社団法人 情報処理学会 研究報告
IPSJ SIG Technical Reports

2007－ＣＳＥＣ－38（44）
2007／7／20

シンクライアントアーキテクチャをベースにした セキュアクライアントの検討

宮本 久仁男† 田中 英彦†

†情報セキュリティ大学院大学 〒221-0835 神奈川県横浜市神奈川区鶴屋町 2-14-1

E-mail: †{dgs064103, tanaka}@iisec.ac.jp

あらまし 現状、情報漏洩をはじめとする多くのセキュリティ上のリスクを回避するために、シンクライアントアーキテクチャをベースにした端末システムが使われている。しかし、もともとシンクライアントは、端末の総所有コスト（TCO）を低減させることを目的に開発されたものであり、全てのセキュリティリスクを回避できるというものではない。本論文では、シンクライアントに対する脅威の可能性とその理由、そしてこれらに対抗するための新たなセキュアクライアントの構造についての検討結果について述べる。加えて本論文では、セキュアクライアントの実装および評価計画、そして現状の検討結果に対する課題についても同時に述べることとする。

キーワード シンクライアント、セキュアクライアント、マルウェア、脆弱性、仮想マシン

Examination for secure client architecture based on thin client architecture

Kunio MIYAMOTO† and Hidehiko TANAKA†

† Institute of Information Security 2-14-1 Tsuruya-Cho Kanagawa-ku, Yokohama-City, Kanagawa, 221-0835 Japan

E-mail: †{dgs064103, tanaka}@iisec.ac.jp

Abstract Now, for preventing many of security risk, especially information leak, terminal system based on thin client architecture is used. But original purpose of thin client architecture is for reducing Total Cost of Ownership (TCO), and not useful to every security risk prevention. In this paper, we explain about probably of threats against thin client architecture, these reason, and our examinations about new secure terminal architecture against these issues. In addition, secure client implementation and evaluation plan and our issue thinking is explained in thes paper.

Keyword Thin Client, Secure Client, Malware, Vulnerability, Virtual Machine

1. システムにおける情報漏洩対策

昨今、情報システムにおいて、セキュリティ対策は喫緊の課題になっている。特に、情報漏洩については、あらゆる団体について対処されなければならないレベルの課題になっている。ただ、情報漏洩対策のみ行われるということは少なく、情報セキュリティに関連した対策を検討・実施する段階で優先度を高めに設定される項目の中に、情報漏洩対策があることが多い。

目的や対策を施すレイヤによって、あらゆるレベルのセキュリティ対策が実施されているが、システムを構成するプラットフォームとして「サーバ」「クライアント」「ネットワーク」と３つに分類した場合、例えば以下のような対策を行っている。

1.1. サーバに対するセキュリティ対策

ユーザなどにサービスを提供する、いわゆるサーバに対するセキュリティ対策としては、例えば以下のような対策を適用し、内外からの脅威に対抗している。

- 動作させるプログラムを、サービスに必要なもののみに絞り、侵入される箇所を減らす
- サーバ OS そのものを最小構成にし、侵入される箇所を減らす
- セキュア OS を導入し、侵入者に対する権限を与えないようにする
- Host-based Intrusion Detection System(HIDS)などを導入し、サーバへの侵入・プログラム等の改ざんがあった場合にそれを検知できるようにする
- 動作させるソフトウェアについて自動更新を設定し、ソフトウェアの脆弱性に関連した修正の適用漏れをなくす

1.2. クライアントに対するセキュリティ対策

ユーザが直接操作する、いわゆるクライアントに適用するセキュリティ対策としては、例えば以下のようなものがある。

- ウィルス対策ソフトウェアを導入し、既知の不正プログラムからクライアントを守る
- パーソナルファイアウォールを導入し、外部からの侵入行為に対抗する
- 機器接続制限用ソフトウェアを導入し、信頼できないデータを不用意に持ち込まれたり、内部で保持しているデータを持ち出されたりしないように備える
- ソフトウェアの自動更新を設定し、ソフトウェアの脆弱性に関連した修正の適用漏れをなくす
- 端末管理用ソフトウェアを導入し、端末の仕様を一様にして、対策を行いやすい環境を整える

