ファイアウォール

～知り尽くされたはずの言葉が変わる！？～

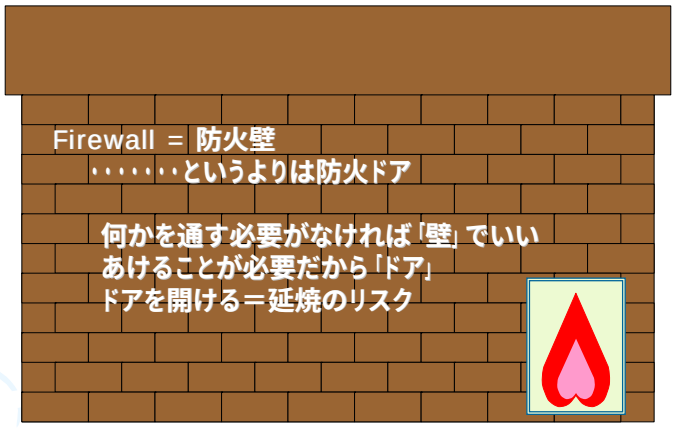
Internet Week 2004 – T1

FUTAGI, Masaaki (SSE)

最終版：11/30

ファイアウォールって何さ！？

- つまり‥



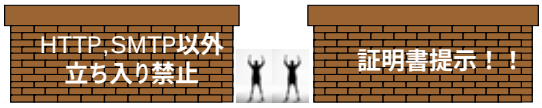
どうして、みんなこの絵を描くのだろう‥‥‥‥‥？？？？？？

ネットワークの検問所

Firewall = **検問所**

セキュリティポリシーの異なるネットワークを相互接続するためのセキュリティゲートウエイ

それぞれのポリシーを維持しながら通信する



ファイアウォールのおさらい

＊これまでのファイアウォールは、こんなもの「だった」！？

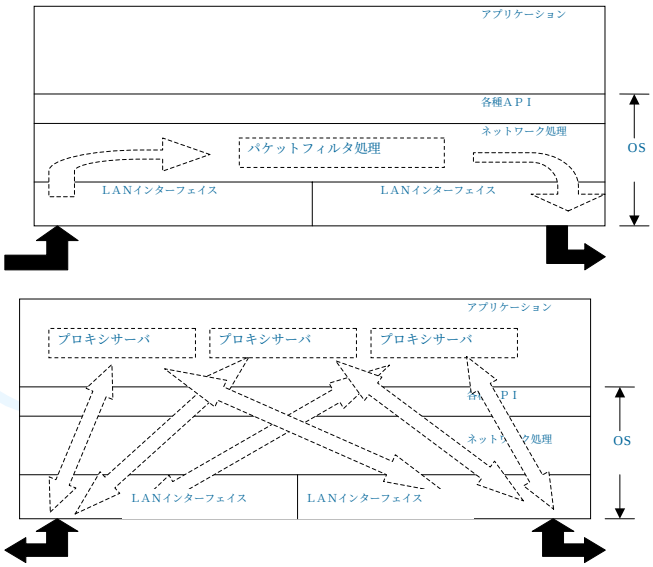
ファイアウォールの仕事

- 基本的な仕事（かならず備えるべき機能）
  - ルータもしくは中継装置としての仕事
  - 通過させていい通信かどうかの判断と通過させてはならない通信の排除
  - 危険な兆候の検出と警告
  - 通信の許可、不許可状況などの記録の保存

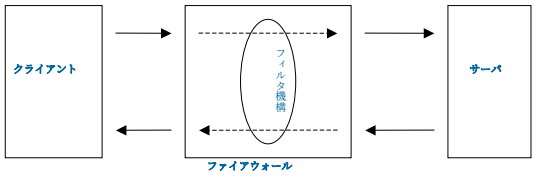
通信の中継機能

- 大別して2種類の方式がある
  - パケットフィルタ方式
    - ルータとしてIPパケットを中継することで、通信を行いたい機器同士が直接通信できる方式
  - アプリケーションゲートウエイ方式
    - Proxy（代理）サーバに一旦接続して、接続相手を指示して代理通信させる必要があるため、Proxy方式とも呼ばれる。
    - 直接的なパケット中継は行わず、要求を受けたProxyが相手方と通信して必要な情報を取得してから受け渡す方式。

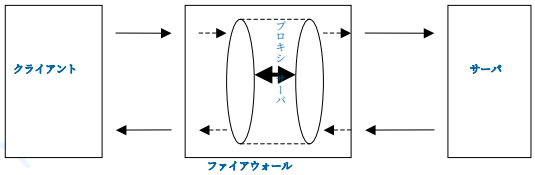
パケットフィルタとProxyの違い



パケットフィルタとProxyの違い



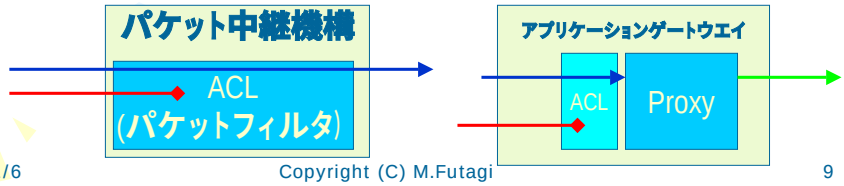
パケットフィルタ方式ではクライアントはサーバと直接通信を行う。
クライアント、サーバ間は単一の通信。



プロキシ方式ではクライアントはプロキシサーバと通信を行う。
クライアント、プロキシ間、プロキシ、サーバ間は独立した通信となり、
プロキシサーバは、その内部で通信内容を相互に受け渡す役割をする。

通信の許可、不許可

- 通信の発信元、相手先のIPアドレスやポート番号で許可、不許可を判断
  - ACL（Access Control List）の適用
    - パケットフィルタ方式では、フィルタ定義としてACLを適用する。
    - アプリケーションゲートウエイ方式では、Proxyサーバごとにアクセス許可情報としてACLを適用。



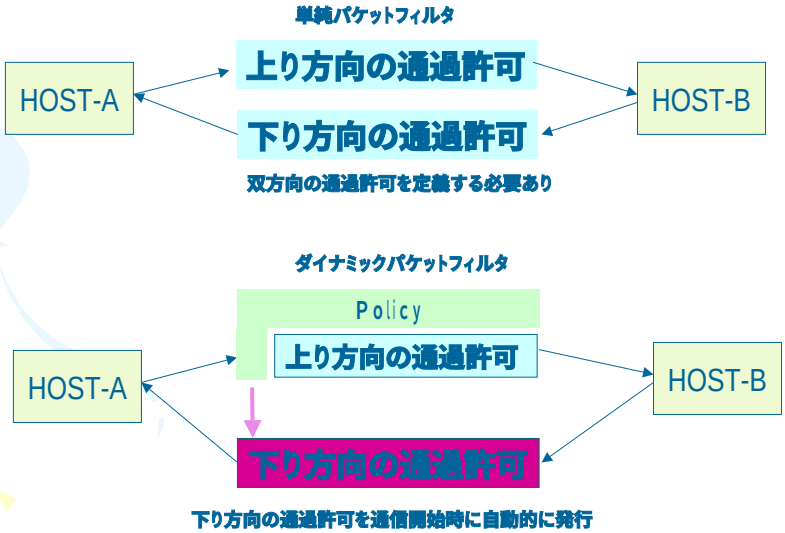
ファイアウォール関連用語・概念

- ダイナミックパケットフィルタ
  - (類) ステートフルインスペクション
- NAT (Network Address Translation)
  - (類) IP Masquerade, NAPT, PAT etc.
- DMZ (De-Militarized Zone)
- VPN (Virtual Private Network)
  - IPSec, L2TP, PPTP etc.

