CVE-2015-3183

Term paper in Computer Security (MTAT.03.134)

Author: Mats Kivistik
Supervisor: Meelis Roos

Tartu, 2016
Introduction

Apache HTTP Server is the world’s most used web server software with over 100 million websites served in 2009 and an estimated 50% share of overall active websites in 2015 [1]. Therefore any security flaws with the software can have a very large worldwide impact. The software is developed and maintained as open-source software. In 20.07.2015 a security vulnerability was reported, which affects all Apache HTTP Servers prior to 2.4.14 [2]. This vulnerability is CVE-2015-3183.

Short description of the vulnerability

The security flaw lies within the chunked transfer coding implementation of the Apache HTTP Server. The code does not parse the chunked requests correctly and the client could force the server to misinterpret the requests length. And therefore there is a possibility for the client to create HTTP request smuggling attacks [2], [3], [4].

Exploiting the vulnerability

The bug makes it possible for the client to generate HTTP request smuggling attacks. But what is HTTP request smuggling? It means that the person attacking the server can generate a request with a chunked header that will force the server to misinterpret the length of the request and therefore embed another request into the body of the first request. And in doing so they can bypass the proxy/firewall with their embedded request that can do harm to the system. Figure 1 shows how one HTTP request can be written inside another HTTP request in the same header. Here proxy parses the first request that starts on line 1. Then it assumes that the body length is 44 bytes and therefore it also includes data from lines 8-10 as the first requests body. And after the request has bypassed the proxy server, the application server/final system will parse the second encapsulated request. With a well crafted request like that a client could send the server requests that can exploit the server in different ways. So exploiting CVE-2015-3183 can lead to these attacks [3], [6]:

1) Cache poisoning
2) Hijacking of credentials
3) Bypass web application firewall protection
4) Conduct XSS attacks

```
1 POST http://SITE/foobar.html HTTP/1.1
2 Host: SITE
3 Connection: Keep-Alive
4 Content-Type: application/x-www-form-urlencoded
5 Content-Length: 0
6 Content-Length: 44
7 [CRLF]
8 GET /poison.html HTTP/1.1
9 Host: SITE
10 Bla: [space after the "Bla:", but no CRLF]
11 GET http://SITE/page_to_poison.html HTTP/1.1
12 Host: SITE
13 Connection: Keep-Alive
14 [CRLF]
```

Figure 1. HTTP request smuggling [5]

**Possible Solutions to the problem**

Chunked transfer means that the server starts transferring data already when the full reply is not ready. So it responds as soon as possible with a bit of the reply until the whole reply is sent. Therefore in that case the connection uses Keep-Alive to make sure that the connection is not broken in the middle of the transfer. One way to prevent anyone of abusing CVE-2015-3183 on your server would be to turn off Keep-Alive. Then the server wouldn’t read multiple requests from one single connection and would close the connection after the request. Another possibility would be to reject connections with chunked encoding [7]. These ways would be sufficient to deal with the CVE-2015-3183, as the latter blocks all possible connections that use the vulnerability.

But since Apache HTTP Servers 2.14... this vulnerability has also been fixed by fixing the chunk header parsing defect. This fix included removing apr_brigade_flatten(), buffering and duplicated code from the HTTP_IN filter, parse chunks in a single pass with zero copy, limit accepted chunk-size to $2^{63}-1$ [8].
Resources


