



Tokyo Research Laboratory, IBM Japan

Web 2.0アプリケーションにおける セキュリティ上の脅威とその対策

日本アイ・ビー・エム(株) 東京基礎研究所
浦本直彦

© 2007 IBM Corporation


東京基礎研究所



概要

- Web 2.0環境における脅威
 - クロスサイトスクリプティング(XSS)
 - クロスサイトリクエストフォージェリ(CSRF)
 - コンテンツの信頼性
 - その他
- 対策
 - 入力の検証
 - コードの動的な実行の回避
 - <iframe>タグの使用
 - 脆弱性チェックツールの使用
- IBMや東京基礎研究所の取り組み

© 2007 IBM Corporation

東京基礎研究所 

Web 2.0アプリケーションの特質

- 使い勝手のよいユーザーインターフェースと、生産性の高いプログラミング手法の両立
 - Ajax, マッシュアップ
- Social Computing
 - ユーザがデータを作成・共有 (User Generated Content)
 - Wiki, ブログ, SNS
- 企業にとっても魅力が高まる
 - 市場やビジネスモデルの急速な変化に対応
 - 既存のコンポーネントやAPIを結合
 - 状況依存型アプリケーション (Situational Application)
- 「簡単で使いやすい」→セキュリティ上の脅威に対して脆弱
 - エンタープライズレベルでの利用には信頼性やセキュリティが不可欠
 - セキュリティに関する細かな知識や作業を要求せずに、安全性を実現するのが課題

© 2007 IBM Corporation

東京基礎研究所 

Web 2.0とセキュリティ

Web 2.0技術の特徴	典型的なアプリケーション	セキュリティ上の問題
コンテンツ指向のWeb2.0アプリ	SNS, Blog, Wiki, ソーシャルブックマーク	悪意を持ったコードの注入が容易
JavaScriptの多用	ブラウザベース Web API Webメール	ブラウザ上での実行コードの追加や上書きなどの問題
Web2.0アプリ実行環境としてのブラウザ	マッシュアップ 軽量クライアント	ブラウザのセキュリティ機能が不十分

