ケーススタディ（不正侵入の実態）

～UNIXシステムがターゲットとなった不正侵入について事例など～

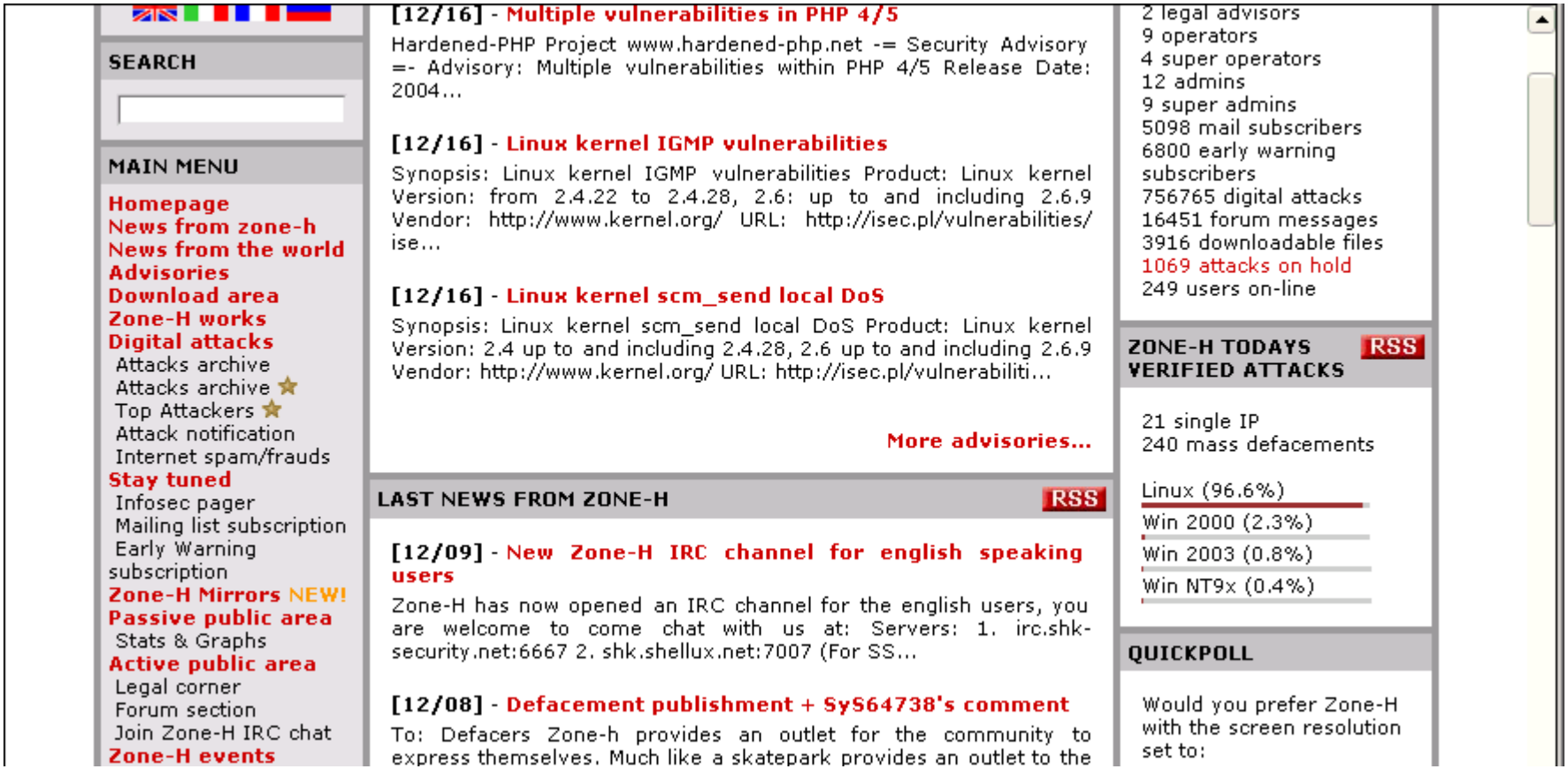
インターネット セキュリティ システムズ株式会社
CIO / エグゼクティブ　セキュリティ　アナリスト
高橋　正和

INTERNET | SECURITY | SYSTEMS®

UNIXは安全か？

ZONE-Hの改竄情報

実は，改竄されたサイトの90%以上がLinux



少々古いですが。。。

- 2002年10月8日 Sendmail
  - メール・サーバー・ソフトsendmail 8.12.6のソース・コードを改ざんしたものが一時的に配布されてしまった
    - http://itpro.nikkeibp.co.jp/members/ITPro/SEC_CHECK/20021018/1/
- 2002年 8月2日　OpenSSH
  - CERT® Advisory CA-2002-24 Trojan Horse OpenSSH Distribution
    - http://www.cert.org/advisories/CA-2002-24.html
- 2002年11月13日　tcpdump / libpcap
  - CERT® Advisory CA-2002-30 Trojan Horse tcpdump and libpcap Distributions
    - http://www.cert.org/advisories/CA-2002-30.html

CVSにかかわる事件

- 2003/1/20
  - CVS contain a flaw that can be used by a remote attacker to execute arbitrary code on the server.
    - http://security.e-matters.de/advisories/012003.html
  - 2004-04-14: Stable CVS Version 1.11.15 Released! (security update)
  - Full-Disclosure] Advisory 07/2004: CVS remote vulnerability
    - http://www.st.ryukoku.ac.jp/~kjm/security/ml-archive/full-disclosure/2004.05/msg01005.html
- 2004年5月27日　More CVS woes
  - cvshome.org <-- PLAY "FIND THE SUCKIT" と書かれているわけだが、www.cvshome.org が現実に侵入されていた (rootkit が仕掛けられていた) ことが明らかになっている。関連: Handler's Diary May 27th 2004 (SANS ISC)。
  - http://isc.sans.org//diary.php?date=2004-05-27&isc=b706c70418829ce920c311bbebc41c04
- バージョン管理ツールCVSのセキュリティ・ホールを突いた不正侵入が続発
  - http://itpro.nikkeibp.co.jp/free/ITPro/NEWS/20040531/145138/
    - 5月24日 cvshome.org
    - 5月24日には，「もじら組」
    - 5月27日には，「namazu」の公式サイトが
    - 5月28日には，「Ruby」の公式サイト

比較的最近の事例

- 2003年11月20日 GNU/Debian Linux
  - ボランティア・ベースのLinuxディストリビューションであるGNU/Debian Linuxのサーバーに不正侵入
    - http://lists.debian.org/debian-announce/debian-announce-2003/msg00003.html
- 2004年1月 SouceForge.jp
  - オープンソース・ソフトウエアの開発環境を提供しているSourceForge.jp
    - http://sourceforge.jp/forum/forum.php?forum_id=4153
- 2004年3月23日　GNOMEサーバ
  - Linuxの代表的なデスクトップ環境であるGNOMEのサーバーwww.gnome.orgに不正侵入の痕跡が発見された
    - http://mail.gnome.org/archives/gnome-announce-list/2004-March/msg00114.html
- 2004年4月10日 MySQLユーザ会
  - オープンソース・コミュニティへの不正侵入が相次ぐ，日本MySQLユーザ会のサーバーも
    - http://itpro.nikkeibp.co.jp/free/ITPro/NEWS/20040412/142780/

Yahoo DDoSの例 (直接的にUNIX ではないですが…)

