複数のオーバレイネットワークを制御するための プライベートなネットワーク環境

光来健一[†] 廣津登志夫[†] 佐藤孝治[†]
明石修[†] 菅原俊治[†] 千葉滋[††]

オーバレイネットワークはユーザが仮想的なネットワークを構築することを可能にする．しかしながら，各ホストの OS は複数のオーバレイネットワークをうまく制御できないため，ユーザが自由にオーバレイネットワークを使うには利便性と安全性の点で問題があった．この問題を解決するために，我々はネットワークの仮想化だけでなくホストの仮想化まで行う**パーソナルネットワーク**を提案する．ホストを仮想化するために OS は**プライベートなネットワーク環境**を提供し，その中で動くプロセスからは特定のオーバレイネットワークしか使えないようにする．プライベートなネットワーク環境により，プロセスは明示的にオーバレイネットワークを使い分ける必要がなくなり，プロセスを介してオーバレイネットワーク間で機密情報が漏れるのを防ぐことができる．我々はこのようなプライベートなネットワーク環境を実現するために，システムのネットワーク空間を多重化した**ポートスペース**とファイルシステムを多重化した**ファイルスペース**を開発した．

Private Network Environments for Controlling Multiple Overlay Networks

Kenichi Kourai ,[†] Toshio Hirotsu ,[†] Koji Sato ,[†]
Osamu Akashi ,[†] Tosiharu Sugawara [†] and Shigeru Chiba [††]

Overlay networks enable users to construct new virtual networks. However, since the operating systems cannot control multiple overlay networks cleverly, convenience and safety are obstacles for users to freely deploy their own overlay networks. To solve this problem, we propose *personal networks*, which virtualize hosts as well as networks. To virtualize hosts, our operating system provides *private network environments*, processes within which can use only a particular overlay network. Private network environments enable processes to use different overlay networks implicitly and to prevent secret data from leaking among overlay networks. To construct private network environments, we have developed *portspaces*, which are multiplexed network spaces, and *filespaces*, which are multiplexed filesystems.

1. はじめに

家庭への常時接続環境の普及やモバイルコンピューティングの普及に伴い，自宅，会社，出先のホットスポットなどで様々なネットワークを利用する機会が増えてきている．このような状況下では，例えば自宅から会社のメールサーバにアクセスしてメールを読む場合のように，ユーザは目的のネットワークの外部からアクセスしなければならない場合が増える．その一方で，各ネットワークの管理者はセキュリティを確保するためにファイアウォールなどを設置して，外部からのアクセスを著しく制限することが多い．そのため，自宅から会社に届いたメールを読むには，一旦会社のホストにログインして，そのホスト上でメールを読むといった不便さを強いられる場合も多い．

現在，研究が進められているオーバレイネットワークの技術を使えば，ユーザは既存のベースネットワークの上に自分専用の仮想的なネットワークを構築することができる．オーバレイネットワークは構築時にユーザを認証し，パケットをトンネリングすることにより，ファイアウォールなどのアクセス制限を安全に回避することを可能にする．そのため，自宅からでも会社のネットワーク内部のメールサーバに直接アクセスすることができる．また，一部のホストだけで構成

† NTT 未来ねっと研究所
NTT Network Innovation Laboratories
†† 東京工業大学情報理工学研究科数理・計算科学専攻
Dept. of Mathematical and Computing Sciences, Graduate School of Information Science and Engineering, Tokyo Institute of Technology

される小さなオーバレイネットワークを構築することもできる．最小限のホストだけを含めることにより，万一，自分のホストがメールウィルスに感染した場合でも，スパムメールをばらまかれる範囲を制限することができる．このように，状況に応じて自由にオーバレイネットワークを使えば，ユーザの利便性と安全性を向上させることができる．

しかしながら，現在の OS は複数のオーバレイネットワークをうまく制御できていない．これは，オーバレイネットワークが従来のネットワークの延長として作られており，共有の資源として扱われるためである．オーバレイネットワークが共有の資源として扱われると，ユーザにとっての利便性と安全性に問題が生じる．利便性の問題は，ユーザが IP アドレスなどによってアクセスするネットワーク（オーバレイネットワークとベースネットワーク）を明示的に使い分けなければならないことである．そのため，使われるネットワークに応じてアプリケーションの設定も変更が必要になる．安全性の問題は，ホスト内で複数のネットワークが混在しているため，互いに影響を与えあってしまうことである．設定の不備や攻撃の影響が全てのネットワークに及ぶ可能性がある．

