Understanding BGP Session Failures in a Large ISP

Lan Wang
lanwang@memphis.edu
Computer Science Dept.
The University of Memphis

Malleswari Saranu
msaranu@memphis.edu
Computer Science Dept.
The University of Memphis

Joel M. Gottlieb
joel.gottlieb@gmail.com
AT&T Labs

Dan Pei
peidan@research.att.com
AT&T Labs – Research

***Abstract*—The current global Internet routing frequently suffers from cascading routing changes and slow routing convergence. Such instability can significantly affect the performance of real-time Internet applications such as VoIP, multimedia conferencing and online gaming. One major cause of routing instability is the failure of BGP peering sessions, but there has been little understanding of the factors that contribute to the failures of *operational* BGP sessions. In this paper, we present a systematic study on the failures of a few hundred BGP sessions using data collected in a tier-1 ISP network over a 9-month period. We first quantify the impact of the session failures on both the control plane and the data plane. We then use syslog events to identify the direct triggers of session failures. Furthermore, we use several heuristics, including link failure information, session down time and traffic level, to identify the root problems that led to these session failures. We found that the major root causes are administrative session resets and link failures, each contributing to 46.1% and 30.4% of the observed session failures.**

I. Introduction

Today more and more people are participating in real-time network applications such as online gaming, VoIP, and video streaming. These real-time applications are very sensitive to routing changes because their performance depends heavily on the timeliness of data delivery. Unfortunately, the current global routing protocol BGP [1] suffers from both frequent routing changes and slow convergence, i.e. routing changes may take up 10s of seconds to stabilize ([2], [3], [4], [5], [6], [7]). Thus, in order to better serve Interent applications, especially real-time applications, we need to understand the major causes of BGP routing changes so that we can improve global routing stability. While there have been previous efforts in inferring the *locations* of routing changes (e.g. [8]), the *root causes* of the routing changes are extremely hard to infer since many different causes, such as physical layer failures, link layer failures, configuration changes, congestion and router software bugs, can lead to the same routing updates. Worse yet, BGP routing updates often provide little support in investigating these causes.

Our work takes a different approach – we use data directly collected from an ISP to analyze the causes and impact of BGP session failures. A BGP session failure in a large ISP can potentially affect the routes of a large number of destination prefixes, triggering a large volume of BGP updates that are propagated far beyond the two peers in the session. There have been a few studies of BGP session failures using analytical modeling and testbed experiments ([9], [10]). Wang et. al. further observed that, during the Nimda attacks, frequent session failures occurred over the *monitoring* BGP sessions used by RIPE NCC to collect routing data from ISPs [11]. However, little is known about exactly how BGP session failures are triggered in *real ISP networks*, perhaps due to a lack of the session failure data. Bonaventure et al. [12] studied the failures of eBGP peering links in one transit ISP, but it is not clear whether these link failures actually led to BGP *session* failures. In this paper, we present a first systematic study of BGP session failures using Syslog messages [13], BGP tables, SNMP traffic data, and router configurations collected in a tier-1 ISP network over a 9-month period. More specifically, we study the impact, characteristics and causes of the failures of a few hundred selected eBGP sessions and iBGP sessions. Note that we do not claim that the results are representative of all ISPs as our data is from a single ISP.

First, we identified all the session failures for these BGP sessions using the syslog messages. Based on historical BGP tables, we then estimated the number of prefixes affected by each session failure. Moreover, we estimated the traffic shift or loss caused by each session failure using the SNMP data (per-interface traffic volume). Our results show that up to thousands of prefixes and a significant amount of traffic could have been affected by a session failure. Such impact underscores the importance of understanding the session failures' causes, minimizing their occurrences and mitigating their effects.

Second, we analyzed the characteristics of the BGP session failures. More specifically, we studied the frequency of session failures, distribution of session down time and distribution of session lifetime. We also compared the differences among different types of BGP sessions. We have the following two observations: (1) the session failures are quite infrequent in general – the majority of the sessions had zero or a very small number of failures during our study period; and (2) multi-link eBGP and iBGP sessions are more stable than single-link eBGP sessions. These characteristics can be used in simulation studies on routing instability and real-time application's adaptability to the routing instability.