一般的にDDoS被害の情報が公表されることはほとんど無いが、2000年2月のYahooに対する攻撃については、Yahooの関係者と思われる人物の技術的な解説がインターネットに投稿されている。

http://packetstormsecurity.nl/distributed/yahoo.txt

1 DDoSの攻撃は、シングルポイントフェィリアと思われるルータがターゲットなった
攻撃者は、十分にネットワークトポロジを調査したものと思われる
2 攻撃を受けた際のトラフィックは、1.5Gbpsに及んだ
大量のトラフィックによりルータがダウンした
ルータを復旧したところ、上流へのルーティングが無くなっていた
この問題は、上流とのネットワークを全て切断することで回復した
3 状況の把握に時間がかかった
関係者がDDoSによるアタックを想定していなかった
パケットの収集が効果的に行えなかった
4 DDoSは、Smurfアタックである可能性が高い
サイト内でパケットをキャプチャしたところ、大量のICMPパケットが観測された
ソースがターゲットなったルータで、ディストネーションがローカルブロードキャスト(255.255.255.255)のICMPパケットを観測した。
5 フィルタリングを実施し150Mbpsまで軽減できた
ICMP ECHO REQUEST , ECHO REPLYだけでは不十分で、他のパケットについてもフィルタリングを実施
この結果、1.5Gbpsあったトラフィックを150Mbpsにすることができた

UNIXの代表的な脆弱性

狙われるUNIXの脆弱性 TOP10(SANS)

- U1 BIND Domain Name System
- U2 Web Server
- U3 Authentication
- U4 Version Control Systems
- U5 Mail Transport Service
- U6 Simple Network Management Protocol (SNMP)
- U7 Open Secure Sockets Layer (SSL)
- U8 Misconfiguration of Enterprise Services NIS/NFS
- U9 Databases
- U10 Kernel

http://www.sans.org/top20/

http://www.sans.org/top20/top20-v50-japanese.pdf

U1 BIND Domain Name System

- U1 BIND Domain Name System
  - CVE-1999-0009, CVE-1999-0024, CVE-1999-0184, CVE-1999-0833, CVE-1999-0837, CVE-1999-0835, CVE-1999-0848, CVE-1999-0849, CVE-1999-0851, CVE-2000-0887, CVE-2000-0888, CVE-2001-0010, CVE-2001-0011, CVE-2001-0012, CVE-2001-0013, CAN-2002-0029, CAN-2002-0400, CAN-2002-0651, CAN-2002-0684, CAN-2002-1219, CAN-2002-1220, CAN-2002-1221, CAN-2003-0914
- **(比較的) 最近発見されたBINDの脆弱性**
  - Cache poisoning via negative responses:
    - http://www.kb.cert.org/vuls/id/734644
  - For the Denial of Service Vulnerability in ISC BIND 9:
    - http//www.cert.org/advisories/CA-2002-15.html
  - For the Denial of Service Vulnerability in ISC BIND 8:
    - http:/www.isc.org/products/BIND/bind-security.html
- **対策**
  - Running the BIND9 DNS Server securely
    - http://www.boran.com/security/sp/bind9_20010430.html
  - Hardening the BIND v8 DNS Server
    - http://www.boran.com/security/sp/bind_hardening8.html
  - Securing an Internet Name Server
    - http://www.linuxsecurity.com/resource_files/server_security/securing_an_internet_name_server.pdf.

U 2 1/2
Web Server

- U2 Web Server Apache
  - CVE-1999-0021, CVE-1999-0066, CVE-1999-0067, CVE-1999-0070, CVE-1999-0146, CVE-1999-0172, CVE-1999-0174, CVE-1999-0237, CVE-1999-0260, CVE-1999-0262, CVE-1999-0264, CVE-1999-0266, CAN-1999-0509, CVE-2000-0010, CVE-2000-0208, CVE-2000-0287, CAN-2000-0832, CVE-2000-0941, CVE-2002-0061, CVE-2002-0082, CVE-2002-0392, CAN-2002-0513, CAN-2002-0655, CAN-2002-0656, CAN-2002-0657, CAN-2002-0682, CAN-2003-0132, CAN-2003-0189, CAN-2003-0192, CAN-2003-0254, CAN-2004-0488, CAN-2004-0492
  - iPlanet/Sun Java System Web Server
    - CVE-2000-1077, CAN-2001-0419, CAN-2001-0746, CAN-2001-0747, CAN-2002-0686, CVE-2002-0845, CAN-2002-1315, CAN-2002-1316
  - OpenSSL
    - CAN-2003-0543, CAN-2003-0544, CAN-2003-0545
  - PHP
    - CVE-2002-0081, CAN-2003-0097, CAN-2004-0594
  - Other
    - CAN-2004-0529, CAN-2004-0734
- **脆弱性チェックの方法**
  - Free: Nessus, SARA、Free Utilites(eEye)
  - Products: Internet Scanner (ISS), Comercial Scanner(eEye)

U 2 2/2
Web Server

- **対策**
  - **パッチレベル**
  - **不要な機能の無効化**
    - cgi**アクセス**、php**サポート**、mod_ssl, mod_proxy**等**
  - cgi-bin: **デフォルト、サンプルスクリプトの削除**
  - PHP**のセキュリティ保護** HTTP**ヘッダ情報開示の無効化、セーフモードでの稼動**
    - Securing PHP: Step-by-step
    - http://www.securityfocus.com/printable/infocus/1706
  - Apache mod_security**の利用**
    - Cross Site Scripting(XSS), SQL Injection**の防御**
    - http://ww.modsecurity.org
  - XSS, SQL Injection**の検査**
    - Nokto http://www.cirt.net/code/nikto.shtml
  - httpd**を**chroot**環境で稼動させる**
  - httpd**をユーザ権限で実行**
  - **情報開示の制限**
    - **例：レスポンストークン、**mod_info**のコントロール、ディレクトリインデックス**
  - **ログの取得**

U３1/3
認証

- U3. Authentication
  - ウィークパスワードまたは、ブランクパスワード
  - よく知られているパスワード（共有パスワード、ポストイットなど）
  - システムまたはソフトウェアで作成されるアカウント(Default Password等)
  - 強度の弱いパスワードの暗号化方式（キャッシュ）

- 対策-1
  - 共有アカウントの禁止、クリーンデスク
  - 初期パスワード
    - 同じ初期パスワード、推測しやすい初期パスワード
  - Shadow Passwordの利用
    - /etc/shadowファイルは、rootの読み取りだけを可能とする
  - NISは使わない（できれば）
  - 退社時、組織変更時などのアカウント無効化
  - デフォルトパスワードの無効化
  - クリアテキストでパスワードが送付されるプロトコルを使わない（telnet, FTP…）

U 3 2/3
認証

- **対策**-2
  - **パスワードを確実に強力にする**
    - **辞書攻撃対策：辞書に載っているような単語は使わない**
    - Npasswd、PAM**対応ライブラリの利用**
    - **パスワードクラッカによる強度検査**
      - Crack, John the Ripper**等（でも、明確な許可を得ないと、危険）**
  - **パスワードを保護する**
    - **ファイルとして保護する（ハッシュの利用）**
    - **ユーザ教育（アカウントやパスワードを漏らさないなど）**
    - **パスワードの有効期間の設定、パスワードの再利用禁止**
  - **アカウントの厳密な管理**
    - **使用されていないアカウントの無効化**
      - **サービスアカウント、管理アカウント、アプリケーションアカウント**
    - **デフォルトアカウントのパスワード変更**
    - **初期パスワードの強化（乱数など）**
    - **定期的な監査の実施**
    - **厳密な手順（プロシジャー）**

