50 Reasons why to choose Dotfuscator Professional

Dotfuscator Professional protects applications and application data for 5,000+ enterprises across industries, devices, and platforms. In the years following the launch of Visual Studio 2002, over 300,000 developers have come to rely upon Dotfuscator to reduce risks stemming from reverse engineering, tampering, and a growing list of application hacking strategies. We can't list all of the factors that have contributed to making Dotfuscator .NET's most trusted application hardening and shielding solution, but we can list the top 50.

Secure Both Applications and The Data They Process

Dotfuscator's unique suite of technologies and controls secure the value of your development investments and the integrity of sensitive application/user data.

Dotfuscator protects development investments by preventing:

1. **IP Theft:** Securing your intellectual property and your knowhow protects you, your business and your clients.
2. **Piracy:** Mitigating the loss of revenue by significantly increasing the level of effort and detecting when piracy occurs saves money and perhaps your business.
3. **Data Loss:** Attackers who know your code will be that much closer to knowing how to exploit it – don't make it so easy!
4. **Malware:** Directly manipulating (tampering) with your code can be an easy on-ramp for malware unless you have taken steps to prevent it.
5. **Trade Secret Misappropriation:** Generating evidence to ensure protection under trade secret regulations – both inside the US and internationally is a critical first step to ensuring protection under the law.
6. **Regulatory Non-Compliance:** Reverse engineering is a common, well-understood practice and the risks are well-known. If you do not take any steps to mitigate these risks, you may well be seen as an enabler (and liable) versus a victim.
7. **Social Engineering:** When people know your software, they can also pretend to be you – fooling your clients into revealing private information.

Dotfuscator secures sensitive application/user data by preventing unauthorized monitoring and tampering with applications running in production environments including:

8. **Unauthorized Login:** Bypassing authentication logic or modifying results subverts the best-designed identity management practices.
9. **Escalation of Privileges:** Direct manipulation of in-memory data structures and symbols allow unauthorized privilege and identity modification.
10. **Unauthorized Code Execution:** Once a hacker has control over your runtime process, all options are available to them.

11. Access to Encrypted Data and/or Keys: access to data when it is in use (and unencrypted) will defeat the best encryption strategy.

Comprehensive Suite of Protection and Security Technology and Controls

Effective security practices layer well-defined controls and processes to protect (prevent) incidents, detect incidents when prevention falls short, respond (correct or neutralize damage), and/or report occurrences. Dotfuscator offers development organizations precisely these kinds of controls and assurances.

Prevent Reverse Engineering

Under the umbrella term of Application Obfuscation, this collection of .NET assembly transforms impede both machine and human inspection and/or reverse engineering.

12. Patented Renaming Algorithm: Induction overloading takes renaming to a new level of complexity. No other supplier can offer this technique.
13. Cross Assembly Renaming: Eliminates the need to preserve original method interfaces across distributed components without requiring that they be merged first.
14. Control Flow: Our control flow algorithms offer a variety of tunable strategies especially designed to defeat hackers' preferred reverse engineering utilities while minimizing performance impact.
15. String Encryption: Our approach is both secure, high performing and legal for US export.
16. Built-in "Smart" Obfuscation: Leverages knowledge of public framework semantics to maximize automated support for XAML, BAML, XAP, Silverlight and ClickOnce scenarios to name just a few.

Prevent Vulnerability Exploits Such As Unauthorized Access To Functionality and Data, Escalation of Privileges, and/or Execution of Unauthorized Code

Hackers do not limit their probes to reverse engineering; they are as likely to attach a debugger. Using a debugger, hackers can track logic hidden through obfuscation, inspect tokens, keys and strings that have been encrypted, and/or look for vulnerabilities in identity, privilege and/or data management logic.

17. Managed Debugger Detection: Dotfuscator can inject logic uniquely able to detect the presence of an unauthorized managed debugger at startup time or AFTER an application has been started. Dotfuscator also injects default and/or developer-defined preventative actions that execute in real-time – whether or not the application is connected to a network.
18. Native Debugger Detection: Dotfuscator can also inject alternative logic that is uniquely able to detect the presence of an unauthorized native debugger – offering the same levels of protection from this lower-level hacking tool as described above for managed debugger defense.

Detection and Response

Applications under attack need to be able to respond in real-time. Application owners and users are more secure as applications become more "self-aware" and are able to mount more comprehensive responses to hostile acts and environments. Dotfuscator injects logic at build time enabling secured applications to detect and respond to:

19. Assembly Tampering: Applications can be re-signed, authentication failures can be ignored. Anti-tamper ensures that only approved versions of your assemblies can run.
20. Unauthorized Use of a Debugger: Debuggers are not only used to probe for vulnerabilities. They are often used as part of a live attack. The same controls outlined above can play a material role in securing the integrity of both your applications and its data – and by extension – your users' operations.