Third, we studied the direct triggers and the root causes of the session failures, since understanding the causes is crucial for reducing the chances of these failures. Based on our measurement results, we identified four major failure triggers (*BGP Notification Received, BGP Notification Sent, Admin Shutdown, and Peer Closed Session*). We also identified six major root causes: (1) administrative session reset; (2) router reboot; (3) link failure; (4) link congestion; (5) maintenance (including peer de-configuration, BGP session shutdown, and router shutdown); and (6) BGP errors. Finally, we developed an algorithm to infer the root causes based on link failure information, traffic congestion level and the signatures of certain root causes. We found that the top two root causes are *administrative session reset* and *link failure*, contributing to 46.1% and 30.4% of the observed session failures. Overall, administrative causes (including root causes 1, 2, and 5) constitute 69.1% of all root causes.

The rest of the paper is organized as follows. Section II provides some background information on BGP and reviews related work on BGP session failure. Section III describes our data sources and methodology for identifying the BGP session failures. Section IV - VII examine the impact, characteristics, triggers and root causes of the session failures. Finally, Section VIII concludes the paper and outlines our future work.

II. Background and Related Work

The Internet is composed of thousands of Autonomous Systems (ASes) and the Border Gateway Protocol (BGP) [1] is the standard Inter-AS routing protocol. BGP routers in neighboring ASes run *exterior BGP* or *eBGP* to exchange routing information, while routers in the same AS run *interior BGP* or *iBGP* to synchronize their routes learned from the outside. Two BGP peers establish a BGP session over TCP to exchange routing updates. When a BGP session is first established, the peers exchange all the routes in their routing tables. Afterwards, only the new changes are exchanged.

Because there is usually no direct signaling from the physical or link layer to BGP regarding failures (unless the new fast-external-failover option [14] is enabled), BGP *Keepalive* messages are exchanged every 60 seconds by default between two peers. If there are no Keepalive or update messages received from the neighboring router in the last 180 seconds (default value), the local router's *session hold timer* expires. After detecting errors such as hold timer expiration and malformed update message, a router will send a *Notification* message to the neighboring router and shut down the session. After the session failure, the local router will remove all the routes learned from the neighbor, and when the session is re-established, the routing tables need to be exchanged again.

A session failure could introduce a significant amount of routing instability due to the large number of routes withdrawn after the failure and re-exchanged after the session re-establishment. Wang et al. [11] showed that over 40% of the observed BGP updates at the RIPE RRC00 monitoring point during the Nimda worm attack can be attributed to the failures of the monitoring BGP sessions. It has been shown that one can infer a local BGP session's failure from archived BGP updates ([15], [16]). However, inferring the failures of remote BGP sessions remains an open question.

A few previous studies have examined BGP session failure behavior using analytical models and testbed experiments ([9], [10]). Shaikh et al. [9] showed that the probability of session failure depends on the congestion level and that BGP's resilience to congestion decreases as the RTT increases. Xiao et al. [10] developed a probability model for iBGP session failures and studied the effects of BGP timers and TCP retransmission behaviors on session failures. Bonaventure et al. [12] showed that eBGP peering link failures were common in one transit ISP, but did not study whether these link failures actually caused BGP session failures or what other factors could cause BGP session failures. Therefore, there has been little understanding of the factors that contribute to the failures of *operational* BGP sessions.

To increase the robustness of iBGP sessions, Xiao et al. proposed a modification to TCP [10]. The Graceful Restart mechanism [17] was introduced to mitigate the routing flaps during certain session failures. It allows a router to continue using the routes learned from its neighbor even when its session with the neighbor is down. In [18], Wang et al. proposed a Bloom-filter based approach that can speed up the table exchange after a session recovers from a failure. Bonaventure et al. [12] proposed the protection tunnel approach – when the underlying link of an eBGP session fails, the packets are forwarded to an alternative egress point via a protection tunnel.

III. Identifying BGP Session Failures

This section discusses our methodology to identify the session failures in a tier-1 ISP. We use router configurations to identify the BGP sessions and use syslog messages to identify the failures of these sessions.

A. Focusing on Inter-ISP Peering Routers

In a tier-1 ISP, there are typically thousands of BGP sessions. However, they are not equally important. Some of the eBGP sessions are used by the *access routers* to provide transit service to customer ASes; a failure of such a session will only affect the traffic to/from that customer. Other eBGP sessions are used by the *peering routers* to exchange traffic with *peer ISPs* (see [19] for AS relationship definitions); a failure of such a session can affect a large number of flows to/from different customers and cause large-scale BGP routing instability. Similarly, an iBGP session between a peering router and its router reflector will have more impact than an iBGP session between an access router and its router reflector. Therefore, in this work we focus on the most important sessions in the ISP: those eBGP and iBGP sessions belonging to the peering routers (they are dedicated to "peering"), as illustrated in Figure 1. We plan to apply our methodology to other BGP sessions in our follow-up work.

