Verifying & Measuring Network Recovery Performance

Purpose

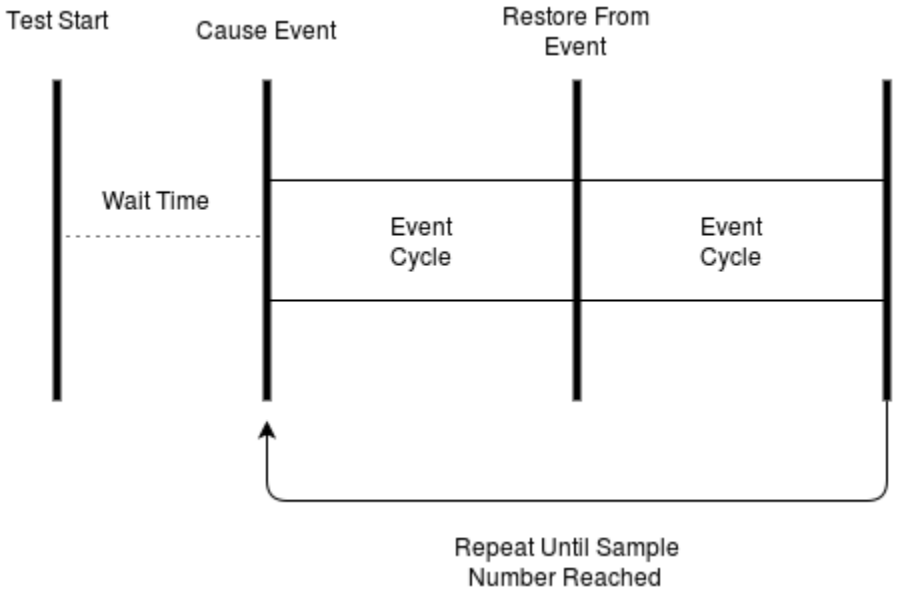
The traffic on Ethernet networks can be disrupted due to cable or device failures; how fast the traffic flow is restored is very important to our industry. To verify the recovery speeds of our network devices after such an event, we need to be able to measure recovery times accurately. To obtain this data, we have designed a set of testing apparatus that has three parts: a packet stream sender, a packet stream receiver, and an event triggering device. By placing the sender at a point in a network, a user can generate traffic flowing in from that network port in a consistent and measurable way. Placing the receiver in a different port in the network, and using the event triggering device to cause a network event (or to restore the network from such an event), allows the test tools to measure how long it takes for the network to recover.

The apparatus also has a mode called "justwatch" which allows a user to set up the test tools to simply watch the test stream and record any disturbances. This is talked about in more depth below.

System Behavior

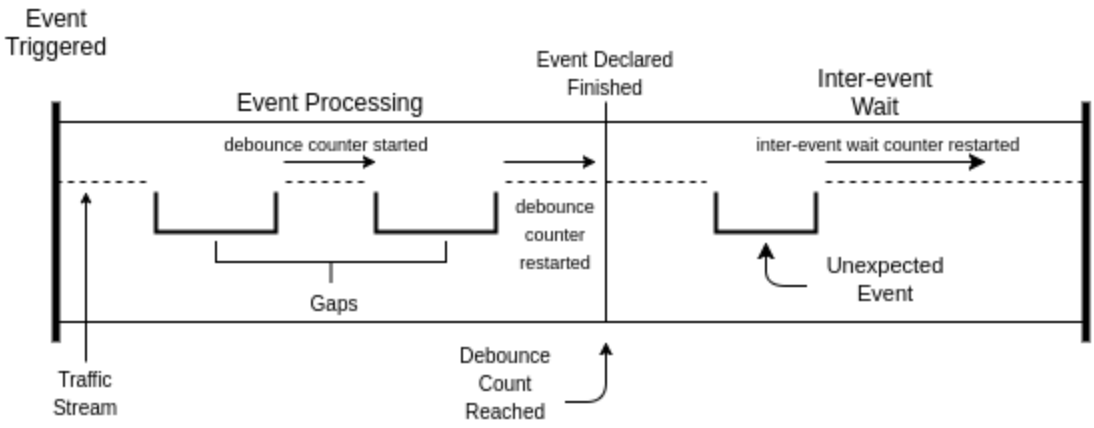
To test the reliability of our devices' recovery speeds we run long series of automated network event tests.