21. **Attempted Usage After Expiry:** Shelf-life – the ability to trigger special behaviors (including deactivation) after a specified (or relative) time period can help to ensure that trials, test versions, and other "time-boxed" editions of your applications do not live beyond their "use-by" dates.

Mobile-Centric

22. **Xamarin Cross Platform Support:** Dotfuscator can seamlessly insert itself into Xamarin's toolchain and extend all of the obfuscation transforms into Xamarin's resulting native applications targeting Android, iOS, UWP, etc.

Xamarin has allowed .NET developers to unleash their talent targeting native Android apps. While Android offers unprecedented opportunities for application developers, it is also one of the most hazardous runtime platforms. Dotfuscator offers Android-specific controls specially designed to protect Android developers and users utilizing Android-only technologies.

23. **Xamarin.Android Rooted Device Usage:** Android rooted device detection is a constantly shifting problem set and the collection of tests injected by Dotfuscator frees Xamarin developers from having to continuously update their own logic.
24. **Xamarin.Android App Tampering:** Modified Android apps can be used to steal data, hijack your brand, steal online resources, commit fraud, and more.

Contextual Response and Reporting

Dotfuscator includes default responses that developers can utilize without requiring any programming whatsoever. However, complex, mission-critical, or regulated applications can rarely rely upon a generic process exit or exception to effectively and safely respond to tampering or hacking. Dotfuscator allows developers to continue to rely upon Dotfuscator's detection and triggered response framework while allowing developers to add application-specific responses including:

25. **App-Specific Responses:** Execute any custom code such as quarantining an app (as the PCI Council recommends) or preventing an application reboot (but not crashing the app) as some medical device manufacturers prefer, or any other contextual response that helps keep users safe while still protecting code and data.
26. **Log or Portal Integration:** Write to a preferred log or reporting portal to trigger an alert or at least an audit trail of each incident.
27. **Watermarking:** We impose structure on unstructured regions of your assemblies to provide undetectable watermarking allowing you to trace the origin of any specific assembly after it has been found "in the wild."

Established Patterns and Practices

Dotfuscator's features backed by skilled support engineers and a knowledgebase of techniques and samples ensure that you won't need an army of professional consultants to implement a professional and proven application and data security solution. Active Dotfuscator users have access to:

Support Services

Post compile obfuscation and the injection of instrumentation into your precious code is no small task and typically happens at the very end of your production lifecycle – where time and a sense of humor are often in equally short supply. That is why we focus on managing your risk and accelerating your work with a hands-on approach.

28. **Our educated, professional team** knows how to program, they know our technology and their entire job is to make you successful. They will actually chat or speak with you live on the phone.
29. **Access to all updates and new versions**
30. **Enterprise-Ready SLA Capacity:** We have the ability to commit to organization-specific SLA's allowing you to transfer risk from your organization.

Developer Resources

31. PreEmptive Application Protection Implementation Project Plan
32. PreEmptive Protection Dotfuscator Implementation Project Plan for Xamarin
33. Detect and Respond to Rooted Android Devices from Xamarin Apps
34. Secure Data and Apps from Unauthorized Disclosure and Use

Quality, Reliability and Scale

Satisfaction is the Best Indicator of Success

35. **Tested and trusted by the industry's largest and most diverse user community.** A diverse install-base and the industry's largest user community offers both proof and ongoing validation that Dotfuscator meets and exceeds security and risk management requirements including the exacting and ever-expanding quality, reliability and scalability expectations inherent in any serious security solution.

You Can Tell a Lot About a Company and its Products by the Company it Keeps

If you're a .NET developer, you know that Dotfuscator Community Edition (CE) has occupied a unique position since 2003 as the only non-Microsoft component embedded inside Visual Studio. Given that the CE version of Dotfuscator is NOT built for commercial use, why should this unique and special relationship matter to professional developers? Here's why:

36. **Exceptional and Expansive Quality Criteria:** Dotfuscator is subjected to Microsoft regression tests, security audits, code reviews and quality gates – certified by Microsoft – as an organic part of the Visual Studio development and release platform and process.
37. **Upward Compatibility:** Dotfuscator will never slow you down. Dotfuscator offers the broadest .NET framework support, due to Microsoft integration process, Dotfuscator is integrated into every version of Windows and the .NET framework – no matter how early your access to their "bits" or "modern" your practices.
38. **Hardened Against Hundreds of Thousands of Real-World Scenarios:** With an active install-base estimated at over 1,000,000 over the course of a decade, you can be reasonably confident that every obvious (and virtually every obscure) defect has been identified, diagnosed, and addressed.
39. **De Facto Development Standard:** A large, active install-base bolstered by published patterns and practices and tight integration with both Visual Studio means that your investments in Dotfuscator protection and instrumentation will be preserved and ultimately amplified by your core investments in Microsoft development tools and platforms.

