

Inefficient Regular Expression Complexity in idank/explainshell



Valid Reported on Dec 22nd 2021

Description

In the latest version of explainshell (ebc5e9f2) I discovered regular expression that is vulnerable to ReDoS (Regular Expression Denial of Service)

Proof of Concept

PoC based on code in explainshell/options.py

```
import logging
import re

if __name__ == "__main__":
    logging.basicConfig(format='%(asctime)s - %(levelname)s: %(message)s',
                        level=logging.DEBUG)
    opt_regex = re.compile(r'''
        (?P<opt>--?(?:\?|\#|(?:\w+-)*\w+)) # option starts with - or -- a
        (?
            (?:\s*(=)?\s*) # -a=
            (?P<argoptional>[<\[])? # -a=< or -a=[
            (?:\s*(=)?\s*) # or maybe -a=<=
            (?P<arg>
                (?<argoptional>
                    # if we think we have an arg (we saw [ c
                    [^\]>]+ # either read everything until the closi
                    |
                    (?2)
                    [-a-zA-Z]+ # or if we didn't see [ or < but just s
                    |
                    [A-Z]+ # but if we didn't have =, only allow up
                )
            )
        )
        (?<argoptional>(?P<argoptionalc>[>\]]) # read closing ] or > if
    )? # the whole arg thing is optional
    (?P<ending>,\s*|\s+|\Z|/|\|)''', re.X) # read any trailing whitesp

for i in range(1, 10000):
    s = '-?' + ' ' * i * 100
```



Chat with us

```
logging.debug('Started with {} spaces'.format(i * 100))

opt_regex.match(s, 0)
logging.debug('Finished with {} spaces'.format(i * 100))
```

Output:

```
2021-12-22 16:57:29,670 - DEBUG: Started with 100 spaces
2021-12-22 16:57:29,853 - DEBUG: Finished with 100 spaces
2021-12-22 16:57:29,853 - DEBUG: Started with 200 spaces
2021-12-22 16:57:35,690 - DEBUG: Finished with 200 spaces
2021-12-22 16:57:35,690 - DEBUG: Started with 300 spaces
2021-12-22 16:58:00,872 - DEBUG: Finished with 300 spaces
2021-12-22 16:58:00,873 - DEBUG: Started with 400 spaces
2021-12-22 16:59:12,585 - DEBUG: Finished with 400 spaces
2021-12-22 16:59:12,585 - DEBUG: Started with 500 spaces
2021-12-22 17:02:08,048 - DEBUG: Finished with 500 spaces
2021-12-22 17:02:08,048 - DEBUG: Started with 600 spaces
2021-12-22 17:08:26,772 - DEBUG: Finished with 600 spaces
2021-12-22 17:08:26,772 - DEBUG: Started with 700 spaces
2021-12-22 17:20:05,980 - DEBUG: Finished with 700 spaces
2021-12-22 17:20:05,980 - DEBUG: Started with 800 spaces
2021-12-22 17:39:39,827 - DEBUG: Finished with 800 spaces
2021-12-22 17:39:39,827 - DEBUG: Started with 900 spaces
2021-12-22 18:14:06,240 - DEBUG: Finished with 900 spaces
2021-12-22 18:14:06,240 - DEBUG: Started with 1000 spaces
2021-12-22 19:12:37,211 - DEBUG: Finished with 1000 spaces
```

Impact


This issues may lead to a denial of service if user controls input passed to pattern matching function.

Occurences

 options.py L23

References

- https://owasp.org/www-community/attacks/Regular_expression_Denial_of_Service_-_ReDoS

 Chat with us

Weakness: [CWE-1333](#)

Severity: **Medium (6.2)**

Visibility:

Private

Status:

Awaiting fix

Confidential

Disclosure Bounty:
\$13.5

Fix Bounty:
\$3.38

Reported by



theworstcomrade

@theworstcomrade

master



Patch Vulnerability

Fork repository

Submit a patch

Other Inefficient Regular Expression Complexity Advisories

fabricjs/fabric.js

• High (7.5)

jaywcjlove/colors-cli

• High (7.5)

alvations/sacremoses

• High (7.5)

