リモートオフィス環境構築とセキュリティ
－個人認証、VPN、脆弱性検査、そして被害予測－

On Constructing the Environment of Secure Remote Office

森井 昌克 †
Masakatu MORII

毛利 公美 †
Masami MOHRI

1. まえがき

オフィス環境は情報通信ネットワークの発展・浸透によって、大きく変化し、最も早くユビキタスコンピューティングとその技術の恩恵を被っている。「サテライトオフィス」という言葉が「ホームオフィス」から「ユビキタス」という環境に取って代わり、いつでもどこでも誰とでもオフィスワークが可能な環境を与えようとしている。実際、ホットスポット等の無線 LAN や公共施設等でのネットワーク環境が整備され、いわゆる「ユビキタスオフィス」を実現する環境が提供されている。距離を克服するリモートオフィスにおいて、通信環境が比較的満たされようとしている現在において、最大の課題がセキュリティの問題である。そのリモートオフィスでのセキュリティを確保する個別技術として、「認証技術」、「通信路暗号化技術」、「不正アクセス対策技術」等が開発・提供されている。「認証」では、指紋、虹彩等を利用したバイオメトリック認証技術、固有の秘密情報に基づくパスワード認証技術を利用した製品群が提供されている。特に IC カードや USB メモリスティックを利用したワンタイムパスワード方式は簡便さゆえ、多くの方式が提案され、製品化され、実際に利用されている。「通信路暗号化技術」では、VPN と方式が提供され、IP-VPN からインターネット VPN、そして最近では、SSL-VPN が提案され、その利用が浸透しつつある。「不正アクセス対策技術」では、従来からのファイアウオール、IDS の利用が一般化し、それらが連携して、積極的に不正アクセスを排除する方式、IDP（Intrusion Detection and Prevention）も提案・製品化されている。

我々はリモートオフィスでのセキュリティを確保するために、上記の個別技術について研究・開発を行い、いくつかの方式を提案し、評価している。また、それらの総合的に評価し、リモートオフィスでの総合的セキュリティシステムの構築を行ってきた。本稿では、ワンタイムパスワード認証、サーバおよびクライアント脆弱性検査、さらに不正アクセスに対する被害予測システムについて述べる。VPN についても我々はクライアント側の VPN クライアントモジュールをサーバ側から遠隔制御し、サーバ側のファイアウォールと連携することで、アクセス制御の集中管理が容易にできる新しい VPN 方式を提案している [1][2] が、この方式の実装・評価、および認証方式との統合等については、稿を改めて発表する。

2. サーバ認証機能を有するワンタイムパスワード方式

ワンタイムパスワード方式として Lamport 方式，CINON, PERM, SAS, OSPA, SAS-2 などがある [3, 4, 5, 6, 7, 8, 10]. Lamport 方式 [3, 4] はあらかじめ一方向性ハッシュ関数を複数回適用したデータをサーバ側に保存しておき，ユーザがハッシュ関数の適用とは逆順にデータをサーバに示すことで認証される方式である．Lamport 方式にはあらかじめ登録されていた認証情報を使い切るたびに再設定する必要があることや，ユーザ側で必要とされる計算量が多いことなどの問題点があった．清水らは認証情報の再設定が必要なく，ユーザ側での計算量も少ない CINON (Chained One-Way Data Verification Method) を提案した [5]．しかし，Lamport 方式，CINON ともにユーザ側で何らかの認証情報を保持しておく記憶媒体が必要であり，特に CINON はユーザ側で擬似乱数を生成できる必要があった．そこで，清水らはユーザ端末側に記憶媒体や擬似乱数生成装置がない状況での認証を実現する方式として PERM (Privacy Enhanced information Reading and writing Management method) を提案した [6]. 上記の Lamport 方式，CINON，PERM はいずれも Man in the Middle attack などの通信回線への能動的な攻撃には対応できない方式であった．そこで Man in the Middle attack への耐性を持った認証方式として SAS(Simple And Secure password authentication protocol) が提案された [7]．SAS は Man in the Middle attack への耐性を持ち，ユーザ側で必要とされる計算量も CINON や PERM よりも低い方式であるが，PERM とは違いユーザ側で擬似乱数を生成する必要があった．さらに，Replay attack や Denial of Service attack が成立することが示された [8]．文献 [8] では Replay attack と Denial of Service attack に耐性のある認証方式として OSPA (Optimal Strong-Password Authentication) が提案されている．OSPA はユーザ側で必要とされる計算量が SAS と同程度であり，ユーザ側に擬似乱数生成装置や記憶領域を必要としない方式であるが，Impersonation attack が成立することが示された [9]．また，上記の各種攻撃法に耐性があり，サーバ認証が可能である方式として SAS-2 が提案されている [10].

本稿では，サーバ認証が可能であり，各種攻撃法に耐性のある新しい認証方式を提案する．提案方式はユーザの秘密情報ではなく，各セッションごとに更新される使い捨ての秘密鍵を用いて認証子を作成する．ユーザの秘密情報は，はじめに使い捨ての秘密鍵を作成するためだけに用いられるため，ユーザとサーバの端末上に残らずネットワーク上を流れることもない.

2.1 記号の定義

本稿で用いる記号を次のように定義する．

A	ユーザ
ID	ユーザ ID
H	認証サーバ
E	攻撃者
S	ユーザの秘密情報
P_i	i 番目のセッションの認証子
R_i	i 番目のセッションの乱数
K_i	P_i を生成するために A と H が共有する使い捨て鍵
h	一方向性ハッシュ関数. $h(m)$ は m を一回ハッシュしたもの
L	データ長．$L(X)$ は X のデータ長
$\oplus$	ビット単位の排他的論理和
$\|\|$	ビット列の連結
$A \longrightarrow B : X$	A は B に X を公開通信路で送信
$A \Longrightarrow B : X$	A は B に X を秘密通信路で送信

2.2 提案方式

提案方式はサーバ認証機能を有するワンタイムパスワード認証方式である．提案方式ではサーバの送信する情報をクライアント側で認証し，クライアントの送信する情報をサーバ側で認証する．また，提案方式ではセッションごとに更新される使い捨て鍵を用いて認証が行われる．ユーザの秘密情報

†徳島大学 工学部 知能情報工学科, Dept. of Information Science and Intelligent Systems, Faculty of Engineering, The University of Tokushima

