インタークラウドにおける インスタンスマイグレーションに関する高速化手法の一検討

山下 暁香[1] 小口 正人[1]

概要：近年，実世界におけるデータ量の増加—ビッグデータにより，大量のデータがローカル端末ではなく，クラウド上で管理されるようになった．クラウドにおける大量のデータ処理には多くの利点があるが，セキュリティという面で無視できない欠点がある．特に，2,3 年以内に実用段階になる技術と言われているインタークラウドでは，VM のマイグレーションはセキュアなネットワークを通して行われることが必須である．しかし，セキュリティとマイグレーションの速度はトレードオフの関係にあり，セキュリティを強くすればマイグレーションの速度は遅くなる．本論文では，インタークラウドにおけるマイグレーションのセキュリティと速度の双方を両立するマイグレーション方法を提案する．本論文における提案手法では，暗号化と復号のタイミング及び部分を上手く調節することで既存のマイグレーション手法の速度が改善できることを示した．本論文の実機実験では，マイグレーション前に VM のイメージを圧縮し，差分部分のみを暗号化することで，120ms 以上の高遅延環境で，既存のマイグレーション手法の 70% 以下の移動時間でマイグレーションが実現できることを示した．

A Study of Fast Instance Migration Method on Inter-cloud Networks

Akika Yamashita[1] Masato Oguchi[1]

1. はじめに

近年のデータ収集技術とデータ解析技術により，実世界とサイバー空間におけるデータや情報量が爆発的に増加している．その結果として，これらの大量の情報は，ユーザのローカル端末ではなく，クラウド上で管理されるようになった．クラウド上でデータや情報を管理する利点としては，以下の 3 点があげられる．(1) ユーザ個人が大容量のストレージや専用のソフトウェアを保持する必要がない，(2) ユーザがいつでも必要なときに，ネットワークを通して，要求するデータにアクセスすることが可能，(3) サーバ故障時や災害時に，クラウド上における仮想マシン (VM) やデータが他サーバに複製，マイグレートされるので，データや情報を失う可能性が極めて低い．

クラウドコンピューティング技術は，プライベートクラウドとパブリッククラウドが分離してそれぞれ使用される「シングルクラウド」，プライベートクラウドとパブリッククラウドがネットワークで接続された「ハイブリッドクラウド」と進化してきた．そして，近い将来，複数のクラウド同士がネットワークで接続された「インタークラウド」が実用段階になるであろうと言われている．しかし，クラウドコンピューティング技術には，セキュリティという観点から見て欠点がある．例えば，サーバ上で稼働しているそれぞれの VM は管理 OS が管理しているので，1 つの VM に対する攻撃，または脆弱性が他の VM にも影響を及ぼす可能性がある．さらに，VM のマイグレーション時に，デバイス共有やライブマイグレーションによる VM の物理移動から派生する問題がある．VM のマイグレーション時には，メモリのイメージを転送するので，マイグレーションの攻撃を防ぐために，暗号化されたセキュアなネットワークを用いることが必須となる．

特に，インタークラウドでは，異なるクラウドプロバイダ間で VM をマイグレーションする状況が考えられるので，マイグレーション時のセキュリティは一層重要である．しかし，セキュリティとマイグレーションの速度はトレー

[1] お茶の水女子大学
Ohanomizu University, 2-1-1 Otsuka, Bunkyo-ku, Tokyo 112–0012, Japan

ドオフの関係にあり，セキュリティを強くすればするほど，マイグレーションの速度は遅くなる．インタークラウドにおけるクラウド間の接続には，例えば，デフォルトで IPsec の使用が想定されるが，IPsec[1] を使う場合，送出されるパケットは，小さなフラグメントに分割され，その一つ一つに対して，暗号化，復号が行われるため，効率が悪く，大きなサイズの VM をマイグレートするときには，多くの時間がかかることが予想される．

