httpbis M. Chen
Internet-Draft Li. Su
Intended status: Informational China Mobile
Expires: December 17, 2021 June 15, 2021

http2 window size use case
draft-chen-httpbis-window-size-use-case-01

Abstract

This document presents an use case which actually happening in our network, when window_size_increment in the window update frame less than 128 bytes and the increased window size also less than 128 bytes, then network connection will come to an error.

Status of This Memo

Copyright Notice

Table of Contents

1. Introduction

The following content is from RFC 7540[RFC7540]

When an HTTP/2 connection is first established, new streams are created with an initial flow-control window size of 65,535 octets. The connection flow-control window is also 65,535 octets. Both endpoints can adjust the initial window size for new streams by including a value for SETTINGS_INITIAL_WINDOW_SIZE in the SETTINGS frame that forms part of the connection preface. The connection flow-control window can only be changed using WINDOW_UPDATE frames.

SETTINGS_INITIAL_WINDOW_SIZE (0x4): Indicates the sender's initial window size (in octets) for stream-level flow control. The initial value is 2^16-1 (65,535) octets.

Window Size Increment defined in the Window_update is 31, the legal range for the increment to the flow-control window is 1 to 2^31-1 (2,147,483,647) octets.

RFC 7540 just Specifies the maximum value of Window and the Window Size Increment, But there is no obvious rule about minimum values.

2. Terminology

The readers should be familiar with the terms defined in.

In addition, this document makes use of the following terms:

Window_update: The WINDOW_UPDATE frame (type=0x8) is used to
   implement flow control;

3. Use Case

This section describes use case which happens between two different manufacturers. They both use HTTP2.0 protocol to transmit messages. We found this phenomenon, one issues a regular registration request,

the other one receives the request, but judged to be attack behaviour.

```
 +---------+                        +----------+
 | Sender  |                        |Receiver  |
 |         |                        |          |
 +----+----+                        +-----+----+
      |    Reqistration request           |
      +----------------------------------->
      |                                   |
      |                                   |
      +----------------------------------->
      |     Http2 Window_update           |
      |    (condition1:window size        |
      |                      increment)   |
      |                                   |window+=window size increment
      |                                   |condition2: window
      |                                   |
      |                                   |condition1<128bytes
      |      connection break             |& condition2<128bytes
      |    +--------------------------+   |=attack behaviour
      |                                   |
      |      Registration Failed          |
      +<----------------------------------+
      +                                   +
```

Figure 1: A normal request is considered an attack

Why determine abnormal attack behavior, the analysis is as follows:

The default initial window size defined by the protocol is 64K. After the data in the receiving window is removed, part of the window occupied by the original data is released.

If there is a large backlog of data in the original receiving window that has not been removed, resulting in a small remaining window, the window added after the normal removal of data will not be too small. If there is little backlog of data in the original receiving window, the window that needs to be added after the data is removed will be small, but the overall available window after the adjustment will be larger. In neither case will the window be too small, So the connection considered to be an attack.

So when need to solve this problem, two approaches can be discussed, specifying a minimum value for window and window size increment, or increasing more detailed flow control strategies.

4. Security Considerations

Failure to set a minimum will result in frequent window_update if only process a small amount of data at a time, it is likely to occur Denial of service attacks, it would be fatal if it happened in an Internet of Things scenario. In draft-ietf-httpbis-http2bis, there are also Denial-of-Service consideration in section 10.5, includes the misuse of some parameters and priorities.

5. IANA Considerations

This document does not require any action from IANA.

6. Acknowledgement

TBD

7. Informative References

[RFC7540] Belshe, M., Peon, R., and M. Thomson, Ed., "Hypertext Transfer Protocol Version 2 (HTTP/2)", RFC 7540, DOI 10.17487/RFC7540, May 2015, <https://www.rfc-editor.org/info/rfc7540>.

Authors' Addresses

Meiling Chen
China Mobile
32, Xuanwumen West
BeiJing, BeiJing 100053
China

Email:
chenmeiling@chinamobile.com

Li Su
China Mobile

32, Xuanwumen West

BeiJing

100053

China

Email:
suli@chinamobile.com