サイエンティフィック・システム研究会　　2005 年 1 月 18 日
システム技術分科会 2004 年度第 2 回会合資料

Web サーバに対するセキュリティアセスメント技術の最新動向

富士通株式会社
奥原　雅之
okuhara@jp.fujitsu.com

[アブストラクト]

クロスサイトスクリプティング脆弱性、コマンドインジェクション脆弱性など、Web アプリケーションには特有のセキュリティ上の問題点が存在する。しかし、今日これらの問題点は必ずしも広く正確に理解されているとは言えず、今なお多くのサイトでセキュリティ脆弱性が未対策のまま運用されているのが実情である。本発表では Web アプリケーションと Web サーバに存在するセキュリティ脆弱性の実態について実例を交えて紹介するとともに、これらの脆弱性を診断するための最新手法について説明する。

[キーワード]

ウェブアプリケーション、脆弱性、アセスメント、クロスサイトスクリプティング

1. はじめに

Web アプリケーションとは、Web サーバ上でアプリケーションを実行することにより、動的に HTML のコンテンツを生成したり、入力フォームを使って利用者からの情報入力を行わせたりすることを可能にする技術である。この技術により、Web サービスシステムは従来の情報発信の仕組みから、本格的な情報処理システムへと変貌することになった。今日では、ビジネスサイトを中心に、広い範囲で Web アプリケーション技術が利用されている。しかし、Web アプリケーション技術自身は発展途上の技術であり、セキュリティの観点では十分熟成されていない面がある。その利便性のみに気をとられていると、思わぬセキュリティ上の問題点の落とし穴にはまることがありうる。実際、これまでに多くのサイトで Web アプリケーションのセキュリティ問題が発見され、社会的に問題になっている事例も多い。これらの問題を解決するには、Web サイトの管理者がまず Web サーバに発生しうる問題点を正確に把握し、正しい対処をすることが必要である。

この問題点の正確な把握のために有用なのが、Web 上のセキュリティの問題点を正確に検出するサーバセキュリティアセスメント技術である。本稿においては、Web アプリケーションおよび Web サーバに存在する代表的な脆弱性について、その概要を説明する。さらに、これらの脆弱性を発見するサーバセキュリティアセスメント技術がどのように商用サービスに活用されているかを紹介する。

2. Web アプリケーション脆弱性の実態

表 1 は、米国 WebCohort 社が 2000 年 1 月から 2004 年 1 月に約 300 サイトに対して行った脆弱性検査の結果をまとめたものである（参考文献 1）。

表 1　脆弱性の発見比率

攻撃の種類	脆弱性の見つかった割合
クロスサイトスクリプティング	80%
SQL インジェクション	62%

パラメータの改ざん	60%
クッキーの改ざん	37%
データベースサーバへの攻撃	33%
Web サーバへの攻撃	23%
バッファオーバーフロー	19%

この調査の結果、調査対象の Web アプリケーションの 92%に何らかのセキュリティ脆弱性があることがわかった。セキュリティ的にまったく問題がないサイトはわずか 8%しかなかったことになる。にわかには信じがたい数字かも知れないが、同様の結果を示す調査は他にも知られている。(米 Sanctum 社 2003 年、産業技術総合研究所 2001 年など。)

これらの調査から、現状では大部分の Web アプリケーションを持つサイトは何らかのセキュリティ脆弱性を持つと考えざるを得ない。もちろん、その脆弱性の重大度はさまざまであり、すべてが致命的な重大性を持つものではない。しかしながら、Web アプリケーションの脆弱性により、以下のような社会的に問題となったセキュリティ事件も発生している。

(1) オンラインショッピングで、消費者のクレジットカード番号が流出した事件

2000 年 1 月、Maxus と名乗るクラッカーが米国のオンライン CD ショップである CD Universe のアプリケーションの脆弱性をついて Web サイトを攻撃。クレジットカードのトランザクション処理を行う Cyber Cash の IC Verify と、SQL Server の脆弱性が原因だとされている。約 30 万人分の消費者のクレジットカード番号が盗み出され、約 2 万 5 千人のクレジットカード情報を Maxus のホームページ上に公開された。米国連邦捜査局(FBI)は、このホームページを即日閉鎖したが、Maxus は更に 10 万人分のカード情報と引き換えに 10 万ドルを支払うように恐喝した。

(2) A 協会 個人情報流出事件

2003 年 11 月、デジタル著作物の著作権保護を進めている A 協会が開設するホームページの質問フォームに記入を行った約 1200 人分個人情報が流出。セキュリティホール発見者の脆弱性公開の方法を巡り、大問題となった。

このように、Web サイトのセキュリティ脆弱性は、Web サイトにおける情報漏えい、サーバ停止といった直接的被害に加えて、サイト運用組織の信頼性低下などの計り知れないダメージを与える問題となりつつある。

3. 代表的な Web アプリケーション脆弱性

Web アプリケーションの脆弱性とはどのようなものなのか。この点についてはまだ統一された見解はないが、例えば米国 OWASP では、代表的な脆弱性カテゴリとして、以下の 10 種類をあげている。(参考文献 2)

(1) Unvalidated Input
(2) Broken Access Control
(3) Broken Authentication and Session Management
(4) Cross-Site Scripting (XSS) Flaws
(5) Buffer Overflows
(6) Injection Flaws
(7) Improper Error Handling
(8) Insecure Storage
(9) Denial of Service
(10) Insecure Configuration Management

ここでは、代表的な Web アプリケーション脆弱性とその攻撃方法について説明する。

3.1 クロスサイトスクリプティング

クロスサイトスクリプティング攻撃は、攻撃者が脆弱性を持つ Web サイトを経由して、Web サイト利用者のクライアントマシン上で Java スクリプトなどのプログラムを実行させる攻撃である。カスタマセションおよび cookie などの個人情報や重要な情報をクライアントマシンから漏洩させる、巧妙なページを偽造し Web サイト利用者に個人情報などを入力させ全く別の Web サーバに送信させる、などが可能となる。

3.2 強制ブラウジング

攻撃者は、推測によるブラウジングを実施することで、 Web サーバに格納されている情報の獲得を試みる。

3.3 想定外の入力データ（パラメータ値）による動作異常

攻撃者は、想定外のリクエストを送信することにより、Web アプリケーションに動作異常を発生させる。例えば、以下のような入力が攻撃としてよく使われる。

(a) パラメータ削除
(b) 空白パラメータ
(c) NULL 値
(d) 各種の問題を引き起こす文字（「’」、「”」、「¥’」、「¥¥’」、「」)、「;」）
(e) オーバーフロー発生パラメータ値
(f) トグル値に対する想定外の値
(g) 入力制限値の範囲外の値

3.4 クエリストリングの書き換え

クエリストリングは URL 内にパラメータを記述する方式である。これにより、Web ページ間で容易にパラメータを受け渡しできる。しかし、クエリストリングに含まれる情報は筒抜けであり、改ざんされやすい。 またこの情報から、攻撃者に Web アプリケーションの仕組みを知られる可能性もある。例えば、Web サイトの URL の“？”以降のクエリストリングから、他の利用者のユーザ ID（例:user＝10335）を設定し、パスワード（例:pw=0919）などを推測して入力することで、他の利用者の個人情報が流出する可能性がある。

3.5 hidden フィールドの書き換え

hidden フィールドを使用すれば、Web ブラウザに表示させずにページ間で容易にデータを受け渡しできる。しかし実際には、hidden フィールドで指定した情報は Web ブラウザへ送信されているため、情報の参照および改ざんが可能となる。hidden フィールドの情報を改ざんされた場合、重要な情報が漏洩する危険性がある。

4. Web セキュリティアセスメント技術

このように、Web アプリケーションの脆弱性は多岐にわたる。また、アプリケーションというその本質上、セキュリティホールのほとんどはアプリケーション自身に依存するため、定型的な手法だけではセキュリティホールを洗い出すことは難しい。従って、Web アプリケーションの脆弱性アセスメントは、高いスキルを持った診断者と、その診断を効率的に支援する診断ツールの組み合わせで実施することが必要である。

Web アプリケーション診断ツールは、すでに多くのものが市販されている。これらは一般に下記

のような機能を持つ。

(1) 全 Web 画面の列挙と遷移図の作成（クローリング）

(2) ルールセットによる定型的な攻撃手法の試行とレスポンスの記録

(3) 結果レポートの作成

一例として、富士通研究所ではアプリケーション診断ツール「玄武」を開発中である。このツールは、(1) のクローリングにおいて、ビジュアルな画面遷移図を対話的に描くことで、より適切な遷移図を作成することができる。

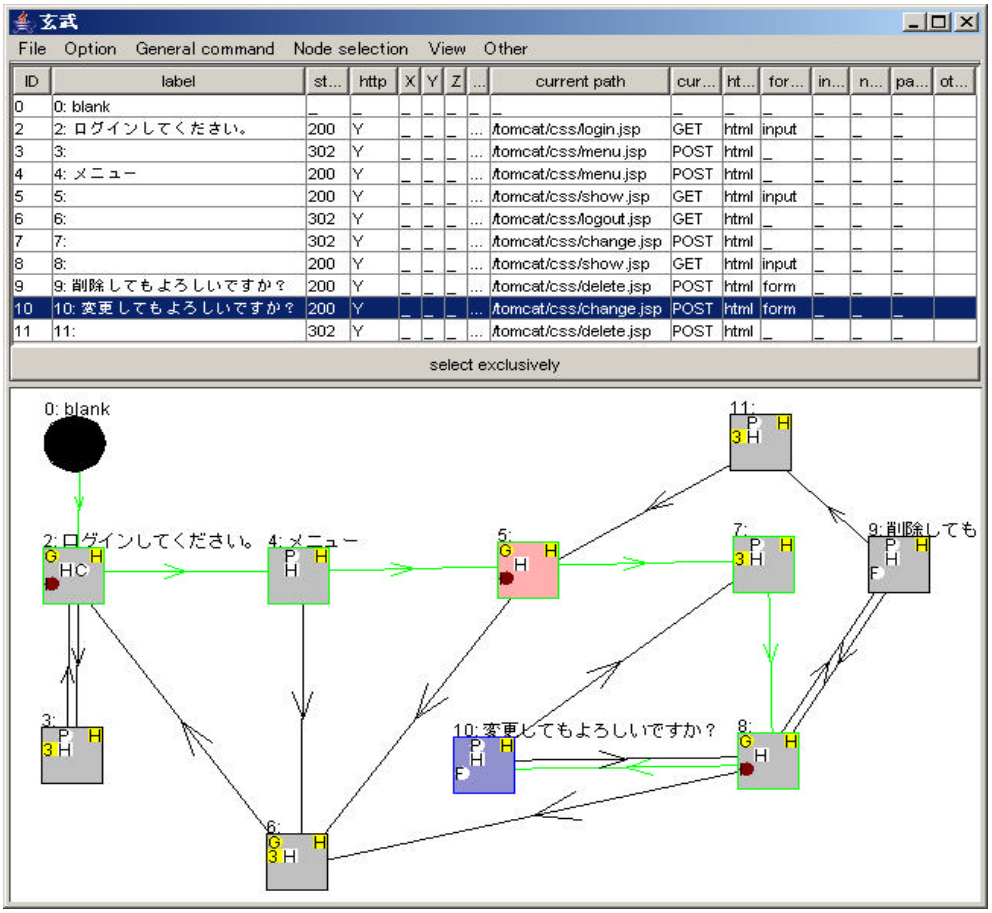

図 1 「玄武」の画面遷移図

しかし、このような診断ツールを用いても、Web アプリケーションの脆弱性検出を完全に自動化するには至っていない。現状では、かなりの部分を実施者の知識と経験によって補いながら、セキュリティアセスメントを実施している。このため、アセスメント結果、特に脆弱性の網羅性については実施者によって大きく異なる。このツールでは補いきれない部分をどのようにしてカバーしていくかが今後の課題となっている。富士通では、このような診断に必要な知識をデータベース化し、診断ガイドラインとして整理する取り組みを実施している。

また、Web サイトの安全確保には、Web アプリケーションだけではなく、プラットフォームとなる OS やミドルウェアのセキュリティ確保も重要である。Web アプリケーションとは異なり、この分野の診断ツールは古くから発展を重ね、十分実用に耐えるツールが多数提供されている。例えば、図 2 および図 3 は米国 Qualys 社が提供する「QualysGuard」という脆弱性アセスメントサービスのレポート例である。このサービスでは、3959 項目（2004 年 12 月現在）の脆弱性を検出することができる。いかに優れたサーバ管理者でも、4000 項目近い脆弱性をすべて把握することは不可能である。今日、この分野のセキュリティアセスメントはこのようなツールに頼ることが唯一の現実解であろう。