我々はこれらの問題を解決するために，ユーザが複数のオーバレイネットワークを容易にかつ安全に使い分けられるようにした**パーソナルネットワーク**を提案する．パーソナルネットワークはオーバレイネットワークを用いてネットワークを仮想化するだけではなく，OS によって提供される**プライベートなネットワーク環境**を用いてホストも仮想化する．プライベートなネットワーク環境は特定のプロセスと特定のオーバレイネットワークを結びつけるため，プロセスはオーバレイネットワークの使い分けを意識しなくてよい．また，パーソナルネットワークは互いに独立しているので，IP アドレスの衝突を心配せずにベースネットワークと同じ IP アドレスを使うことができ，互いに影響を与えあうこともない．

我々はこのようなプライベートなネットワーク環境を実現するために**ポートスペース**と**ファイルスペース**を開発した．ポートスペースは 1 つの IP アドレスに対応するネットワーク空間を多重化したものであり，IP 層，トランスポート層，オーバレイネットワークを独立して管理する．同一ホストのポートスペースは全て同じ IP アドレスを使うので，一般のユーザでも簡単に新しいポートスペースを作ることができる．また，ポートスペース同士は互いに見えないので，ユーザは安全にポートスペースを使うことができる．さらに，より簡単にポートスペースを作れるようにするために，ポートスペースは親ポートスペースの状態を継承することができる．一方，ファイルスペースはファイルシステムを多重化したものであり，プロセスに異なるファイルシステムを見せる．ファイルスペースも親ファイルスペースを継承して作ることができる．

以下，2 章では 1 つのホストで複数のオーバレイネットワークを扱う場合の問題点について述べる．3 章では複数のオーバレイネットワークを制御できるようにしたパーソナルネットワークについて述べる．4 章ではパーソナルネットワークの実装の詳細について述べる．5 章ではポートスペースのオーバヘッドを調べた実験について述べる．6 章で関連研究について述べ，最後に 7 章で本稿をまとめる．

2. オーバレイネットワーク

この章ではオーバレイネットワークについて説明し，複数のオーバレイネットワークを制御する際の従来の OS の問題点について述べる．

2.1 オーバレイネットワークの利点

オーバレイネットワークは既存のネットワーク（ベースネットワーク）の上に仮想的なネットワークを構築する技術である．仮想プライベートネットワーク（VPN）や ssh のポート転送などはその一種である．オーバレイネットワークは構築時にユーザを認証し，ホストやネットワークの間でパケットをトンネリングして運ぶ．ファイアウォールが認証されたオーバレイネットワークを信用するように設定されていれば，トンネリングされたパケットはファイアウォールのアクセス制限を受けない．また，プライベートアドレスしか持たないホストなど直接到達できないホストへのルーティングも可能にする．

オーバレイネットワークはユーザが様々なネットワークを使う必要がある時に，それらのネットワークを結合させ，利便性を向上させることができる．例えば，自宅のネットワークと会社のネットワークからなるオーバレイネットワークを作ることにより，外部からの自由なアクセスを許さない会社のネットワークに自宅からでもアクセスすることができる．また，ユーザが万一の場合に備えて最小限のホストからなるネットワークを作り，安全性を向上させることもできる．ユーザが自分のホストと POP サーバだけからなるオーバレイネットワークを作れば，万一，自分のホストがメールウィルスに感染しても，スパムメールがそのオーバレイネットワークの外に送られるのを防ぐことができる．

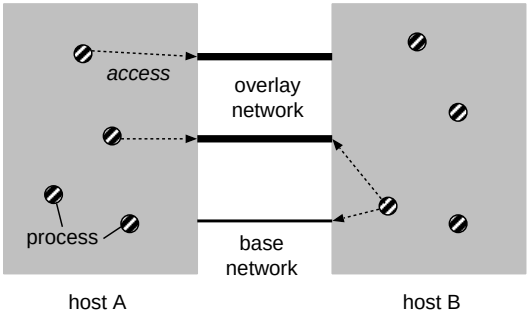

図 1 制御されていない複数のオーバレイネットワーク
Fig. 1 Overlay networks that are not controlled

2.2 複数のオーバレイネットワークの制御

ユーザが自由にオーバレイネットワークを作れるようになると，ユーザは状況に応じて複数のオーバレイネットワークとベースネットワークをそれぞれ使い分ける必要が出てくる．例えば，自宅から会社のウェブサーバにアクセスする時，認証が必要なページの閲覧には暗号化を行う専用のオーバレイネットワークを使って機密情報が外に漏れないようにすべきである．一方で，それ以外のページはベースネットワークを使って高速に閲覧したいだろう．また，ショッピングサイトでオンラインショッピングをする時には，会社との間のオーバレイネットワークとは別のオーバレイネットワークを使うべきだろう．

