

## Obfuscator Clone: Babel Obfuscator – Ottobre 2008

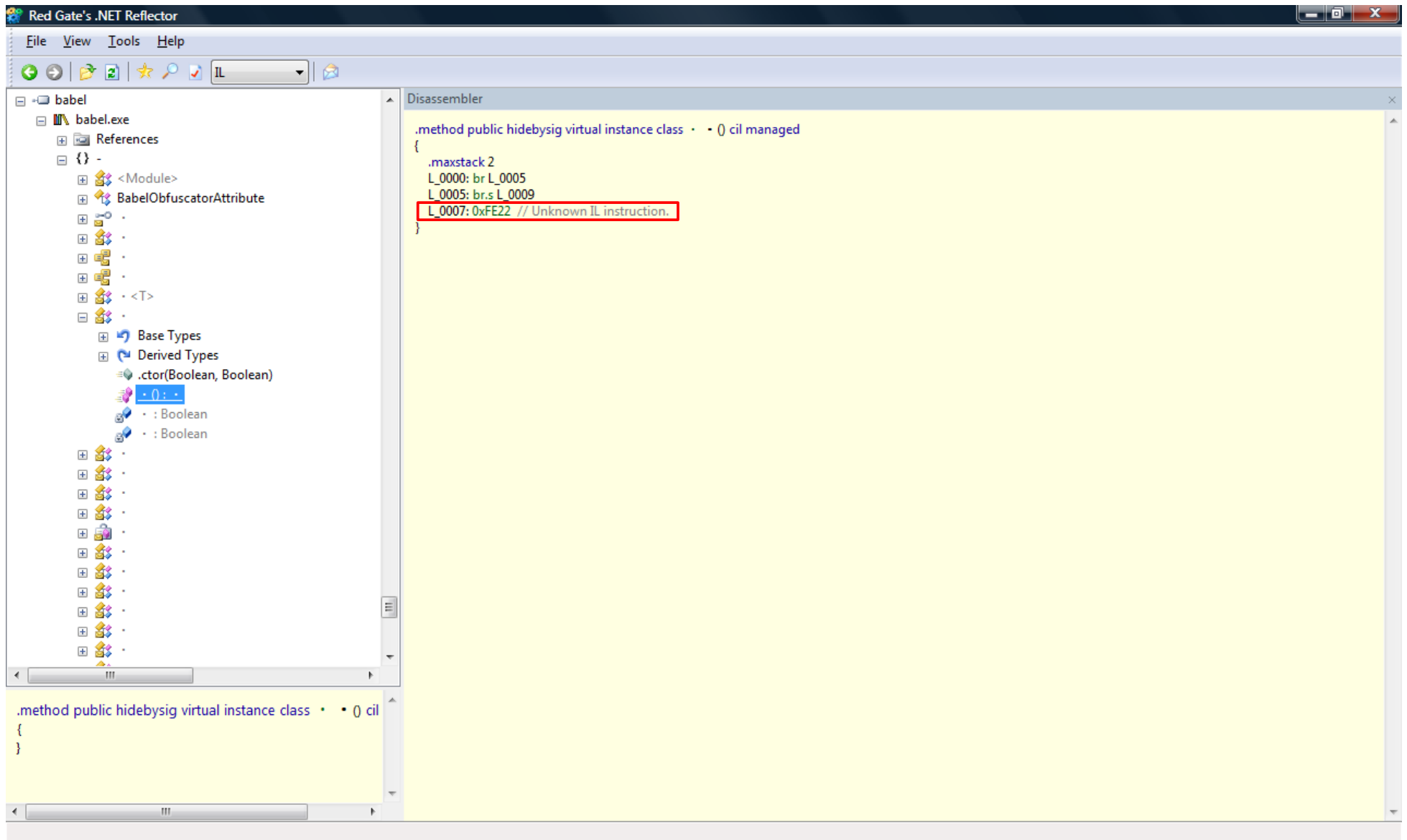
The screenshot shows the Red Gate's .NET Reflector interface. The left pane displays the assembly structure for 'babel.exe', including references and base types. The main Disassembler window shows the assembly manifest for 'babel' with the following metadata:

```
.assembly babel
{
  .ver 1:2:0:0
  .hash algorithm 0x00008004
  .publickey = (00 24 00 00 04 80 00 00 94 00 00 00 06 02 00 00 24 00 00 52 53 41 31 00 04 00 00 01 00 01 00 6B BC D8 56 A4 05 AF 17 4E 39 35 3C 2D 19 7D 7A F8 66 0C 7D
  .custom instance void [mscorlib]System.Diagnostics.DebuggableAttribute::ctor(valueType [mscorlib]System.Diagnostics.DebuggableAttribute/DebuggingModes) = { i
  .custom instance void [mscorlib]System.Runtime.CompilerServices.RuntimeCompatibilityAttribute::ctor() = { WrapNonExceptionThrows=bool(true) }
  .custom instance void [mscorlib]System.Reflection.AssemblyTitleAttribute::ctor(string) = { string('Babel Obfuscator') }
  .custom instance void [mscorlib]System.Reflection.AssemblyInformationalVersionAttribute::ctor(string) = { string('1.2.0.0') }
  .custom instance void [mscorlib]System.Reflection.AssemblyCompanyAttribute::ctor(string) = { string('http://code.google.com/p/babelobfuscator/') }
  .custom instance void [mscorlib]System.Reflection.AssemblyProductAttribute::ctor(string) = { string('Babel Obfuscator') }
  .custom instance void [mscorlib]System.Reflection.AssemblyFileVersionAttribute::ctor(string) = { string('1.2.0.0') }
