平成 27 年度 卒業論文概要			
所　　属	機械情報工学科	指導教員	光來 健一
学生番号	12237065	学生氏名	二神 翔太
論文題目	クラウドにおけるネストした仮想化を用いた安全な帯域外リモート管理		

1　はじめに

近年，ネットワークを経由してサービスを提供するクラウドコンピューティングが普及している．そのサービスの一つとして，ユーザに仮想マシン（VM）を提供する IaaS 型クラウドがある．ユーザが VM にアクセスできるようにするために，クラウドは**帯域外リモート管理**と呼ばれる機能を提供している．帯域外リモート管理では，管理 VM と呼ばれる VM を経由してユーザの VM にアクセスすることにより，ユーザ VM 内のシステムの管理を行う．しかし，クラウドのシステム管理者は必ずしも信頼できるとは限らないため，管理 VM においてリモート管理の入出力情報を盗聴される可能性がある．これまでに，管理 VM における盗聴を防ぐ手法が提案されてきたが，仮想化システムの一部を信頼しなければならないため様々な問題が生じていた．

本研究では，仮想化システムの外側で安全な帯域外リモート管理を実現する *VSBypass* を提案する．

2　帯域外リモート管理における情報漏洩

帯域外リモート管理は，図 1 のように，管理 VM 経由で間接的にユーザ VM にアクセスする管理手法である．SSH を用いる場合，ユーザは管理 VM にログインし，ユーザ VM の仮想的なシリアルデバイスにアクセスすることにより，ユーザ VM への入出力を行う．ユーザ VM に SSH で直接ログインして管理を行うことも可能だが，ユーザ VM のネットワーク障害時などユーザ VM のネットワークに接続できない場合でも，ユーザ VM 内のシステム管理を行えるという利点がある．

管理 VM はクラウドのシステム管理者によって管理されているが，クラウド事業者は信頼できるとしても，その管理者は必ずしも信頼できるとは限らない．実際，Google の管理者がユーザのプライバシを侵害するという事件が発生している．また，サイバー犯罪の 28% は内部犯行であり，管理者の 35% は機密情報に無断でアクセスしたことがあるという報告もある．悪意のある管理者は管理 VM において帯域外リモート管理の入出力情報を容易に盗聴することができる．それにより，ユーザ VM のログインパスワードなどの機密情報が漏洩する恐れがある．

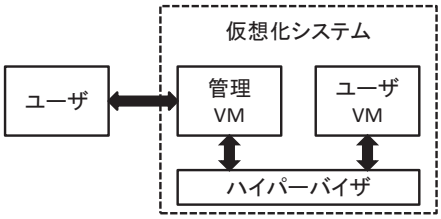


図 1: ユーザ VM の帯域外リモート管理

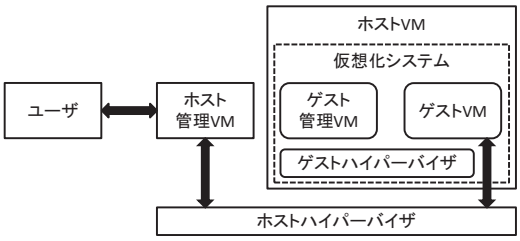


図 2: VSBypass のシステム構成

従来，仮想化システムの根幹であるハイパーバイザを信頼して，管理 VM における情報漏洩を防ぐ手法が提案されてきた．例えば，SCCrypt [1] ではユーザとハイパーバイザの間で入出力を暗号化することで機密情報の漏洩を防ぐ．しかし，システム管理者が信頼できない場合，ハイパーバイザを含めた仮想化システム全体を管理させられなくなるという問題がある．また，管理 VM からハイパーバイザを攻撃するのは比較的容易であるという報告もある．さらに，ハイパーバイザと管理 VM が明確に分離されている仮想化システムにのみ適用可能という制限がある．その上，リモート管理ツールに暗号化・復号化のためのサポートを追加する必要がある．

3　VSBypass

本研究では，ネストした仮想化と呼ばれる技術を用いて，仮想化システムの外側で安全に帯域外リモート管理を実現する VSBypass を提案する．ネストした仮想化は，仮想化システム全体を VM の中で動作させる技術である．VSBypass では図 2 のように，従来の仮想化システムを動作させる VM をホスト VM と呼び，ホスト VM を動作させるためのハイパーバイザをホストハイパーバイザと呼ぶ．ホスト VM を管理するための VM はホスト管理 VM と呼ぶ．これらは信頼できるクラウド事業者によって管理されるものとする．一方，ホスト VM 内の従来の VM とハイパーバイザをそれぞれゲスト VM，ゲストハイパーバイザと呼ぶ．VSBypass では，ユーザはホスト管理 VM を経由してホスト VM 内のゲスト VM に直接アクセスすることにより帯域外リモート管理を行う．

VSBypass は従来の帯域外リモート管理における様々な問題を解決することができる．第一に，ゲスト管理 VM やゲストハイパーバイザを経由せずに帯域外リモート管理を行えるため，信頼できない管理者がホスト VM 内の従来の仮想化システム全体を管理することができる．第二に，ホスト VM の内部からホスト管理 VM やホストパイパーバイザを攻撃するのは難しいため，従来手法より安全である．第三に，従来の仮想化システム全体を仮想化するため，どのような仮想化システ

