様　式　Ｃ－１９、Ｆ－１９－１、Ｚ－１９　（共通）

科学研究費助成事業　　研究成果報告書

令和　２ 年　 ５ 月 １９ 日現在

機関番号：３４３０４
研究種目：基盤研究(C)（一般）
研究期間：2017～2019
課題番号：１７Ｋ００１９６

研究課題名（和文）名前難読化の評価指標の確立

研究課題名（英文）Robustness and efficiency metrics of the name obfuscation methods

研究代表者
玉田　春昭（TAMADA, Haruaki）

京都産業大学・情報理工学部・准教授

研究者番号：３０４５７１３９

交付決定額（研究期間全体）：（直接経費）　　3,500,000円

研究成果の概要（和文）：本研究は，難読化の中でよく使われている名前難読化を対象にその性能を評価した．評価のため，命令列と引数の型をもとに元のメソッドの動詞の推薦を行なった．その結果，約40%の動詞が復元できた．この数値は完全な形ではないにせよ難読化により隠した情報が暴露する可能性を表し，名前難読化に一定の脆弱性があることを示せたと言える．
この実現にはバイナリソフトウェアの解析が必要となる．難読化されたソフトウェアのソースコードは公開されないためである．そこでこの技術を応用し，バイナリを対象にしたソフトウェア同士の比較の規模拡大にも取り組んだ．その結果，閾値が0.2の時，従来の40%程度の時間で比較できるようになった．

研究成果の学術的意義や社会的意義
難読化手法の中でも名前難読化手法は，非常によく使われる手法でありながら評価の難しさから評価されてこなかった．暗号分野に倣い難読化手法も，多くの研究者・開発者により攻撃されることにより，堅牢性を評価する必要がある．本研究は，名前難読化の評価を行う初めての試みである．本研究により，脆弱な難読化手法が淘汰され，難読化手法の世代交代が進むことが期待できる．そして，本研究の成果である評価手法により，難読化ツール同士で性能の比較が行えるようになり，ツール選定の基準が生まれる．加えて，プログラム中の名前の良し悪しに関する議論が深まり，名前に基づいたプログラムの評価も可能になろう．

研究成果の概要（英文）：The purpose of this work is to evaluate the tolerance of the identifier renaming obfuscation (IRM), which is the most popular obfuscation method. For this, we proposed the recommendation method for the verbs of the methods. The experimental results showed that our method could restore about 40% verbs of methods. The result is not high; however, it indicates the IRM has some vulnerabilities since 40% of methods has the possibility of exposure to their behaviors by the proposed method.
To achieve the above, we need the analysis techniques of binary software because the source codes of obfuscated software do not open. Then, we applied the techniques for scaling up the birthmarking method, which is calculating similarities between the binary software. As a result, the method successfully reduces to 40% comparing the time of the conventional method, when the threshold was 0.2.

研究分野：ソフトウェア工学

キーワード：名前難読化　逆変換　ソフトウェアバースマーク

様　式　Ｃ－１９、Ｆ－１９－１、Ｚ－１９（共通）

１．研究開始当初の背景

ソフトウェア内部の秘密情報を保護するため，難読化手法が用いられる場合がある．難読化とは入出力の仕様を保ったまま，理解が困難になるようプログラムを変換する技術である．プログラム中で秘匿したい情報を保護するために用いられている．例えば，Blue-ray などでは，暗号化されたディスクを再生するために再生機器に復号鍵が埋め込まれている．復号処理中にメモリ上に復号鍵が現れるとそこから復号鍵が盗まれる恐れがある．これを防止，もしくは困難化するために難読化手法が用いられている．難読化はプログラム中のデータフローやコントロールフロー，レイアウトなど着目する情報や，どのように保護するかにより様々な手法が提案されている．

