平成 22 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	07237018	学生氏名	宇都宮　寿仁
論文題目	VM マイグレーションを可能にする IDS オフロード機構の研究		

1　はじめに

サーバへの不正アクセスが年々増加してきており、それらへの対抗手段の一つとして侵入検知システム（IDS）が用いられている。IDS はディスクやメモリの内容を監視することにより攻撃者の侵入を検知することができるが、この IDS が攻撃者により攻撃され停止させられる事態が起きている。IDS を停止させられてしまうと侵入を検知できなくなるため、近年、仮想マシンを用いた IDS のオフロードという手法が提案されている [1]。IDS と監視対象のシステムを別々の仮想マシンで動作させることにより、IDS が攻撃を受ける危険性を減らすことができる。しかし、IDS をオフロードすると仮想マシンを正常にマイグレーションすることができなくなるという問題がある。仮想マシンはサーバマシンのメンテナンスが必要となった際や負荷が高くなった際などに、マイグレーションを行うことでサービスを停止させることなく別のサーバマシンに移動させることができる。IDS をオフロードすると、オフロード先の仮想マシンをマイグレーションすることができないため、監視を継続することができなくなる。

そこで本研究では、オフロードされた IDS も一緒にマイグレーションすることができる機構を提案する。この機構によりサーバを監視したままマイグレーションを行うことができる。

2　IDS オフロード時のマイグレーション

仮想化ソフトウェアとして Xen を用いる場合、IDS は図 1 のようにドメイン U と呼ばれる仮想マシンからドメイン 0 と呼ばれる仮想マシンにオフロードされる。Xen はドメイン U とドメイン 0 という二種類の仮想マシンから成り、ドメイン U は一般のサーバを動かすために使われ、ドメイン 0 はドメイン U を管理するために使われる。IDS を別の仮想マシンで動かすためには、仮想マシンをまたがって監視する権限が必要になるため、特権をもったドメイン 0 にしか IDS をオフロードすることができない。

ドメイン U はマイグレーションと呼ばれる操作によって、別のサーバマシンに移動させることができる。例えば、物理マシン自身をメンテナンスするためには物理マシンを停止させなければならないが、その際には仮想マシン上で提供されているサービスが停止してしまう。マイグレーションはサービスを停止させることなく別の物理マシンへ仮想マシンを移動させることができる機能である。

しかし、ドメイン 0 はマイグレーションによって別の物理マシンに移動させることができない。ドメイン 0 はドメイン U を管理するために物理マシンごとに 1 つだけ存在する特殊な仮想マシンであり、マイグレーションできるように設計されていないためである。IDS のオフロードはドメイン 0 にしか行えないため、ドメイン U のオフロード時に IDS を一緒にマイグレーションすることができなかった。ドメイン U だけがマイグレーションされると、IDS による監視が行われていない状態になるため、セキュリティが低下する。

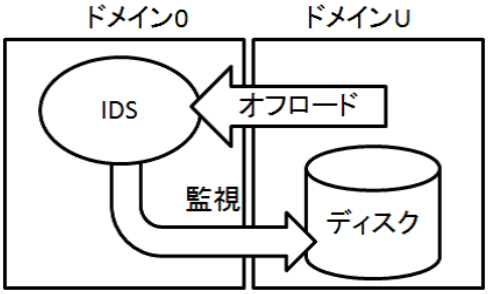

図 1　IDS オフロード

3　ドメイン M：オフロード専用仮想マシン

本研究ではオフロードした IDS を実行することができ、マイグレーションを行うこともできるオフロード専用仮想マシンであるドメイン M を提案する。ドメイン M はドメイン 0 の持つ特権を監視対象のドメイン U に対してだけ持つ。また、ドメイン M はドメイン U を監視したまま別の物理マシンへのマイグレーションを実行でき、マイグレーション後も監視を継続することができる。

3.1　ディスクの監視

ドメイン U の仮想ディスクをネットワーク上に配置することで、ドメイン M 上の IDS からのディスク監視を可能にする。ネットワーク上の仮想ディスクへのアクセスは NFS サーバを用いて実現しており、ドメイン U は NFS サーバ上のディスクイメージを使って起動する。ドメイン M は NFS サーバ上のドメイン U 用のディスクイメージをマウントすることによってドメイン U のディスクを監視することができる。例えば、Tripwire を用いてディスクの監視を行うことができる。Tripwire はディスク上のファイルを読み込み整合性をチェックする IDS である。Tripwire の設定ファイルをマウントした仮想ディスクをチェックするように書き換えることでディスクの整合性をチェックできる。

ドメイン M はマイグレーション後も NFS サーバ上のドメイン U の仮想ディスクにアクセスすることでディスクの監視を継続することができる。ドメイン U をマイグレーションしても図 2 のように同じ NFS サーバ上のディスクイメージを使い続けるためである。

