

# Improve TCP Handling of Out-of-Window Packets to Mitigate Ghost ACKs

Prof. Dr. Christian Rossow and Yepeng Pan

[rossow@cispa.de](mailto:rossow@cispa.de), [yepeng.pan@cispa.de](mailto:yepeng.pan@cispa.de)

[1] [Improve TCP Handling of Out-of-Window Packets to Mitigate Ghost ACKs \(ietf.org\)](https://www.ietf.org/)



# List of Content

Overall, the submitted ID focuses on the **ACK number validation** for **established TCP connections** and aims to solve a problem that allows segments with out-of-window ACK numbers to be accepted ("**Ghost ACK**").

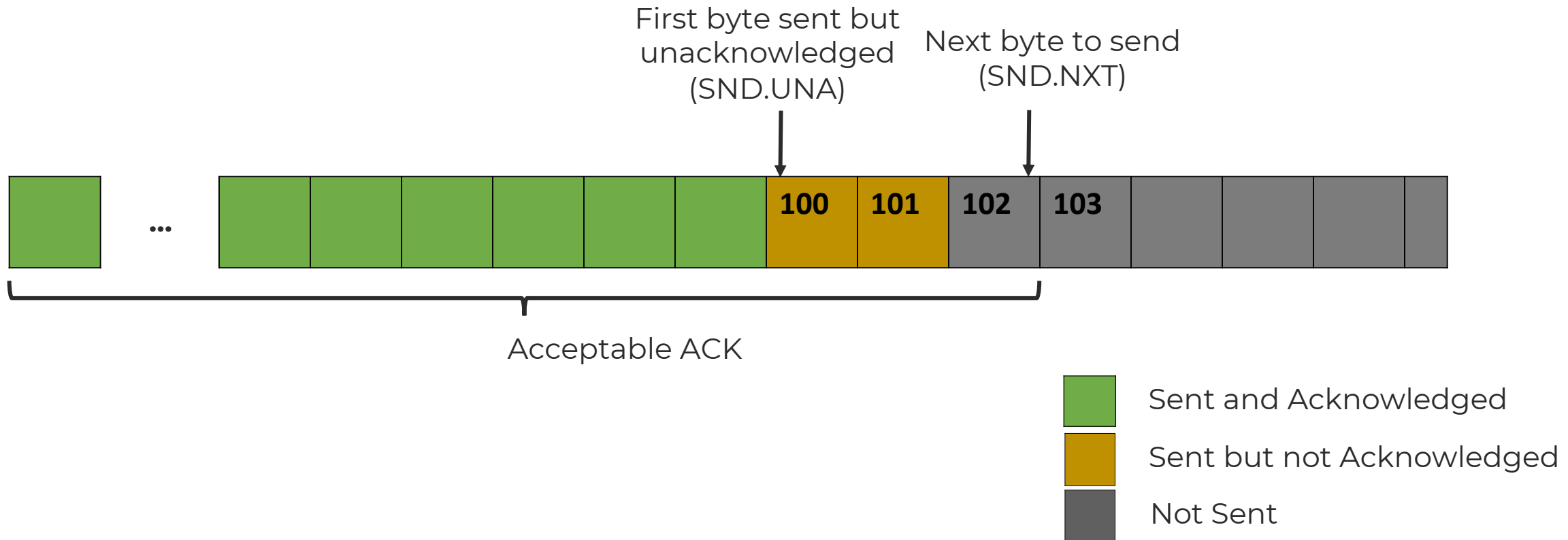
- Review of current RFC
- Definition of Ghost ACK
- Proposed Solution
- Implementation Status



# Current State (1/2) (RFC 9293)

As of the current RFC for TCP, [RFC 9293](#), the TCP stack would employ the following ACK and SEQ check for incoming segments of an established connection:

- o SEG.ACK: As per RFC 9293, packets with  $!SEG.ACK > SND.NXT$  are acceptable (though  $SEG.ACK \leq SND.UNA$  are duplicates), while packets with  $SEG.ACK > SND.NXT$  acknowledge never sent data, and thus are not acceptable.