We are processing your report and will contact the **idank/explainshell** team within 24 hours. 7 days ago

We created a **GitHub Issue** asking the maintainers to create a SECURITY.md 6 days ago

We have contacted a member of the **idank/explainshell** team and are waiting to hear back 5 days ago

idank 4 days ago

Maintainer

Nice find. Do you have an idea for a fix?

theworstcomrade submitted a patch 2 days ago

theworstcomrade 2 days ago

Chat with us

@idank please look at my fix which I submitted. It works good and covers below tests.

```
import logging
import re


if __name__ == "__main__":
    logging.basicConfig(format='%(asctime)s - %(levelname)s: %(message)s',
                        level=logging.DEBUG)
    opt_regex = re.compile(r'''
        (?P<opt>--?(?:\?|\#|(?:\w+-)*\w+)) # option starts with - or -- and can have
        (?
            (?:\s*((?P<argoptional>(=<|=<|=\\)|=)\s*)) # -a= or -a=< or -a=[ or maybe
            (?P<arg>
                (? (argoptional) # if we think we have an arg (we saw [ or <)
                    [^\>]+ # either read everything until the closing ] or >
                    |
                    (? (2)
                        [-a-zA-Z]+ # or if we didn't see [ or < but just saw =, read
                        |
                        [A-Z]+ # but if we didn't have =, only allow uppercase let
                    )
                )
            )
            (? (argoptional)(?P<argoptionalc>[^\>])) # read closing ] or > if we have an
        )? # the whole arg thing is optional
        (?P<ending>,\s*|\s+|\Z|/|\|)''', re.X) # read any trailing whitespace or the

    for i in range(1, 10000):
        s = '-?' + ' ' * i * 1000

        logging.debug('Started with {} spaces'.format(i * 1000))
        opt_regex.match(s, 0)
        logging.debug('Finished with {} spaces'.format(i * 1000))
```

Output:

```
2021-12-26 22:01:12,654 - DEBUG: Started with 1000 spaces
2021-12-26 22:01:12,655 - DEBUG: Finished with 1000 spaces
2021-12-26 22:01:12,655 - DEBUG: Started with 2000 spaces
2021-12-26 22:01:12,655 - DEBUG: Finished with 2000 spaces
2021-12-26 22:01:12,655 - DEBUG: Started with 3000 spaces
2021-12-26 22:01:12,655 - DEBUG: Finished with 3000 spaces
2021-12-26 22:01:12,655 - DEBUG: Started with 4000 spaces
2021-12-26 22:01:12,655 - DEBUG: Finished with 4000 spaces
2021-12-26 22:01:12,655 - DEBUG: Started with 5000 spaces
2021-12-26 22:01:12,656 - DEBUG: Finished with 5000 spaces
2021-12-26 22:01:12,656 - DEBUG: Started with 6000 spaces
2021-12-26 22:01:12,656 - DEBUG: Finished with 6000 spaces
2021-12-26 22:01:12,656 - DEBUG: Started with 7000 spaces
```