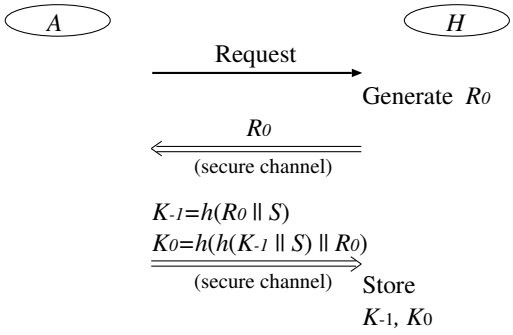

図 1: 登録フェーズ

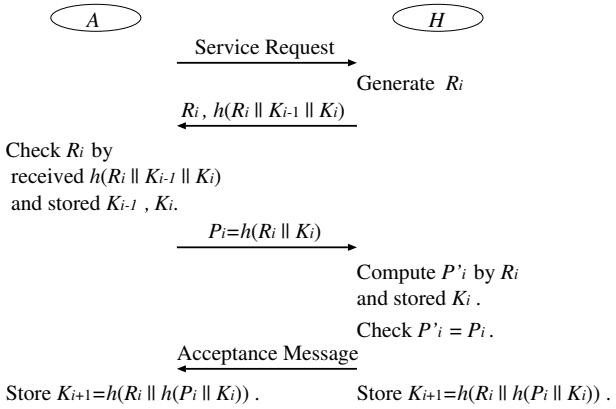

図 2: 認証フェーズ

は登録フェーズにおいて使い捨て鍵を生成するためだけに用いられ，即座に削除されるためユーザ端末やサーバ上に残らず，ネットワーク上を流れることもない．

2.3 プロトコル

提案プロトコルは登録フェーズと認証フェーズに分けられる．登録フェーズは初めに一度だけ行われる．認証フェーズはユーザがログインする度に毎回行われる．登録フェーズの手順を図 1，認証フェーズの手順を図 2 に示す．

登録フェーズ

A は次のように初回登録を行う．

Step R1 $A \longrightarrow H$: 登録要求.

Step R2 $H \Longrightarrow A : R_0$.

Step R3 A は以下のように K_{-1} と K_0 を計算し，保存.
$K_{-1} = h(R_0||S)$
$K_0 = h(h(K_{-1}||S)||R_0)$

Step R4 $A \Longrightarrow H : K_{-1}, K_0$.

Step R5 H は K_{-1}, K_0 を保存.

認証フェーズ

A の i 回目のログインに対して次のように認証を行う．

Step A1 $A \longrightarrow H$: 認証要求.

Step A2 $H \longrightarrow A : R_i, h(R_i||K_{i-1}||K_i)$.

Step A3 A は受信した R_i を自身が所有している K_{i-1}, K_i を使って $h(R_i||K_{i-1}||K_i)$ と比較し，R_i を検証.
$P_i = h(R_i||K_i)$ を計算.

Step A4 $A \longrightarrow H : P_i$.

Step A5 H は $P_i' = h(R_i||K_i)$ を計算し，$P_i' = P_i$ ならば接続許可するとともに，$K_{i+1} = h(R_i||h(P_i||K_i))$ を計算し，保存.

Step A6 $H \longrightarrow A$: 接続許可.

Step A7 A は $K_{i+1} = h(R_i||h(P_i||K_i))$ を計算し，保存.

2.4 安全性の評価

代表的な攻撃手法である Man in the Middle attack，Replay attack，Denial of Service attack，Impersonation attack についての耐性を考察することで提案方式の安全性を評価する．

2.4.1 Man in the Middle attack

Man in the Middle (MIM) attack とは，攻撃者 E が通信する 2 者の間に割り込み，通信データの盗聴や改ざんを行う攻撃である．

提案方式に対して MIM attack を適用することを考える．このとき，E は H の送信する R_i を改ざんして偽のデータを A に送り，それに対する A の返答を搾取して正しい返答を H に送信する必要がある．しかし，提案方式では A は受信した R_i を毎回検証するため，E の改ざんを検知できる．また，A の返答は A と H しか知らない使い捨て鍵を用いて生成されるため，E は正しい返答を生成することができない．以上より，提案方式は MIM attack に対して安全であると言える．

2.5 Replay attack

Replay attack とは過去の認証セッションの通信データを E が盗聴し，再利用することでなりすましを行う攻撃である．

提案方式に Replay attack を適用する場合，i 回目のセッションで E が盗聴できる通信データは次の通りである．

$$R_i, h(R_i||K_{i-1}||K_i)$$
$$P_i = h(R_i||K_i)$$

これらのデータから i 回目以降のセッションの通信データや使い捨て鍵が作成できれば攻撃は成功であると言える．仮に E が各セッションにおける乱数を自由に選択できるとする．例えば $R_j = 0, j = 0, 1, 2, \ldots$ とする．このとき使い捨て鍵の更新は次のように表せる．

$$K_{i+1} = h^2(P_i||K_i)$$

E は P_i は知り得るが，K_i を知ることができないため，K_{i+1} を作成することはできない．また，$i+1$ 回目の通信データは次のように表される．

$$0, h(K_i||K_{i+1})$$
$$P_{i+1} = h(K_{i+1})$$

E は K_{i+1} が作成できないため，$h(K_i||K_{i+1})$ と $P_{i+1} = h(K_{i+1})$ を作成することもできない．以上より，たとえ E が自由に乱数を決定できたとしても提案方式は Replay attack に対して安全であると言える．

2.6 Denial of Service attack

Denial of Service (DoS) attack とは E が通信する 2 者の正規の認証を不可能にする攻撃である．DoS attack の実現方法として，E が A もしくは H に偽の認証情報を受理させ

ることで誤った認証情報を登録させ，以降の認証セッションを不可能にする方法がある．

提案方式に対して DoS attack を適用するには，E が A もしくは H に偽の認証情報を受理させる必要がある．しかし，提案方式では H から A に送られる R_i は $h(R_i||K_{i-1}||K_i)$ によって検証され，A から H に送られる P_i は $P_i' = h(R_i||K_i)$ によって検証される．K_i を知らない E は $h(R_i||K_{i-1}||K_i)$, P_i ともに作ることができず，偽の認証情報を受理させることができない．以上より，提案方式は DoS attack に対して安全であると言える．