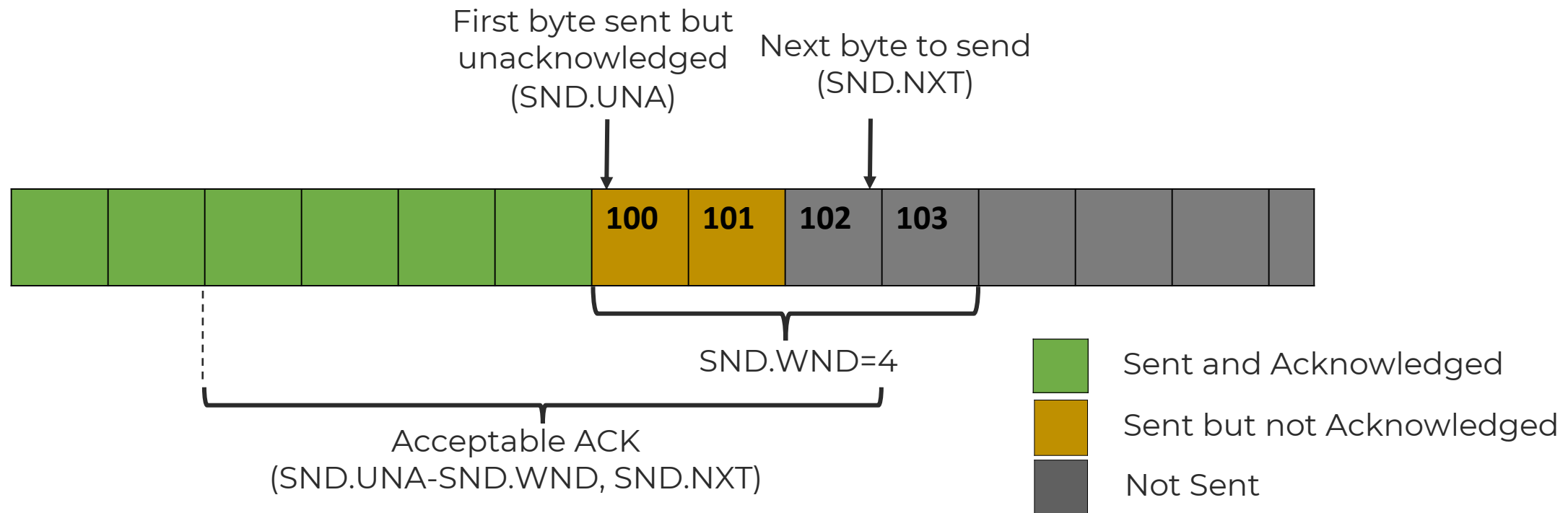


## Current State (2/2) (RFC 5961)

- [RFC 5961](#) proposes to apply stricter checks over **acceptable** SEG.ACK numbers:

The ACK value is considered acceptable only if it is in the range of  $((\text{SND.UNA} - \text{MAX.SND.WND}) \leq \text{SEG.ACK} \leq \text{SND.NXT})$ .

