James Jeffrey BURT
Philadelphia, PA

Preferred contact: jobnotice@jamesburt.com
Alternate contact : 267-560-3899
Website: www.jamesburt.com
Updated: 2024-12-14

ADDENDUM

Initiatives

I led the initiative to harden our Apache server, mostly through tweaks to the httpd.conf/ssl.conf, as well as performing regular testing of the encryption level and HTTP headers.

I led the preparation for, and hosted the inspections by the ICSA TrueSecure inspection at the bank for two years. This inspection considers technical, system, practices, and personnel vulnerabilities, and we passed both times. Our website was ICSA certified..

Products

Some Products I have built:

A companion application to an online MMORPG. It leverages TamperMonkey scripts, Cross-Domain Requests, PHP Rest APIs, Lightweight framework, NOSQL, application cache (browser storage), a graphing javascript framework, a tables javascript framework,

A Cordova/Phonegap android applicattion (and companion single-page web app) that allows the user to make anonymous SMS messages. The application is built on w3c.css for its responsive features, and makes a restful CORS connection to an API which in turn interchanges the SMS messages with a gateway to the destination telephone.

Two responsive mobile-ready websites for local small customers in NYC. One, a social media site gave me an opportunity to devise an interesting solution to calculating radial distance between two users, so that match criterion can include distance. I was able to leverage some data that is available from the US Postal service in these calculations. (Demo-1)

A PHP web mash-up of financial information. The user can select a series of equities/indices and display their current data (price, last trade etc) and graph their historical closing prices for a period of one week to ten years. The financial instruments that the user selected would be retained as his personal choice. The data was retrieved through a Yahoo-Pipes / Yahoo-Tables mashup that I designed to extract the data from a series of webservice calls. This data was returned to our application as XML, parsed, and added to memcache. It was then served to the user via JSON and displayed by the Highcharts JS library. (Demo-3)

A PHP interface to screen file uploads which a user submitted, scanning them for malicious content. This used the ESETS library. I built a custom wrapper to invoke the library prior to accepting the upload.

An extension to the PHP framework Codeignitor which adds an additional storage medium to its cache library.

An extension (hook) for the PHP framework Codeignitor, which performs XSRF/CSRF protection. The framework now automatically provides XSRF/CSRF protection.

A J2EE application for internal fund managers which allowed them to enter mutual fund trades real-time into the processing network. It was a J2EE application built upon a Struts framework. It used Java APIs into a mainframe computer, Weblogic App Server, Netscape Webserver, and Oracle database.

A J2EE application for clients (i.e.: BearStearns, Vanguard), to allow users to navigate from the client website to the CheckFree website without separate log-in (Single Signin pre-Oauth). It performed triple-DES encoding of the user information forwards him to CheckFree without the requirement to log-in again. Included a series of web-based tools to synchronize encryption key updates. All stored information (3DES keys, userId, userPw etc) was encrypted with a rotating-modulus encryption bean to protect it from network snooping and database views. I was the architect and programmer of this product.

An HTML-Application designed to capture results of a search query to a web page, and save the results to an excel spreadsheet, using Windows Scripting Host. I was the architect and programmer of this product.

A J2EE MVC-Compliant Password Reset application that allowed users to reset their passwords without help-desk intervention. It used the familiar "secret phrase" technique, and emailed a new password to the user's email address. I was the architect and programmer of this product.

An MVC-Compliant auto-vetting PKI Certificate enablement product which used mySQL and Tomcat. It stored requests for client-certificates in a mySQL database, and upon approval, sent an email to the certificate recipient directing him to a Novell E-Security product where he received his client certificate. It managed user information through an administrative interface, and used LDAP to determine user privileges. I was the architect and programmer of this product.

A marketing website belonging to a financial corporation. This was a largely static website, which I designed to meet the needs of the marketing department. It had server-side includes for statistics, and CGI processing of form input. I managed this product from 1998-2001.

The Customer Service Console web applications for BlackRock Mutual Funds, and the Investment Accounting System Error Correction Interface. These were part of a series of products moving legacy applications from 3270 screens onto web applications built on Bluestone Sapphire Web 5.0, and OpenConnect Java Screen Scraper.

A set of transaction processing web applications similar that used a TN-3270 gateway, screen scraper, and Java web application server. I was the architect and programmer of this product.

A series of transaction processing web applications for mutual fund companies. This was the www.funds-info.com website for Harris, Numeric, Blackrock, and Citizens Funds (Sunsetted April 2002, I retain a mock-up). The thin client (web browser) used extensive object-oriented JavaScript and Dynamic HTML to prefetch all account information, and pre-process transactions. The front-end was a Netscape webserver running on SunOS, custom CGI's written in PERL, server-side JavaScript, and server-parsed HTML (server-side includes). The middleware was an OS/2 box running an application server programmed in Rexx. The back-end legacy systems stored all data, and processed transactions.