  .custom instance void [mscorlib]System.Runtime.CompilerServices.CompilationRelaxationsAttribute::ctor(int32) = { int32(8) }
  .custom instance void [mscorlib]System.Reflection.AssemblyDescriptionAttribute::ctor(string) = { string('Dedicated to Francesco.') }
  .custom instance void [mscorlib]System.Reflection.AssemblyConfigurationAttribute::ctor(string) = { string('Release') }
  .custom instance void [mscorlib]System.Runtime.CompilerServices.SuppressIlDasmAttribute::ctor()
  .custom instance void BabelObfuscatorAttribute::ctor()
  .custom instance void [mscorlib]System.Reflection.AssemblyTrademarkAttribute::ctor(string) = { string('') }
  .custom instance void [mscorlib]System.Reflection.AssemblyCopyrightAttribute::ctor(string) = { string('Copyright \u00a9 2008 by Alberto Ferrazzoli') }
  .custom instance void [mscorlib]System.Runtime.InteropServices.ComVisibleAttribute::ctor(bool) = { bool(false) }
  .custom instance void [mscorlib]System.CLSCompliantAttribute::ctor(bool) = { bool(true) }
  .custom instance void [mscorlib]System.Runtime.InteropServices.GuidAttribute::ctor(string) = { string('c50d6e0b-0883-4d3c-83ce-a7bcb496ed14') }
  .custom instance void [mscorlib]System.Security.Permissions.SecurityPermissionAttribute::ctor(valueType [mscorlib]System.Security.Permissions.SecurityAction) = { v
  .custom instance void [mscorlib]System.Security.Permissions.SecurityPermissionAttribute::ctor(valueType [mscorlib]System.Security.Permissions.SecurityAction) = { v
}
```