しかしながら，従来の OS は複数のオーバレイネットワークをうまく制御できていない．これは，オーバレイネットワークが従来のネットワークの延長として作られており，共有の資源として扱われるためである．このようにして作られるオーバレイネットワークは図 1 のように，どのプロセスからでもアクセス可能になる．1つのプロセスが複数のオーバレイネットワークにアクセスすることさえ可能である．オーバレイネットワークを作る時に，従来の OS は新しいネットワークアドレスの設定など，ネットワークの作成は支援するが，作られたネットワークへのアクセス制限は提供しない．そのため，オーバレイネットワークの使い分けはプロセスやユーザに依存する．

OS がオーバレイネットワークの使い分けを支援しないと，ユーザが複数のオーバレイネットワークを使う際に利便性と安全性の点で問題が生じる．利便性の問題は，ユーザが IP アドレスなどによってアクセスするオーバレイネットワークを明示的に使い分けなければならないことである．例えば，1つのサーバホストにオーバレイネットワークとベースネットワークの両方を使ってアクセスしたい場合，そのホストに別々の IP アドレスを割り当てて使い分けなければならない．このように，利用するネットワークに応じてサーバホストの指定方法を変えなければならないのは，ユーザにとって負担が大きい．IP アドレスの変更の影響はアプリケーションの設定にも及び，ユーザは利用するネットワークに応じて設定ファイルを書き換えなければならない．この問題を避けるために，VPN のようにホストは同時に1つのオーバレイネットワークだけしか使えないように制限することもある．

安全性における問題は，ユーザがホスト上の全てのオーバレイネットワークを同時に使えてしまうことである．そのため，独立させるべきオーバレイネットワークが1つのホスト上に複数ある場合，機密情報が互いに漏れないようにしたり，互いに攻撃されないようにすることはできない．例えば，ユーザの操作ミスで，あるオーバレイネットワーク内でのみ閲覧できる機密情報を，他のオーバレイネットワーク上のユーザに見せてしまう可能性がある．また，自分のホストで動いているプロセスに脆弱性があった場合，1つのオーバレイネットワークが受けた攻撃の影響が，そのプロセスを介して他の全てのオーバレイネットワークに拡がる可能性がある．この問題を避けるためには，ユーザがオーバレイネットワークを使う時には，それを唯一のネットワークとして使わざるを得ない．

3. パーソナルネットワーク

この章ではパーソナルネットワークと，パーソナルネットワークの特徴である各ホストのプライベートなネットワーク環境について述べる．

3.1 パーソナルネットワーク

我々はユーザが複数のオーバレイネットワークを容易にかつ安全に使い分けられるようにした**パーソナルネットワーク**を提案する．従来のオーバレイネットワークはネットワークだけを仮想化していたため，各ホストでは複数のオーバレイネットワークが混在したままであった．そこで，パーソナルネットワークでは，各ホストの OS が**プライベートなネットワーク環境**を提供してホストを仮想化し，それを特定のオーバレイネットワークと結びつける．これにより，パーソナルネットワークは図 2 のようにそれぞれのオーバレイネットワークを完全に独立させることができる．さらに，プライベートなネットワーク環境は特定のプロセスと特定のオーバレイネットワークを結びつける役割を果す．各ホストのプロセスは1つのプライベートなネットワーク環境内でのみ実行され，プライベートなネットワーク環境に結びつけられた1つのオーバレイ

