iSCSIを利用したシンクライアントPCシステムSTRAGEX

市川俊一[†] 岡順一[†] 鷲坂光一[†]

個人情報保護法の施行を背景に，企業や自治体において重要情報の漏洩防止対策が課題となっている．この課題にシンクライアントシステムが有効であるが，既存システムは効果的なデータの一元管理と高スケーラビリティを両立できていない．そこで，我々は新たにオープンなストレージプロトコルであるiSCSIを基盤とした，機能分離型のアーキテクチャを持つシンクライアントPCシステムを提案する．提案システムはマルチOSに対応し，OS，アプリケーションとユーザデータの効果的な一元管理を実現する．また，我々はクライアントPCのOSとしてWindowsXPとRedHat Linuxを対象に提案システムを実装し，評価を行った．そして，提案システムが，100台程度のクライアントPCを収容しても，実用的な時間で処理できる高いスケーラビリティを有していることを確認した．

STRAGEX: iSCSI-based Thin Client PCs System

Toshikazu Ichikawa,[†] Junichi Oka[†] and Mitsukazu Washisaka[†]

Against the background of the law protecting personal information, the prevention of information leak is an important issue for corporations and governments. Although thin client system has an advantage over this issue, existing systems fail to achieve both an effective consolidation of data and a high scalability of system. We propose the thin client PCs system which is based on iSCSI as an open storage protocol and is designed to a distributed architecture. Our system is capable of running multi operating system, and achieves an effective consolidation of OS, application and user data. We implemented our system for WindowsXP and RedHat Linux as a client PC's OS, and evaluated it. The evaluation shows that our system has the high scalability to handle about a hundred client PCs within practical time.

1. はじめに

近年，企業や自治体の保持する顧客情報などの個人情報の流出が問題となっている．さらに2005年4月に個人情報保護法とe文書法が完全施行され，情報漏洩を防ぐための取り組みが企業や自治体などで行われている．顧客情報の漏洩は企業イメージの失墜を招き，企業活動にも大きな影響を与えるため，顧客情報の管理は情報システムの重要な課題となっている．

NPO日本ネットワークセキュリティ協会の調査[1]によると，個人情報の漏洩原因として順に，盗難36.1%，紛失・置忘れ21.6%，続いて誤操作10.7%となっており，盗難，紛失・置忘れという物理的な要因が，57.7%を占めている．また，ガートナーの調査[2]によると，54%のユーザがPCに個人情報を保有しており，営業・販売職に限ればその比率は73%にも達する．つまり，重要な多くの顧客情報がクライアントPCに保存されていることが情報漏洩のリスクを高めている．さらに，Dantz Development Corporationの調査[3]によると，企業が所有するデータの90%以上がサーバではなくPCに保存されており，約85%の企業がPCのデータをほとんどバックアップしていない．

すなわち，クライアントPCに重要な情報が保存されている昨今のIT環境においては，

- 重要情報の漏洩防止
- セキュリティ水準の維持徹底
- 故障や災害による業務停止の防止

という3つの要求を，ユーザの利便性を損ねず，また運用コストを抑えて実現するシステムが求められている．データの暗号化，サーバへのバックアップ，常駐監視プログラムの導入といった手法の組合せでは，利便性を保ちつつ運用コストを抑えることは難しい．

そこで，我々はその実現方法としてOS，アプリケーションとユーザデータをサーバ側に保存し一元管理できるシンクライアントシステムに注目した．そして，新たにオープンなストレージプロトコルであるiSCSI[4]を基盤とした機能分離型のアーキテクチャを持つシン

† 日本電信電話株式会社NTT情報流通プラットフォーム研究所
NTT Information Sharing Platform Laboratories, NTT Corporation

表 1 ディスクレスの実現方法
Table 1 Method of diskless.

	画面転送方式	ネットワークブート方式
特徴	画面イメージ・キーボード・マウスなどの入出力情報をサーバとクライアントで交換	専用サーバやストレージサーバが提供する論理ディスクを使ってクライアント PC を起動
ネットワーク	数十 kbps の帯域が必要	数十 Mbps の帯域が必要
クライアント用 OS	サーバの CPU で動作	クライアント PC の CPU で動作
描画性能	マルチメディアに不向き	マルチメディアに対応
ユーザビリティ	処理性能のゆらぎが直接影響	非シンクライアントと同等の安定感
クライアント PC	ディスクレスとするには専用 PC が必要	汎用 PC を活用可能
互換性	画面転送プロトコルに依存	クライアント PC のデバイスに依存

表 2 クライアント用 OS とアプリケーションの管理方法
Table 2 Management of client OS and application.

	CPU・ディスク占有モデル	CPU・ディスク共有モデル	CPU 占有・ディスク共有モデル
特徴	クライアント PC ごとに独立のリソースをサーバ側に準備	複数のユーザが同時にサーバにログインして利用	同一パーティションから複数のクライアント PC を同時に起動
対応方式	画面転送方式とネットワークブート方式	画面転送方式のみ	ネットワークブート方式のみ
データ管理	一元管理不可	一元管理可能	一元管理可能
データ保存	通常の PC と同じ	通常の PC と同じ	再起動で失われるため別の場所に保持する仕組みが必要
クライアント OS の更新	制約なし	更新対象サーバを利用しているクライアント PC の停止が必要	更新対象ディスクを利用しているクライアント PC の停止か必要
処理性能	リソースを柔軟に拡張可能	アプリケーションサーバがボトルネック	ストレージサーバがボトルネック

表 3 ユーザ情報とユーザデータの管理方法
Table 3 Management of user information and user data.