B. Multi-Link eBGP Sessions

During our study, we found that some eBGP sessions are configured as "eBGP multi-hop" sessions. This was surprising at first glance, because eBGP multi-hop configuration is often used when two peers do not have any physical connection directly. But our later discussions with the operators and configuration checking revealed that eBGP multi-hop is enabled in order to implement the so-called "loopback peering". That is, two eBGP neighbors have multiple direct links between them but have only one BGP session between their loopback

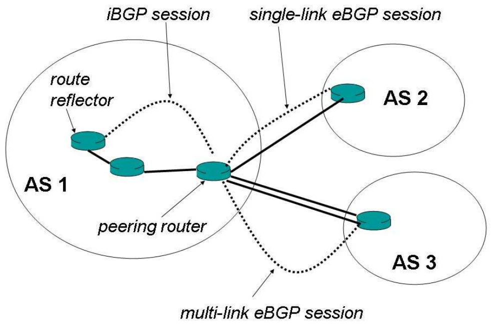


Fig. 1. BGP sessions in our study. Dashed line: BGP session; solid line: physical link.

addresses (a loopback address does not belong to any of the interfaces). This approach is preferred over having one BGP session for each direct link, as it offers easier policy management, lower memory consumption and load balancing. Usually, static routes are used to reach the neighbor's loopback address to avoid potential session oscillation problems that may arise when dynamic BGP routes are used to reach the neighbor. In this paper, we call an eBGP session with multiple physical links a *multi-link eBGP session*, and an eBGP session with one physical link a *single-link eBGP session*.

C. Removing Short-lived and Monitoring Sessions

We obtained BGP session information from the router configurations. During the 9-month study period, the peering routers had a few hundred BGP sessions in total, of which 42.2%, 17.4% and 40.4% are single-link eBGP, multi-link eBGP and iBGP sessions respectively. The set of sessions varied over time during the study period. This may be due to session removals and additions. According to [20], "it is common for a network to set up a trial peering session to determine the amount of traffic that would be exchanged should a session be turned up". We thus need to choose a set of sessions that last long enough to make our session failure measurement statistically meaningful.

For each session, we measured its *session existence duration* by counting the total number of days in which the session was administratively configured to be "up" in our study period. Note that the session existence duration is *not* the same as the total lifetime of a session as the session may still go down due to unexpected causes such as link failures even if it is configured to be up. We removed the sessions with an existence duration of less than 86 days. 86 was chosen as the cut-off threshold mainly because [20] indicates that a typical trial period is 3 months. As a result, 10.8% of all single-link sessions and 18.0% of all multi-link eBGP sessions were removed, but this process did not eliminate any iBGP sessions. Moreover, 15.5% of the iBGP sessions are *dedicated* monitoring sessions used to collect BGP updates from the peering routers, thus they were removed.

In summary, after removing the short-lived sessions and monitoring sessions, 43.7%, 16.6%, and 39.7% of the sessions in our dataset are eBGP single-link sessions, eBGP multi-link sessions, and iBGP sessions, respectively (see Table I).

TABLE I
SESSION AND FAILURE STATISTICS

Session types	% of all sessions	% of sessions after cleaning	% of all failures
eBGP single-link	44.2%	43.7%	77.0%
eBGP multi-link	17.4%	16.6%	9.7%
iBGP	40.4%	39.7%	13.3%

D. Using Syslogs to Identify BGP Session Failures

We used syslog messages to identify BGP session failures. Routers generate and send syslog messages that record, among other things, BGP session failures, physical layer, and data link layer failures, to a central server via UDP using the Syslog protocol [13]. Each syslog message has a sequence number that can be used to detect message losses. A detailed description of relevant syslog messages is presented in Section VI. In particular, the message "BGP-5-ADJCHANGE" indicates BGP session failure and session recovery events. Among the failures we located using this message, 77.0% are for single-link eBGP sessions, 9.7% are for the multi-link eBGP sessions, and 13.3% are for the iBGP sessions, as summarized in Table I. We then used SNMP traffic data to measure the impact of the session failures and again used the syslog messages to identify the immediate failure triggers as well as the root causes.

E. Normalizing Results

For proprietary reasons, we cannot present some results using their absolute values. Therefore, we normalize those results using either the maximum or the median of the data values. The normalization still allows the readers to understand the distribution of the results and make comparison among the different types of BGP sessions. Whenever possible, we also mention the order of magnitude of the absolute values.