1.3. ネットワークに対するセキュリティ対策

サーバとクライアントを接続し、データをやりとりするために利用されるネットワークに対するセキュリティ対策としては、例えば以下のような対策を適用し、内外からの脅威に対抗している。

- 各種ファイアウォールを導入し、内部／外部からの攻撃が攻撃対象に到達しないようにする
- Network Intrusion Detection System(NIDS)、Intrusion Detection and Prevention system,(IDP)、Unified Threat Management(UTM)をはじめとするセキュリティアプライアンスを導入し、内外の脅威に備え
- ネットワークトラフィックの監視を行い、不審な挙動を示すネットワークトラフィックの発生を検知することで、攻撃に備える
- ネットワークトラフィックを監視し、サーバがサービスを提供している、もしくはクライアントがサービスを受けるなど「以外」で発生するような、不自然な量のトラフィックを検知ことで、そのトラフィックに対応する
- ネットワークを通過するパケットを全てキャプチャ・保存することで、何らかのインシデントが発生した時の調査を容易にする

1.4. クライアント対策の難しさ

ここまでに、システムを構成する要素として「サーバ」「クライアント」「ネットワーク」という3つを例に出した。

この3つのうち、サーバについては、通常は提供するサービスに絞れば問題は局所化できる。また、ネットワークについても、トラフィックの種類を絞るなりしておけば、脅威の種類を局所化できる。しかし、クライアントはそういうわけにはいかない。

サービス提供を行う性質としてのサーバは、エンドユーザが直接コンソールを含むサーバのハードウェアや、基本ソフトウェアの機能を操作するものではなく、ネットワーク機器も同様の性質を持っている。しかし、クライアントは、ユーザが直接操作を行うことで、その役割を果たす。このような性質を持つため、守る対象としてのクライアントは、通常は物理的にユーザの数に比例してその数が増える。サーバやネットワークについては、一度システムを構築してしまえばそう大きな変動がない反面、クライアントはユーザ数が増えることで、その数を増やし、より管理を困難にする結果となる。

2. 端末に適用する情報漏洩対策とその問題点

通常、システムにおける端末として大量に利用されているパーソナルコンピュータ(PC)は、必要なソフトウェアをインストールされ、ユーザはそのソフトウェアとクライアントをセットで使用することで、クライアントはその役割を果たす。しかし、PCを端末として採用した場合、その管理は通常はユーザにゆだねられることになる。もちろん、ユーザに管理をさせず、端末管理用のソフトウェアを導入することで、集中的にPCのソフトウェア構成を管理することも可能ではあるが、プログラムやデータを格納したPCの「物理的な管理」は、そのPCのユーザ、もしくはそのPCを設置してある場所の管理者が管理を行うことが多い。

しかし、PCが端末として採用されている状況では、程度の大小はあれど、

- ユーザは（極力）自由に使いたい
- ソフトウェアの使用制限は嫌だ
- でも危険は回避したい

というジレンマを抱えることになる。このようなジレンマを抱える結果として、ユーザが使う環境を均一化出来ず、脅威のレベルもクライアントごとにまちまち、もしくは不明な状態に陥ることが多い。

しかしユーザから見ると、「PCとしての自由度は求めるが、その自由度ゆえの危険は回避したい」ということが多い。

結果として、端末を所有する組織は、その管理のために各種ソフトウェアのインストールを制限したり、

周辺機器の利用を制限したりするソフトウェアを追加で導入することになる。しかし、これらのソフトウェアを導入したとしても、以下のような問題がある。

- ユーザが使う環境下で、必ずしも作成者が意図したとおりに（正しく）動くとは限らない
- 導入したソフトウェアが、脅威の発端になる可能性がある
- ソフトウェアをいくら導入したところで、ハードウェア自体を持ち去られる

