L-029

KVM 上のゲスト OS 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 Focusing on Privilege Changes in Guest OS on KVM

福本 淳文　　**山内 利宏**
Akifumi Fukumoto　　Toshihiro Yamauchi

1. はじめに

オペレーティングシステム（以降，OS）の脆弱性は数多く報告されている．また，OS のソースコード量は膨大であるため，脆弱性を完全に取り除くことは困難である．OS の脆弱性を悪用した攻撃の 1 つに権限昇格攻撃が存在する．この攻撃では，OS の脆弱性を悪用するプログラムを実行することで，プロセスの権限をより高い権限へと昇格させる．権限昇格攻撃が成功すると，攻撃者は本来与えられる権限よりも，高い権限でシステムを操作できる．特に，攻撃者に管理者権限を奪取されると，システム全体のセキュリティが脅かされる可能性がある．このため，権限昇格攻撃に対処することは重要である．

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，システムコール処理の前後に権限の変更内容を監視することで権限昇格攻撃を防止する手法 [1]（以降，従来手法）が提案されている．従来手法は，システムコールサービスルーチンの前後において，プロセスの権限に関する情報（以降，権限情報）のうち，そのシステムコールが変更しない権限が変更された場合，権限昇格攻撃が実行されたと判断し，攻撃を防止する．

しかし，従来手法を導入するためには，Linux カーネルのソースコードを変更する必要がある．このため，カーネルを再構築ができない状況では，従来手法をシステムに導入することができない．また，従来手法は，権限情報の変更を検証するために，システムコール処理前に権限情報をカーネルスタックに保存する．このため，攻撃者はカーネルスタックに保存された権限情報を改ざんすることで，権限の変更の検証をバイパスできる．

これらの課題に対処するために，我々は仮想マシンモニタである KVM 内に従来手法と同様のセキュリティ機構を実現する手法 [2] を提案した．本稿では提案手法の有用性を示すために，Linux カーネルの脆弱性を悪用した権限昇格攻撃を用いて，提案手法を評価した結果について述べる．

2. KVM 上のゲスト OS における権限の変更に着目した権限昇格攻撃防止手法

2.1 考え方

提案手法は，仮想マシンモニタである KVM 内に従来手法と同様のセキュリティ機構を実現することによって，以下に示す従来手法の課題に対処する．

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

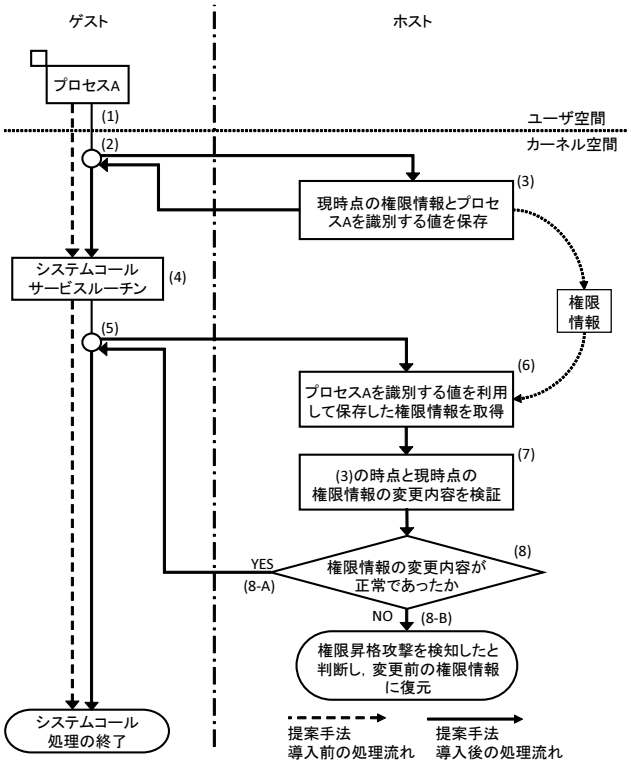

図 1 提案手法の処理流れ

（課題 1） 導入にカーネルソースコードの変更が必要
（課題 2） 保存した権限情報を改ざんすることが可能

仮想マシンモニタ内に提案手法を実現することによって，手法の導入にカーネルソースコードの変更が不要となり，(課題 1) に対処できる．また，ゲスト OS のプロセスはホストのメモリに対して読み書きを実行できない．このため，システムコール処理前のプロセスの権限情報をホスト OS 側のメモリに保存することによって，権限情報の改ざんが困難となり，(課題 2) に対処できる．

2.2 提案手法の処理流れ

提案手法の処理流れを**図 1** に示し，以下で説明する．

(1) ゲスト OS のプロセス（以降，プロセス A）がシステムコールを発行する．
(2) システムコールサービスルーチン（システムコール本来の処理）への移行をフックし，提案手法の処理へ移行する．

表 1 権限昇格攻撃の検知防止実験結果

CVE 番号	脆弱性の概要	検知防止可能
CVE-2016-0728	keyctl() における整数オーバフローならびに解放済みメモリの使用	✓
CVE-2014-0038	recvmmsg() におけるパラメータのチェック不備によるメモリ破壊	✓