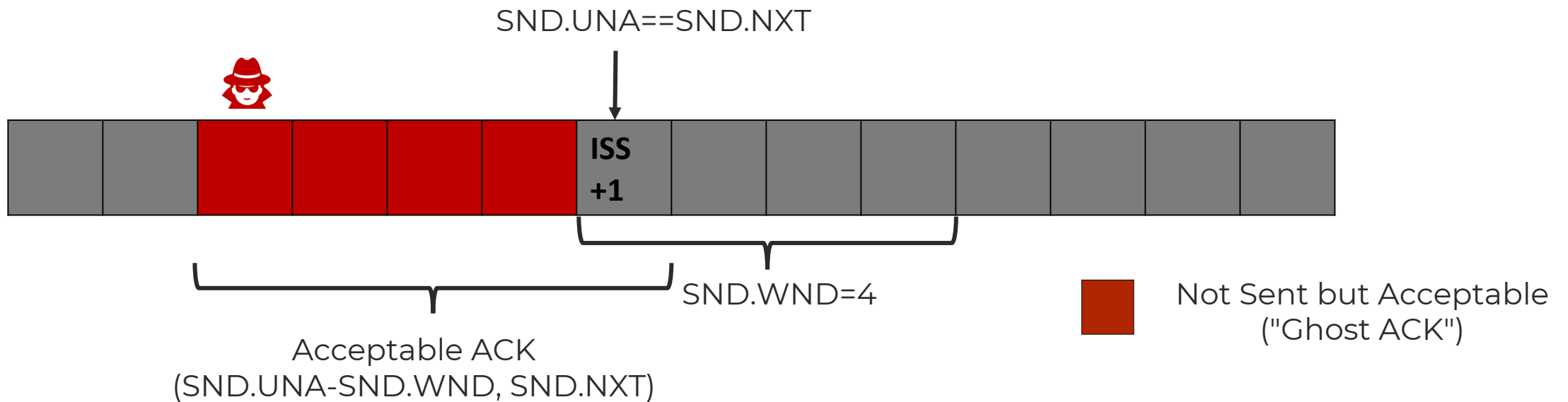
All incoming segments whose SEG.ACK value doesn't satisfy the above condition **MUST** be discarded and an SEG.ACK sent back.





# Ghost ACKs

- The current standards (incl. RFC 5961) do not explicitly treat duplicate ACKs that acknowledge data that was never sent ("Ghost ACKs")
- Standards implicitly interprets Ghost ACKs as "duplicate ACKs", as they fulfil:
  - RFC 5961:  $(\text{SND.UNA} - \text{MAX.SND.WND}) \leq \text{SEG.ACK} \leq \text{SND.UNA}$ , and
  - RFC 9293:  $\text{SEG.ACK} \leq \text{SND.UNA}$





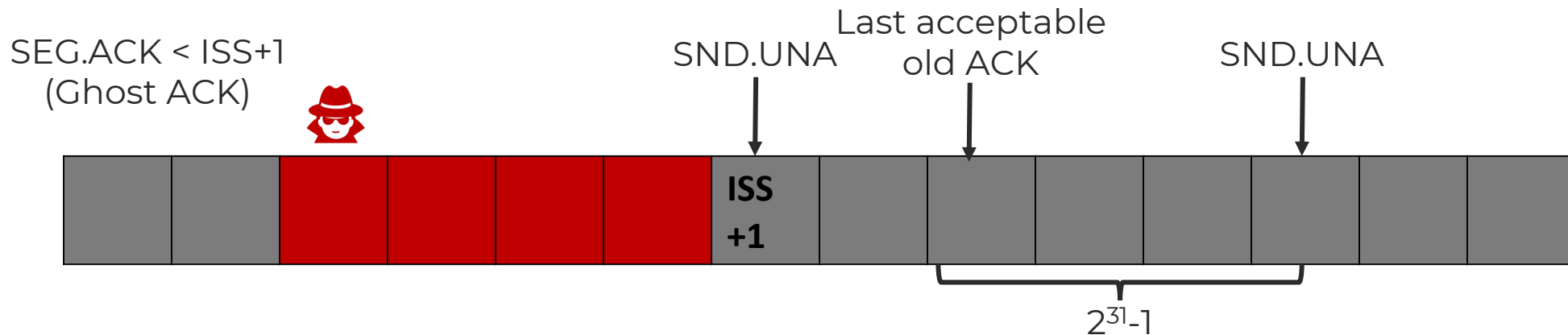
# Proposed Solution (Generic)

- An additional state variable `NO_ISS_CHECK` for each established connection is required to implement this mitigation:
- When validating the ACK value of any incoming segments, TCP stacks apply the following additional check:

$$\text{NO\_ISS\_CHECK} \parallel \text{SND.UNA} \leq \text{SEG.ACK} \parallel \text{ISS} + 1 \leq \text{SEG.ACK}$$