U 3 3/3
認証

- **対策**-3
  - **ログインの暗号化**
    - **クリアテキストでパスワードを送るプロトコルは使わない**
      - telnet, FTP, HTTP, Berkeley r**サービス**
    - **望ましいなプロトコル**
      - ssh(ssh, sftp, scp), SSL
  - **スーパーユーザアカウント**
    - **リモートからの**root**ログインを許可しない**
    - **ローカルにおいても、**root**ログインを限定する**
    - root**アカウントを利用できるユーザを限定する**(Wheel)
    - sudo**利用**
  - **ジェネリックアカウント**
    - **まず、最初は無効にして、どうしても必要な際だけに利用する**
    - **アプリケーションアカウント、ベンダアクセス**
  - **監査証跡**
    - **認証要求は全てログに記録**
    - su/sudo**の実行について適切にログに記録**

      http://www.loganalysis.org/ **を参照**

U４
バージョン管理システム

- U4 Version Control Systems(CVS, Subversion)
  - CAN-2004-0396, CAN-2004-0414, CAN-2004-0416, CAN-2004-0417, CAN-2004-0418, CAN-2004-0397, CAN-2004-0413
- **脆弱性チェックの方法**
  - pserver(2041/tcp), svn(3690/tcp)**が許可されており、以下のバージョンが稼動しているもの**(cvs ver**で確認**)
    - CVS stable release version　1.11.16**以前**
    - CVS feature release version1.12.8**以前**
    - Subversion 1.05
- **対処方法**
  - **最新のバージョンに更新する**
  - 2041/tcp, 3690/tcp**へのアクセスをブロックする**
  - **スタンドアローンシステムでは、匿名アクセスは読み込みのみを許可する**

U５
メール転送サービス

- U5 Mail Transport Agent
  - あまりに、多いので、主要項目だけ紹介します
    - バッファオーバラン、ヒープオーバラン
    - オープンリレーの悪用（第三者中継）
    - スパム、ソーシャルエンジニアリングツール
- 脆弱性チェックの方法
  - バージョン
    - 各MTAの情報を参照してください
  - オープンリレー（第三者中継）
    - http://www.abuse.net/relay.html
    - http://www.cymru.com/Documents/auditing-with-expect.html
  - メールサーバの監査
    - Internet Scanner
    - Nesuss, SARA
- 対処方法
  - 実行、公開する必要のないMTAの対処（停止、隠蔽）
  - 内部トラフィックを処理するMTAを分離する
  - 権限レベルを制限、chroot環境の利用
  - オープンリレー（第三者中継）の禁止

U 6
SNMP

- U6 Simple Network Management Protocol (SNMP)
  - CVE-1999-0294, CVE-1999-0472, CVE-1999-0815, CVE-1999-1335, CVE-2000-0221, CVE-2000-0379, CVE-2000-0515, CVE-2000-1058, CVE-2001-0236, CVE-2001-0487, CVE-2001-0514, CVE-2001-0564, CVE-2001-0888, CVE-2002-0017, CVE-2002-0069, CVE-2002-0302, CAN-1999-0186, CAN-1999-0254, CAN-1999-0499, CAN-1999-0516, CAN-1999-0517, CAN-1999-0615, CAN-1999-0792, CAN-1999-1042, CAN-1999-1126, CAN-1999-1245, CAN-1999-1460, CAN-1999-1513, CAN-2000-0147, CAN-2000-0885, CAN-2000-0955, CAN-2000-1157, CAN-2000-1192, CAN-2001-0046, CAN-2001-0352, CAN-2001-0380, CAN-2001-0470, CAN-2001-0552, CAN-2001-0566, CAN-2001-0711, CAN-2001-0840, CAN-2001-1210, CAN-2001-1220, CAN-2001-1221, CAN-2001-1262, CAN-2002-0012, CAN-2002-0013, CAN-2002-0053, CAN-2002-0109, CAN-2002-0305, CAN-2002-0478, CAN-2002-0540, CAN-2002-0812, CAN-2002-1048, CAN-2002-1170, CAN-2002-1408, CAN-2002-1426, CAN-2002-1448, CAN-2002-1555, CAN-2003-0137, CAN-2003-0935, CAN-2003-1002, CAN-2004-0311, CAN-2004-0312, CAN-2004-0576, CAN-2004-0616, CAN-2004-0635, CAN-2004-0714
  - Using SNMP for Reconnaissance
    - http://www.sans.org/resources/idfaq/snmp.php
- **脆弱性チェックの方法**
  - **ツールによる監査**
    - Internet Scanner, SNMPing, SNScan, Nesus
- **対処方法**
  - SNMP**が必須でなければ無効化する**
  - **可能な限り**SNMPv3**ユーザベースセキュリティモデルを採用する**
  - **最新のパッチを適用する**
  - **ネットワークの入り口で**SNMP**をブロックする**
  - TCP-Wrapper/Xinetd**を使って、接続先を制御する**
  - **デフォルト・推測可能なコミュニティ名を避ける**
  - **できるだけ、**MIB**は読み取り専用にする**

U 7
Open Secure Sockets Layer(SSL)

- U7 Simple Network Management Protocol (SNMP)
  - CVE-1999-0428, CVE-2001-1141, CAN-2000-0535, CAN-2002-0655, CAN-2002-0656, CAN-2002-0557, CAN-2002-0659, CAN-2003-0078, CAN-2003-0131, CAN-2003-0147, CAN-2003-0543, CAN-2003-0544, CAN-2003-0545, CAN-2003-0851, CAN-2004-0079, CAN-2004-0081, CAN-2004-0112,
  - **特に**…
    - CAN-2002-0655, CAN-2002-0656, CAN-2002-0557, CAN-2002-0659 and CAN-2003-0545.
- **脆弱性チェックの方法**
  - **バージョン**
    - 0.9.7C**以前**、0.9.6l**以前**
  - **その他**
    - Apache，CUPS，Curl, OpenLDAP, Stunnel, Sendmail**等の影響もある**
- **対処方法**
  - **最新バージョンへの更新**
  - **可能であれば，**OpenSSL**を有効にしたサーバに接続するシステムを制限する**
    - ipfillter/netfilter, **その他の**FW

U 8 (NIS/NFS) 1/3 Misconfiguration Enterprise Services

- U8 Misconfiguration of Enterprise Services NIS/NFS
  - NFS
    - CVE-1999-0002, CVE-1999-0166, CVE-1999-0167, CVE-1999-0170, CVE-1999-0211, CVE-1999-0832, CVE-1999-1021, CVE-2000-0344, CVE-2002-0830
    - CAN-1999-0165, CAN-1999-0169, CAN-2000-0800, CAN-2002-0830, CAN-2002-1228, CAN-2003-0252, CAN-2003-0379, CAN-2003-0576, CAN-2003-0680, CAN-2003-0683, CAN-2003-0976, CAN-2004-0154
  - NIS
    - CVE-1999-0008, CVE-1999-0208, CVE-1999-0245, CVE-2000-1040
    - CAN-1999-0795, CAN-2002-1232, CAN-2003-0176, CAN-2003-0251
- **脆弱性チェックの方法**
  - **共通：バージョン**
    - **ベンダが提供する最新版であることを確認**
      - rpc.mountd –version, ypserv --version
  - NIS
    - NIS map**内に**root**が保持されていないことを確認**
    - **ユーザパスワードが安全であることの確認**
    - **パスワードのハッシュに**DES**では無く，**Blowfish/MD5**を使用する**
  - NFS
    - /etc/exports**内の**host, netgroup, permission**を確認する**
    - showmount –e SERVER_IP**を実行して、何が**export**されたかを確認する**

U 8 (NIS/NFS) 2/3 Misconfiguration Enterprise Services

- **対処方法**
  - NIS
    - **各クライアントで、**BIND**先の**NIS**サーバを設定**
    - DBM**ファイルの作成中に** YP_SECURE**をアクティブにする。**
      - makedbm **に** s **スイッチを使用**
    - **信頼できるホストとネットワークを、**/var/yp/securenets**に含める**
      - ypserv, ypxfrd
    - NIS**クライアントのパスワードファイルに**+:*:0:0:::**を含める**
    - SSH**のような安全なプロトコルを利用した**NIS**を検討する**
      - Secure NFS and NIS via SSH Tunnel
      - http://www.math.ualberta.ca/imaging/snfs/
  - NFS
    - /etc/exports**では、**Alias**を利用しない**（IP**アドレスか**FQDN**を利用する**）
    - /etc/exports**で、**NFS**ファイルシステムへのアクセスを制限する**
      - NFS**クライアントの亜どれエスの後に**secure**パラメータを追加**
      - NFS**ファイルシステムに適切なパーミッションをつけて**Export
        - » /home 10.20.1.25(ro)
      - **可能であれば、**NFS**クライアントのアドレスの後に** root_squash**を利用する**

