大学・研究所における
Webアプリケーションのセキュリティ

「守るべき情報を持たない」
Webサイトにおけるセキュリティの在り方

すべては最適のために。
SofTek
Security Leading Company

株式会社 ソフテック
セキュリティソリューション事業部
事業部長　芝田 幸彦

すべては最適のために。 SofTek Security Leading Company

アジェンダ

- Webアプリケーションにおける脆弱性の現状
- 大学・研究サイトにおけるWebアプリケーション
  セキュリティの認識
- セキュアなWebアプリの開発・運用に必要なポイント
- まとめ

すべては最適のために。
SofTek
Security Leading Company

今日のポイント

- 大学・研究所サイトにおいては、
  - Webアプリケーションセキュリティに対する
    リスク認識が低いのでは？
- 大学・研究所サイトにおいても、
  - 安全なWebサイトを構築するために必要なことは
    セキュアなWebアプリケーションの開発・運用である

    --> どうやって実現すれば良い？

すべては最適のために。
SofTek
Security Leading Company

ソフテック　会社紹介

- 会社概要
  - 会社名　:株式会社ソフテック　SofTek Systems, Inc.
  - 本社　:〒154-0004
    世田谷区太子堂1-12-39 三軒茶屋堀商ビル
  - 設立　:1991年3月25日
  - 資本金　:4,000万円

- 役員
  - 代表取締役会長　武田喜一郎
  - 代表取締役社長　加藤努
  - 取締役　芝田幸彦
  - 監査役　大林善次(公認会計士)
  - 技術顧問　高木浩光

独立行政法人 産業技術総合研究所
情報セキュリティ研究センター
ソフトウェアセキュリティ研究チーム 主任研究員

すべては最適のために。
SofTek
Security Leading Company

ソフテックが提供するセキュリティサービス

すべては最適のために。
SofTek
Security Leading Company

Section. 1

Webアプリケーションにおける脆弱性の現状

SofTek Security Leading Company

いま、何が起きているのか？

- 話題になった不正アクセス事件（年表）

2002　クロスサイトスクリプティング
：
2005　SQLインジェクション　（カカクコム等）
　　クロスサイトリクエストフォージェリ（ミクシィ）
：
2008　SQLインジェクション　（サウンドハウス等）