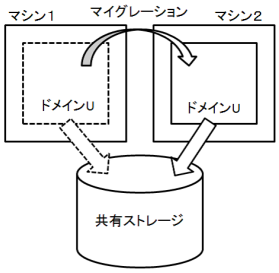

図 2　ディスク監視

3.2　メモリの監視

ドメイン M に監視先のドメイン U のメモリにアクセスする権限を与えることでメモリの監視を可能にする。ドメイン M にこのような特権を与えるために、Xen の機能を用いてどの仮想マシンからどの仮想マシンを監視できるようにするかの設定を行う。ドメイン U のメモリページをドメイン M にマップすることにより、ドメイン M 上の IDS がメモリの監視を行うことができる。例えば、ドメイン U の OS カーネルのメモリを監視することで、不正な書き換えを検出することができる。

ドメイン M のマイグレーション時のメモリ管理は基本的にはドメイン U の場合と同様に行われる。ドメイン 0 でコマンドを実行してマイグレーションを開始すると、ドメイン M のメモリの内容を読み出し、ネットワーク経由でマイグレーション先の物理マシンに送る。マイグレーション先では新たにドメイン M を作成し、送られてきたメモリの内容をそのドメイン M のメモリに書き込む。これにより、マイグレーション元で動いていた仮想マシンのメモリイメージをマイグレーション先に移動させることができる。

これに加えて、マイグレーション後も監視先のドメイン U のメモリを監視できるようにするため、ドメイン U のどのメモリページをマップしているかという情報も記録する。ドメイン M のメモリをマイグレーション先に送る際にページテーブルを調べて、ドメイン U のメモリがマップされていれば、マップしていることを示すフラグを立てる。マイグレーション先でメモリを復元する際にこのフラグが立っていれば、同時にマイグレーションしたドメイン U のメモリを再びマップする。

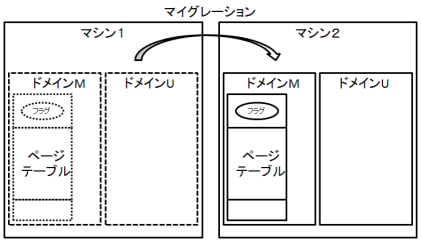

図 3　マイグレーション

4　実験

ドメイン U のメモリをマップした状態でドメイン M をマイグレーションするときのオーバヘッドを計測した。ドメイン M で動かした IDS にはドメイン U の OS カーネルのメモリのチェックサムを計算するプログラムを使用した。チェックサムは誤り検出方法の一つでブロック内のデータを数値とみなし合計を計算するものである。今回は 1448 ページをマップした。実験に使用したマシンは 2 台ともに CPU は Intel Quad 2.83GHz、メモリは 4GB であった。Xen 4.0.1 を用い、ドメイン 0、ドメイン M、ドメイン U で Linux 2.6.32.25 を動作させた。ドメイン M、ドメイン U には 512MB のメモリを割り当てた。

ドメイン U のメモリをマップした状態とマップしていない状態で、ドメイン M をマイグレーションするのにかかる時間を計測した。実験結果は表 1 のようになり、マップしている状態のほうが 1 秒ほど高速になった。ドメイン U のメモリをマップしていた時のほうが処理が増えるため時間がかかるはずである。

表 1　マイグレーションにかかる時間

	マイグレーション [秒]
マップなし	90.6
マップあり	89.6

そこでドメイン M をサスペンド、レジュームするのにかかる時間を計測した。マイグレーションは仮想マシンをサスペンドし、メモリイメージを転送してからレジュームするという操作を行うため、サスペンド・レジュームの時間を調べることができる。この実験の結果は表 2 のようになり、マップした状態のほうがサスペンド・レジュームともに 1 秒ほど高速になっていることが分かる。測定誤差とも考えられるが、詳細な分析は今後の課題である。

表 2　サスペンド・レジュームにかかる時間

	サスペンド [秒]	レジューム [秒]
マップなし	12.9	21.6
マップあり	11.9	20.6

5　おわりに

本研究では IDS をオフロードしたままマイグレーションが行える仮想マシンであるドメイン M を提案した。ドメイン M は NFS を用いることでドメイン U のディスクにアクセスできるようにし、マイグレーション後も監視を継続することができるまた、特定のドメイン U に対する特権を与えることによってメモリをマップできるようにし、マイグレーション後もメモリマップを復元することで監視を継続することができる。今後の課題は、ライブマイグレーションに対し、ドメイン U のネットワークも監視できるようにすることである。

参考文献

[1] Livewire(T. Garfinkel and M. Rosenblum, A Virtual Machine Introspection Based Architecture for Intrusion Detection, In Proc. of the 10th Annual Network and Distributed System Security Symposium, 2003.)