Verification of Zebra as a BGP Measurement Instrument

Hongwei Kong

Agilent Labs, China

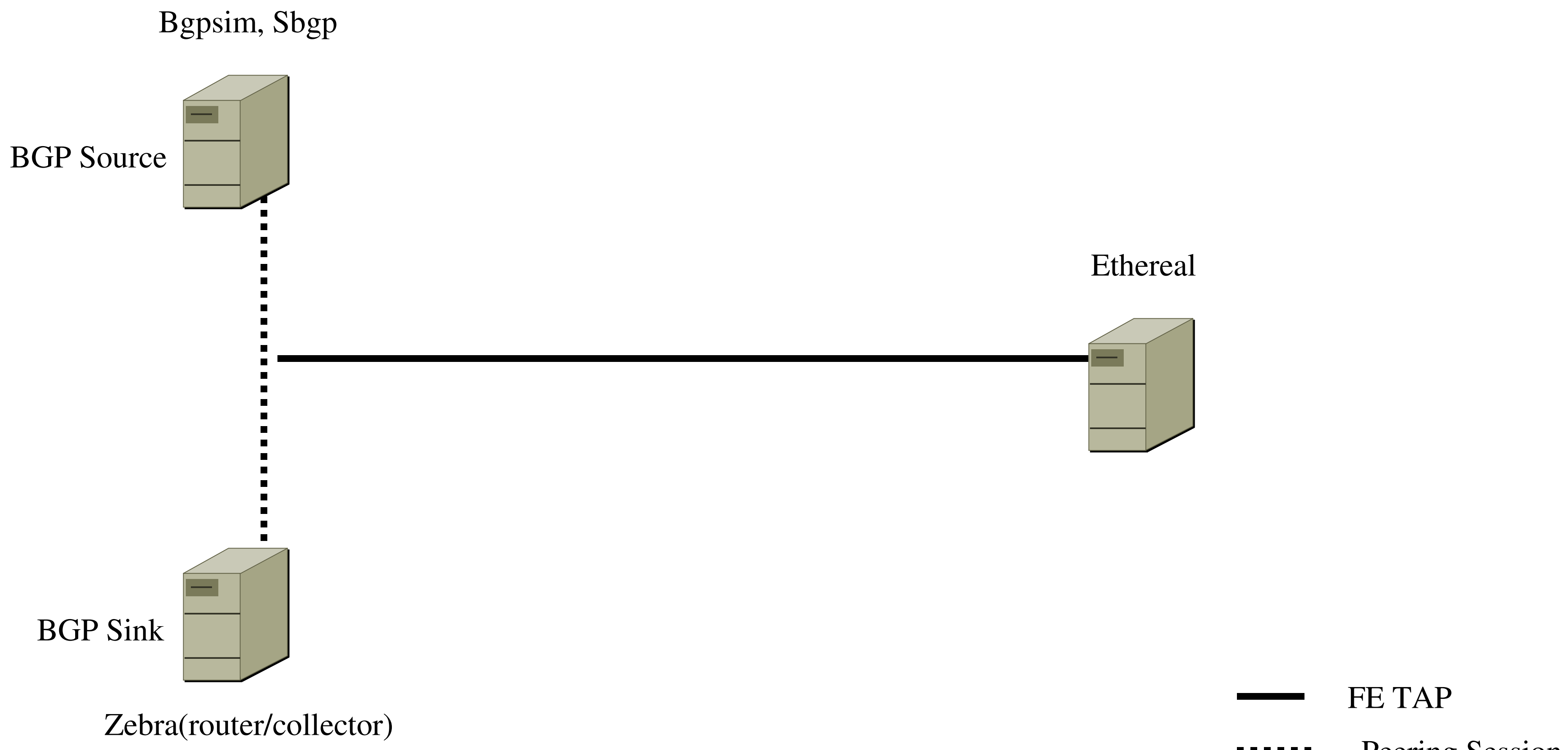
Should You Believe What You See

- Zebra is in use at RIPE and Oregon RouteViews as a BGP message recorder

- We, the research community, have been using BGP data recorded by Zebra for analysis of BGP behavior for several years now

- How good is the Zebra Data?