© 2007 IBM Corporation

東京基礎研究所 

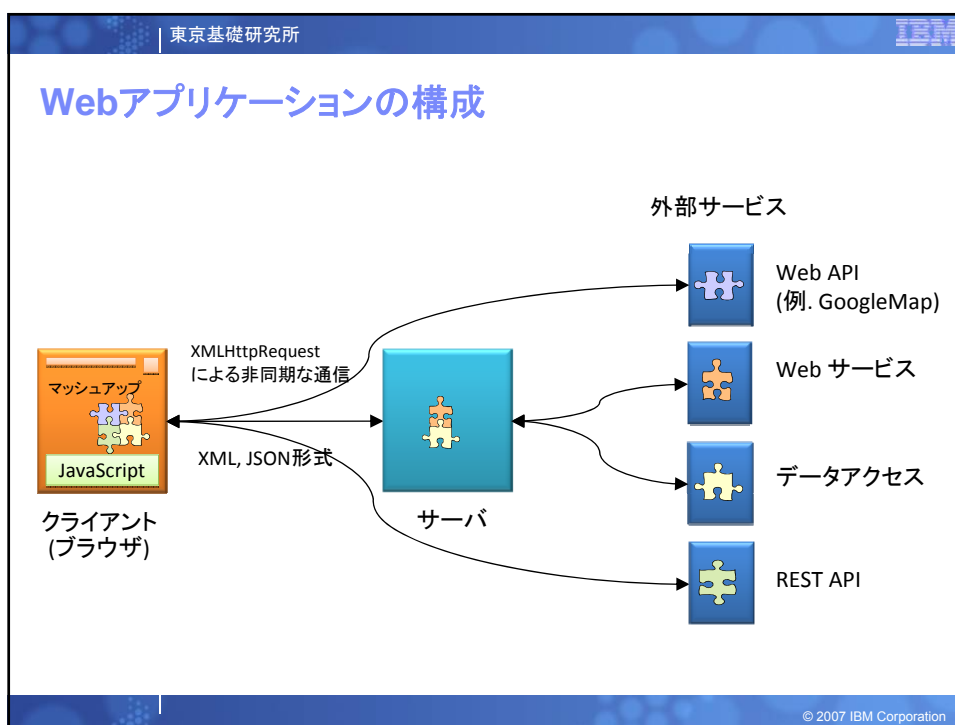
Web アプリケーションにおける脅威の変遷

From OWASP Top Ten Project

2004	2007
<ul style="list-style-type: none"> ▪ A1 Unvalidated Input ▪ A2 Broken Access Control ▪ A3 Broken Authentication and Session Management ▪ A4 Cross Site Scripting (XSS) ▪ A5 Buffer Overflow ▪ A6 Injection Flaws ▪ A7 Improper Error Handling ▪ A8 Insecure Storage ▪ A9 Application Denial of Service ▪ A10 Insecure Configuration Management 	<ul style="list-style-type: none"> A1 Cross Site Scripting (XSS) A2 Injection Flaws A3 Malicious File Execution A4 Insecure Direct Object Reference A5 Cross Site Request Forgery (CSRF) A6 Information Leakage and Improper Error Handling A7 Broken Authentication and Session Management A8 Insecure Cryptographic Storage A9 Insecure Communications A10 Failure to Restrict URL Access

http://www.owasp.org/index.php/OWASP_Top_Ten_Project

5 © 2007 IBM Corporation

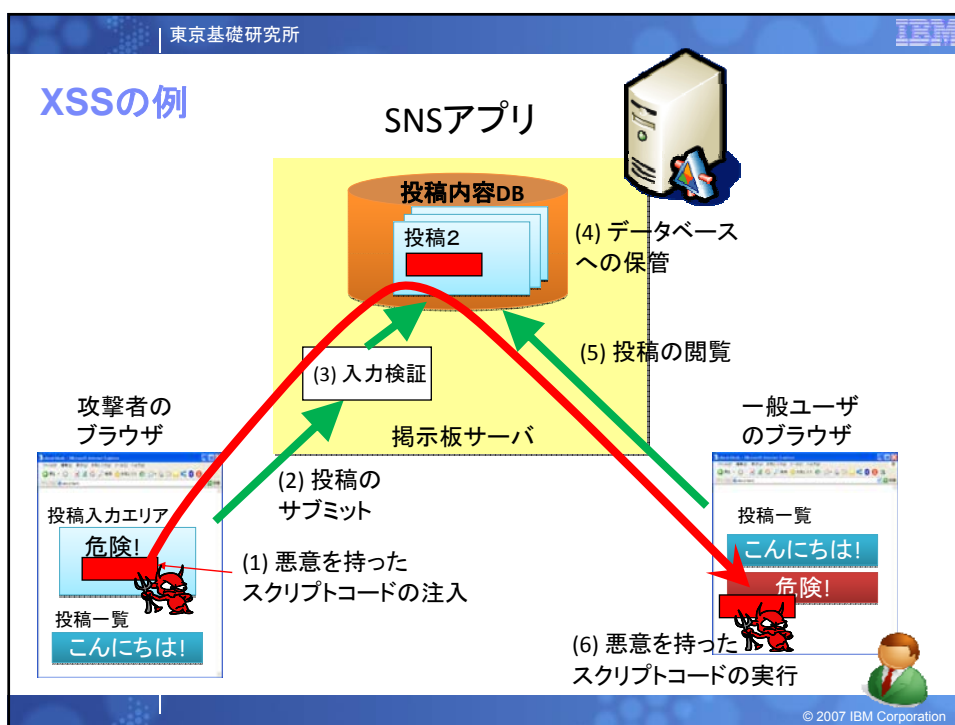


東京基礎研究所 IBM

Web 2.0環境における脅威(1): XSS

- Webアプリケーションが動的に生成するHTMLの中に、悪意を持ったスクリプトを混入させる
- ユーザーがこのHTMLにアクセスすると、スクリプトが実行される
- Web 2.0では複数ユーザーからの入力が発生するため、特に脅威となる
 - (従来のWebアプリケーションにも存在する問題)

© 2007 IBM Corporation

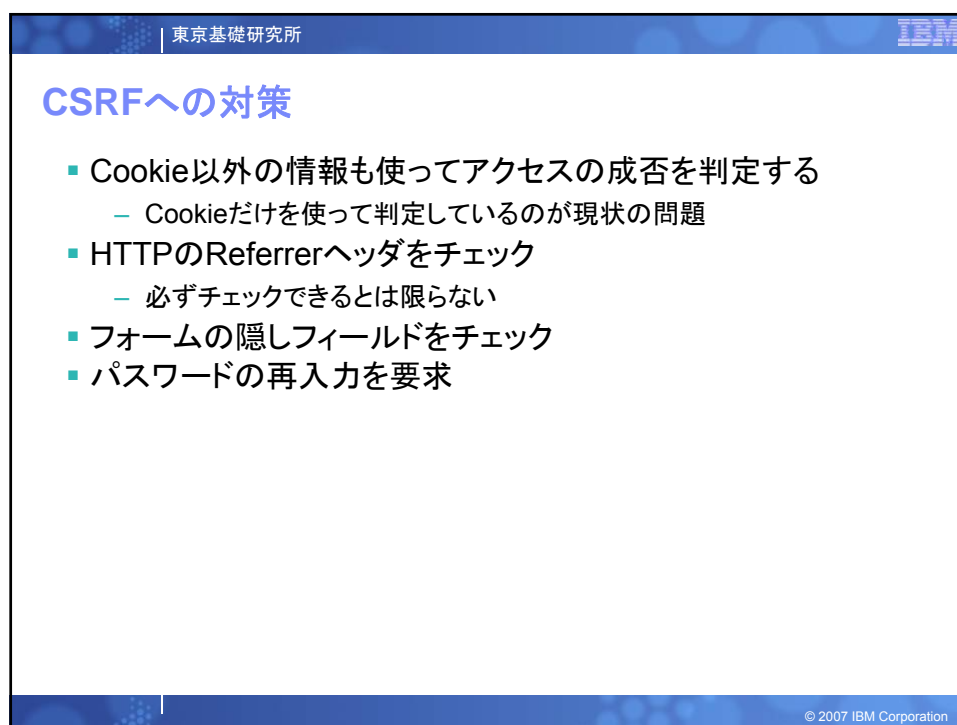
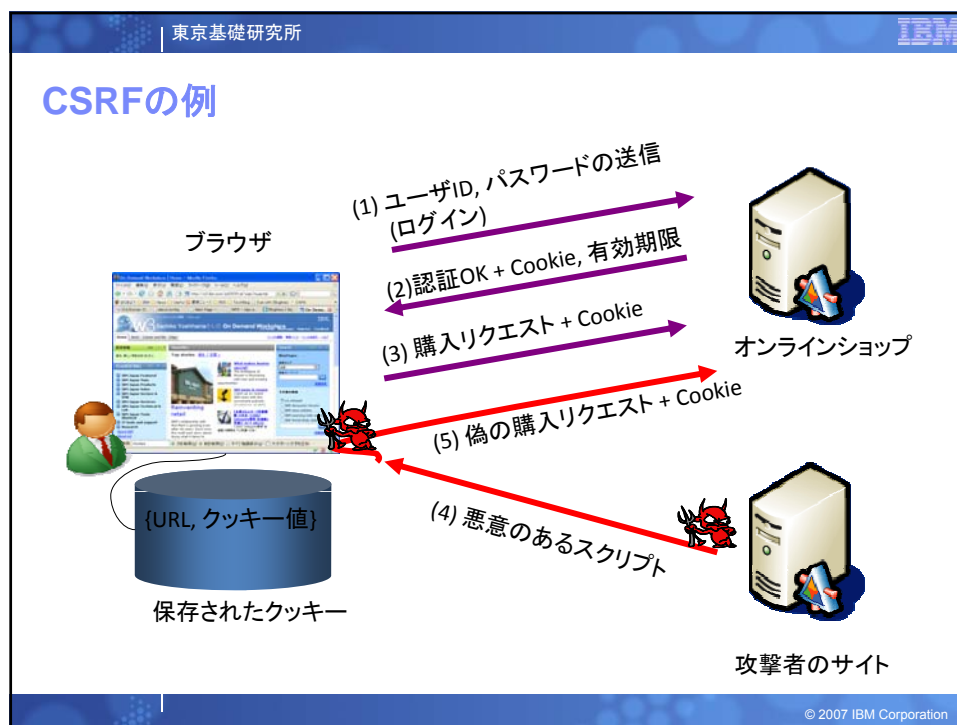