ダイナミックパケットフィルタ

- ファイアウォール製品とルータのフィルタ機能の最大の相違点
  - 通過を許可した通信パケットへの応答や付随する他のセッションなどを総合的に管理、自動処理を行う。
  - ポリシー設定を単純化できる。(許可するセッションの方向のみ定義)

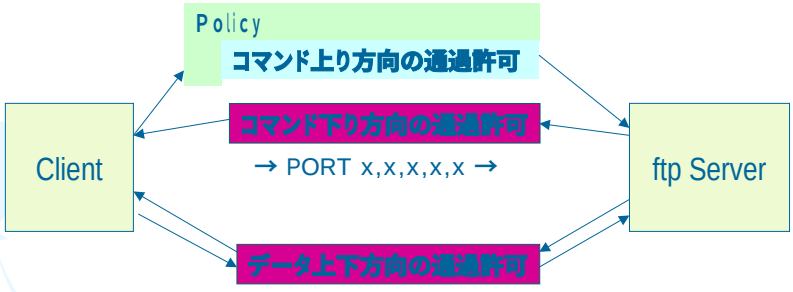
単純パケットフィルタとの比較



ダイナミックフィルタの特徴

- １コネクションのみで構成される通信は確実に対応可能
- 複数コネクション／セッションから構成される通信は対応できないものあり。（ストリーミング系の通信など）

FTPの場合のダイナミックフィルタ



FTP の通信は2つのコネクションから構成される。データコネクションの開設や使用するポート番号は、コマンドコネクション内でネゴされる。また、データコネクションはデータ転送のたびに新しいコネクションが生成される。

ステートフルインスペクション

- Checkpoint社オリジナルの用語
  - 本来は、単なるパケットヘッダのみのチェックではなく、アプリケーションレイヤまで、プロトコルをデコードして細部の検査ができる方式のこと。
  - 一般にはダイナミックフィルタと同義に使用されることが多い。C社以外のファイアウォールの場合、厳密にはこの言葉に該当しないものが多いが、ステートフルと称することが多い。

NAT, IP Masquerade, Etc.

- 内部アドレスにプライベートアドレスを使用したネットワークとインターネットの境界にファイアウォールを置く場合に必須。(除く、アプリケーションゲートウエイ型F/W)
- プライベートアドレスネットワークを起点とする通信がファイアウォールを通過する時点で、発信元をグローバルアドレスに変換する。

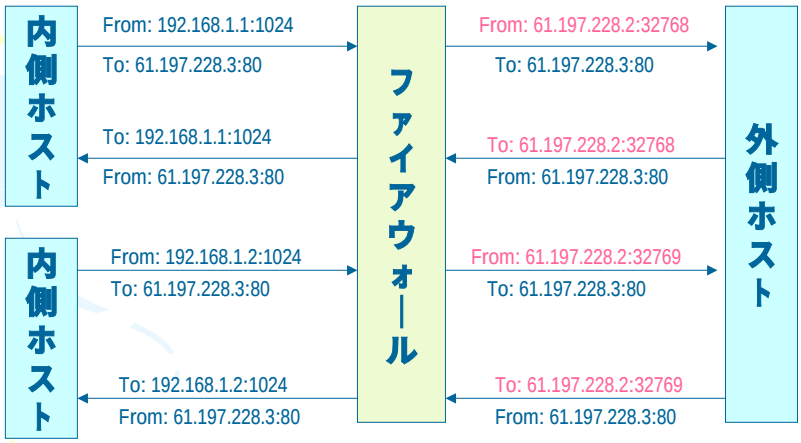
NAT （RFC1631）

- グローバルアドレスプールからアドレスを割り当て。
- 内部側ホストが外部と通信する際にプールからアドレスを一次的に割り当てて、アドレスを変換
- 同時通信数はグローバルアドレスの数に制約される
- 主に双方向通信を行う場合に利用

NAPT,IP Masquerade, PAT

- 1個もしくは少数のグローバルアドレスを多数の内部ホストで共有
- アドレス変換後のセッションが重複しないように発信元のポート番号も含めて変換
- 利用可能なポート番号数×アドレス数分の同時セッションをサポート
- 一部のプロトコルに対応が困難
- 外部からの着信は不可

NAPT／IP Masquerade



NAT使用上の注意点

- 複数のコネクションを使うプロトコルで対応できない可能性がある。（ダイナミックフィルタと同様の理由）
- データとしてIPアドレスを受け渡すようなアプリケーションの動作を保証できない。（FTPなどは一般に対応されているが、新しいアプリケーションでは未対応のものも多い）
- パケットヘッダの改ざんチェックを行うようなプロトコルに対応できない。（IPSecなど）

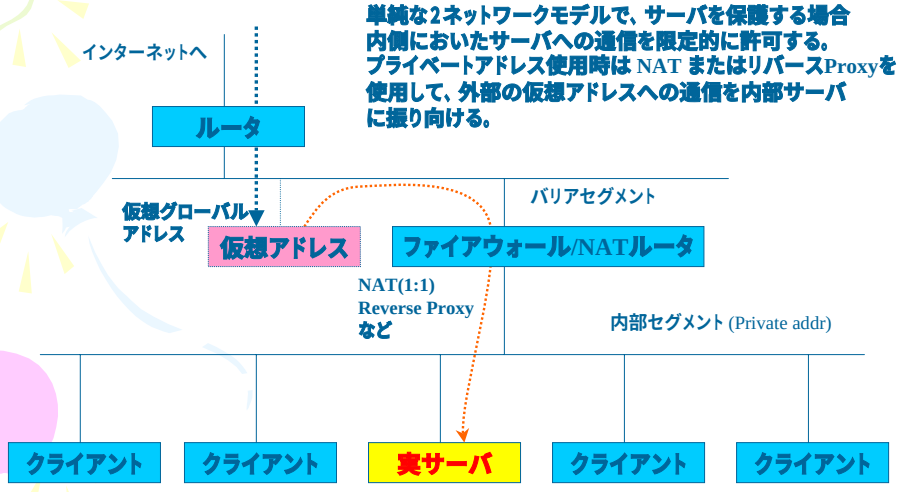
サーバ保護とDMZ

- DMZの意味
  - もともとは軍事用語
  - DeMilitarized Zone = 非武装地帯（直訳）
  - 直接侵入を防ぐための「緩衝地帯」的意味合いが強い（決して「非武装＝無防備」ではない）
  - 危機管理的考え方

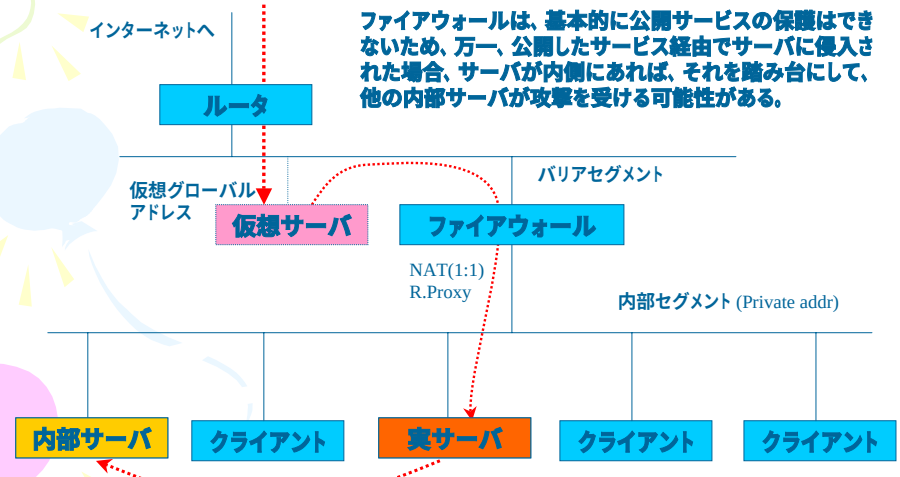
公開サーバの保護

- サーバをファイアウォール下に配置
- 外部からサーバに対して、公開するサービス以外へのアクセスを禁止
- しかし、公開サービスは通さねばならない
  - サーバの公開サービスに脆弱性があると、攻撃、侵入の可能性がある。
  - これをファイアウォールで防ぐことは難しい