- 繰り返し、かつ成功する攻撃
- SQLインジェクションによる大量の個人情報漏洩が深刻に

SofTek Security Leading Company

いま、何が起きているのか？

- 公開されるWebアプリケーション脆弱性の傾向

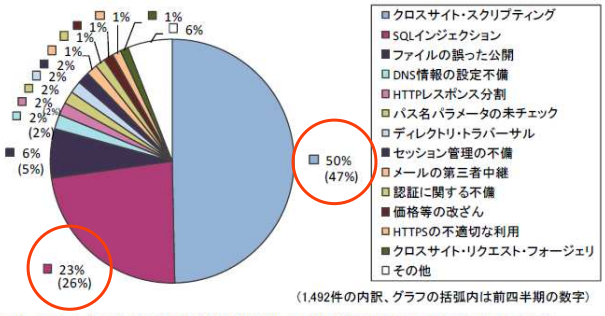


図2-2.ウェブサイトの脆弱性 種類別内訳（届出受付開始から2008年6月末まで）

- SQLインジェクション＋クロスサイトスクリプティングが届けられる脆弱性の **73%** を占める

IPA:JPCERT/CC ソフトウエア等の脆弱性関連情報に関する届出状況［2008年第2四半期（4月～6月）］より抜粋

SofTek
Security Leading Company

いま、何が起きているのか？

- 2008年に入ってから届出件数が激増している

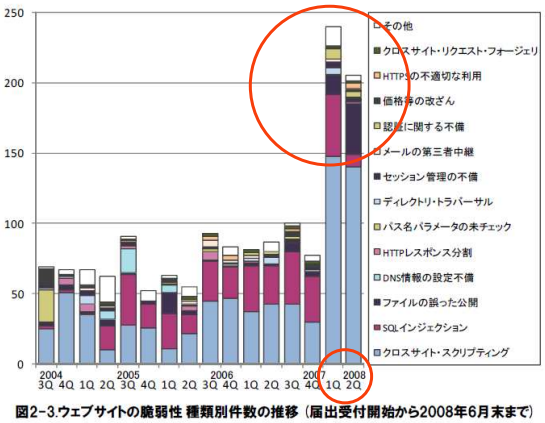


図2-3.ウェブサイトの脆弱性 種類別件数の推移（届出受付開始から2008年6月末まで）

IPA:JPCERT/CC ソフトウエア等の脆弱性関連情報に関する届出状況 [2008年第2四半期(4月～6月)] より抜粋

SofTek
Security Leading Company

いま、何が起きているのか？

- 不正アクセスの目的は、「目立って嬉しい」より「マネー」へ
  - 以前は、愉快犯や自己顕示欲が大半を占めていたが…
  - 現在は、金銭目的にハッキリ変わってきている
    - フィッシング詐欺団とかRMT(リアルマネートレーディング)など
    - クレジットカード情報を始めとする個人情報の漏洩
  - 一般ユーザを犯罪に巻き込む可能性のある悪質なものも
    - 犯行予告検知システムに潜在する脆弱性アクセスしただけで、掲示板に犯行予告を書き込むトラップ（罠）が仕掛けられる

すべては最適のために。
SofTek
Security Leading Company

いま、何が起きているのか？

● 組織的かつ広域的なフィッシング詐欺グループの存在

京都府警など、フィッシングで盗んだIDを使ったオークション詐欺団を摘発

京都府警ハイテク犯罪対策室と宇治署、静岡県警、熊本県警は30日、フィッシングサイトにより盗んだIDを利用してオークション詐欺を行なったとして、東京都の男性（34歳）を不正アクセス禁止法違反および詐欺の疑いで逮捕した。また、この男性の他にも7人を逮捕する予定で、**組織的かつ広域的なフィッシング詐欺グループの摘発となる**

京都府警によると、容疑者らは2005年9月に、フィッシングサイトにより入手したIDとパスワードを利用してオークションサイトに腕時計を架空出品し、落札した東京都の男性から24万円を振り込ませた疑いが持たれている。また、2006年4月にも同様の手口でCDデッキを架空出品し、愛知県の男性に25 万円を振り込ませたという。

新聞報道などによれば、**容疑者らはヤフーの偽サイトによりIDとパスワードを入手。Yahoo!オークションで犯行を繰り返しており、被害者は約700人、被害総額は約1億円に上るとみられるという。**

http://internet.watch.impress.co.jp/cda/news/2006/05/30/12130.html より抜粋

すべては最適のために。
SofTek
Security Leading Company

いま、何が起きているのか？

● 金銭目的から恐喝行為に及ぶ事件も

NHN Japan、未成年者によるフィッシング事件を受けて会員に注意喚起

オンラインゲームポータルサイト「Hangame（ハンゲーム）」を運営するNHN Japanは30日、同サイトの会員を狙ったフィッシング事件で未成年者が摘発されたことを受け、ID・パスワードの管理について注意を喚起する「クリー ンコミュニティキャンペーン」を開始した。

（中略）

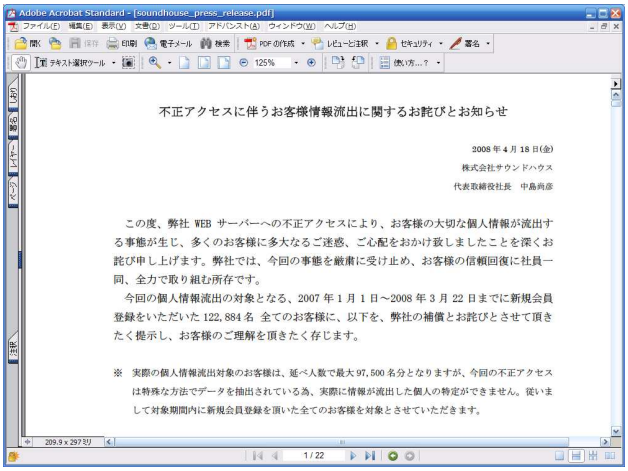
警視庁ハイテク犯罪対策総合センターと池袋署が同日、**Hangameを装ったサイトを開設してフィッシング行為をしたとして、名古屋市内に住む14歳の少年を、不正アクセス禁止法違反と著作権法違反の疑いで書類送検**したことを受けてのもの。**少年は、偽サイトで94人分のIDなどを不正に入手していた疑いが持たれている。**NHN Japanから3月に被害届が提出され、捜査が進められていた。