Secure DevOps Process Integration

Protection and instrumentation don't occur in a vacuum – they happen inside your development process – a process that is as likely to be purpose-built to your business as it is to change from release to release. That's why we've invested at least as much in providing reliable, flexible and scalable deployment options to accommodate the most agile startup and the most mechanized and automated manufacturing processes.

40. **MSBuild:** We offer a first class MSBuild task.
41. **Command Line:** Of course we can be called from a command line.
42. **Stand-Alone:** Don't have a seat of Visual Studio to spare? Not a problem.
43. **Visual Studio Integration:** Want to preserve a seamless integration with Visual Studio? Again, not a problem.
44. **Xamarin:** Dotfuscator can obfuscate your assemblies which can then be packaged by Xamarin for deployment to devices.

45. **.NET Core Support:** Dotfuscator can harden your Universal Applications.
46. **Incremental Obfuscation:** Releasing patches? Don't want to redistribute your entire application suite? Not a problem.
47. **Cross-Assembly Obfuscation:** Want to extend renaming and other advanced protection capabilities across multiple and distinct binaries? Not a problem.
48. **Distributed Development Support:** Need to do continuous integration or at least distributed across groups? We are built for that.
49. **Delayed Signing Capability:** Need us to sign your app when we're done? It is already built in as an automated feature.
50. **ClickOnce Support:** ClickOnce is complicated to transform, but not for us!
51. **Silverlight and WPF Protection:** Enhanced by advanced capabilities like extending the renaming algorithm into the XAML/BAML resources.
52. **NET Framework Release Targeting:** Working on VS2018 but you need to target an earlier edition? Piece of cake – we will not break your app.
53. **100% .NET Compliance:** Need to PE Verify? Looking to deploy on Mono? It's all good because we break nothing!

Align Application and Data Protection with Recognized Risk Management Frameworks

Effective risk management requires a consistent, balanced, and disciplined approach to assessing, preventing (protecting), detecting, and responding (correcting, neutralizing, and/or reporting) to risks. Application risk management is no exception. Dotfuscator's features backed by PreEmptive Solutions' service, support, and operational stability (vendor viability) can be readily mapped to the risks and controls found inside today's most comprehensive and vetted risk management frameworks including:

54. CIS Controls
55. COBIT
56. ISO/IEC 27000 family - Information security management systems
57. NIST Cybersecurity Framework
58. OWASP

Help to Meet Regulatory and Statutory Obligations

Application development organizations are facing increasing pressure to demonstrate how application lifecycle management practices, functional design, and quality controls are proactively contributing (meeting the requirements) of regulations and statutes. Dotfuscator features, backed by PreEmptive Solutions help users to meet these kinds of requirements including those included in:

59. GDPR
60. HIPAA
61. PCI Data Security Standard
62. Defend Trade Secrets Act
63. **Consultative Support:** Have specific questions on how others are using our technology to help meet their regulatory and compliance obligations? Call our support and they will arrange for a conference call.

The PreEmptive Difference

Over the past 19 years, our team has focused on protecting, monitoring and measuring application value. With over 5,000 clients and software on millions of developer desktops, there can be no question that organizations that are serious about building and securing the best software possible trust PreEmptive Solutions. Here are a few final reasons why.

64. Category Creators: We don't just lead in the obfuscation and application risk management – we literally created them.
65. Proven Quality and Service: The largest manufacturers, life science companies, aerospace corporations, financial institutions, and (of course) software development organizations all trust PreEmptive Solutions to secure and measure their work without compromising the quality and functionality of their code.
66. Cross Platform: From Microsoft Azure to Universal to Xamarin to WPF and Silverlight to the next "big thing" – we support these frameworks, platforms and surfaces on the day they arrive (and for as long as Microsoft supports them).
67. Beyond .NET: developing in Java? Android? iOS? PreEmptive Solutions provides a cross-platform solution to secure and manage all of your application development investments.
68. Exceed Expectations: Just as we delivered 68 reasons to work with Dotfuscator when we only committed to 50 – we will exceed your expectations for quality, responsiveness and capability.

PreEmptive Solutions
Visit us at www.preemptive.com

Worldwide Headquarters
767 Beta Drive, Suite A
Mayfield Village, OH 44143
Phone +1 440.443.7200
solutions@preemptive.com

European Headquarters
140 bis rue de Rennes
75006 Paris, France
Phone +33 01.83.64.34.74
eurosolutions@preemptive.com

Find us: