サーバーに対する不正アクセスの検証

○早川正人、鬼頭良彦、岡田佳浩、千代谷一幸、青木延幸

名古屋大学全学技術センター　部局系技術支援室工学技術系

はじめに

サーバー管理者の業務においてセキュリティ対策は重要な課題であり、日常対策としてアクセス制限、暗号化通信などはよく利用されているが、それでも防ぎきれない場合も出て来ている。不正アクセスによる各種の障害を未然に防ぐには、日常的な監視でその兆候をいかに見つけるかがポイントとなり、アタック手法の情報収集が必要となる。そこで今回は、Linux にセキュリティツールをインストールしメジャーなアタックツールによるハッキング時のログを検証したので、その報告を行う。

1　サーバーの構成

サーバーの OS には、Vine Linux 3.0 を使用、インストールオプションにてフルインストールを選択してインストールを実行した。その後、IP アドレスによるホストのアクセス制御を TCP_Wrappers の使用により設定した。なお、セキュリティ上問題のある telnet、ftp のデーモンは起動せず OpenSSH を利用する事とした。

2　セキュリティ確保のためのツールの導入

アクセス制限や、暗号化通信などのホストのセキュリティを向上する手段を取っても、これだけではホストに対して不正なアクセスが試みられているか関知するには、十分ではない場合も有る。そこで、セキュリティツールとして、Tripwire、Snort、swatch、Logwatch の各ソフトウェアをインストールした。

2.1　Tripwire

Tripwire はシステムの整合性をチェックするソフトウェアである。監査対象となるディレクトリやファイル属性の HASH 値をデータベースに保存しておき、データベースと現在の属性を比較することで整合性のチェックを行う。チェックの結果については、メールによるレポートの送信が可能なので、cron で定期的に整合性のチェックを行い、レポートを管理者にメールで送る事によって、管理運用が楽になる。

- Tripwire のダウンロード URL　http://www.tripwire.org/downloads/index.php

2.2　Snort

Snort はシグネチャマッチング型の NIDS（Network-Based Intrusion Detection System）である。ネットワーク上を流れる通信の内容を監視し、あらかじめ用意された「ルールセット」と呼ばれる攻撃方法のパターン（シグネチャ）とマッチした場合に異常を検出する。Snort はポートスキャンなども検知できるので、不正なアクセスを監視して、被害を未然に防ぐために利用できる。

Snort の記録するログ（alert ファイル）は情報量が多いので、目的に応じた処理を行い、状況を判断できるようにする必要がある。そこで、解析結果を HTML 形式や PDF 形式で保存でき、メールで送る事も出来るプログラム SnortALog（Snort Analyser Logs）を利用する事にした。

また、Snort のルールファイルの更新は、プログラム Oinkmaster を利用し cron で定期的に実行する。

- Snort のダウンロード URL　http://www.snort.org/

- SnortALog のダウンロード URL　http://jeremy.chartier.free.fr/snortalog/
- Oinkmaster のダウンロード URL　http://oinkmaster.sourceforge.net/

2.3　swatch

swatch はリアルタイムで任意のログファイルを監視し、ログファイルに swatch の設定ファイルに記述した文字列が出力された場合に指定したアクション（ビープ音を鳴らす、ターミナルにそれらの文字列を表示、メールを送信）を実行するツールである。なお、監視したいログファイルが複数ある場合は、それぞれ別々に swatch を起動する必要がある。

- swatch のダウンロード URL　http://swatch.sourceforge.net/

2.4　Logwatch

Logwatch はログの監視を毎日定期的に行い、ログから特定のパターンを含む行を比較的見やすい状態に加工して、メールでレポートしてくれるツールである。Logwatch を定期的に実行するには、cron を使えばよいが、RPM 版の Logwatch はインストール時に cron の登録設定も行ってくれる。

- Logwatch のダウンロード URL　http://www2.logwatch.org:81/

3　不正アクセスの工程

サーバーに対しての不正アクセスを調査し対策を講じる上で、攻撃者の手口を知っておくことは重要なことである。攻撃者は不正アクセスを試みる際に、いきなり侵入を試みるためのプログラムを実行するのではなく、まずはそのサーバーの様々な情報や、どのようなセキュリティホールが有るかを調査してから行うと考えられる。次に、図１に沿って不正アクセスの基本的な流れについて説明する。

3.1　ターゲッティング

ターゲッティングとは、ターゲットとなるサーバーを慎重に選び、サーバーが提供している各種サービス内において合法的に様々な情報を調べる。例えば、UNIX の一般的なコマンドを利用する事によって、ホストの有無、ネットワーク経路、DNS 情報などが集められる。