2.6.1 Impersonation attack

Impersonation attack とは，E が過去のセッションの情報だけではなく，現在のセッションや未来のセッションの情報も用いてなりすましを行う攻撃である．E が未来のセッションの情報を得るためには，H になりすまして A との認証セッションを行い A からの認証情報を集める必要がある．

提案方式に対して Impersonation attack を適用することを考える場合，E は未来のセッションの認証情報を得ることはできない．E が A からの返答を得るためには H になりすまして乱数とその認証子を A に送信する必要があるが，E には適切な認証子が作成できないからである．E が $i-1$ 回目までのセッションの通信データを盗聴し，i 回目のセッションの通信データを改ざんすることで $i+1$ 回目以降のセッションで A になりすまそうとする場合を考える．このとき，E は i 回目のセッションでなんらかの都合のいいデータを H に受理させる必要があるが，H では $P_i' = h(R_i||K_i)$ によって受信したデータを検証するため，P_i 以外の認証子を H に受理させることはできない．以上より，提案方式は Impersonation attack に対して安全であると言える．

3. ネットワーク脆弱性自動検査システム

常時接続環境の普及や高性能なパソコンの出現によって，個人や企業でも低コストでインターネットサーバを構築することが可能となっている．

インターネットサーバはその性質上, ネットからの攻撃に曝されることになるため, 外部に公開しているサーバプログラムに脆弱性が発見された場合は早急にプログラムをアップデートする必要がある．しかし，実際にはサーバの脆弱性を調べる作業は非常に繁雑であり，十分な専門知識を持った管理者の不足等の理由によって脆弱性を持ったサーバがそのまま運用されていることが多い．

本章では, サーバソフトウェアの脆弱性に関する情報収集およびバージョンの確認を自動化し，脆弱性が発見されたサーバソフトウェアの使用が確認された場合に限り管理者へ通知するネットワーク脆弱性自動検査システムを提案する．

3.1 既存のセキュリティ監査システム

ネットワーク管理者の作業を補助するためのセキュリティ監査システムは既に数多く存在しており, ローカルホストだけでなく，リモートホストも監査できるものがある．リモートホストを監査する際，目的のホストやネットワークに対してポートスキャンを実施し，それにより得られる情報をもとに脆弱性の有無を調べる．

しかしながら, 監査される側からすると，ポートスキャンはクラックの準備と見なすことができる．このため，ポートスキャンを検出した場合はしばらくファイヤーウォールでスキャンを行なったサイトからの通信を遮断するといった設定をすることが多い．したがって，ポートスキャンを用いたセキュリティ監査ツールは自サイトへの適用に留めるべきであり，他のサイトへの使用は謹むべきである．

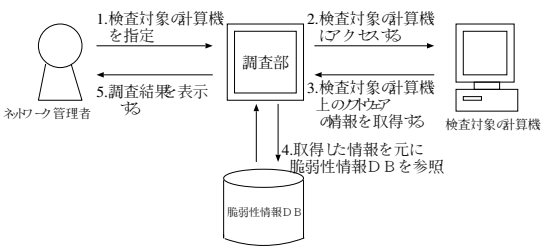

図 3: 本システムの動作概念

3.2 ネットワーク脆弱性自動検査システム

本研究で開発した “ネットワーク脆弱性自動検査システム” について述べる．本システムのインタフェースは CGI として実装されており，ユーザは検査を行うドメイン名のみをブラウザから入力することでサーバの脆弱性を確認でき，専門知識を持たない人でも簡単に脆弱性を確認できる．

本システムでは，入力されたドメイン名を用いて DNS への問い合わせを行い，各種サーバの IP アドレスを取得する．DNS を利用することで，個々のサーバの IP アドレスを入力する必要はなくなる．また，前節に挙げた理由から，ポートスキャンを使わず，サーバへの通常の問い合わせから得られるバージョン情報を脆弱性情報データベースと照合して検査を行う方式とした．ただし，本システムの脆弱性検査結果は攻撃対象の下調べとしても有効なため，不特定多数のユーザにシステムを開放するのは危険である．そこで本システムはユーザのアカウント管理をし，利用者の制限を行っている．

本システムは「脆弱性情報データベース部」と「サーバ情報収集部」で構成される．本システムの動作概念を図 3 に示す．

3.2.1 脆弱性情報データベース部

脆弱性の情報は CERT[12] やベンダなどから報告されている．脆弱性情報データベース部では自動的に Web 上で公開されているサーバソフトウェアの脆弱性情報を収集し，その情報を元にして脆弱性情報データベースを更新する．この処理を定期的に行い，自動的に最新の脆弱性情報を収集する．

インターネットでは，さまざまなサイトにおいて脆弱性の告知が行なわれているが，本システムが対象とする管理者には日本語で情報提供されていることが望ましい．そこで本システムでは情報収集の対象として “`JPCERT/CC Vendor Status Notes(JVN)`” を選択した [13]．JVN 以外のサイトでも，提供されるデータが一定のフォーマットに従っていれば情報収集の対象として追加することも可能である．

3.2.2 サーバ情報収集部

サーバ情報収集部では，検査対象のホストが使用しているサーバソフトウェアの名称とバージョン番号を取得する．対象となるホストは IP アドレスやドメイン名で指定する．本システムにおいて情報収集の対象としているサーバは Web(HTTP) サーバ，Mail(SMTP) サーバおよび DNS サーバである．取得した情報をもとにあらかじめ構築した脆弱性情報データベースを参照し，それぞれのソフトウェアの脆弱性に関する情報を返す．

また，本システムでは Windows の修正プログラムの適用状況の検査も行うことができる．検査対象のホストで WindowsOS が稼働している場合，上記の 3 つの検査に加え，Windows の修正プログラムがインストールされているか否かを検査する．

［情報収集手順］
サーバ情報収集部における情報収集の手順 (図 3 における 2, 3 の処理に相当) について述べる．

Web サーバ　指定された IP アドレスのポート 80 番もしくは指定された URL にアクセスし，コマンド “`HEAD / HTTP/1.0`” を送出する．コマンドの返答から Web サーバで使用しているソフトウェアの名称とバージョン番号を得る．