Our Method to Test for Truth



First We Verify route_btoa

- Method: Send known BGP data across wire. Record-Decode-Verify
  - Tested on Linux and Solaris with different results
- **Here’s Why**
- When multi-protocol NLRI reachable/unreachable attribute present for IPv6 prefixes route_btoa cannot decode correctly
  - Interesting these messages were only observed on rrc03 (AMS-IX).
  - route_btoa can support this but support tied to capabilities of the kernel during compilation. Checks for kernel IPv6 support.
- When multi-protocol NLRI reachable/unreachable attribute present for IPv4 multicast prefixes route_btoa cannot decode correctly
  - Interesting we didn’t see any of these on any of the RIPE systems
  - Turns out route_btoa does support this, but it is tied to capabilities of the kernel during compilation. Checks for kernel multicast routing support

While Verifying route_btoa We Found A Couple of Odd Things With Zebra…

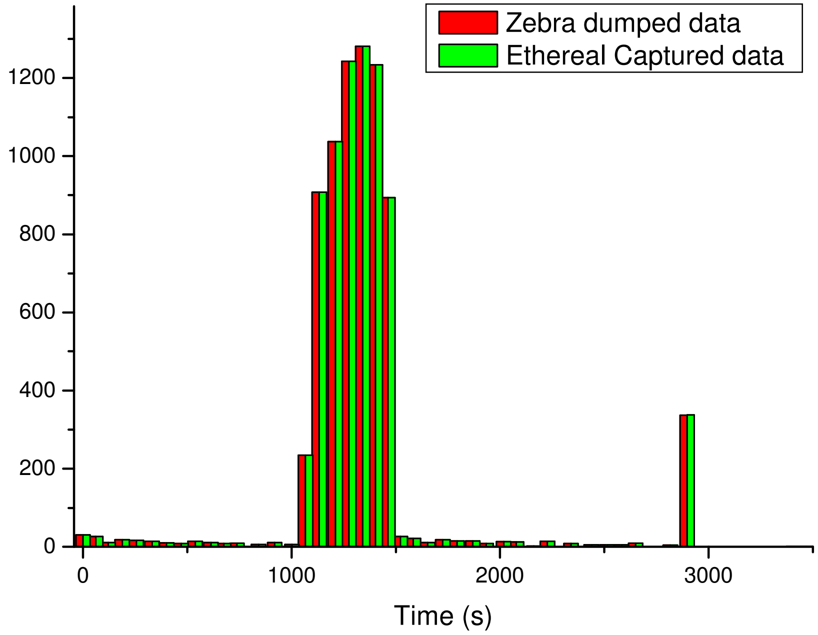
- Some, but not all BGP “OPEN” messages are saved by Zebra in an alternative format, a format not recognized by route_btoa- reason is unknown
    - This does not occur on Zebra-to-Zebra sessions, but does occur on Zebra-to-bgpsim and Zebra-to-sbgp sessions. Observed in RIPE data.
    - AS and IP addresses, both source & destination are recorded as 0. Causes route_btoa to decode message as NULL.
- Some BGP messages are recorded with all zero source & destination Addresses. These turn to be Multiple protocol NLRI reachable/unreachable for IPv6 prefixes
- Some BGP messages are recorded with no source & destination Addresses. This can be due to old version of Zebra.

