VFER
High Performance Transport in User Space

Ivan Beschastnikh Stanislav Shalunov

University of Washington Internet2

“10 Gbps or even 1 Gbps data rates are still unattainable by most users”

- TCP is difficult to tune for high performance
- Alternatives to Reno **require** kernel modifications
  - Require root access
  - Can bring down the system
  - Difficult to install and difficult to experiment with
  - Most users stay with default transport

TCP's Rate Equation

- Loss-based congestion control **ignores** information

TCP Rate <= (MSS/RTT)(1 / sqrt{p})*

- *Rate*: is the TCP transfer rate or throughput
- *MSS*: is the maximum segment size (fixed for each Internet path, typically 1460 bytes)
- *RTT*: is the round trip time (as measured by TCP)
- *p*: is the packet loss rate.

Our Goals

1. Give normal users the ability to use something other than the default TCP stack on their machine

2. Optimize for High Bandwidth transfers out of the box- unlike TCP

VFER – Bridging the Hero Gap

- **Easy to install and configure**
  - Open-source user-space transport tool
  - No kernel modifications nor root access
  - Portable
- Two ways to use
  - Library with a socket-like API
  - File transfer tool
- Advanced congestion control
  - Performs better than TCP out of the box
  - Delay-based and TCP-friendly
- Security layer – if necessary

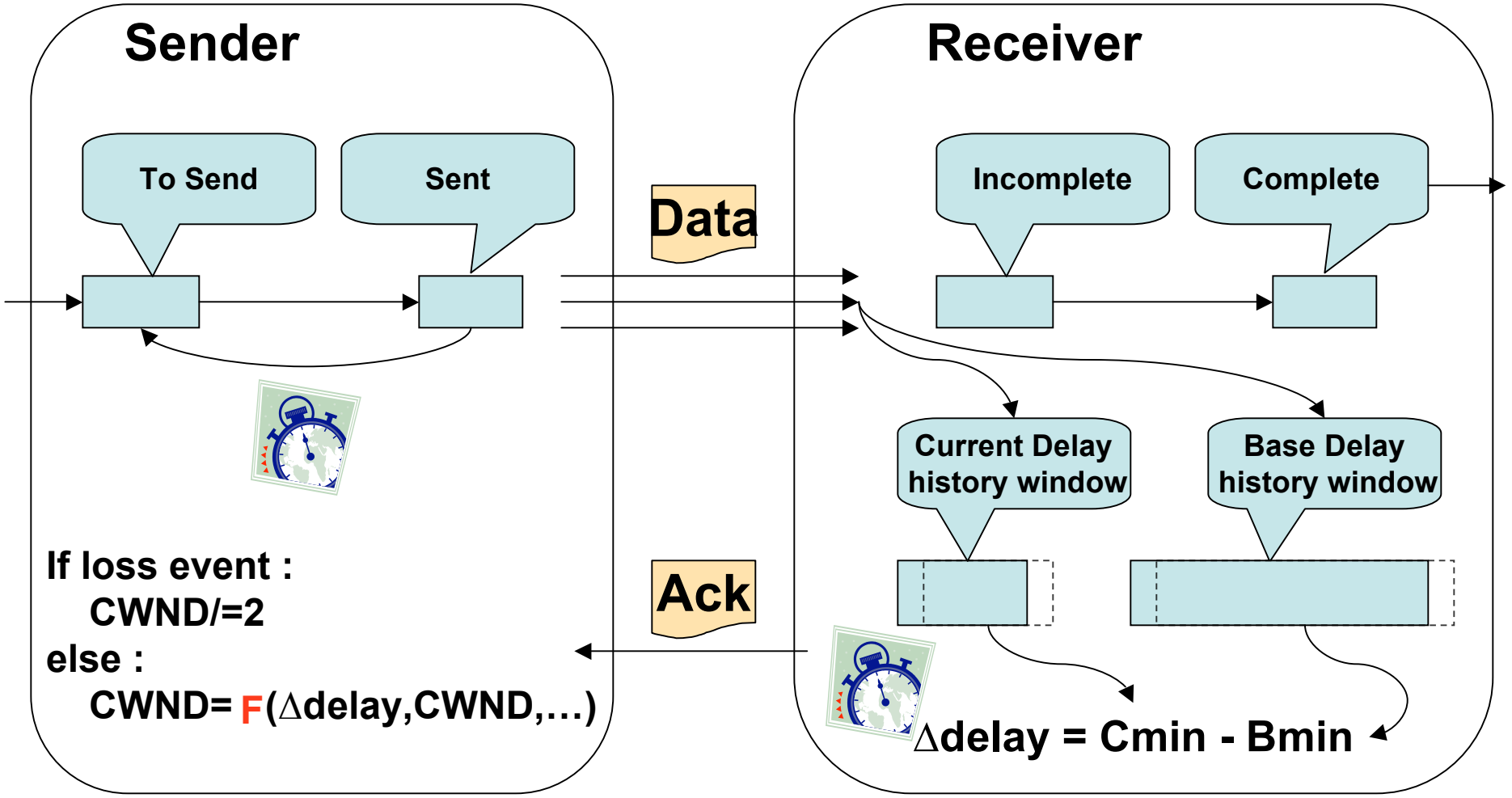
Protocol Features

- In-order reliable delivery of variable-sized objects
- Connection-oriented
- Congestion control uses delay **and** loss
- Path MTU discovery