また、Web サイトを運営しているところでは、Web ページの内容からも様々な情報が得られる事もあるので、公開する情報には注意が必要である。

3.2　スキャン

スキャンとは、ターゲットとなるサーバーの仕様などの情報を得るために行う信号の送受信のようなもので、アタック方法としては、Ping スイープ、ポートスキャン、OS 特定、アタックフィンガープリンティング、ネットワークリソースの列挙、ユーザー・グループの列挙、アプリケーションバナーの列挙などがある。例えば、Ping スイープはシステムが稼動しているどうか調べるのに使用し、ポートスキャンはターゲットホスト上でどのサービスが実行中なのか、または待機（listen）状態なのかを特定するために実行される。

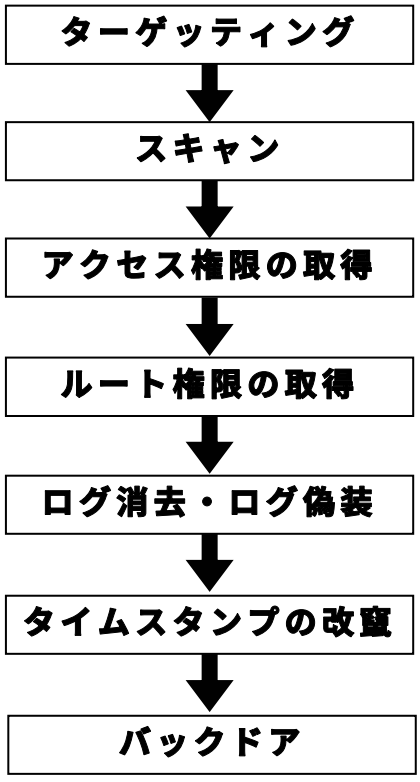

図１．不正アクセスの工程

また、ターゲットのサーバー上で実行中のサービスに対してコマンドを実行して、その応答メッセージ（バナー）からソフトウェアの種類やバージョンを調べる。この情報によりターゲットの弱点となるセキュリティホールを探し出す。

このような攻撃者からサーバーを守るためには、最新のセキュリティパッチの適応は勿論、不要なサービスは無効にしておき、ソフトウェアのバナーはできるだけ変更しておく必要がある。また、Snort などの IDS を導入しネットワークのパケットを監視することも重要である。

3.3　アクセス権限の取得、ルート権限の取得

ターゲットとなるサーバーに不正侵入するためには、ユーザーID やパスワードを解読する必要がる。これはブルートフォースアタックと呼ばれ、辞書などの用語を組み合わせ、総当り方式でパスワードの解読をする。特にログイン ID とパスワードが同一や平易な英単語の使用などは解読されやすので、使用は避け定期的に変更することが望ましい。

一方、スキャニングでターゲットサーバーにセキュリティホールを発見した場合は、その脆弱性をついたプログラムを使って進入を試みる。もしプログラムにバッファオーバーフローのセキュリティホールがある場合、スタックフレームの当たる文字列に悪意のあるコードを埋め込んであれば、それがそのプログラムのユーザー権限で動作する。よって、そのプログラムが root 権限で動作していれば、悪意あるコードが root 権限で実行されることになる。

バッファオーバーフローを回避するには、基本的にはセキュリティホールを作らないことだが、Avaya が提供する Libsafe などのバッファオーバーフロー対策ツールの導入も有効である。

- Libsafe のダウンロード URL　http://www.research.avayalabs.com/project/libsafe/

3.4　ログ消去・ログ偽装、タイムスタンプの改竄

攻撃者が不正侵入したサーバーのログから自分の痕跡を消すためには、ログファイルそのものを削除するか、内容を偽装する必要がある。ログファイルを削除した場合は、管理者に気付かれてしまう可能性が高いので、通常は攻撃者自身に関係する痕跡だけを改竄する。

また、システムクロックやタイムゾーン、ファイルのタイムスタンプを変更することによって、管理者を混乱させ追跡を困難にさせる。

ログの改竄を防止するには、ログを別のサーバー（ログ収集専用サーバー）に転送して保存することが考えられる。また、システムログなどを入力情報とする不正検出ツールの swatch を導入して、リアルタイムにログファイルを監視することも有効である。

3.5　バックドア

バックドアとは、一度侵入に成功した攻撃者が、次回から侵入するために仕掛けておく秘密の入り口である。攻撃者はバックドアを仕掛けておくことで、次回から簡単に侵入ができ、たとえ後日、侵入時に利用した脆弱性が利用できなくなったとしても、自らが設置したバックドアによって容易にサーバーに侵入することができる。

