1L-03

保護コードの自然さに着目した命令カムフラージュ

永井 晃人 † 神崎 雄一郎 ‡ 門田 暁人 ††
† 熊本高等専門学校 電子情報システム工学専攻 ‡ 熊本高等専門学校 人間情報システム工学科
†† 奈良先端科学技術大学院大学 情報科学研究科

1 はじめに

悪意を持った攻撃者からソフトウェア内の秘密情報を保護する一方法として，命令のカムフラージュ法 [1] が提案されている．この方法は，攻撃者に知られたくない命令を別の命令で偽装することで，ソフトウェアの解析を困難にする．偽装に用いる命令 (偽装命令) の決定は保護方法の使用者に委ねられているが，カムフラージュされたコード (プログラム断片) が「不自然」，すなわち，コンパイラが出力したコードとしてもっともらしくない状態となった場合，偽装した事実が攻撃者に発見されやすくなり，結果として保護の強さの低下を招く可能性がある．

そこで本論文では，偽装命令の前後のコードを考慮して，命令を「自然」に偽装する命令のカムフラージュ法を提案する．提案方法では，文献 [2] で提案されているコードの不自然さメトリクスを用いて，偽装命令とその周辺の命令から構成されるコードの不自然さを評価し，その不自然さが最小となるように偽装命令を決定する．自然な命令でカムフラージュされたコードは，カムフラージュされていないコードとの区別がつきにくく，攻撃者による偽装箇所の発見が困難になると考える．

2 コードの不自然さ

コード (命令列) の不自然さを，文献 [2] と同様，次のように定義する．まず，N-gram モデルによって与えられたコードを構成する命令列 $i_1^n = i_1 i_2 \ldots i_n$ の生成確率 $P(i_1^n)$ を，次式のように近似する．

$$P(i_1^n) \approx \prod_{k=1}^{n} P(i_k \mid i_{k-N+1}^{k-1}) \tag{1}$$

すなわち，k 番目命令 i_k の生成確率が，直前の (N-1) 命令 $i_{k-N+1} \ldots i_{k-2} i_{k-1}$ にのみ依存すると考える．また，命令列 i_1^n で構成されるコード C の不自然さ $A(C)$ を，生成確率 $P(i_1^n)$ を用いて，次のように定義する．

$$Artificiality \;\; A(C) = -\log_{10} P(i_1^n) \tag{2}$$

$A(C)$ の値が大きいほど，アセンブリ言語としてもっともらしくなく，C は不自然であると考える．本論文では，この不自然さの値が低いことを「自然」と呼ぶ．

3 自然な偽装命令の決定手順

命令のカムフラージュ法 [1] は，プログラム中の偽装対象命令 I を異なる偽装命令で偽装 (上書き) し，自己書換え機構を用いたルーチンにより，実行時のある期間のみ元来の命令に復元する．

ここでは任意のコード C の偽装対象命令 I について，自然な偽装命令を決定する手順を，図 1 に沿って説明する．ここで C は，偽装対象命令 I，および，I の周辺命令列 (I の前後 m 命令，m は自然数) から構成されるものとする．まず，偽装命令の候補の集合 D に含まれる各アセンブリ命令で I を偽装し，$C_1, C_2, \ldots, C_n$ を得る．次に，$C_1, C_2, \ldots, C_n$ に対してそれぞれ不自然さを測定し，$A(C_1), A(C_2), \ldots, A(C_n)$ を得る (不自然さの値は，N-gram の分割単位や，コーパスの内容によって変化する)．得られた不自然さのうち，値が最小となったコードに用いていた偽装命令を，自然な偽装命令として決定する．なお，偽装後の命令列がコーパスに存在しなかった場合は，偽装命令の候補から除外する．

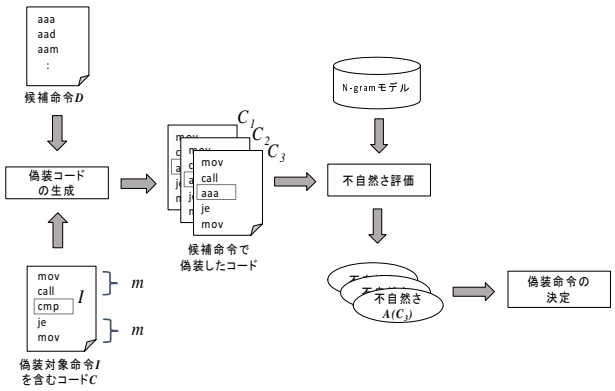

図 1 自然な偽装命令の決定手順

Instruction Camouflage Method Focusing on the Artificiality of Protected Code
Akihito Nagai†, Yuichiro Kanzaki‡, Akito Monden††
†Advanced Course of Electronics and Information System Engineering, Kumamoto National College of Technology
‡Department of Human-Oriented Information Systems Engineering, Kumamoto National College of Technology
††Graduate School of Information Science, Nara Institute of Science and Technology