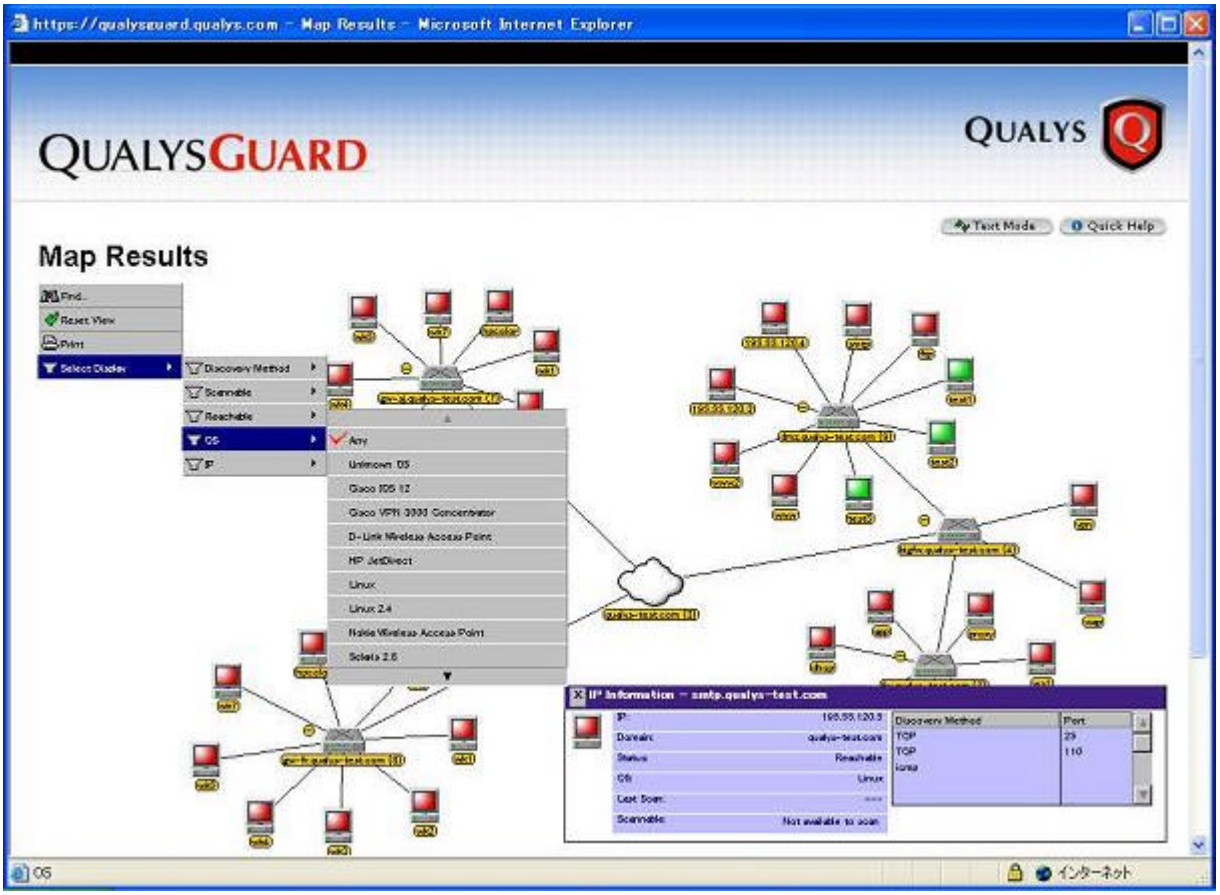

図 2　QualysGuard(Map 機能)

図 3　QualysGuard (脆弱性レポート)

[参考文献]

(1) 米 WebCohort 社プレスリリース（2004 年 2 月 2 日）
http://www.imperva.com/company/news/2004-feb-02.html

(2) The Open Web Application Security Project (OWASP)
http://www.owasp.org/