SafeNet | SENTINEL SOFTWARE MONETIZATION SOLUTIONS

Sentinel Envelope

White Papaper

目次

はじめに

ソフトウェアの不正使用は深刻化しています。その発生は広範囲に及び、追跡が難しく、防止・禁止はますます困難になっています。ソフトウェアの不正使用は潜在的な収益の損失であるとともに、正規ソフトウェアを購入する顧客にとっては、不正使用による損失コストも結果的に負担させられるという悪影響を及ぼします。

現代のコンピューティング環境においてソフトウェア ベンダが直面する最大の課題の一つは、ソフトウェアの正規品を購入して使用する顧客に余計な手間をかけさせることなく、いかにソフトウェアの不正使用を防ぐかということです。

「ソフトウェアの不正使用」は「ソフトウェアの著作権侵害」とも呼ばれ、増加の一途をたどっています。インターネット上に溢れるリバースエンジニアリング情報によって誰もが情報を簡単に入手して知識を得ることができることが原因です。ほとんどの国にはソフトウェアに適用される著作権法があるとはいえ、その順守とコンプライアンスの程度は千差万別で、著作権侵害が多発している国があることは周知の事実です。

ソフトウェアの不正使用は深刻化しています。その発生は広範囲に及び、追跡が難しく、防止・禁止はますます困難になっています。ソフトウェアの不正使用は潜在的な収益の損失であるとともに、正規ソフトウェアを購入する顧客にとっては、不正使用による損失コストも結果的に負担させられるという悪影響を及ぼします。ソフトウェアの著作権保護に積極的に取り組むソフトウェア ベンダは間違っていませんが、ソフトウェアのハッキングによるセキュリティ侵害は増加し続けており、完全に防止することは不可能かもしれません。ソフトウェア ベンダはライセンス管理ソリューション ベンダやハードウェア プロテクション製品メーカーと協力してセキュリティの水準を絶えず更新・強化することが不可欠です。革新的なプロテクション／セキュリティ手法を製品のライフサイクルの中に組み込むことで、潜在的な脅威を未然に防止できます。

本書では、ソフトウェアの著作権侵害と知的財産の盗難を防ぐために Sentinel Envelope に搭載されているさまざまな機能について解説します。

概要

Sentinel Envelope は、暗号化とネイティブ コード難読化を組み合わせて史上最強のプロテクションを提供し、貴重な知的財産を守ります。

ラッパー ー定義と使用法

「ラッパー」は、その名前が示すとおり、実行可能ファイルに変更を加えて同等の新しいファイルを作成するツールです。ファイルの圧縮のため、またはリバースエンジニアリングに対するプロテクション手法として用いられます。その仕組みはロシアのマトリョーシカという人形と同じで、元の人形の中に人形が入っていて、その中にまた人形が入っているという入れ子構造に基づきます。ラッピング プロセスでは、元の実行可能ファイルに 1 つ（またはそれ以上）のセクションを追加し、通常の実行前にプログラムをラッピング解除するためのローダを付加します。ローダはオペレーティングシステムとして機能し、アプリケーションをロードしながらアンチデバッグ、トレース検出、ライセンス管理、バックグラウンド チェックなどのリアルタイム タスクを実行する役割を担います。

Sentinel Envelope ー ワンクリックで簡単に使えるソリューション

Sentinel Envelope は、ファイル暗号化、コード難読化、およびシステムレベルのアンチデバッギングによってソフトウェアのリバースエンジニアリングを防止する自動ファイル ラッパーです。実行可能ファイルと DLL のラッピング プロセスにより、アルゴリズム、企業秘密、専門的なノウハウをハッカーから確実に守ります。

Sentinel Envelope は、アプリケーションをハードウェアまたはソフトウェアベースのプロテクション キーにバインドする役割を果たすプロテクション シールドを追加することにより、アプリケーションをプロテクトします。