本論文では，VM をセキュアに，かつ，早くマイグレートする手法を提案し，実機における評価実験によって，その性能向上率を示した．本論文における提案手法では，暗号化されたネットワークでマイグレーションをするのではなく，VM のイメージの必要部分を暗号化し，暗号化された VM をマイグレーション先のサーバに移動した後で復号して，VM の差分を更新するという手法である．実機実験では，IPsec トンネルにおいて，通常のマイグレーションコマンドを利用する既存手法と，VM 全体に対して暗号化処理を施す提案手法，さらに，VM を圧縮することにより，必要部分のみに対して暗号化処理を施す場合の静的な VM マイグレーション速度を比較した．実験の結果により，必要部分のみを圧縮することで，120ms 以上の高遅延環境で提案手法が既存手法の 70% 以下の速度で実現できることを示した．

構成：2 節でインタークラウドにおけるマイグレーション技術の特徴を述べ，3 節で本論文の実験環境を説明する．4 節で実験概要と結果を述べ，5 節で関連研究を紹介し，本提案手法の有効性を示す．そして，6 節で本論文をまとめる．

2. インタークラウド

2.1 インタークラウドのアーキテクチャ

図 1 に示したように，インタークラウドは，シングルクラウド，ハイブリッククラウドの発展系である．

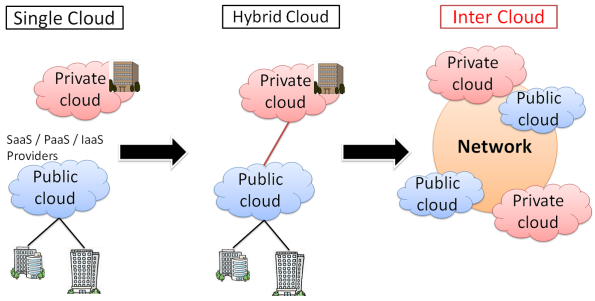

図 1 クラウドコンピューティング技術の進化

シンブルクラウドでは，パブリッククラウドとプライベートクラウドがそれぞれ別々に使用されていたが，ハイブリットクラウドでは，パブリッククラウドとプライベートクラウドがネットワークによって接続されるようになった．さらに，インタークラウドでは，異なるクラウドプロバイダが管理するクラウドが同じネットワーク上で接続され，VM が異なるクラウド間を移動するようになる．

インタークラウドの構成は**図 2** に示すように，Cloud Broker, Cloud Manager, Cloud Directory から構成されている．

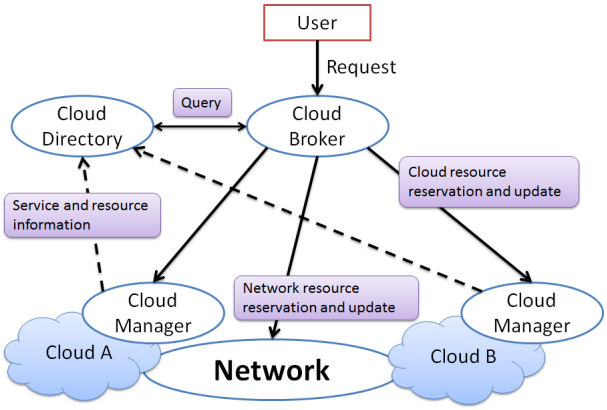

図 2 インタークラウドの構成

Cloud Manager は，各クラウドリソースの管理，課金，リソース割り当てを実行する．Cloud Directory は，Cloud Manager からクラウドサービスやリソースの情報を収集して，Cloud Broker からの適切なクラウドサービスの問い合わせに対応する．

2.2 インタークラウドにおける VM マイグレーション

インタークラウドでは，異なるクラウド間でネットワークを通して VM のやりとりをするので，マイグレーションが安全に行われることが必須である．本論文の評価実験では，暗号化されたネットワークの代表例として，IPsec[1] を用いたときのマイグレーションを既存手法とした．既存手法におけるマイグレーションを**図 3** に示した．ストレージに関して，グローバルな環境における VM マイグレーション実行時に iSCSI などの共有ストレージを用いると，マイグレーション先から VM がストレージへ遠隔アクセスをする必要があり，非現実的である．よって，本論文における実験では，それぞれの VM が用いるストレージは，ローカルストレージとした．