世の中にはいくつかの難読化ツールがリリースされている．それら難読化ツールは，それぞれの特色に従って複数の難読化手法が併用されている．しかし，難読化による保護度合いを測定できる機構が存在しないため，どの難読化ツールを選べば良いかの判断は容易ではない．加えて，何らかの基準により特定の難読化ツールを選んだとしても，そのツールにより難読化されたプログラムがどの程度堅牢であるかの測定も容易ではない．攻撃方法が整理されておらず，定量的な評価指標が確立されていないためである．本研究課題ではその中でも特にシンボル名を意味のない名前に変更する名前難読化（IRM; Identifier Renaming Method）に着目する．世の中の大半の難読化ツールが異なる実現方法で実装し，採用している手法であるにもかかわらず，評価指標が存在しないためである．逆に言えば，この IRM の評価指標が確立できれば，難読化ツールで共通の手法が比較できるようになり，難読化ツール同士の比較が行えるようになる．一方で，もし IRM が脆弱な難読化手法であった場合，危険な手法が勘違いされたまま使われていることになる．そのため，IRM の有効性と堅牢性の評価指標の策定は急務であると言える．

２．研究の目的

本研究の最終的な目標である IRM の有効性･堅牢性評価のために次の 2 つの小目的を設定した．(a)攻撃方法の整理および IRM への攻撃，そして(b)バイナリレベルのプログラムの比較方法のスケールアップである．(a)に上げたように難読化手法も，暗号手法と同様に様々な攻撃に対する耐性を評価する必要がある．そこで，難読化された名前を，元の名前に復元することを考える．前提として，難読化されたプログラムのソースコードが入手できることは考えにくい．そのため，バイナリを対象に復元を試みる必要がある．

一方，バイナリであっても同一のプログラムが公開されていれば，どのように名前が難読化されていたとしても難読化が無効化できる．つまり，バイナリレベルでのプログラムの比較ができれば，難読化対策に有効であろう．そこで，(b)研究代表者が従来から取り組んでいるバイナリレベルでの比較手法であるソフトウェアバースマーク技術のスケールアップについても取り組む．バースマークの本来の目的は大規模なソフトウェア群を対象に，自分の持つオリジナルなソフトウェアと似たものがないかを見つけるものである．しかし，大規模なソフトウェア群といっても高々数万程度である．世の中のすべてのプログラムを調べることはできないまでも，Maven Central Repository や npm など何らかの基準で集められたリポジトリ内のプログラムを調べることは重要である．そのために，バースマーク技術の手順を見直し，より大規模なソフトウェア群に対してバースマークの適用を狙う．

３．研究の方法

(a)IRM への攻撃では，プログラム中のシンボル名の復元を試みる．その復元には何らかの情報が必要であるものの，難読化で敏感に変化するような情報では高性能な復元は期待できない．また，プログラム中のシンボル名は動詞や名詞に目的語などの複数の単語で構成される場合が多い．目的語はプログラムのドメインに依存する場合が多く復元が難しい．そこで，メソッド名の動詞の復元を試みる．メソッドには難読化の影響を受けにくい命令列が含まれており，これを復元の手掛かりとする．そして，世の中には数多くのプログラムが OSS として公開されている．そこでこれらの情報をあらかじめ集めデータベース化しておく．そして，命令列と引数の型情報を復元情報として，機械学習（ランダムフォレスト）を用いメソッド名を復元する．
(b)ソフトウェアバースマーク技術のスケールアップでは，従来のバースマークの手順を整理し，収集，抽出，比較，判断の 4 つの段階の比較の前に，絞り込み段階を導入する．世の中に膨大な数のソフトウェアはあるが，そのほとんどは盗用ではない．そのため，明らかに盗用ではない対象を除外し，続く比較段階で俎上に載せるソフトウェア数を減らすことを目指す．そこで絞り込み段階では，全文検索エンジンを用いることで，バースマークのラフで高速な比較を行う．