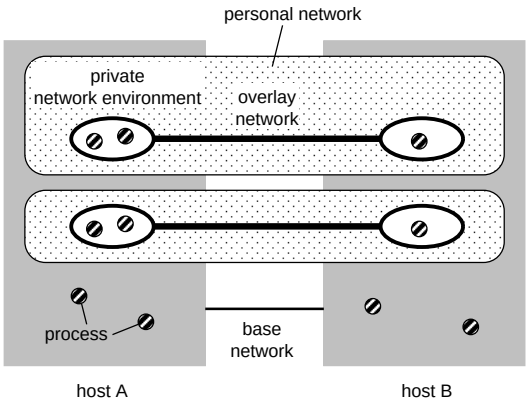

図 2 パーソナルネットワークの構成
Fig. 2 Personal networks

ネットワークだけを利用することができる．

パーソナルネットワークの中ではプロセスはどのオーバレイネットワークを使っているかを意識する必要はない．１つのパーソナルネットワークの中には１つのオーバレイネットワークしか含まれず，プロセスはそれ以外のオーバレイネットワークを使うことはできないからである．また，パーソナルネットワークは互いに独立しているため，IP アドレスの衝突を心配することなくベースネットワークと同じ IP アドレスを使うことができる．そのため，ユーザはどのパーソナルネットワークにおいても，同じサーバホストに対しては同じ IP アドレスでアクセスできる．さらに，パーソナルネットワークに外からアクセスする手段はないので，機密情報が漏れたり，攻撃の影響が拡がったりすることもない．

3.2 プライベートなネットワーク環境

プライベートなネットワーク環境は，仮想的なネットワーク空間であるポートスペースと仮想的なファイルシステムであるファイルスペースから構成される．

3.2.1 ポートスペース

ポートスペースは１つの IP アドレスに対応するネットワーク空間を多重化したものである．図 3 は，２つホストのそれぞれの IP アドレスについて２つのポートスペースに多重化している例である．各ポートスペースは以下の情報を独立して管理する．

- IP 層の設定
- IPsec の設定
- トランスポート層の設定
- トランスポート層のポート空間
- ポートに対するサービスのバインド情報

IP 層の設定にはルーティングテーブルやファイアウォールのルールなどが含まれ，各ポートスペース毎に独自

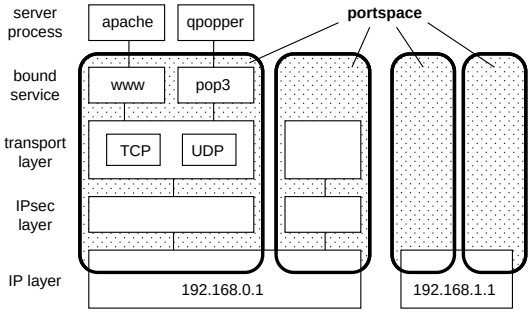

図 3 IP ネットワーク空間を多重化するポートスペース
Fig. 3 Portspaces created by multiplexing the IP network space

の設定を行うことができる．ただし IP アドレスは変更できない．ポートスペースは IPsec によるオーバレイネットワークの構築を想定しており，各ポートスペースが独立して IPsec の暗号化通信路を管理する．また，TCP や UDP などのトランスポート層は複数の IP アドレスを使うことによりそれぞれについて独立に存在させることができていたが，ポートスペースを使えば１つの IP アドレスで複数のトランスポート層を存在させることができる．これらに関連して，ポートスペース単位でソケットを管理する．

同一ホストのポートスペースは全て同じ IP アドレスを持つので，ユーザはポートスペースを作る際に新しい IP アドレスを必要としない．そのため，管理者に IP アドレスを割り当ててもらう必要がなくなり，一般のユーザでも自由にポートスペースを作ることができるようになる．その反面，ポートスペースの共存は許されないが，パーソナルネットワークは１つのホストにつき１つのポートスペースしか含まないため，パーソナルネットワークの中では IP アドレスの衝突の心配はない．

また，同一ホストのポートスペース同士は影響を与えあうことはできないので，ユーザは安全にポートスペースを使うことができる．ユーザがルーティングなどのネットワーク設定を間違えてしまった場合でも，他のポートスペースのネットワーク設定とは独立しているので，影響は１つのポートスペース内にとどめられる．そのため，最悪の場合でも，そのポートスペースを含むパーソナルネットワーク内にしか影響は及ばない．同様に，ポートスペース内で悪意のあるプログラムが実行されてしまったとしても，そこから他のポートスペースで動いているサーバプロセスを攻撃することはできない．これは，ポートスペース毎にポート空間が独立しており，別のポートスペースからは攻撃対象であるポートにアクセスできないためである．