特に、情報漏洩対策という観点から見たら、最後に上げた「持ち去られる」という脅威は大きなものになりうる。特に、小型軽量な端末であっても、その中に格納されている情報は膨大であり、機密情報を扱っているような端末の場合には、端末を持ち去られることは、即情報漏洩事故につながる。

そんな中で、近年クローズアップされてきた対策の1つ、端末としてのシンクライアント導入である。

3. シンクライアント[1]

もともとシンクライアントは、PCがまだそれほど安価ではなかった時期に、高価でメンテナンスの手間が掛かるコンピュータの導入を最小限に抑え、コストを安くすませようという発想から開発された。「一人に一台ずつ高価なPCを使う」よりも、1台の高性能なコンピュータ（シンクライアントサーバ）を複数で利用し、個々の手元に「PCのように使えるが、複雑な記憶装置などは持たない安価な端末」（シンクライアント端末）を置くという考え方で、端的にいうと「TCOの削減」が元々のテーマになっている端末システムである。

現在、このような目的に合致する形でのシンクライアントシステムは、大きく3つの方式で実現されている。

3.1 ネットワークブート型シンクライアント

これは、ユーザが操作する端末にシステム起動用の記憶装置などを持たず、すべてネットワーク経由でアクセスするタイプのものである。具体的には、シンクライアント端末を起動する際には、シンクライアントの起動用サーバ上に配置した「システム起動用のデータ」を用いる。起動後の端末処理は、以下のような形になる。

- 主処理を行うCPUおよびメモリは端末側に配置する
- 記憶装置はサーバ側に配置する
- 主処理を行う端末とサーバの間では、NFS、iSCSI、もしくは独自プロトコルなどを使用して入出力処理を行う
- キーボード、ディスプレイ、マウスの存在およびその制御は主処理およびメモリと同様に、端末側に配置する

3.2 画面転送型シンクライアント

これは、シンクライアントサーバで構成した仮想画面を、シンクライアント端末に転送、投影するタイプのものである。システム起動用のデータは、CD-ROMなどの読み出し専用デバイスに配置したり、端末に内蔵したハードディスクなどの記憶装置に配置したりするが、本質的な処理は、シンクライアントサーバ側で実施するため、その処理結果を内蔵したハードディスクなどに記録することはない。なお、起動後の端末処理は、以下のような形になる。

- 主処理を行うCPUおよびメモリはサーバ側に配置する
- 記憶装置はサーバ側に配置する
- キーボード、ディスプレイ、マウスおよびそれらの制御は端末側に配置する
- 端末とサーバの間の情報はネットワークでRDP、ICA、X、もしくは独自のプロトコルを用いてやりとりする

描画自体は端末のみで行う場合（X Window System）と、サーバ側で仮想画面に描画し、その結果を端末に反映する場合（Windows Terminal Serverなど）がある

3.3 ハイブリッド型シンクライアント

これは、3.1と3.2で挙げた方式を組み合わせたものである。具体的には、画面転送型のシンクライアント端末を起動するために、ネットワーク経由でのシステム起動を行うというものである。

3.4 ネットワークブート型と画面転送型の共通点

3.3で述べた内容を整理すると、ネットワークブート型シンクライアントと、画面転送型シンクライアントの処理や制御の主体は以下のようになる。

(1) ネットワークブート型の処理や制御の主体
この方式では、処理を行うための装置や、ユーザが操作を行う装置の存在・制御は以下のとおりになる。

- 二次記憶装置はシンクライアントサーバ側に配

置される
- キーボード、ディスプレイ、マウスの配置およびその制御は端末側になる
- 本来意図するプログラムの主処理は端末側で実施される

(2) 画面転送型
この方式では、処理を行うための装置や、ユーザが操作を行う装置の存在・制御は以下のとおりになる。

- 二次記憶装置はサーバ側に配置されている
- キーボード、ディスプレイ、マウスの配置およびそれらの制御は端末側になる
- 本来意図するプログラムの主処理はシンクライアントサーバ側で実施される