	OS 標準方式	外部サーバ方式	OS 連携サーバ方式
特徴	OS が提供する標準的なユーザ情報とユーザデータの管理	ユーザデータをストレージサーバに保存	ユーザ情報を OS のログイン処理と連携するサーバに保存
データ管理（CPU・ディスク共有モデル）	クライアント PC ごとにデータが散在し，一元管理不可	サーバで一元管理可能	サーバで一元管理可能
データ管理（CPU・ディスク占有モデル）	サーバで一元管理可能	サーバで一元管理可能	サーバで一元管理可能
データ管理（CPU 占有・ディスク共有モデル）	再起動でデータが消失	サーバで一元管理可能	サーバで一元管理可能

クライアントシステムを実現することで，既存システムにあったスケーラビリティの問題を改善し，特定の OS に依存せずに上記 3 つの要求を達成した．本稿では，2 章で既存のシンクライアントシステムとその構成技術についてまとめる．3 章で我々のシステムを提案する．3.1 節でそのアーキテクチャについて示し，3.2 節で Microsoft 社の WindowsXP を対象にした実装について述べる．また 4 章で，提案システムを実環境で評価する．さらに 5 章で，RedHat Linux を対象にした実装について述べ，評価を加える．最後に 6 章でまとめと今後の課題を述べる．

2. 従来手法

本章ではシンクライアントシステムを，ディスクレスの実現方法，クライアント用 OS とアプリケーションの管理方法，および，ユーザ情報とユーザデータの管理方法の 3 つの観点からまとめる．

ディスクレス PC を実現する方法は，画面転送方式とネットワークブート方式に分類[5] できる．その特徴を**表 1** にまとめる．画面転送方式を実現する手段として，VNC[6] や MetaFrame[7] などがある．また，ネットワークブート方式を実現する手段には，iSCSI HBA などのハードウェアで実現する方法と，NIC の PXE[8] 機能などを用いソフトウェアで実現する方法がある．ソフトウェアで実現する方法には，NFS サーバから Linux をブートする方法[9),10)] iSCSI ストレージから Linux をブートする方法[11),12)]，iSCSI ストレージから Windows をブートする iNBP[13),14)]，専用サーバから Windows をブートする Ardence[15)~17)] などがある．

クライアント用 OS とアプリケーションの管理方法は，CPU・ディスク占有モデル，CPU・ディスク共有

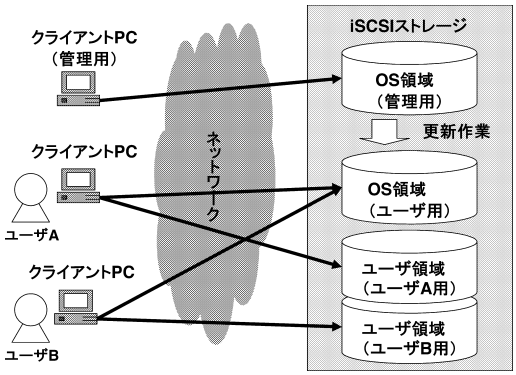

図 1 STRAGEX のアーキテクチャ
Fig. 1 Architecture of STRAGEX.

モデルと CPU 占有・ディスク共有モデルの 3 つに分類できる．その特徴を**表 2** にまとめる．CPU・ディスク共有モデルを実現するシステムとして，MetaFrame がある．CPU 占有・ディスク共有モデルを実現するシステムとして，Ardence の Shared Image Mode がある．前記したその他のディスクレス実現方法を用いた一般的なシステムは，CPU・ディスク占有モデルである．

ユーザ情報とユーザデータを管理する方法は，OS 標準方式，外部サーバ方式と OS 連携サーバ方式の 3 つに分類できる．その特徴を**表 3** にまとめる．外部サーバ方式には，Linux では NFS が，Windows では CIFS がある．OS 連携サーバ方式には，Linux では NIS[18] が，Windows では IntelliMirror[19] がある．外部サーバ方式と OS 連携サーバ方式は組み合わせて使うことができる．

3. 提 案 手 法

3.1 アーキテクチャ

従来のシステムは，サーバに機能と処理が集中する形でデータの一元管理を実現していたため，システムのスケーラビリティが問題であった．そこで，我々はオープンなストレージプロトコルである iSCSI を基盤とした機能分離型のアーキテクチャを特徴とするシステムを提案する．提案システムを以下，STRAGEX と呼ぶ．STRAGEX のアーキテクチャを**図 1** に示し[20)~22)]，その特徴について以下に述べる．

3.1.1 ディスクレス実現方法

画面転送方式は専用サーバの処理能力，すなわち CPU がシステムスケーラビリティのボトルネックとなる．一方，ネットワークブート方式はストレージの I/O とネットワークのスループットがボトルネックとなる．一般的に，CPU の処理能力はムーアの法則に従い，18 カ月で 2 倍になるのに対し，ネットワークの帯域はギルダーの法則に従い，6 カ月で 2 倍になると見込まれている．そこで，我々は今後もスケーラビリティに高い改善が見込まれる後者のネットワークブート方式を用いることとした．

