4F-2 ハッキング手順模擬機能を有するセキュリティホール診断ツールの試作[1]

河内 清人, 藤井 誠司, 勝山 光太郎[2]

三菱電機(株) 情報技術総合研究所[3]

1. はじめに

不正アクセスに対するサイトの脆弱性を定期的に監査する必要がある、という認識が年々高まりつつある。それを受け、近年、ISS[1],Nessus[2]等、様々なセキュリティホール診断ツールが開発・販売されている。

しかし、これらは、指定されたホスト上の、セキュリティホールを検出し、それぞれの危険度を評価、報告するのみである。そのため、

- 複数のセキュリティホールが組み合わされた場合のリスクを評価できない。
- 致命的なセキュリティホールを持つホストを踏み台として、更に内部のホストへの攻撃が行われた場合のリスクを評価できない。

という問題点が挙げられる。

そこで、今回筆者は、

- クラッカーの一般的な攻撃手順をスクリプトに記述し、複数のセキュリティホールを組み合わせた攻撃を自動的に試みる機能
- 侵入に成功したホストを踏み台として、さらに内部のホストへの侵入を試みる機能

を持ったセキュリティホール診断ツールが必要であると考え、Linux 上で設計・試作を行った。

以下、2 節では、本ツールの全体構成、及び各構成要素の動作について概説する。次に 3 節及び 4 節で上記機能の実現方式についてより詳しく説明し、5 節でまとめと今後の課題について述べる。

2. 全体構成

本節では、まず本ツールの全体構成について述べる。図 1 は、本ツールの全体構成を表している。以下、各構成要素の動作について、概説する。

はじめに、ユーザは GUI を通じて検査の開始を**スクリプト実行部**に対して指示する。スクリプト実行部は、**スクリプトストレージ**よりスクリプトをロードし、実行する。

スクリプトには、ハッキングの手順が記述されており、スクリプト実行部はスクリプトの内容に従って、**検査プラグイン**と呼ばれる共有ライブラリをロード、実行する。

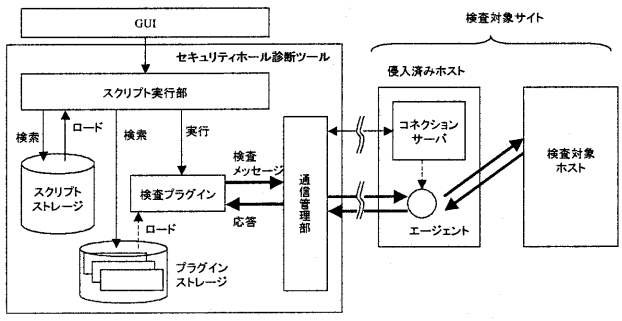

図 1 診断ツール全体構成

通信管理部は、**コネクションサーバ**と連携し、検査プラグインのために、検査対象ホストへの通信路を確立する。**コネクションサーバ**とは、侵入に成功したホスト上で、本ツールが起動するサーバプログラムである。

実際に検査プラグインからの攻撃メッセージを中継するのは**エージェント**と呼ばれる構成要素である。**エージェント**は通信管理部の要請に応じて、コネクションサーバによって生成される。

3. スクリプトによる侵入検査の自動化

本節では、スクリプトによる侵入検査の自動化について、より詳しく説明する。

3.1. スクリプトの検索

スクリプトは、コード部の他にラベル領域を持っている。ラベル領域には、そのスクリプトの機能と

---

[1] Prototyping of a security assessment tool which can simulate hacking steps

[2] Kiyoto KAWAUCHI, Seiji FUJII, Kotaro KATSUYAMA

[3] Mitsubishi Electric Corporation, Information Technology R&D Center 5-1-1, Ofuna, Kamakura, Kanagawa, 247, Japan

実行に必要なパラメータを記述した文字列が格納される。

スクリプトを実行中に、他のスクリプトの機能が必要になった場合には、必要な機能名とパラメータを指定してスクリプト実行部を呼び出すことで、該当するラベルを持ったスクリプトが自動的に検索・実行される(図 2)。

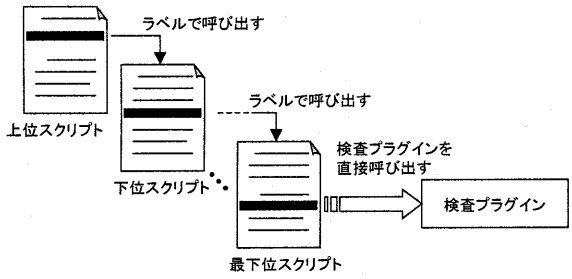

図 2スクリプトの階層化

3.2. スクリプトの実行

スクリプトは、スクリプト実行部によって起動された Perl[3]インタプリタによって実行される。スクリプトはパイプを通じてスクリプト実行部と通信を行う。

4. 踏み台を利用した内部ホストへの侵入検査

本ツールは、直接接続不能な内部ホストに対しても、検査プラグインが使用可能な通信路を実現することができる。以下にその実現方式について述べる。

4.1. コネクションサーバ

コネクションサーバは、通信管理部からの要求に応じてエージェントを生成する。

各コネクションサーバは互いに接続し、全体としてネットワークを形成している。

コネクションサーバはバッファオーバフロウなどを利用してダウンロード、実行される。

4.2. エージェント

エージェントは検査プラグインと様々なエンティティとの通信を中継する。内部ホストへの検査には、次の 2 種類のエージェントが使用される。

- ソケットエージェント ... 検査対象ホストとの通信を中継
- 中継エージェント ... 他のエージェントとの通信を中継

4.3. 通信路の確立

検査プラグインが、検査対象ホストへの通信要求を通信管理部に送ると、図 3 の様な過程を経て、通信路が確立される。

① 通信管理部は、検査対象ホストに接続可能なコネクションサーバを探し、そこに対してソケットエージェント生成要求を送る(1)。その要求は、複数のコネクションサーバを中継して、目的のコネクションサーバに送られる(2)。

② ソケットエージェント生成要求を受け取ったコネクションサーバは、ソケットエージェントを生成する。又、①の処理過程で、途中メッセージを中継したコネクションサーバ上でも、同様に中継エージェントが生成される。

③ 末端のソケットエージェントから通信管理部まで、エージェント同士が接続する(3),(4)。

④ 最後に、検査プラグインからのソケット接続要求がソケットエージェントに送られ(5),(6)、検査対象ホストとの通信路が確立する(7)。

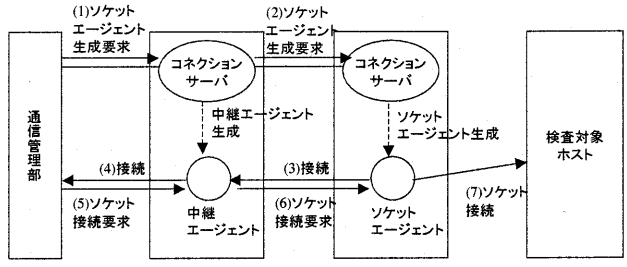

図 3検査対象ホストへの通信路確立

5. まとめ・今後の課題

侵入過程の検証機能を実現するため、**スクリプトによる侵入検査の自動化**、及び**エージェントを経由したリモート検査**という特長を持つセキュリティホール診断ツールを設計・試作した。

今後の課題として、①ラベルの命名規則の明確化、②スニファ、トロイの木馬等、長期間に渡る攻撃を用いた検査への対応が挙げられる。

参考文献

[1] Internet Security Systems,http://www.iss.net

[2] Nessus Project, http://www.nessus.org

[3] Larry.Wall 他 "Programming Perl", O'Reilly