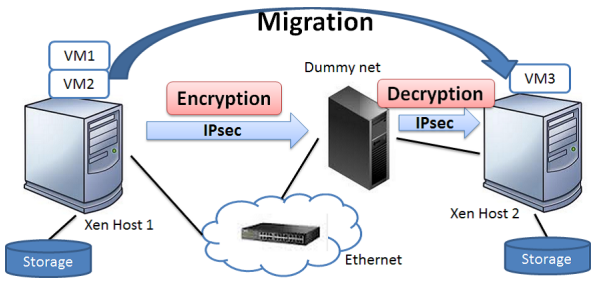

図 3 インタークラウドにおけるマイグレーションの既存手法

手順は以下の通りになると考えられる．

(1) マイグレーション元であるホストサーバにおける VM の停止

(2) IPsec トンネルを通して VM を他のホストサーバへマイグレート

(3) マイグレーション先であるホストサーバにおける VM の再起動

このように，既存手法で IPsec トンネルを通して VM をマイグレートする場合，データは，IP 層で IP パケットに分割され，それぞれの分割されたパケットに対して，暗号化処理と復号処理が行われる．IPsec とは，インターネット上の 2 地点間に仮想的なトンネルを作り，そこに IP パケットを通すものである．トンネルを通すパケットには，暗号をかけることで，パケットの中身が第 3 者によって盗聴されることを防ぐ．トンネルの入り口である IPsec のゲートウェイは，LAN から受け取ったパケットを暗号化し，トンネルの出口のゲートウェイを宛先にした IP パケットに暗号化したデータを入れ（カプセル化），宛先 IP アドレスへ転送する．トンネル出口のゲートウェイは，受信パケットからカプセル化をほどいて，暗号化されたパケットを取り出し，送信側と同じ暗号鍵を用いて，復号する．

IPsec には 2 種類の動作モード（トランスポートモード，トンネルモード）があり，トランスポートモードでは，端末同士が 1 対 1 対応の関係で，やり取りをするパケットのデータの部分のみに対して暗号化処理を施す．さらに，2 種類のプロトコル（ESP，AH）があり，ESP では，暗号化，送信元確認，およびデータの改ざん検出機能を提供するのに対して，AH には暗号化が無い．本実験では，マイグレーションは，2 台の端末間で 1 対 1 の関係で行われるので，トランスポートモードを用い，プロトコルについては，よりセキュアである ESP カプセル化を利用した．

トランスポートモードにおける ESP カプセル化パケットを**図 4** に示す．

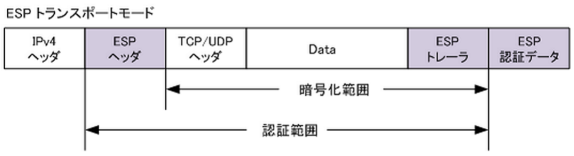

図 4 トランスポートモードにおける EPS カプセル化パケット

このように，それぞれの IP パケットに対して，暗号化，認証を行うので，マイグレーションの完了までに時間がかかり，効率が悪い．実際に，本論文の予備実験にて，IPsec トンネルと普通のトンネルの通信実験を行った結果，IPsec トンネルを使うと，約 0.5〜1MB/s だけスループットが減少することを確認した．

既存手法に対して，本論文における提案手法（**図 5**）では，IPsec を用いてそれぞれの IP パケットに対して暗号化処理を施すのではなく，VM のマイグレーション前に，VM の必要部分のみを暗号化し，SCP (Secure Copy Protocol) を用いて暗号化された必要部分をマイグレーション先のサーバへ転送し，マイグレーション先サーバで復号処理を行い，差分を更新する．VM 自体を暗号化しているので，IPsec トンネルを使わなくても，第 3 者によって盗聴される心配はない．また，MV の転送に実験では SCP を用いたが，既に暗号化されたデータであるため，RCP や FTP などの非暗号化ファイル転送を用いても，セキュリティ上は問題ない．