NHN Japanによると、事件は2月から3月にかけて発生した。Hangameの運営スタッフを装って送信されたミニメール（Hangameのサイト内メール）を通じて、**Hangameの「お問い合わせフォーム」を装ったフィッシングサイトに誘導し、会員のIDやパスワードを入力させていた**という。さらに、**この方法で入手したIDとパスワードを使ってHangameにアクセスして不正利用したり、パスワードを変更するなどの行為を繰り返していた。**このほか一部報道によれば、**パスワードを変更されてアクセスできなくなった正規の会員に対して、パスワードを返す代わりにわいせつ画像を送信するよう要求**していたと いう。

http://internet.watch.impress.co.jp/cda/news/2006/05/30/12139.html より抜粋

いま、何が起きているのか？

- SQLインジェクションを使った情報漏洩も依然として発生
  - サウンドハウスにおける個人情報漏洩事故



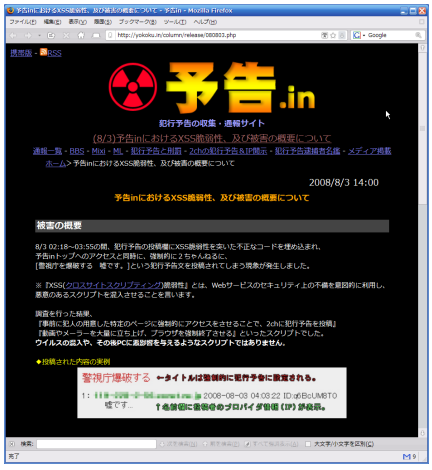
http://www.soundhouse.co.jp/news/20080418.pdf より抜粋

いま、何が起きているのか？

- 第三者を犯罪に巻き込む可能性も発生

「予告.in」サイトに潜在する
「クロスサイトスクリプティング」
脆弱性を悪用。

「予告.in」サイトにアクセスした
だけで、「2ちゃんねる」に自動的に
犯行予告を書き込むトラップが
仕掛けられた。



http://yokoku.in/column/release/080803.php より抜粋

すべては最適のために。
SofTek
Security Leading Company

Webアプリケーションの脆弱性について

Webアプリケーションに関する脆弱性リスト

- SQL インジェクション
- クロスサイトスクリプティング(XSS)
- パラメータの改ざん
- Hidden フィールドの不正操作
- バッファオーバーフロー
- シェルコマンド・インジェクション
- OS コマンド・インジェクション
- 強制ブラウジング
- サードパーティ製品の設定ミス
- 既知の脆弱性
- バックアップファイルの検出
- バックドアとデバッグオプション
- HTML 中のコメント
- ディレクトリトラバーサル
- 不適切なエラーハンドリング
- クロスサイトリクエストフォージェリ(CSRF)
- セッション管理に関する脆弱性

すべては最適のために。
SofTek
Security Leading Company

Webアプリケーションの脆弱性について

- 代表的なWebアプリケーション脆弱性
  - SQLインジェクション
    - SQLデータベースに対し、外部から任意のSQLを実行できる脆弱性で、データベースから任意のデータが抽出される可能性があります。
  - クロスサイトスクリプティング
    - 入力フォームやURLに含まれるパラメータ等に設定した細工された文字列が、JavaScriptを含む任意のHTML要素としてWebページに埋め込まれることで、ページの偽装やセッションIDの不正取得等が行われる可能性があります。

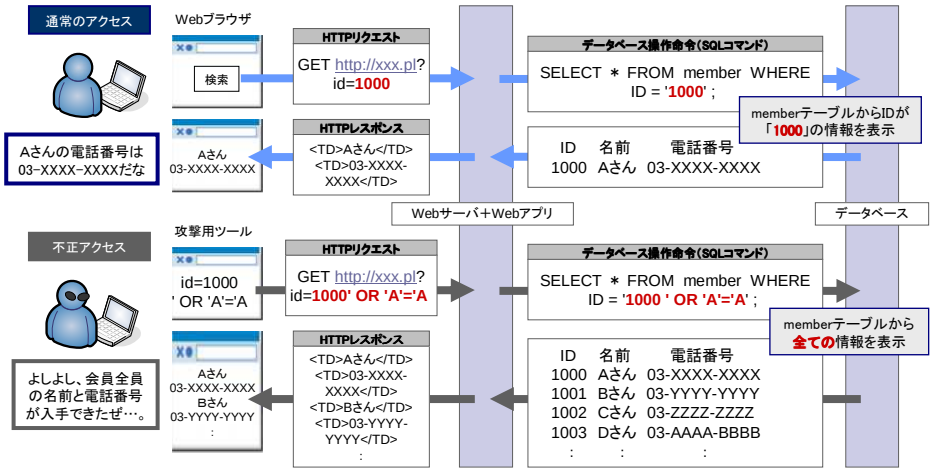
Webアプリケーションの脆弱性について

- OSコマンドインジェクション
  - 外部からサーバに対し任意のOSコマンドを実行することが可能な脆弱性です。例えば、Webブラウザ上のフォームから入力されたデータがそのままシェルにOSコマンドとして渡せるようになっている場合などが考えられます。

