# Stephen Patrick Enright ·

3rd

Senior Software Engineer at DiManEx

Ireland · Contact info

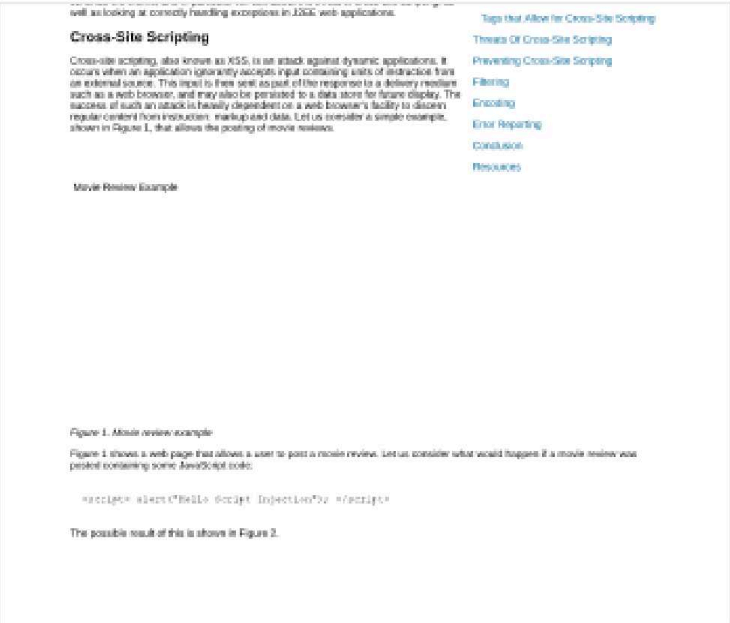
229 connections

Message More

**DiManEx**

**London University**

## Featured



well as looking at correctly handling exceptions in J2EE web applications.

### Cross-Site Scripting

Cross-site scripting, also known as XSS, is an attack against dynamic applications. It occurs when an application ignorantly accepts input containing units of instruction from an external source. This input is then sent as part of the response to a delivery medium such as a web browser, and may also be persisted to a data store for future display. The success of such an attack is heavily dependent on a web browser's facility to discern regular content from instruction: markup and data. Let us consider a simple example, shown in Figure 1, that allows the posting of movie reviews.

Tags that Allow for Cross-Site Scripting

Threats Of Cross-Site Scripting

Preventing Cross-Site Scripting

Filtering

Encoding

Error Reporting

Conclusion

Resources

Movie Review Example

*Figure 1. Movie review example*

Figure 1 shows a web page that allows a user to post a movie review. Let us consider what would happen if a movie review was posted containing some JavaScript code:

```
<script> alert("Hello Script Injection"); </script>
```

The possible result of this is shown in Figure 2.

**Handling_Java_Web_Application_Input_Part2.pdf**

attackers have to spend vast amounts of time researching way
security infrastructure of an application. An attacker can use t
scan for vulnerable websites. Using these findings, an attacke
to ghost straight through firewall rule sets on port 80, altering
application. This is true never more so than today. There are
and frameworks available. Engineers are under increasing pr
time, and hence place a heavy reliance on such tools. Howeve
not adequately deal with user input to meet all cases, and as
unintentional security vulnerabilities. Therefore, it is of param
secure coding practices are in place to close any possible do
nefarious attacks to take place.

The purpose of this series of articles is to explain common se
emphasizes the importance of handling application input corre
ensuring the security of an application. This series is aimed at
maintaining software systems that are unaware of such issue
best practices, along with SQL injection attacks. In later article
deal with cross-site scripting attacks and error-handling techni

### The Importance Of Server-Side Validatio

The most common web application exploits are the result of n
validating user input: client-side validation and server-side va
experience by improving the responsiveness and usability of
interfaces through a combination of JavaScript and HTML ele
is less sophisticated than server-side validation and, if not us
serious vulnerabilities into an application.

For example, consider an online e-commerce application that
through the standard process of checking out: entering payme
validation is performed, and the state at each step is stored in
confirm the transaction. On confirmation, the order is wrapped
On receipt, the server performs no validation, but simply acce

Now, if you haven't already spotted the vulnerability, the threa
attacker can view the HTML source, change the price stored
client-side checks by disabling scripting through the browser
page locally. The attacker can then load the newly crafted che
checkout process, by submitting the order to the server for pro

Although the above example is very simplistic, you are proba
similar to manifest itself. The point here is that data input into
a user inputs, whether well-meaning or otherwise. Therefore,
user, which is a sure recipe for disaster. For example, conside
page formatting and possibly deface a web page by passing i

As a result, never gamble: always identify where data flows fr
findings reveal input being used to generate content, carefully
stringent and prudent as possible, by employing server-side

### Validation Best Practices

A critical validation practice is to always test for valid data rath
case you simply cannot perceive. For example, consider a sim

**Handling_Java_Web_App**

## Activity

228 followers

Posts Stephen Patrick created, shared, or commented on in the last 90 days are displayed here.

See all activity

## Experience

### Senior Software Engineer

DiManEx · Full-time

Sep 2017 – Present · 3 yrs 11 mos

### Search / Data Science Team

Workday · Full-time

Jan 2017 – Present · 4 yrs 7 mos

County Dublin, Ireland

As a senior software engineer I worked as part of a team building a new search platform scaling it for fortune 500 customers that used data science & machine learning for improved search relevance. ...see more

### Senior Software Engineer / Team Lead

Fidelity Investments · Full-time

2010 – Present · 11 yrs

County Dublin, Ireland

Worked on platform that managed trades / transactions used by fund mangers / traders to make trading decisions.

### Senior Software Engineer

Arconics

2009 – 2010 · 1 yr

### Software Engineer

IBM, DUBLIN SOFTWARE LAB

Dec 2003 – Sep 2008 · 4 yrs 10 mos

## Education

**London University**

Bachelor’s Degree, Computer Science, First Class

## Licenses & certifications

**Sun Certified Enterprise Architect (SCEA)**

Issued 2005 · No Expiration Date

**Sun Certified Java Business Component Developer (SCJBCD)**

Issued 2004 · No Expiration Date

**Sun Certified Java Developer (SCJD)**

Issued 2004 · No Expiration Date

Show more ︾