L-030

64-bit ARM 環境における権限の変更に着目した 権限昇格攻撃防止手法の評価
Evaluation of Privilege Escalation Attack Prevention Method by Focusing on Privilege Changes on 64-bit ARM

吉谷 亮汰　　山内 利宏
Ryota Yoshitani　Toshihiro Yamauchi

1. はじめに

オペレーティングシステム（以降，OS）は，計算機が動作するための基盤として高い信頼性が求められる一方で，毎年数多くの脆弱性が報告されている．OS のコード量は膨大であり，すべての脆弱性を取り除くことは困難である．また，脆弱性が見つかった場合であっても，システムの運用形態やデバイスの特性によっては，脆弱性の修正パッチの適用が困難である可能性がある．

OS の脆弱性を悪用する攻撃の 1 つに権限昇格攻撃 (privilege escalation) がある．権限昇格攻撃が成功した場合，攻撃者は，本来与えられている権限より高い権限でシステムを操作できるようになり，情報漏えいやサービスの妨害などの被害を受ける可能性がある．特に，攻撃者に管理者権限が奪取された場合，システム全体のセキュリティが脅かされることになる．

Linux カーネルの脆弱性を悪用する権限昇格攻撃の対策として，文献[1]では，システムコールによるプロセスの権限の変更に着目し，システムコール処理の前後における権限の変更内容を監視する権限昇格攻撃防止手法が提案されている（以降，従来手法）．従来手法はシステムコールハンドラに変更を加えることで，システムコール処理の前後をフックし，権限に関する情報（以降，権限情報）の変更内容を監視する．

一方で，システムコールハンドラはアーキテクチャに依存しており，従来手法は x86-64 環境でのみ実現している．モバイル端末や IoT (Internet of Things) 機器では，消費電力を抑える目的から ARM アーキテクチャが多く採用されている．このため，ARM アーキテクチャにおける権限昇格攻撃への対策は重要である．

この課題に対処するため，我々は，従来手法を拡張し，64-bit ARM 環境において同様の監視機構を実現する手法を提案した（以降，提案手法）[2]．本稿では，提案手法の有用性を示すために，提案手法を導入した 64-bit ARM 環境において権限昇格攻撃を実施し，提案手法が権限昇格攻撃を検知できるか否かを評価した結果について報告する．

2. 権限の変更に着目した権限昇格攻撃防止手法の ARM への拡張

2.1 考え方

システムコールの発行直後に実行されるシステムコールハンドラはアーキテクチャに依存している．このため，64-bit ARM アーキテクチャ用のシステムコールハンドラである SVC (Supervisor Call) ハンドラ内で行われるシステムコールサービスルーチン呼び出しについて，**図 1** の処理流れで権限情報の変更内容を監視する．以降では，Linux 4.4.0 (64bit) を例に，提案手法[2]の処理流れについて述べる．

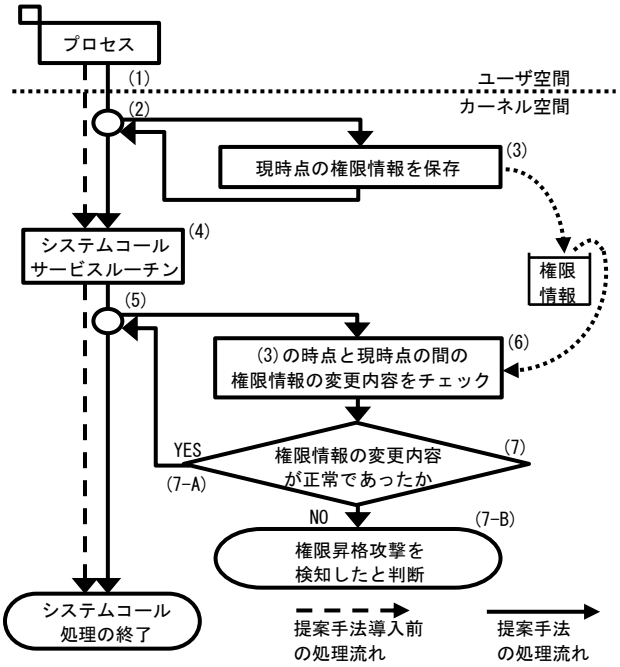

図 1　提案手法の処理流れ

2.2 提案手法の処理流れ

提案手法は，システムコールによる権限情報の変更内容を監視し，正常でない変更を検知することで権限昇格攻撃を防止する．正常な変更とは，それぞれのシステムコール処理において，そのシステムコールが変更し得る権限情報のみが変更されることである．

提案手法の処理の流れを図 1 に示し，以下で説明する．

(1) プロセスがユーザ空間からシステムコールを発行し，カーネル空間へ処理を移行
(2) システムコールサービスルーチンへの移行をフックし，提案手法の処理へ移行
(3) 現時点（システムコール処理前）の権限情報を保存
(4) システムコールサービスルーチンの実行
(5) システムコールサービスルーチンの実行の直後に処理をフックし，提案手法の処理へ移行
(6) (3) で保存したシステムコール処理前の権限情報と現時点の権限情報の差分（システムコール処理による権限の変更内容）をチェック

岡山大学大学院自然科学研究科, Graduate School of Natural Science and Technology, Okayama University

```
[  123.685169] AKO: detected unauthorized change of uid. syscall=285 original: uid=1000, euid=1000,
fsuid=1000, suid=1000 attempt: uid=0, euid=0, fsuid=0, suid=0
[  123.687199] AKO: detected unauthorized change of gid. syscall=285 original: gid=1000, egid=1000,
fsgid=1000, sgid=1000 attempt: gid=0, egid=0, fsgid=0, sgid=0
[  123.688463] AKO: detected unauthorized change of capability. syscall=285 original: inh[0]=0 inh[1]=0
per[0]=0 per[1]=0 eff[0]=0 eff[1]0 amb[0]=0 amb[1]=0 attempt: inh[0]=0 inh[1]=0 per[0]=4294967295
per[1]=63 eff[0]=4294967295 eff[1]63 amb[0]=0 amb[1]=0
```

