

نسخه جدید برنامه [Eazfuscator](#) به همراه دو قابلیت جالب یکی کردن و همچنین مدفون نمودن اسمبلی‌ها ارائه شده است:

یکی کردن چند اسمبلی با هم

[Eazfuscator](#) برای یکی کردن اسمبلی‌ها از برنامه معروف [ILMerge](#) استفاده می‌کند با این تفاوت که دیگر نیازی نیست تا پارامترهای آنرا تنظیم کرد و بسیاری از مسایل را به صورت خودکار مدیریت می‌کند. جهت فعال کردن این قابلیت، یکی از روش‌های کار به صورت زیر است: فایلی به نام ObfuscationSettings.cs را به پروژه خود اضافه کرده، سپس محتویات آنرا حذف نموده و با چند سطر زیر جایگزین و کامپایل کنید:

```
using System;
using System.Reflection;

[assembly: Obfuscation(Feature = "merge with file1.dll", Exclude = false)]
[assembly: Obfuscation(Feature = "merge with file2.dll", Exclude = false)]
[assembly: Obfuscation(Feature = "merge with file3.dll", Exclude = false)]
```

همانطور که ملاحظه می‌کنید این چند سطر حاوی نام اسمبلی‌هایی می‌باشند که قرار است با اسمبلی جاری یکی شوند. سپس اسمبلی جاری را (می‌خواهد فایل exe باشد یا یک dll ، فرقی نمی‌کند) بر روی Eazfuscator کشیده و رها کنید. پس از چند لحظه اسمبلی نهایی تولید شده شامل تمام کلاس‌ها و منابع اسمبلی‌هایی خواهد بود که در فایل ObfuscationSettings.cs ذکر شده‌اند؛ به همراه Obfuscation خودکار آن‌ها.

مدفون کردن اسمبلی‌ها در یک اسمبلی

قابلیت دیگر این برنامه دفن (embedding) چند اسمبلی در اسمبلی نهایی است. برای فعال سازی آن روش کار همانند قبل است با این تفاوت که بجای merge with باید نوشت embed . برای مثال:

```
[assembly: Obfuscation(Feature = "embed Common.dll", Exclude = false)]
```

به این ترتیب اسمبلی‌های ذکر شده پس از رمزنگاری و فشرده شدن به صورت منابع اسمبلی جاری ذخیره خواهند شد. مدیریت استفاده از آن‌ها هم خودکار است و نیازی نیست تا کاری در این مورد صورت گیرد.

برای نمونه برنامه معروف LINQPad از همین روش استفاده می‌کند و لازم به ذکر است که ... هنوز که هنوز است هیچ ک.ر.ک. کارسازی برای فعال سازی قسمت intellisense آن که رایگان نیست ارائه نشده و تمام وصله‌های جدید ارائه شده کار نمی‌کنند ...

تفاوت مدفون کردن با یکی کردن چیست؟

در حالت یکی کردن اسمبلی‌ها، سربرار اولیه بارگذاری برنامه همانند روش مدفون سازی وجود ندارد. اما این سربرار آنقدر ناچیز است که کسی آنرا احساس نخواهد کرد. مورد دیگر، عدم پشتیبانی از روش مدفون سازی در سایر سکوها کاری مانند ویندوز فون، Compact Framework و غیره است. اما باید در نظر داشت که برای مثال ILMerge روی اسمبلی‌های دارای XAML کار نمی‌کند (مطابق [مستندات رسمی](#) آن). بنابراین همیشه نمی‌توان از روش یکی سازی استفاده کرد و محدودیت‌های خاص خودش را دارد. در کل روش مدفون سازی به دلیل Obfuscation ، فشرده سازی و رمزنگاری همزمان، امنیت بیشتری را نسبت به حالت Obfuscation تنها ارائه می‌دهد (حداقل شخص "علاقمند" به مطالعه این نوع اسمبلی‌ها باید از چند لایه رد شود و تجربه برنامه

LINQPad ثابت کرده که این روش در مقیاس کلان (در انتظار عمومی هزاران علاقمند) بسیار موفق بوده است).

## نظرات خوانندگان

نویسنده: Mohsen Aghamohammadi

تاریخ: ۲۰:۰۶:۴۶ ۱۳۹۰/۰۱/۱۹

جناب نصیری اطلاعاتی از الگوریتم‌های رمزنگاری اسمبلی‌ها موجود است؟ آیا از روش‌های کلید عمومی و خصوصی با الگوریتم‌های RSA یا MD5 ممکن است استفاده گردد یا صرفاً روش‌ها بازگشت پذیرند؟

نویسنده: وحید نصیری

تاریخ: ۲۰:۲۰:۲۳ ۱۳۹۰/۰۱/۱۹

پاسخ به این سؤال نیاز به مهندسی معکوس دارد ولی عموماً از روش‌های متقارن رمزنگاری استفاده می‌شود.