IV. IMPACT OF BGP SESSION FAILURES

In this section, we measure a session failure's impact on both the control plane and the data plane.

A. Control Plane Impact

At the control plane, when a session between the local router and the peer fails, the local router will remove from its routing table all the routes learned from the peer. If any removed route was a best route before the session failure, the local router will re-compute its best path among the alternative paths, and send to all its peers the new best path or a withdrawal if no alternative paths are available. As a result, other routers in the network will need to re-compute their paths if their best paths were going through the failed session. Therefore, we estimate the control plane impact using the number of prefixes affected by each session failure as described below.

For the peering routers we studied, there are daily BGP table snapshots taken roughly at the beginning of each day in the last 5 months of our study period. Each BGP table lists both the best path and the alternative paths for each prefix and the nexthops of these paths. We match the nexthop with the peer

IP address in the session failure data and thus we are able to count the number of prefixes whose best paths were learned over the failed session. We then use this number to *estimate* the number of prefixes affected by the session failures on the same day. Interestingly, we found that this number does not change much (less than 5%) on different days, so we would expect that the inaccuracy due to not using the table just before the session failure is not significant either. Note that this methodology is not applicable to iBGP sessions due to different BGP nexthop semantics. Furthermore, Wu et al. [15] found that the operators might gracefully move the routes and traffic to other sessions before the planned maintenance. In such cases, our methology can over-estimate the control plane and data plane impact.

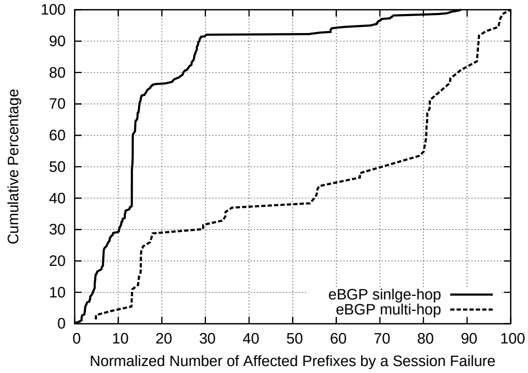


Fig. 2. Normalized Control Plane Impact of eBGP Session Failures

For proprietary reasons, we show the number of affected prefixes per session failure, normalized by the maximum number of affected prefixes of all session failures. Let $p(s)$ be the number of affected prefixes per session failure, and let max be the maximum number of affected prefixes in one single session failure. The normalized number of affected prefixes is $100 * p(s)/max$. Figure 2 shows the cumulative distribution of the normalized number of affected prefixes per session failure. Both single-link and multi-link session failures have significant impact on the control plane because thousands of prefixes can be affected. On the other hand, the median number of affected prefixes of multi-link sessions is about 6 times of that of single-link sessions. Our conjecture about the difference between single-link and multi-link eBGP sessions is that the multi-link sessions are likely to be more important (thus carrying more routes) than single-link ones.

B. Session Down Time

We define *session down time* as the duration when a session remains down after a session failure. We measure it as the difference between the timestamp of a session "down" message and the immediate next session "up" message in the syslog (we exclude the cases where there are lost messages in between). Figure 3 shows the CDF of the session down time, normalized by the maximum down time of all session failures in our data set. We observe that while the majority of downtime is fairly short (less than a few minutes), there exists very long downtime in eBGP sessions. One more interesting observation is that the single-link eBGP sessions have both extremely short and extremely long session down times. The session down times are likely to be related to the root causes of session failure, which will be investigated in Section VII.

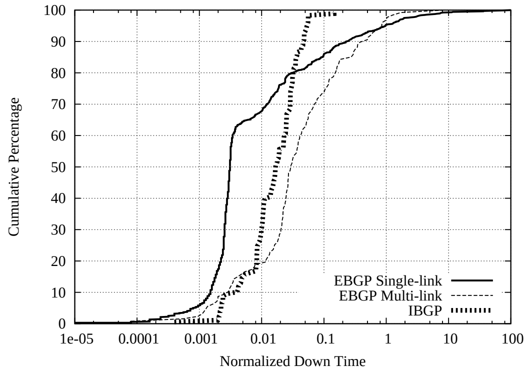


Fig. 3. Normalized Down Time of the BGP Sessions

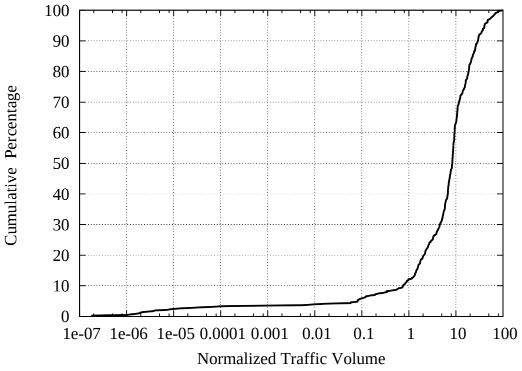


Fig. 4. Impact of BGP Session Failures on Traffic (single-link eBGP session)