XSSでの問題点

- コンテンツは信頼できるサーバからダウンロードされる
 - 気づかずにスクリプトが実行されてしまう危険性
 - サーバ単位で信頼の可否を指定する手法の限界
- 攻撃者は入力検証の回避を日々試みている
 - `<script src="...">`
 - ブラウザの寛大さにも問題の一端
 - 100%確実な検証(フィルタリング)は不可能
 - いたちごっこ

Web 2.0環境における脅威(2): CSRF

- Web経由でコマンドを受け付けるアプリケーションのほとんどに適用可能
- ユーザーがWebサイトにログインすると発行されるCookieを悪用
 - ログインしたWebサイトへのアクセスには、必ずCookieが送付される
- 攻撃手順
 - 攻撃者のWebページに、上記Webサイトへのコマンド(商品の購入、記事の投稿など)を表すURLへのアクセスが自動実行されるようなスクリプトを記述
 - 被害者が、Cookieを受け取っている状態でこのページにアクセス
 - Webサイトは(Cookieが含まれているため)認証済みのユーザーによる正規のリクエストと判断し、コマンドが実行されてしまう



Web 2.0環境における脅威(3): コンテンツの信頼性

- 個人が生成した情報の蓄積(集合知)が価値を生む
 - 例: Wikipedia
 - 従来は企業・団体や少数の権威者だけが発信
- 誹謗中傷も大きな力を持つてしまうことに
 - 1人のユーザーの心無い書き込みがエスカレート
 - 認証によってある程度は防止可能
- 個人情報やその他の機密情報をうっかり書き込んでしまう危険性
 - 不適切な語句を排除するフィルタが求められる

Web 2.0環境における脅威(4): その他の問題

- Ajaxによって、ユーザーの明示的な操作なしに通信が行われる危険性
 - 良い例: メール自動保存、1文字入力する毎の検索候補の表示
 - 悪い例: キーロガー
- JSON形式のデータに対する誤った処理
 - JSON形式のデータはJavaScriptのコードとしても使えるため、そのまま読み込んで実行してしまっているアプリケーションやライブラリが見られる
 - スクリプトが混入している危険性

セキュアなWeb 2.0環境構築のために

- Web 2.0はWeb 1.0(従来のWeb)の上に成り立っている
 - Web 1.0向けのセキュリティの重要性は変わらない
 - 認証、セッション管理、SSL、...
- 入力の検証
 - 意図しないスクリプトの混入(XSS)を防ぐ
 - 検証のための有用な関数が用意されている言語も
 - ブラックリスト方式・ホワイトリスト方式
- コードの動的な実行の回避
 - ユーザーが入力したデータをそのまま実行しない
 - 十分な検証が必要

セキュアなWeb 2.0環境構築のために(続き)

- <iframe>タグの使用
 - マッシュアップされたコンポーネント間でのアクセスを禁止したい場合
 - 例: 悪意を持った広告サイトのコンテンツが、ショッピングサイト上に入力されたクレジットカード番号を盗む
 - <frame>の代わりに<iframe>を使うとインタラクションを遮断
- 脆弱性チェックツールの使用
 - Web 2.0アプリケーションは攻撃者にとって主要な対象の1つ
 - 新しい攻撃が日々生まれる
 - 信頼性の高いツールやサービスを利用するのが有効

IBMや東京基礎研究所の取り組み

- IBMは、Webアプリケーションをセキュアにするためのさまざまな製品やサービスを提供中
 - Tivoli
 - ISS
 - セキュリティ・アプライアンス(侵入防御・異常検出など)
 - WatchFire
 - Webアプリケーションの脆弱性検査を自動化
- 東京基礎研究所は、Web 2.0環境におけるエンド・ツー・エンドのセキュリティに関する研究開発を行う
 - コンテンツ・フィルタリング
 - 安全なマッシュアップのためのライブラリ
 - 次世代ブラウザ(Web 2.0アプリケーション実行環境)
 - OpenAjax Allianceのセキュリティ部会に参加

Tivoli. software



watchfire
an IBM company