筑波学院大学紀要第13集　115 ～ 126 ページ　　2018年

マルチスクリーン環境に対応したセキュリティ機能を実装したクラウド型 Web サイトの開発

篠崎　健一*

Development of the Cloud-style Website Equipped with Security Functions Responding to the Multiscreen Environment

Kenichi Shinozaki *

Abstract

Recently, school websites are regarded as an effective communication tool that contributes to "making a school open to the society" by conveying everyday educational activities.

On the other hand, they have pointed out the operation problems including technical issues related to updating the website, technical problems regarding design, and the security measures.

Under these circumstances, we have developed a cloud-style website accessible from anywhere and anytime, which is also equipped with security functions and the multiscreen environment.

Key words: school promotion, school homepage, transmission of information, information education, security

1．はじめに

昨今、学校のホームページ（以下、学校サイトと記載する。）は、日常の学校教育活動を伝える有効な情報伝達ツールと位置付けられており、「開かれた学校づくり」の一役を果たしている[1, 2)]。

しかし、学校サイト運用の問題点とされている要因に、「サイトのデザイン等の技術的な問題」、「サイト更新に係る技術的課題」、「セキュリティ対策」等があげられている[3, 4)]。「サイトのデザイン等の技術的な問題」として、Google 社マルチスクリーンワールド調査によれば、日常生活で接触するメディアの91％がスクリーンメディアであり、現代は、一人の利用者が複数の端末を使うマルチスクリーン社会であると述べている[5)]。

さらに、ネット閲覧者のオンライン行動の起点の72％がスマホであり、その67％が PC でも同 URL を閲覧していると分析している。また、総務省が実施したスマホ利用率調査では子供はスマホ、40代以上の者は PC で閲覧

＊　非常勤講師、Tsukuba Gakuin University

する傾向にあると報告しており、マルチスクリーン社会への迅速な対応を求めている。

これらより、2014年４月１日～８月20日の期間、筆者の本務先である水戸工業高等学校の情報技術科のサイト（以下、本科サイトと記載する。）へのアクセス情報を分析した。

その結果、本科サイトも様々な画面サイズのデバイスからアクセスされていることが分かった。

次に、「サイト更新に係る技術的課題」である。通常、県立高校にはホームページを管理する技術者は居ない。学校内に誰も学校サイトを作成・更新できる者がおらず、教育委員会から要求される学校情報公開指標などにあわせて、管理職に指導を受け、担当者が更新をしているケースが一般的であり、年に数回の更新にとどまる学校が多く、新規性の高い情報を、公開できていないのが実情である[6, 7]。

そして最後に、「セキュリティ対策」の問題である。近年、Web サイトの脆弱性を狙った攻撃が多発している。IPA の報告によれば、Web サイトを標的とした攻撃は増加傾向にあり、NRI セキュアテクノロジーズの報告では、現在公開されている Web 等の33％には危険な脆弱性が存在すると報告している。

そこで、何時でも何処でも、十分なセキュリティ対策を施し、なお且つマルチスクリーン環境にも対応した県立高校が運営するクラウド型の Web サイトを研究開発する必要であると考えた。

以上のような、学校サイト運用の３つの問題点に対処した、県立高校が運営する Web サイトの技術論文は皆無である。

システム開発ツールに関しては、HTML や JavaScript、PHP と MySQL を利用し、Web アプリケーションという形態で構築した。本科サイト完成後、危険な脆弱性を排除すべくセキュリティ対策と脆弱性検査ツールを利用した検証も行った。

２．本科サイトの開発

本科サイトの URL を次に記載する。

http://mito-th.sakura.ne.jp/homepage1/info/

２．１　開発環境

通常、県内の高校のサイトは、茨城県教育研修センターが管理するサーバ上に構築されている。しかし、この茨城県内の学校サイトが動作しているサーバでは、PHP 等のサーバサイド言語が使用不可能であるため、学校全体のサイトは県が管理するサーバに配置し、本科サイトは、各開発環境の構築が可能な、さくらインターネットのレンタルサーバ環境での開発・設置し、リンクすることにした（表１）。