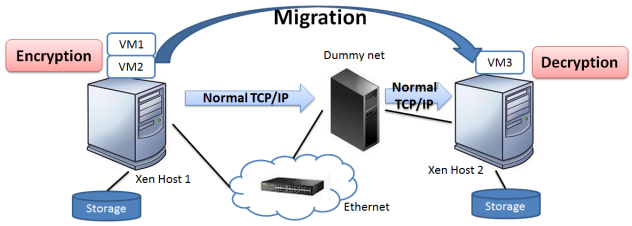

図 5 インタークラウドにおけるマイグレーションの提案手法

手順は以下の通りである．

(1) マイグレーション元であるホストサーバにおける VM の停止

(2) VM のイメージの必要部分を暗号化

(3) SCP を用いて暗号化されたイメージを転送

(4) マイグレーション先であるホストサーバにおいてイメージを復号し，差分を更新

(5) VM の再起動

VM のイメージの必要部分のみを暗号化する手法として，本実験では，VM のイメージを圧縮した後で暗号化処理を施すことにより，意味のあるデータのみを暗号化した．また，比較のため，VM のイメージ全体を暗号化する場合の実験も行った．VM のイメージ全体の暗号化をする場合が，本論文における提案手法の最長の時間がかかるマイグレーションであると言える．なお，本論文における実験は全て静的なマイグレーションを用いたもので，VM を停止させずに他サーバへ移動するライブマイグレーションについては今後の課題とする．

3. 実験環境

本論文の評価実験では，VM マイグレーションについて，3 種類の実機実験を行い，比較，評価した．既存手法におけるマイグレーション（図 3）と提案手法におけるマイグレーション（図 5）の VM 全体を暗号化する場合と VM のうち，意味のあるデータ部分のみを暗号化する場合である．

本論文における実験では，2 つの高遅延環境に存在する異なるクラウド間で VM をマイグレーションする状況を想定している．高遅延環境にあるクラウド間で iSCSI などのストレージ共有を行うのは非現実的なので，マイグレー

ション元とマイグレーション先のサーバは，それぞれローカルストレージを用いる．

3.1 実験に用いた端末

3.1.1 Xen

クラウドを管理する仮想化ソフトウェアとして，それぞれのホストサーバに Xen[2] を導入した．Xen は一つのハードウェアでマルチ OS の並列処理と管理を提供する．Xen では，それぞれの VM は Domain と呼ばれ，Xen ハイパーバイザは一つ，または複数の OS をサポートし，物理 CPU のスケジューリングを行う．ホスト OS は Domain-0（dom0）と呼ばれ，Dom0 上に新たに作られたゲスト OS，つまり，仮想マシン VM は Domain-U（domU）と呼ばれる．それぞれのホストサーバでは，一つの dom0 が複数の VM である domU を管理しており，dom0 は，ハイパーバイザが起動し，認証が完了すると，自動的に起動する．全ての物理ハードウェアは dom0 から直接アクセス可能であり，システムの管理者は dom0 を通して，全ての domU にログインすることが可能である．

マイグレーション元と先である 2 台の Xen ホストサーバのスペックは表 1，サーバ上の VM，つまり，それぞれの Domain のスペックは表 2 の通りである．

表 1 Xen ホストサーバ 1 と 2 の設定

OS	Linux 2.6.32-5-xen-amd64 and xen-4.0-amd64
Distribution	Debian GNU / Linux 6.0.2
CPU	Intel(R) Xeon(R) CPU 3.60GHz
Memory	4 GByte
Disk	222 GByte

表 2 Xen ホストサーバ上における VM の設定

OS	Linux 2.6.32-5-xen-amd64 and xen-4.0-amd64
Distribution	Debian GNU / Linux 6.0.2
CPU	Intel(R) Xeon(R) CPU 3.60GHz
VCPU	1 core
VCPU Memory	2 GByte
Disk	4 GByte

3.1.2 IPsec

既存手法の実験では，VM を IPsec トンネルを通してマイグレートした．IPsec は，openswan[3] をインストールすることにより導入した．IPsec のトランスポートモードを用い，VM のマイグレーション元である Xen ホストサーバが IPsec クライアント，マイグレーション先である Xen ホストサーバが IPsec サーバである．IPsec トンネルの詳細設定は 3 の通りである．