Webアプリケーションの脆弱性について

- SQLインジェクション

ユーザからの「細工された」文字列がSQLコマンドに挿入される

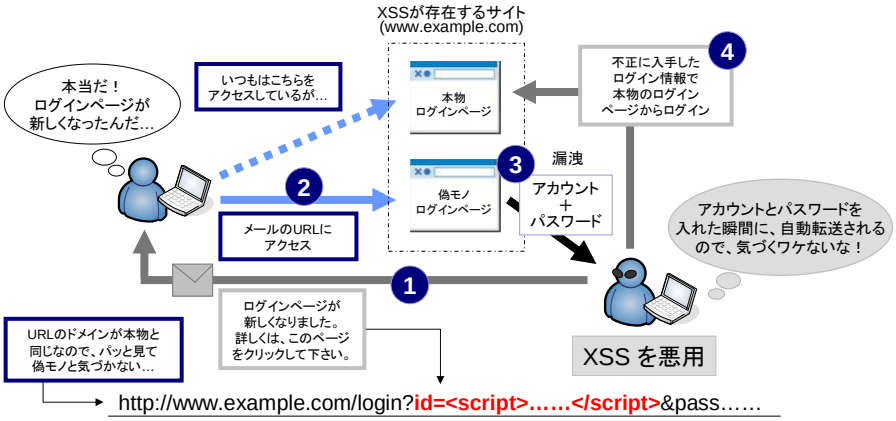


Webアプリケーションの脆弱性について

- クロスサイトスクリプティング

ユーザからの「細工された」スクリプトがそのまま実行される

ユーザからの不適切な設定データに対するWebアプリ側での「チェックが甘い」



Webアプリケーションの脆弱性について

- 攻撃パターンによる分類
  - 「悪用のされやすさ」（被害の大きさ）に繋がる

	脆弱性	主な目的	利用方法
直接的攻撃	SQL インジェクション	DB情報	データベースに 直接アクセス
間接的攻撃	クロスサイト スクリプティング	認証情報	トラップを仕掛けて ユーザを誘導
直接的攻撃	コマンド インジェクション	サーバ 乗っ取り	Webページ経由で シェルコマンドを 実行

すべては最適のために。
SofTek
Security Leading Company

大学・研究サイトにおける Webアプリケーションセキュリティの認識

すべては最適のために。
SofTek
Security Leading Company

Webサイトへのセキュリティについて積極的ではない？

- 「守るべき情報があまり無いので…」
  - 個人情報漏洩ばかりがクローズアップ
  - そのためか、学術系サイトでは、Webサイトセキュリティはあまり積極的ではないように見える

すべては最適のために。
SofTek
Security Leading Company

Webサイトへのセキュリティについて積極的ではない？

- 「情報漏洩が発生したとしても大した被害にならない」
  - この判断は「致命的なミス」といえる
  - 個人情報を持っていなくても、不正アクセスの標的となり、被害を受けている現状を理解するべき

すべては最適のために。
SofTek
Security Leading Company

Webセキュリティ被害は個人情報がなくても発生する

- 例えば、今年のセキュリティ事例で考えてみる
  - SQLインジェクション攻撃の傾向も二つに大別できる
    - 個人情報漏洩

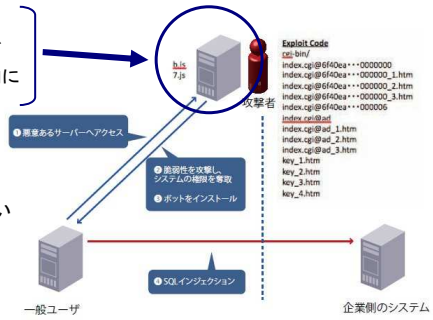
      クレジットカード情報の漏洩 ←--- 学術系サイトにとってはレアケース
    - ウィルス/マルウェア汚染