デフォルトのプロテクション方式が選択されている場合、Sentinel Envelope によるプロテクション プロセスは数秒で完了します。一部またはすべてのオプションを使用するためにデフォルト以外の方法を用いる場合は、処理時間が多少長くなります。Sentinel Envelope はアプリケーションのソースコードへのアクセス権限を持たないソフトウェアベンダに対し、極めて強力なプラットホームを提供します。たとえば再販業者や卸売業者は、プロテクトされていないソフトウェアを現地市場向けにプロテクトして販売したい場

合に、Sentinel Envelope のデフォルトの基本設定を使用して簡単にすばやくプロテクトできます。

プロテクトされたアプリケーションの起動時に、Sentinel Envelope のローダ（Envelope ランタイム）はプロテクション キーにクエリを送信してその存在を確認します。有効な Sentinel プロテクション キーが存在する場合、ローダはプロテクション キーの暗号化エンジンと連携して、開発者が以前に暗号化したアプリケーション ファイルを復号化します。

Sentinel プロテクション キーが存在していないか、または無効である場合、アプリケーションは停止されて、動作しなくなります。また、Sentinel プロテクション キーが使用可能になっていない場合、アプリケーションのバイナリ ファイルは復号化されません。

データ ファイル暗号化ユーティリティ

アプリケーションの実行可能ファイルをプロテクトする機能に加えて、アプリケーションがアクセスするデータ ファイルを暗号化することにより、知的財産を確実に守ることができます。そのためには、ハッカーとアプリケーションの間に追加のセキュリティ レイヤを配置します。データ ファイル暗号化ユーティリティ「DataHASP」は、Sentinel Envelope と連携してデータファイルを暗号化します。DataHASP がデータファイルを事前に暗号化した後、プロテクトされたアプリケーションがデータ ファイルを暗号化／復号化します。データ ファイル暗号化プロセスの後、適切なプロテクション キーが検出された場合にのみ、データ ファイルへのアクセスが許可されます。

機能

Sentinel Envelope の機能と特長

·自動ファイル ラッパー ーファイル暗号化とコード難読化により、ソフトウェアのリバースエンジニアリングを強力に防御

·アプリケーションとハードウェアとのバインド ー プロテクションキーによってアプリケーションをハードウェアに強固にバインド

·セキュアな通信チャネル ー プロテクトされたアプリケーションとプロテクション キー間の通信をセキュア チャネルで保護することにより中間者攻撃を防止。Java エンベロープはこの機能を使用して、ハッカーによる通信の盗聴およびプロテクションキーから返されるデータへのアクセスを防止

· ランタイム復号化 ー すべてのクラス ファイルを同時に仮想マシンにロードするのではなく、ランタイムにリクエストされた時点でファイルを復号化。ハッカーによるアプリケーション全体の再構築を防止

Sentinel Envelope は、ファイル暗号化、コード難読化、システムレベルのアンチデバッギング、ホワイトボックス暗号化、AppOnChip など極めて先進的な機能により、ソフトウェアの知的財産（IP）をリバースエンジニアリングから強力に保護します。これらの機能による強固なセキュリティをハッカーが破るには多大な時間と手間を要するため、アプリケーション コードが盗まれる危険はなくなります。Sentinel Envelope の各機能は、ハッカーがアプリケーションへの不正侵入を試みる際に用いるさまざまなテクニックを考慮して設計されています。

アンチデバッギング／アンチトレーシング手法

通常、デバッガーはソフトウェア開発者がアプリケーションの開発段階でバグを検出して問題をトレースするために使用されます。しかし、アプリケーションに不正侵入しようとするハッカーは、埋め込まれたプロテクションコードの検出とトレースのために開発者と同じデバッガーを使用します。ハッカーの最終目的はプロテクション コードの改変、無効化、または完全な削除です。

Sentinel Envelope の極めて強力な機能の 1 つである「デバッガー検出メカニズム」は、アクティブなデバッガーを常時検知する機能です。Sentinel Envelope は、疑わしいコマンドと間違った情報をアクティブなデバッガーに送信することにより、デバッガーの重要ジョブの実行を阻止します。Sentinel Envelope によるデバッガーの識別は簡単で、正当なデバッガーと不正なデバッガーを容易に見分けることができます。また、Sentinel Envelope はアンチトレーシング ツールが起動されているかどうかを検知し、プロテクトされたアプリケーションの実行を必要に応じて中止します。