動作確認を行う Web ブラウザとして Google Chrome、FTP クライアントソフトとして Cyberduck や FFFTP、エディタとして Sublime Text や Notepad++ を使用した。その他の使用ソフトについては後述する。

２．２　レスポンシブデザインの適用

マルチデバイス対応に改善する代表的な手法に、レスポンシブデザインがある（図１）。

レスポンシブデザインとは、表示された機器の種類やサイズに応じて表示内容が最適な状態に変化するよう設計する開発技法である。

表１　開発環境

ソフトウェア	サーバ環境
OS:Mac OS X Yosemite OS: Windows 8.1 Google Chrome 39.0.x Cyberduck4.6.1 Sublime Text3 Notepad++ FFFTP PictBear OWASP ZAP 2.3.1 XAMPP 5.6.3	OS:FreeBSD9.1 CPU:Intel Xeon E312xx メモリ容量 :18GB Apache 2.2.25 MySQL 5.5 PHP 5.4.35

2．3　ページデザインに関する使用技術

2．3．1　Google API

本科サイトでは、多くのGoogle APIを利用している。

提供しているAPIの種類は幅広く、Googleで一覧表も提供している。

進路情報・各科教育課程ページでは、グラフ作成サービスであるGoogle Chart ToolsAPIを使用した。これで、教育課程ページの単位数に応じた棒グラフ、進路ページでは円グラフとした（図2）。

2．3．2　fancybox

本科サイト内で画像を表示させる際、fancybox v1.3.4を使用した。Fancyboxは動作時に必要なJavaScriptやclose画像は用意されているので、HTMLで表示させたい画像を指定するだけで、fancyboxの動作を実現できる。fancyboxが立ち上がり、画像を表示させるエフェクト（効果）は、JavaScriptによって制御されている。これはあらかじめ組まれているので、加筆・変更は必要ない。

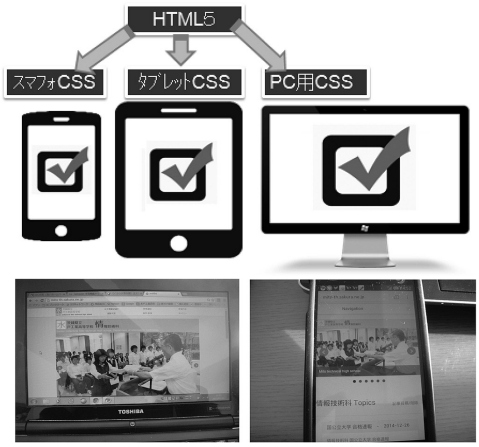

図1　レスポンシブデザイン

図2　進路情報ページ

2．3．3　bxSlider

本科サイトではスライドショー風に画像を配置している。画像をスライドショー風に表示させる際に使用しているのがbxSliderというスライダープラグインである。このbxSliderは、jqueryと一緒にを読み込ませればulでもdivでもスライダーが使える上、スライド動作も3種類選べる。fancybox同様HTMLで配置したい画像を記述する。また、設置も簡単である。

2．3．4　Google Analytics サーバ

本科サイトは、Google Analyticsを適用している。Google Analyticsは Google 社が提供する無償の Web アクセス解析ツールである。このツールを使用してアクセス解析を行うことで Web サイトの状態がわかる。得られるデータとしては、サイト訪問者の訪問日時や、曜日別のアクセス数の推移、さらにどの検索エンジンでどのようなキーワードで訪れたかなどがある。これらを集計し、分析することで Web サイトの現状が判明する。

利用方法はユーザ登録をした際に自動生成される解析用の JavaScript コードを、アクセス履歴を取得したいページに埋め込んでいる。Google Analytics では cookie と JavaScript コードを使用して、サイトを訪問したユーザ情報を収集し、データをトラッキング（追跡）する。