Congestion Control

- Respond to loss events like TCP
- Observe one-way delay
  - Estimate queuing delay
  - Need statistical preprocessing
  - Maintain and process history at receiver
  - Periodically communicate to sender
- Drive queue length to target

Congestion Control Details



Timekeeping

- Accurate time stamping with the TSC-I2 library
- User space daemon to sample TSC-register values
- Faster than gettimeofday()
- Recalibration of values for accuracy

Security

- Transferred information might be sensitive or must be kept private by users
- VFER implements secure sockets
  - Uses existing SSH credentials
  - Poly1305-AES MAC to ensure msg integrity
  - Diffie-Hellman key exchange

Results

- Coming into use by
  - The electronic-Very Long Baseline Interferometry (e-VLBI) community
  - The Visible Human Project (VHP)

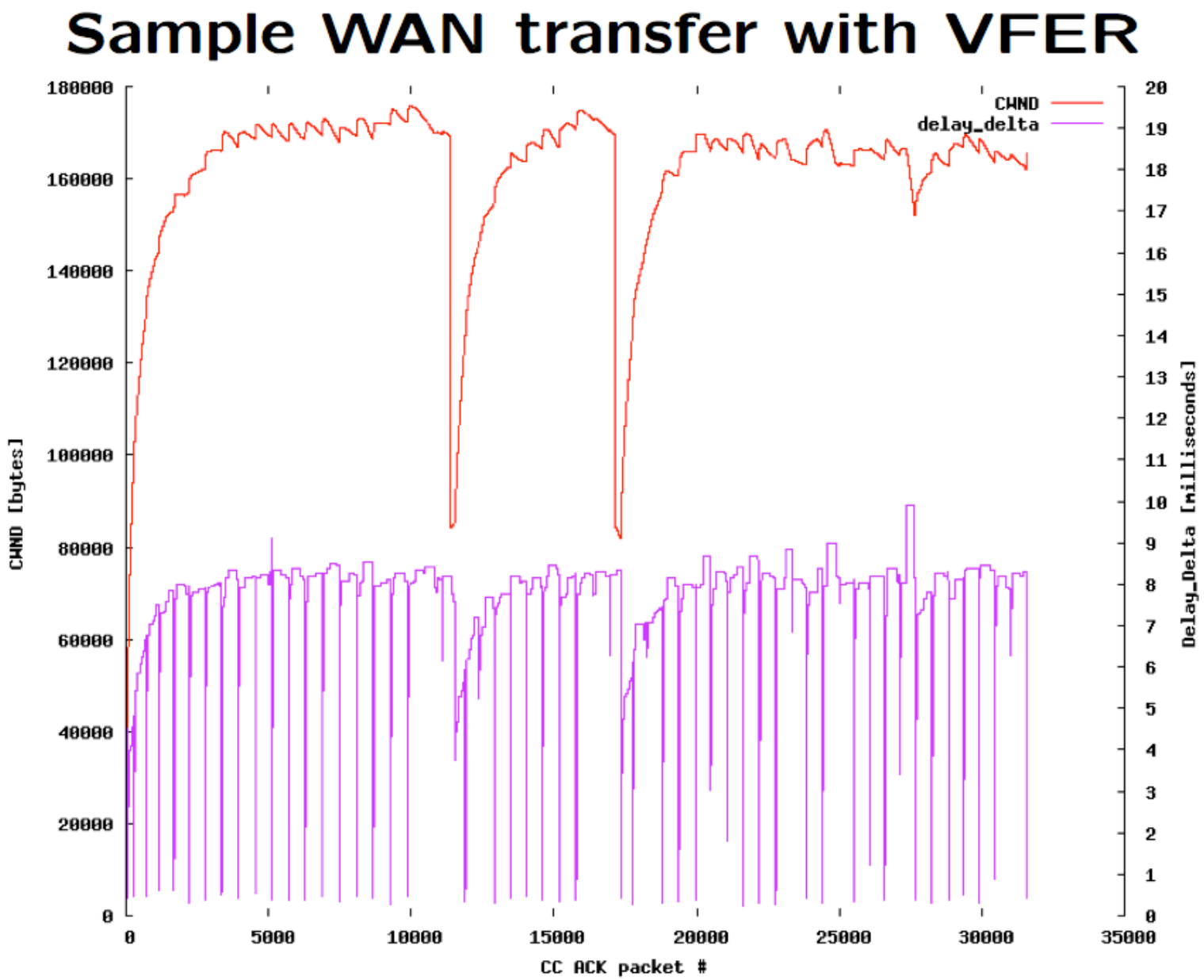
Summary

- User-space implementation
  - **Easy to install and use by average users**
  - Library with a familiar API
  - File transfer tool