ハッカーとソフトウェア開発者は同じデバッガーを使用するので、Sentinel Envelope は正当な開発者によるデバッグとハッカーによるハッキングを区別できなければなりません。そのため、Sentinel Envelope はデバッガーの検出時にメッセージを表示し、プロテクトされたアプリケーションのロードを中止します。正当な開発者はその段階でデバッガーを無効にすれば、アプリケーションを正常にロードして実行できます。しかし、アプリケーション

のロード／実行後にデバッガーが起動された場合、それは明らかにアプリケーションへの不正アクセスを試みる著作権侵害行為であるため、アプリケーションは停止します。

Sentinel Envelope では、ISV がアプリケーションの一部またはアプリケーション ファイル全体を暗号化できるようにすることで、個々のニーズに基づいたさまざまなセキュリティ機能の構成が可能です。

最大の弱点を守る

ラッピング メカニズムによってプロテクトされたアプリケーションの最大の弱点は、アプリケーション ファイルと追加されたプロテクション コードとの継ぎ目です。この継ぎ目が破られると、ライセンスを含むプロテクション キーへのリンクが切断されて、アプリケーションは完全に無防備な状態になります。そのため、多くのハッカーはこの継ぎ目に攻撃を仕掛けようとします。ハッカーはプロテクトされたファイルを研究してプロテクション コードを解析し、装着されるプロテクション キーへのリンクを分析します。プロテクション コードを解読してリンクの場所がわかったら、次のいずれかの方法で攻撃を仕掛けることができます。

- アプリケーション固有のクラッキング ― 特定のアプリケーション ファイルのプロテクション リンクを破ります。
- 包括的クラッキング ― 同じメカニズムによってプロテクトされたファイルに同一のメソッドが繰り返し出現する場合に、それらすべてのプロテクション リンクを破ります。

継ぎ目は最大の弱点

したがって、プロテクトされるアプリケーション ファイルと追加されるプロテクション コードとの継ぎ目を曖昧にしてトレース不可能にし、プロテクション メカニズムを解読しようとするハッカーに膨大な時間と手間をかけさせることが不可欠です。Sentinel Envelope の特長の 1 つは、継ぎ目をプロテクトして障害物を多数配置し、プロテクション リンクが破られない

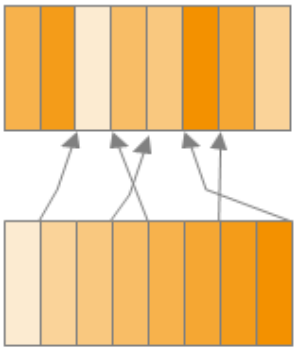
ようにすることです。そのために、プロテクション プロセスの実行中、アプリケーションにマルチレイヤ構造のプロテクション コードを追加します。プロテクション コードの各レイヤは電車の車両のように連結されるコードの断片です。各プロテクション セッションにおいて、Sentinel Envelope はプロテクション コードを構成する個々のレイヤを元のアプリケーション ファイルに追加する際、配置の順序を変化させます。

Sentinel Envelope の各プロテクション セッションでレイヤの配置はダイナミックに変化するため、プロテクトされた各ファイルはユニークになります。完全に同じファイルをプロテクトしても、プロテクト後のファイルに類似点は存在しません。Sentinel Envelope のプロテクション コードの最後の命令からアプリケーション コードの最初の命令への移行部分は、プロテクトされるアプリケーションごとに異なります。元のコードの開始位置は各アプリケーションで異なるため、Sentinel Envelope のプロテクション コードとアプリケーション ファイルとの継ぎ目をトレースすることはほとんど不可能です。ハッカーが個々のレイヤを研究し、プロテクトされるファイル内での配置がわかったとしても、同じファイルが Sentinel Envelope の別のセッションでプロテクトされると無意味です。プロテクションをさらに強固にするために、Sentinel Envelope はレイヤの配置を変化させるだけでなく、プロテクトするファイルごとにレイヤの数を変化させます。また、各レイヤを暗号化する方法を変化させます。アプリケーションの実行中、各レイヤはランダムな暗号鍵を用いて、その次のレイヤを復号化する役割を担います。