定期的にデータをダウンロードし、サイト分析等に利用している。

2．4　ページデザインに関する工夫点

2．4．1　本科サイトトップページ

トップページの画像を図3に記載する。

（A）本科ロゴ

本科ロゴの周辺は科色（黄色）とした（図

3（A））。

（B）デザイン統一

本科の特色を視覚的に伝える画像を採用したスライドショーは、トップページと同様にbxSlideを使用した（図3（B））。

また、ページ下部のリンクボタンもリンク先の情報を視覚的に伝える画像を採用した。これらは、トップページと同様の配置にすることでサイト全体のデザインの統一を行った。

（C）本科紹介ページ

本科のページ上部･下部（（図3（C））には、教育課程･学習内容･進路情報･雑学散策（生徒作品）にページ分けし掲載した。

（C-1）教育課程ページ

学年ごとに単位数をグラフ表示している（図4）。Google APIを使用しているため、単位数の変更時はデータ内の数字を変えるだけで更新される。

（C-2）学習内容ページ

本科での学習内容をまとめている。reavel.jsを使用してパワーポイント風に学習内容をまとめた（図5）。情報端末の環境によって、上手く動作しない場合もHTML5環境で閲覧できる（図6）。

さらに、本科について説明する動画も掲載した。動画の下部には、文字と画像でも閲覧できるテキストも用意した（図7）。これによりどのような環境・機器でも動画の内容を閲覧できる（図8）。

（C-3）進路状況ページ

合格者体験記では、企業や大学の合格者のコメントを掲載し（図9）、卒業生よりでは

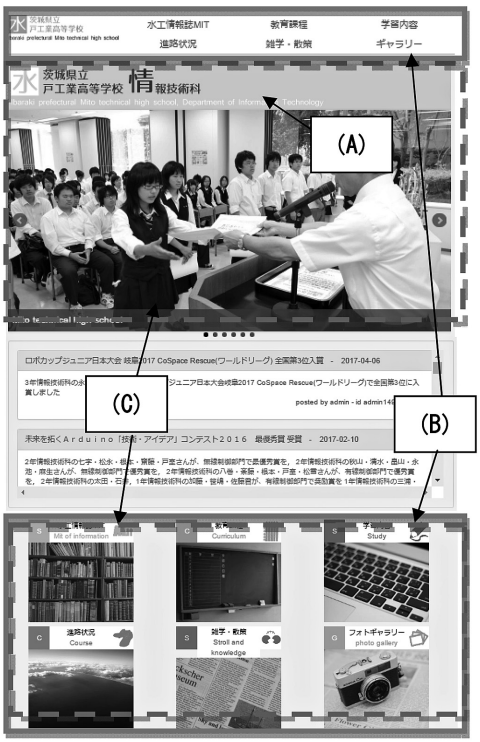

図3　本科サイトトップページ

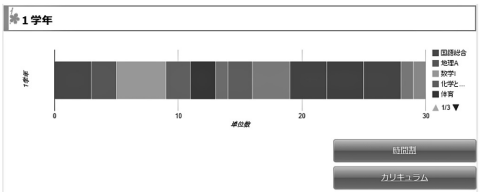

図4　教育課程ページ

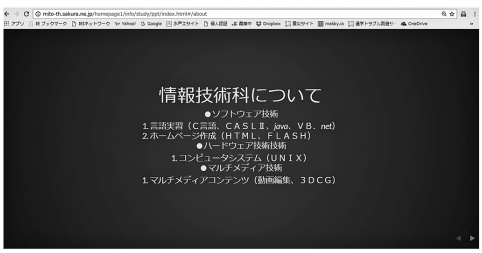

図5　reavel.js 版

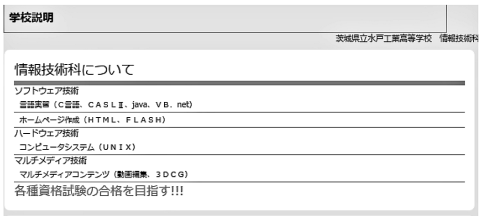

図6　HTML5 版

今後どのような目標を立てているのかなどを掲載した。(図10)

(C-4) 雑学散策ページ

生徒が作成した様々な作品を紹介するページを雑学散策として作成した。掲載した作品は、実際に操作できるソフトウェア系はURLをロボットなどのハードウェアは動画を使い掲載した。