The tests consist of two event types: a Link-Down Event (that is, a connection is broken or device powered off) and a Link-Up Event (the connection is restored or device powered on). Together, these create a sample.



If the test has just started, the receiver will flip the state of the event triggering device to the "connected" state and wait the specified amount of time. For link-break tests we give the network time to settle, and for power-cycling tests we give the switch of interest time to reboot and reintegrate itself with the network.

Each event cycle is made up of several parts that can be tweaked by the user which are as follows:



The following section details the diagrams above:

- The event triggering device is triggered to cause the network event. This is controlled by the receiver, and may disturb the stream of packets to certain network devices.
- The **debounce** period begins as soon as the receiver sees a disturbance after the event has been triggered/recovered from. This is where the receiver starts counting packets until the number of sequential packets without a disturbance is equal to the specified debounce number. If an additional disturbance *does* happen during the debounce period, the counter is reset and started again. These disturbances are referred to as **gaps**, and the act of starting the counter over is called a bounce. We get this name from the terminology used with electromechanical contacts that will physically bounce when actuated.

  It should also be noted that other non-typical traffic patterns can be considered disturbances. These disturbances include *duplicate packets* and *out-of-order packets*. When either of these appear, the debounce counter is reset like it is with the gaps described above and the type of disturbance is logged in the summary.
- Once the debounce period has finished, the event is considered done, and the **inter-event wait time** begins. This is another counter, waiting for a specified amount of time and giving the devices involved in the test time to do any additional house-keeping they may need to do, like send/record logs. Causing events too close together can negatively back up some of these processes. If a network event happens during this time, it is considered an **unexpected event** and logged as such. Unexpected events are still considered full events, meaning that the system will act as though the new event has happened and be placed back in "event processing" mode. It will then process a proper event, and then return to the inter-event wait period again.

One thing to note is: depending on where the sender, receiver, and event triggering device are in relation to each other on the network is it possible that the triggered network event will not disrupt traffic between the sender and receiver. This can also happen because the network may heal very quickly (faster than 1/4 ms). When this happens, it is possible for the system to hang because the receiver will not have perceived failure in the network and will be waiting until it sees a disturbance that will never happen. In order to avoid this, there is an option that may be enabled called **Trust**, which allows unmeasureable events to still be considered events. With Trust enabled the system will trigger the event and wait to detect an event, but if an event never is detected (after a period of time) the event will be marked as a "non-event" and the system will continue.

The Sender

The creation of a consistent, measurable stream is handled by the sender. It sends individual sequentially-numbered packets out of its ethernet port at a rate of 1 per quarter-millisecond (or 4000 per second). Knowing this, the receiver can measure system events in terms of time by looking at the last-received packet's number and comparing it to the newly-received packet's number and then translating the number of lost packets into time.

NOTE: The packet stream being sent is a Layer 2 multicast packet, with a GOOSE ethertype and a VLAN of 5 (by default; The VLAN can be changed). There are plans to make a unicast version of the tool. For more information on the data stream and it's accuracy see the Testing Equipment Accuracy page. The packet stream sent by the sender must be able to be seen by the receiver on the network or the test will not work.

The Receiver

The second part of the tool is more complex because it performs multiple functions. The receiver handles: recording the packet numbers at they pass into it, controlling the link breaker/power cycler, calculating the time of network events, and finally presenting the findings in several understandable formats.