Mail サーバ　あるドメインにメールを送る場合，そのドメインの MX(Mail eXchanger) サーバとして指定されているホストにメールが送られる．本システムでは，実際のメール配送に使われる MX サーバを確認するため，まず指定された IP アドレスやドメイン名に対して “`nslookup -type=mx` ドメイン名 `or IP` アドレス” を実行して MX レコードを検索し，MX サーバのアドレスを入手する．得られた MX サーバの SMTP ポート (TCP 25 番) にアクセスし，接続時のメッセージから「Mail サーバのソフトウェア名称」と「バージョン番号」を得る．

DNS サーバ　あるドメインを受け持つ DNS サーバのアドレスを確認するため，まず指定された IP アドレスやドメイン名に対して “`nslookup -type=mx` ドメイン名 `or IP` アドレス” を実行して DNS に NS レコードを検索し，DNS サーバのアドレスを入手する．得られたサーバに対して DNS のバージョンを問合せるコマンド “`nslookup -type=txt -class=chaos version.bind.` DNS サーバのドメイン名 `or IP` アドレス” を送ることでバージョン情報を得ることができる．

以上の方法で得られたサーバソフトウェアの名称とバージョンを元に脆弱性情報データベースを参照し，脆弱性の有無について確認を行なう．

次に，WindowsOS が稼働するホストにおける修正プログラムのインストール状況の検査方法について述べる．

Windows の修正プログラムのインストール状況を検出
Windows の修正プログラムのインストール状況を検出するために，まず WindowsOS が稼働しているホストを検出する．指定された IP アドレスとドメイン名から nmap コマンドにオプションとして-O を指定して実行し，Remote operationg system guess の項目を正規表現で調べる．これにより，指定されたホスト上で稼働している OS を検出できる．

次に，Windows の修正プログラムのインストール状況を調べる．この方法として，本システムでは Nessus[14] のプラグインを用いる．Nessus のプラグインを用いた脆弱性の検査結果より，指定されたホストの Windows の脆弱性の有無を調べることができる．この脆弱性の有無により，Windows の修正プログラムのインストール状況を検出できる．

3.2.3　脆弱性検査の対象範囲

本システムでは指定されたドメイン名から DNS へ MX や NS レコードの問い合わせを行い，検査対象となるサーバのアドレスを求める．DNS や MX はバックアップ用として組織外にセカンダリーサーバを設置することがある．本システムでは DNS に登録されているサーバを検査するため，指定したドメイン外のサーバも情報収集の対象となることがある．そこで, 脆弱性検査の範囲を指定したドメイン内のサーバに限定するために，ユーザが利用しているホストと同一ネットワーク内のサーバのみ情報収集するオプションを用意した．このオプションを利用した場合は，ユーザの IP アドレスと上位 3 オクテットが同一のサーバのみを検査対象とする．

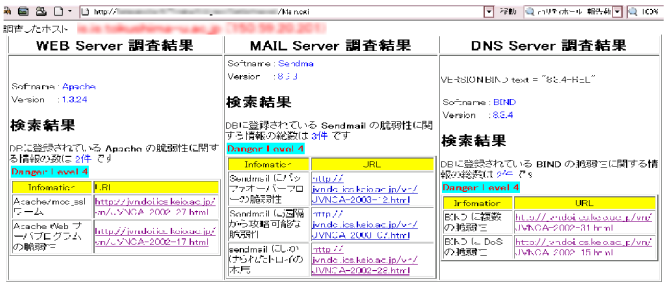

図 4: 指定されたドメインに対する検査結果

3.3　実行結果

本システムを稼働させるために必要な環境を示す．

- **Web サーバ：** Apache (CGI が動けば他のサーバでも可)
- **データベース：** PostgreSQL
- **CGI スクリプト実行環境：** Perl

本システムに対してあるドメインを入力し，脆弱性の検査を行なった結果を図 4 に示す．

このように検査対象となるドメイン名を入力すると，そのドメインに対応する Web, Mail, DNS サーバのバージョンを確認し，脆弱性データベースと照合した結果をブラウザに表示する．また，WindowsOS の稼働を検出し，その修正プログラムの適用状況の検査結果表示している．サーバに脆弱性が存在する場合は，その脆弱性を指摘するレポートへの URL を表示する．管理者はレポートから脆弱性の詳細な情報を入手し，サーバへの対策を行うことができる．

4.　イベント依存モデルによる不正アクセスの被害予測

不正アクセス対策の一つとしてファイアウォールが広く利用されている．しかしながら，ファイアウォールを迂回する不正アクセス手法が存在するため，IDS(侵入検知システム：Intrusion Detection System) を併せて導入するところが増えている．近年，IDS はセキュリティ分野で注目されており，IDS に関する研究や製品開発が活発に行われている．

IDS は不正アクセスからネットワークやシステムを守るために利用される．ネットワークのトラフィックを検査対象とし，不正アクセスを検出する IDS をネットワークベース IDS(Network-based IDS) [20, 21, 22] と呼ぶ．システムの活動やログを検査対象とし，不正アクセスを検出する IDS をホストベース IDS(Host-based IDS) [23, 24] と呼ぶ．IDS の主な役割は，検査対象データに不正アクセスの痕跡が含まれていないかを調べることである．

IDS の問題点として次の 2 つがある．1 つ目の問題点は，ローカルドメインのネットワーク管理者にセキュリティに関する高度な専門知識が要求されることである．なぜなら，IDS が発する多数のアラートから実施された不正アクセスを見つけ出し，さらにその対策を調べなければならないことと，IDS が発するアラートはシステムまたはサービス間の不正アクセスに関する関連性を考慮していなからである．2 つ目の問題点は，不正アクセスに対して事後の対処を強いられることである．それは，IDS が不正アクセスを検知するだけであり，不正アクセスに対して事前に対処するものではないことに起因している．1 つ目の問題点に関して，ローカルドメインに設置した IDS の遠隔監視サービス [25, 26] や被害解析支援システム [27] などが存在する．ローカルドメ

インで何らかのインシデントが発生し IDS が不正アクセスを検知した場合，IDS の遠隔監視サービスではサービス提供者がアラートを受け取る．そして，ローカルドメインの管理者に代わってアラートを分析し，インシデントに関する報告書を作成する．被害解析支援システムではそのシステムが IDS のアラートを受け取り，ローカルドメインの管理者に代わって被害の検出，原因の特定および対策の提示を行う．しかしながら，IDS の遠隔監視サービスや被害解析支援システムなどを用いても 2 つ目の問題点は未解決のままである．そのため，不正アクセスによる将来起こり得る被害を予測し，迅速に対応することで被害を最小限度に抑えることが必要とされている．