      SQLインジェクションの脆弱性がWebサイト上にあるだけで、第三者に「ウィルス/マルウェア」をばら撒く作業に加担することに

SofTek
Security Leading Company

Webセキュリティ被害は個人情報がなくても発生する

- SQLインジェクション・ワームによるウィルス/マルウェア汚染
  - SQLインジェクションを悪用し、ウィルス/マルウェアを置いたサイトからダウンロードさせる Javascript を脆弱なWebサイト内に埋め込むことで、そのサイトにアクセスしたPCを汚染させる。
  - ダウンロードしたウィルス/マルウェアがボットだった場合、ボットがGoogle等を使い脆弱性のあるWebサイトを検索してそのサイトにSQLインジェクション攻撃を行い、攻撃に成功すると、またウイルス/マルウェアを仕掛ける（以下、繰り返し）。



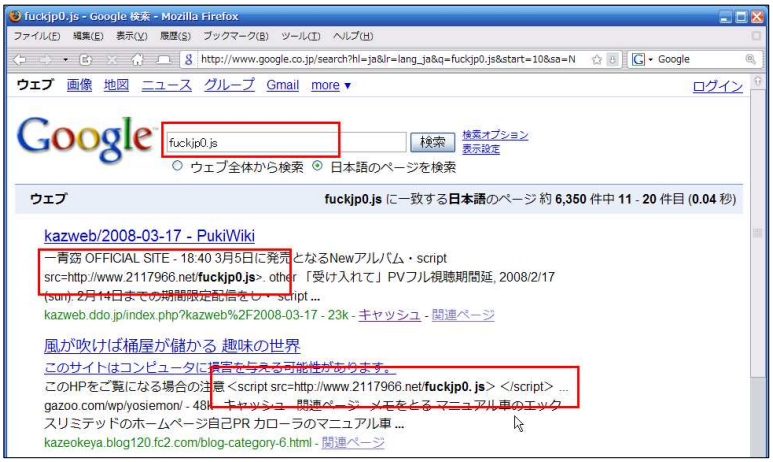
Webセキュリティに十分な取組みをしていない組織であることが周知になってしまい、組織自体の信頼性に悪影響を与える。

http://itpro.nikkeibp.co.jp/article/NEWS/20080821/313180/ph2.jpg より抜粋

SofTek
Security Leading Company

Webセキュリティ被害は個人情報がなくても発生する

- SQLインジェクション・ワームによるウィルス/マルウェア汚染



Copyright © 2008 SofTek Systems, Inc. All rights reserved.

すべては最適のために。
SofTek
Security Leading Company

自分で開発/メンテナンスする文化

- 自分で開発/メンテナンスする文化
  - 小規模なサイトが多い
  - 研究者レベルで構築・メンテナンス
  - 自分で開発するのが悪いのではない

すべては最適のために。
SofTek
Security Leading Company

自分で開発/メンテナンスする文化

- 研究者レベルで構築・メンテナンス
  - 「正しく動くだけでOKの時代」は終わった
  - Webアプリケーションの品質は「正常動作」とともに「安全動作」が要求される社会情勢

すべては最適のために。
SofTek
Security Leading Company

自分で開発/メンテナンスする文化

- 自分で開発・メンテするのが悪いのではない
  - 予算や体制の問題で仕方ない面も
  - 正しいセキュリティ知識を持たずに開発するのが危険

セキュアなWebサイトの開発方法論を知ることが重要。
「それが難しい」というなら、アウトソースすべき

すべては最適のために。
SofTek
Security Leading Company

外部発注における問題点

- アウトソースする際には、様々な問題が
  - 曖昧なセキュリティ仕様しか書けない…
    - セキュリティに関する知識不足
    - 開発側は、仕様に書いてある通りに作るもの
      （曖昧な仕様では、低いレベルのものしか出来ない）
    - 受入側も、いい加減なレベルでしか検収できない
      （何を元にチェックしていいのか？指針がない）

すべては最適のために。
SofTek
Security Leading Company

外部発注における問題点

- アウトソースする際には、様々な問題が
  - 将来的なメンテナンスについて考慮していない
    - アプリケーションのライフサイクルについては？
      - 例えば、PHP4 問題