表1　決定した自然な偽装命令とその周辺命令列

	オリジナル	1-gram	2-gram	3-gram
コード	mov	mov	mov	mov
	call	call	call	call
	cmp	mov	test	test
	je	je	je	je
	call	call	call	call
不自然さ		0.35	0.32	0.79

表2　自然な偽装命令とその不自然さ

	1-gram	2-gram	3-gram
1	mov 0.35	test 0.32	test 0.79
2	call 1.12	mov 0.48	cmp 1.19
3	lea 1.32	cmp 0.66	mov 2.96
4	cmp 1.40	lea 2.74	inc 3.34
5	je 1.42	repz 2.96	sub 4.36
6	test 1.44	add 3.40	dec 4.44

4　ケーススタディ

4.1　概要

秘密情報をもつプログラムに対して提案方法を適用し，カムフラージュされたコードの不自然さについて考察するケーススタディを行った．ケーススタディで用いる N-gram モデルは，文献 [2] と同じコーパスを用いて生成した．また，偽装の候補の集合 D としてはコーパスに出現する全命令を使用した．適用対象のコード C は「mov call cmp je call」とし，cmp 命令を偽装対象命令 I とした．このコードは，文献 [3] に例示されている DRM のプログラムのメディアを再生するモジュールに含まれるものである．なお，不自然さを測定する際の N-gram の分割単位は，1-gram，2-gram，3-gram と変化させた．

4.2　結果

提案方法によって決定された偽装命令とその周辺コードを表 1 に示す．前後の命令を考慮しない 1-gram の場合は，コーパス中の出現頻度が最も高い mov に決定された一方，2-gram および 3-gram の場合は，test に決定された．

表 2 は，不自然さが低かった自然な偽装命令とその周辺コードの不自然さを，上位 6 つまで示したものである．どれにおいても，一般的なプログラムにおいてよく見られる mov 命令，test 命令などの不自然さが低くなっている．

4.3　議論

1-gram では，周辺命令列は考慮されず，単純にコーパス中の出現頻度が高い命令ほど自然な命令と判定される．しかしながら，1-gram では自然なカムフラージュを行うには必ずしも十分ではないと考えられる．たとえば，1-gram において 2 番目に自然な命令と判定された call を偽装命令とすると，「call call je」という不自然な命令列が生じてしまう．一方，2-gram および 3-gram においては call 命令は自然な偽装命令として上位には出現していない．また，2-gram および 3-gram において最も自然な偽装命令と判定された test は，後続の条件付きジャンプ命令 je を考慮しても自然な命令であると考えられる．

今回のケーススタディから，1 命令単位で考えた場合に出現頻度が高い命令でも，周辺の命令を考慮したときには，不自然な命令になる可能性があることがわかった．一方，2-gram，3-gram といった周辺の命令を考慮して判定した偽装命令は，自然な，保護されていない命令との区別がつきにくい命令となり得ることがわかった．

5　おわりに

本論文では，保護コードの自然さに着目した命令カムフラージュ方法を提案した．今後の課題としては，偽装対象命令のオペコードだけでなくオペランドを考慮して自然な命令を決定できるよう改良することを考えている．また，偽装対象命令を複数命令で偽装する場合の自然な偽装命令列の決定や，自己書換えを行うルーチンの自然な挿入位置の決定ができるよう提案方法を拡張する予定である．

参考文献

[1] 神崎雄一郎，門田暁人，中村匡秀，松本健一：命令のカムフラージュによるソフトウェア保護方法，電子情報通信学会論文誌，Vol.J87-A, No.6, pp.755-767 (2004).

[2] 神崎雄一郎，尾上栄浩，門田暁人：コードの「不自然さ」に基づくソフトウェア保護機構のステルシネス評価，情報処理学会論文誌，Vol.55, No.2 pp.1005-1015 (2014).

[3] Collberg, C. and Nagra, J.: *Surreptitious Software: Obfuscation, Watermarking, and Tamperproofing for Program Protection*, Addison-Wesley Professional (2009).