さらに、各レイヤのコードはダミーOP コードを使用することで難読化されます。ダミーOP コードは有効な命令コードと命令コードの間に挿入されます。そのため、コードの解読は極めて困難となり、逆アセンブラによるプロテクション メカニズムの解析やコードの逆アセンブルは役に立ちません。

Sentinel Envelope は、ソース コードをラッピングすることにより、リバースエンジニアリングを強力に防御し、貴重なアルゴリズムや営業秘密を守ります。Sentinel Envelope によってプロテクトされた各ファイルはそれぞれ異なるランダム シードを用いて暗号化されるので、元のファイルが同じであっても、プロテクション プロセスを経てまったく異なるファイルになります。アプリケーション ファイルは複数のブロックに分割されます。ブロ

ックはスケーラブルで、プロテクション時に開発者が事前に決定できます。各ブロックは AES 暗号アルゴリズムによって、別々のシードを用いて暗号化されます。



Sentinel Envelope のプロテクションコード

クラッキング検出時の挙動の変化

デバッガーをブロックするために Sentinel Envelope で用いられるもう一つの手法は「挙動の変化」です。オペレーティング システムとデバッガーではアプリケーションの実行の仕方が異なり、ライセンスを含むプロテクション キーにはその違いを利用する精巧なコード設計が用いられています。(整合性チェックなどにより) クラッキングの試みが検出された場合、ソフトウェアの反応は遅くなり、その結果「原因」と「結果」の論理的関係が絶たれます。反応の遅れは、クラッキングの試みと特定のクラッキングの試みに対するソフトウェアの否定的反応との真の論理的関係を隠し、ハッカーを混乱させます。クラッキングの試みが検出された場合にソフトウェアが機能不全になるなどの挙動は、非常に効果的です。

プライベート API

ソフトウェア プロテクション ソリューションを展開するセキュリティ ベンダの多くは、すべての顧客に対して同じ API ライブラリを提供しています。この API ライブラリは、セキュリティ侵害の発生時に単一障害点となります。SafeNet は、はるかにセキュアなソリューションとして ISV 固有の

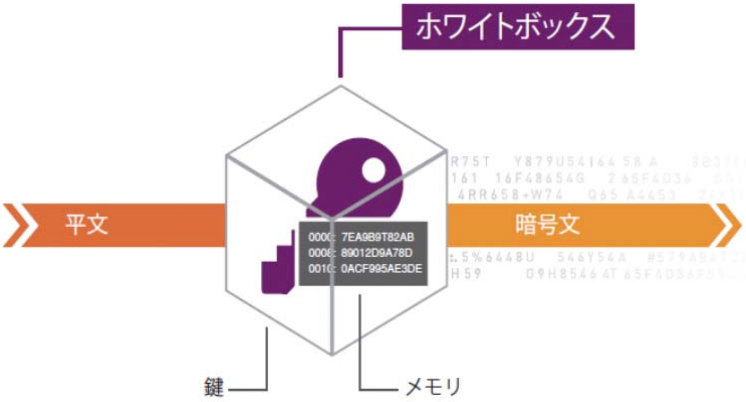
API ライブラリを提供しています。ISV 固有のプライベート API は SafeNet のサーバ上でビルドおよびカスタマイズされるので、ハッカーの攻撃の手は及びません。これらの API によって、ISV 各社は構造的に異なるコンポーネントを自社のアプリケーションの中に組み込むことができます。プライベート API は各 ISV でカスタマイズの仕方が異なり、アプリケーションへの組み込みプロセスの一環としてユニークなホワイトボックス暗号化方式によって補強され、最後に厳重な難読化とプロテクション技術によって守られます。生成されるライブラリは一般的なクラッキングをブロックします。ハッカーが 1 つの ISV の API ライブラリのセキュリティを破ったとしても、その方法を他の ISV に対して活かすことはできません。Sentinel Envelope は、開発者のコンピュータ上にダウンロードされた API のコピーからプライベート API を捕捉し、プロテクション時にアプリケーションに挿入します。強力にプロテクトされた ISV 固有の API が Sentinel Envelope ランタイムによって使用され、プロテクトされたアプリケーションへの正当なアクセスのみが許可されます。