この2つの方式で共通するのは、二次記憶の位置(シンクライアントサーバ側)とキーボード、ディスプレイ、マウスなどユーザと直接対話する部分の位置(端末側)である。

3.5 シンクライアント端末の定義

ここまで述べた中で、シンクライアント「端末」自体は、ユーザが使うために必要な入出力装置を備えており、実装から見た「シンクライアント」の共通点は、「主処理に関連した永続的な記憶」を持たないことである。もちろん、シンクライアントサーバ側では永続的な記憶を保持するが、この状況と端末側の状況は独立である。

このため本稿では、シンクライアント端末と呼ばれるものの定義を以下に提案する。

「ユーザーの手元に、任意かつ永続的な記録を保持しない端末」

上記の用件を備えた端末と協調して動作するサーバをシンクライアントサーバとして扱い、シンクライアント端末とシンクライアントサーバを組にしたものを、シンクライアントシステムと呼ぶことにする。

4. 関連研究

シンクライアントに関連した研究は、国際的には数えるほどしかなく、日本国内でもその数は決して多いものではない。

国外では、シンクライアントを利用した場合の性能に着目したもの[2]および、シンクライアントの今後の進化[3] に関連したものが挙げられる。また、国内では、トラステッドコンピューティングを基調としたセキュアクライアントの研究[3]が行われているが、よく使われている方式を想定した場合、シンクライアントには、後述するような欠点や課題も持っている。

5. シンクライアントシステムの利点と欠点

シンクライアントシステムは、すでに述べたとおり、もともとは TCO 削減を目的として開発されたものであり、情報セキュリティ的な観点から設計されたものではない。にもかかわらず、前述のような定義にあてはまる端末であるため、特に情報漏洩系のインシデントには強いとされている。このような事項も含め、シンクライアントシステムの利点と欠点を、情報システムのユーザ／管理者の観点と、情報セキュリティ的な観点の情報システムのユーザの視点から述べる。

5.1 情報システムのユーザから見た利点と欠点

シンクライアントシステムは、情報システムを使ったり、それを管理する側から見て利点と欠点を持っている。それを以下に挙げる。

- メンテナンスが容易（利点）
  単一のシンクライアントサーバをメンテナンスするか、個々の端末をメンテナンスするか、という違いがある。

もちろん実務的には、単一のシンクライアントサーバをメンテナンスするにあたっては相応のスキルを持った要員を割り当てる必要があるため、コストがかかるというように言われるが、これは別の観点でいうならば「コストの可視化」が行えるということにつながる。

- 自由度が低い（欠点）
  情報セキュリティの観点から見たら利点ではあるが、ユーザの観点から見たら欠点となりうるのが自由度の低さが欠点として挙げられる。
  もちろん、シンクライアントサーバ上の「ユーザ環境」を自由にさせるという手もあるが、そのようにすることで、「メンテナンスが容易である」という利点をなくしてしまうことにもつながる。

上記以外にも、端末の冗長度がなくなるため、シンクライアントシステムのトラブルによって、そのシステムに依存する全ての業務が同時に停止する、というものもあるが、この部分については今回は触れない。

5.2 情報セキュリティの観点から見たシンクライアント

情報システムのユーザから見た利点と欠点は前述のとおりだが、情報セキュリティの観点から見た利点と欠点はまた別になる。まず、シンクライアント端末でないクライアントがさらされている脅威を図1に示す。

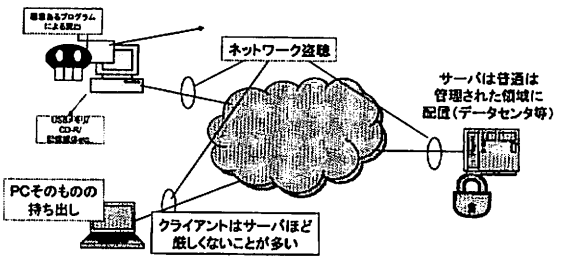

図1 サーバ、クライアント、ネットワークの脅威の例

サーバと比較すると、端末はより多くの脅威にさらされるといえる。シンクライアントシステムを構成する要素の1つであるシンクライアント端末は、このような脅威にどのように対抗できるのか。シンクライアント端末が持つ脅威への耐性について、その仮説を以下に示す。