        2007年12月31日　メンテナンス終了
        2008年　8月　8日　脆弱性対応終了

        既にメンテナンス終了したアプリケーションが新規開発の仕様に…

「アウトソースする/しない」にかかわらず、仕様を適切に
作成することが、安全なWebサイトを構築・運用する近道

すべては最適のために。
SofTek
Security Leading Company

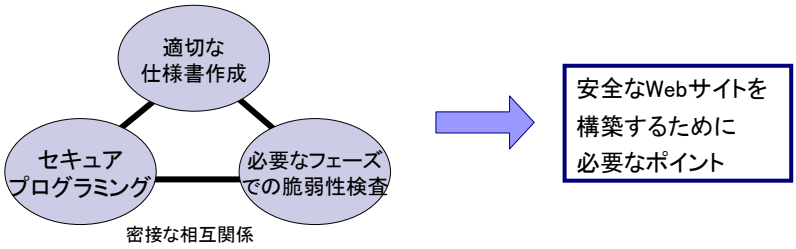
安全なWebサイト構築の本質的な点は、どこでも同じ

- 大学・研究サイトでも、やるべきことは民間サイトと同じ
- セキュアなWebアプリケーションの開発・運用が
  本質的な対策方法であることには変わりはない！

それを実現するために必要なポイントを整理して
自分のサイト規模に合わせた、Webサイト構築を
実施することが重要。

セキュアなWebアプリケーションの開発・運用に必要なポイント

- 適切な仕様書作成
- セキュアプログラミング
- 必要なフェーズでの脆弱性検査



すべては最適のために。
SofTek
Security Leading Company

適切な仕様書の作成

- セキュリティ要件の明確化
  - アウトソースが必要な状態なら重要なファクタなのだが…
  - Webアプリの脆弱性に対する責任は「発注側にある」という認識が低い
  - ガイドラインを活かした仕様策定が効果的

すべては最適のために。
SofTek
Security Leading Company

適切な仕様書の作成

- お世辞にも誉められない現状
  - 未だに「1行仕様」がまかり通っている
    - 「セキュリティに対して十分に考慮して…」
  - 前例主義による「不適切な仕様」のコピー&ペースト
    - PHP4 など、ライフサイクルに考慮してなかったり…
  - 開発されるWebアプリの品質に完全に依存するのに…

すべては最適のために。
SofTek
Security Leading Company

適切な仕様書の作成

- セキュリティ品質は、全て「仕様」が決める
  - 仕様に書かれてないものは作れるわけがない
    - 開発者にそれ以上のものを求めるのが、無理難題
  - セキュリティ要件＝開発コストを正しく理解する必要
    - 「安かろう悪かろう」を払拭しなければならない

すべては最適のために。
SofTek
Security Leading Company

適切な仕様書の作成

- ガイドラインを有効活用した仕様策定を

「高等教育機関の情報セキュリティ対策のためのサンプル規定集」

国立情報学研究所
　国立大学法人等における情報セキュリティポリシー策定作業部会

電子情報通信学会
　ネットワーク運用ガイドライン検討WG

http://www.nii.ac.jp/csi/sp/

- Webアプリケーション開発における仕様策定に便利

A3112　ソフトウェア開発における情報セキュリティ対策実施手順（策定手引書）
A3113　外部委託における情報セキュリティ対策に関する評価手順

SofTek
Security Leading Company

適切な仕様書の作成

A3112 ソフトウェア開発における情報セキュリティ対策実施手順（策定手引書） 抜粋

P.388
【クロスサイトスクリプティングに関するテストを実施する場合】

(7) 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に関する脆弱性の検証を行うこと。

攻撃者は、入力データを外部に出力する処理を実行しているアプリケーションにおいて、情報の妥当性の検証の脆弱性を突いて、悪意あるスクリプトを利用者のWeb ブラウザ上で実行する可能性がある。これによって正規の利用者のセッション情報を盗んだり、偽のページを表示させてフィッシング詐欺等に利用することがある。

チェック対象となる脆弱性レベルまで、具体的に記述する

http://internet.watch.impress.co.jp/cda/news/2006/05/30/12130.html より抜粋

SofTek
Security Leading Company

セキュアプログラミング

- セキュアプログラミングポリシーの重要性
  - 開発環境に合わせたセキュリティ面を考慮したプログラミング規約
  - 開発アプリのセキュリティレベルを個人のスキルに依存せずに底上げすることが目的

