http2 window size setting
draft-chen-httpbis-window-size-01

Abstract

   This document proposed the minimum value setting mechanism for
   HTTP2.0 Window and Window_update, and a Window_update frame sending
   mechanism.

Status of This Memo

   This Internet-Draft is submitted in full conformance with the
   provisions of BCP 78 and BCP 79.

   Internet-Drafts are working documents of the Internet Engineering
   Task Force (IETF). Note that other groups may also distribute
   working documents as Internet-Drafts. The list of current Internet-
   Drafts is at https://datatracker.ietf.org/drafts/current/.

   Internet-Drafts are draft documents valid for a maximum of six months
   and may be updated, replaced, or obsoleted by other documents at any
   time. It is inappropriate to use Internet-Drafts as reference
   material or to cite them other than as "work in progress."

   This Internet-Draft will expire on December 17, 2021.

Table of Contents

1. Introduction

The following content is from RFC 7540[RFC7540]

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. Only DATA frames are subject to
flow control.

HTTP/2 defines only the format and semantics of WINDOW_UPDATE frames,
and does not specify how the receiver decides when to send frames,
what values to send, or how the sender chooses to send packets. And
RFC 7540 just Specifies the maximum value of Window and the Window
Size Increment, But there is no obvious rule about minimum values.
The implementation can choose any algorithm that meets the
requirements, this makes the difference.

In the current network, there is no standard minimum setting, which
leads to the inconsistency of message processing between
communication parties, which may led to the situation that the
message will be determined as an attack by the recipient, actually
frequent window_UPDATE frames can result in a denial of service. In
draft draft-ietf-httpbis-02 section 10.5 also give some Denial-of-
Service considerantions.

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;

SETTING The SETTING frame (type=0x4) is used to transmitting
   configuration informations which will affect the communication
   process of the data stream.

3. Setting minimum window size

The parameter of Setting frame in RFC 7540 does not have the function
of Setting the minimum window size. This chapter proposes to add
this new parameter for Setting. The SETTINGS frame (type=0x4)
conveys configuration parameters that affect how endpoints
communicate, such as preferences and constraints on peer behavior.
The SETTINGS frame is also used to acknowledge the receipt of those
parameters.

3.1. Setting new parameter

The following new parameter is defined.

SETTINGS_MINIMUM_WINDOW_SIZE(0x7): Indicates the minimum window size
set by the sender. Allows the sender to inform the remote endpoint
of the minimum window size. For example, when set to 128 Bytes, the
minimum window size is 128 Bytes.

If the sender sends the last Data frame and the Window decreases to
less than the minimum Window, it will stop sending Data frame until
it receives window_UPDATE frame to increase the Window, and the
modified Window value is greater than the minimum set Window, then it
can start sending Data frame again. Note that this is more detail
than RFC7540 discribed, where Data frames can be sent as long as the
Window value is greater than zero.

3.2. new parameter setup process

```
   +------+              +--------+
   |sender|              |receiver|
   +--+---+              +----+---+
      |        SETTING        |
    +--------------------->+
      |     identifier:0x04   | Set the initial
      |                       |  window size
      |        SETTING        |
    <---------------------+
      |     Flags:ACK         |
      |                       |
      |        SETTING        |
    +--------------------->+
      |     identifier:0x07   | Set the Minimum
      |                       | window size
      |        SETTING        |
    +<---------------------+
      |     Flags:ACK         |
      |                       |
```
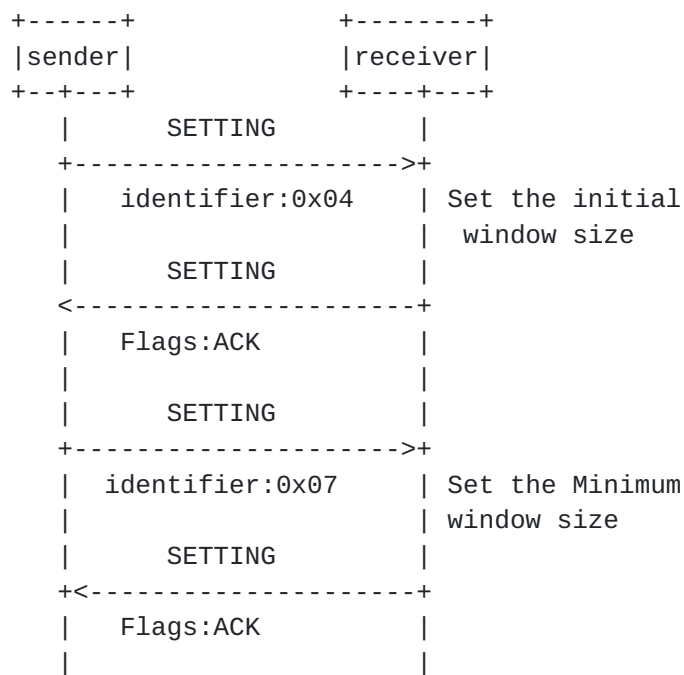
   Figure 1: the process of setting window size

   First, set the initial window size with the identifier
   SETTINGS_INITIAL_WINDOW_SIZE (0x4), values above the maximum flow-
   control window size of 2^31-1 MUST be treated as a connection error
   of type FLOW_CONTROL_ERROR. An ACK is received to indicate that the
   setup is complete.