不正アクセスによる被害を予測する研究として，鴨田らが提案したニューラルネットワークを用いた被害予測方式 [28] がある．この方式は，ニューラルネットワークを用いて被害を “実害なし ”，“環境情報漏洩 ”，“性能低下 ”，“システムダウン ”，“不正操作 ” の五つに大まかに分類し，現在までに起こった不正アクセスから予想されるシステムやネットワークの被害状況を予測するものであり，具体的な将来起こり得る被害を予測するものではない．そのため，被害を最小限に抑えるという目標を達成することは困難である．

本稿では不正アクセスによって発生するイベントに着目し，現在までに行われた不正アクセスからその後に行われる具体的な不正アクセスを予測するためのイベント依存モデルを提案する．本モデルを使うことで，精度の高い被害予測ができ，ローカルドメインのネットワーク管理者は被害を最小限に抑えるための対策方法を容易に得ることができる．そして，得た対策を実施することで，不正アクセスに対して事前予防 (プロアクティブな対応) が可能となる．

4.1 被害予測システム

4.1.1 概要

本稿で提案するイベント依存モデルを用いた被害予測システムの目的は IDS が発するアラートから将来起こり得る被害を予測することである．被害予測システムの最も簡単な使用環境は単一サイト内で使用することである．被害予測システムを設置したサイトで何らかのインシデントが発生した場合，サイト内に設置されている IDS は自動的に被害予測システムに接続する．そして，アラートを被害予測システムに送る．被害予測システム側では，受け取ったアラートをイベント依存モデルを用いて分析し，将来の被害を予測する．不正アクセスを受けたサイトのネットワーク管理者に提示される通知結果には，予測される被害とその対策が含まれる．

被害予測システムを単一サイト内で用いる場合，IDS をサイト内に設置しなければならずネットワーク管理者にとって負担が大きい．なぜなら，IDS の能力を最大限発揮するためにはシグネチャを常に最新のものに更新し続けなければならないからである．また，IDS には固有の特徴があり，検知できる不正アクセスもあれば検知できない不正アクセスもあることが知られている．これらの 2 つの問題点を解決するものとして著者らが提案している Center Management Type Intrusion Detection System(以下センター集中管理型侵入検知システムと称する) [29] がある．本稿では，センター集中管理型侵入検知システムに被害予測システムを利用して被害を予測することを考える．

図 5 はセンター集中管理型侵入検知システムのモデル図である．センターは複数のサイトを監視する．監視しているサイトで何らかのインシデントが発生した場合，そのサイトは自動的にセンターに接続する．そして，パケットキャプチャエージェントが記録した通信データ (以下監査データと称する) をセンターに送る．センター側では数種類のネットワークベース IDS を配置しておき，受け取った監査データを分析する．センターに配置された被害予測システムでは数種類の IDS が出力したアラートから被害を予測する．その

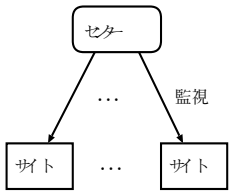

図 5: センター集中管理モデル

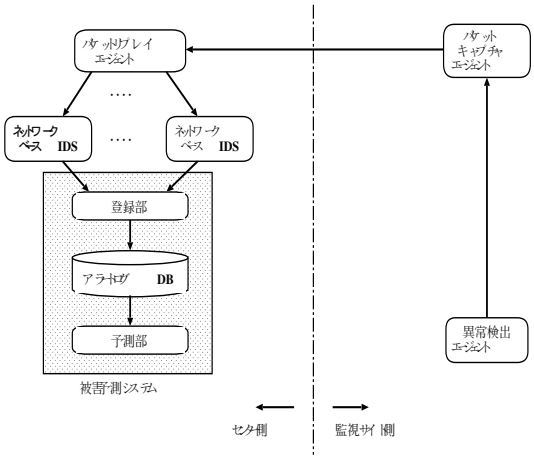

図 6: 被害予測システムの構成と処理の流れ

際，各アラートの依存関係を調べることでイベント依存モデルを作成し，被害を予測する．そして，不正アクセスを受けたサイトの管理者に予測される被害とその対策を通知する．

4.1.2 構成

図 6 は被害予測システムの構成と処理の流れを示している．被害予測システムは異常検出エージェント，パケットキャプチャエージェント，パケットリプレイエージェント，数種類のネットワークベース IDS，登録部，アラートログ DB と予測部から構成される．

パケットキャプチャエージェントは監視サイト側のゲートウェイ付近に設置され，監視対象を出入りする全ての通信を記録する．パケットリプレイエージェントと数種類のネットワークベース IDS はセンター側のシミュレーションネットワーク上に配置される．監視対象に設置された異常検出エージェントはホストの状態やファイアウォールのログを監視する．ホストの状態やファイアウォールのログから異常が検出された場合，異常検出エージェントはパケットキャプチャエージェントに指示を出す．これにより，パケットキャプチャエージェントは監査データをパケットリプレイエージェントに送信する．パケットリプレイエージェントは受け取った監査データをシミュレーションネットワーク上に再発生させ，数種類のネットワークベース IDS に分析させる．数種類のネットワークベース IDS が出力したアラートは被害予測システムの登録部に送られる．登録部では各アラートから送信元 IP などの通信に関する情報が抜き出され，アラートログ DB に登録される．予測部はイベント依存モデルを用いアラートログ DB に登録された情報から被害を予測する．そして，予測した被害とその対策を含めた報告書を監視サイトの管理者に通知する．

次に，上で述べた各要素について説明する．

[**異常検出エージェント**]

異常検出エージェントの役割は監視しているホストの状態

```
[**] [1:628:2] SCAN nmap TCP [**]
12/24-17:00:00.65421 100.0.0.1:1800 -> 192.168.67.211:23
TCP TLL:64 TOS:0x0 ID:39717 lpLen:20 DgmLen:60 DF
******S* Seq: 0x16720CE3 Ack: 0x0 Win: 0x7D78 TcpLen: 40
TCP Pptions (5) => MSS: 1460 SackOK TS: 1234164278 0 NOP WS: 0
```

図 7: Snort のアラートログの一例

やファイアウォールのログから異常を検出することである．もし異常検出エージェントが異常を検出した場合，インシデントの分析を開始させるために，パケットキャプチャエージェントに指示を出し，監査データをセンターに送信させる．異常検出エージェントが検出する異常として攻撃者からの偵察行為やファイルの改ざんなどが想定される．異常検出エージェントとして，既存のホストベース IDS を用いる．