また，ネットワークブート方式の中では iSCSI によるディスクレス実現方法を採用した．NFS サーバによる方法[9),10)] と比較すると，iSCSI を用いることでサーバ側に高い処理性能[23),24)] が期待できるうえ，ファイルシステムに依存しないため複数種類のクライアント用 OS にも対応できる．Ardence[15)~17)] と比較すると，Ardence はサーバとの通信を UDP ベースの独自プロトコルで実現している．そのためディスク I/O に対する専用サーバの CPU 処理能力がシステムスケーラビリティのボトルネックとなるが，iSCSI を用いることで TCP Offload Engine（TOE）などのハードウェア技術が活用でき，高い処理性能が期待できる．また，TCP であるため損失や遅延のある広域ネットワークに対応できる．

Ardence が専用サーバで独自の仮想ディスクを実現する集中型のアーキテクチャであるのに対し，STRAGEX は iSCSI ストレージを用いた機能分離型のアーキテクチャを特徴とする．制御信号を専用サーバに，ディスク I/O を iSCSI ストレージに分離することで，従来のボトルネックであったサーバ負荷を軽減する．また，iSCSI というオープンなストレージプロトコルを使うことで，OS やアプリケーションを含めたデータのバックアップやリストア技術を STRAGEX に容易に組み合わせることを可能とする．

3.1.2 クライアント用 OS とアプリケーションの管理方法

我々はネットワークブート方式においてクライアント用 OS とアプリケーションの一元管理を実現するため，1 つのディスクから複数のクライアント PC を同時に起動可能な CPU 占有・ディスク共有モデルを採用した．

しかし，従来システム[15)~17)] では，クライアント用 OS やアプリケーションを更新するためには，更新対象となるディスクを利用しているクライアント PC をいったん停止させる必要があるという問題があった．そこで，我々は管理者が更新を行うためのディスクとクライアント PC が使うディスクを別々に準備しておき，利用中のクライアント PC を止めることなく次回の起動時に更新されたディスクを利用する方式（以下，世代管理方式と呼ぶ）を提案する．世代管理方式では更新作業後にディスクの複製を行い，管理者用ディス

クと同じ内容の別ディスクをクライアント PC 用として準備する．クライアント PC は次回起動時からそのディスクを使う．

3.1.3 ユーザ情報とユーザデータの管理方法

我々はネットワークブート方式の CPU 占有・ディスク共有モデルにおいて，ユーザ情報の一元管理を実現するため，従来システムと同様に，OS 連携サーバ方式を採用した．

また，従来システムはユーザデータの一元管理を実現するために NFS や CIFS などの外部サーバ方式を用いる．しかし，ファイルレイヤのプロトコルはブロックレイヤに比べ，ストレージサーバの処理能力がボトルネックになりやすい[23),24)]．

そこで，我々はユーザデータをクライアント用 OS やアプリケーション同様に，iSCSI ストレージに保存することとした．ユーザデータを含むファイルシステムをディスクに入れて，そのディスクをユーザごとに管理する．すべてのデータを iSCSI ストレージに格納することで，システムのバックアップを一元的に行うことも可能となる．以後，クライアント用 OS とアプリケーションを保存するためのディスクを OS 領域と，ユーザデータを保存するためのディスクをユーザ領域と呼ぶ．

STRAGEX はネットワークブート方式と外部サーバ方式に iSCSI を用いることを特徴とする世代管理を備えた CPU 占有・ディスク共有モデルである．

3.2 実　　装

3.2.1 システム構成

STRAGEX のシステム構成を**図 2** に示す．STRAGEX は次の 5 つの機能コンポーネントから構成される．

- クライアント PC 起動時に，PXE の仕様に従ってブートに用いる OS 領域に関する情報を与えるための STRAGEX PXE サーバ
- クライアント PC 起動後に，リアルモードで動作し，iSCSI ストレージに格納された OS を OS 領域からブートするための STRAGEX ブートローダ
- クライアント PC の OS 起動途中から，プロテクトモードで動作し，STRAGEX ブートローダが確立した OS 領域との iSCSI セッションを保持するための STRAGEX ドライバ
- クライアント PC の OS 起動後に，ユーザログイン処理で iSCSI ストレージに格納されたユーザ領域を提供するための STRAGEX クライアント
- クライアント PC，ユーザ，OS 領域などのシステム情報を管理する STRAGEX 制御サーバ

また，既存コンポーネントとして，クライアント PC，DHCP サーバ，TFTP サーバ，iSCSI ストレージと OS 連携サーバを組み合わせてシステムを構築した．

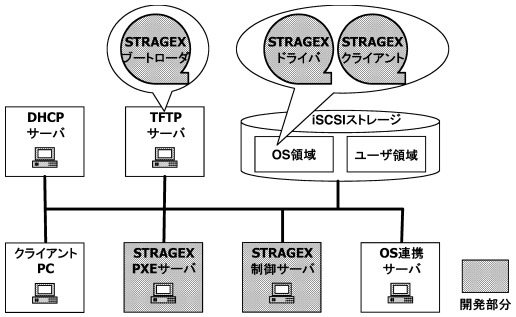

図 2 STRAGEX のシステム構成
Fig. 2 System composition of STRAGEX.