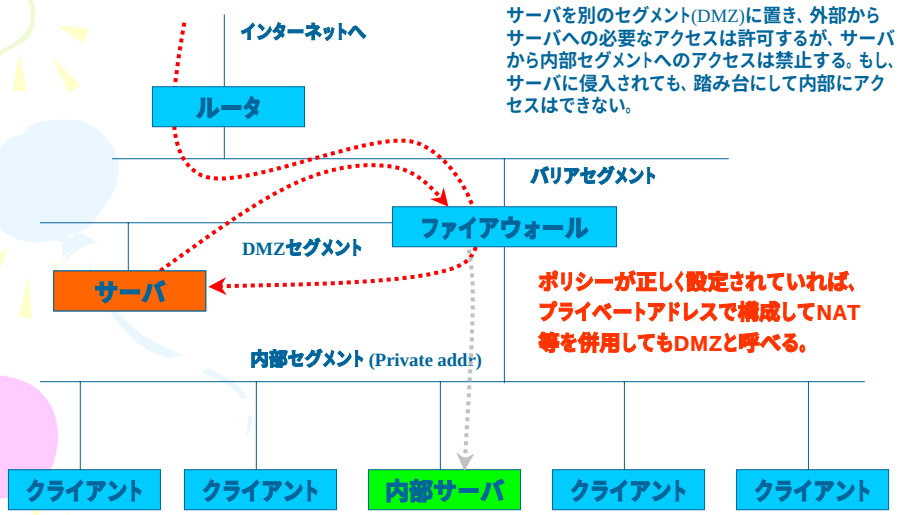
単純な保護モデルの場合



単純モデルで攻撃を受けたら



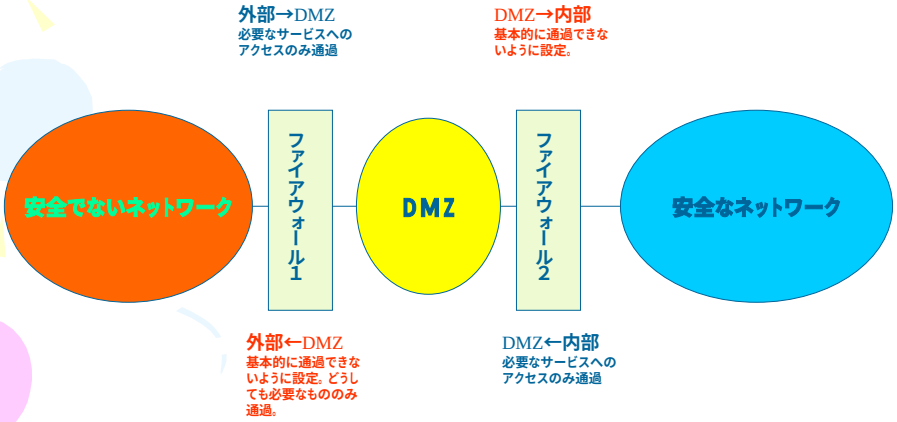
DMZモデルの場合



DMZを構成する意味

- **DMZは中間的な保護層**
  - 公開サーバ群をファイアウォールで保護し、必要以外のアクセスを排除する。
  - 万一、公開サーバが不正アクセスにより侵入されるなどの事態が生じても、そこから内部に直接入れないようにすることで、安全性の向上をはかる。(不正アクセスに対応する時間をかせぐ)
  - さらに、外部へのアクセスも制限することで、侵入されたサーバを踏み台にして外部を攻撃することも困難にする。(かごの鳥作戦)
  - 不正アクセスによって深刻な事態に陥るような重要なホストは置かない。

DMZ本来の形



DMZを正しく理解するために

- DMZは「非武装＝無防備」ではない
- 正しくポリシー設定しなければDMZではない
  - 外部からDMZへのアクセスは必要なものに限定
  - DMZから内部へのアクセスは原則不許可
  - DMZから外部へのアクセスも必要最小限に限定

VPNとファイアウォール

- VPNゲートウエイ機能
  - インターネットなどの安全でない（セキュリティポリシーの異なる）ネットワークを介して、安全にネットワーク間接続を行う。
  - VPNゲートウエイは相手側のネットワークに対するルータの役割をする。
  - ゲートウエイ間は暗号通信によって、通信の内容が保護される。
- セキュリティの観点から見ればファイアウォールに別のネットワークを追加接続したのと同じ意味合い。
  - 接続先ネットワークのセキュリティが破られれば、当然、リスクにさらされることに注意

VPNは「セキュリティ」ではなく「ネットワーキング」である

VPNの利用目的

- 同一組織のブランチ間のインターネット経由接続
  - 専用回線の代替えまたはバックアップ、回線費用の節約
- モバイルアクセスのコスト削減と安全性の確保
- 複数組織の協同ネットワーク（エクストラネット）構築
  - 回線費用の節約
  - インターネットの利用による柔軟性の確保
- 組織内のネットワークセキュリティーの階層的強化
  - 組織内 LAN のセキュリティー階層化
  - セキュリティーの低いネットワークを使って重要なネットワークを接続
  - ワイアレスLANのセキュリティ強化策

VPN利用時の注意点

- VPNは安全か？
  - 通信方式は安全でも、相手によっては接続すること自体に問題が生じることに注意（ポリシー設定による利用制限やサービス単位の認証はきちんと行う必要あり）
- パケットサイズ（MTU）に注意
  - カプセリングを行うことで最大パケットサイズ(MTU)が実質的に減少するため、フラグメントが発生する可能性あり。（Path MTU Discovery などによる動的対応、またはホスト側のMTU制限で回避）
- NAT越えの場合、通信できない場合あり
  - IPSec の場合、特殊な方法（NAT Traversal）を使用する必要がある。

ファイアウォールの運用・管理

- ログは宝の山
  - 定期的な解析を・・・・
    - 異常ログの検査
    - 通信傾向の掌握（各プロトコルの利用状況など）
  - ファイアウォールのログだけでも・・・・・
    - 特定プロトコルセッションの異常増加→ワーム侵入？
    - 拒否ログの増加→ポートスキャンなどの偵察活動？
    - DMZ から外部への接続→サーバ上での不正行為？
    - などなど・・・・

たとえば・・・

- バックドア、トロイの木馬検出
  - 内部側から外部の悪意あるサイトへのHTTPやSSLを使った通信を発見するには・・・
    - 相手は多くの場合「踏み台」サイト
    - 「踏み台」はマイナーなサイトが多い。
      - つまり、ふつうは誰もアクセスしないはず
- ファイアウォールのHTTPやHTTPSのセッションログを取得
- ログ解析ソフトを使って、相手先、発信元アドレス別に集計
- アクセスランキング最下位から順に相手方サイトをチェック

従来型の

ファイアウォールがもたらす安全とは

- 基本的にはIPアドレスやサービスをベースにした通信の到達性の制御
- つながるか、つないでいいかの制御
- つないだことの記録、つながなかったことの記録
- アクセスされる必要のないホスト、サービスに到達できなくなること。

従来型の

ファイアウォールが苦手なこと

- 通信内容の厳密なチェックと内容による通信制限
  - 特にパケットフィルタ系はこれが苦手（複雑な処理は負担が大きいため）
  - アプリケーションゲートウエイ系はこうしたことも可能だが、スピードはそれなり。
  - オール・イン・ワン型もあるにはあるが・・・・
  - 基本的に通過させたサービスに関する保護はサーバ側で行うのが基本となる

ファイアウォール：参考資料

- 過去の IWチュートリアル資料など
  - http://www.kazamidori.jp/SECURITY/index.html
- 情報セキュリティプロフェッショナル教科書
  - JNSA 監修：12月に出版予定・・・・
  - 株式会社秀和システム刊
- Stateful Inspection 資料（CheckPoint社）
  - http://www.checkpoint.co.jp/products/technologies/stateful_inspect.html