[**パケットキャプチャエージェント**]

パケットキャプチャエージェントの役割は監視対象のゲートウェイを出入りする全ての通信を記録することである．そのため，監視サイトの出入り口であるゲートウェイ付近に設置される．異常検出エージェントよりインシデント分析の指示を受けた場合，監査データをパケットリプレイエージェントに送信する．この際，記録している全ての監査データを送信するのではなく，インシデントが検知された時刻付近の監査データのみをパケットリプレイエージェントに送信する．なお，パケットキャプチャエージェントでは監視サイトから出ていく通信も記録する．これは，攻撃者が監視サイトから得た情報を被害予測システム側でも知り，被害予測時に用いるためである．

[**パケットリプレイエージェント**]

パケットリプレイエージェントの役割は受け取った監査データをセンター側のシミュレーションネットワークに再発生させることである．再発生した監査データは数種類のネットワークベース IDS で分析される．数種類の IDS によって監査データを分析することで検知率の向上が望める．

[**アラートログ DB**]

アラートログ DB の役割は過去にさかのぼって被害予測を行うことができるように，アラートに関する情報を保存することである．これにより，偵察行為と偵察行為後に実施される不正アクセスに十分間があいても被害予測が可能となる．

アラートログ DB が管理するデータ項目について述べる．検知された攻撃を特定するための情報として，IDS 名とシグネチャ名が必要である．出力されたアラートの順序関係を把握するための情報として日付，時刻が必要である．攻撃元と攻撃先を特定するための情報としてそれぞれ送信元 IP/Port，送信先 IP/Port が必要である．また，プロトコルの種類に関する情報も必要である．上記要素を持つアラートログ DB の形式は表 1 のようになる．

[**登録部**]

登録部の役割は各種 IDS から受け取った書式の異なるアラートから被害予測に用いる情報を抜き出し，統一されたフォーマットをもつアラートログ DB に格納することである．センター側では数種類のネットワークベース IDS を利用する．IDS が出力するアラートの形式は IDS ごとに異なる．そこで，“アラートログ DB ” で述べたどのシグネチャにも含まれ，かつ被害予測に役立つ情報をアラートから抜き出し，アラートログ DB に登録する．図 7 は Snort が出力するアラートの一例を示している．

[**予測部**]

予測部の役割はアラートログ DB に保存されている情報から被害を予測することである．その際，**3** 節で述べるイベント依存モデルを用いて被害を予測する．予測部では複数の視点から被害を予測する．不正アクセスが同一の IP から実施された場合，その送信元 IP に関する情報をアラートログ DB から抜き出し被害を予測することが有効だと考えられる．また，不正アクセスが IP を変更しながら実施された場合，アラートログ DB に保存されている全ての情報から被害を予測する．

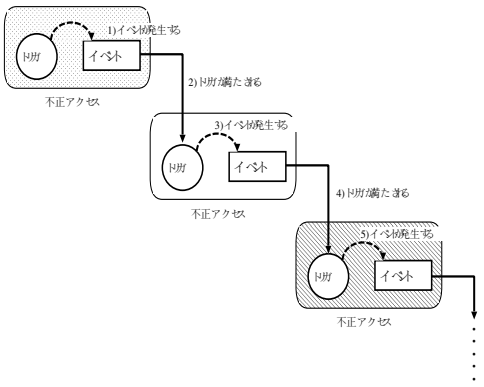

図 8: 不正アクセスの連続性

4.2 イベント依存モデル

不正アクセスの多様性には，攻撃対象の環境による多様性と不正アクセス手法の進化による多様性の 2 種類がある．攻撃対象の環境による多様性とは，攻撃対象にインストールされている OS の種類やバージョンなどの環境に攻撃者が実施する不正アクセスが依存するということである．不正アクセス手法の進化による多様性とは，日々新たな不正アクセス手法が考え出されるということである．そのため，不正アクセスの傾向をあらかじめ DB 化するだけでは上で述べた不正アクセスの多様性に対応することが困難である．そこで，本研究では図 8 で示したようにある不正アクセスによって発生したイベントがさらなる不正アクセスを発生させるトリガとなり，新たにイベントが発生するという不正アクセスの連続性に注目する．

IDS のアラートなどの不正アクセスの痕跡から得られる情報のイベント情報から依存関係を調べる．イベント依存モデルとは，不正アクセスによって発生するイベントをもとに各不正アクセスの依存関係を表したものである．ここで，イベントとは不正アクセスによって起こる攻撃者の状態の変化である．イベントの依存関係がわかればある不正アクセスが実施された場合，多数ある不正アクセス手法の中から次に実施される不正アクセスを絞り込むことができる．さらに，ある不正アクセスが実施された場合，今後実施される不正アクセスを予測できる．したがって，今後実施される不正アクセスを予測することで，被害を予測することができる．

提案システムでは不正アクセスの痕跡を示す情報の一つである IDS のアラートからイベント依存モデルを作成し，それを用いることによって被害予測を行っている．

4.3 被害予測

4.3.1 被害の予測方法について

不正アクセスの痕跡を示す情報の一つであるアラートを被害予測システムへの入力とし，イベント依存モデルを用いることで，今後実施される不正アクセスを予測する．図 9 は，予測部の構成を示している．

イベント依存モデルを作成する手順を次に示す．

Step1) 依存元の指定 … 入力されたアラートから依存元となるアラートを一つ選ぶ

Step2) イベント情報の取得 … Step1 で依存元に指定したアラートのイベント情報を取得する

Step3) 依存先候補を指定 … 依存元を除いたアラートの中から，依存先候補とするアラートを一つ選ぶ

表 1: アラートログ DB の形式

IDS 名	シグネチャ名	日時						送信元情報		送信先情報		プロトコル
		年	月	日	時	分	秒	IP	Port	IP	Port	
Snort	SCAN namap TCP	2003	12	24	17	00	00	100.0.0.1	1800	192.168.67.211	23	TCP
Snort	SCAN fingerprint attempt	2003	12	24	17	01	00	100.0.0.1	1801	192.168.67.211	23	TCP
Snort	TELNET Bad Login	2003	12	24	17	01	30	100.0.0.1	2000	192.168.67.211	23	TCP
Snort	TELNET Bad Login	2003	12	24	17	01	40	100.0.0.1	3000	192.168.67.211	23	TCP
Snort	DOS Jolt attack	2003	12	24	17	02	00	100.0.0.1	4000	192.168.67.211	100	TCP
.	.	.						.		.		.
.	.	.						.		.		.
.	.	.						.		.		.