Sentinel プロテクション キーの定期的な呼び出し

Sentinel Envelope の利点の 1 つは、コンパイル済みファイルに適用できることです。アプリケーションのソース コードに変更を加える必要はまったくありません。プロテクション キーの呼び出しは、アプリケーション ファイルに追加されたプロテクション コード（Sentinel Envelope ランタイム）によって定期的に実行されます。Sentinel プロテクション キーの存在は暗号化手法を用いて確認されますが、ISV の開発者は、Sentinel プロテクション キーがチェックされる時間間隔を設定できます。これは、プロテクション時に ISV が全面的に設定できる多くのパラメータの 1 つにすぎません。

ホワイトボックス暗号化

SafeNet は業界で初めて、プロテクトされたアプリケーションとプロテクション キーとの通信をハッカーが盗聴することを防ぐ手段として「ホワイトボックス暗号化」を採用し、通信チャネルを完全に暗号化するソリューションを発表しました。ホワイトボックスベースのセキュアな通信チャネルは、ISV 固有のコンポーネントを活用し、プロテクトされたバイナリ ファイルから暗号鍵が動的・静的を問わず抽出されることを防ぎます。



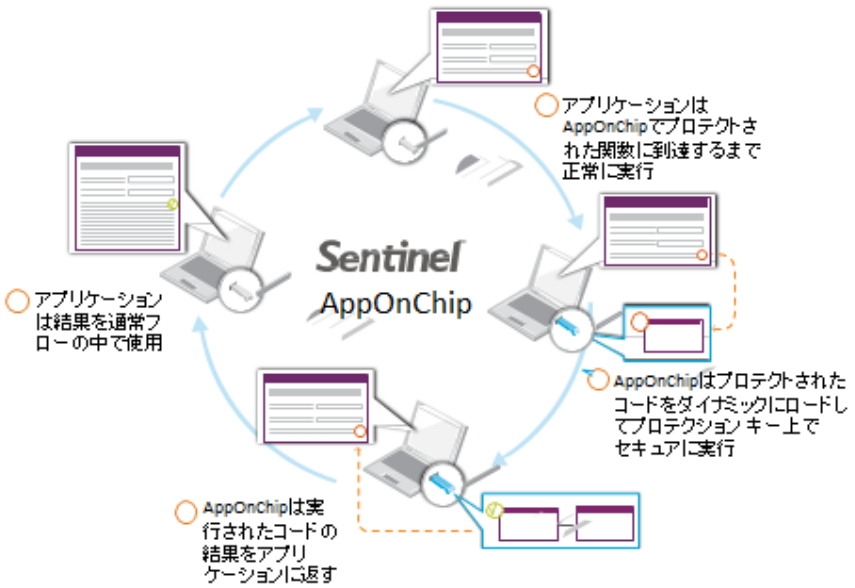
Sentinel Envelope のホワイトボックス暗号化の特徴は下記のとおりです。

- 暗号鍵は決してメモリに存在せず抽出不可
- ランダムなセッションＩＤによってセッション間で情報伝達が異なることを保証
- 一方向だけに機能 (暗号化のみもしくは、解読のみ)
- 色々な開発者間において高い多様性

Sentinel によってソリューションをゼロから構築する時間と手間をかけずに一流のセキュリティを簡単に適用できるため、貴社の技術部門は本来の業務に専念できます。

AppOnChip

Sentinel Envelope のセキュリティ強化機能の 1 つに「AppOnChip」があります。これは、Sentinel ハードウェア キーとアプリケーションとのバインドを切断不可にする機能で、ISV にとって極めてセキュアなソフトウェア プロテクション ソリューションです。この機能は完全に自動化されており、AppOnChip 機能と互換性のあるコード ブロックが含まれたアプリケーションの関数リストを ISV に提示します。暗号化と署名によってプロテクトされたコード ブロックは Sentinel ハードウェア キー自体にロードして実行できます。このセキュリティ強化手法により、Sentinel Envelope は業界で最もセキュアなソフトウェア ライセンシング ソリューションとして知られています。AppOnChip の特長は、セキュリティの強固さ、実装の簡単さ、ライセンシングの自由度の高さ、エンドユーザに透過的な処理、自動化されたオペレーションなどです。