The Disassembler window also shows a partial view of the assembly manifest at the bottom:

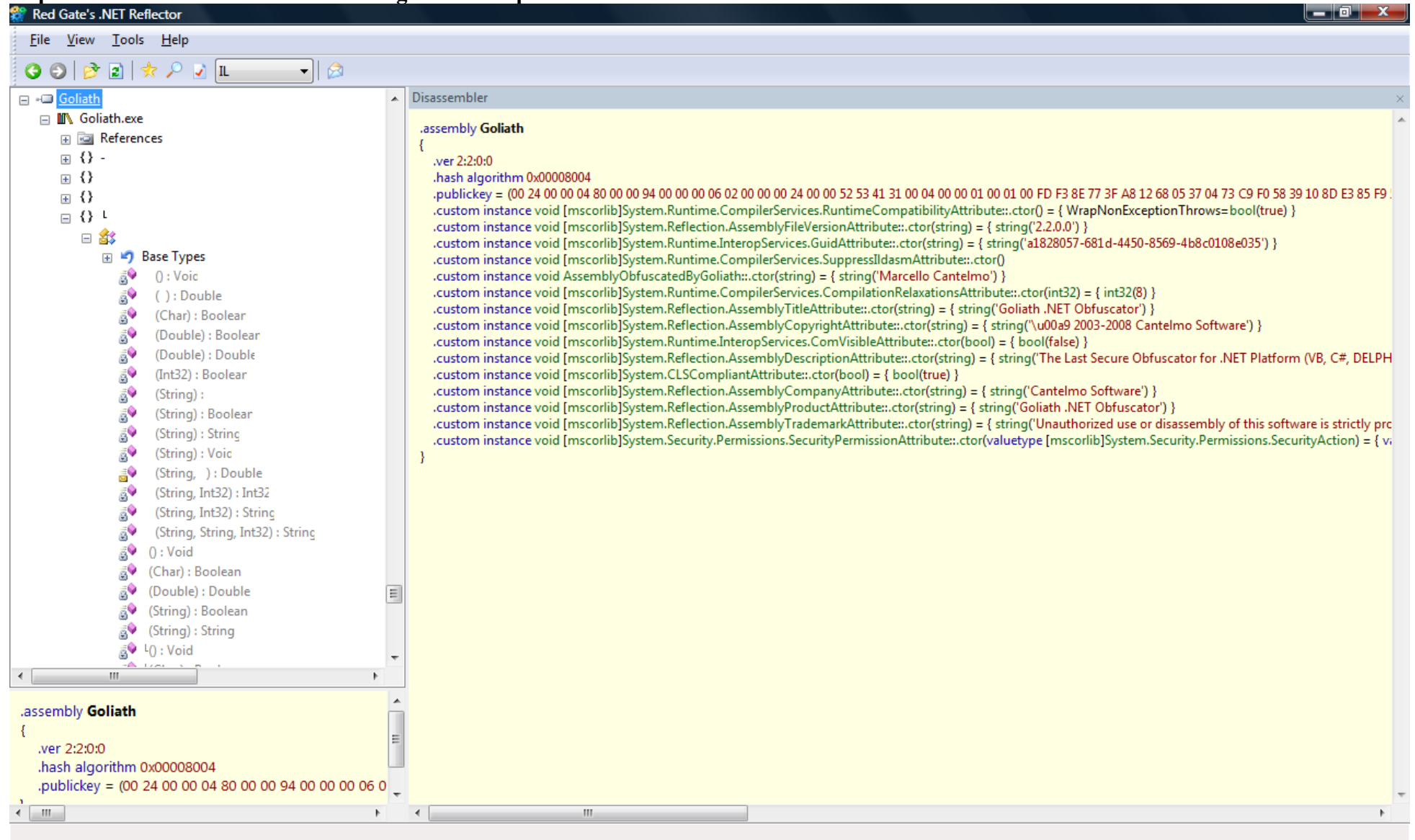
```
.assembly babel
{
  .ver 1:2:0:0
  .hash algorithm 0x00008004
  .publickey = (00 24 00 00 04 80 00 00 94 00 00 00 06 0
```