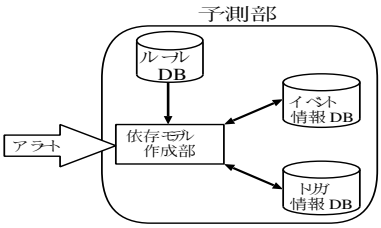

図 9: 予測部の構成

Step4) トリガ情報の取得 … Step3 で依存先候補に選んだアラートのトリガ情報を取得する

Step5) 依存関係の調査 … ルールにもとづき Step2 で得たイベント情報と Step4 で得たトリガ情報を比較し，依存関係を調べる

指定した依存元アラートに対して，Step3 〜 Step5 の処理を繰り返すことで，依存関係をある程度の範囲まで調べる．これにより，幅広く被害を予測する．また，入力となる全てのアラートに対して，Step2 〜 Step5 の処理を繰り返し行うことで，予測結果が重なり合うことが考えられる．これにより，実施される可能性が高い不正アクセスを予測することができる．

不正アクセスを実施したとき，アラートから抜き出せない情報を攻撃対象から攻撃者は得ている場合がある．例えば，攻撃者がポートスキャンを実施した場合，攻撃者は攻撃対象でサービスを提供しているポート番号に関する情報を得る．これに対し，ポートスキャンの痕跡を示すアラートから攻撃対象でサービスを提供しているポート番号を得ることはできない．そこで，被害予測の精度向上のために攻撃者が攻撃対象から得た情報の中でアラートから抜き出せない情報を被害予測システム側で調べ，被害予測に用いる．**4.1.2** 節でパケットキャプチャエージェントが監視サイトのゲートウェイを出入りする全ての通信を記録することを述べた．通信内容を調べることで攻撃者が監視サイトから得た情報を被害予測システム側でも知ることができる．攻撃者が攻撃対象から得た情報の中でアラートから抜き出せない情報を被害予測で用いることができるため，攻撃者が今後実施する不正アクセスと被害予測システムが予測する不正アクセスの方向性が同じになる．したがって，攻撃者が得た情報を被害予測に用いることで，予測結果に明らかに実施されるはずもない不正アクセス名が含まれることを防げる．

なお，攻撃者が攻撃対象から得た “OS” や “Software” の種類やバージョンに関する情報や攻撃対象でサービスを提供している “Port” に関する情報を監査データから被害予測システム側で調べる．

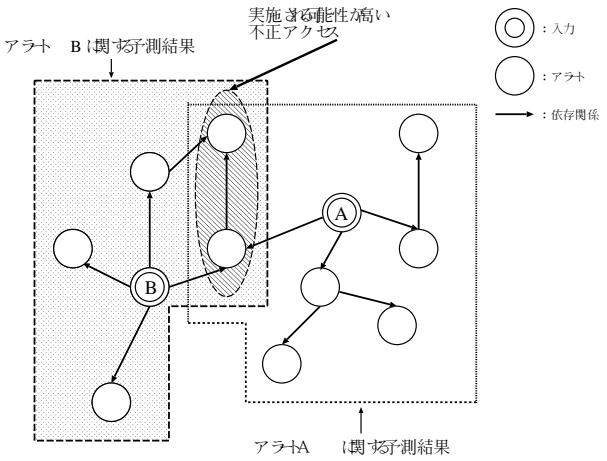

図 10: イベント依存モデルの作成例

4.3.2 被害予測システムの動作

イベント依存モデルを用いて被害を予測する場合の動作例を示したものが図 10 である．アラートを被害予測システムへ入力し，**4.3.1** 節で述べた予測法にしたがってイベント依存モデルを作成する．

入力したアラートを依存元として依存関係が認められたアラートに対応する不正アクセスが今後実施される可能性がある．依存関係が認められた全てのアラートに対応する不正アクセスを予測結果として監視サイトのネットワーク管理者に通知する．予測結果を通知する際に実施される可能性が高い不正アクセスとその対策を優先して通知する．なお，今後実施される可能性が高い不正アクセスであるかどうかは，イベント依存モデルを作成したときの予測の重なり具合で判断する．

4.3.3 動作例

図 11 はイベント依存モデルを用いた被害予測システムの動作例を示している．一例として，Web サーバの一つである IIS(Internet Information Service) に含まれる脆弱性をもつサンプルファイルに関する情報収集行為を示すアラートである “WEB-IIS iissamples access” が出力されたとする．イベント依存モデルは脆弱性を含んだサンプルファイルを利用してさらに不正アクセスが実施されるといった不正アクセスの連続性から将来起こり得る不正アクセスとして “WEB-IIS htimage.exe access” と “WEB-IIS /iisadm-pwd/aexp2.htr access” を予測することができる．ここで，“WEB-IIS htimage.exe access” はファイルのパス情報を取得しようとする不正アクセスであり，“WEB-IIS /iisadm-pwd/aexp2.htr access” はパスワードを管理するサンプルファイルへアクセスしようとする不正アクセスである．この予測

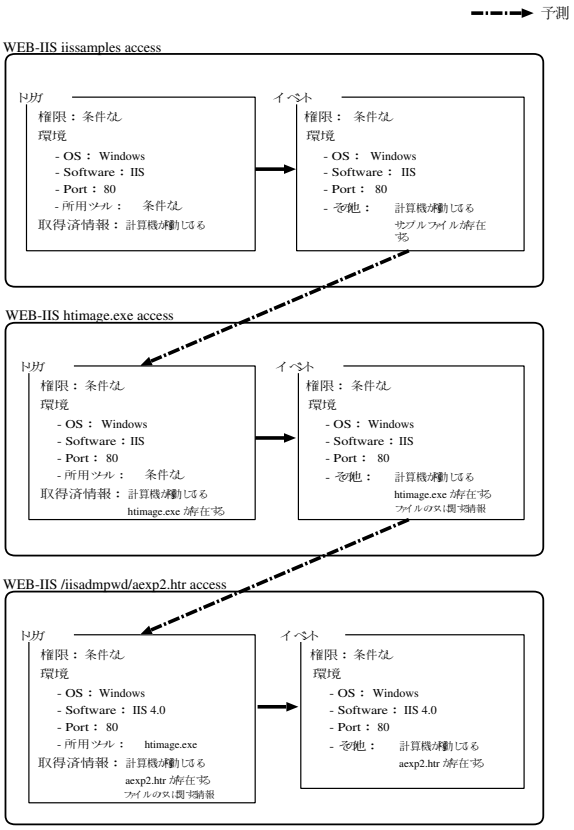

図 11: イベント依存モデルの動作例

結果をもとに，ネットワーク管理者はファイル “htimage.exe” と “awxp2.htr” を削除するといった対策を実施することで被害を未然に防ぐことが可能となる．

