Transport Layer Reneging

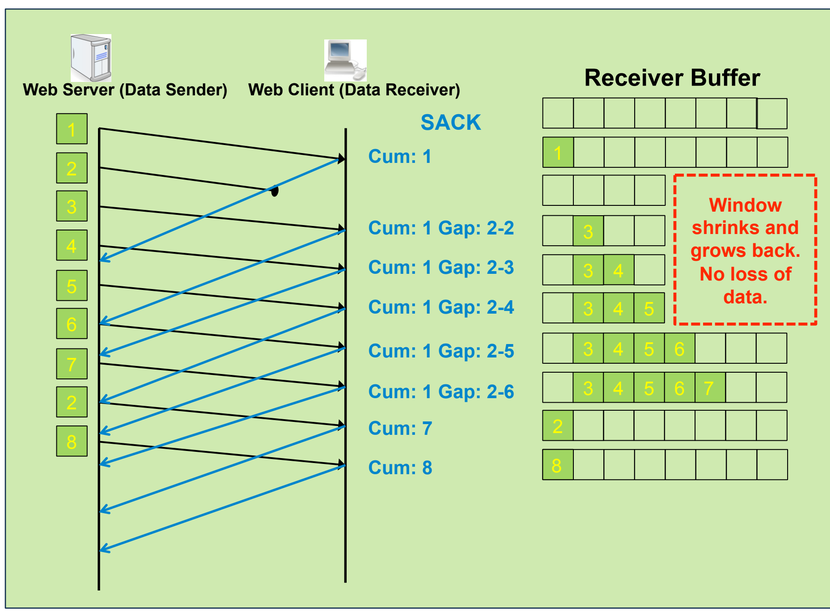
Prof. Paul Amer, Nasif Ekiz, Jonathan Leighton,
Ertugrul Yilmaz, Aasheesh Koli,
Ersin Ozkan, Fred Baker (Cisco)

Protocol ● Engineering ● Laboratory
University of Delaware

Definition: Shrinking the Window

- RFC 793 (TCP):
  - The mechanisms provided allow a TCP receiver to advertise a large window, and to subsequently advertise a much smaller window without having accepted that much data. This action is called shrinking the window, and is strongly discouraged.

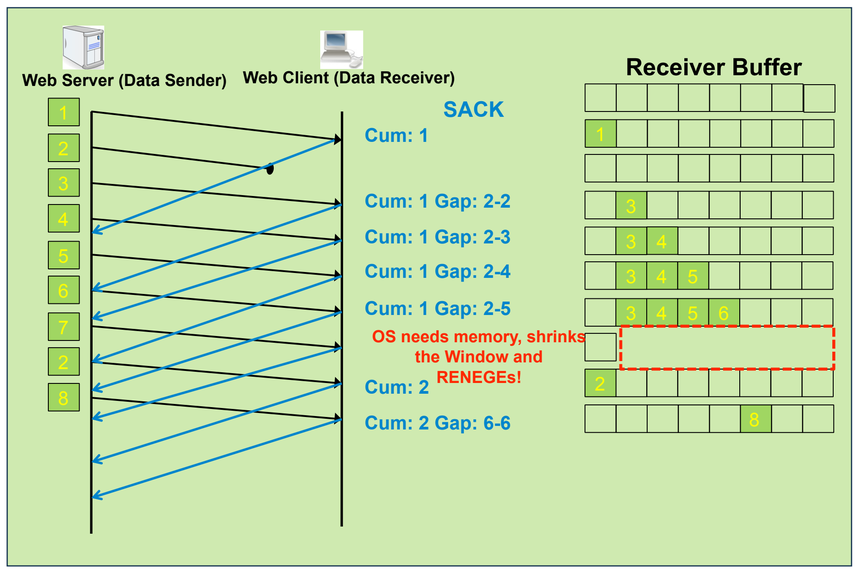
Example: Shrinking the Window



Definition: Data Reneging

- RFC 2018 (SACK):
  - The SACK option is advisory, in that, while it notifies a data sender that the data receiver has received the indicated segments, the data receiver is permitted to later discard data which have been reported in a SACK.
- If no reneging happens, the SACKed data is kept wastefully in the send buffer of the data sender until cumulatively acknowledged.

Example: Data Reneging



References

[1] P. Natarajan, P. Amer, E. Yilmaz, R. Stewart, J. Iyengar. "Non-Renegable Selective Acks (NR-SACKs) for SCTP," IETF Internet Draft, draft-natarajan-tsvwg-sctp-nrsack

[2] P. Natarajan, N. Ekiz, E. Yilmaz, P. Amer, J. Iyengar, R. Stewart. "Non-renegable selective acks (NR-SACKs) for SCTP," Int'l Conf on Network Protocols (ICNP), Orlando, 10/08

[3] E. Yilmaz, P. Natarajan, N. Ekiz, P. Amer, F. Baker. "Performance analysis of SCTP Non-Renegable Selective Acks (NR-SACKs)", (in progress)

[4] S. Jaiswal, G. Iannaccone, C. Diot, J. Kurose, D. Towsley. "Inferring TCP Connection Characteristics Through Passive Measurements", IEEE INFOCOMM, 3/04

[5] J.Padhye, S. Floyd. "On Inferring TCP Behavior", ACM SIGCOMM, 8/01

[6] http://www.caida.org/data/passive/passive_2009_dataset.xml

[7] Net-SNMP - http://net-snmp.sourceforge.net/

[8] Network Mapper (nmap) - http://nmap.org/

Motivation

- Previously, we proposed an extension to SCTP protocol --- Non-Renegable SACK (NR-SACK) [1].
- NR-SACKs better utilize a data sender's retransmission queue, and improve end-to-end throughput [2, 3].
- If the proposed research shows that RENEGING never or rarely occurs, NR-SACK extension could be set as the default ack mechanism for reliable transport protocols.