3.2.2 OS のブート処理

OS のブート処理の流れを**図 3** に示す．ブート処理は，以下の PXE，ブートローダとドライバの 3 つのフェーズからなる．

STRAGEX PXE サーバは，Java で動作するサーバプログラムである．STRAGEX PXE サーバは，PXE の仕様に従って，クライアント PC からの要求に対して OS 領域に関する情報を返す．OS 領域に関する情報には，iSCSI ターゲットの IP アドレス，ポート番号，ターゲット名，LU 番号，CHAP ユーザ，CHAP パスワードが含まれる．我々は，これらの情報を PXE の Vender Options としてベンダ依存仕様用に準備されたタグに定義して，機能を拡張した．また，クライアント PC が利用する OS 領域は STRAGEX 制御サーバで管理されており，STRAGEX PXE サーバは要求に応じて制御サーバに問合せを出す．さらに，PXE の仕様

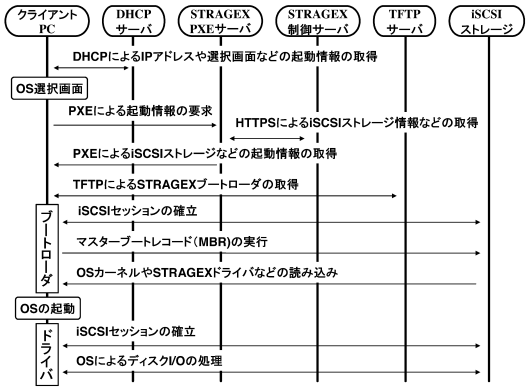

図 3 ブートシーケンス
Fig. 3 Boot sequence.

にあるPXE_BOOT_MENUとPXE_BOOT_ITEMを用いることで，クライアントPCの起動画面において，OS領域すなわちブートOSの選択を可能にしている．

STRAGEXブートローダは，ACPI BIOSとPXEに対応したIntel x86アーキテクチャPCのリアルモードで動作する．TFTPサーバに配置され，PXEの仕様に従ってクライアントPCにダウンロードされ，実行される．STRAGEXブートローダは，まずDHCPとPXEのOptionからiSCSIディスクなどの情報を取得し，iSCSIストレージとのセッションを確立する．次に，BIOSコールをフックすることで，1番目のハードディスクへの要求をiSCSIディスクへの要求として処理されるようにして，iSCSIディスクからの読み込みとディスク情報の取得を可能にする．最後に，iSCSIディスクからMBRを読み込み，iSCSIディスクに格納されているOSを起動させる．

STRAGEXドライバは，起動するOSに依存するコンポーネントであり，今回開発したプログラムは，WindowsXPでデバイスドライバとして動作する．SCSIミニポートドライバとして，NDISネットワークスタック上で動作する．STRAGEXドライバは，OS起動途中，プロテクトモードに切り替わった直後から動作し，STRAGEXブートローダが書き込んだiSCSIストレージなどの情報を取得するため，物理メモリの固定番地から読み出しを行う．そして，iSCSIストレージとのセッションを再確立する．OSからのSCSI Control Block要求（SCB）をiSCSIパケットに変換し，NDISドライバを介してiSCSIターゲットに送る．またiSCSIターゲットから受信したiSCSIパケットをSCBに変換し，OSに送る．OSが終了するまで稼動し続け，OS領域への読み書きを処理する．

また，CPU占有・ディスク共有モデルを実現するため，すなわち1つのディスクから複数のクライアントPCを同時に起動させるため，我々はSTRAGEXドライバに，iSCSIディスクへ書き込みを行わずにメモリに書き込みを行うWrite Bufferと呼ぶ機能を実装した．STRAGEXドライバは，Write Bufferの設定をレジストリに持ち，Write Bufferが有効な場合は，書き込みをメモリ上のバッファ領域にキャッシュしiSCSIディスクには送らない．また，読み出しについてもキャッシュがある場合はiSCSIディスクではなくバッファ領域の内容を読み込む．バッファ領域はブロック単位で読み書きされ，LBAをキーにしたツリー構造のインデックスで管理されている．

Ardenceはブートローダと専用サーバが直接通信を行うが，STRAGEXはブート処理のシステム制御をPXEサーバ経由で行う．すなわち，STRAGEXブートローダとSTRAGEXドライバだけを既存コンポーネントと組み合わせてWindowsXPをブートすることもできる．さらに，Write Bufferの機能がSTRAGEXドライバに閉じているため，サーバ側の機能に依存せず，任意のiSCSIストレージを対象にCPU占有・ディスク共有モデルを実現できる．

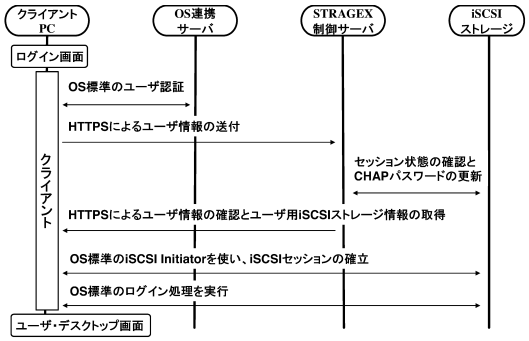

図4 ログインシーケンス
Fig. 4 Login sequence.

3.2.3 ユーザのログイン処理

ログイン処理の流れを**図4**に示す．ログイン処理は以下のSTRAGEXクライアントによって行われる．

STRAGEXクライアントは，起動するOSに依存したコンポーネントであり，今回開発したプログラムは，WindowsXPでGINA DLL[25]として動作する．

我々は，ユーザ情報の一元的な管理を実現するため，OS連携サーバとしてWindows Server 2003のActive Directory[19),26)]を利用した．Active DirectoryはKerberosを認証プロトコルとして用い，クライアントOSとサーバ間のセキュアな通信を実現する．そこで，我々はActive Directoryとクライアント OSに格納されているKerberos認証に用いられる共通鍵を必要に応じて一致させることで，CPU占有・ディスク共有モデルであるSTRAGEXにおいてActive Directoryを利用できるようにした．

