

# Inefficient Regular Expression Complexity in nltk/nltk

Valid Reported on Dec 7th 2021

0

## Description

`nltk` is vulnerable to ReDoS attack because of `^~?[0-9]+(.[0-9]+)?$` regex. If attacker succeeds to use malicious payload against `RegexpTagger` used in function `get_pos_tagger` and `malt_regex_tagger`, it will cause a nasty DoS.

## Proof of Concept

```
// PoC.py
import re, time

pattern = re.compile("^~?[0-9]+(.[0-9]+)?$")
s = "-"
s += "0" * 50000
s += "q"

t = time.time()
print("searching...")
re.search(pattern, s)
print(time.time() - t)
```

On my new machine I needed only 50k characters to cause a 23+ seconds matching. For instance, in similar [report](#) to this project 160k characters were processed just in 3+ seconds.

## Issue

The issue here is that in `^~?[0-9]+(.[0-9]+)?$` groups `[0-9]+(.[0-9]+)` match each other, which causes a nasty backtracking in case of failure.

## Impact

This vulnerability is capable of causing DoS due to CPU resources consumption.

## Occurrences

glue.py L706

CVE  
CVE-2021-3842  
(Published)

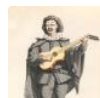
Vulnerability Type  
CWE-1333: Inefficient Regular Expression Complexity

Severity  
High (7.5)

Visibility  
Public

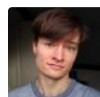
Status  
Fixed

Found by



Scaramouche  
@scara31  
unranked

Fixed by



Tom Aarsen  
@tomaarsen  
maintainer

This report was seen 487 times.

We are processing your report and will contact the `nltk` team within 24 hours. a year ago

We have contacted a member of the `nltk` team and are waiting to hear back a year ago

Tom Aarsen validated this vulnerability a year ago

Chat with us

Scaramouche has been awarded the disclosure bounty ✓

The fix bounty is now up for grabs

Tom Aarsen submitted a patch a year ago

Tom Aarsen a year ago

Maintainer

Thank you for reporting this! A patch should be good to go soon.

Tom Aarsen marked this as fixed with commit 2a50a3 a year ago

Tom Aarsen has been awarded the fix bounty ✓

This vulnerability will not receive a CVE ✗

glue.py#L706 has been validated ✓

Scaramouche a year ago

Researcher

@admin

Greetings, I was told that CVEs are assigned and published in roughly 1 hour after the fix. This repo used to assign CVEs for the same bug: <https://nvd.nist.gov/vuln/detail/CVE-2021-3828>  
Has something changed?

Jamie Slome a year ago

Admin

@scara31 - thanks for getting in touch!

Our system no longer automatically assigns CVEs for certain CWE types, including Inefficient Regular Expression Complexity, however, if the maintainer (@tomaarsen) is happy, we can go ahead and publish a CVE for this report.

Scaramouche a year ago

Researcher

@admin

Got it, thanks for reply! Then I will try to contact @tomaarsen

Tom Aarsen a year ago

Maintainer

@scara31 Consider me contacted -

I'm happy with the fix that is in place, but I must say that a fixed release has not yet been published. I'm unsure whether the CVE ought to only be created when such a release is out. If so, then we should wait.  
Otherwise, feel free to publish the CVE.

Scaramouche a year ago

Researcher

@tomaarsen

That's good to hear, of course I can wait as much as you need!

Tom Aarsen a year ago

Maintainer

The newest release has been published, containing this patch. Thanks again.

Scaramouche a year ago

Researcher

@tomaarsen

It's great to hear it! Should I ask admin to assign the CVE, if you let me?

Tom Aarsen a year ago

Maintainer

That sounds wise. Feel free.

Jamie Slome a year ago

Admin

CVE published! 🎉

Sign in to join this conversation

## huntr

[home](#)

[hacktivity](#)

[leaderboard](#)

[FAQ](#)

[contact us](#)

[terms](#)

[privacy policy](#)

## part of 418sec

[company](#)

[about](#)

[team](#)