C. Data Plane Impact

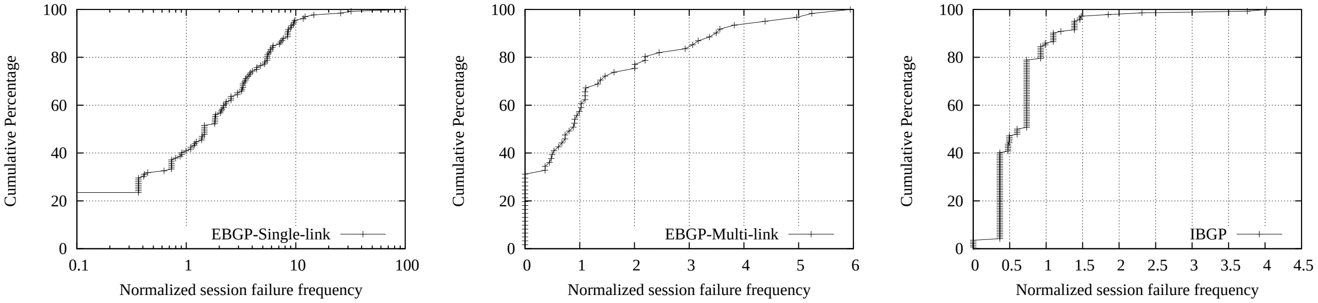
As the result of a session failure, the packets destinated to the affected prefixes can be dropped or redirected to alternative paths. To estimate the data plane impact, we measure the *total* volume of the "affected" (either lost or re-directed) traffic that could have gone through the link if the session had not failed. We use hourly SNMP traffic rate data *at the same hour exactly one week before the session failure occurred* to avoid the effects of "seasonal" factors (days vs nights, weekdays vs weekends). The affected traffic volume is the traffic rate times the minimum of a "re-direction time" and the actual session down time. We use a re-direction time of 60 seconds based on previous work on routing convergence [6].

Figure 4 shows the cumulative distribution of the estimated amount of affected traffic volume, normalized by the maximum value of affected traffic volume in our data set. The majority of the affected traffic volume is on the order of gigabytes. This shows the importance of understanding their causes.

V. Characteristics of the Session Failures

In this session, we characterize the session failures by measuring the failure frequency and session lifetime.

We define the *failure frequency* of a session as the ratio of *the total number of failures* to *the number of weeks during which the session existed* in our 9-month study period. Figure 5



(a) Single-link eBGP

(b) Multi-link eBGP

(c) iBGP

Fig. 5. Normalized Frequency of Session Failures

shows the CDF of the failure frequencies, normalized by the maximum failure frequency of all the sessions.

We define *the lifetime* of a session as the time during which the session stays “up”. We measure it as the difference between the timestamp of a session “up” event (or the start of our study period) and that of the immediate next session “down” event (or the end of our study period). We also exclude the cases where there are lost syslog messages in between. For each session, we calculate its average and maximum lifetime (a session with multiple failures has multiple lifetime values). Figure 6 shows the CDF of the average session lifetime, normalized by the median value of all the average lifetimes of the single-link eBGP sessions. The maximum lifetime distribution gives us similar observations, so it is omitted for brevity. Note that because of the finite observation window, our estimation is a lower bound on the session lifetime.

We also measure the *failure inter-arrival time* of a session, defined as the time between two consecutive failures of the same session. The trend of the CDF for failure inter-arrival time (not shown) is similar to that of the session lifetime. Based on these results, we make the following observations.