STRAGEXクライアントはログイン要求を処理するため，まずOS標準の認証機構を呼び出し，ユーザの認証をActive Directoryで行う．次に，入力されたユーザ情報をもとに，STRAGEX制御サーバに対応するユーザ領域のiSCSIディスク情報を問い合わせる．そして，そのiSCSIディスク情報に基づいて，Microsoft iSCSI Initiatorを用いてiSCSIセッションを確立し，ユーザ領域がマウントされたファイルパスがユーザのホームディレクトリとなるようレジストリを編集する．最後に，OS標準のログイン処理機構を

呼び出し，ユーザをログインさせる．また，ログアウト時はログインと逆順に同様の処理を行う．

ユーザデータはユーザ領域に保存され，クライアント PC に iSCSI でマウントされる．もし 2 台以上のクライアント PC が同じユーザ領域に同時マウントした場合，ファイルシステムレイヤで不整合が生じ，データが失われる危険がある．そこで，我々は STRAGEX 制御サーバに，ユーザ領域への二重マウントを防止するための 2 つの機能を加えた．1 つ目は，クライアント PC からユーザ領域の問合せがあった場合に，そのユーザ領域の使用状態を検査し，使われていない場合に限ってログインを許可する機能である．2 つ目は，ネットワーク切断などにより iSCSI セッションを検査時に一時的に失っていたクライアント PC が，再試行などで接続してしまわないように，ユーザ領域の接続情報をクライアント PC に送るときに，CHAP パスワードを新しいものに変更する機能である．古い接続情報を保持しているクライアント PC が接続を試みても，認証エラーとなり iSCSI ストレージに接続できない．

3.2.4 クライアント PC，OS とユーザの管理

STRAGEX 制御サーバは，Java で動作するサーバプログラムである．ユーザ，クライアント PC，OS 領域とユーザ領域の管理を行い，システム全体を制御する．ユーザをユーザアカウントで，クライアント PC を MAC アドレスで，OS 領域とユーザ領域を iSCSI ターゲット名と LU 番号で識別する．これらの関係を**図 5** に示す．

クライアント PC と OS 領域は CPU 占有・ディスク共有モデルに対応するため N：N の関係で，ユーザとユーザ領域は 1：1 の関係を持つ．また，対象機種や用途に応じて同一 OS で複数の OS 領域を作成することが想定される．そこで，我々は OS 領域とユーザ領域の組合せを，OS 種別という概念を介して管理した．OS 領域とユーザ領域はどれか 1 つの OS 種別に属し，同じ OS 種別の範囲で組合せが可能とした．ユーザはその OS 種別に属する OS 領域が動作するどのクライアント PC からもログインでき，ユビキタスなログイン環境が提供される．

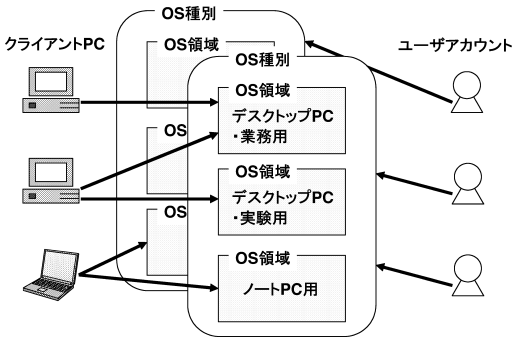

図 5 クライアント PC，OS とユーザの関係
Fig. 5 Relation between client PCs, OS and users.

STRAGEX 制御サーバは，クライアント PC を停止させず任意のタイミングで更新することを可能にする世代管理方式を実現するため，1 つの OS 領域に対して，待機系，運用系，バックアップ系の 3 つのディスクを準備する．待機系は管理用のクライアント PC が OS やアプリケーションを更新するためのマスタとなるディスクであり，運用系とバックアップ系は通常のクライアント PC が使うディスクである．更新作業後の OS 領域の切替えでは，待機系の複製を作り，これを新しい運用系にする．このとき，元の運用系がクライアント PC に使われている場合を考慮して，そのクライアント PC を停止させないために，元の運用系を新しいバックアップ系にする．クライアント PC が再起動した時点で，STRAGEX 制御サーバから新しい運用系がクライアント PC に提供され，クライアント PC は更新の反映された新しいディスクを使うことになる．

3.2.5 セキュリティ

iSCSI ストレージのデータは，CHAP 認証によるアクセス制御により秘匿性が保たれている．CHAP 認証用のパスワードは，ユーザがログイン認証用に使うパスワードとは独立であり，STRAGEX サーバが要求に応じて発行するワンタイムパスワードである．また，CPU 占有・ディスク共有モデルで，クライアント PC 間で共有されるディスクは Read-Only 属性となっており，他のクライアント PC による改ざんを防止している．

STRAGEX サーバと STRAGEX クライアント，STRAGEX サーバどうしの間の通信は，HTTPS プロトコル上の規定のフォーマットで行われ，暗号化による秘匿とサーバ認証による正当性確認が行われる．

4. 評　　価

4.1 試 験 環 境

STRAGEX を評価する．試験環境では，STRAGEX PXE サーバ，STRAGEX 制御サーバ，DHCP サーバと TFTP サーバを 1 台の PC 上に稼動させ（以下，STRAGEX サーバと呼ぶ），STRAGEX ドライバの Write Buffer 機能を有効にして測定した．クライアント PC は，通常の環境を想定して 100 Mbps で接続し，iSCSI ストレージは 1 Gbps の 3 ポートを並列して接

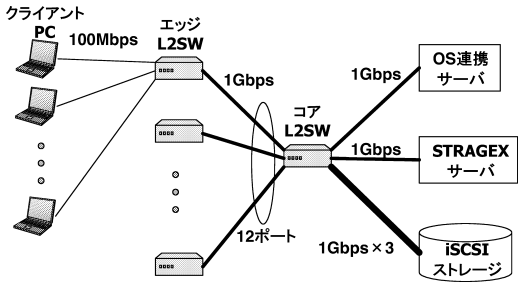

図 6 試験環境の構成
Fig. 6 Composition of evaluation environment.