- 物理媒体を経由した漏洩
  物理媒体自体のアクセスが制限されているため、体制は高いと考えられる
- 端末そのものの紛失／略取
  端末に情報を持っていないため、耐性は高い
- マルウェアへの耐性
  シンクライアントサーバと端末ともに、構成によっては脅威に晒される可能性がある
- ネットワーク盗聴
  プロトコルなどに依存する

5.3 シンクライアントのセキュリティ上の利点と欠点

俯瞰的に脅威を見て、シンクライアントに関連した仮説、そして情報システムのユーザから見た利点と欠点を勘案し、シンクライアントを使った場合のセキュリティ上の利点と欠点を以下に示す。

- メンテナンスやセキュリティ対策が容易（利点）
  シンクライアントサーバについては集中的なメンテナンスおよび情報セキュリティ上の対策が可能であり、個々に対策を行うよりも効果的かつ効率的である。
- シンクライアント端末が完全にセキュアであるとは限らない（欠点）
  ユーザが触れる側の環境には、汎用 OS を使用しているものもある。
  また、稼動しているソフトウェアに、何らかの脆弱性が潜んでいる可能性も考えられる。このようなことを考えると、シンクライアント端末の動作環境が完全にセキュアであるということは必ずしも言い切れない。

端末が攻撃を受け、制御を乗っ取られた場合、その端末が各種攻撃の拠点となる危険性があるが、それらの課題は、システムによらず存在する。但し、シンクライアント端末を安全なものとできない場合、その端末からの攻撃にもシンクライアントサーバは備えねばならない。このような攻撃は本来は実施しなくてもすむのが望ましく、そのために、端末が乗っ取られても、端末に近いところもしくは端末そのものに、防御や攻撃拡散防止のためのしくみを搭載することで、攻撃への対処コストを下げられる可能性がある。以降は、そのような端末の構成検討を行った結果を述べる。

6. シンクライアント端末をベースにしたセキュアクライアントの構成案

シンクライアントサーバ、シンクライアント端末、そしてその間を接続するネットワークの全てが安全であると担保できて、初めて「安全な端末システム」といえる。

ただ、前述のとおり、端末がセキュアでない状況があると、「安全でない」箇所が出来てくる。また、脅威に対する耐性を考えると、必ずしもシンクライアント端末が安全であるとはいえない。但し、端末の構成によっては、サービスを提供するコンピュータとしてのサーバなどと同様に、リスクを局所化することが可能になると考える。そのために、以下のような構成を検討している。

6.1 検討したセキュアクライアントの構成案

今回検討したセキュアクライアントの構成を図2に示す。セキュアな VMM と、その VMM ほどはセキュアでない VM の組み合わせで実現するが、本質的に信用ならない VM を限定的な用途に利用する、そのためのアプローチ案も図に含める。