表 3 IPsec トンネルの設定

モード	トランスポートモード
IPsec アルゴリズム	ESP
認証アルゴリズム	HMAC-SHA-1
暗号化アルゴリズム	AES 128 （Pre-Shared Key）

提案手法における VM の暗号化についても，公平な比較のため，AES 128bit 鍵を用いた．暗号化と復号の実行には，OpenSSL[4] が提供するコマンドを利用した．

3.1.3 Dummynet

高遅延環境にある 2 つのクラウド環境を実現するために，Xen ホストの 2 ノードの間に Dummynet を挟んだ．Dummynet は，高遅延通信を人工的に発生させる装置である．Dummynet の端末のスペックは表 4 の通りである．

表 4 Dummynet

OS	FreeBSD 6.4-RELEASE
CPU	Intel(R) Xeon(R) CPU 3.60GHz
Disk	64 GByte

3.2 具体的な実験手順

各手法の具体的な実験手順を以下に示す．

3.2.1 既存手法におけるマイグレーションの実行手順

(1) IPsec トンネルを張る（AES key 128 bit）
(2) Xen を起動（dom0 が自動的に起動）
(3) マイグレーション元で domU を起動
(4) マイグレーション先 Xen ホストサーバの IP アドレスを指定して，migration コマンドを実行

3.2.2 提案手法のうち，VM 全てを暗号化する場合の実行手順

(1) Xen を起動（dom0 が自動的に起動）
(2) マイグレーション元で domU を起動
(3) マイグレーション元で domU を停止
(4) openssl コマンドを用いて VM のイメージファイル（disk.img）を暗号化（AES 128bit key）
(5) SCP コマンドで暗号化した VM のイメージファイルをマイグレーション先へ転送
(6) マイグレーション先で openssl コマンドを用いて VM のイメージファイル（disk.img）を復号
(7) domU を起動

3.2.3 提案手法のうち，意味のあるデータ部分のみを暗号化する場合の実行手順

(1) Xen を起動（dom0 が自動的に起動）
(2) マイグレーション元で domU を起動
(3) マイグレーション元で domU を停止
(4) VM のイメージを圧縮
(5) openssl コマンドを用いて圧縮済みの VM のイメージファイル（disk.img）を暗号化（AES 128bit key）

(6) SCP コマンドで暗号化した VM のイメージファイルをマイグレーション先へ転送
(7) マイグレーション先で openssl コマンドを用いて VM のイメージファイル（disk.img）を復号
(8) VM のイメージを解凍
(9) domU を起動

4. 実験結果と考察

既存手法と提案手法におけるマイグレーションの時間を**図 6** に示した．グラフはそれぞれ，既存手法におけるマイグレーションと，提案手法における VM 全体を暗号化，VM イメージの意味のあるデータのみを暗号化した場合の結果となっている．グラフの縦軸はマイグレーションにかかった時間（分），横軸は RTT:往復遅延時間 (ms) である．遅延の値については，0ms～200ms の値を用いた．インタークラウドで VM をグローバルに遠隔にあるサーバ（クラウド）へマイグレーションする場合を想定している．例えば，東京-大阪間の RTT は 20ms 程度，東京-アメリカ西海岸は 120ms 程度，東京-ヨーロッパ諸国は 200ms 程度である．横軸の値は，これらの RTT を元に 0ms - 200ms にした．全ての実験は，3 回施行した結果の平均をとっている．