表 4 試験環境の機器の仕様
Table 4 Equipment specification of evaluation environment.

コンポーネント	仕様
サーバ	HP DL 380 CPU: Intel Xeon 3.4 GHz Memory: 1 GB
iSCSI ストレージ	Equallogic PeerStorage 100E SATA250 GB×14, RAID50, firmware 2.2.2
コア L2SW	Cisco Catalyst 3750
エッジ L2SW	Cisco Catalyst 2950
クライアント PC	TOSHIBA dynabook J40 170L/5 CPU: Intel Pentium M 1.7 GHz Memory: 512 MB

続した．また，それ以外のリンクは 1 Gbps で接続し，中継ネットワークには十分な帯域を準備した．構築した試験環境のネットワーク構成を**図 6** に，使用した機種とソフトウェアの情報を**表 4** にまとめる．

4.2 OS 起動とログインによるトラフィック

クライアント PC を 1 台だけ動作させ，どの程度のトラフィックが発生するか計測した．一般的な PC の動作において，OS 起動とログイン処理で発生するトラフィックが圧倒的に多く，システム設計におけるボトルネックとなるため，この両者に注目した．また，スループットは iSCSI ストレージからの読み出しに相当する下り方向の流量が支配的であるため，下り方向の値を示す．クライアント PC の NIC が DHCP サーバへの通信を始めた時点を起点として，スループットの時間変化を**図 7** に示す．

5 秒付近にあるピーク（図中の (1)）は，STRAGEX ブートローダによるカーネルやドライバの読み込みであり，20 秒から 40 秒にかけてのピーク（図中の (2)）は，STRAGEX ドライバによるドライバやサービスの読み込みである．ログイン画面は，46 秒に表示されたが，60 秒付近（図中の (3)）に，OS 内部プログラムによる読み込みが発生しており，その後に大きなトラフィックは発生しなかった．また，83 秒にログイン入力を行ったところ，100 秒から 120 秒まで（図中の (4)）ログイン処理によるトラフィックが発生した．100 秒まで流量が少ないのは，その間に WindowsXP の認証処理が行われているためである．また，ユーザのデスクトップ画面は 111 秒に表示され，しばらくトラフィックが流れ続けたが，その後に大きなトラフィックは発生しなかった．

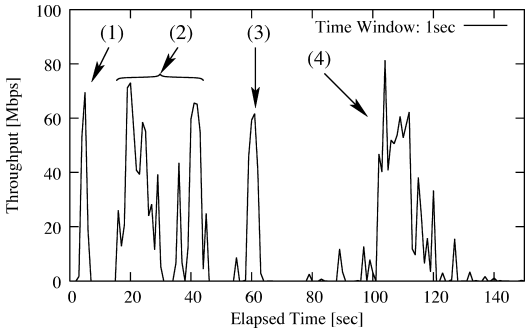

図 7 ネットワークトラフィックの時間変化（WindowsXP）
Fig. 7 Time variation of network traffic (WindowsXP).

クライアント PC の動作により，OS 起動とログイン処理でバースト的に大きなトラフィックが発生することが分かった．OS 起動でトータル上り 8 MB 下り 128 MB，ログイン処理で上り 4 MB 下り 78 MB のトラフィックが発生した．また，OS 起動時の Write Buffer のバッファ量は 10 MB 程度と少なく，Write Buffer の性能への影響はほとんどない．

4.3 スケーラビリティ

実際の利用場面を想定すると，大きなトラフィックがバースト的に発生するのは，OS 起動とログインの処理である．そこで，これらの処理を複数のクライアント PC で同時に実行し，処理にかかる時間を計測することで，システムのスケーラビリティを評価する．

OS 起動所要時間は電源投入からログイン画面表示までの時間，ログイン所要時間はアカウント情報入力からデスクトップ画面表示までの時間とした．クライアント PC を 1 台から 120 台まで変えて，各所要時間の平均値と最大値を計測した．なお，クライアント PC を増やすときは，トラフィックが分散するように可能な限り別のエッジ L2SW に収容した．また，STRAGEX は CPU 占有・ディスク共有モデルで動作させ，1 つの OS 領域から複数のクライアント PC を起動させた．結果を**図 8** に示す．

電源投入はマジックパケットを用いていっせいに行い，ログイン画面表示の時間は，クライアント PC からサーバに通知をするプログラムを用いて計測した．OS 起動所要時間にはクライアント PC の BIOS 起動

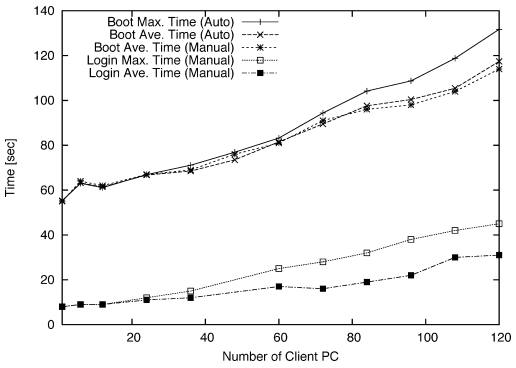

図 8 同時 OS 起動とログインの所要時間
Fig. 8 Time required for simultaneous OS boot and user login.