The results produced by the receiver are presented in three different ways:

- In a text-based format, with each of the different stats presented numerically.
  - Text-Based Example

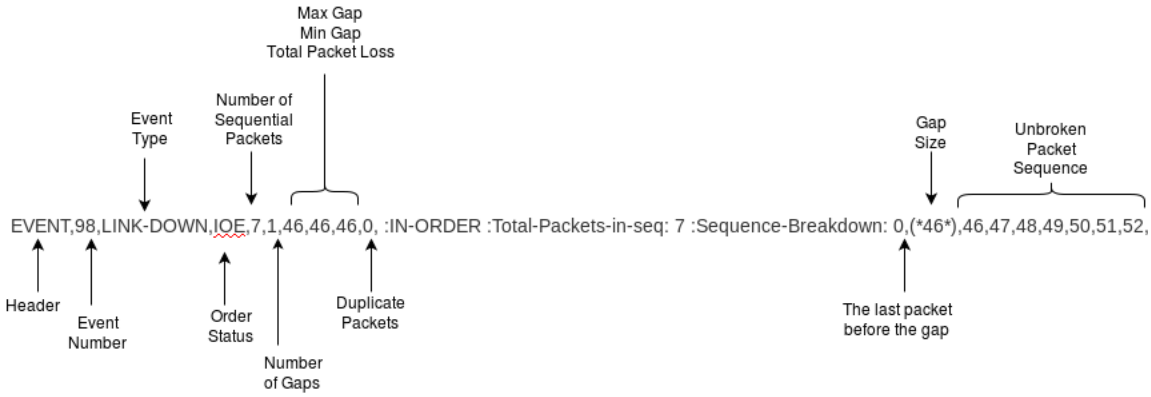
    The stats presented are made up of three parts that are as follows:

Individual Lines

The data in the results is laid out in (mostly) CSV format, and have an indicator of which variety they are as their first item. For example: HEADER, EVENT, INFO, etc.

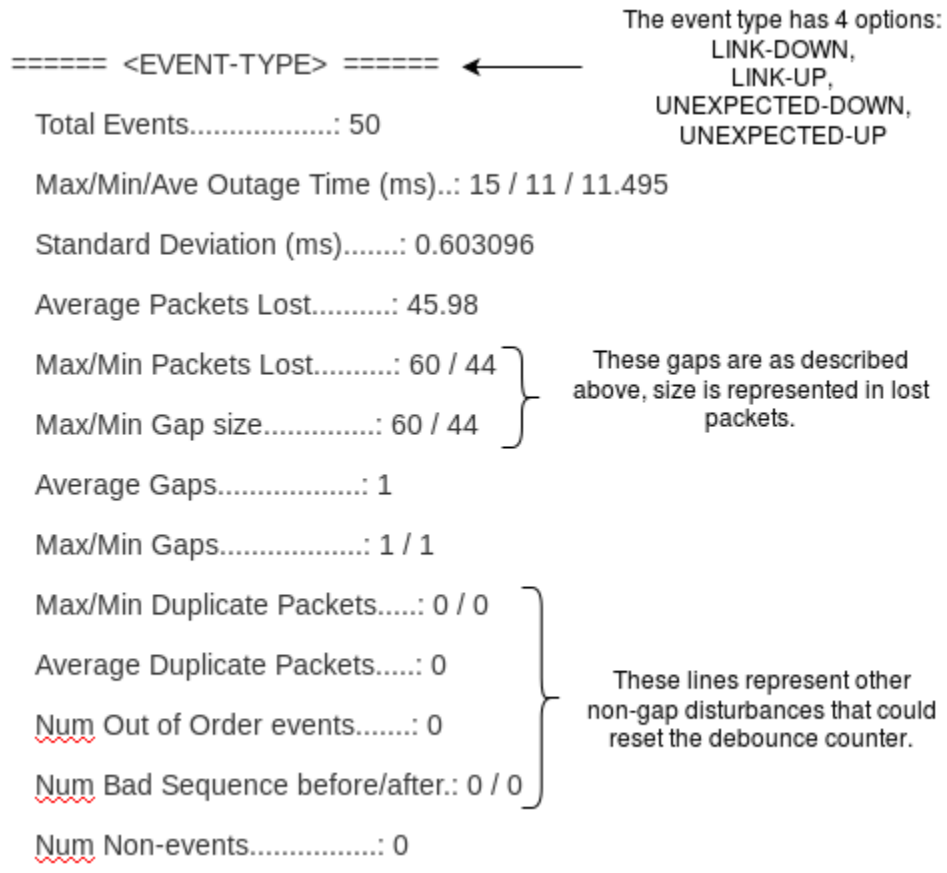
The HEADER line is just a description of the columns of the event lines which are described below.