オリジナル エントリ ポイント（OEP）プロテクション

オリジナル エントリ ポイント（OEP）とは、アプリケーションの開始点のアドレスです。このアドレスからオペレーティング システム（OS）はアプリケーションの実行を開始します。プロテクトされたアプリケーションをハッカーがラッピング解除するためには、このアドレスを見つけてプロテクトコードを削除し、アプリケーションのオリジナル エントリ ポイントからアプリケーションの起動を試みなければなりません。

Sentinel Envelope は、他のラッパーとは異なり、オリジナル エントリ ポイント命令をデフォルトの場所から削除して、Sentinel Envelope のランタイム コード内部に断片（チャンク）を分散させます。分散されたチャンクからハッカーがオリジナル エントリ ポイントを探して再構築しようとしても、チャンクの場所とサイズがランダムなため、それは実質的に不可能です。

メソッドレベルのプロテクション

Sentinel Envelope は、メソッドレベルのプロテクションを定義することにより、.NET と Java の実行可能ファイルのプロテクションを強化します。.NET または Java アセンブリがプロテクト対象として選択されると、Sentinel Envelope は個々のプロテクションに使用できるメソッドを自動で判別します。それにより、知的財産のセキュリティがさらに強化され、個々のアプリケーションに最適のプロテクションが適用されます。

インポート アドレス テーブルの削除

クラッキングを防御するためのその他の手段として、「インポート アドレス テーブルの削除」というプロセスがあります。インポート アドレス テーブルには、プロテクトされるアプリケーションが使用する外部 DLL の関数のアドレスが格納されています。元のアプリケーションのラッピング プロセスにより、インポート アドレス テーブルが削除され、ディスクまたはメモリから消失して Sentinel Envelope のプロテクション コード内部に情報が分散されます。つまり、各インポート アドレス オペレーションは Sentinel Envelope によってプロテクトされて内部で扱われます。また、各インポー

ト アドレス オペレーションはそれぞれ異なる難読化コードによって別々のメモリ アドレスに分散されるので、ハッカーはパズルのピースを集めるように各インポート アドレス オペレーションを個別に解析・解読しなければなりません。従来のプロテクション ツールの場合、ハッカーはインポート アドレス テーブルを利用して、各エントリの解析が終了する時間を算出します。Sentinel Envelope ではインポート アドレス テーブルは使用されないため、ハッカーはクラッキングに要する時間を算出できず、作業を完了できるかどうか確信が持てません。その上、Sentinel Envelope はさまざまな手法によってインポートを隠します。そのため、クラッキングされたアプリケーションでは後で障害が発生し、結局は完全に使用不能となります。

元のアプリケーション コードと Sentinel Envelope のプロテクション コードの強固なバインド

一般的なラッパーには、元のアプリケーション コードとプロテクション コードとのバインド機能はありません。Sentinel Envelope はラッパーとプロテクトされたアプリケーションとの事実上のバインドを強固にします。Sentinel Envelope は、プロテクション時に行われるコード フロー解析に基づき、アプリケーション フローの中にプロテクション コードを組み込みます。プロテクション コードはそれとはわからない形でアプリケーションの中に組み込まれるので、ハッカーは削除できません。ランタイムで制御フローが指定のアドレスに到達すると、元のアプリケーションのコードが継続的に実行されている間、明示的な実行シーケンスがさまざまな検証・確認オペレーションを実行します。制御フローが改変されていない場合、アプリケーションは動作します。アプリケーションの整合性が疑わしい場合は、プロセスは停止します。

Stolen Bytes

メモリのスナップショットとダンピングの手法は広く用いられていますが、状況によってはハッカーに元のアプリケーションのソース コードを盗み見るチャンスを与えかねません。これはハッカーがプロテクトされたアプリケーションをクラッキングする際の最初のステップとなるため、アンチハッキング ソリューションは特にその防御に優れている必要があります。