 Chat with us

```
2021-12-26 22:01:12,656 - DEBUG: Finished with 7000 spaces
2021-12-26 22:01:12,656 - DEBUG: Started with 8000 spaces
2021-12-26 22:01:12,657 - DEBUG: Finished with 8000 spaces
2021-12-26 22:01:12,657 - DEBUG: Started with 9000 spaces

2021-12-26 22:01:12,657 - DEBUG: Finished with 9000 spaces
2021-12-26 22:01:12,657 - DEBUG: Started with 10000 spaces
2021-12-26 22:01:12,658 - DEBUG: Finished with 10000 spaces
2021-12-26 22:01:12,658 - DEBUG: Started with 11000 spaces
2021-12-26 22:01:12,659 - DEBUG: Finished with 11000 spaces
2021-12-26 22:01:12,659 - DEBUG: Started with 12000 spaces
2021-12-26 22:01:12,659 - DEBUG: Finished with 12000 spaces
2021-12-26 22:01:12,659 - DEBUG: Started with 13000 spaces
2021-12-26 22:01:12,660 - DEBUG: Finished with 13000 spaces
2021-12-26 22:01:12,660 - DEBUG: Started with 14000 spaces
2021-12-26 22:01:12,661 - DEBUG: Finished with 14000 spaces
2021-12-26 22:01:12,661 - DEBUG: Started with 15000 spaces
2021-12-26 22:01:12,662 - DEBUG: Finished with 15000 spaces
2021-12-26 22:01:12,662 - DEBUG: Started with 16000 spaces
2021-12-26 22:01:12,663 - DEBUG: Finished with 16000 spaces
2021-12-26 22:01:12,663 - DEBUG: Started with 17000 spaces
2021-12-26 22:01:12,664 - DEBUG: Finished with 17000 spaces
2021-12-26 22:01:12,664 - DEBUG: Started with 18000 spaces
2021-12-26 22:01:12,665 - DEBUG: Finished with 18000 spaces
2021-12-26 22:01:12,665 - DEBUG: Started with 19000 spaces
2021-12-26 22:01:12,666 - DEBUG: Finished with 19000 spaces
2021-12-26 22:01:12,666 - DEBUG: Started with 20000 spaces
2021-12-26 22:01:12,667 - DEBUG: Finished with 20000 spaces
2021-12-26 22:01:12,667 - DEBUG: Started with 21000 spaces
2021-12-26 22:01:12,668 - DEBUG: Finished with 21000 spaces
2021-12-26 22:01:12,668 - DEBUG: Started with 22000 spaces
2021-12-26 22:01:12,669 - DEBUG: Finished with 22000 spaces
2021-12-26 22:01:12,669 - DEBUG: Started with 23000 spaces
2021-12-26 22:01:12,670 - DEBUG: Finished with 23000 spaces
2021-12-26 22:01:12,670 - DEBUG: Started with 24000 spaces
2021-12-26 22:01:12,671 - DEBUG: Finished with 24000 spaces
2021-12-26 22:01:12,672 - DEBUG: Started with 25000 spaces
2021-12-26 22:01:12,673 - DEBUG: Finished with 25000 spaces
```

```
-a=VALUE
-a = VALUE
-a =VALUE
-a<=VALUE
-a <=VALUE
-a <= VALUE
-a<= VALUE
-a<VALUE
-a=[VALUE
```

Are the two functionally equivalent? They seem to be but I'm not 100% sure. (:

If we're certain they are, would be happy if you sent me a pull request. I'll merge it and update the backend.

✉ We have sent a follow up to the [idank/explainshell](#) team. We will try again in 7 days. 2 days ago

theworstcomrade 2 days ago

Researcher

@idank I checked it once again and in my first fix was bug - not all built in tests finished with success. I have made one more commit, which covers sample above file and tests from test-options.py.

The change I made is only limited to the number of spaces before and after =
Before:

```
(?:\s*(=)\s*)          # -a=  
(?P<argoptional>[<\[])? # -a=< or -a=[  
(?:\s*(=)\s*)          # or maybe -a<=
```

After:

```
(?:\s?(=)\s?)          # -a=  
(?P<argoptional>[<\[])? # -a=< or -a=[  
(?:\s?(=)\s?)          # or maybe -a<=
```

By answering your question yes, I am sure it is functionally equivalent to the previous version
Here You have PR <https://github.com/idank/explainshell/pull/290>


✓ **idank** validated this vulnerability 41 minutes ago

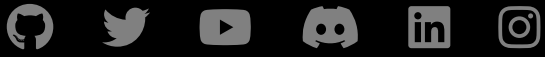
\$ **theworstcomrade** has been awarded the disclosure bounty ✓

\$ The fix bounty is now up for grabs

 Write a comment (supports markdown). Use @admin for support...

00 Pr

 Chat with us



huntr

[home](#)

[blog](#)

[FAQ](#)

[contact us](#)

[terms](#)


[privacy policy](#)

part of 418sec

[company](#)

[about](#)

[team](#)

 [Chat with us](#)