(3) 現時点の権限情報（システムコール処理前の権限情報）とプロセス A を識別できる値を保存する．
(4) ゲスト OS へ処理を移行し，システムコールサービスルーチンが実行される．
(5) システムコールサービスルーチンの実行直後に処理をフックし，提案手法の処理へ移行する．
(6) プロセス A を識別できる値を取得し，この値を利用して，保存した権限情報の中からプロセス A の権限情報を取得する．
(7) (3) の時点で保存した権限情報と現時点での権限情報を比較し，権限情報の変更を検証する．
(8) システムコール処理による権限情報の変更が正常なものであったかを確認する．

(A) 権限情報の変更が正常なものであった場合，権限昇格攻撃は行われていないと判断し，ゲスト OS に処理を移行する．

(B) 権限情報の変更が正常なものでなかった場合，権限昇格攻撃が行われたと判断し，プロセス A の権限情報を (3) の時点での権限情報に復元する．

3. 評価

3.1 評価内容と評価環境

本章では，提案手法を導入した状態で，ゲスト OS 上で**表 1** に示す脆弱性を悪用した権限昇格攻撃を実施し，提案手法が攻撃を検知防止できるか否かを評価した．権限昇格攻撃には Web 上で入手できる 2 つのエクスプロイトコード [3][4] を利用した．権限昇格攻撃を検知した際に，提案手法は攻撃を検知したことをログに記録する．このため，攻撃を検知できたか否かは，ログに攻撃を検知したことを示す記録が残っているか否かで判断した．また，用いたエクスプロイトコードはすべて，攻撃が成功すると root 権限でシェルを起動する．このため，攻撃を防止できたか否かは，エクスプロイトコードを実行した際に root 権限のシェルが起動したか否かによって判断した．なお，評価ではゲスト OS に Ubuntu 14.04 LTS (Linux 3.13.0, 64bit) を使用し，ホスト OS に Ubuntu 18.04 LTS (Linux 4.15.18, 64bit) を使用した．

3.2 評価結果と考察

表 1 から，提案手法はそれぞれの脆弱性を悪用した権限昇格攻撃を検知防止できたことが分かる．具体的には，CVE-2016-0728 では，keyctl システムコールの前後で uid 群，gid 群，cap_permitted，および cap_effective が変化していることを検知した．CVE-2014-0038 では，open システムコールの前後で uid 群，gid 群，cap_permitted，および cap_effective が変化していることを検知した．また，2 つのエクスプロイトコードのどちらを実行しても攻撃が防止され，一般ユーザ権限でシェルが起動した．

今回の検知防止実験で用いたエクスプロイトコード以外による攻撃であっても，システムコール処理中に権限情報を改ざんする攻撃であれば，提案手法により検知防止できると推察できる．しかし，提案手法が検知防止できるのは，システムコール処理中に権限情報を改ざんする権限昇格攻撃のみである．たとえば，所有者が root であり，setuid ビットがセットされた実行ファイルに脆弱性があり，この脆弱性を悪用して，root 権限で動作するプログラムの実行を奪取する権限昇格攻撃に対して，提案手法は検知防止できない．

また，提案手法は仮想マシンモニタには脆弱性がなく，信頼できると仮定している．このため，仮想マシンモニタに脆弱性があり，ホスト OS 側のメモリ領域を改ざんできる場合，攻撃者はホスト OS 側のメモリ領域に保存されている権限情報を改ざんすることで，提案手法をバイパスすることができる．

4. おわりに

提案手法の有用性を確かめるために，Linux カーネルに存在する 2 つの脆弱性を悪用した権限昇格攻撃を実施し，提案手法がこれらの攻撃を検知防止できるか否かを評価した．評価結果から，提案手法を用いることにより，複数の種類の権限昇格攻撃を検知し，システムに被害が生じる前に防止できることを示した．

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] Yamauchi, T., Akao, Y., Yoshitani, R., et al.: Additional Kernel Observer to Prevent Privilege Escalation Attacks by Focusing on System Call Privilege Changes, Proceedings of the 2018 IEEE Conference on Dependable and Secure Computing (IEEE DSC 2018), pp.172–179 (2018).

[2] 福本淳文，山内利宏：KVM 上のゲスト OS 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，情報処理学会報告，vol.2019-CSEC-84，no.7，pp.1–7 (2019).

[3] PerceptionPointTeam: cve_2016_0728 exploit, GitHub Gist, available from ⟨https://gist.github.com/PerceptionPointTeam/18b1e86d1c0f8531ff8f⟩ (accessed 2019-06-10).

[4] ExploitDatabase: Linux Kernel 3.4 ¡ 3.13.2 (Ubuntu 13.04/13.10 x64) - 'CONFIG_X86_X32=y' Local Privilege Escalation (3) available from ⟨https://www.exploit-db.com/exploits/31347⟩ (accessed 2019-06-10).