(C-5) 時間割表

時間割表はPDFファイルを読み込む形式とした（図11)。

図７　学習内容ページの動画

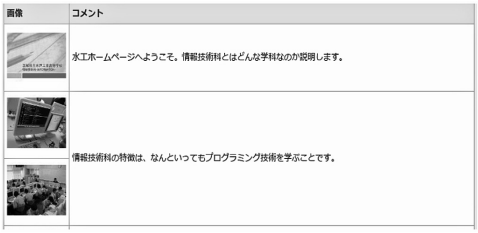

図８　動画テキスト

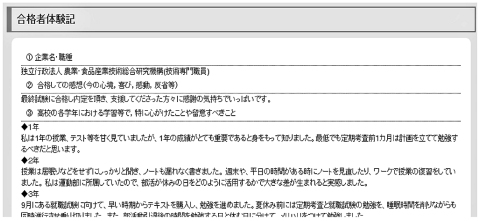

図９　合格者体験記

２．５　セキュリティ対策に関する使用技術

２．５．１　ログイン機能

サイト管理者が、各データを変更する場合、本科サイトにアクセスすると、Webサーバ上でログイン処理等の要求に応じて、PHPのプログラムが動作し、各データの変更処理が可能となる（図12)。

本機能は、入力データに対応したSQLステートメントを生成し、データベースへのデータ登録や検索を行う。それぞれの機能は、投稿・削除を行う前にログインを行う仕様になっている。図13のようなフォームから入力された閲覧者IDとパスワードが一致した場合成功するものとする。この際、

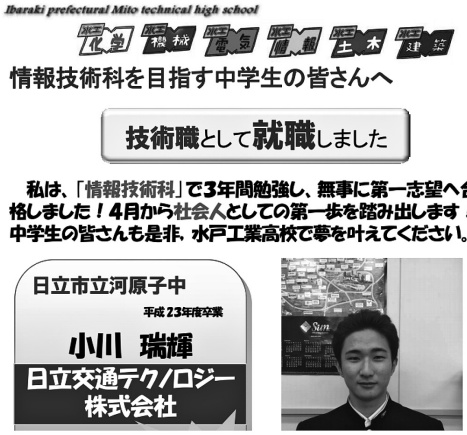

図 10　卒業生より

図 11　時間割

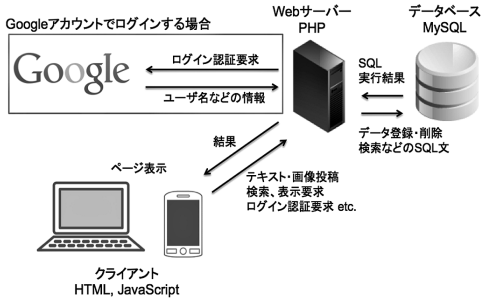

図 12　システム概要

図 13　ログインフォーム

Google アカウントでログインを行うこともできる。

このような流れで処理を行う機能として、トピックスシステム、MIT 閲覧システム、フォトギャラリーを本科サイトに実装した。

2．5．2　トピックスシステム

トピックスシステム（図14）は、様々な大会での成績等の各科のトピックスを Web から更新可能にしたものであり、画像をアップロードすることで、画像を添付したトピックスを更新することも可能である。

図15のように、画像が添付されたトピックスの場合、トピックス本文をクリックすることで、画像がポップアップ表示される。

トピックスの更新は、図16のフォームからタイトル、日付、コメント（本文）を入力することで行う。この際、“ファイルを選択”ボタンから画像を選択し添付することができる。“投稿する”ボタンがクリックされると、データの登録を行う PHP プログラムにフォームの内容が送られ、それぞれの内容をデータベースに記録する。

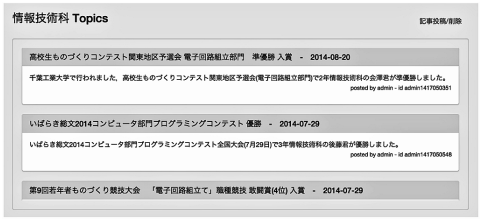

図 14　情報技術科トピックス

図 15　画像が添付されたトピックス

図 16　トピックス投稿フォーム

画像が添付されていた場合はサーバ上にコピー、リサイズし、そのファイルの相対パスを保存する。トピックスを表示する際は、PHP からデータベースにアクセスしこれらのデータを読み出し、HTML に整形し表示を行っている。これにより、従来のトピックス更新のように HTML ファイルを作成してから FTP でアップロードするという手間を減らすことができる。