にかかる時間（約 12 秒）が含まれる．また，ログイン時のアカウント情報入力は，サーバからクライアント PC を操作するプログラムを用いていっせいに行った．デスクトップ画面表示の時間は，プログラムによる計測が困難であったため，ストップウォッチによる手動計測とした．なお，手動計測の精度の妥当性を検証するため，OS 起動所要時間についても手動計測（図中 Boot Ave. Time（Manual））を行い，プログラムによる自動計測（図中 Boot Ave. Time（Auto））と比較し，誤差がほとんどないことを確認した．

クライアント PC の台数が 1 台から 24 台にかけて，所要時間の平均値にわずかな増加傾向が見られるが，平均値と最大値に開きは見られない．iSCSI ストレージのインタフェースのスループットが 3 Gbps であり，各クライアント PC のスループットが 100 Mbps であるため，台数増加の影響がほとんどない．

クライアント PC の台数が 36 台以上になると，所要時間は増加し，平均値と最大値に開きが生じている．iSCSI ストレージのインタフェースのスループットがボトルネックになるためである．36 台の OS 起動とログインの所要時間と 72 台・108 台のそれを比較すると，単純に台数に比例した 2 倍・3 倍とはなっていない．図 7 に示したように発生するトラフィックはバースト的であり，つねに帯域を使いきっているわけではない．すなわち，バースト的なトラフィックが平滑化され，ボトルネックにおけるネットワークの利用効率は高まったためと考える．

クライアント PC の台数が 96 台を超えると，所要時間の増加傾向がさらに急になっている．この台数では，iSCSI ストレージのインタフェースのトラフィックがほぼ飽和していることが確認できた．すなわち，利用効率の増加が頭打ちになったためと考える．

1 台と 120 台の所要時間の平均値を比べると，OS 起動は 2 倍程度，ログイン処理は 4 倍程度となっている．OS 起動の方がトラフィックが間欠的であるため，平滑化効果が高い．また最大値の絶対値については，OS 起動は 2 分 30 秒以内，ログイン処理は 45 秒以内に収まっている．従来システム[17] は，OS 起動の平均所要時間を 2 分として設計する場合，50 台未満のクライアント PC しか収容することはできなかったが，STRAGEX は 120 台のクライアント PC を収容できる．STRAGEX が 100 台規模のクライアント PC を実用的な処理時間を達成しつつ収容できることが分かった．高いパフォーマンスを持つ iSCSI ストレージに I/O の負荷を分離して，I/O リクエストの平滑化による大群化効果を引き出すことで，システムの高いスケーラビリティが実現できることを検証できた．

5. マルチ OS への対応

前章まではクライアント PC で動作する OS を WindowsXP として述べた．本章では RedHat Linux を利用するための実装方法について述べ，その評価を行う．

現状，RedHat Linux を iSCSI ストレージからブートする方法として，主に PXELINUX[11),27)] と iNBP[12)] の 2 つがある．PXELINUX はカーネルを TFTP サーバからロードするが，STRAGEX では iNBP 同様に STRAGEX ブートローダが iSCSI ストレージの OS 領域より直接ロードする．そのため，WindowsXP と RedHat Linux の OS 領域を同じスキームで管理・利用することができる．STRAGEX ドライバには従来手法と同じく Linux iSCSI initiator[28)] を用い，unionfs[29)] を組み合わせることで Write Buffer に相当する機能をファイルレイヤで実現した．そして，STRAGEX クライアントを，PAM モジュールとして実装した．

4.1 節で示した試験環境において，RedHat9 をランレベル 3（X Window なし）でセットアップし，クライアント PC を 1 台だけ動作させ，そのトラフィックを計測した．クライアント PC の NIC が DHCP サーバへの通信を始めた時点を起点として，スループットの時間変化を**図 9** に示す．

4 秒付近にあるピーク（図中の (1)）は，STRAGEX ブートローダ動作環境下の GRUB による initrd などのカーネルの読み込みである．12 秒から Linux iSCSI Initiator が動作しているが，5 秒間（図中の (2)）は Discovery フェーズのため読み込み処理は進んでいない．17 秒から 46 秒にかけて（図中の (3)），カーネルモジュールやデーモンの読み込みが発生しており，ロ

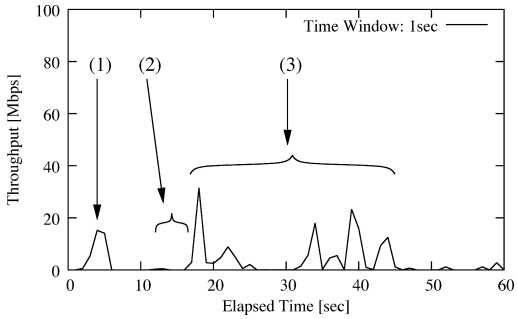

図 9 ネットワークトラフィックの時間変化（RedHat9）
Fig. 9 Time variation of network traffic (RedHat9).