今、そして、これから・・・・

ファイアウォール製品はどこへ向かうのか・・・・・・・・

ファイアウォール製品の付加機能

- ウイルス対策機能
  - 電子メールに対するウイルス対策
  - Webアクセスやファイル転送に対するウイルス対策
- 侵入検知と防御機能
  - IPS (Intrusion Prevention System)機能
  - Application Firewall （アプリケーション防御）

ファイアウォールとウイルス対策

コンピュータウイルス･ワームって？

- コンピュータ、情報機器を標的に、それに侵入して望まれない動作をさせてしまう「悪性プログラム」(Malicious Code)
  - プログラムファイル等に寄生して、その動作を乗っ取るもの ＝ ウイルス
  - 単独のプログラムとしてコンピュータ内で動作するもの ＝ ワーム
  - 侵入の手引き、情報持ち出し･･･ ＝トロイの木馬
  - 最近では「ウイルス」とひとくくりにされてしまうことも多い

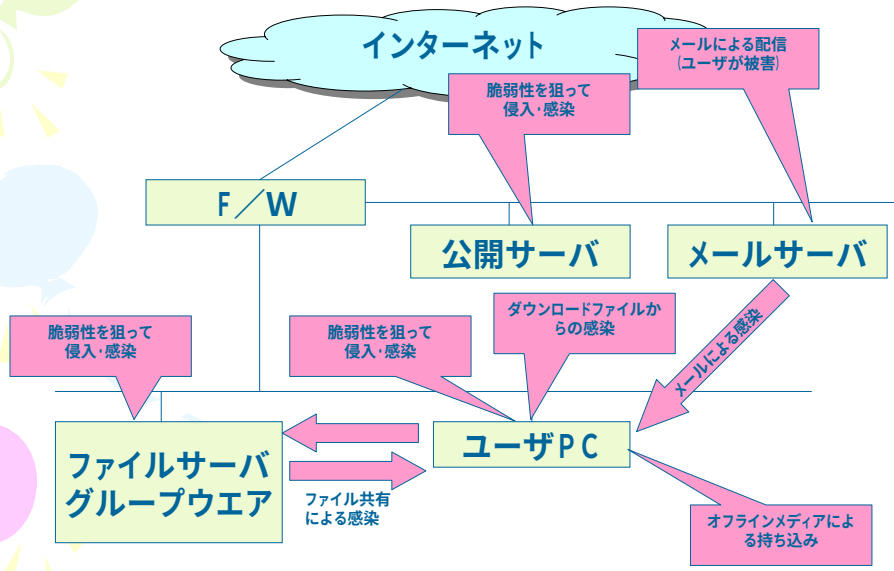
ウイルス・ワーム感染のタイプ

- ファイル媒介型
  - 特定のファイル（プログラムやスクリプト）に自分を組み込んで、誰かがそれを他のコンピュータに持ち込むのを待つもの
- 電子メール媒介
  - 主に、電子メールの添付ファイルに組み込まれ、それを実行すると、そのコンピュータ内に感染するもの。多くの場合、感染すると、そのコンピュータ内に格納されているメールアドレスを使って大量のウイルス付きメールを送信する。
- Web媒介
  - ワームなどのケースでは、侵入したWebサイトのページに自分自身を組み込んで、ブラウザの脆弱性を利用して参照したユーザＰＣに感染するものもある。
- 自動感染（脆弱性攻撃）型
  - 人手を介さずに感染を広げる、最も危険なタイプ。主に特定のアプリケーションやプラットホームに残された脆弱性（セキュリティホール）を攻撃して、そのコンピュータの制御を奪い、侵入する。侵入したコンピュータを踏み台にしてさらに感染を広げるものが多い。感染、流行の速度が非常に速いのが特徴。

ウイルス・ワーム侵入経路

- オフラインでの侵入（古典的経路）
  - FD, CD(R,RW),MO,USB-Device ...
- オンラインでの侵入
  - 電子メール（主に添付ファイルからの感染）
  - Webアクセス（ファイルダウンロード、不正なスクリプト読み込みなど）
  - ファイル共有（Windows, NFS, P2P ...）
  - 脆弱性攻撃による自動感染

ウイルス･ワームの侵入経路



ウイルス･ワームのターゲット

- 主にユーザ（クライアント）ＰＣやファイルサーバなど
  - 電子メール、ファイル感染系のウイルスは明らかにユーザＰＣへの感染が目的（混乱誘発、業務妨害、情報漏洩･･･）
  - 自動感染型は対象を選ばない。（脆弱性があれば、どれでも感染）

ゲートウエイにおけるウイルス対策の意味

- 各ＰＣやサーバにウイルス対策ソフトが入っていればいいのではないか･･･？という疑問
  - すべてのＰＣを確実に管理できますか？（最新のパターンファイルに即時更新できますか？）
  - すべてのＰＣユーザが対策ソフトのアラームに適切に対処できますか？
  - 大流行時にヘルプデスクがパニックになったりしませんか？
- ウイルス流入経路の８割以上が電子メール
  - メールサーバもしくはその前でウイルスを排除できれば、かなりリスクを減らすことができる
  - それでも、ユーザＰＣには対策が不可欠（オフライン感染、持ち出し感染の問題）

現在のウイルス対策モデル

- 多段構成の防御
  - インターネットとの接続点での流入、流出阻止
  - ファイルサーバ、グループウエアなどを介した蔓延の防止
  - ユーザＰＣへの直接感染防止
- 集中管理
  - すべてのパターンファイル更新の管理
    - でも現実に１００％更新は困難
  - ウイルス検出状況の管理
    - でも、「感染」の検知は困難････（だって、これは予防策）

ファイアウォールでの実装

- ウイルス対策サーバと連携するもの
  - CVP (Content Vectoring Protocol) での連携
  - iCAP (Internet Content Adaptation Protocol) での連携 (HTTPなど)
- 自前でエンジンを搭載するもの
  - 内部的にはProxyとして実装されたものが多い
  - パケットフィルタ系への実装は難易度が高い

ファイアウォール実装の問題点

- 負荷の問題
  - 大量のウイルス検索を瞬時に行う必要がある
  - CPU能力やメモリ容量の多くを、この処理に消費する可能性が高い
  - 場合によっては、基本的なファイアウォールの性能に影響が出る可能性も
- 「遅い」・・・・
  - HTTP、FTP などのウイルス検査を行うと、方式によっては、「急に遅くなったように感じる」ことがある
    - (ファイアウォールに限らない、ゲートウエイでの対策固有の問題)
    - 検査が完了しないと、流せない・・・という宿命
    - ファイアウォール側の方式やクライアント側のソフトによっては、タイムアウトしてしまう場合も・・・