トピックスの削除は、投稿フォーム右上のボタンから削除フォームに移動し、トピックスの ID を入力することで行う。ID は投稿時

閲覧者名と現在時刻を元に生成され、各トピックスの右下に表示される。

2．5．3　MIT 閲覧システム

これは、図17のように記事を検索し、また、視覚的に内容を把握できるようサムネイル画像を検索結果に表示した。さらに、cookie を利用し、最近の検索ワードを保存することで、閲覧システムトップページにおすすめの記事を表示する、月別表示も可能にする、PDF 版の記事へのリンク配置など利便性の向上を計った。

MIT の投稿は、図18のフォームから行う。日付、記事 No、記事タイトル、記事本文を入力し、PDF ファイルをアップロードする。本文中に正式な科名が入っていない場合には、タグに正式な科名を入力することで、正しく検索を行えるようにする。PDF ファイルは、画像にも変換され、PDF ファイルと JPG ファイルをそれぞれ別のディレクトリに保存される。データベースには、これらの情報が保存され、検索や PDF へのリンク時に PHP でデータベースにアクセスし利用する。

記事の削除は、投稿フォーム右上のボタンから削除フォームに移動し、記事 No を指定すると削除できる。このため、投稿時記事 No が衝突しないようチェックを行い、過去の投稿と同じ記事 No を指定すると警告を行うようにした。

2．5．4　フォトギャラリーシステム

これは、図19のように生徒の活動の様子を画像で閲覧・投稿できるシステムである。画像をクリックすると、図20のように撮影場所とコメントも確認できる。

画像の投稿は、図21のフォームから行う。

画像投稿フォームでは、画像ファイル、日時、場所、コメントを入力・登録する。この際、現在地を端末の位置情報から取得している。端末の位置情報から緯度経度を取得し、Google Maps API での検索を行うことで、現在地名を取得し、自動入力する。また、ス

図 17　MIT 閲覧システム

図 18　MIT 投稿フォーム

図 19　フォトギャラリーシステム

マートフォンで撮影した画像をアップロードすることが多くなると想定されるため、スマートフォンを横向きにして撮影した画像でも、正しい向きに表示されるようにした。画像の Exif 情報を PHP で読み込むことで、撮影時のスマートフォンの向きが数値で取得できる。その向きに応じて画像を反転させている。（図22）さらに、大きな画像はサイズを縮小してコピーしている。

図22に関して

・$filepath は保存した画像の相対パス

・Orientation は端末の向き
　6 は端末が右向きの場合

・imagerotate（）で画像を回転し上書き保存

図 20　コメントの確認

図 21　画像投稿フォーム

```
//画像の向きを修正
$exif_datas = exif_read_data($filepath);
if(isset($exif_datas['Orientation']) and $exif_datas['Orientation'] == 6){
$source = imagecreatefromjpeg($filepath);
$rotate = imagerotate($source, -90, 0);
imagejpeg($rotate, $filepath, 100);
}else if(isset($exif_datas['Orientation']) and $exif_datas['Orientation'] == 8){
$source = imagecreatefromjpeg($filepath);
$rotate = imagerotate($source, 90, 0);
imagejpeg($rotate, $filepath, 100);
}else if(isset($exif_datas['Orientation']) and $exif_datas['Orientation'] == 3){
$source = imagecreatefromjpeg($filepath);
$rotate = imagerotate($source, -180, 0);
imagejpeg($rotate, $filepath, 100);
}
```

図 22　画像の向きの修正を行うコード

画像の削除は投稿フォーム右上のボタンより削除フォームへ移動して行う。ファイル名を入力することで削除が可能である。

3．セキュリティ上の脅威と対策

本科サイトのセキュリティ上の問題点を検証するため、次に記載する検査を行った。

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は、閲覧者の入力をページの出力結果に反映するような形式の Web ページ（検索、コメント等）でクロスサイトスクリプティングに対する脆弱性が存在する場合に、悪意のあるスクリプトを実行されてしまう攻撃のことである[3)]。閲覧者の入力に対して、エスケープ処理を行わず直接出力に反映している場合、クロスサイトスクリプティングの脆弱性がある状態といえる。