The session failures are quite infrequent in general. 22% of the eBGP single-link sessions, 31% of the eBGP multi-link sessions, and 3.3% of the iBGP sessions did not have any session failures during our study period. The absolute values of the 90th-percentile data points for all three types of sessions are very small (not shown in the figure for proprietary reasons). Note that the two vertical lines in Figure 5(c) correspond to those sessions who existed in the whole 9-month period and have only very small number of failures, thus their failure frequencies are the same.

iBGP sessions are more stable than multi-link eBGP sessions, which in turn are more stable than single-link eBGP sessions. This observation is clear by comparing the median and maximum failure frequencies in Figure 5 and the median session lifetimes in Figure 6. We attribute the difference among different types of sessions to whether a session has multiple underlying physical links/paths. First of all, the two peers of the multi-link session can reach each other via multiple underlying physical direct links. Hence even if one of the links fails, the others may remain intact and the session can remain up. Similarly, the iBGP sessions run on top of IGP, so an iBGP session can remain up as long as there is at least one working IGP path between the two iBGP peers. The chance is quite small that the two iBGP routers are partitioned by one link failure in a well-connected network.

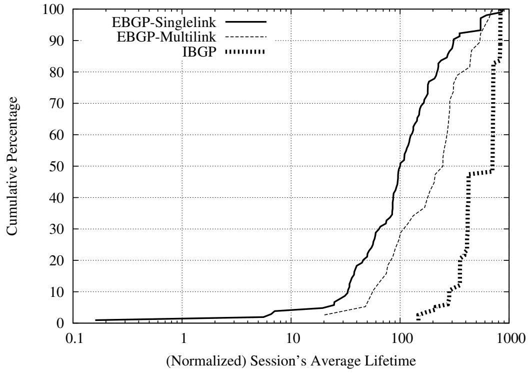


Fig. 6. CDF of Normalized Average BGP Session Lifetimes

VI. Session Failure Triggers

We now investigate what caused the BGP session failures, i.e. the *failure triggers* and the *root causes*, by examining various syslog messages relevant to session failures (see Table II).

We define the *the trigger* of a session failure as the reason that is perceived by BGP to shutdown the session. More specifically, it is the “reason” of the failure encoded in the BGP-5-ADJCHANGE message and its preceding BGP-3-NOTIFICATION message if any. On the other hand, the *root cause* of a session failure is the condition which caused the failure trigger to happen. For example, a session failure trigger can be “Hold Timer Expired” and the root cause can be a physical link failure that caused the timer to expire. We focus on the failure triggers in this section and study the root causes in the next.

A. Failure Trigger Definitions

Each failure trigger has two components: trigger type and error code. *Trigger type* is what BGP-5-ADJCHANGE says about the failure trigger. The table in Figure 7(a) lists the four types of failure triggers that we have observed in the syslog messages and each type is assigned a code. *Error code* is more detailed trigger information in the BGP-3-NOTIFICATION message. The table in Figure 7(b) lists all the error codes

TABLE II
SYSLOG MESSAGES MOST RELEVANT TO BGP SESSION FAILURE

Event Symbol	Meaning	Event Details
BGP-5-ADJCHANGE	BGP session with a neighbor goes up or down.	neighbor IP, up/down, event trigger
BGP-3-NOTIFICATION	A BGP notification message is sent/received.	neighbor IP, sent/received, error code
LINEPROTO-5-UPDOWN	An interface's data link layer changes state.	interface ID, up/down
LINK-3-UPDOWN	An interface's physical layer changes state.	interface ID, up/down

Trigger Type	Meaning
1. Notification Received	Peer sent a BGP notification msg to local router.
2. Notification Sent	Local router sent a BGP notification msg to peer.
3. Admin Shutdown	Local router's administrator issued a shutdown. No notification msg is sent.
4. Peer Closed Session	Peer terminated the session. No notification msg is received.

(a) Observed Trigger Types in BGP-5-ADJCHANGE

Error Code	Meaning of Code
3/1	Malformed attribute list in BGP update
4/0	Hold timer expired
6/1	Maximum number of prefixes reached
6/3	Peer de-configured
6/4	Administrative reset
6/6	Other configuration change

(b) Observed Error Codes in BGP-3-NOTIFICATION

Fig. 7. Failure Trigger: (trigger-type, error-code). see [1], [21] for a complete list of error codes list.

that we have observed in the study. Note that only trigger types 1 (Notification Received) and 2 (Notification Sent) have corresponding error codes.

Each error code, defined in [1] and [21], has two parts: a major code and a subcode. For example, in the error code $3/1$, the major code 3 means that there is a problem in the received BGP update message and the subcode 1 means that the problem is a malformed attribute list. RFC 4271 [1] specifies six major error codes and the subcodes for some of them, while the subcodes for error 6 (CEASE) are specified in [21]. In our study, we observed “UPDATE Message Error” (code 3), “ Hold Timer Expired” (code 4) and “Cease” (code 6), but we did not observe any failures with the major codes “Message Header Error” (1), “OPEN Message Error” (2), or “Finite State Machine Error” (5).

