14

Open redirect on https://blog.fuzzing-project.org

Share:

State	Resolved (Closed)
Disclosed	November 10, 2018 12:54pm +0100
Reported To	Hanno's projects
Asset	*.fuzzing-project.org (Domain)
Weakness	Open Redirect
Severity	Medium (4 ~ 6.9)
Participants	
Visibility	Disclosed (Full)

Collapse

TIMELINE · EXPORT

juliocesar submitted a report to **Hanno's projects**. Jun 29th (about 1 year ago)

Summary:

There is an Open Redirect on https://blog.fuzzing-project.org/exit.php?url= due to the application not checking the value passed by the user to the "url" parameter.

Description:

Unchecked redirects occur when an application redirects to a destination controlled by attackers. This often occurs in functionality returning users to a previous page, e.g. after authenticating.

An attacker can control the value of the "url" parameter and make it redirect to a malicious endpoint.

https://blog.fuzzing-project.org/exit.php?url=

Steps To Reproduce:

Here is a proof of concept to demonstrate how an open redirect occurs. Please note that this particular example is not a vulnerability and just here for demonstration purposes.

PoC: https://blog.fuzzing-project.org/exit.php?url=aHR0cHM6Ly93d3cuaW5mb3NlYy5jb20uYnI=

The URL looks like it should go to https://blog.fuzzing-project.org, but you are redirected to https://www.infosec.com.br

Supporting Material/References:

Mitigation:

When possible, do not allow user input to directly control redirect destinations; rather, generate them on the server side (e.g. via ID -> URL mapping). When this is not an option, a strict whitelist is highly recommended. Finally, a last-ditch mitigation can be performed by removing protocol specifiers from user input prior to redirection. This last method will not fix intra-site redirect exploits, but can prevent redirects to an attacker-controlled website.

Reference:

https://www.owasp.org/index.php/Unvalidated_Redirects_and_Forwards_Cheat_Sheet

Impact

Attackers may be able to use this to execute believable phishing attacks, bypass authentication, or (in rare circumstances) violate CSRF mitigations.

1 attachment:

F313661: Screencast_2018-06-29_10_00_28.mp4

juliocesar posted a comment. Jun 29th (about 1 year ago)

BTW you have to base64 encode the url

hanno posted a comment. Jun 29th (about 1 year ago)
Thanks for the report. I can reproduce it.

The affected page is running the thirdparty blog software serendipity. I reported the issue upstream. Will update you with the state.

If s9y decides to handle this as a security bug shall I tell them that they mention you as the finder if there's a public advisory? And if yes under what name?

juliocesar posted a comment. Jun 29th (about 1 year ago)
Hi @hanno

Yes, you can mention my name. You can user Julio Soares from Infosec Security, www.infosec.com.br

Can you change the status of the report to triage?

If you need anything else please let me know.

hanno changed the status to o Triaged. Jun 29th (about 1 year ago)

oreamnos filed a duplicate (#373932) and was invited to participate in this report. Jun 29th (about 1 year ago)

hanno posted a comment. Jun 30th (about 1 year ago)
Can you tell me if you found a link anywhere from the existing blog to exit.php? Or did you only learn about its existence by investigating the serendipity source code?

As a temporary workaround I have deleted the file (on all three affected blogs that all use the same software), as I believe it's not needed for my blog to function. I believe it's for optional functionality to set links in a way that don't send referrers.

juliocesar posted a comment. Jun 30th (about 1 year ago)
Hey @hanno

I kinda stumbled on this one, but then I found this piece of their code on github

1 attachment:
F313843: PoC.jpg

juliocesar posted a comment. Jun 30th (about 1 year ago)
BTW the vulnerability is gone.

oreamnos posted a comment. Jul 11th (about 1 year ago)
For the record, the vulnerability was reported to upstream in https://github.com/s9y/Serendipity/issues/558.

juliocesar posted a comment. Jul 11th (about 1 year ago)
Hi @oreamnos

I sent an email to the application support but got no response. So I decided to report the issue through their github repository.

hanno posted a comment. Jul 18th (about 1 year ago)
This is now fixed with a patch I got from s9y's dev (as should be all other open issues).
I'd appreciate if you could check that and also look for further issues, s9y will soon release a security update.

oreamnos posted a comment. Jul 18th (about 1 year ago)
I'm OK with the fix (https://github.com/s9y/Serendipity/commit/19513cdf143ef5659f8afbfb3b16df921060d550#diff-97ba293111573db36dfc13b568cf8510).

juliocesar posted a comment. Jul 18th (about 1 year ago)
Hi @hanno
Looks good to me

juliocesar posted a comment. Jul 24th (about 1 year ago)
Hi @hanno

You can close this report as resolved

hanno posted a comment. Jul 24th (about 1 year ago)

I'll close all the reports once the upstream software (serendipity) is fixed and has made a release.

hanno closed the report and changed the status to ○ **Resolved**. Aug 16th (12 months ago)

Fixed in Serendipity and updated:
http://blog.s9y.org/archives/278-Serendipity-2.1.3-released.html

hanno posted a comment. Oct 12th (10 months ago)

Hi @juliocesar it seems it's not possible for me to make this report public.
Do you want to make this issue public, e.g. posting it to a public mailing list like oss-security, so we have a public reference for it?

juliocesar posted a comment. Oct 12th (10 months ago)

Hi @hanno

Your program is private because of that you won't be able to make this report public. Its ok you don't have to make it public.

hanno requested to disclose this report. Nov 9th (9 months ago)

juliocesar agreed to disclose this report. Nov 10th (9 months ago)

This report has been disclosed. Nov 10th (9 months ago)