図23のように JavaScript をフォームから入力すると、画面上にアラートを表示できる。このように任意のスクリプトが実行可能な状態であると JavaScript を利用してページの内容が改ざん等されるおそれがある。

対策として、入力がスクリプトとして解釈されることを避けるため、要素を出力する際にエスケープ処理する必要がある。図24の JavaScript や HTML で使用する記号を実体参

図 23　XSS 実行例

照に置き換える。

htmlspecialchars 関数を利用して記号の置き換えた。

3．2　SQL インジェクション

SQL インジェクションは、検索等の処理を行う際に行う SQL 問い合わせにおいて、閲覧者からの入力を反映する場合、適切な処理を行っていない場合、悪意のあるデータベース操作をされてしまう攻撃のことである[3)]。閲覧者の入力に対して、適切な処理を行わずに SQL の生成を行うと、悪意ある入力によって任意の SQL を実行され、不正ログインや情報を窃取されるといったような攻撃をされるおそれがある。対策として、PHP と MySQL を使用した環境で、SQL で使用される特殊記号のエスケープ処理を行うことのできる関数 mysql_real_escape_string を利用したエスケープ処理に加え、3.3と3.4の対策を行った。

3．3　Google アカウントを利用した認証

ある Web サイトにおいて、通常のログイン処理を行う場合、データベースに閲覧者 ID とパスワードの組み合わせを記したテーブルが存在することになる。

このテーブルに対して攻撃が行われ情報が流出した場合、当該のサイトに不正ログインが可能になるだけでなく、ID とパスワードの組を利用して、他の複数の Web サイトに対してログインを試みるリスト型攻撃を行われる可能性がある。

そこで、データベースに重要な情報を持たないような工夫として Google アカウントを利用したログイン認証を実装した。

Google は、OAuth を利用して外部 Web サイトから閲覧者の情報を読み取ることを許可している。OAuth は、閲覧者の同意のもとに Web サイト間でそれぞれの閲覧者情報をやりとりする技術である[7)]。同意の際には、Google 側でログイン処理を行う必要がある。これを利用することで、Google から閲覧者のメールアドレスの情報を受け取る。データベースには、閲覧者のメールアドレスのみを記録しておき、それをホワイトリストとして Google と受け取った情報と照合することでログインを行うことができる（図25）。

3．4　クロスサイトリクエストフォージェリ

クロスサイトリクエストフォージェリは、ログインした閲覧者が処理を行う際に正規のページからリクエストが行われているか適切なチェックを行っていない場合、意図しない処理を攻撃者に行われる攻撃である。攻撃者は、ターゲットサイトの処理ページへ悪意あるリクエストを送信する Web ページを作成し、ターゲット閲覧者にアクセスさせる。ログイン状態にある閲覧者がこのページにアクセスした場合、ログイン状態を利用され、意図しない書き込みや投稿を行ってしまう。対策として、処理のリクエストが正規のページから行われたものであるのかを、ページ間で

&	→	&amp;
<	→	&lt;
>	→	&gt;
“	→	&quot;
‘	→	&#039;

図 24　実体参照への変換

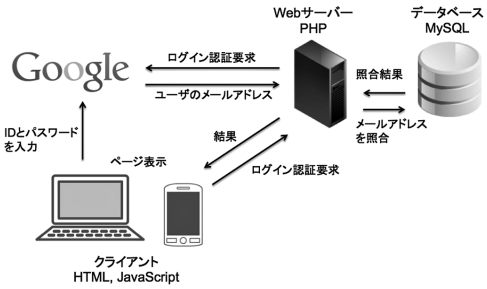

図 25　Google アカウントでのログイン

```
//処理入力時、トークンを生成しセッション変数に保存
//また、POSTで処理を受け付ける画面へ送信
$_SESSION['token'] = sha1(uniqid(mt_rand(), true));

//処理受付時、セッション変数のトークンとPOSTされたトークンを比較
if(!isset($_SESSION['token']) or $_SESSION['token'] != $_POST['token']){
      echo "トークンの不一致、不正なアクセスです。";
      exit;
}
```