## Obfuscator Clone: Babel Obfuscator – Ottobre 2008



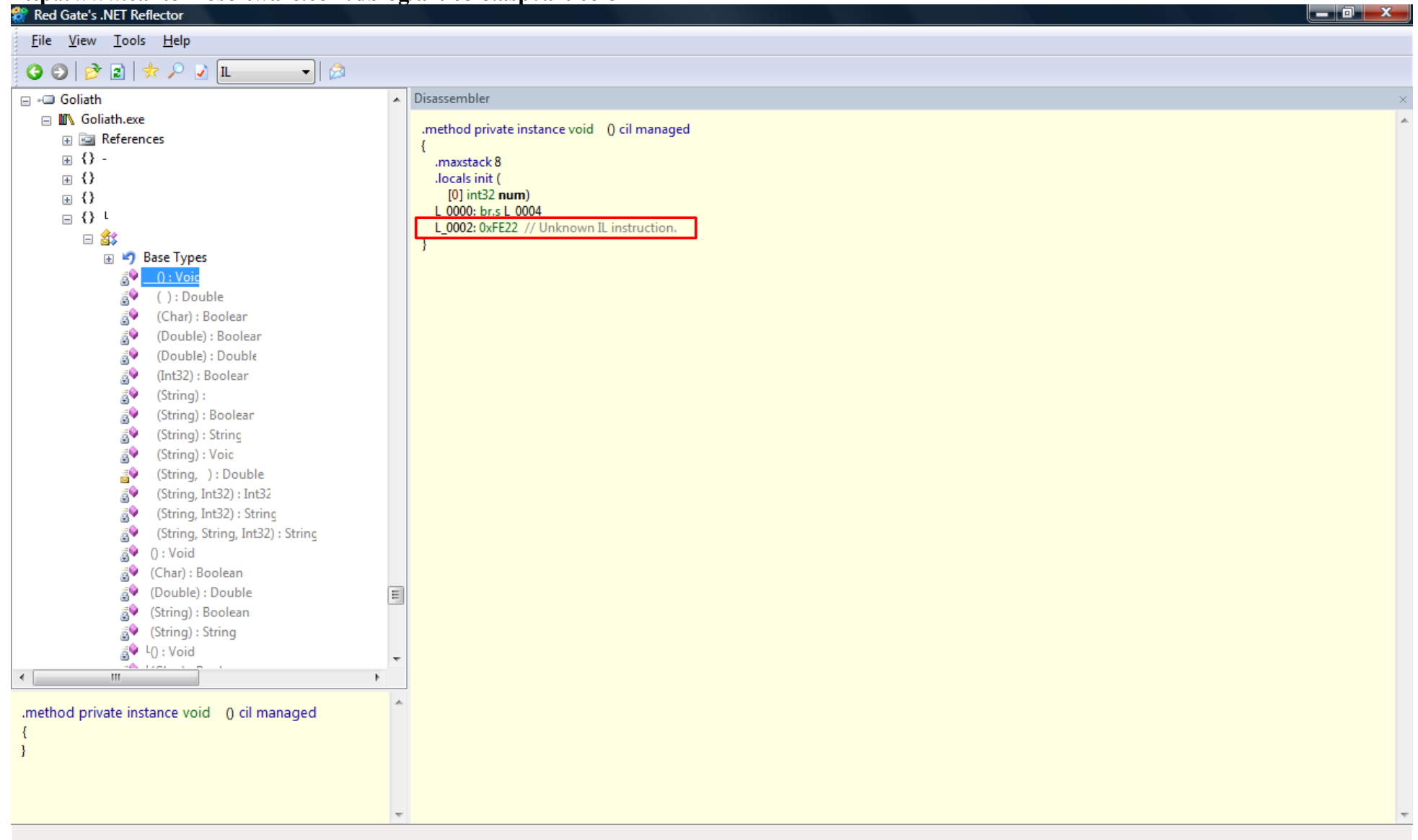
SCHEMA DI PROTEZIONE UNICO e presente SOLO in Goliath .NET Obfuscator sin dalla rel. 2.0.0 (da Gennaio 2006). Necessario per bloccare: Reflector, Reflexil, Dile, Mono-Cecil, Microsoft Phoenix, ect.)

<http://www.cantelmosoftware.com/dblog/articolo.asp?articolo=21>



SCHEMA DI PROTEZIONE UNICO e presente SOLO in Goliath .NET Obfuscator sin dalla rel. 2.0.0 (da Gennaio 2006). Necessario per bloccare: Reflector, Reflexil, Dile, Mono-Cecil, Microsoft Phoenix, ect.)

<http://www.cantelmosoftware.com/dblog/articolo.asp?articolo=21>



la tecnica è stata camuffata dal clone (babel obfuscator) nella successiva release 1.3.0