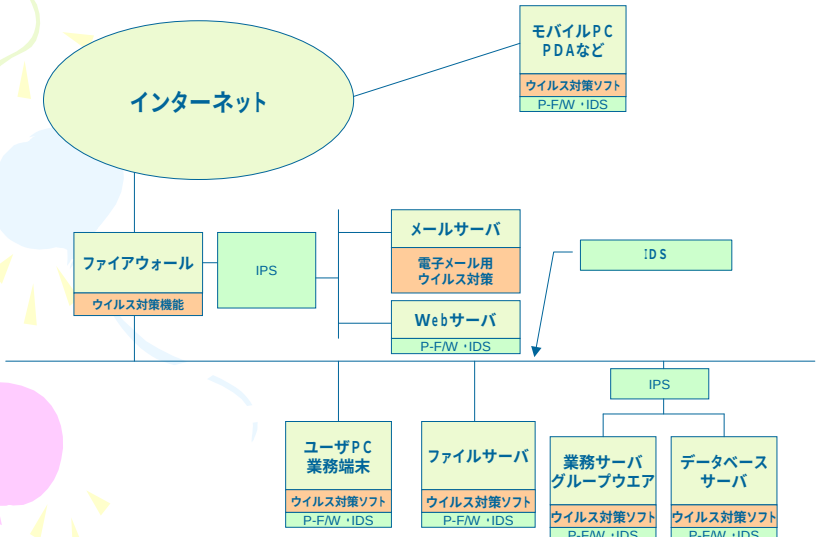
ファイアウォールを使うべきか

- 基本的には、ケース･バイ･ケース
  - 分けるにこしたことはない・・・・
    - 餅は餅屋・・・
    - ボトルネック回避・・・・
  - 中小規模のサイトにはコスト面から見て良いかも
    - 負荷が大きくなりすぎないことが前提
    - H/W化など性能面での向上も期待できる
  - 大規模サイトにはあまりおすすめしない
    - そもそもただでさえ、ファイアウォールの負荷が高い
    - ウイルスの大流行時にファイアウォールがDDoS状態に陥りかねない
    - お金があるサイトはウイルス対策サーバを買おう！！

ウイルス対策の限界も知っておこう

- 「後手」の技術である
  - 新種の悪性プログラムにはすぐに対応できない
    - 最近のウイルス･ワームは流行のスピードが速い
    - パターンファイル提供は後手にまわる（発見後1時間から数時間くらいはかかる）
    - 「未知ウイルス発見機能」は完全ではない
- 「先手」対策も組み合わせて使おう
  - 脆弱性の排除（パッチ、ワークアラウンド）
  - 侵入検知･防御対策の併用

情報システムとウイルス･ワーム対策



メール拡散型ウイルス対策

- 最近のウイルスは
  - 自分自身でSMTPエンジンを持っている
  - 自分の組織のメールサーバを使わず、直接相手側のメールサーバと通信してウイルスを送り込む
  - 従って、一般のクライアントから直接、大量のSMTPコネクションがファイアウォールを通過して外部に対して発生する
- 直接のSMTPをファイアウォールで禁止しよう
  - 新種ウイルス感染の可能性は小さくない
  - 世間様に迷惑をかけないようにしよう
  - 仕事上のメールは自社メールサーバ経由で出そう

ウイルス対策：参考資料

- IPA（情報処理推進機構）ホームページ
  - http://www.ipa.go.jp/security/isg/virus.html
- ITmedia 特集記事
  - http://www.itmedia.co.jp/enterprise/special/0407/virus/

☺ちょっと休憩 ☺

侵入検知と防御

Intrusion Detection / Prevention

セキュリティホール

- セキュリティ上の脆弱性 (Vulnerability)
  - OSやサーバソフトウエアのバグによるもの
  - システムのミスコンフィグレーションによるもの
  - アプリケーションのバグによるもの
  - これらの設計ミス

セキュリティホールの影響

- システムへの侵入・乗っ取り
  - コマンド実行権、システム管理権限の奪取
  - 任意のコードの実行
- 情報窃取や改ざん、破壊
  - アカウント、パスワード情報の盗用
  - データベース上の顧客情報、業務情報の窃取
  - サイトの詐称やなりすまし、詐欺的行為
- サービス妨害行為
  - システムダウンを引き起こしたり過負荷を発生
  - ワーム、ウイルスの拡散

侵入検知
(Intrusion Detection)

- 「侵入」ではなく「攻撃」を検出する手法
  - 攻撃コード (Exploit) の特徴 (signature) を利用する方法
  - 脆弱性 (vulnerability) 自体の特徴 (signature) を利用する方法
  - 不正な挙動 (behavior)を検出する方法
    - 利用ポリシーへの違反など
  - 統計的な異常 (anomaly) を検出する方法
    - 日常とかけはなれたトラフィックの検出など

IDS（侵入検知システム）と問題点

- アラームの信頼度が低い
  - 特定の脆弱性や攻撃コードの特徴検出
    - Signature の構成やチェックの深さによっては「誤認」「見落とし」が生じる
    - 攻撃であっても「有効」なものかどうかの判断ができない。（ノイズ的アラームが多い）
    - 基本的には「既知」の脆弱性もしくは攻撃が対象
  - 統計的異常の検出
    - 「既知」の脆弱性への攻撃に限らず検出できる可能性はあるが、確率的（非確定的）である
- アラームを受けたら行動を起こすのは人間
  - 誤報排除や緊急対応
  - IDSは「対策」にあらず
    - IDS + IRT（インシデント対応チーム）

ガートナーレポートの衝撃

Information Security Hype Cycle (2003/06)

- （ネットワーク型）IDS に将来はない
  - 多くのユーザはノイズ的アラームに辟易している
  - プロテクション機能が貧弱
  - もう誰も買わないだろう（投資に見合う効果なし）
- これからはファイアウォールの時代

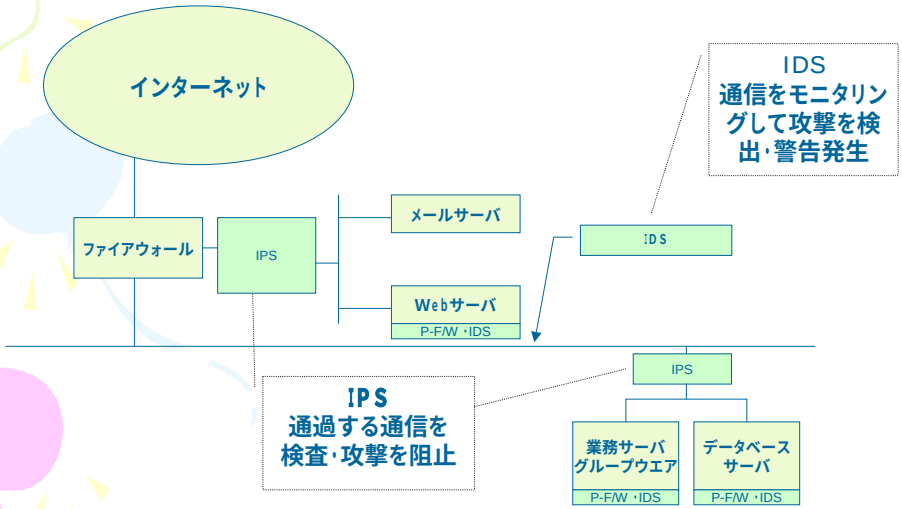
「言い過ぎ」、ではあるが重要なポイントを含んでいるのは確か。
もちろん、IDSはなくならないが・・・・

ファイアウォールという言葉の再定義が必要になる。

侵入検知・防御システム(IPS)

- 検出したら止めてしまおう、という発想
  - 「誤認」の減少によって可能に
  - ノイズ軽減のためのチューニングは必須ではない(無効な攻撃でも止めて問題なし)
    - (但し、パフォーマンスへのインパクトは考慮が必要)
- インライン型(ファイアウォール的)導入モデル
  - 従来のIDSのTCPリセットやFW連携機能では不十分
    - 単発パケットやTCP以外での攻撃を防御困難
    - インライン型ならば検査完了までパケットを保留しておける
  - ネットワークモニタはパケットロスを発生
    - インライン(ゲートウエイ)型は、少なくとも通過させた全パケットをモニタ可能
      (ドロップが発生したら、相手に届かないという意味で)
    - 遅延やボトルネックの危険性とのトレードオフではあるが・・・・

IDSとIPS



これってファイアウォールじゃ・・？

- IPSは一種のファイアウォール
  - 従来よりも、より深い検査が可能なファイアウォールと言える
  - ファイアウォールに取り込まれるべき機能とみるべきかもしれない（実際に組み込まれつつある）
    - ガートナーが言う「ファイアウォール」の形
    - 性能とのトレード･オフ問題は？（MPU高速化やASIC処理によるH/W化で克服可能？）
    - 障害耐性問題は冗長化による対応で充分か？
    - 基幹部分は従来型、内部サブネットやDMZ保護はIPS機能付きという使い分けも