While Verifying route_btoa We Found A Couple of Odd Things With Zebra…

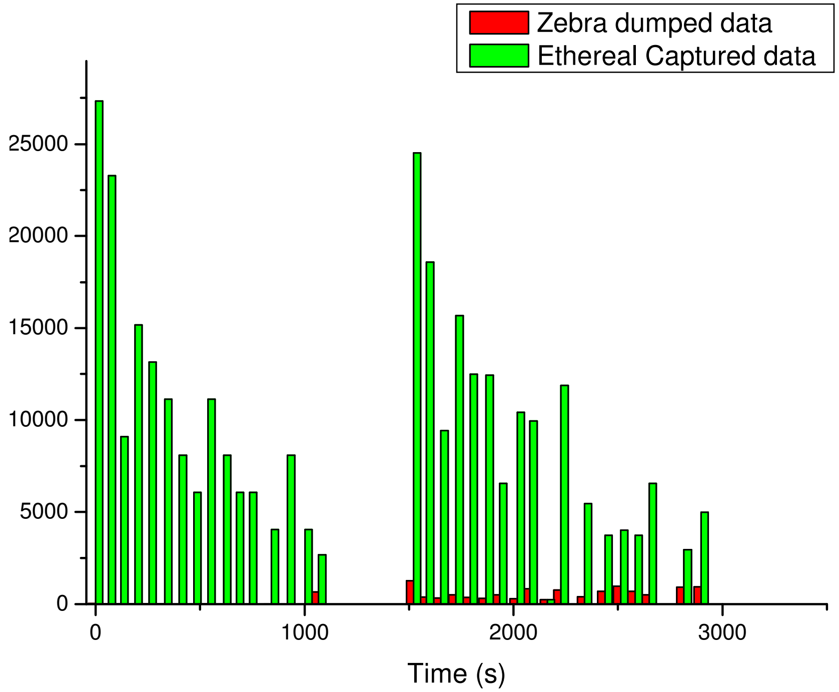
- **Very Large BGP Messages are Incompletely Saved by Zebra**
  - In fact, this happens with all messages with a length field of 4096 bytes
  - Zebra dump module buffer size is
    bgp-max-packet-size(4096Bytes) + bgp-dump-header-size(12bytes)
  - Zebra dump module does not take into account bgp-dump-message-header
    - Includes things like: source & destination AS, Interface index, Address Family, IP addresses
  - Zebra Bug Fixed by adding 40 bytes to buffer. Solved after the version after the quagga-0.96. (bug ID: 28)
- **Observation**
  - **Zebra should save in correct format and/or route_btoa should support irregular dump headers**

Loss of Prefixes Due to Incompletely Captured BGP Messages:

Count of Messages Recorded



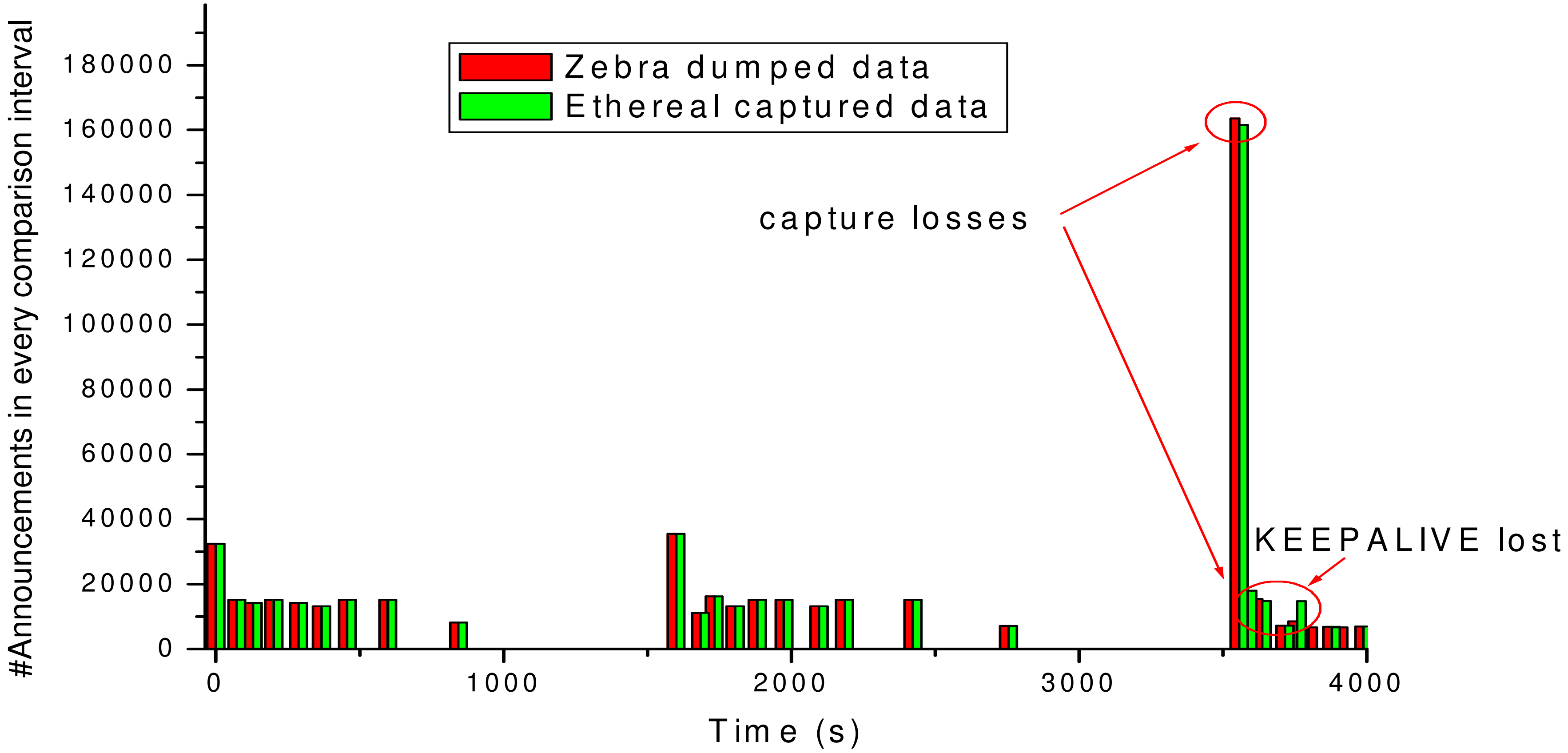
Count of Prefixes Recorded



Finding a Bug in the Ethereal BGP Dissector

- Using fixed version of Zebra we compared on-wire observations with Zebra dumps using a known data stream
- Zebra matched the known stream but data obtained from Ethereal using the Ethereal BGP Dissector contained fewer Announcements than expected
- **Here's Why**
  - If a BGP message header spans two TCP segments then it is not recognized by BGP Dissector and is not decoded
- Bug reported and fixed in version 0.9.12 of Ethereal