U 8 (NIS/NFS) 3/3 Misconfiguration Enterprise Services

- NFS続き
  - 匿名的なパーミッションを利用してExportする場合、“all_squash”パラメータを使用する
  - NFSBugを使って設定をテストする
    - .http://ftp.cerias.purdue.edu/pub/tools/unix/sysutils/nfsbug/
  - Port Monitoring機能をアクティブにする（Solaris)
    - /etc/systems に set nfssrv:nfs_portmon = 1を追加する
- NIS/NFSに関する一般的な事項
  - アクセスコントロールにより、利用可能なクライアント・ネットワークを限定する
  - ベンダ提供のパッチを適用する。
  - CERTのUNIX Security Checklistを参照
    - http://www.cert.org/tech_tips/usc20_full.html
  - NFS/NISサーバではないシステムで、NFS/NISを無効化する

U９　1/2
データベース

- U9 Mail Transport Agent
  - Oracle:
    - CVE-2002-0567, CVE-2002-0571
    - CAN-1999-0652, CAN-1999-1256, CAN-2002-0858, CAN-2002-1264, CAN-2003-0095, CAN-2003-0096, CAN-2003-0222, CAN-2003-0634, CAN-2003-0727, CAN-2003-0894
  - MySQL:
    - CVE-1999-1188, CVE-2000-0045, CVE-2000-0148, CVE-2000-0981, CVE-2001-0407
    - CAN-1999-0652, CAN-2001-1274, CAN-2001-1275, CAN-2002-0229, CAN-2002-0969, CAN-2002-1373, CAN-2002-1374, CAN-2002-1375, CAN-2002-1376, CAN-2003-0073, CAN-2003-0150, CAN-2003-0515, CAN-2003-0780, CAN-2004-0381, CAN-2004-0388, CAN-2004-0627, CAN-2004-0628
  - PostgreSQL:
    - CVE-2002-0802
    - CAN-1999-0862, CAN-2000-1199, CAN-2001-1379, CAN-2002-0972, CAN-2002-1397, CAN-2002-1398, CAN-2002-1399, CAN-2002-1400, CAN-2002-1401, CAN-2002-1402, CAN-2003-0040, CAN-2003-0500, CAN-2003-0515, CAN-2003-0901, CAN-2004-0366, CAN-2004-0547
- **脆弱性チェックの方法**
  - **ツールによるスキャンの実施**
    - MySQL Network Scanner
    - Nessus
    - Foundstone, Quelys, eEyE Retina
    - APPSecInc, ISS Database Scanner

U92/2
データベース

- **対処方法**
  - **ベンダ情報**
    - Oracle http://otn.oracle.com/software/index.html
    - MySQL http://www.mysql.com/products/mysql/
    - PostgreSQL ftp://ftp.postgresql.org/pub
  - **一般的な対処**
    - **最新のパッチを適用する**
    - **最低限の権限を使用する**
    - **データベース権限を持つシステムアカウントデフォルトパスワードの削除･変更**
    - **不要なストアードプロシジャを削除･無効**
    - **全てのフィールド長に制限をつける**
    - **サーバサイドのデータ全てを確認する（長さ、フォーマット、タイプ）**
    - **有効なリソース**
      - Oracle (http://otn.oracle.com/deploy/security/index.html)
      - MySQL (http://dev.mysql.com/doc/mysql/en/Security.html)
      - PostgreSQL (http://www.postgresql.org/docs/7/interactive/security.htm)
      - Oracle**向けチェックリストなど**
        - » http://www.sans.org/score/oraclechecklist.php
        - » http://www.cisecurity.org/bench_oracle.html
        - » http://www.petefinnigan.com/orasec.htm

U 10　1/3
カーネル

- U10 kernel
  - CVE-1999-0295, CVE-1999-0367, CVE-1999-0482, CVE-1999-0727, CVE-1999-0804, CVE-1999-1214, CVE-1999-1339, CVE-1999-1341, CVE-2000-0274, CVE-2000-0375, CVE-2000-0456, CVE-2000-0506, CVE-2000-0867, CVE-2001-0062, CVE-2001-0268, CVE-2001-0316, CVE-2001-0317, CVE-2001-0859, CVE-2001-0993, CVE-2001-1166, CVE-2002-0046, CVE-2002-0766, CVE-2002-0831
  - CAN-1999-1166, CAN-2000-0227, CAN-2001-0907, CAN-2001-0914, CAN-2001-1133, CAN-2001-1181, CAN-2002-0279, CAN-2002-0973, CAN-2003-0127, CAN-2003-0247, CAN-2003-0248, CAN-2003-0418, CAN-2003-0465, CAN-2003-0955, CAN-2003-0984, CAN-2004-0003, CAN-2004-0010, CAN-2004-0177, CAN-2004-0482, CAN-2004-0495, CAN-2004-0496, CAN-2004-0497, CAN-2004-0554, CAN-2004-0602
- **脆弱性チェックの方法**
  - **ベンダ情報の取得（登録時にセキュリティ更新メールを受け取るようにする）**
  - **セキュリティに関するメーリングリストを購読する**
  - **稼動しているカーネルのバージョンを追跡する**
  - **セキュリティ評価ソフトを使用する**
    - Internet Scanner, Nesuss
      - **ただし、サービスに影響がでる可能性があることを考慮する**

U 10 2/3
カーネル

- **対処方法**
  - **システムカーネルを適切に調整する**
    - Solaris Tunable Parameters Reference Manual (Solaris 8)
      - http://docs.sun.com/app/docs/doc/816-0607
    - Solaris Tunable Parameters Reference Manual (Solaris 9)
      - http://docs.sun.com/app/docs/doc/806-7009
    - Solaris Operating Environment Network Settings for Security
      - http://www.sun.com/blueprints/1299/network.pdf
    - Solaris Kernel Tuning for Security or http://www.securityfocus.com/infocus/1385
      - http://www.leonine.com/sunlib/kernel/solaris_kernel_tuning.html
    - Linux Kernel Hardening
      - http://www.securityfocus.com/infocus/1539
    - The Linux Kernel Archives
      - http://www.kernel.org/
    - Linux Kernel Hardening
      - http://www.sans.org/rr/papers/index.php?id=1294
    - AIX Kernel Tuning
      - http://publib16.boulder.ibm.com/pseries/en_US/aixbman/prftools/kerneltun.htm

U 10 3/3
カーネル

- **システムカーネルを適切に調整する (続き)**
  - HP-UX Kernel Tuning and Performance Guide
    - http://docs.hp.com/hpux/onlinedocs/os/11.0/tuningwp.html
    - http://docs.hp.com/hpux/pdf/5185-6559.pdf
    - http://docs.hp.com/hpux/pdf/TKP-90203.pdf
    - http://docs.hp.com/cgi-bin/otsearch/hpsearch
    - http://docs.hp.com/
  - FreeBSD Handbook (contains information on kernel tuning):
    - http://www.freebsd.org/doc/en_US.ISO8859-1/books/handbook/index.html
  - OpenBSD:
    - http://www.openbsd.org/faq/index.html
    - http://www.openbsd.org/docum.html (for more info)
  - NetBSD Tuning, Kernel Tuning
    - http://www.netbsd.org/Documentation/tune/ktune.html