For each event, a line is written in the results file, which is preceded with EVENT and is as described below.



Not shown in the example above is how duplicate packets and multiple gaps are handled. In the sequence breakdown, multiple gaps are represented by a much longer string of packets, and multiple highlighted "gap size" numbers, to show where in the sequence the additional gaps occurred. For duplicate packets, the type of event changes to OOE in the shorthand, OUT-OF-ORDER in the longhand, and the duplicate packets are highlighted.

Event Summaries



The maximum, minimum and average outage times are totals from any and all disturbances. The stats aren't particular about which disturbance causes the worst/best recovery, meaning the outage times reflect the longest and shortest recovery times out

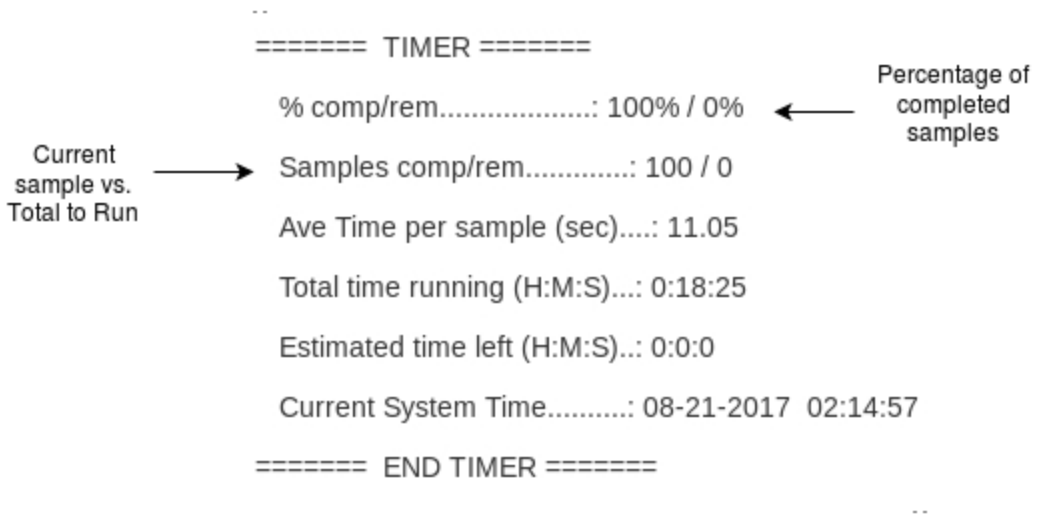
of all disturbances up to the point the stats were printed. If non-events occur, the minimum sizes of all statistics will be 0.

Just after the summary is a parsable version of the information in the summary, which is used to populate the statistics box on the graphical representations of the data. The information in it is ordered the same as in the header:

STATUS-HEADER,event-type,total-events,max-ot,min-ot,avg-ot,sd,avg-pkt-loss,max-pkg-loss,min-pkg-loss,max-gap,min-gap,avg-gap,max-dup,min-dup,avg-dup,avg-OOE,num-bad-seq-b4,num-bad-seq-after,num-non-events