「プログラミング規約」サンプル

項目	説明
クロスサイトスクリプティング	
全般	Webページを構成する要素として、Webページの本文やHTMLタグの属性値などに相当する全ての出力要素にエスケープ処理を行う。
Webページの表示に影響する特別な記号文字（「<」、「>」、「&」など）を、HTMLエンティティ文字（「<」、「>」、「&」など）に置換する。	脆弱性防止の観点からエスケープ処理が必須となるのは、外部からウェブアプリケーションに渡される「入力値」の文字列や、データベースやファイルから読み込んだ文字列、その他、何らかの文字列を演算によって生成した文字列などだが、テキストとして出力するデータ全てにエスケープ処理を行う。
HTMLタグを出力する場合は、その属性値を必ず「"」（ダブルクォート）で括るようにする。 また、「"」で括られた属性値に含まれる「"」については、HTMLエンティティ文字「"」にエスケープする。	同上
<script>…</script> 要素の内容を動的に生成しないようにする。	Webページに出力する<script>…</script>要素の内容が、外部からの入力に依存する形で動的に生成される場合、任意のスクリプトが埋め込まれてしまう可能性がある。 危険なスクリプトだけを排除する方法も考えられるが、危険なスクリプトであることを確実に判断することは難しいため、<script>…</script>要素の内容を動的に生成する仕様は、避けることが望ましい。

プログラムをする際のポリシーを簡潔にまとめて記述する

- チェックリストによる開発者のセルフチェック
  - ポリシーに沿って開発されているかをチェック
  - チェック内容を実際のリクエストレベルまで具体化
  - ポリシーとの整合性があれば、脆弱性検査ツールを併用することも可能

「チェックリスト」サンプル

チェック項目	チェック内容	チェック結果	コメント
クロスサイトスクリプティング			
(1) 入力フォームからデータ入力を行う場合			
’><s>test</s>	入力した文字列がそのまま表示されること		
"><s>test</s>	入力した文字列がそのまま表示されること		
><s>test</s>	入力した文字列がそのまま表示されること		
%27%3E%3Cs%3Etest%3C%2Fs%3E	「’><s>test</s>」が表示されること		
%22%3E%3Cs%3Etest%3C%2Fs%3E	「"><s>test</s>」が表示されること		

具体的な検証用のコードを記述する

仕様を元にした検査基準を記述する

実際にチェックをする場合の例:

正常なリクエスト

http://www.example.com/?login=name&pass=pass

チェックする場合のリクエスト

http://www.example.com/?login=**"><s>test</s>**&pass=pass

- 対象となるパラメータに対して、チェックデータを設定して、入力したHTMLタグやJavascriptが実行されないかをチェック
- POSTデータの場合は、HTTPキャプチャツールを使ってチェックデータをパラメータに設定する

- 実際に開発する際のガイドライン
  - 安全なウェブサイトの作り方　改訂第3版　（IPA *）
    http://www.ipa.go.jp/security/vuln/documents/website_security.pdf
  - セキュア・プログラミング講座　（IPA）
    http://www.ipa.go.jp/security/awareness/vendor/programmingv2/

＊ IPA ： 独立行政法人 情報処理推進機構

- 「最低限」これだけは意識して開発してほしいポイント

インジェクション系の脆弱性についての対応

SQLインジェクション・クロスサイトスクリプティングへの「基本的な対策」

- 入力値のチェック
  - formからの入力だけでなく、パラメータの内容についてもチェック
  - プログラム仕様に沿っているかをチェック
- 出力時におけるエスケープ
  - HTMLやSQLコマンドを構成する際に特殊文字をエスケープ

すべては最適のために。
SofTek
Security Leading Company

セキュアプログラミング

HTMLにおけるエスケープ処理

変換前	変換後
<	<
>	>
&	&
″	"
’	'

SQLにおけるエスケープ処理

変換前	変換後
’	’’ あるいは ¥’
¥	¥¥

多くのプログラミング言語のSQLライブラリに用意されているバインドメカニズムの利用との併用が望ましい。バインドメカニズムを利用すると、SQL文が確定したものになるため、SQLインジェクションの危険性は少なくなる。