セキュリティ監査の現場から

基本事項 – 1

- パスワード管理
  - あまりに脆弱なパスワード設定が多い
    - ブランクパスワード (PDCのAdministratorがブランクの例も)
    - デフォルトパスワード (インターネットで簡単に入手できてしまう)
    - 推測可能なパスワード (root = root, Administrator = Administrator)
  - パスワード設定が強制されていない
    - パスワードの複雑さ、有効期限等機械的な強制が行われていない
  - パスワードの問題は、サーバに限らない
    - ルータのデフォルトパスワード (＋TELNET接続の許可…)
    - HSRPのデフォルトパスワード (スイッチ・ルータは大丈夫？)
    - データベースシステムのデフォルトパスワード
    - ミドルウェアのデフォルトパスワード
  - アカウントのロックアウト機構の未使用
    - システム、データベース等

基本事項 – 2

- 管理者権限に対するアクセスコントロール
  - 管理者権限で稼動するサービス
    - ウェブサーバ、メールサーバ、DNSサーバ
    - データベース、ミドルウェア
  - 管理者ページへのアクセスコントロール
    - ウェブを使って管理できるものがあるが..
      - 認証が省略されている場合がある
      - IPレベルのアクセスコントロールが行われていない場合が多い
- データベースのアカウントコントロール
  - このあたりがしっかり作られないと、データ流出や改ざんの可能性大
    - Publicが管理権限をもっている。。。
    - OSアカウントによるデータベースの利用許可
    - Resource　Roleの不適切な付与
      - クラスタ、プロシジャ、トリガの追加が可能

主に構築に関わる問題– 1

- **ネットワーク構成とネットワークデバイス**
  - Default Acceptなファイアウォール
    - 特定のプロトコル（ポート）だけをブロックして、これ以外のプロトコルを許可している
      - SMB(NetBIOS), SQL, TFTP等がファイアウォールを通過している
        - » ワームはこれほど拡散する理由のひとつ
      - なぜ、チャットが必要なのか？？？
  - ファイアウォールを信用しすぎている
    - ファイアウォールの内側は、セキュリティ対策が行われていない
      - MS Blast、Nimdaなどで苦労した方も多いのでは…
      - ファイアウォールを通過するポートに対して防御はできない
  - セグメント間のアクセスコントロールがきっちりと設計されていない
    - DMZからは、全てのセグメントにアクセスができた事例
  - ルータやスイッチの対策が行われていない
    - ルータのインターネット側のインタフェースでTELNETが稼動
    - アカウントやパスワードが工場出荷時のまま。。。

主に構築に関わる問題– 2

- **サーバ等の設定**
  - **不要なサービスが稼動している**
    - **特に、知らぬ間に立てられたサーバが危ない**
  - **ログが記録されていない**
    - **特に**Windows**サーバ**
  - **利用しない機能が有効になっている**
    - **ウェブサーバの動的オブジェクトや各種モジュール**
  - Rlogin**の稼動と** .rhost**の設定**
    - /etc/rhosts**に** ‘+’ **がかかれている**
    - ~/.rhost **の存在**
  - **パーミッション**
    - **ログや**/etc**が、ワールドで書き込み可能になっている**
  - **セキュリティパッチの未適用**
    - **後述**

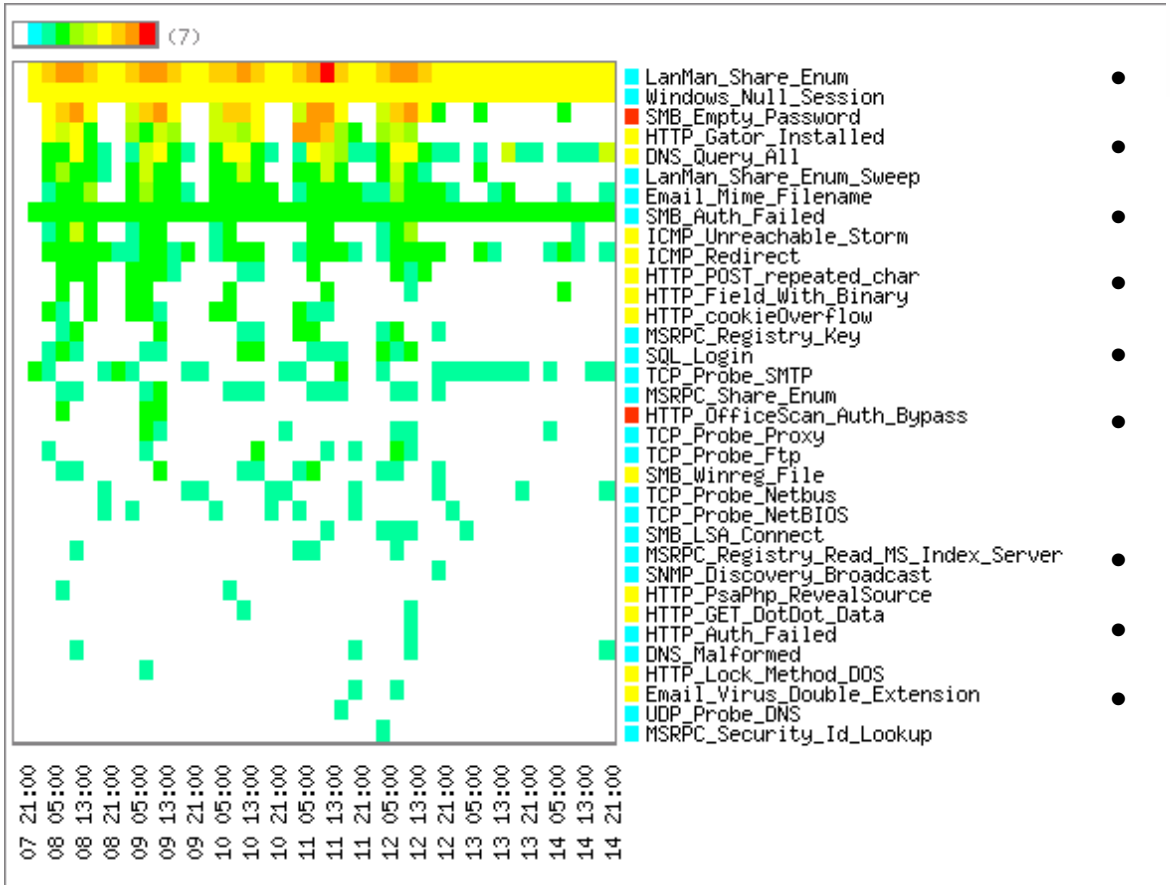
主に運用に関わる問題

- セキュリティパッチ
  - 運用計画にセキュリティパッチの適用が盛り込まれていない
    - 運用計画にないから、パッチが適用できない
    - 無条件にWindows Updateを行うサーバ …
- バックアップ
  - バックアップは適切か？
    - バックアップは適切にとられていない
    - リストアの実績がない
      - バックアップが適切かどうかがわからない
      - バックアップから復旧できるかわからない
    - どのメディア（テープなど）を使えばよいか明確だろうか？
  - リカバリープランは明確だろうか？
    - どの位の時間で復旧することを想定しているのだろうか？（1時間？1週間？）
- 緊急時対応の体制が構築されていない
  - どうやって問題をみつけて、誰が、どのように対処するのか？
  - 連絡体制は運用担当者だけで閉じていないだろうか？

主に現状把握に関わる問題

- 見知らぬIPと見知らぬポート
  - セキュリティ監査の結果…
    - あるはずのないIPを発見
    - あるはずのないサービス（ポート）を発見
- 構成管理
  - 現在の実環境における以下の情報を、何分で出せるか？
    - ネットワーク構成図とファイウォールのルール
    - あるOSを利用しているサーバの一覧
- 通常状態の把握ができていない
  - DDoSを受けました、助けて！！
    - 実は。。。。
  - なんだか、ネットワークが遅いんだが、なにが悪いんだろう。。。
  - ネットワーク管理者の方は、チャット、P2P、ウェブメールが、これだけ社内で利用されていることを知っているのだろうか？