Overcoming Limitations of libpcap



Agilent Technologies

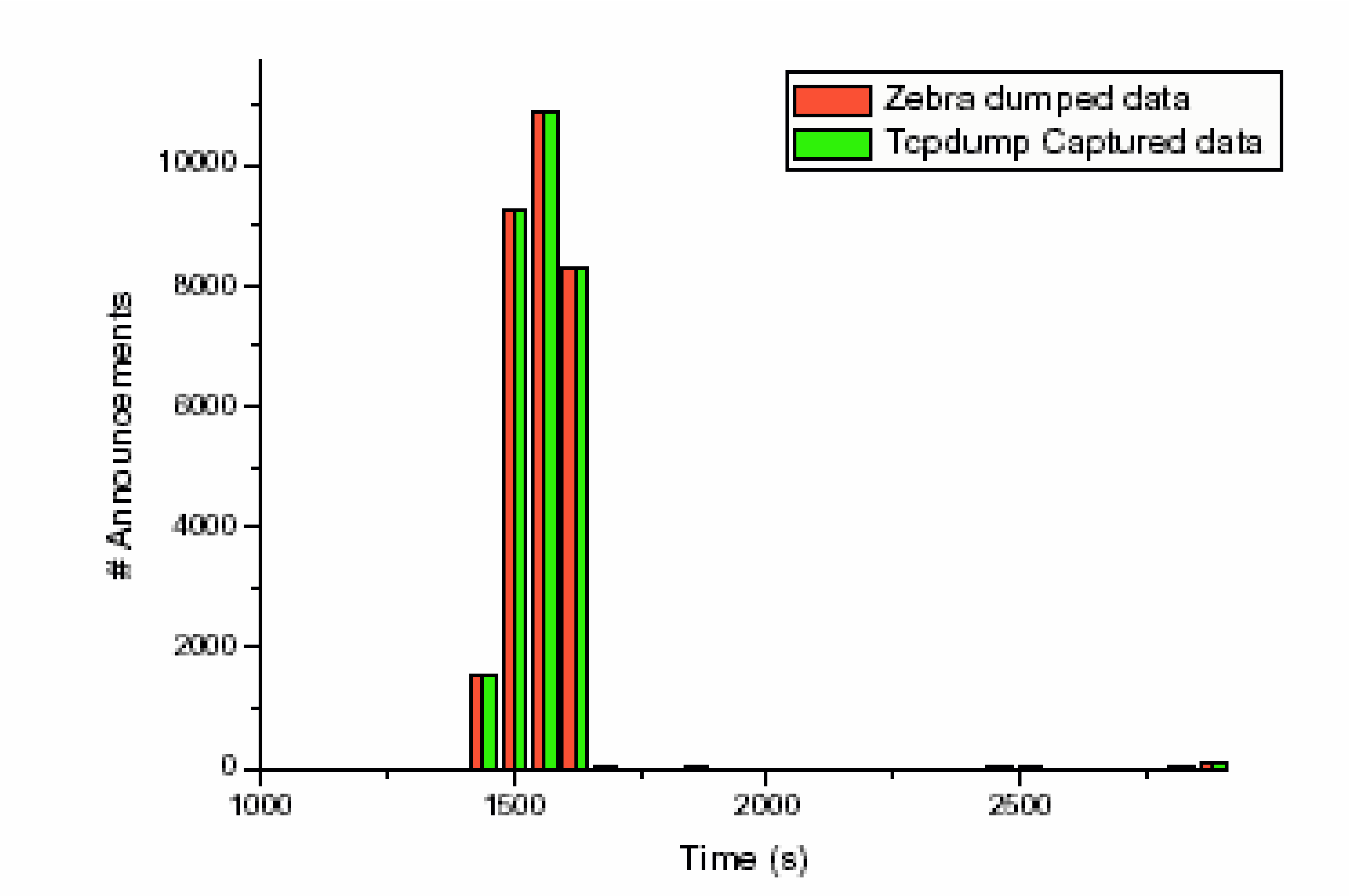
Overcoming Limitations of libpcap

- We observed losses in libpcap under heavy load
- **Here’s Why**
  - Queue overflows in libpcap ver.0.7.2
  - Libpcap 0.8.030314
    - allows network adapter to directly capture to system memory
    - Implements large ring queue in system memory
- Rebuilt Ethereal with libpcap 0.8.030314
  - All loss was eliminated

More Problems With BGP Dissector

- **Next we introduced TCP segment losses using NIST Net**
  - Using BGP Dissector to reconstruct the session we found "extra" BGP messages.
  - Problem was reported to Ethereal developers