We represent a failure trigger as *(trigger-type-code, error-code)*. For an instance, (2, 4/0) means that a notification message has been sent (from the local router to the peer router) containing the error code 4/0, which suggests that the local router is closing the BGP session because its hold timer expired. Since there are no BGP notifications for trigger type 3 and 4, we use their trigger-type code to represent them.

B. Removing Outliers

In order to estimate the contribution of different failure triggers, we need to prevent some problematic sessions from skewing our results. For example, one of the single-link EBGP sessions had an abnormal number of session failures with the code (1, 6/6), i.e. notification received due to other configuration changes. This particular session contributed to 40% of all the session failures in this category. Obviously, this is an abnormal case that should be separately considered.

For each failure trigger, we count the number of failures per session and identify abnormal sessions using the IQR (Inter Quartile Range). Because our data contain extreme values, we use non-parametric statistical measures [22] such as median and IQR instead of mean and standard deviation. The IQR of a data series is the difference between the third quartile (Q_3) and first quartile (Q_1). We consider a value an outlier if it is outside the range $[Q_1 - 3 \times IQR, Q_3 + 3 \times IQR]$ and remove the outliers from each failure trigger. 31%, 15.5%, and 0% of the failures for single-link eBGP, multi-link eBGP and iBGP sessions are classified as outliers respectively. The results in the rest of this section are those after removing the outliers.

C. Observations

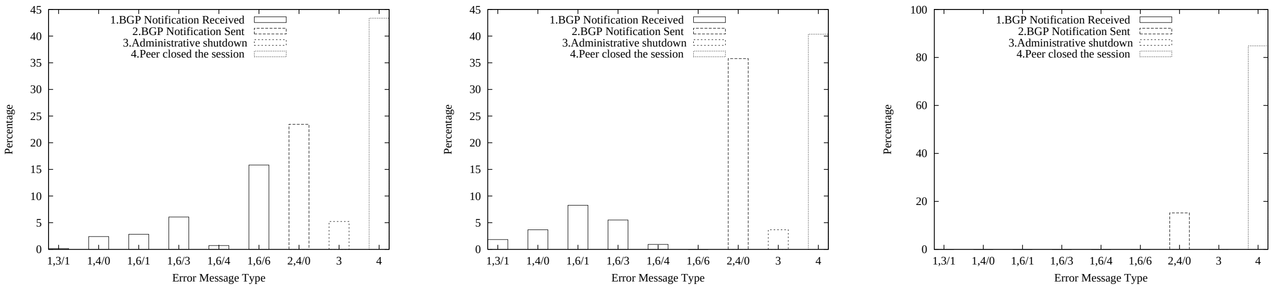
Figure 8 shows the distribution of failure triggers. We make the following two observations (these observations must be interpreted carefully given that multiple root causes can cause the same trigger to happen, as explained in Section VI-D).

First, *in all 3 types of sessions, the biggest contributor was “Peer Closed Session” (failure trigger 4) and the second biggest contributor was “Hold Timer Expired at the Local Router”(failure trigger (2, 4/0))*. “Peer Closed Session” contributed to 43.4%, 40.4%, 84.4% of the failures in single-link eBGP, multi-link eBGP, and iBGP sessions, respectively. “Hold Timer Expired at the Local Router” contributed to 23.4%, 35.8%, 15.6% of the failures in single-link eBGP, multi-link eBGP, and iBGP sessions, respectively. All other triggers contributed to only a small percentage of the failures.

Second, *there were fewer local triggers (trigger types 1 and 4) than remote triggers (trigger types 2 and 3)*. Local triggers contributed to about 30%, 40%, and 20% of the single-link eBGP, multi-link eBGP and iBGP failures, respectively, while remote triggers combined contributed to about 70%, 60% and 80% of the failures of the three types of sessions, respectively. One explanation could be that the peer routers might have had more administrative events (such as router reboot) than the routers we studied. In fact, in the next section we show that 72.5% of the iBGP session failures we studied can be correlated in time with the peers' reboots. Investigating the exact reasons will be our future work.

D. Multiple Root Causes for One Failure Trigger

While the root causes of most failure triggers are unambiguous based on their error codes (e.g., “malformed update”) if one exists, other triggers could have multiple potential root causes. For example, the top 2 triggers (“hold timer expired”



(a) Single-link eBGP Sessions

(b) Multi-link eBGP Sessions

(c) iBGP Sessions

Fig. 8. Distribution of Failure Triggers (Outliers Removed)

and “peer closed sessions”) can be caused by multiple root causes (e.g. link failure and congestion).

