

New issue

Jump to bottom

[Security] XSS in HelpModal leading to RCE via imported malicious data file #338

Closed magicOz opened this issue on Jun 21, 2020 · 4 comments · Fixed by #560

Labels bug fixed-vnext

magicOz commented on Jun 21, 2020

The `help text`-modal utilizes the React component attribute `dangerouslySetInnerHTML` when rendering the `Info`, `Abuse Info`, etc. texts. E.g.

BloodHound/src/components/Modals/HelpTexts/GenericAll/GenericAll.jsx
Lines 31 to 37 in 338e197

```
31     dangerouslySetInnerHTML={Abuse(  
32       sourceName,  
33       sourceType,  
34       targetName,  
35       targetType,  
36       targetId  
37     )}
```

This makes the application vulnerable to XSS unless the input parameters are properly sanitized/encoded.

It turns out that the parameter `targetId` (objectid) isn't encoded, and is reflected in multiple `Abuse Info`-texts - making the application vulnerable.

BloodHound/src/components/Modals/HelpModal.jsx
Line 57 in 338e197

```
57     settargetId(target.objectid);
```

Since Bloodhound is built using Electron, it is possible to spawn child processes from an XSS vector - leading to a RCE vulnerability.

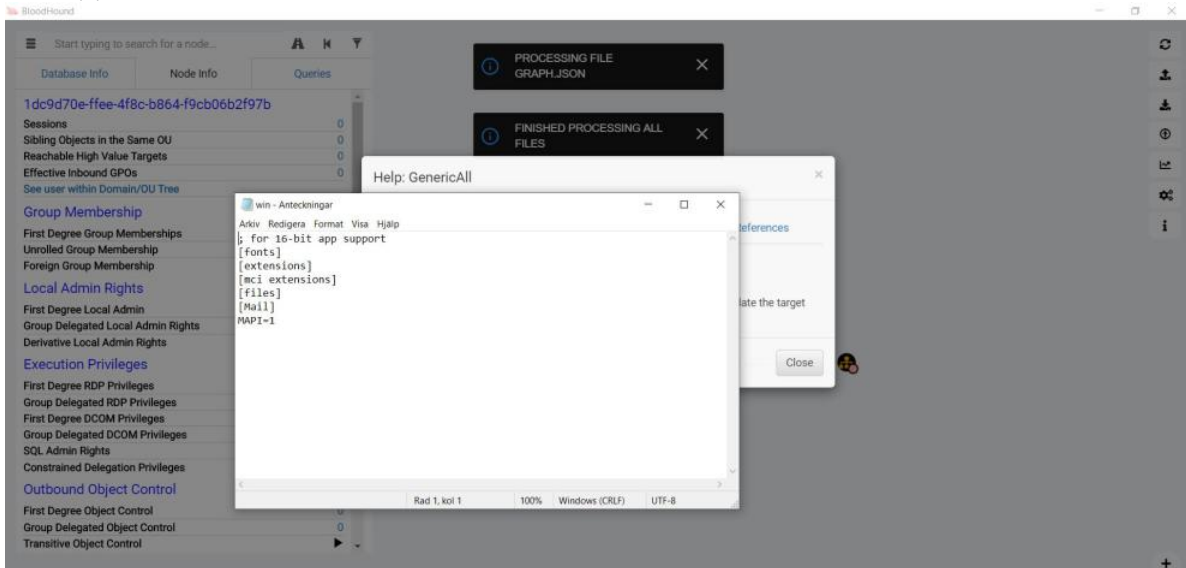
By getting the victim to import a malicious data graph file and clicking `Help` on an edge connected to a malicious node, the XSS payload will trigger.

To mitigate this, encoding `objectid` the same way the node labels are encoded should do the trick.

PoC (Windows):

I've attached a zip, [graph.zip](#), containing a malicious file, `graph.json`. (You may need to unzip and manually import the JSON-file).

1. Import the file `graph.json` into BloodHound.
2. Click `Help` on the edge between `NODE1@DOMAIN.COM` and `MALICIOUS@DOMAIN.COM`.
3. This should pop `notepad.exe C:/windows/win.ini`



magicOz changed the title ~~XSS in HelpModal leading to RCE via imported malicious data file~~ [Security] XSS in HelpModal leading to RCE via imported malicious data file on Jun 30, 2020

OS-WS commented on Feb 21, 2021


Hi, is there any fix for this security issue?
If so, in what commit?

Thanks!

HHAC commented on Jul 19, 2021

Following up on the previous question. Had this issue been resolved? If so what commit should I be looking at?

Thank you!

 **rvazarkar** added the `bug` label on Jan 11

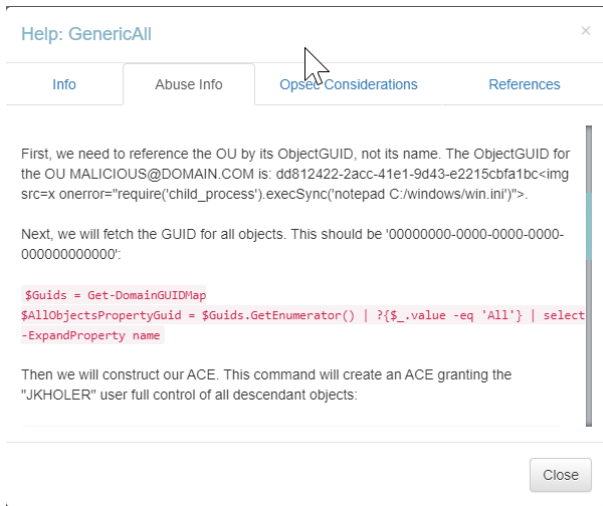
ghost commented on Jul 19

Is there a specific reason why the help texts for example in `Abuse.jsx` are templated strings, not JSX components?

rvazarkar commented on Jul 19

Contributor

Mostly because it would take ages to convert them to JSX components at this point. I'm updating the code to encode all the variables passed to the help modal to prevent this for now. I've tested on the 4.2 branch with the PoC and it no longer works.



 **ghost** mentioned this issue on Jul 26

Fix XSS by rewriting help text modals as JSX components #560

🔗 Merged

 **rvazarkar** added the `fixed-vnext` label on Aug 1

 **rvazarkar** closed this as completed in `e1845c0` on Aug 3

Assignees

No one assigned

Labels

`bug` `fixed-vnext`

Projects

None yet

Milestone

No milestone

Development

Successfully merging a pull request may close this issue.

🔗 Fix XSS by rewriting help text modals as JSX components

4 participants