バックドアを実現するためには、大きく分けて２つの方法があり、１つは不正侵入したサーバーのコマンドやツールを駆使してバックドアを作る方法と、もう１つはバックドア専用のプログラムを使用する方法である。これらの専用ツールには、バックドアを設置するだけでなく、特定のファイルやプロセスを隠蔽したり、コマンドを改竄する機能なども含まれており、より強力である。

バックドアの対策としては、ファイルが改竄されていないかどうかを確認するためのソフトウェア Tripwire の導入や、バックドア検出ツール chkrootkit の使用がある。

- chkrootkit のダウンロード URL　http://www.chkrootkit.org/

4　まとめ

日々増加しているサーバーへの不正アクセスに対する対応は、日常のログの監視により、いかにして初期の段階（ターゲッテング、スキャン実行時）に兆候を見つけて対策を取れるかが重要である。

サーバーのセキュリティ維持管理におけるログの重要性に付いては、以前から考えてはいたが、実際に管理者が全てのログを常時監視し、解析することは不可能である。今回、各種セキュリティ対策ツールを検証した結果（図２）、それぞれの状況に応じてセキュリティ対策ツールを選び利用する事によって、実際のサーバー管理業務における管理者の負担を軽減できると確信した。

また、不正アクセスを行ってくる攻撃側の手口を調査した後、ローカルなネットワーク上で実際に検証用サーバーに対して一部のスキャン行為を実行した。この結果、サーバーが不正アクセスを受けた時のログが収集でき、状況などが検証できた。今後は、これらを基にログの監視状況を検討し、より効果的なサーバーのセキュリティ維持管理に役立てていきたい。

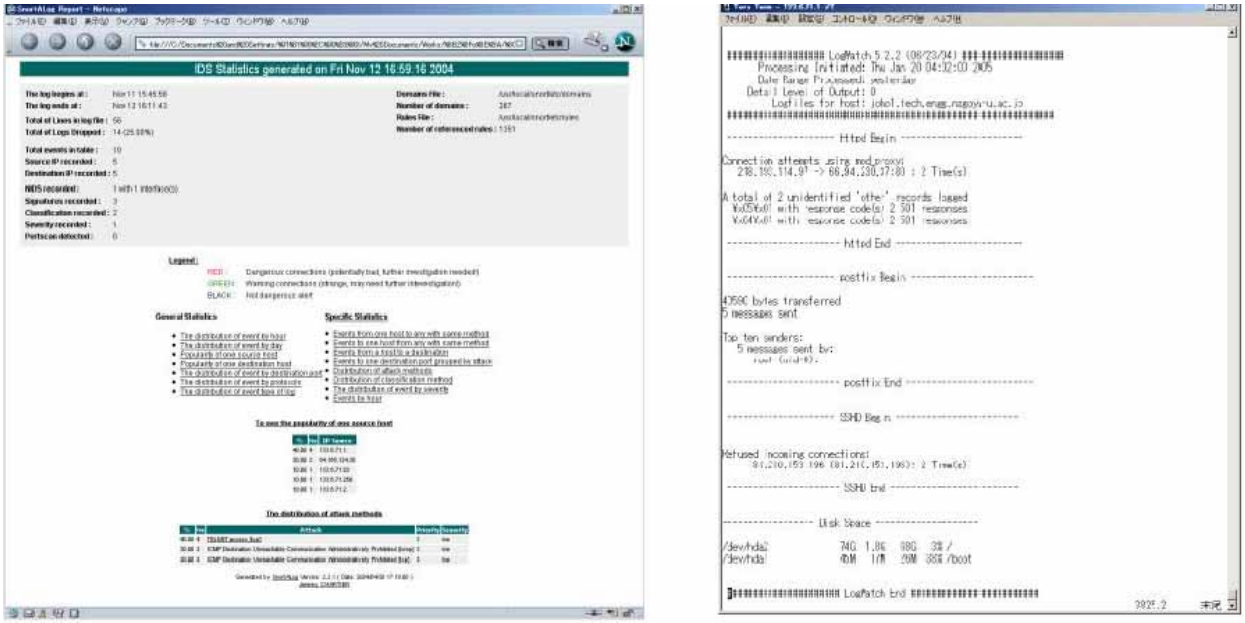

図２．SnortALog による HTML 形式の解析結果＆Logwatch のメールによるレポート

参考文献

[1]　高町健一郎,“ UNIX ネットワークセキュリティ導入・運用ガイド ”，秀和システム

[2]　一条博,“ ネットワーク監視システム ”，工学社

[3]　IPUSIRON,“ ハッカーの教科書 ”，DATA HOUSE

[4]　http://akademeia.info/main/security_lecture.htm

[5]　http://www.atmarkit.co.jp/fsecurity/rensai/iprotect01/iprotect01.html