図 2　権限昇格攻撃を検知した際のログ

(7) システムコール処理による権限の変更内容が正常なものであったかを確認
  (A) 権限の変更内容が正常であった場合，権限昇格攻撃は行われていないと判断し，元々の処理流れに戻り，システムコール処理を終了
  (B) 権限の変更内容が正常でなかった場合，権限昇格攻撃が行われたと判断．また，攻撃を検知したことを示すログを出力

64-bit ARM 環境の場合，プロセスは SVC 命令を使用してカーネル空間へ処理を移行し，アセンブラで記述された SVC ハンドラを呼び出す．SVC ハンドラでは，発行されたシステムコールに対応したシステムコールサービスルーチンが実行される．そこで，(2) と (5) では，提案手法は SVC ハンドラに変更を加えることで，システムコールサービスルーチン呼び出しの前後において処理をフックする．

3. 評価

3.1 評価内容と評価環境

提案手法を導入した 64-bit ARM 環境において，権限昇格攻撃を実施し，提案手法が権限昇格攻撃を検知できるか否かを評価した．提案手法が想定する権限昇格攻撃では，本来権限情報を変更し得ないシステムコールの処理中に権限情報が変更される．評価には，自作したシステムコールを発行するプログラムを利用した．このシステムコールは，プロセスに root 権限を与えるコードを実行し，不正な権限昇格を行う．

評価は，提案手法を導入した Linux 4.4.0 (64-bit) が動作する 64-bit ARM 仮想計算機上で実施した．仮想化ソフトウェアには QEMU 2.5.0 を使用し，CPU アーキテクチャは ARM Cortex-A57，コア数は 1，メモリは 1024MB を指定している．

3.2 評価結果と考察

図 2 の評価において，提案手法が不正な権限昇格を検知し出力したログを示す．提案手法は，システムコール処理中の不正な権限情報の変更を検知した際の権限情報について，その内容，システムコールの番号，およびシステムコール処理前後における値をログとして出力する．図 2 のログの先頭の文字列 “AKO” は，ログが提案手法によって出力されたものであることを示す．また，文字列 “detected unauthorized change of (uid | gid | capability)” は，権限情報であるユーザ識別子 (UID)，グループ識別子 (GID)，および特定の処理の実行をプロセスに許可するか否かを示すケーパビリティセットのそれぞれについて，不正な変更があったことを示している．したがって，提案手法が不正な権限昇格の検知に成功していることが分かる．

OS の脆弱性を悪用して権限昇格攻撃を行う場合，カーネル空間から攻撃者の用意したユーザ空間のコードを実行する方法がある．keyctl システムコールの脆弱性 CVE-2016-0728 を悪用した攻撃[3]では，カーネル内の関数ポインタをユーザ空間のコードを指すように改ざんする．また，他の攻撃方法として，futex システムコールの脆弱性 CVE-2014-3153 を悪用した攻撃[4]のように，ユーザ空間とカーネル空間の境界アドレスを改ざんすることで，ユーザ空間のコードからカーネル空間へのアクセスを許可する方法がある．提案手法は，システムコール処理中に監視対象の権限情報を変更する攻撃であれば，脆弱性の種類にかかわらず検知可能である．

一方で，提案手法は，システムコール処理を介さない権限情報の改ざんを検知できない．また，提案手法が権限情報として監視対象としている UID，GID，ケーパビリティセット，およびユーザ空間とカーネル空間の境界アドレス以外のデータを改ざんすることで権限昇格が可能な場合，提案手法はこの攻撃を検知できない．

4. おわりに

64-bit ARM 環境におけるシステムコールの権限の変更に着目した権限昇格攻撃防止手法の有用性を示すために，不正な権限昇格を行うシステムコールを利用した攻撃を実施し，提案手法が攻撃を検知できるか否かを評価した結果について報告した．評価の結果から，提案手法はシステムコールの処理中に不正な権限昇格を行う攻撃を検知可能であることを示した．残された課題として，報告されている OS の脆弱性を悪用した権限昇格攻撃の検知実験がある．

謝辞　本研究の一部は，JSPS 科研費 JP19H04109 の助成を受けたものです．

参考文献

[1] 赤尾洋平，山内利宏：システムコール処理による権限の変化に着目した権限昇格攻撃の防止手法，情報処理学会シンポジウムシリーズ コンピュータセキュリティシンポジウム 2016 (CSS2016) 論文集，vol.2016，no.2，pp.542-549 (2016) .
[2] 吉谷亮汰，山内利宏：権限の変更に着目した権限昇格攻撃防止手法の ARM への拡張，情報処理学会研究報告，Vol.2018-CSEC-82，No.29，pp.1−7 (2018).
[3] PerceptionPointTeam: cve_2016_0728 exploit, GitHub Gist, available from ⟨https://gist.github.com/PerceptionPointTeam/18b1e86d1c0f8531ff8f⟩ (accessed 2019-06-18).
[4] ExploitDatabase: Linux Kernel 3.14.5 (CentOS 7/ RHEL) - 'libfutex' Local Privilege Escalation, available from ⟨https://www.exploit-db.com/exploits/35370⟩ (accessed 2019-06-18).