４．研究成果

(a)では，Maven Central Repository から集めたプログラムから命令列の 2-gram から頻度ベクトルを構築し，データベースに格納した．また，引数の型，戻り値の型のうち名前難読化の影響を受けないシステム定義の名前からも頻度ベクトルを構築し，同様にデータベースに格納した．そして，データベースに格納された情報からランダムフォレストを用いて復元モデルを構築した．復元対象のプログラムからも同じように命令列の 2-gram と型情報の頻度ベクトルを構築し，復元モデルに適用し，候補となる動詞を導出した．結果として，全体の 31.62%の動詞の復元に成功した．また，動詞の意味的な類似度に基づいた評価では，同義語では 33.94%，上位語の関係では 40.07%のメソッド名の動詞を復元できた．
この結果は決して高いものではない．加えて，復元できるものもメソッド名の動詞に限定されている．しかしそれでも名前難読化の保護が十分ではない可能性を示した点において意味があると考えられる．先にも述べたように，名前難読化は実装が簡単なことから多くの難読化ツールに採用されている．しかしこの結果が示しているように，40%程度は動詞を復元できる可能性を示した．つまり，名前難読化単体の保護では破られる可能性があることを示している．そのため，難読化を適用するときには，少なくとも各メソッドの命令列が大きく変化していることを確認する必要があると言える．この手法は命令列が大きく変化しないことを前提としているためである．
続いて(b)では全文検索エンジン Apache Solr を採用し，バースマーク絞り込みシステム Mituba を構築し，盗用の疑いのあるプログラムを見つける実験を執り行った．閾値（この値より類似度が大きい場合，盗用であると判断される）が 0.2 のとき，所要時間は従来の 40%以下に抑えられ，80%以上のプログラムが無関係と判定された．残ったプログラムのうち，誤検出は 90%程度と非常に高いものの，検出漏れは 0%であり，精度も 70%以上となっている．また難読化などによりプログラムが変更された場合の耐性評価においても，一番強力な難読化が施された場合であっても 80%以上のプログラムを見つけ出せた．これらの結果を元に最適な閾値を議論した結果，標準的には 0.6 程度の閾値が最適であるが，ユーザの問題設定によっては，閾値が 0.2 でも本手法は

有効であることを示した．

上記手法は，静的解析で得られる静的バースマークに限定される．動的解析で得られる動的バースマークは抽出の自動化が困難であるため，上記手法は適用できない．ただし，出自が明らかな原告ソフトウェア（例えば OSS のソフトウェア）に限れば，単体テストコードが入手できる．単体テストコードが入手できれば，単体テストを実行することで，プログラムを実行できる．つまり，アスペクト指向と組み合わせることで動的解析の自動化が実現でき，動的バースマークの抽出の自動化が可能となる．実際にこの方法で OSS から動的バースマークを抽出し，従来手法で抽出した動的バースマークとの比較を行った．比較結果は異なるソフトウェア同士の比較では最大 0.039，平均 0.013，同一ソフトウェアの異なるバージョン同士では最大 0.629，平均 0.235 といずれも低い類似度であった（実験 2）．ただし，異なるソフトウェア同士の類似度は同一ソフトウェアの結果結果に比べ，より顕著に低いためこの手法は有効であると考える．そして，この手法の攻撃耐性を評価するため，ProGuard と yGuard を用いてソフトウェアを難読化した後，この手法を適用した．その結果，実験 2 の結果との相関係数は 0.834 (ProGuard)，0.999 (yGuard) と非常に強い結果を示した($p < 0.0001$)．つまり，この手法は少なくとも ProGuard，yGuard に対して耐性があると言える．

５．主な発表論文等

〔雑誌論文〕　計4件（うち査読付論文　4件／うち国際共著　0件／うちオープンアクセス　0件）

１．著者名 磯部陽介，玉田春昭	４．巻 60
２．論文標題 ランダムフォレストを用いた名前難読化の耐タンパ化性能の評価	５．発行年 2018年
３．雑誌名 情報処理学会論文誌	６．最初と最後の頁 1063-1074
掲載論文のDOI（デジタルオブジェクト識別子） なし	査読の有無 有
オープンアクセス オープンアクセスではない、又はオープンアクセスが困難	国際共著 －

１．著者名 Takehiro Tsuzaki, Teruaki Yamamoto, Haruaki Tamada, and Akito Monden	４．巻 5
２．論文標題 Scaling Up Software Birthmarks Using Fuzzy Hashing	５．発行年 2017年
３．雑誌名 International Journal of Software Innovation (IJSI)	６．最初と最後の頁 89-102
掲載論文のDOI（デジタルオブジェクト識別子） 10.4018/IJSI.2017070107	査読の有無 有
オープンアクセス オープンアクセスではない、又はオープンアクセスが困難	国際共著 －

１．著者名 横井 昂典，玉田 春昭	４．巻 60
２．論文標題 単体テストコードとアスペクト指向を用いた動的バースマークの抽出コストの削減	５．発行年 2019年
３．雑誌名 情報処理学会論文誌	６．最初と最後の頁 1247-1259
掲載論文のDOI（デジタルオブジェクト識別子） なし	査読の有無 有
オープンアクセス オープンアクセスではない、又はオープンアクセスが困難	国際共著 －