今のIPS機能で保護できるもの

- 市販のプラットホーム、アプリケーションに対する攻撃からの保護
  - 「既知」の脆弱性に対する「既知」の手法を用いた攻撃（Exploit ベースの検知）
  - 「既知」の脆弱性に対する一般の攻撃（脆弱性ベースの検知）
  - アノマリー検出によるDoS攻撃などの緩和（但し、限定的）

・ユーザが開発したアプリケーションは対象外
・未知（未公開）の脆弱性に対する攻撃には対応できないと考えた方がよい
・既知の脆弱性でも、新たな種類の攻撃方法には対応できない場合もある

IDS/IPS：参考資料

- 日本Snortユーザ会ホームページ
  - http://www.snort.gr.jp/
- 主なIPS製品
  - Juniper (旧Netscreen) IDP
    - http://www.juniper.co.jp/products/intrusion/
  - McAfee (旧Intruvert) IntruShield
    - http://www.mcafeesecurity.com/japan/products/intrushield.asp
  - TippingPoint Unity One
    - http://www.tippingpoint.com/

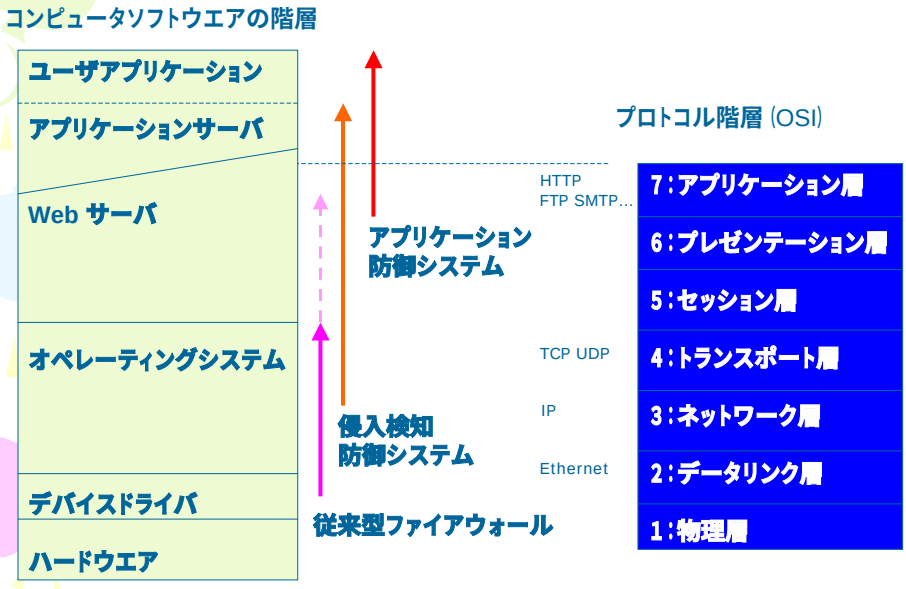
Beyond the IPS

- それでもまだ、攻撃対象は残る・・・・・・
  - たとえば、Web アプリケーションには・・・・・
    - ソフトウエアだから、バグはある・・・・・・
    - 設計ミスだってある・・・・・・
    - コンフィグレーションミスだってある・・・・・・
    - 思わぬ裏口だってある・・・・かも・・・・
  - つまり攻撃可能なセキュリティホールはまだ存在する
    - IPS では、ユーザアプリの脆弱性までは面倒みきれない
- 「Layer 7対応プロテクション」では不十分？
  - Layer 7の意味は？（プロトコル？、アプリケーション？）

アプリケーション脆弱性と防御

Another “Firewall”

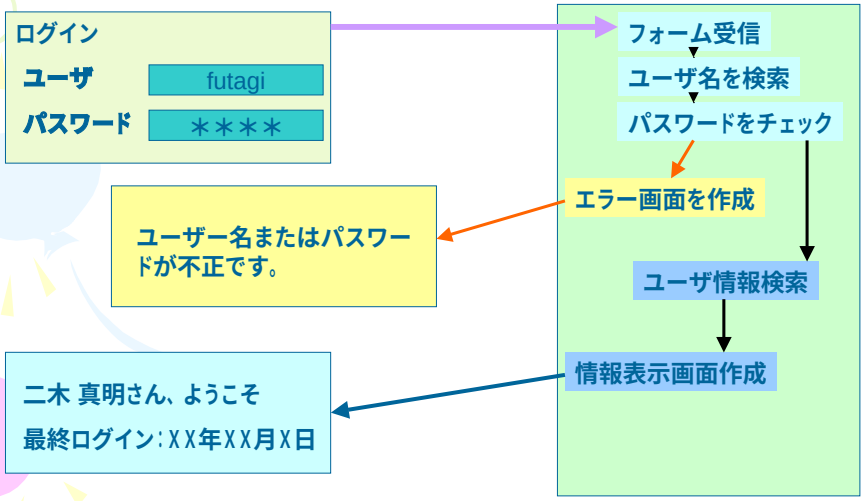
プロトコル階層とソフトウエア階層



アプリケーション脆弱性

- ユーザが開発したアプリケーション（業務用プログラム）のセキュリティホール
  - そのアプリケーションに固有の問題
  - すべてのアプリケーションに存在しうる
  - 特に、Web系アプリケーションに顕著
    - ステートレスプロトコルとセッション管理の問題
    - HTMLの柔軟性、ブラウザの高機能化を逆手にとった攻撃
    - 背後にあるデータベースへの攻撃

Webアプリケーションの仕組み



Webアプリケーションの処理

- 処理すべきデータの受信
  - フォームの受信 (GET/POST)
  - URL 引数による受け渡し
  - アプレット、スクリプト処理による受け渡し
- 応答ページの作成
  - 入力データに対する処理 (検索、その他のデータ処理の実行)
  - 入力データもしくは処理結果を使って応答ページ (HTMLデータ) を作成、クライアントに送信

危険が生じるポイント

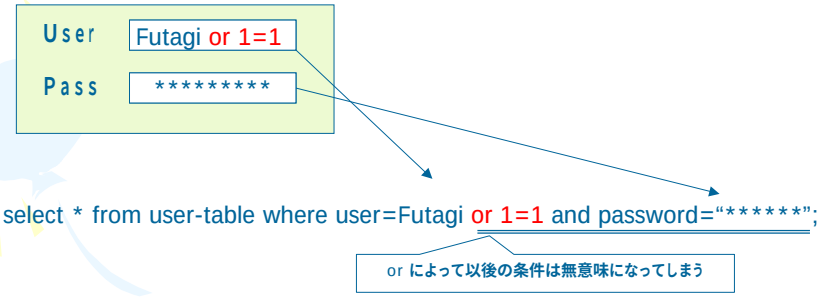
- 入力データの利用方法に注意
  - 入力データの一部を応答ページになんらかの形で転記する場合。
  - 入力データを検索条件にしてデータベース等を検索する場合 (SQLの構成)
  - 入力データを引数にして他の処理やコマンドを実行する場合
  - 入力データを使って、行う処理を選択する場合

たとえば・・・・

- 入力データを応答ページに転記
  - もし、入力データに<script ...>....</script> などのHTMLタグが含まれていたら・・・・
- 入力データをSQL分の検索条件に使う場合
  - たとえば、”futagi or 1 = 1 “ というような値を入力データに与えたら、どんな動作をするだろうか。
- 入力データをコマンドの引数に使う場合
  - たとえば、”; rm –f /usr” などというデータを与えられても大丈夫だろうか・・・・

