name="WebServiceHandlerFactory-Integrated"/>
<add name="ScriptHandlerFactory" verb="*" path="*.asmx" preCondition="integratedMode"
type="System.Web.Script.Services.ScriptHandlerFactory, System.Web.Extensions, Version=3.5.0.0, Culture=neutral, Public
<add name="ScriptHandlerFactoryAppServices" verb="*" path="*_AppService.axd" preCondition="integratedMode"
type="System.Web.Script.Services.ScriptHandlerFactory, System.Web.Extensions, Version=3.5.0.0, Culture=neutral, Public
<add name="ScriptResource" preCondition="integratedMode" verb="GET,HEAD" path="ScriptResource.axd" type="System.Web.Handler
</handlers>
</system.webServer>
<system.serviceModel>
<bindings>
<basicHttpBinding>
<binding name="ObjectServicePortBinding" closeTimeout="00:10:00" openTimeout="00:10:00"
receiveTimeout="00:10:00" sendTimeout="00:10:00" allowCookies="false"
bypassProxyOnLocal="false" hostNameComparisonMode="StrongWildcard"
maxBufferSize="2147483647" maxBufferPoolSize="2147483647" maxReceivedMessageSize="2147483647"
messageEncoding="Text" textEncoding="utf-8" transferMode="Buffered" useDefaultWebProxy="true">
<readerQuotas maxDepth="214748364" maxStringContentLength="2147483647" maxArrayLength="2147483647"
maxBytesPerRead="2147483647" maxNameTableCharCount="2147483647" />
</binding>
</basicHttpBinding>
</bindings>
<services>
<service name="[redacted].Service1" behaviorConfiguration="[redacted].Service1Behavior">
<!--Service Endpoints-->
<endpoint address="" binding="basicHttpBinding" bindingConfiguration="ObjectServicePortBinding"
contract="[redacted].IService1">
</endpoint>
<endpoint address="mex" binding="mexHttpBinding" contract="IMetadataExchange"/>
<host>
<baseAddresses>
<add baseAddress="[redacted]service1.svc"/>
</baseAddresses>
Web.config - WordPad
File Edit View Insert Format Help
<appSettings/>
<connectionStrings>
<add name="[redacted]ConnectionString" connectionString="Data Source=SQL[redacted];Initial
Catalog=[redacted] User ID=p[redacted];Password=p[redacted]" providerName="System.Data.SqlClient" />
</connectionStrings>
For Help, press F1
Ready
NUM

WCF communication (SOAP/xml, REST/json) - (basicHTTP vs. WS) vs. SSL, hashing, auth etc.

- WCF in details (rolling in the deep)
- SOAP vs. REST inside MS service
- Bindings and their security (basic, ws, web, nettcp, custom…)
  - transport, encoding & protocol
- SSL tunneling or secure binding (WS – SCT/RSTR)?
  - Transport vs. Message level, in-transit vs. Processing
- Filtering remote access to WCF services (IP)
- Message integrity check
  - Hashing xml/json messages (HMAC, SHA1..)
- Custom authorization for service access
  - SOAP header based, method argument, WCF session…
- Request load-balancing (WCF throttling)
  - Setting maxConcurrent(Calls (16),Sessions (10),Instances)

Ri ing d.o.o.

Runtime security – CAS vs. RBS, stack walking, reflection, advance framework configuration

- RBS - Role-Based Security
- CAS - Code-Access Security (evidence based permissions)
  - Evaluate Assembly
- Stack walking? Method access grant..
- LinkDemand vs. stack walk (Luring Attacks)
  - Immediate caller vs. all callers
- Garbage collector
- Reflection (ReflectionPermission (CAS))
  - Reflection.Emit – create assembly
  - System.Reflection.Assembly.Load
  - System.Reflection.MethodInfo.Invoke
- Configuring framework (tuning settings)

Outro [conclusion]

- Microsoft .NET platform == secure env. ???
  - This is the world as we know it: brainless development
- Stay tuned and up2date: “Keep your friends close, and your enemies closer.” Sun Tzu
- Make safest possible .NET environment (CAS, RBS..)
- Constrain and sanitize all input data
- Encrypt your config’s (Triple DES)
- WCF wargames (“You have all the weapons you need… now fight!”)
  - Encrypt, authenticate, check integrity, authorization
  - DPAPI for sensitive data (protected registry key)
- Secure assemblies (obfuscate, pack, request minimum)
  - .NETZ compressor (OSS), .netshrink (LZMA alg. + pass)..

Ri ing d.o.o.

thank you for your attention

questions and comments

?

ante.gulam[at]ri-ing.hr

Shouts: h4z4rd, c0ld, n00ne, fr1c, c0de, all gnoblets, phZine crew…