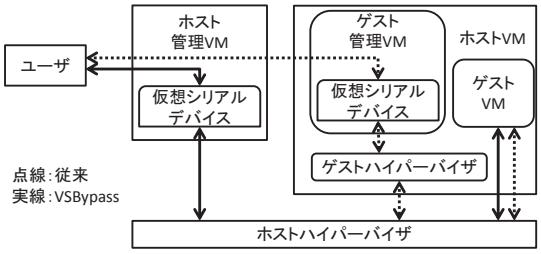
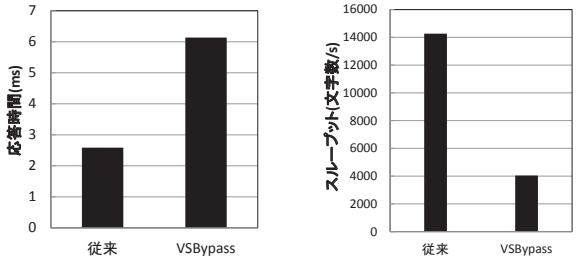


図 3: 入出力命令の横取り



(a) 応答時間　(b) スループット

図 5: 実験結果

ムであっても利用することができる．第四に，入出力の暗号化を行わないため，ユーザは既存のリモート管理ツールを用いることができる．

3.1 ゲスト VM の入出力命令の横取り

VSBypass は，図 3 のように，ゲスト VM 内で実行されたシリアルデバイスに対する入出力命令をホストハイパーバイザにおいて横取りする．従来，ネストした仮想化においては，ゲスト VM が入出力命令を実行すると一旦，ホストハイパーバイザに制御が移り，その後で入出力命令をゲストハイパーバイザに転送していた．そして，ゲスト管理 VM 上の仮想シリアルデバイスで処理が行われていた．VSBypass ではその代わりに，入出力命令をホストハイパーバイザで処理し，ホスト管理 VM 上の仮想シリアルデバイスで処理を行う．このように入出力処理の経路を変更することによって，信頼できないゲスト管理 VM とゲストハイパーバイザを経由せずに処理を行うことができる．

3.2 ゲスト VM への割り込み要求の転送

VSBypass は，ホスト管理 VM 上の仮想シリアルデバイスが発生させた割り込み要求（IRQ）をゲスト VM の仮想 CPU に転送する．IRQ は実行中の処理を中断して指定した処理を実行するように CPU に送られる信号である．従来，仮想シリアルデバイスはハイパーバイザの機能を用いて VM に IRQ を送信していた．しかし，ホストハイパーバイザはゲスト VM に直接 IRQ を送信することができないため，VSBypass では図 4 のようにゲストハイパーバイザ経由で IRQ を送信する．IRQ には機密情報は格納されていないため，信頼できないゲスト管理 VM やゲストハイパーバイザを経由しても情報漏洩の恐れはない．IRQ が正しく転送されない場合は，帯域外リモート管理が正しく行えなくなるため，ユーザはすぐに気がつくことができる．

3.3 プロキシ VM による互換性維持

従来の帯域外リモート管理との互換性を持たせるために，VSBypass はゲスト VM ごとにプロキシ VM を提供する．プロキシ VM はホストハイパーバイザ上で動作するホスト VM であり，ホスト管理 VM 上でゲスト VM 用の仮想シリアルデバイスを提供するためだけに用いられる．そのため，プロキシ VM の CPU とメモリは最低限でよく，ディスクやネットワークは必要ない．ユーザはプロキシ VM の仮想シリアルデバイスにアクセスすることにより，対応するゲスト VM に接続することができる．

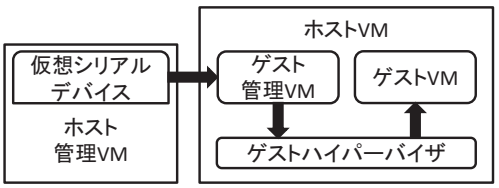


図 4: IRQ の転送

4 実験

実験には，Intel Xeon E3-1290v2 の CPU，8GB のメモリを搭載した PC を使用した．ホスト VM には 1 個の CPU と 4GB のメモリを割り当て，ゲスト VM には 1 個の CPU と 2GB のメモリを割り当てた．仮想化ソフトウェアには Xen 4.4.0 を使用し，OS には Linux 3.13 を用いた．

まず，ゲスト管理 VM とゲストハイパーバイザにおいて帯域外リモート管理の盗聴を行った．VSBypass を用いた場合には，ゲスト管理 VM からゲスト VM の仮想シリアルデバイスに接続しても入出力を盗聴することはできなかった．また，ゲストハイパーバイザにおいてゲスト VM が実行した入出力命令を盗聴することもできなかった．

次に，VSBypass と従来システムにおいて，SSH を用いた帯域外リモート管理を行い，文字を入力してからそれがゲスト VM によって出力されるまでの時間を測定した．実験結果を図 5(a) に示す．従来システムに比べて，VSBypass では 3.5 ミリ秒程度，応答時間が長くなることが分かった．また，ゲスト VM でテキストファイルを表示する際のスループットを測定した．実験結果は図 5(b) のようになり，従来のシステムと比べて，VSBypass では 28% 程度にスループットが低下することが分かった．性能低下の原因は，IRQ の転送およびネストした仮想化のオーバヘッドだと考えられる．

5 まとめ

本研究では，ネストした仮想化を用いて，仮想化システムの外側で安全に帯域外リモート管理を実現する VSBypass を提案した．VSBypass では，ゲスト VM からのシリアルデバイスへのアクセスを横取りすることで，信頼できないクラウド管理者が管理する仮想化システムをバイパスして，帯域外リモート管理を行うことができる．今後の課題は，複数のゲスト VM への帯域外リモート管理を同時に処理できるようにすることである．

参考文献

[1] K. Kourai and T. Kajiwara, Secure Out-of-band Remote Management Using Encrypted Virtual Serial Consoles in IaaS Clouds, TrustCom-15, 2015.