トラフィック監査の例



- **ブランクパスワード**
- **安易なパスワード**
- **認証の失敗**
- SQL**接続**
- **ウィルスの存在**
- SQL Injection**の可能性**

- Chat(IRC)
- Web Mail
- IPv6

その他の問題 -1

- アンチウィルスソフトのアップデート
  - 導入自体は行われているが、自動アップデートに設定されていない

- 人的？教育的？管理的？問題
  - マシンルームで監査をしていると、年配の方が突然入室
  - サーバをシャットダウンしてしまった
  - 操作を見ていると、パスワードがブランク。。。
  - 「パスワードをつけたほうが良いですよ」といったところ

    - だって、おぼえられないじゃない ☺

その他の問題 -2

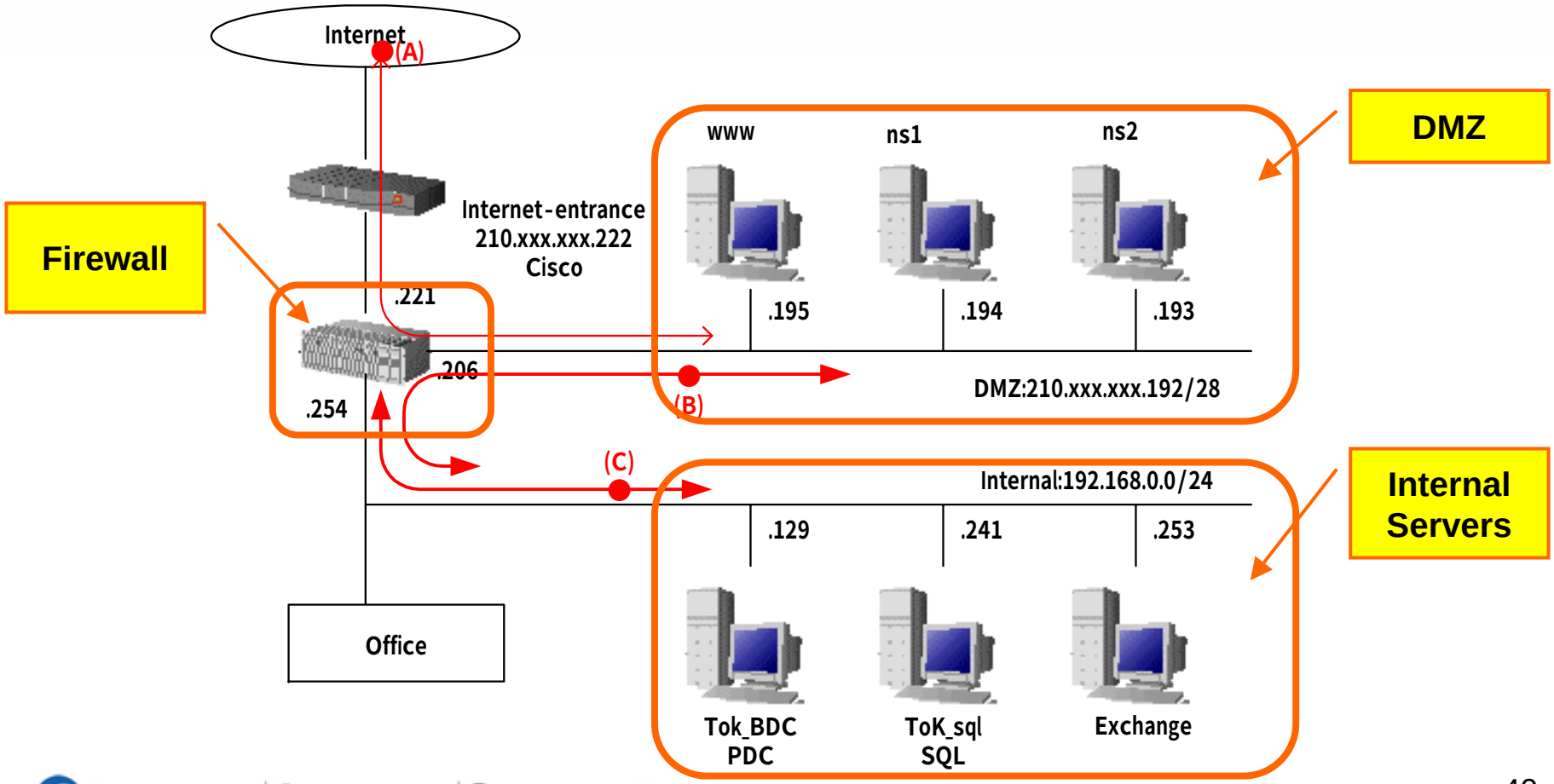
- データベース設計の問題
  - アプリケーションで利用するアカウントパスワードが、プログラムに組み込まれていて変更不可能
    - ちなみに、xxx = xxx だった。
  - 権限が分離されていない
    - SQLサーバへつながれば、だれもが書き込めるデータベース
  - SQLでデータを引けるIPが限定されていなかった
    - 某所で問題になっているパターンと思われる
- メールサーバの設定
  - 第三者中継も重要ですが、送受信の容量制限が重要
    - 非常に大きい添付ファイルをつけてメールを送信した者がいた。
    - 送信はされたのだが、しばらく経ってからエラーで戻された。
      - 相手のサーバがダウンした可能性もある
    - これが、また中途半端にエラーになるので、繰り返し送りつけられて、メールサーバが停止した。。。

被害を広げないためには

- **サーバ等の要塞化は必要だが…**
  - **全てのノードの要塞化は、かなり難しい**
  - **セグメント化と、セグメントによる防御が重要**
- **例えば…**
  - **監査事例の紹介**