Research Questions

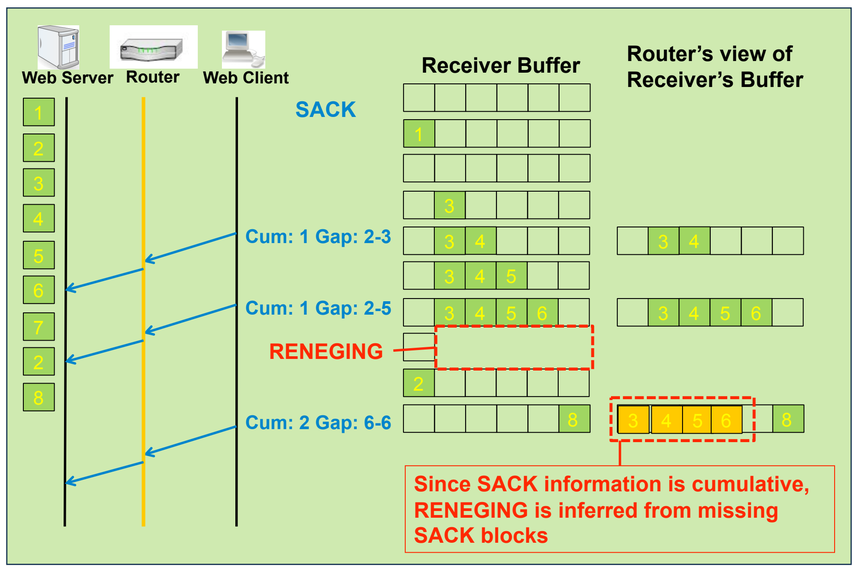
- Does RENEGING actually occur in today's Internet, and if so, how often?
- Can we characterize the circumstances when RENEGING occurs?
- Which major operating systems have a built-in mechanism allowing a transport connection to RENEGE?
- Can we develop a tool forces a remote peer to RENEGE?

Proposed Research

- Proposed solution:
  1. To investigate RENEGING using Internet's passive measurements as in [4].
  2. To examine the implementations of transport layer protocol stacks such as TCP and SCTP in the major open source operating systems to detect built-in RENEGING mechanisms.
  3. To extend TCP and SCTP SNMP agents in FreeBSD to report RENEGING instances.
  4. To implement a tool like TBIT [5] which can generate user-specified sequence of transport PDUs to make a remote host RENEGE.

1. Internet's Passive Measurements

- In [4], authors introduced a passive measurement technique to infer and keep track of cwnd and RTT of a TCP connection.
- We have anonymized Internet traffic traces from CAIDA [6] which monitors 50+ million TCP connections.
- We are developing RENEG-INFER, a tool to keep track of the data receiver's receive buffer's state on TCP connections supporting SACK, and identifies RENEGING instances.



2. OS's that RENEGE

- To better understand why an operating system would RENEGE, we will examine the stack implementations of TCP and SCTP in open source operating systems: Linux, FreeBSD, Mac OS.
- We will try to contact implementer's of Windows networking code, and inquiry about Microsoft's support for RENEGING.
- Below you will find the Linux's TCP function invoked to clean main memory used by out-of-order data in case of RENEGING.
- Result of the investigation will be a summary table as below:

Operating System	Popularity	Support for RFC 2018	Support for RENEGING
Windows XP	62.85%		
Windows Vista	23.42%		
Mac OS X 10.5	6.00%	Yes	Yes
Mac OS X 10.4	2.66%	Yes	Yes
Linux	0.90%	Yes	Yes
iPhone	0.49%		
Android 1.1	0.06%		
FreeBSD	0.01%	Yes	Yes

Example: Reneging Code

```
/*
 * Purge the out-of-order queue.
 * Return true if queue was pruned.
 */
static int tcp_prune_ofo_queue(struct sock *sk)
{
    struct tcp_sock *tp = tcp_sk(sk);
    int res = 0;

    if (!skb_queue_empty(&tp->out_of_order_queue)) {
        NET_INC_STATS_BH(sock_net(sk), LINUX_MIB_OFOPRUNED);
        __skb_queue_purge(&tp->out_of_order_queue);

        /* Reset SACK state.  A conforming SACK implementation will
         * do the same at a timeout based retransmit.  When a connection
         * is in a sad state like this, we care only about integrity
         * of the connection not performance.
         */
        if (tp->rx_opt.sack_ok)
            tcp_sack_reset(&tp->rx_opt);
        sk_mem_reclaim(sk);
        res = 1;
    }
    return res;
}
```

3. Detecting Reneging via SNMP

- Simple Network Management Protocol (SNMP) is one of the most widely used protocols for managing network entities.
- Operating systems such as FreeBSD and Mac OS provide mechanisms to RENEGE on out-of-order data when the kernel state **net.inet.tcp.do_tcpdrain** is enabled.
- TCP and SCTP SNMP agents (in net-snmp [7]) for those operating systems can be extended to report RENEGING instances.

4. RENEGING Tool

- We will try to implement RENEG-CAUSE, a tool similar to TBIT [5] which will send a specific sequence of PDUs to a remote TCP (or SCTP) end point to make the remote host RENEGE.
- A tool such as nmap [8] can be used to determine the remote operating system which then can be used to specify the sequence of TCP (or SCTP) PDUs to be sent.
- TCP (or SCTP) PDUs will be generated at user level using the raw IP sockets.

Sponsors: U.S. ARMY RESEARCH LABORATORY, CISCO