  - As of Ethereal ver 0.9.12 problem is still not fixed
- **Consequences**
  - Pay attention particularly when evaluating multi-hop BGP sessions reconstructed using BGP Dissector

Finally They Match-Most Bugs Fixed, Others Avoided



Other Zebra Issues Of Concern to Researchers

•Timestamps don’t reflect on-the-wire times

- •Caused us to need to use keep-alives as synchronization markers

•Missed keep-alives

- •Causes session to break and retransmit of full table

•Records only inbound BGP messages

- •Miss outbound NOTIFICATION messages

•Sends NOTIFICATION messages which break session

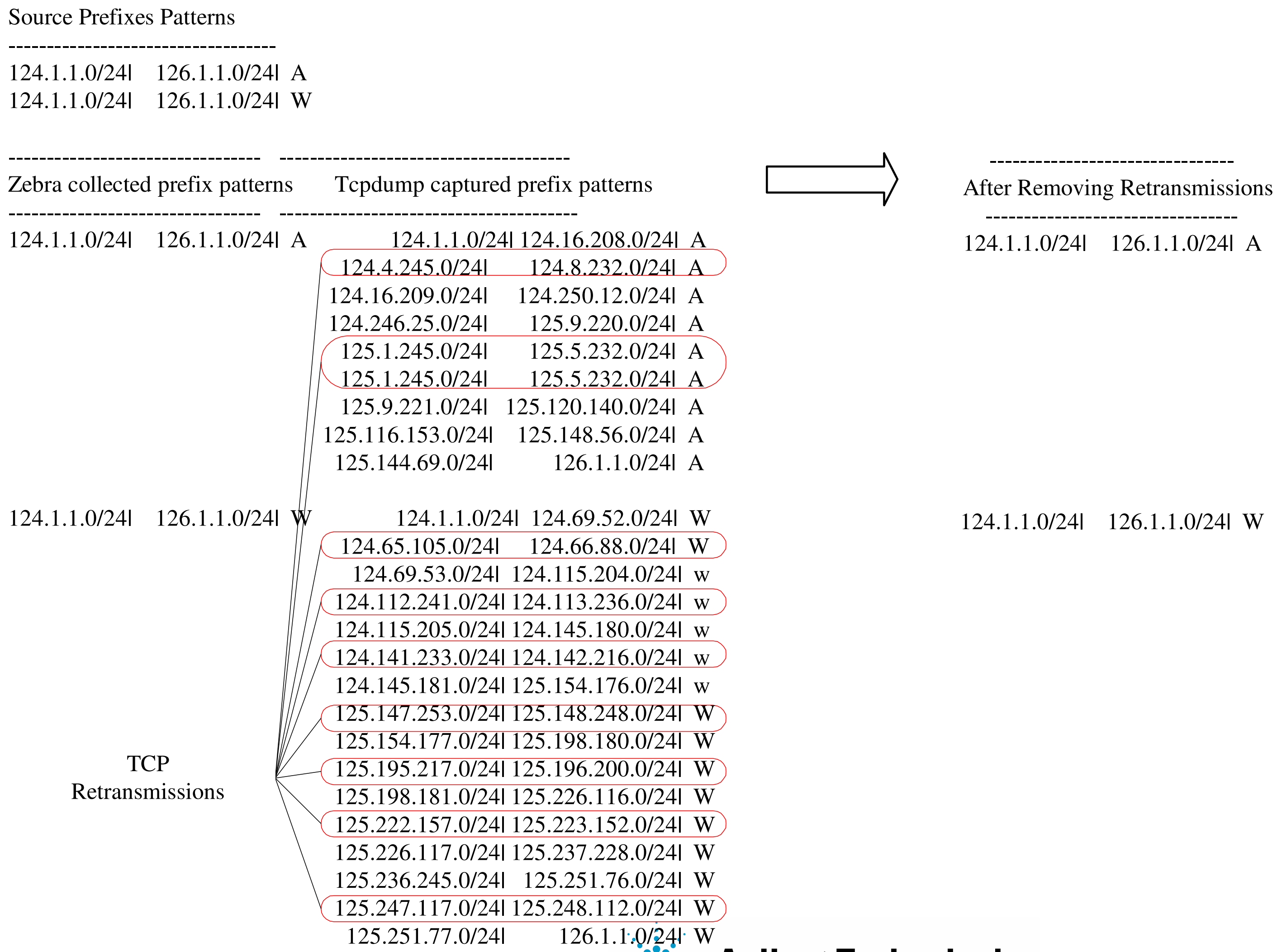
•10+ Second recording dead time after session reset