図 26　アクセスの正当性の検証

攻撃者に予測不能なランダムな情報をやりとりすることでチェックする。図26はコードの一部である。処理入力画面でサーバのセッション変数にランダムな情報を持たせる。ランダムな情報は、mt_rand 関数で生成した乱数をプリフィックスとして uniqid 関数で生成した ID を sha1関数でハッシュ化したものを使用している。そして、処理の確定画面でセッション変数の情報とフォームから送信された情報が一致するか確認する。不一致の場合は不正なリクエストとして拒否する。

4．セキュリティ対策の効果の検証

セキュリティ対策の効果を測るため、以下の様な方法で脆弱性検査を行った。

4．1　OWASP ZAP を利用した脆弱性検査

OWASP ZAP は、Open Web Application Security Project が開発する脆弱性検査ツールである。マルチプラットフォーム、多言語対応で比較的扱いやすく、IPA が推奨するツールであるため OWASP ZAP を使用した検査を行うことにした。使用方法は、図27のように、起動後 URL を指定するだけだが、レンタルサーバ上のサイトを直接検査することは、攻撃とみなされるおそれがある。また、SQL インジェクションの検査でデータベース構造が破壊されるおそれもあるため、XAMPP を利用したローカル環境で検査を行った。

XAMPP は、Apache、MySQL、PHP 等 Web アプリケーションの実行に必要なソフトウェアを一括インストール・管理できるツールであり、Apache Friends が開発している。これらをインストールし、ファイルやデータベース構造をコピーし、ローカル環境で作成した Web アプリケーションを実行できるようにした。

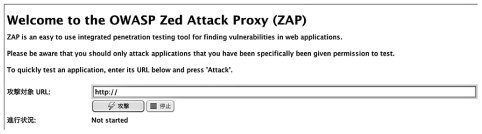

図 27　攻撃対象の指定画面

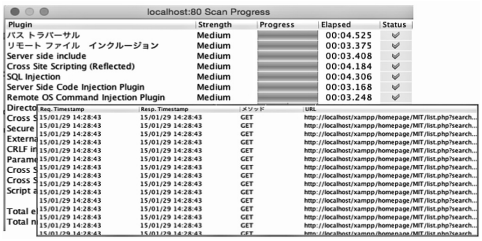

図 28　検査コードの送信と検査進捗

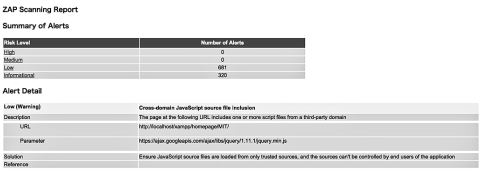

図 29　検査レポート

OWASP ZAP の攻撃対象に、localhost を指定することで検査を行った。OWASP ZAP は検査用コードを攻撃対象の各フォーム等に入力し、そのレスポンスを解析することで、脆弱性の有無を判定する（図28）。

検査レポートを元に、エスケープ処理の実装漏れ等を修正していくことで最終的に危険な脆弱性は検出されなくなった（図29）。

4．2　CSRF 脆弱性の検証

OWASP ZAP の検査では、クロスサイトリクエストフォージェリの脆弱性は検出されな

図 30 外部アクセスの拒否

かったが、実際にクロスサイトリクエストフォージェリを行った際に、想定した結果が出力されるかを確認するため、外部サイトに攻撃用ページを作成し、検証を行った。

正規の投稿フォームに偽装したページを作成し、リクエスト先に、正規サイトの処理確定ページを指定した。結果、図30のように、不正なアクセスと判断し、表示を行った。

5．授業実践

本科の３年生38人を対象とし、科目「実習」の中で､「ネットワークシステムのセキュリティ対策」と「レシポンシブデザインの基礎」という題材で、本システムを実際に操作して、サーバサイド技術における Web システムに関する授業実践を行った。そしてその授業に関する結果や感想を報告書に記述させた。