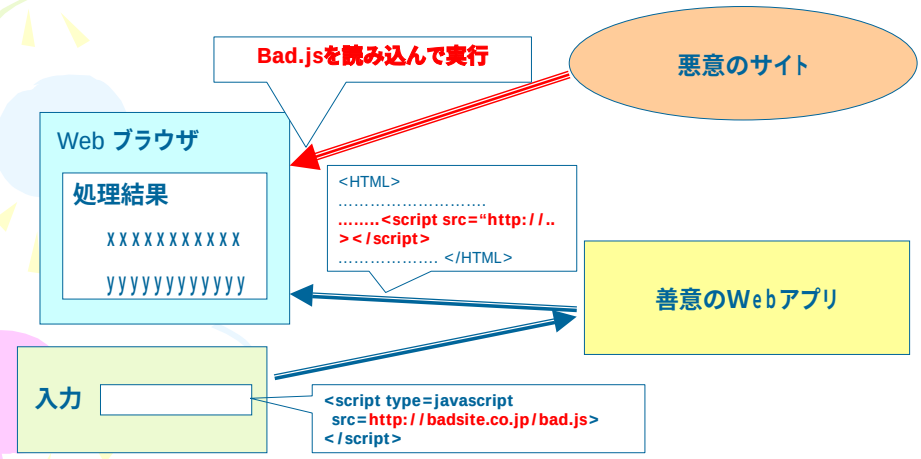
SQLインジェクションの例

生成する検索条件 select * from user-table where user=User and password=Pass;



＊結果として、パスワードに関係なく問い合わせが成功してしまう。

クロスサイトスクリプティング（ＸＳＳ）の一例



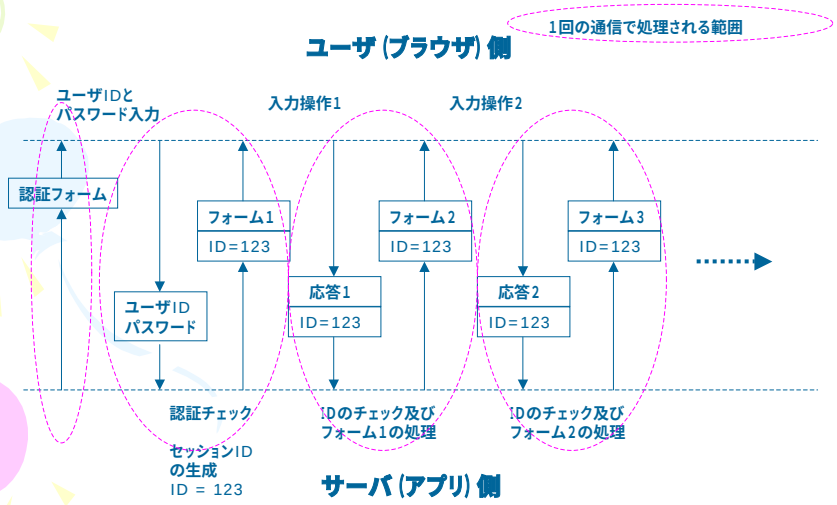
たとえば・・・・

- 隠し（hidden）パラメータを処理に使っている
  - フォームを保存し、パラメータを書き換えられて送信されても大丈夫？
- Cookie をセッション管理に使っている
  - Cookie の内容を見られても大丈夫？
  - Cookie の内容を改ざんして送られても大丈夫？
- 古いページやデバッグ用ページが残っている
  - リンクされていなくても直接アクセス可能では？
  - 思わぬトラブルや不正アクセスの元凶に

Webアプリとセッション管理

- Webを使ったアプリの特性
  - HTTPはステートレスなプロトコル（一回ごとに通信が簡潔）
  - 認証からセッション管理の機構をユーザがアプリケーションで実現しなければいけない
    - ログイン
    - 処理選択～処理の実行（繰り返し）
    - ログアウト

Webアプリにおけるセッション管理の例



手口のおさらい

- URL パラメータやフォーム
  - Hidden パラメータ、URLパラメータ改ざん
  - フィールドオーバーフロー
  - SQL インジェクション（フィールドへのSQLコマンドや記号の入力）
  - コマンドインジェクション（システムコマンドや記号の入力）
  - クロスサイトスクリプティング（HTMLタグやスクリプトの挿入）
- Cookie、セッションパラメータ
  - 内容の窃取、改ざん
  - セッションの乗っ取り
- 隠し（消し忘れ）ページやファイル
  - ありがちな名前（debug, admin .....）で検索・・・・・など
  - 文書と一緒におかれたデータファイル（password.dat ....）

アプリケーション攻撃対策

- 第一義的に、「みつけて」「修正する」こと・・・だが
  - バグはなくならないし、発見は難しい
  - 修正→テスト→リリースには時間がかかる
  - 古いアプリを直せるか・・・・（現実問題）
- 攻撃を食い止める方法は・・・
  - 対策は個々のアプリケーションに依存する
  - 通信の内容の細部までをチェックできれば・・

アプリケーションファイアウォール

- アプリケーション保護に特化したF/W
  - URLやフォームのようなアプリケーションに与えられるデータ、パラメータを監視
  - Cookie やセッションパラメータの監視、保護
  - Web Service (SOAP) などの通信の監視
  - データベースへのリクエストの監視
- アプリケーションに対するIPSとして機能。
  - シグネチャではなく、原理（方法論）的な防御

アプリケーションファイアウォールの技術的困難さ

- 多種多様なアプリケーションへの対応
  - ページ、フィールドごとに監視・保護ポリシーが異なる場合が多い
  - 設定が煩雑化しがち（サイトに依存して設定項目が多い）
- 性能面での問題
  - プロトコルの7階層の処理に加えて、さらに深いチェックが必要。ボトルネックになる危険性

課題へのチャレンジ

- ポリシー、設定の煩雑化
  - アプリケーションの自動学習機能
    - 一定期間の通信監視によるもの（受動型）
    - アプリケーションスキャナによる調査（能動型）
- 性能面の問題
  - PCサーバH/Wの高性能化　（消極策）
  - ASIC等の専用H/W化　（積極策）

アプリケーションF／Wの将来

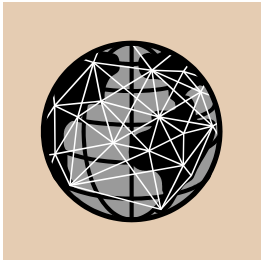
- 最終的には「ファイアウォール」として統合されていくであろう
  - たとえば、従来型の機能を併せ持つDMZ用ファイアウォールといった形
  - アクセスの多いサイトに使うには、性能面と管理のしやすさがポイント

アプリケーション保護：参考資料

- IPA セキュアプログラミング講座
  http://www.ipa.go.jp/security/awareness/vendor/programming/index.html
- IPA アクセス制御機構の機能不全を検出・検証するシステム
  (株) ソフテック、 独立行政法人産業技術総合研究所 高木浩光氏
  http://www.ipa.go.jp/security/fy14/development/web-auth/test-tool.html
- OWASP Top 10 日本語訳 (高橋 聡氏 訳)
  https://sourceforge.net/project/showfiles.php?group_id=64424&package_id=70827
- OWASP Guide to Building Secure Web Applications
  http://www.owasp.org/documentation/guide/guide_downloads.html
- 安全なWebアプリ開発40箇条の鉄則
  独立行政法人産業技術総合研究所 高木浩光氏
  http://staff.aist.go.jp/takagi.hiromitsu/#2003.6.19
  http://java-house.jp/~takagi/paper/idg-jwd2003-takagi-dist.pdf

IPv6 とファイアウォール

ユビキタスネットワークの夢とセキュリティの葛藤・・・・・・・・・



IPv6にしよう・・・・・

- 事実上無限のアドレス空間
  - 世界中のすべての機器を繋いでもまだ余る・・・・
  - 可能なすべての機器をネットワークに・・・
  - 夢のユビキタスコンピューティング社会の実現
- 強化されたセキュリティ
  - IPSec の標準サポート（暗号通信の標準化）