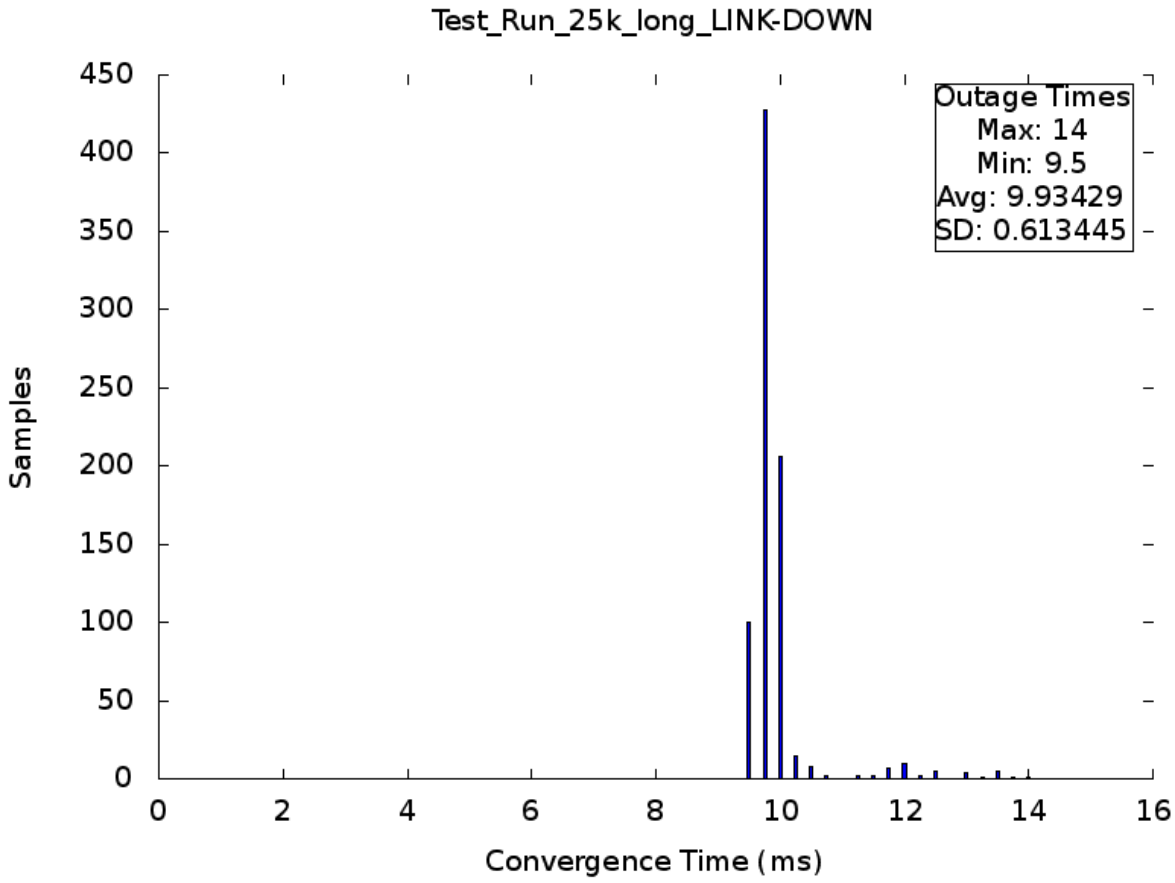
STATUS,<EVENT-TYPE>,50,15,11,11.495,0.603096,45.98,60,44,60,44,1,1,1,0,0,0,0,0,0,0