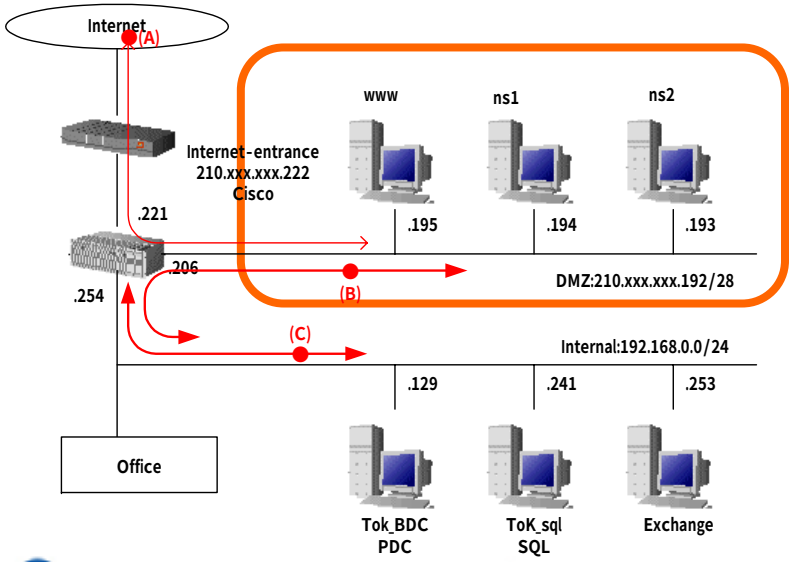
監査結果サンプル

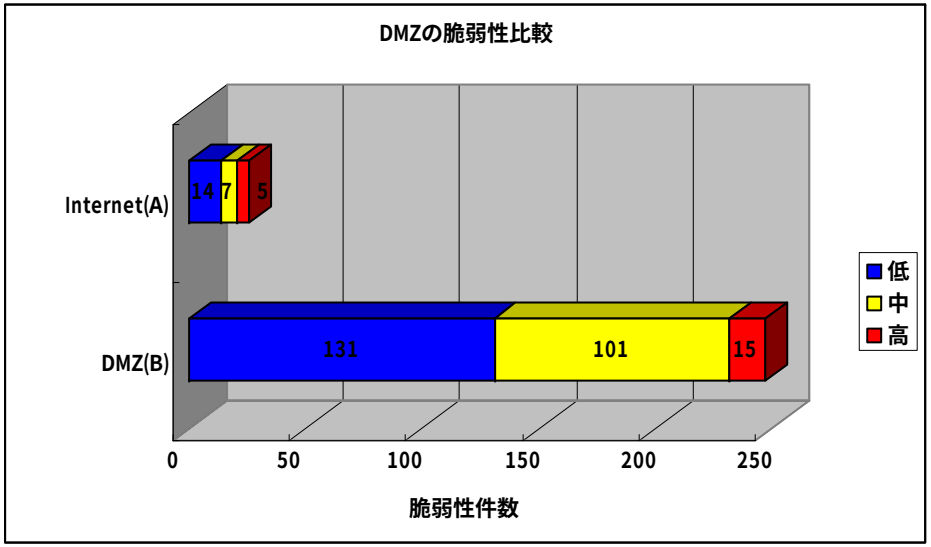
- 以下のネットワークに対するセキュリティ監査を実施



DMZの脆弱性診断結果

- FWによるコントロールは、ある程度対策されている
- FWでフィルタされているサービスなどに、多数の脆弱点がある。

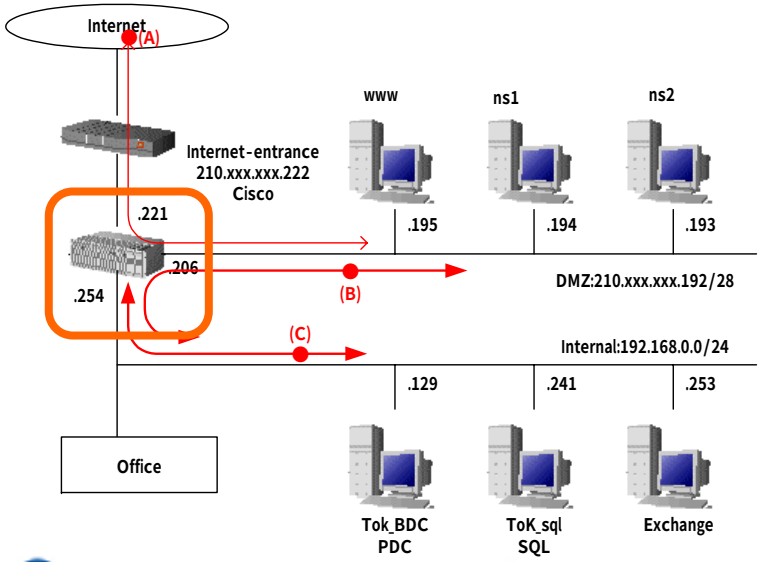


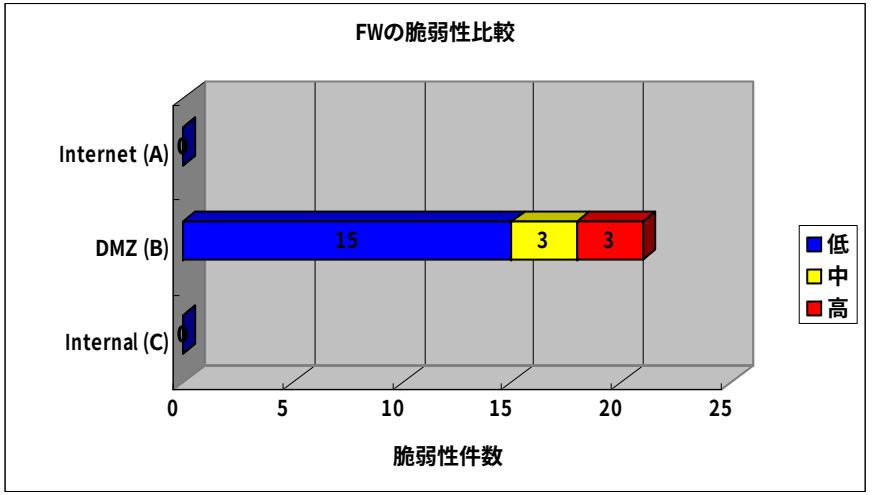


INTERNET | SECURITY | SYSTEMS®

FWの脆弱性診断の結果

- 外部および、内部ネットワークから、FW自身の脆弱点は見つからなかったが、DMZからの診断では、21件の脆弱点が見つかった
- また、DMZに対してコントロールがまったく行われておらず、DMZへの侵入へ成功すると、FWのアクセス件を取得できる可能性が高い

