Authors: M. Chen   Li. Su
         China Mobile   China Mobile

http2 window size setting

Abstract

This document proposed the minimum value setting mechanism for
HTTP2.0 Window and Window_update, and a Window_update frame sending
mechanism, used to solve the gap caused by the inconsistency of the
minimum value, f the use case 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. Only DATA frames are subject to
flow control.

9114 HTTP/3(still under the editing phase) is also involved:

SETTINGS_INITIAL_WINDOW_SIZE (0x4): QUIC requires both stream and
connection flow-control window sizes to be specified in the initial
transport handshake. Specifying a setting with the identifier 0x04
(corresponding to the SETTINGS_INITIAL_WINDOW_SIZE parameter) in the
HTTP/3 SETTINGS frame is an error.

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.

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 disdetermined 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. **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     |
     +<-------------------------------+
     +                                +
```
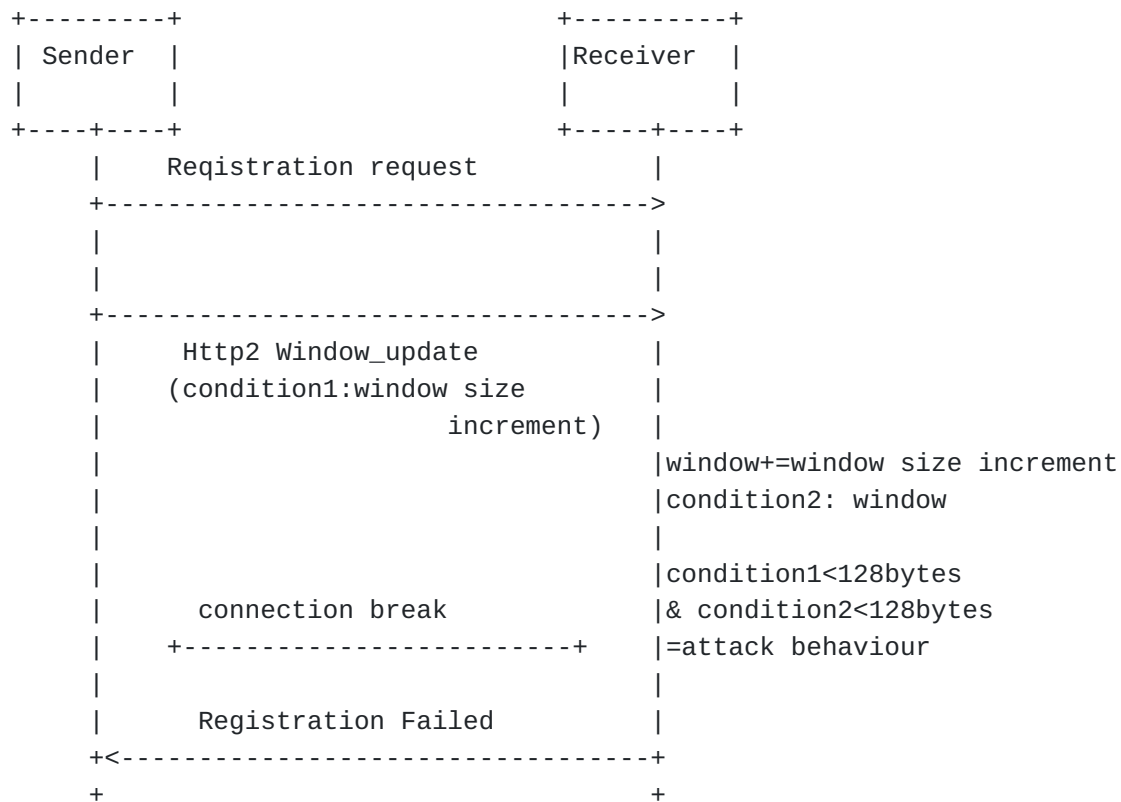   Figure 1: A normal request is considered an attack

   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. 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.

4.1. 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.

4.2. 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          |
   |                       |
```
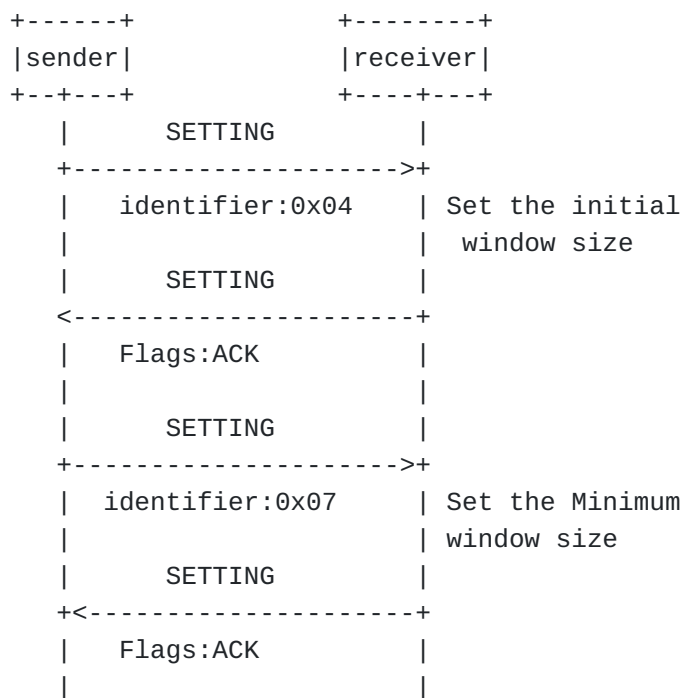
 Figure 2: the process of setting window size