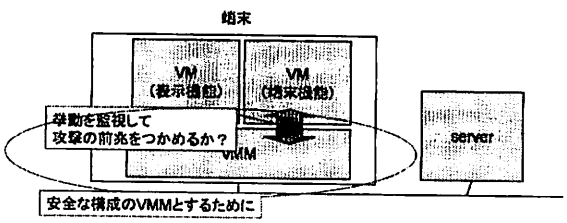

図2 セキュアクライアントの構成概要

これをさらに、具体的な実装に近いモデルにしたのが図3である。

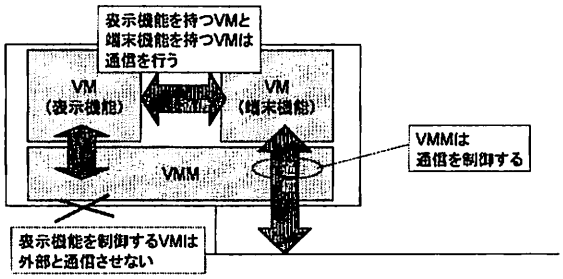

図3 セキュアクライアントの実装モデル

このようなモデルで示したシステムに対する脅威とその対策案を、以下で説明する。

6.2 検討したセキュアクライアントに対する脅威とその対策

本論文で検討したセキュアクライアントについては、そのまま実装するだけでは脅威の対策にならない。具体的にどのような脅威が考えられるかを図4に示す。

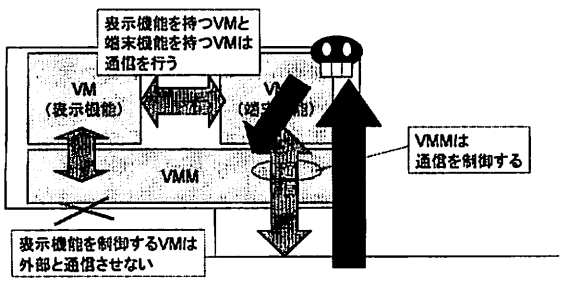

図4 VMに侵入された場合のMalwareの攻撃例

このような脅威については、外部との間で発生する通信処理が媒介になることがほとんどである。このような脅威に対し、いくつかのアプローチを入れ込んだ例を図5に示す。

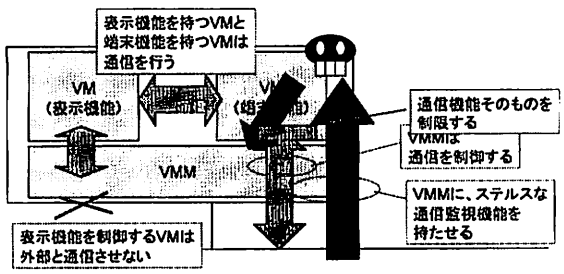

図5 脅威へのアプローチを行った場合の例

ここでの対処は、大きく以下の4つになる。

(1) VMMに、VMや外部からは直接存在を知ることが出来ない監視機能を持たせ、攻撃を検知／ブロックする
(2) VM側のTCP/IPスタックの機能を制限する。例えば、プロセスやVMそのものからの接続に関する条件を設けるなどの制限を加える
(3) 前述の「TCP/IP スタックの機能制限」に加え、通信制限の内容をより厳しくする。厳しくする内容／程度は要検討
(4) VM環境のセキュアOS化をはじめとする対策を施すことで、そもそもMalwareが動作できない環境にする

上記の対策のうち、(1)はVMMを用いた場合に採用可能なものであり、(2)(3)(4)はVMMの有無にかかわらず実現可能なものである。但し、VMMによる対策を組み合わせることで、当該アプローチを実現しやすくする、もしくは別の観点から同様のことを実現できる可能性がある。

7.今後の予定

6で述べた構成を実装の上で、課題として挙げた脅威およびその対処の妥当性を検証する。そのための課題としては、1つには脅威への耐性をどう定量化し、測定する方法の具体化が挙げられる。これまで挙げた仮説および脅威については、定性的なものであるため、その部分の具体化を行う。

また、セキュアクライアントの端末を構成した次のステップとしては、セキュアクライアントの主処理を行う、シンクライアントシステムで言うところのシンクライアントサーバ側の構成検討とその実装／実現を行う。

文　献

[1] 宮本久仁男, "シンクライアントの真価を問う," ITmedia エンタープライズ オンラインムック plus,, Jun 2006.

[2] Albert M. Lai and Jason Nie, "On the Performance of Wide-Area Thin-Client Computing," ACM Transactions on Computer Systems, Vol. 24, No. 2, May 2006, pp. 175–209.

[3] Niraj Tolia, David G. Andersen, and M. Satyanarayanan, "Quanitifying Interactive User Experience on Thin Clients," IEEE Computer, Mar. 2006, pp.46-52

[4] 中村めぐみ 宗藤誠治 須崎有康 飯島賢吾 八木豊志樹 大澤一郎, "トラステッド・コンピューティングによる HTTP-FUSE KNOPPIX クライアントのセキュリティ強化," 信学技報, vol.106, pp.223-230, Jul 2006.