- When a connection is first established, `NO_ISS_CHECK` is initialized to False. Once the `SND.UNA` satisfies the following condition, `NO_ISS_CHECK` is set to true

$$\text{SND.UNA} \neq \text{ISS} \ \&\& \ \text{ISS} + 1 \leq \text{SND.UNA} - (2^{31}-1),$$

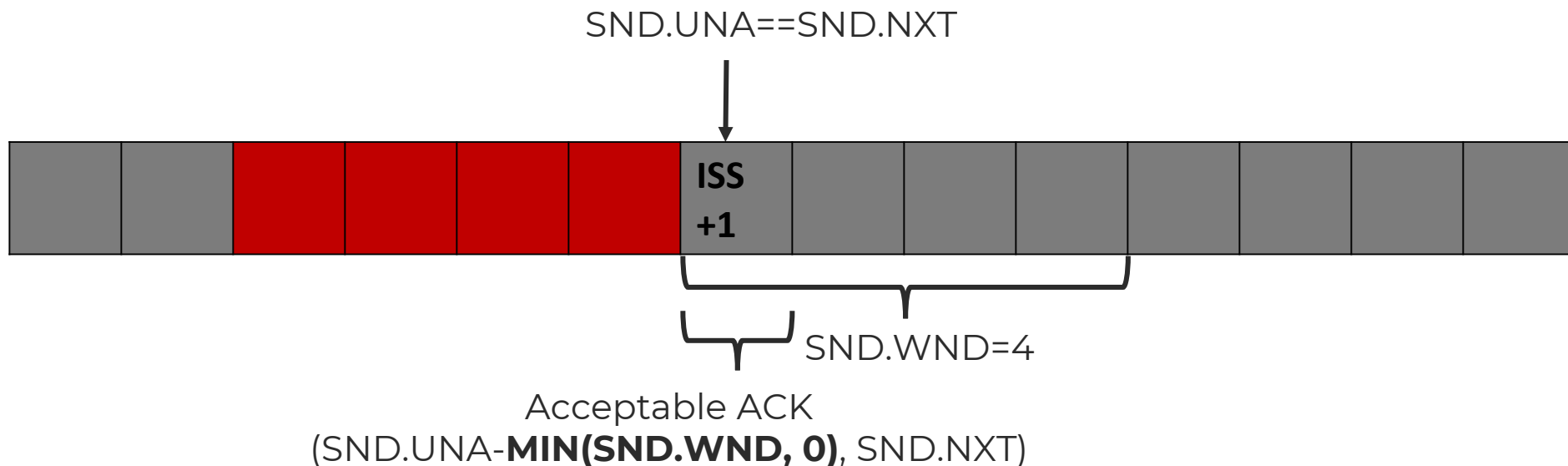




# Proposed Solution (need RFC 4898 implementation)

- RFC 4898 introduced the **tcpEStatsAppHCThruOctetsAcked** statistics, which tracks the number of bytes that is already acknowledged by the peer.
- TCP stacks implemented the RFC 4989 statistics can add the following check for incoming segments:

$$\text{SND.UNA} - \min(\text{MAX.SND.WND}, \text{tcpEStatsAppHCThruOctetsAcked}) \leq \text{SEG.ACK}$$





# Implementation Status

Linux has adopted the input check for Ghost Acks in (RFC 4898 specific solution) [1].

FreeBSD has also adopted the input check for Ghost ACKs recently (generic solution, with optimized conditions) [2]

[1] <https://lore.kernel.org/netdev/20231205161841.2702925-1-edumazet@google.com/T/#u>

[2] <https://reviews.freebsd.org/D45894>





## Next Steps

We kindly ask for your suggestions and help to improve and develop the current Internet- Draft.

- Leave it up to implementation?
- Add as an errata for RFC 9293/RFC 5961?
- Require RFC 5961 support as a prerequisite?
- Make it a short RFC?