グインプロンプトは，46 秒に表示された．

OS 起動でトータル上り 5 MB 下り 21 MB のトラフィックが発生した．また，ログイン処理ではほとんどトラフィックは発生しなかった．なお，ランレベル 5 として X Window を動作させると，WindowsXP に近いトラフィックが発生した．RedHat Linux についてもそのトラフィックは WindowsXP の場合と同等の量と性質を示しており，STRAGEX は高いスケーラビリティを達成できる．

6. ま と め

iSCSI を利用したネットワークブート方式による，マルチ OS に対応したシンクライアント PC システム STRAGEX を提案した．STRAGEX は，オープンなストレージプロトコルである iSCSI を基盤とした機能分離型のアーキテクチャを特徴とし，データの一元管理と高スケーラビリティを両立する．我々は，クライアント PC の OS として WindowsXP と RedHat Linux を対象に実装を行い，評価した．そして，STRAGEX が 100 台規模のクライアント PC を収容するのに十分なスケーラビリティを有していることを確認した．

本稿では，STRAGEX ドライバで OS 領域への書き込みをクライアント PC のメモリ上にバッファした．しかし，用途によってはクライアント PC に多くのメモリを搭載する必要があるという問題がある．今後の課題は，バッファ領域を iSCSI ストレージやローカル HDD に確保する機能を追加し，クライアント PC のメモリサイズに対する要求条件を緩和することである．

参 考 文 献

1) NPO 日本ネットワークセキュリティ協会：2004 年度情報セキュリティインシデントに関する調査報告書 (2005).
2) ガートナー：プレスリリース 2005 年 2 月 28 日 (2005).
3) Dantz Development Corporation Press Release, Sep. 10, 2003 (2003).
4) Satran, J., et al.: Internet Small Computer Systems Interface (iSCSI), RFC3720 (2004).
5) 我慢を強いないシンクライアント，日経バイト (Aug. 2005).
6) Richardson, T., et al.: Virtual Network Computing, *IEEE Internet Computing*, Vol.2, No.1, pp.33–38 (1998).
7) MetaFrame. http://www.citrix.com/
8) Intel Corporation: Preboot Execution Environment (PXE) Specification Version 2.1 (1999).
9) Goede, H.: Root over nfs clients and server Howto (1999). http://www.tldp.org/HOWTO/Diskless-root-NFS-HOWTO.html
10) Nobuhara, S.: Diskless Linux by PXELinux or GRUB (2002). http://vision.kuee.kyoto-u.ac.jp/ nob/doc/diskless/diskless.pdf
11) Bolen, B.: iSCSI-Root mini-HOWTO (2004). http://eludicate.com/ bolen/iscsi/
12) Cisco Systems Inc.: iSCI Remote Boot with Linux EDCS# 378954 San Jose, CA (2004).
13) Cisco Systems Inc.: Cisco Network Boot Installation and Configuration Guide, San Jose, CA (2004).
14) Cisco Systems Inc.: Release Notes for Cisco iSCSI Driver Version 4.2.1 for Microsoft Windows, San Jose, CA (2005).
15) Ardence. http://www.ardence.com/
16) 山口光大ほか：VID を使った diskless Windows，情報処理学会誌，Vol.45, No.3 (2004).
17) 吉田宏一ほか：ディスクレス Windows 端末の問題点と改善，情報処理学会「分散システム/インターネット運用技術」研究報告，No.30, pp.35–40 (2003).
18) Weiss, P.: Yellow Pages Protocol Specification, Sun Microsystems Inc. (1985).
19) Microsoft Corporation: Microsoft Windows 2000 Server, Introduction to IntelliMirror Management Technologies, White Paper (1999).
20) 三栄武ほか：ストレージセントリックネットワーク技術，NTT 技術ジャーナル，Vol.16, No.5, pp.44–46 (2004).
21) 野口清広ほか：ストレージセントリックネットワークシステム V1 の開発，NTT 技術ジャーナル，Vol.17, No.5, pp.42–45 (2005).
22) ディスクレス PC と大容量ストレージを使ったセキュリティ管理システム — STRAGEX，月刊ビジネスコミュニケーション (2006). http://www.bcm.co.jp/site/rd-pro/rdpro02.pdf
23) Radkov, P., et al.: A Performance Compari-

son of NFS and iSCSI for IP-Networked Storage, *Proc. 3rd USENIX Conference on File and Storage Technologies*, pp.101–114 (2004).
24) TechnoMages, Inc.: Performance Comparison of iSCSI and NFS IP Storage protocols, Technical report.
25) Brown, K.: Security Briefs: Customizing GINA, Part 1, *MSDN Magazine*, Vol.20, No.5 (2005).
26) Boswell, W.: *Inside Windows Server 2003*, Addison-Wesley (2003).
27) Anvin, H. P.: PXELINUX. http://syslinux.zytor.com/pxe.php
28) Linux-iSCSI Project. http://linux-iscsi.sourcefourge.net/
29) Wright, C.P. and Zadok, E.: Unionfs: Bringing File Systems Together, *Linux Journal*, No.128, pp.24–29 (2004).

(平成 18 年 1 月 27 日受付)
(平成 18 年 5 月 8 日採録)

市川 俊一（正会員）

2000 年早稲田大学理工学部電子・情報通信学科卒業．2002 年同大学院理工学研究科修士課程修了．同年日本電信電話（株）入社．ネットワークストレージの研究開発に従事．電子情報通信学会，日本データベース学会各会員．

岡 順一

1996 年早稲田大学理工学部機械工学科卒業．1998 年同大学院理工学研究科修士課程修了．同年日本電信電話（株）入社．ネットワークストレージの研究開発に従事．電子情報通信学会会員．

鷲坂 光一（正会員）

1985 年大阪大学工学部情報工学科卒業．1987 年同大学院修士課程修了．同年 NTT 入社．以来，広域 IP 網の研究・開発に従事．現在，NTT 情報流通プラットフォーム研究所主任研究員．