•Amount/complexity of code is overkill- only need a recorder

Summary

- **Verified the the behaviors of the tools used to process Zebra BGP data files.**
  - revised these tools and solved the problems found
    - http://www.ris.ripe.net/source/libbgpdump-1.4-rc1.tar.gz
- **Explored the consistency of Zebra BGP data collections**
  - Found bugs in Zebra
- **Verified Zebra BGP data collecting module**
  - Without BGP session break, Zebra collects BGP data consistently
  - During session break, Zebra BGP data may not be consistent with on-wire captured data
  - Zebra can delay sending KEEPALIVE messages to the peer when there is heavy BGP traffic and result in session break and corrupted data.
  - Zebra Data capturing is delayed when there is heavy BGP traffic

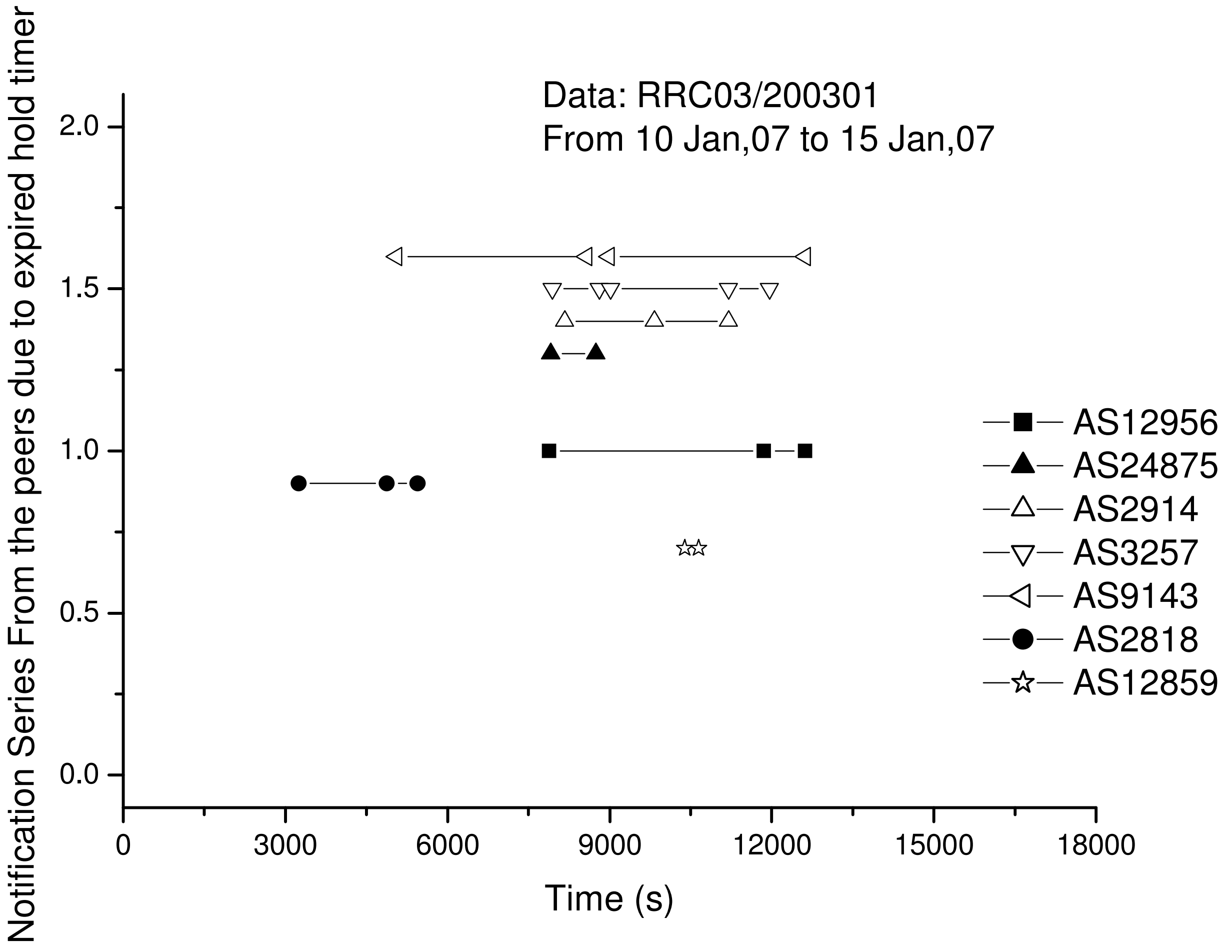
Tethereal decoding problem due to retransmitted TCP segments



```
Source Prefixes Patterns
-----------------------------------
124.1.1.0/24|    126.1.1.0/24|  A
124.1.1.0/24|    126.1.1.0/24|  W

---------------------------------   ---------------------------------------
Zebra collected prefix patterns       Tcpdump captured prefix patterns
---------------------------------   ---------------------------------------
124.1.1.0/24|    126.1.1.0/24|  A          124.1.1.0/24| 124.16.208.0/24|  A
                                        124.4.245.0/24|       124.8.232.0/24|  A
                                       124.16.209.0/24|     124.250.12.0/24|  A
                                       124.246.25.0/24|       125.9.220.0/24|  A
                                        125.1.245.0/24|       125.5.232.0/24|  A
                                        125.1.245.0/24|       125.5.232.0/24|  A
                                        125.9.221.0/24|   125.120.140.0/24|  A
                                      125.116.153.0/24|     125.148.56.0/24|  A
                                       125.144.69.0/24|          126.1.1.0/24|  A

124.1.1.0/24|    126.1.1.0/24|  W          124.1.1.0/24|  124.69.52.0/24|  W
                                        124.65.105.0/24|      124.66.88.0/24|  W
                                          124.69.53.0/24|  124.115.204.0/24|  w
                                       124.112.241.0/24| 124.113.236.0/24|  w
                                       124.115.205.0/24| 124.145.180.0/24|  w
                                       124.141.233.0/24| 124.142.216.0/24|  w
                                       124.145.181.0/24| 125.154.176.0/24|  w
                                       125.147.253.0/24| 125.148.248.0/24|  W
                                       125.154.177.0/24| 125.198.180.0/24|  W
         TCP                           125.195.217.0/24| 125.196.200.0/24|  W
   Retransmissions                     125.198.181.0/24| 125.226.116.0/24|  W
                                       125.222.157.0/24| 125.223.152.0/24|  W
                                       125.226.117.0/24| 125.237.228.0/24|  W
                                       125.236.245.0/24|   125.251.76.0/24|  W
                                       125.247.117.0/24| 125.248.112.0/24|  W
                                         125.251.77.0/24|        126.1.1.0/24|  W
```

```
--------------------------------
After Removing Retransmissions
---------------------------------
124.1.1.0/24|    126.1.1.0/24|  A


124.1.1.0/24|    126.1.1.0/24|  W
```

Session break problem of Zebra due to missed keepalive messages



Agilent Technologies