「Stolen Bytes」の概念は、プロテクトされたアプリケーションと Sentinel Envelope のプロテクション コードとの依存関係を強化することです。元のソース コードのさまざまな位置からランダム バイトのチャンクを選択して Sentinel Envelope のプロテクション コードの内部にランダムに分散させます。プロテクトされた元のアプリケーション コードの実行中、これらのコードのチャンク（「Stolen Bytes」）はランダムな新しい位置で実行されます。この仕組みは、元のアプリケーション コードの終了点と Sentinel Envelope のプロテクション コードの開始点の継ぎ目を曖昧にすることにより、元のアプリケーション コードと Sentinel Envelope のプロテクション コードとの依存関係を強化します。

コードとシンボルの難読化

難読化は、意味のある文字列をアルファベットまたは数字で構成されるランダムな文字列に変換するプロセスです。Sentinel Envelope を使用することにより、アンチ リバースエンジニアリングのためのセキュリティ手法である難読化技術を適用できます。デフォルトでは、プロテクション プロセスの一環として、プロテクトされる.NET アセンブルの中のすべてのシンボル名が難読化されます。さらに、選択されたメソッドのコード全体を難読化することもできます。コード難読化はアプリケーションの処理速度を低下させることがあるため、デフォルトではコード全体の難読化は有効になりません。プロテクト対象のメソッドのリストで選択されていないメソッドに対しても、コード難読化を適用できます。

まとめ

Sentinel Envelope では、ISV がアプリケーションの一部またはアプリケーション ファイル全体を暗号化できるようにすることで、個々のニーズに基づいたさまざまなセキュリティ機能の構成が可能です。セキュリティ対策には一定のコストがかかります。必然の結果として、常に完璧なセキュリティを備えることは不可能です。そのため、アプリケーション自体に求められるセキュリティ レベルを適切に評価することが極めて重要です。つまり、守る必要があるものの価値と、潜在的なリスクを考慮しないことで後々生じる損失の大きさを併せて評価する必要があります。

ハッカーによる営業秘密やノウハウの盗難を積極的に防止することにより、ISV は産業スパイを確実に防御し、競争上の優位性を高めることができます。Sentinel Envelope は、暗号化とネイティブ コード難読化を組み合わせて史上最強のプロテクションを提供し、貴重な知的財産を守ります。さらに、ソリューションをゼロから構築する時間と手間をかけずに一流のセキュリティを簡単に適用できるため、貴社の技術部門は本来の業務に専念できます。

SafeNet Sentinel ソフトウェア収益化ソリューション

SafeNet は、世界中のソフトウェアベンダとテクノロジーベンダに、革新的で信頼できるソフトウェアライセンシングおよびエンタイトルメント管理ソリューションを 25 年以上も提供してきました。

統合しやすく使いやすい、革新的な機能重視の Sentinel® ソフトウェア収益化ソリューションは、規模や技術要件、組織構造を問わず、どんな組織に対しても固有のライセンス有効化、施行、管理要件を満たすように設計されています。全体の収益性を向上させ、社内業務を改善し、競争力を維持し、顧客やエンドユーザとの関係を深めつつ、著作権侵害対策、IP 保護、ライセンス有効化、ライセンス管理の課題、あらゆるソフトウェア収益向上に関する課題に SafeNet はお客さまとともに取組んでいます。SafeNet には、進化し続けるマーケットに対応するため、新たな要件に適応し新たなテクノロジーを取り入れてきた実績があります。世界中の 25,000 以上のお客様が、Sentinel を選択することが、今日、明日、そしてその先のビジネスのやり方を発展させていく自由を手に入れることだと考えています。

Sentinel Envelope を含む Sentinel LDK 評価キットは、下記の URL からダウンロードの依頼が可能です。

http://www.safenet-inc.com/sentinel-dk-download/

Sentinel LDK Envelop technical Guide

2014 年 4 月発行

日本セーフネット株式会社

http://jp.safenet-inc.com