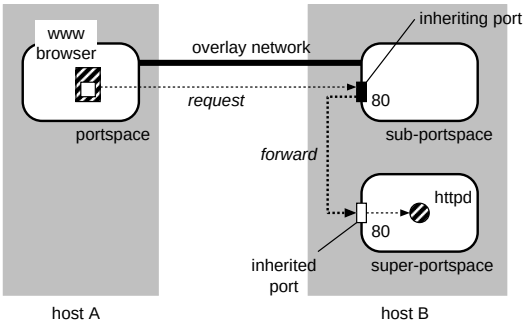

図 4 継承したウェブサービスへのリクエストの流れ
Fig. 4 Sending a request to an inherited web service

3.2.2 ポートスペースの継承

より簡単にポートスペースを作れるようにするために，ユーザは親ポートスペース（super-portspace）の状態を継承させて新しいポートスペースを作ることができる．ポートスペースの継承により，子ポートスペース（sub-portspace）はネットワーク設定や提供されているサービスを親ポートスペースから引き継ぐ．ポートスペースの管理する情報を参照する時には，まず，対象のポートスペースで検索を行い，見つからなければ親ポートスペースで検索する．これを見つかるまで繰り返し，最終的に見つからなければ検索に失敗する．例えばウェブサービスを継承する場合，継承したサービスがバインドされている子ポートスペースの 80 番ポートにリクエストを送ると，親ポートスペースの 80 番ポートに転送され，そのポートをバインドしているサーバプロセス（httpd）によって処理される（図 4）．この機構により，ユーザは親ポートスペースの状態を基に，必要に応じて子ポートスペースをカスタマイズできる．

子ポートスペースは親ポートスペースのネットワーク設定や提供されているサービスを上書きしたり，その一部を隠蔽することもできる．例えば，親ポートスペースの 80 番ポートでウェブサーバが動いていても，子ポートスペースの 80 番ポートで別のウェブサーバを動かすことができる．従来は別のウェブサーバを動かそうとすると別のポート番号を使わなければならなかったが，この機能により，ユーザはサービスを利用する時に常にデフォルトで割り当てられているポート番号を使うことができるようになり，直感的に分かりやすくなる．一方，親のポートスペースで提供されている POP サービスだけ継承し，その他のサービスは隠蔽することにより，POP によるメール配信に特化したポートスペースを作ることができる．このような専用ポートスペースを作ることは，万一攻撃された場合に親ポートスペースのサービスをできるだけ使わせないようにするのに役立つ．

3.2.3 ポートスペースの継承の安全性

ポートスペースを継承しても子ポートスペースが親ポートスペースの設定に影響を及ぼすことはできない．子ポートスペースで行った設定の変更は親ポートスペースには反映されないからである．同様に，子ポートスペースにおけるサービスの上書きや隠蔽も親ポートスペースには影響しない．逆に，親ポートスペースからは子ポートスペースが見えないので，子ポートスペースを作った後で影響を及ぼすことはできない．

一方，継承したサービスを提供しているサーバプロセスに脆弱性があった場合，子ポートスペースから親ポートスペースを攻撃できる可能性がある．攻撃者が子ポートスペースへの攻撃を成功させ，その上で，継承したサービスを提供している親ポートスペースのサーバプロセスへの攻撃に成功すれば，親ポートスペースにおける制御を奪うことが可能になる．しかし，子ポートスペースにはパーソナルネットワーク内からしかアクセスできないので，不特定多数の信頼できないユーザから攻撃を受ける可能性は低い．ただし，ユーザが子ポートスペースの中で悪意あるプログラムを実行してしまった場合には，攻撃の第 1 段階である子ポートスペースへの攻撃が成功したことになるので，この種の攻撃が成功する可能性が高まる．このようにあらかじめ危険が予測される処理を行う場合には，継承を用いずにポートスペースを作るか，必要最小限の脆弱性がないと考えられるサービスだけを継承するようにすべきである．

3.2.4 ファイルスペース

プライベートなネットワーク環境を実現するにはネットワーク空間の多重化だけでは不十分で，ファイルシステムも多重化する必要がある．この多重化されたファイルシステムを**ファイルスペース**と呼ぶ．各ファイルスペースは独立しており，ファイルスペース内のファイルやディレクトリは他のファイルスペースを利用しているプロセスからはアクセスできない．そのため，ユーザはファイルスペースを作って，DNS の設定ファイル（`/etc/resolv.conf`）やサーバの設定ファイルなどを自由に作成することができる．また，悪意を持ったプロセスがファイルシステムを介して他のポートスペースを攻撃するのを防ぐこともできる．

実用的に使えるファイルスペースを簡単に作れるようにするために，ユーザは親ファイルスペースを継承して子ファイルスペースを作ることができる．ファイルスペースを継承することにより，子ファイルスペー

スから親ファイルスペースのファイルを参照することができる．子ファイルスペースでファイルを変更した場合には，変更されたファイルは子ファイルスペースだけで参照することができる．子ファイルスペースで新しくファイルを作った場合やファイルを削除した場合も，親ファイルスペースには反映されない．この継承機能により，ユーザは新しいファイルスペースを作るたびに必要な実行ファイルやライブラリを用意して，一からファイルシステムを構築する必要がなくなる．