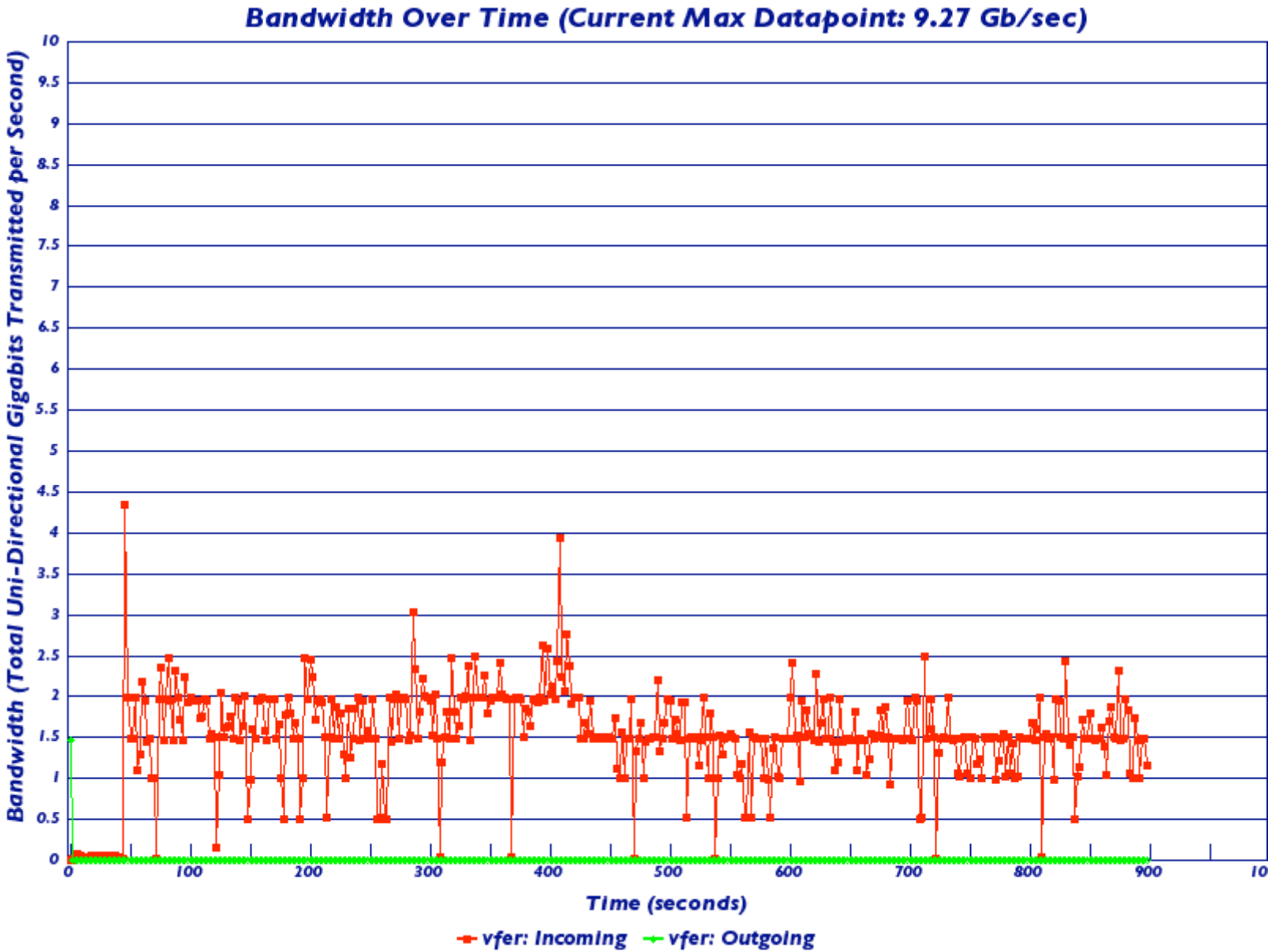
- Advanced congestion control

http://vfer.internet2.edu

VFER


Sample WAN transfer with VFER
CWND
delay_delta
CWND [bytes]
Delay_Delta [milliseconds]
CC ACK packet #
0
20000
40000
60000
80000
100000
120000
140000
160000
180000
0
1
2
3
4
5
6
7
8
9
10
11
12
13
14
15
16
17
18
19
20
0
5000
10000
15000
20000
25000
30000
35000

VFER


Bandwidth Over Time (Current Max Datapoint: 9.27 Gb/sec)
Bandwidth (Total Uni-Directional Gigabits Transmitted per Second)
10
9.5
9
8.5
8
7.5
7
6.5
6
5.5
5
4.5
4
3.5
3
2.5
2
1.5
1
0.5
0
0
100
200
300
400
500
600
700
800
900
10
Time (seconds)
vfer: Incoming
vfer: Outgoing