Timer Information

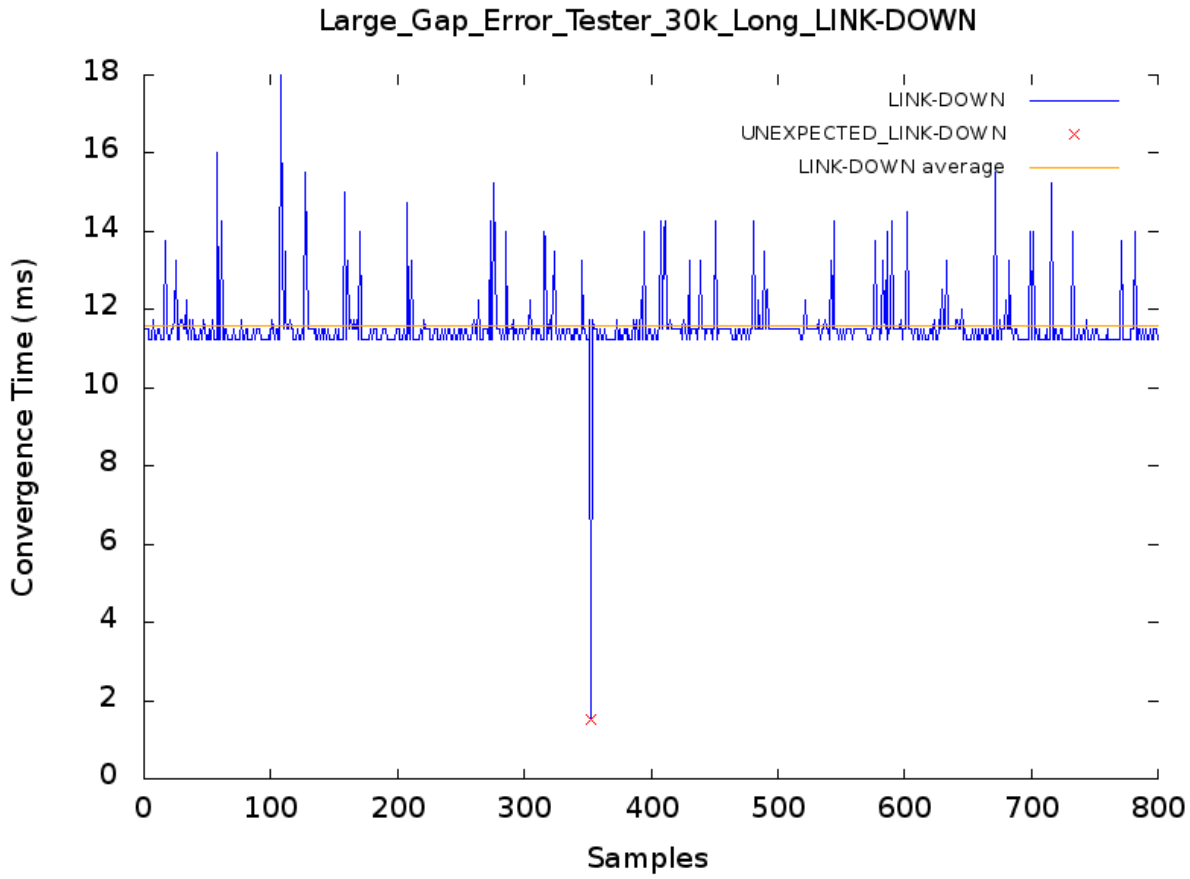


Something to note is: unexpected events are treated as triggered events in terms of count. Meaning, you might see 100 events completed, but only have 48 Link-Down events and 2 Unexpected Link-Down events.

- In histogram format, with separate histograms for link-up events, link-down events, unexpected link-up events, and unexpected link-down events.
  Histogram Example



- In line graph format, with link-up and unexpected link-up events combined and link-down and unexpected link-down events combined.
  - Line Graph Example



Methods for Triggering Events

There are two flavors of automated event triggering devices:

- A Network Link Breaker, which simply cuts and restores the connection between two ethernet ports quickly and efficiently.
- A power-cycler, which cuts and restores power to an entire device. This functions the same as the link breaker, but requires additional time per event for the system to settle.

Just Watch Mode

The receiver can also can be run in "justwatch" mode which is configured and operates the same as any other run but the receiver does not trigger events automatically. The receiver just watches the stream and if it observes a network disturbance, it then records it using the event monitoring configurations described above. This mode is useful when a specific test is not needed, but just manual observations of the performance of the network.