3.3 パーソナルネットワークの利用例

ユーザは所属する様々なネットワーク内部のウェブサーバ毎にパーソナルネットワークを作り，それぞれの内部情報の機密性を保ちつつ，1台のクライアントホストで閲覧することができる．それぞれのウェブサーバではウェブサービスを継承したポートスペースを作り，クライアントホストでは機密情報が漏れないようにポートスペースの継承を行わない．クライアントホストはこのようなパーソナルネットワークをいくつも保持するが，相互に分離されているので機密情報がパーソナルネットワーク間で漏れることはない．たとえファイルに保存したとしても，独立したファイルスペースのために他のプライベートなネットワーク環境内のプロセスからは読むことができない．

ユーザは自分のホストとメールサーバだけからなるパーソナルネットワークを作り，安全にメールを読むことができる．近年，メールにウィルスが添付されることが多くなっており，万一に備えてメールはウィルスの被害が拡がらない環境で読んだ方がよい．そのため，メールサーバのポートスペースではPOPサービスだけを継承し，他のサービスは隠蔽する．クライアントホストではポートスペースの継承は行わない．このようなパーソナルネットワークの中でメールを読めば，万一ウィルスに感染しても継承したPOPサーバに脆弱性がない限り，パーソナルネットワークの外に被害が拡がることはない．

また，ユーザは自分のホームディレクトリが存在する複数のホストで構成されるパーソナルネットワークを作り，自分の手元のホストでNFSマウントしてファイルを参照することができる．各ホストのポートスペースでは自分専用のNFSサーバを動かし，自分のホームディレクトリだけをエクスポートする．手元のホストのポートスペースではエクスポートされたホームディレクトリをNFSマウントする．このようなパーソナルネットワークを構築することで，一般のユーザでも簡単に安全なファイル共有ができるようになる．

4. 実　　装

パーソナルネットワークはIPsecを用いたオーバレイネットワークと，プライベートなネットワーク環境を実現するポートスペースとファイルスペースから成る．我々はポートスペースとファイルスペースを提供するPersona OSをFreeBSD 4.5をベースにして開発した．

4.1 ポートスペース

ポートスペースは多重化されたネットワーク空間を作るために，OSカーネル内の以下のようなデータベースを管理する．

- プロトコルコントロールブロック（PCB）・リスト
- IPsecセキュリティポリシー・データベース（SPD）
- IPsecセキュリティアソシエーション・データベース（SAD）
- ファイアウォール・ルール
- ルーティングテーブル

PCBリストはTCPやUDPなどのトランスポートプロトコル毎に作られ，ポートへのサービスのバインドなどを管理する．IPsecのSPDはどの2点間での通信にIPsecを使うかというポリシーを管理し，SADは確立された通信路を管理する．

ユーザは`mkportspace`システムコールを用いてポートスペースを作成することができる．このシステムコールを発行したプロセスとその子孫のプロセスが作成されたポートスペースに所属する．ポートスペースに対してIPsecの設定を行うことで，IPsecの通信路を介して他のポートスペースと結合され，パーソナルネットワークを形成する．

ポートスペース間の通信は図5のように行われる．プロセスがパケットを送信するシステムコールを発行すると，カーネルはそのプロセスに対応するポートスペースが保持するIPsecのSPDを検索する．IPsec通信を許可するセキュリティポリシーが見つかれば，確立されているIPsecの通信路を使ってパケットを送信する．受信側のカーネルがパケットを受け取ったら，配送すべきポートスペースを検索する．この検索はパケットのIPsecヘッダに格納されているセキュリティパラメータインデックス（SPI）をキーとして行われる．ポートスペースが見つかったら，そのポートスペースが管理しているPCBリストを使って受け取るべきソケットを検索し，見つかればパケットの配送が完了する．

4.2 ポートスペースの継承機構

サービスを継承する場合，図4のように子ポートス

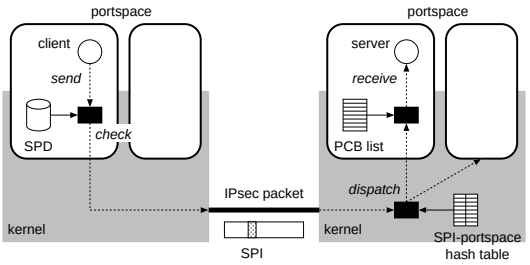

図 5 ポートスペース間の通信
Fig. 5 Communication between two portspaces

ペースのクライアントプロセスと親ポートスペースのサーバプロセスの間で TCP コネクションを確立することになるが，その時にサーバ側で作られるソケットは親ポートスペースではなく，子ポートスペースに属する．これは以降のクライアントとの通信を，子ポートスペースの通信路を使って行えるようにするためである．ソケットを親ポートスペースに属させると，親ポートスペースの通信路を使ってしまいうまく通信できない．