   Then, set the minimum window size with the identifier
   SETTINGS_MINIMUM_WINDOW_SIZE(0x7), ACK is received to indicate that
   the minimum window size setup is complete. A FLOW_CONTROL_ERROR
   error is thrown when the following SETTINGS_MINIMUM_WINDOW_SIZE set
   in the Setting frame is below the negociative initial minimum value.

4. Setting minimum window update size

   The WINDOW_UPDATE frame (type=0x8) is used to implement flow control.
   The payload of a WINDOW_UPDATE frame is one reserved bit plus an
   unsigned 31-bit integer indicating the number of octets that the
   sender can transmit in addition to the existing flow-control window.
   the unsigned 31-bit integer is knew as Window Size Increment and the
   Size range is (1, 2^31-1), that means the default minimum is 1. So
   this could lead to a problem, frequent sending of Window_UPDATE
   frames with small value of Window Size Increment(such as 1 byte) will
   result in the consumption of computing and network resources, and in
   some cases can even trigger a denial of service attack.

We propose to add new parameter of SETTING frame for Implementation
that set a minimum update window value, It's actually the Window Size
Increment.

4.1. Setting new parameter

The following new parameter is defined.

SETTINGS_MINIMUM_WINDOW_UPDATE(0x8):Indicates that the sender has set
the minimum window_UPDATE update size. For example, when set to 128
Bytes, the minimum window update size is 128 Bytes.

If the buffering data processed by receriver at one time is less than
the minimum window update value, it needs to accumulate to the
minimum value before sending Window_update once to update the traffic
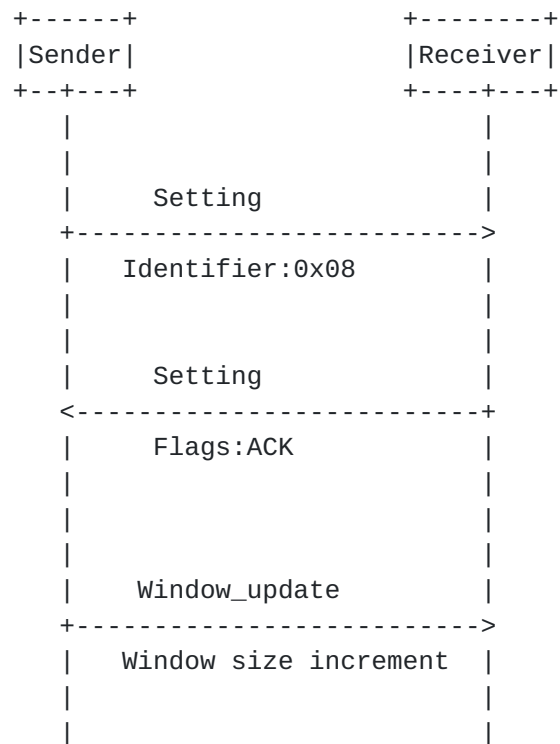window.

4.2. new parameter setup process

```
+------+                    +--------+
|Sender|                    |Receiver|
+--+---+                    +----+---+
   |                            |
   |                            |
   |        Setting             |
   +--------------------------->
   |     Identifier:0x08        |
   |                            |
   |                            |
   |        Setting             |
   <---------------------------+
   |        Flags:ACK           |
   |                            |
   |                            |
   |                            |
   |      Window_update         |
   +--------------------------->
   |     Window size increment  |
   |                            |
   |                            |
```

 Figure 2: the process of setting window_update size

First, set the minimum window_update size with the identifier
SETTINGS_MINIMUM_WINDOW_UPDATE(0x8), An ACK is received to indicate
that the setup is complete. Minimum window_update policy can only be
enabled if SETTINGS_MINIMUM_WINDOW_UPDATE is set.

Then, only when the cumulative amount of processing is greater than
the value of SETTINGS_MINIMUM_WINDOW_UPDATE, can an window_update
frame be sent which will inform the peer to increase the window
value. When the following Window Size Increment value in a
Window_update frame is less than the set negociative initial minimum,
a FRAME_SIZE_ERROR error is thrown.

5. Security Considerations

It solves the attack problem caused by the failure to set the minimum
value of window and window update frame, such as CVE-2019-9511, and
avoids the link congestion caused by small incremental update.

6. IANA Considerations

This document does not require any action from IANA.

7. Acknowledgement

TBD

8. 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