１．著者名 中村 潤，玉田 春昭	４．巻 61
２．論文標題 大量のソフトウェアを対象にしたソフトウェアバースマークによる盗用検出 ―全文検索システムを用いた検査対象の絞り込み手法	５．発行年 2020年
３．雑誌名 情報処理学会論文誌	６．最初と最後の頁 454-473
掲載論文のDOI（デジタルオブジェクト識別子） なし	査読の有無 有
オープンアクセス オープンアクセスではない、又はオープンアクセスが困難	国際共著 －

〔学会発表〕　計11件（うち招待講演　1件／うち国際学会　3件）

1．発表者名
玉田 春昭, 神崎 雄一郎

2．発表標題
オペコードの編集距離を用いたJVM向け難読化ツールの難読化性能の評価

3．学会等名
2019年暗号と情報セキュリティシンポジウム予稿集（SCIS 2019）

4．発表年
2019年

1．発表者名
Tanaknori Yokoi, and Haruaki Tamada

2．発表標題
A Beforehand Extraction Method for Dynamic Software Birthmarks using Unit Test Codes

3．学会等名
19th ACIS International Conference on Software Engineering, Artificial Intelligence, Networking and Parallel/Distributed Computing (SNPD 2018)（国際学会）

4．発表年
2018年

1．発表者名
Yosuke Isobe, and Haruaki Tamada

2．発表標題
Are Identifier Renaming Methods Secure? --An Evaluation Focuses on Opcodes using Random Forest--

3．学会等名
19th ACIS International Conference on Software Engineering, Artificial Intelligence, Networking and Parallel/Distributed Computing (SNPD 2018)（国際学会）

4．発表年
2018年

1．発表者名
西 陽太, 神崎 雄一郎, 門田 暁人, 玉田 春昭

2．発表標題
難読化されたJavaバイトコードに対するシンボリック実行攻撃の困難さ評価の検討

3．学会等名
第25回ソフトウェア工学の基礎ワークショップ（FOSE2018）

4．発表年
2018年

１．発表者名
横井 昂典, 玉田 春昭

２．発表標題
単体テストコードを利用した動的バースマークの抽出

３．学会等名
コンピュータセキュリティシンポジウム2017（CSS 2017）

４．発表年
2017年

１．発表者名
磯部 陽介, 玉田 春昭

２．発表標題
ランダムフォレストによる名前難読化の逆変換

３．学会等名
第24回ソフトウェア工学の基礎ワークショップ（FOSE2017）

４．発表年
2017年

１．発表者名
中村 潤, 玉田 春昭

２．発表標題
検索エンジンを用いたソフトウェアバースマークによる検査対象の絞り込み手法

３．学会等名
第24回ソフトウェア工学の基礎ワークショップ（FOSE2017）

４．発表年
2017年

１．発表者名
Jun Nakamura, Haruaki Tamada

２．発表標題
mituba: Scaling up Software Theft Detection with the Search Engine

３．学会等名
International Conference on Software Engineering and Information Management (ICSIM 2018)（国際学会）

４．発表年
2018年

１．発表者名
大槻 成輝，玉田 春昭，神崎 雄一郎

２．発表標題
JVM環境におけるオペコード列と名前に着目した適用難読化ツールの特定

３．学会等名
2020年暗号と情報セキュリティシンポジウム予稿集（SCIS 2020）

４．発表年
2020年

１．発表者名
玉田 春昭，神崎 雄一郎

２．発表標題
Javaバイトコードを対象とした命令の頻度解析による適用難読化ツールの特定

３．学会等名
コンピュータセキュリティシンポジウム2019予稿集（CSS 2019）

４．発表年
2019年

１．発表者名
磯部 陽介，玉田 春昭

２．発表標題
ランダムフォレストを用いた名前難読化の耐タンパ化性能の評価

３．学会等名
ソフトウェアシンポジウム2019（招待講演）

４．発表年
2019年

〔図書〕　計0件

〔産業財産権〕

〔その他〕

―

６．研究組織

	氏名 （ローマ字氏名） （研究者番号）	所属研究機関・部局・職 （機関番号）	備考