ファイアウォールのルールを継承する場合，デフォルトのルールに必ずマッチしてしまい，親ポートスペースのルールを適用できないので，デフォルトのルールが「許可」の場合は親のルールを検索するようにする．「許可」をデフォルトのルールとして適用したい場合は，ファイアウォールのルールを継承しないようにすればよい．ルーティングテーブルを継承する場合も同様に，デフォルトルート以外にマッチしなければ，親のルーティングテーブルを検索する．

4.3 ファイルスペース

ファイルスペースは BSD のユニオンファイルシステムを利用して実装した．子ファイルスペースでは /をマウントポイントとして，親ファイルスペースの/.filespace/<id>/（<id>はファイルスペースの ID）を透過的にマウントする．読み出しは親ファイルスペースから行われ，書き込みは子ファイルスペースに行われる．従来のマウント機構では全てのプロセスに同じマウント状態を見せることしかできないので，我々は利用しているファイルスペースに応じてマウント先を切り替えられるようにした．ファイルスペースの継承を行わない場合には，ユニオンマウントの代わりに通常のマウントを行い，親ファイルスペースは見えないようにする．現在の実装では，ファイルスペースは `mkportspace` システムコールを発行した時に同時に作られ，システムコールを発行したプロセスのカレントディレクトリやオープンしているファイルの参照先を新しいファイルシステム上に変更する．

表 1 ネットワーク別の TCP/IP の往復のレイテンシ
Table 1 TCP/IP round-trip latency by network

	μsec
ベースネットワーク（IPsec 使用）	131
パーソナルネットワーク	131
パーソナルネットワーク（継承あり）	131

5. 実　　験

我々の開発した FreeBSD 4.5 ベースの Persona OS を用いて，パーソナルネットワークのオーバヘッドを調べる実験を行った．実験には PentiumIII-S 1.4GHz の CPU，512MB のメモリ，Intel Pro/100+の NIC を搭載した PC を 2 台使用し，スイッチを介して 100baseT のイーサネットで接続した．IPsec はトランスポートモードで動作させ，ESP プロトコルを利用した．暗号化と認証のオーバヘッドの影響を最小にして測定するために，ESP プロトコルは NULL 暗号化アルゴリズムと NULL 認証アルゴリズムを用いた．

用意した 2 台の PC 間で，（1）ベースネットワークで IPsec を用いた場合について，（2）パーソナルネットワークについて，（3）パーソナルネットワークでサービスを継承した場合について，TCP/IP の往復のレイテンシを測定した．この実験結果は表 1 のようになった．この結果から，通信の際にポートスペースのオーバヘッドはほとんどないことが分かる．また，サービスを継承することによるオーバヘッドもほとんどなかった．

6. 関 連 研 究

オーバレイネットワークに関する研究には，IP の新しい機能を使うための M-Bone[4] や 6-Bone[5]，賢いルーティングを行う Detour[6] や RON[1] などがある．これらの研究はネットワーク管理者が設定し，ホストはオーバレイネットワークだけを使うか，ベースネットワークと異なる IP アドレスで使うことを想定している．それに対し，パーソナルネットワークではユーザがオーバレイネットワークを設定し，同じ IP アドレスで使い分けることができる．一方，複数のオーバレイネットワークを動的に作る研究には，X-Bone[7] や Genesis Kernel[2] などがある．これらの研究は，オーバレイネットワークのトポロジや，ルータにおけるルーティングや QoS が主な対象であり，エンドホストにおけるオーバレイネットワークの使い分けについては考慮されていない．我々のパーソナルネットワークはこれらの研究を補完するものである．