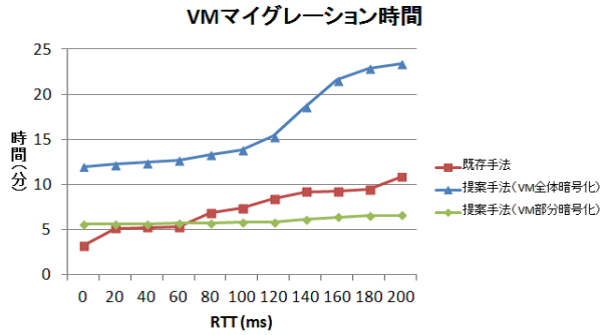

図 6 既存手法と提案手法における VM のマイグレーション時間

既存手法と提案手法のうち，意味のあるデータのみを暗号化する場合の時間を比較すると，遅延が 60ms よりも低い環境では，既存手法の方がマイグレーション時間が短いが，遅延が 80ms 以上になると，提案手法の方が既存手法よりもマイグレーション時間が短くなることがわかった．既存手法において，低遅延環境でマイグレーション時間が 5 分以上かかってしまうのは，**図 7** に示すように，VM の圧縮と解凍に時間が 4 分程度かかっているためである．本実験におけるマイグレーション元とマイグレーション先のイメージの差分は，全体のイメージ 4GB のうち，300MB 程度であった．

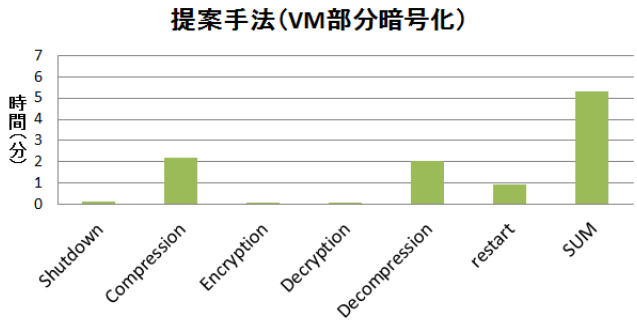

図 7 VM の部分を暗号化する場合の固定時間

一方，提案手法のうち，VM 全体を暗号化する場合は，**図 8** より，暗号化処理と復号処理が 5 分程度かかっており，4GB ある容量の VM イメージを転送するので，160ms 以上の高遅延環境では，マイグレーション時間が 20 分以上かかっている（図 6 の提案手法（VM 全体暗号化）より）．

本提案手法の今後の課題では，マイグレーション元とマイグレーション先の端末にあるイメージの差分のみを暗号化し，転送することでライブマイグレーションにも対応させていくことを目指している．つまり，本論文で実験をした，VM 全体を暗号化する提案手法にてかかったマイグレーション時間が最長であり，暗号化する VM の部分が少なくなるよう調節することで，大幅な改善が期待できる．

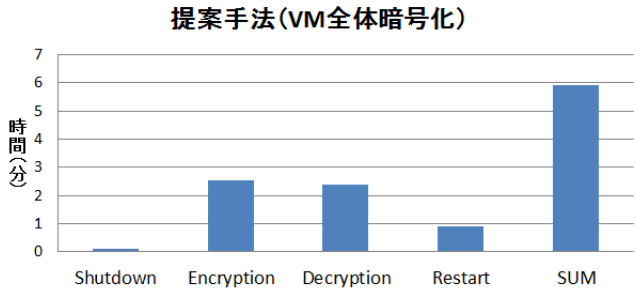

図 8 VM 全体を暗号化する場合の固定時間

既存手法におけるマイグレーション時間に対する，提案手法のうち，VM を部分暗号化した場合のマイグレーション時間の割合を**図 9** に示した．グラフからわかるように，80ms 以上の高遅延環境で，提案手法が既存手法のマイグレーション時間を下回り，さらに，120ms 以上の高遅延環境では，約 70% 以下のマイグレーション時間に抑えられている．

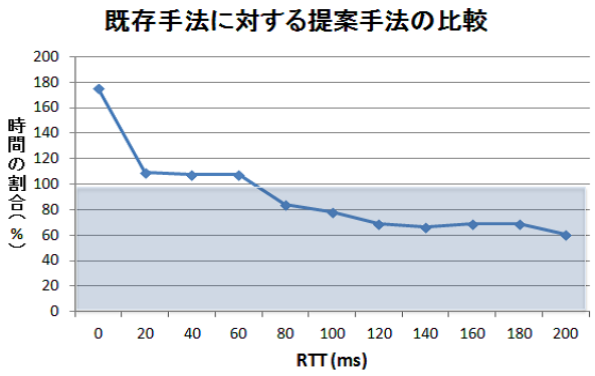

図 9 既存手法に対する提案手法のマイグレーション時間の割合

5. 関連研究

VM のマイグレーションに関する分野の研究では，主にマイグレーションのパフォーマンス向上が注目されてきた．VM を停止させずにマイグレーションをするライブマイグレーションに関しては [5]，メカニズムについては [6] で説明されている．[7] では，インタークラウドにおけるアプリケーションのサービスのスケーラビリティについて述べられている．[8] では，インタークラウドにおける VM のマイグレーション技術が述べられているが，プライバシーとセキュリティの考慮が必要である．[9], [10] では，マイグレーションのパフォーマンス改善について述べられている．