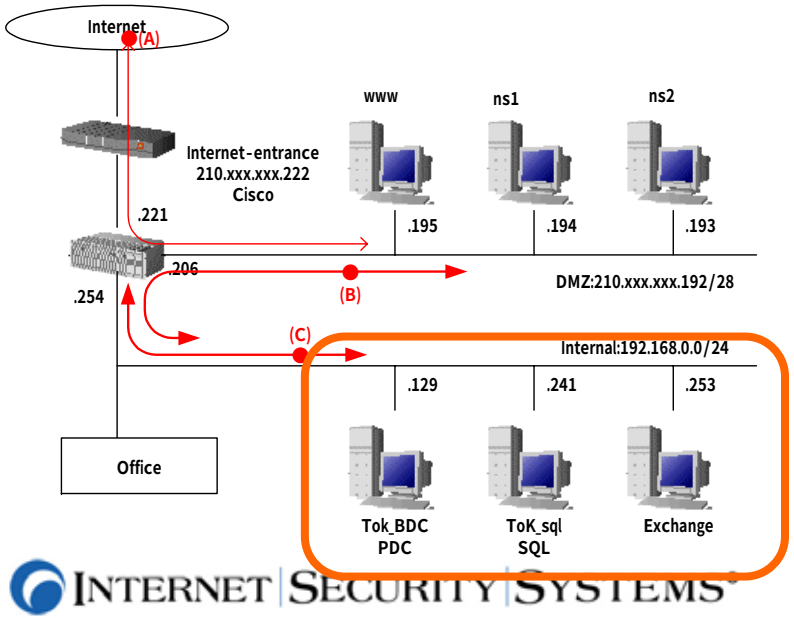


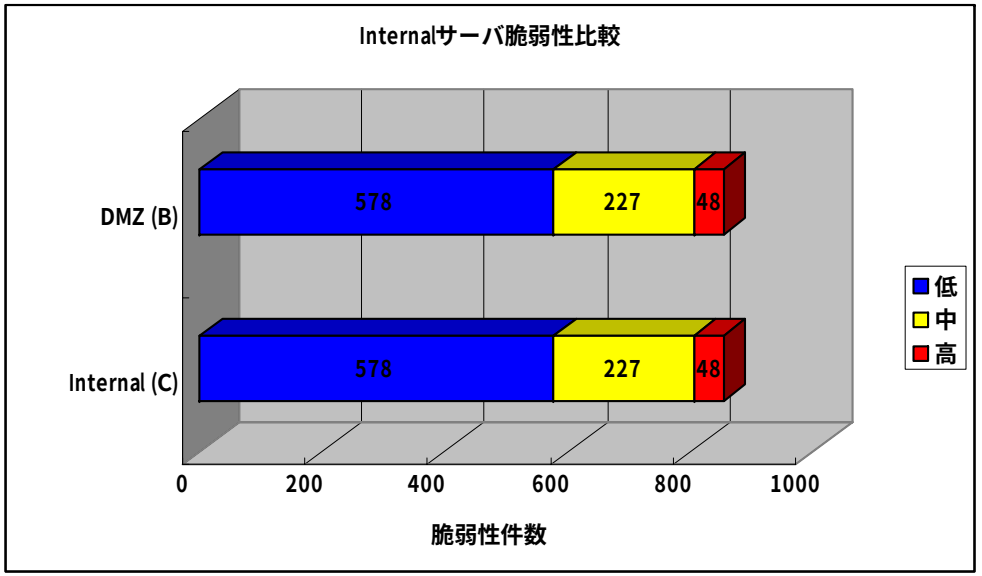


INTERNET | SECURITY | SYSTEMS®

インターナルネットワーク

- DMZからの診断結果と、インターナルネットワーク内部からの診断結果がまったく同じ
- つまり、DMZからのパケットは、まったくフィルタリングされていない。

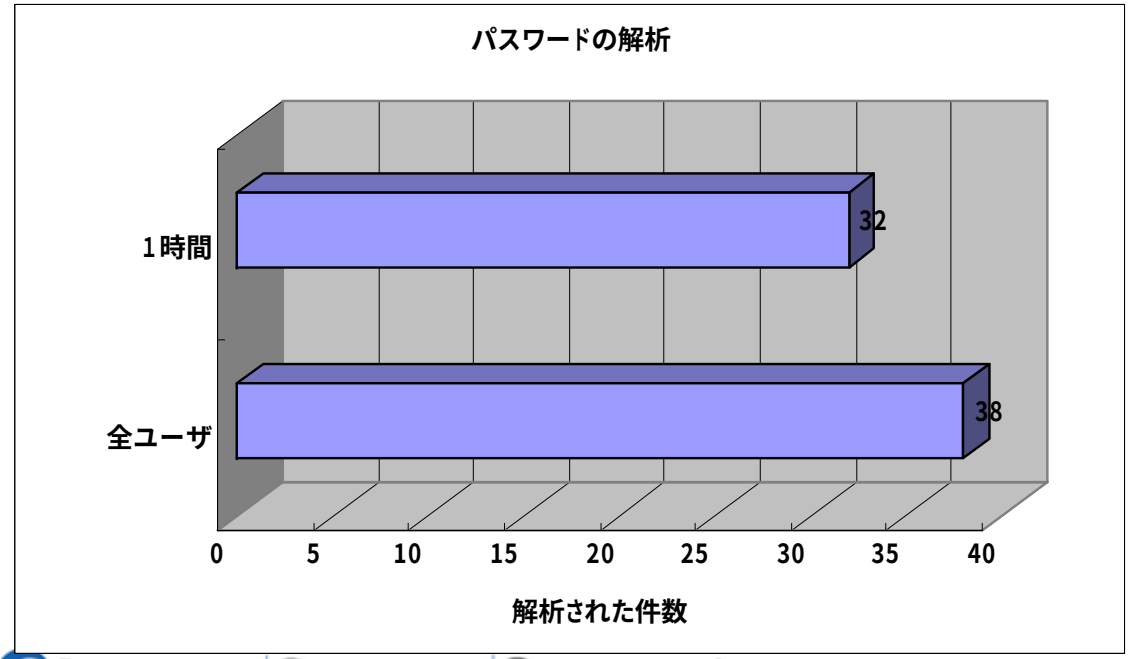




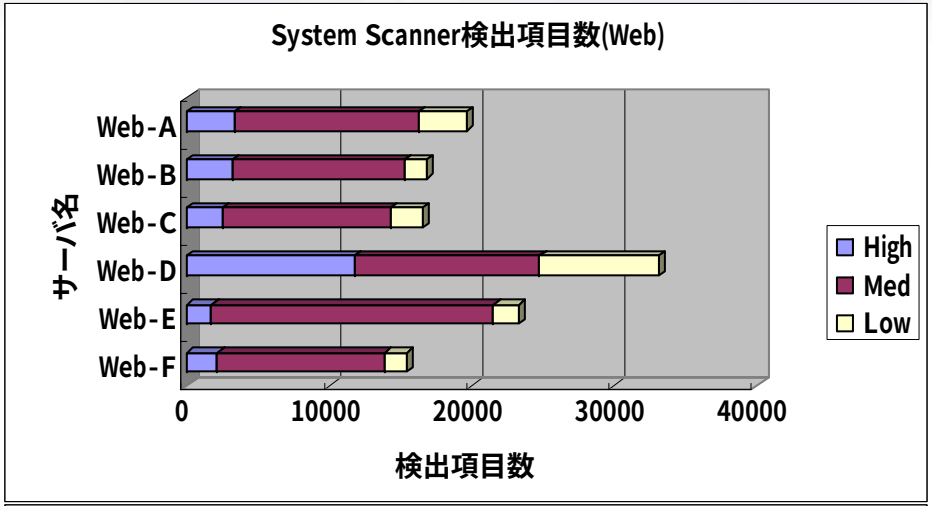
INTERNET SECURITY SYSTEMS

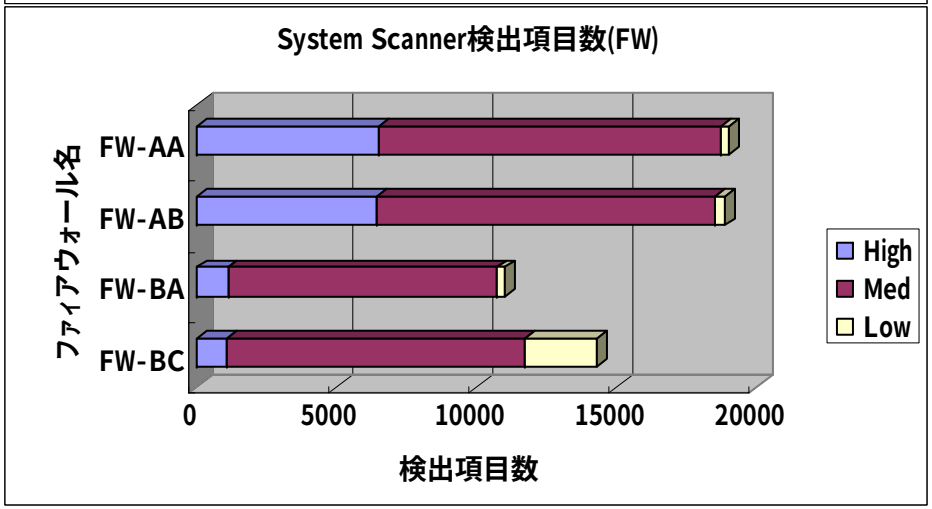
パスワードの強度検査結果

- **一時間で、88%のアカウントのパスワードを破ることが出来た。**



要塞化失敗の例





ロードバランスされたシステムは、本来同じ設定になっているはず

しかし、この事例では、左グラフのように、極端に設定が違うものがあった。

まとめ

まとめ

- UNIXが安全とは限らない
  - Zone-Hでは、99%がUNIX系のシステム
  - そもそも、リモートから使いやすいシステムですし。。。
- アプリケーションへの移行
  - システムへの攻撃から、アプリケーションへの攻撃に移行している
    - Database, Web Application
- クライント(PC)を忘れずに
  - サーバは、UNIXでもPCの大半はWindows
  - クライアントからコネクションを張られてしまうと、FWでは防げない
  - また、クライアントをターゲットしたインシデントも増えているので、これも忘れずに。。。

まとめ

- インシデント被害の確率を減らすための手順は充実している
  - ツール関係は、かなり充実しており、十分に優位な対策を実施することが可能となってきている
- しかし、解決していない問題がある
  - 時間の問題
    - 脆弱性が公表されてから、ツールやワームが出るまでの時間が、短くなっている．（0～数週間）
  - 人間の問題
    - マス・ソーシャルエンジニアリング（ワーム、フィッシング）が一般化したことにより、改めて人間の問題が問われている．
    - 正しく動いていることを確認する作業が、あまり行われていない．