IPv6で何が変わるか・・・

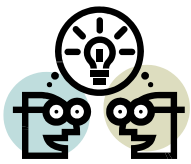
- 事実上無限のアドレス空間による「総グローバル化」（ユビキタスネットワークの真髄？）
- 接続されるデバイスの急増（数、種類）
- 上位層プロトコルの種類の増加（映像、音声系のストリーミング、家電等の制御、複数ホストの協調動作・・）
- IPSecの標準的な利用による暗号通信の一般化

セキュリティ屋の悩み

- ネットワーク上の様々なデバイスをどう管理するのか
  - そもそも、なんでもかんでも繋がって、管理できるのか
- 情報機器以外の機器を接続した時の安全性はどう確保しようか
  - 家電機器の貧弱なCPUにあまり多くのセキュリティ機能を持たせるのも・・・
  - お風呂を空だきされちゃったら嫌だよね・・・
  - 家電機器にもパッチが必要なのかな・・・・インターネットに繋がないと更新できなかったらどうしよう・・・
- IPSecで安全というけれど、逆に・・・
  - 通信の内容を監視できなくなるのは困る
  - IDS やモニタリングデバイスは「終わり」！？
  - 接続の管理、認証は個々のPCでやるの？

でも、IPv6はいいね・・・・だから

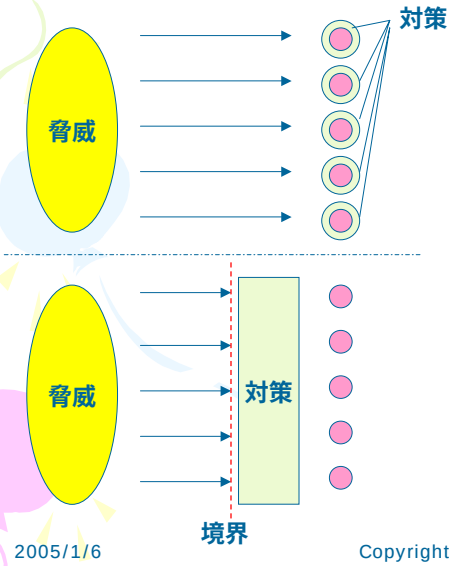
- どうやってセキュリティを守るかを考えよう
  - 既存の考え方で使えるものは何か？
  - 使えなくなるものの代替策は？
  - 新しく必要になるものは？



ファイアウォールの場合

- そもそも必要なのか、という議論
  - IPSecでエンドポイント間通信のセキュリティは確保可能
  - アクセスコントロールはすべて認証ベースでやるのか？
  - 企業内のすべてのエンドポイントのポリシーをきちんと集中管理できるのか？(もしできれば、それもあり？)
  - でも、Perimeter Defense (境界防御) の考え方は残る・・・
- IPS機能、アンチウイルスなど通信内容検査機能とIPSecの問題
  - そのままでは使えない
  - なんらかの対応または代替策が必要
- トラフィック量は？
  - 接続されるデバイスが増えるので、当然増加するはず
- 移行期におけるファイアウォール
  - v4,v6 共存環境への対応は？

Perimeter Defense (境界防御)

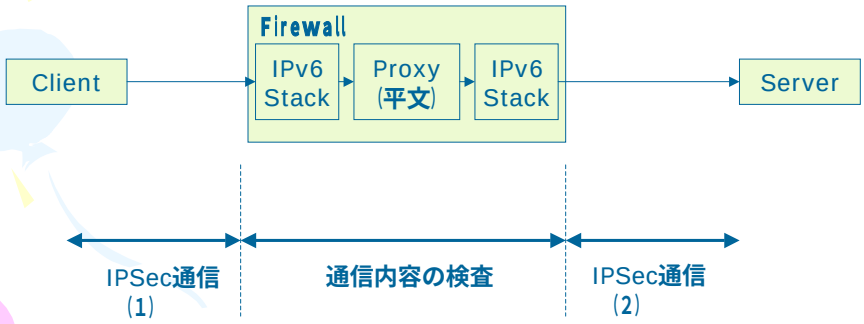


- 防衛線を作って、そこで戦おうという考え方
  - 街の出入り口を限定し、そこに検問所を作ることで、不審な人物の出入りを規制すれば、街のパトロールは簡素化できるはず。
  - ネットワークの場合は、入り口にあるファイアウォールと内部のネットワークの関係
- 個々のエンドポイントに対する脅威を境界でコントロールすることで、全体としてのリスクを軽減する
  - 主要なセキュリティ対策の局所化
- よって、ファイアウォールの概念はなくならない

IPSec問題とファイアウォール

- 通信内容の検査は本当に不可能か？
  - 鍵を持たせて復号（実用的かは？？）
  - いずれにせよ、ファイアウォール内を平文で通ればよい・・・・
- Proxy によるIPSec termination （**Just idea**!）
  - Proxy を使えば、通信は2つに分解される。
  - それぞれは別個のIPSec通信となり、データはProxy内部では平文
  - 問題は、透過的に使えるかどうか・・・（利便性の課題）
  - 個々のサービス単位でProxyを用意する必要性
  - エンドポイント間の相互認証はどうするかという問題も

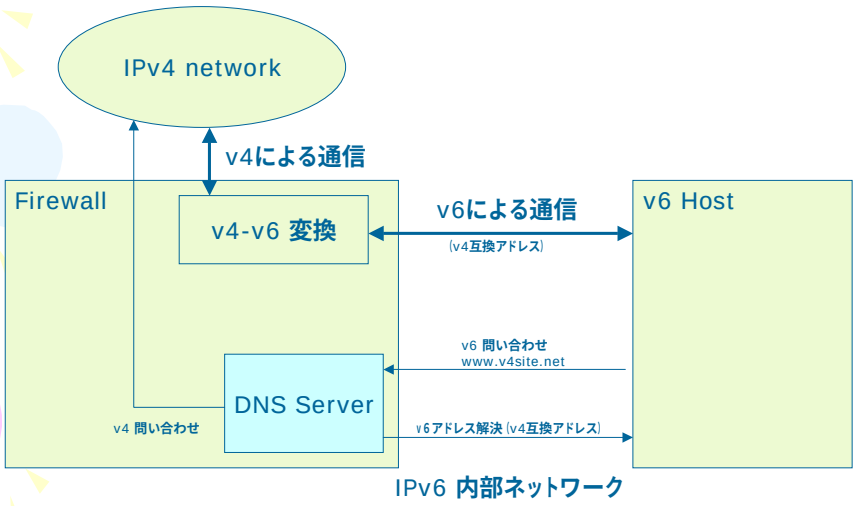
ProxyによるIPSec termination



移行期のファイアウォール

- v4, v6 の共存
  - デュアルスタック
  - ネイティブ、トンネル接続のサポート
- v4←→v6 変換（必要なのでは？）
  - 組み込み系でのデュアルスタックは負担が重いため、v6のみの実装になる可能性もある
  - しかし、インターネットの大半がv4の現状では、これらの機器はv4ネットワークとの通信も必要

v4,v6変換のアイデア



そしてファイアウォールは進化する

- 周辺機能の取り込みは続く
  - ファイアウォールメーカーの生き残り策
  - セキュリティ企業の合併、吸収の影響もあり・・
- 正しい進化と誤った進化を見極めよう
  - ユーザが淘汰していくしかない
- IPv6への対応を視野に
  - ファイアウォールの v6 対応はv6の推進力にもなりうる・・・

日本のメーカー、開発者にもっともっとがんばって欲しい！！！！

ご清聴ありがとうございました

ご質問？

最終版資料は、後日、IW2004サイト及び
以下に掲示する予定です。

http://www.kazamidori.jp/SECURITY/

ふたぎ　まさあき

住商エレクトロニクス(株)
ネットワークセキュリティ事業部
技術担当副事業部長

futagi.masaaki@sse.co.jp
futagi@kazamidori.jp