参考文献

[1] Yoshiaki Shiraishi, Youji Fukuta and Masakatu Morii, “Remote Access VPN with Port Protection Function by Mobile Codes”, The 4th International Workshop on Information Security Applications(WISA2003), LNCS2908, pp.16–26, Jeju island, Korea, Aug. 25-27, 2003.

[2] Yoshiaki Shiraishi, Youji Fukuta and Masakatu Morii, “Port Randomized VPN by Mobile Codes”, 2004 IEEE Consumer Communications and Networking Conference(CCNC2004), Las Vegas, Nevada, USA, Jan. 5-8, 2004.

[3] L. Lamport, “Password Authentication with Insecure Communication,” Commun. ACM, vol.24, no.11, pp.770–772, Nov. 1981.

[4] N. Haller, “The S/KEY One-Time Password System,” Proc. Internet Society Symposium on Network and Distributed System Security, pp.151–158, Feb. 1994.

[5] A. Shimizu, “A Dynamic Password Authentication Method Using a One-way Function,” System and Computers in Japan, vol.22, no.7, pp.32–40, 1991.

[6] A. Shimizu, T. Horioka, and H. Inagaki, “A Password Authentication Method for Contents Communication on the Internet,” IEICE Trans. Commun., vol.E81-B, no.8, pp.1666–1673, Aug. 1998.

[7] M. Sandirigama, A. Shimizu, and M.T. Noda, “Simple and Secure Password Authentication Protocol (SAS),” IEICE Trans. Commun., vol.E83-B, no.6, pp.1363–1365, Jun. 2000.

[8] C.L. Lin, H.M. Sun, and T. Hwang, “Attacks and Solutions on Strong-Password Authentication,” IEICE Trans. Commun., vol.E84-B, no.9, pp.2622–2627, Sep. 2001.

[9] T. Tsuji and A. Shimizu, “An Impersonation Attack on One-Time Password Authentication Protocol OSPA,” IEICE Trans. Commun., vol.E86-B, no.7, pp.2182–2185, Jul. 2003.

[10] T. Tsuji, T. Kamioka, and A. Shimizu, “Simple And Secure password authentication protocol, ver.2(SAS-2),” IEICE Technical Report, OIS2002-30, vol.102, no.314, pp.7–11, Sep. 2002.

[11] 北島忠征, 白石善明, 森井昌克, “サーバ認 証機能を有するワンタイムパスワード方式，” 第 26 回情報理論とその応用シンポジウム予稿集，Vol.2, pp.417-420，Dec. 2003.

[12] CERT: `http://www.cert.org/`

[13] JVN: `http://jvn.doi.ics.keio.ac.jp/`

[14] Nessus: `http://www.nessus.org/`

[15] “Web Server Survey”, `http://www.netcraft.com`.

[16] “*.com mail exchanger survey”, `http://cr.yp.to/surveys/smtpsoftware4.txt`

[17] “in-addr version distribution”, `http://www.isi.edu/%7Ebmanning/in-addr-versions.html`

[18] 毛利公美，曽根直人，高橋秀郎，神薗雅紀，森井昌克，“ネットワークサーバにおける脆弱性自動監査システム，” コンピュータセキュリティシンポジウム 2003 予稿集，pp.271-276，Oct. 2003.

[19] 毛利公美，高橋秀郎，広岡俊彦，曽根直人，森井 昌克，“ネットワーク資源に対する脆弱性自動監査システムの開発，” 信学技報，pp.-, May 2004.

[20] M. Roesch, “Snort: Lightweight Intrusion Detection for Networks,” Proc. of 13th Systems Administration Conference(LISA’99), pp.229–238, 1999.

[21] P. Porras and P. Neumann, “EMERALD: Event Monitoring Enabling Responses to Anomalous Live Disturbances,” Proc. of 9th ACM Conference on Computer and Communications Security, pp.245–254, 2002.

[22] 岡本 忠志, 白石 善明, 大家 隆弘, 森井 昌克, “なりすましに対する不正侵入検知システム (IDS-M),” 信学技報, OFS99-15, pp.39–46, 1999.

[23] Internet Security Systems, Inc., RealSecure intrusion detection system, http://www.iss.net/.

[24] Tripwire, Inc., Tripwire, http://www.tripwire.com/.

[25] KDDI 株式会社, セキュリティ監視サービス, *available at http://www.kddi.com/.*

[26] Internet Initiative Japan Inc., IIJ ネットワーク侵入検知サービス, *available at http://www.iij.ad.jp/.*

[27] Y. Tachibana, H. Takeuchi, H. Kurauchi and M. Morii, “Damage Analysis Support System for Illegal Access,” Proc. of 7th World Multi-Corference on Systems, Cybernetics and Informatics(SCI2003), Jul. 2003.

[28] 鴨田 浩明, 馬場 達也, 小久保 勝敏, 松田 栄之, 矢口 博之, “ニューラルネットワークを利用した不正アクセス被害予測方式,” コンピュータセキュリティシンポジウム 2002(CSS2002)，pp.131–136, Oct. 2002.

[29] Y. Shiraishi, T. Kuribayashi and M. Morii, “Center Management Type Intrusion Detection System,” Proc. of 7th World Multi-Corference on Systems, Cybernetics and Informatics(SCI2003), Jul. 2003.

[30] Snort, Snort Rules Database, *available at http://www.snort.org/snort-db/.*

[31] 栗林利光, 白石善明, 森井昌克 “イベント依存モデルによる不正アクセスの被害予測,” 2004 年暗号と情報セキュリティシンポジウム予稿集，pp.–, Jan. 2004.