VM のマイグレーションにおけるセキュリティの考慮に関しては，J. Rexford らが，ノーハイパーバイザを提案した．ハイパーバイザをセキュアにすることは，セキュアなマイグレーションを実現する良い手法であるが，ここで提案されている手法は，現在の VM マイグレーションメカニズムの脆弱性に対して，十分ではない．

マイグレーション技術では，セキュリティとパフォーマンスの両立が大切であり，これらがトレードオフの関係にあることを注意する必要がある．

6. まとめと今後の課題

本論文では，インタークラウドにおいて，IPsec のようなネットワークのトンネルを使うのではなく，VM 自体の必要な部分に対して暗号化と復号処理を施すことで，マイグレーション時のセキュリティとパフォーマンスを両立する手法を提案した．実機実験の結果，既存のマイグレーション手法と比べて，120ms 以上の高遅延環境で提案手法では 70% のマイグレーション時間で抑えられることを示した．

本論文における提案手法のポイントとしては，(1) 暗号化処理のタイミング (2) 部分を調節することで効率的なマイグレーションを実現する．今後の課題として，Xen における既存のマイグレーションのボトルネックを調査し，ライブマイグレーションにおける効率的な手法を提案したい．

謝辞

本研究を進めるにあたり，MONASH 大学の Eng Keong Lua 教授に大変有用なアドバイスをいただきました．深く感謝いたします．

参考文献

[1] 馬場 達也，”マスタリング IPsec”，オイラリー・ジャパン，2001 年 10 月
[2] Xen project : `http://xen.org/`
[3] Openswan : `https://www.openswan.org/projects/openswan/`
[4] Jobn Viega, Matt Messier, Pravir Cbandra, ”Network Security with OpenSSL”, O’REILLY, June, 2002.
[5] Diego Perez-Botero, ”A Brief Tutorial on Live Virtual Machine Migration From a Security Perspective”, `http://www.cs.princeton.edu/~diegop/courses.html`, Princeton University, Princeton, NJ, USA, 2011.
[6] Jansen and Gerardus T., ”Mechanism for Inter-Cloud Live Migration of Virtualization Systems”, 2012.
[7] R. Buyya and R. ranjan and R. N. Calheiros, ”InterCloud: Utility-Oriented Federation of Cloud Computing Environments for Scaling of Application Services”, the 10th International Conference on Algorithms and Architecture for Parallel Processing (ICA3PP2010), 13 – 31, May, 2010.
[8] K. Nagin and D. Hadas and Z. Dubitzky and A. Glikson and I. Loy and B. Rochwerger adn L. Schour, ”Intercloud mobility of virtual machines”, the 4th Annual International Conference on Systems and Storage Article(SYSTOR2011), No. 3, May, 2011.
[9] M. Tsugawa and P. Riteau and A. Matsunaga and J. Fortes, ”User-level virtual networking mechanisms to support virtual machine migration over multiple clouds”, IEEE GLOBECOM Workshops (GC Wkshps 2010), 568–572, December, 2010.
[10] Hong Xu and Baochun Li, ”Egalitarian Stable Matching for VM Migration in Cloud Computing”, IEEE INFOCOM Workshop on Cloud Computing(INFOCOM Workshop2011), 631–636, 2011.
[11] Eric Keller, Jakub Szefer, Jennifer Rexford, Ruby B. Lee, ”NoHype: Virtualized Cloud Infrastructure without the Virtualization”, ’10 Proceedings of the 37th annual international symposium on Computer architecture (ISCA2010), 350-361, 2010.