5．1　感想からの評価

生徒が提出した報告書の感想に関する記述から①理解、②疑問点にあたる部分を抽出し、それらを KJ 法で分類した。

まず、理解（表２）に関して「セキュリティ対策の重要性の理解できた」と分類できる文章が32 人であった。その他､「脆弱性検査のやり方など理解できた」が26人と、セキュリティ対策について理解を深めた生徒が多かった。

次に、疑問点（表３）では､「セキュリティ対策アルゴリズムにはどの様なものがあるのだろうか。」､「GoogleAPI はどの様な仕組みなのだろうか。」等という発展的な内容を記載した生徒も多く、学習の動機付けがある程度できたと思われる。

表２　理解できたこと（感想から抽出）

内容	人数
セキュリティ対策の重要性の理解できた。	32
脆弱性検査のやり方など理解できた	26
レシポンシブデザインの仕組みを理解できた	22
サーバサイド技術の意味が理解できた	20
セキュリティ技術の基礎について理解できた	19
プログラムの仕組みが理解できた	19

表３　疑問点（感想から抽出）

内容	人数
セキュリティ対策アルゴリズムにはどの様なものがあるのだろうか。	15
GoogleAPI はどの様な仕組みなのだろうか	14
脆弱性検査だけで完璧なのだろうか	4

その他「脆弱性検査だけで完璧なのだろうか」と回答した生徒も４人おり、セキュリティに関する意識も高揚したと思われる。

6．まとめ

「デザイン等の技術的な問題」の対策として、レスポンシブデザインを適用した。

HTML や CSS の記述が正しいかを W3C 評価を行って確認したところ、問題が無かった。デザイン的にも､「2．本科サイトの開発」の「2.3　ページデザインに関する使用技術」,「2.4　ページデザインに関する工夫点」の機能を実装した。

「セキュリティ対策」として､「4．効果の検証」より、当面は脆弱性等、セキュリティに関する問題点が無いことが証明された。しかし、当面は安全な Web サイトを作成できたと考えるが、サイバー攻撃は日々高度化しており、利用者と開発者の双方が情報セキュリティについての意識を高め、知識を深めていく必要がある。

「サイト更新に係る技術的課題」として、学校特有の課題がある。

殆どの都道府県では、学校サイトの運営は

県教育委員会が統一的に行っており、サイト開発において多くの制限が存在する。その中で、一部の教職員が対応している。しかし、その他の業務（生徒・保護者の対応等）が、多忙で修正や更新を行う時間的余裕がないという点では、何時でも何処でも、学校サイトにアクセス・ログイン・書き込みができるクラウド型ということで、担当者の負担軽減に繋がるという意見を多数頂いた。

これらの実情より、マルチスクリーン環境に対応した、セキュリティ機能を実装したクラウド型の学校サイトに関する技術論文は、現時点では学会報告等は皆無の状態である。

さらに、実業系の高校などでは、本サイトは、情報系の教材としても活用できるため、産業教育や情報教育の中で極めて意義深い研究であると考える[7)]。

7. 参考文献・資料

1）文部科学省 地域に開かれた安全・安心な学校づくりガイドブック
http://www.mext.go.jp/a_menu/shotou/zyosei/syuppan/03103101.htm
2）町田智雄他：組織的・継続的な学校ホームページ運用のための体制構築，日本教育工学研究報告会 JSET08-5 pp.155-160. pp149-154.
3）豊福晋平他：学校広報を前提としたウェブサイト導入プログラム，日本教育工学研究報告会 JSET08-4 pp.155-160.
4）豊福晋平他：学校サイトに適したハイブリッド型 CMS の開発，日本教育工学研究報告会 JSET08-1 pp.227-232.
5）マルチスクリーン ワールド調査データサイト https://www.ja.advertisercommunity.com/t5/Google-AdWords/ct-p/Google_AdWords
6）鷲尾健仁：教職員・児童・保護者による全校的学校ホームページ運用体制の構築，日本教育工学研究報告会 JSET09-1 pp.199-206.
7）篠崎健一：工業高校におけるサーバサイド Java による制御学習 Web システムの開発，日本産業技術教育学会論文誌 第48巻第3号 2006，p173-181