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.

5. 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.

5.1. 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.

5.2. new parameter setup process

```
+------+                 +--------+
|Sender|                 |Receiver|
+--+---+                 +----+---+
   |                          |
   |                          |
   |      Setting             |
   +------------------------->
   |    Identifier:0x08       |
   |                          |
   |                          |
   |      Setting             |
   <--------------------------+
   |      Flags:ACK           |
   |                          |
   |                          |
   |                          |
   |      Window_update       |
   +------------------------->
   |   Window size increment  |
   |                          |
   |                          |
```
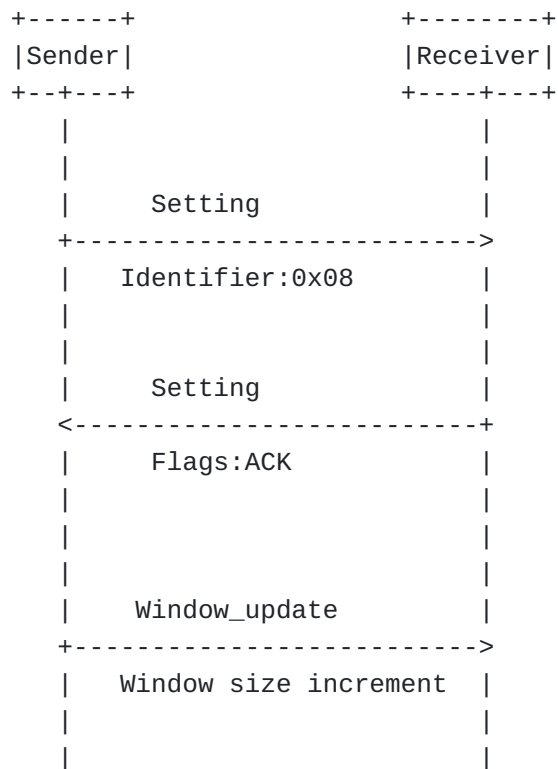
 Figure 3: the process of setting window_update size

   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.

6. 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.

   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.

7. IANA Considerations

This document does not require any action from IANA.

8. Acknowledgement

TBD

9. 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