パーソナル VPN[8] は VPN の構築およびその利用

を特定のユーザに限定する．パーソナルネットワークとの違いは，パーソナル VPN はルータなどの中間ホストとの間に VPN を張り，複数の VPN を結合して目的のホストとの間に仮想ネットワークを構築できる点である．パーソナルネットワークは目的のホスト間でエンド・ツー・エンドの仮想ネットワークを構築することを基本としている．パーソナルネットワークでもオーバレイネットワークの構築の仕方次第で同様のことができるが，むしろ，エンドホストにおけるリソース管理を研究の対象としている．また，リソーススペース[9] もネットワークを多重化して使い分けを強制できるが，ネットワーク機器のサポートが必要な VLAN を使って，ネットワーク管理者が行うことを前提にしている点がパーソナルネットワークとは異なる．しかし，ホストが複数のネットワークを扱う場合の安全性については，パーソナルネットワークと問題を共有している．

プライベートなネットワーク環境を構築する方法として，仮想的な環境を構築する様々な手法が存在する．FreeBSD の `jail` などは特定のプロセスを異なる IP アドレスと専用ファイルシステムの下で動かすことができる．さらに仮想化を進めたものに，User Mode Linux[3] などの仮想 OS やマイクロカーネル上の OS サーバがあり，完全に独立したリソース管理の下で仮想的なプロセスを動かすことができる．また，VMware のような仮想マシンはハードウェアのレベルから独立した環境を作る．これらの手法に共通するのは安全性は非常に高いが，利便性が低いという点である．どの手法も一からネットワークとファイルシステムの設定を必要とする．我々の開発したポートスペースとファイルスペースは継承を用いなければこれらの手法とほぼ同じであるが，継承により一般のユーザでも簡単に設定を行えるようになっている．また，パーソナルネットワークという利用法に特化することにより，どの仮想環境でも同じ IP アドレスを使うことができる．

7. まとめと今後の課題

本稿ではユーザが複数のオーバレイネットワークを容易にかつ安全に使い分けられるようにしたパーソナルネットワークを提案した．パーソナルネットワークは各ホストでプライベートなネットワーク環境を用いることにより，オーバレイネットワークを完全に独立させる．我々はプライベートなネットワーク環境を実現するために，ネットワーク空間を多重化するポートスペースとファイルシステムを多重化するファイルスペースを開発した．

完全にプライベートなネットワーク環境を実現するためには，ネットワーク空間やファイルシステム以外の様々な事柄について考える必要がある．例えば，プライベートなネットワーク環境を作った一般のユーザが，その中で管理者権限を必要とするサーバを立ち上げるためには，ユーザ ID やファイルへのアクセス権などをうまく制御しなければならない．また，サービスの継承によりサーバが複数のパーソナルネットワークから使われる場合には，プロセス内部のメモリなどのリソースについて従来とは異なる管理が必要になる．このような問題を解決するには，プロセスに対するリソース管理の根本的な見直しが必要である．

参 考 文 献

1) Andersen, D., Balakrishnan, H., Kaashoek, F. and Morris, R.: Resilient Overlay Networks, *Proceedings of the 18th ACM Symposium on Operating Systems Principles*, pp.131–145 (2001).
2) Campbell, A. T., Meer, H. G. D., Kounavis, M. E., Miki, K., Vicente, J. and Villela, D. A.: The Genesis Kernel: A Virtual Network Operating System for Spawning Network Architectures, *Proceedings of the 2nd IEEE International Conference on Open Architectures and Network Programming* (1999).
3) Dike, J.: User Mode Linux, `http://user-mode-linux.sourceforge.net/`.
4) Eriksson, H.: Mbone: The Multicast Backbone, *Communications of the ACM*, Vol. 37, No. 8, pp. 54–60 (1994).
5) Guardini, I., Fasano, P. and Girardi, G.: IPv6 Operational Experience within the 6bone, *Proceedings of Internet Society Conference* (2000).
6) Savage, S., Anderson, T., Aggarwal, A., Becker, D., Cardwell, N., Collins, A., Hoffman, E., Snell, J., Vahdat, A., Voelker, G. and Zahorjan, J.: Detour: A Case for Informed Internet Routing and Transport, *IEEE Micro*, Vol. 19, No. 1, pp. 50–59 (1999).
7) Touch, J. and Hotz, S.: The X-Bone, *Proceedings of the 3rd Global Internet Mini-Conference at Globecom'98*, pp. 75–83 (1998).
8) 宇崎央泰, 千葉滋, 光来健一: 特定のユーザーだけが利用可能な仮想プライベートネットワーク, 日本ソフトウェア科学会第 19 回大会 論文集 (2002).
9) 廣津登志夫, 福田健介, 明石修, 佐藤孝治, 山崎憲一, 菅原俊治: 仮想データリンクを用いた多重通信クラスに関する一考察, 第 3 回インターネットテクノロジーワークショップ (2000).