すべては最適のために。
SofTek
Security Leading Company

必要なフェーズでの脆弱性検査

- Webサイトに対して必要なタイミングで実施
  - 開発フェーズでの検査
  - 納品フェーズでの検査
  - 運用フェーズでの検査

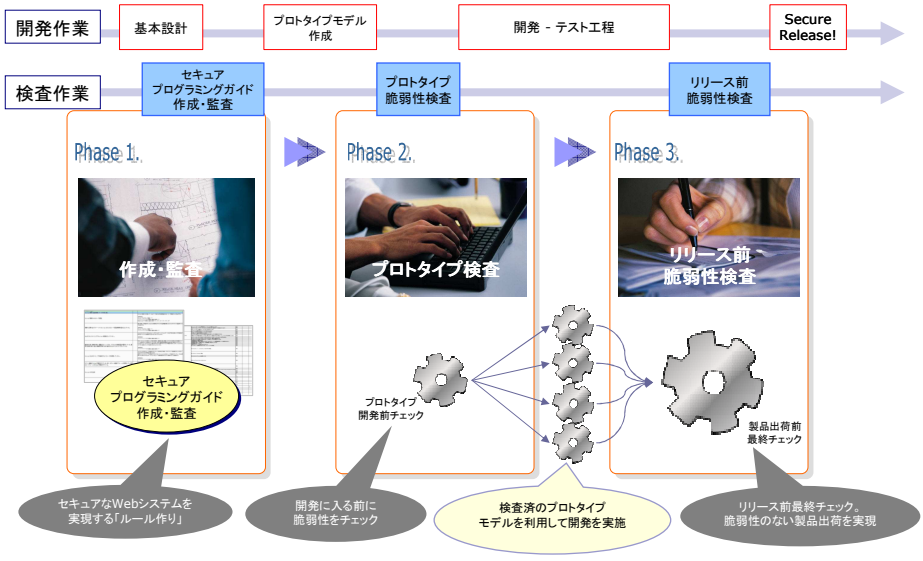
SofTek
Security Leading Company

必要なフェーズでの脆弱性検査

- 開発フェーズ
  - 納品前の最終チェック
    - 仕様に対する整合性
    - 最終的なセキュリティレベルの確認
  - 開発当初からの検査実施
    - プロトタイピングの作成段階で脆弱性を潰してしまう
    - 結果的に開発コストの削減

SofTek
Security Leading Company

必要なフェーズでの脆弱性検査



必要なフェーズでの脆弱性検査

- 納品フェーズ
  - 発注仕様に合わせたセキュリティチェックは必須
    - 脆弱性検査ツールを使ったセルフチェックは有効

市販ツール

Rational AppScan	(IBM)
HP WebInspect	(HP)
Web Vulnerability Scanner	(Acunetix)
WebProbe（セッション管理脆弱性専用）	(ソフテック)

HTTPキャプチャツール(フリー)

プロキシ：
Paros、Odysseus、Proxomitron、BurpProxy など

ブラウザプラグイン：
firefox: LiveHTTPheaders, TamperData など
IE : ieHTTPHeaders, Web Development Helper など

必要なフェーズでの脆弱性検査

- 納品フェーズ
  - 発注仕様に合わせたセキュリティチェックは必須
    - 検査品質を重視するなら、スキルフルな検査が必要
      - ツールのカスタマイズで対応範囲は広がるが、検査ツールはあくまで「道具」。持っている検査パターンに脆弱性がマッチしなければ、誤検出、未検出の場合も多い。
      - 特にセッション管理については、ツールだけでは検査が難しい。

予算的に可能であれば、第三者による専門的な検査を受けることも視野に入れるべき

----> ここがリリース前の最終防衛線

すべては最適のために。
SofTek
Security Leading Company

必要なフェーズでの脆弱性検査

- 運用フェーズ
  - 定常的な脆弱性検査が必要
    - 機能追加のタイミングによる脆弱性の発生
    - プラットフォームの脆弱性公開による新たなリスクの発生
    - 新たな攻撃手法の出現

すべては最適のために。
SofTek
Security Leading Company

Section. 4

まとめ

すべては最適のために。
SofTek
Security Leading Company

まとめ

- 大学・研究所サイトにおいて、
  - 「守るべき情報」がなくても、踏み台等の攻撃に備え、**Webアプリケーションセキュリティに対するリスク管理を適切かつ継続的に行う**ことが、社会から求められている。
  - セキュアなWebアプリケーションの開発・運用の実現には、
    - 適切な仕様書作成
    - セキュアプログラミング
    - 必要なフェーズでの脆弱性検査

    を**自分のWebサイト規模に合わせた形で実施**することが重要。

すべては最適のために。
SofTek
Security Leading Company

ご静聴ありがとうございました