Furthermore, the lack of support for CEASE subcodes (when error code is 6) by some router vendors makes it more challenging to analyze the root causes. RFC 4271 [1] does not require the CEASE subcodes to be implemented. If a router that does not support the CEASE subcodes encounters such a condition, it will close its BGP session and log the trigger as “Admin Shutdown” (3) without sending a notification. As a result, the peer router will register a “Peer Closed Session (4)” or “Local Router’s Hold Timer Expired” (2, 4/0).

In fact, according to the BGP implementation survey dated Jan. 2006 [23], some router vendors’ BGP implementations (such as Cisco’s) do not support the CEASE subcodes. All the routers we studied and their iBGP peers are from one such vendor, which explains why no CEASE subcodes were observed in the iBGP sessions, and no *(2, 6/*)* messages were observed in the eBGP sessions. For the same reason, some of the eBGP peers might not support the cease codes. We observed that only 38% of the eBGP peers sent notifications containing the CEASE subcodes. However, it is difficult to know whether the remaining 62% peers do not support CEASE subcodes or they did not encounter the conditions that would trigger these codes. Therefore, some of the failures triggered by “Admin Shutdown”, “Peer Closed the session” and “Local Router’s Hold Timer Expired” in Figure 8 might actually have been caused by some of the CEASE subcode conditions. In Section VII, we develop some heuristics based on link failure information, session downtime and link congestion to *infer* the root causes of these three triggers.

VII. Root Causes of Session Failures

We classify the root causes into 6 categories: (1) administrative session reset; (2) router reboot; (3) link failure; (4) link congestion; (5) maintenance (including peer de-configuration, BGP session shutdown, and router shutdown); and (6) BGP errors. The difference between (1) and (5) is that a BGP session is immediately brought up after a session reset, but it may stay down for some time during maintenance. We use the failures of single-link eBGP sessions as a case study to examine the root causes. Below we first describe our inference methodology and then present the results.

Our methodology has two parts. The first part is to find direct evidences such as link failures from syslog messages and link congestion from SNMP data, and then correlate them with the session failures. The second part is to build a “signature” of the session down time for each root cause based on failures whose root causes are clear. For example, a failure caused by a malformed update is likely to be quickly followed by a session re-establishment, and a failure caused a router reboot may last as long as a few minutes. The signatures can then be used to infer whether similar conditions occurred and caused “Peer Closed Session”, “Hold Timer Expired” and “Admin Shutdown” to happen.

A. Finding Evidence of Link Failures and Congestion

We identify link failures using physical layer and data link layer syslog messages, i.e. the LINK-3-UPDOWN and LINEPROTO-5-UPDOWN events (see Table II for more details). These two messages have the interface IDs, but not the BGP neighbor IP information, so we use the router configurations to associate an interface with a neighbor IP. To correlate link failures with a BGP session failure, we look for them in the local router’s syslog messages within a time window preceding the session failure. We use a 240-second window because it is slightly longer than the default BGP hold timer value (96% of the studied sessions use the default 180-second timer value and the others use shorter timer values). We also tried other time window values with similar results.

After examining the link failure information, we found the following three cases for the session failures triggered by the local routers’ hold timer expiration. First, 44.9% of these session failures were preceded by both a physical link failure and a data link layer failure. Second, 15.9% of these failures were not preceded by any physical or link layer failures. We suspect that the peer router might have failed in these cases. Third, 39.2% of these failures had a link layer failure without any physical link failure. This is possible because, unlike the physical layer, the link layer uses keep-alive messages to detect failures and it may have false alarms. Therefore, to be conservative, we only consider a link failure to have occurred if the *physical* layer failed.

Congestion can also cause a BGP session to fail by delaying or dropping BGP updates and keep-alive messages. To identify whether a link was congested before a session failure, we obtain the traffic rate for each link 5 minutes (which is the smallest granularity of our SNMP data) before the failure using archived SNMP data. We then calculate the link utilization and compare it with a threshold to determine if the link is congested. Based on our discussion with operators, we choose a threshold value of 85%.

B. Session Downtime vs. Router Reboots

One possible root cause of session failures is the reboot of the peer router. Given that we do not have eBGP peer routers' syslogs, we use iBGP sessions to learn the characteristics of the session failures caused by peer router reboots. Since the syslog sequence number is initialized to a small integer when syslog process starts, and is increased by 1 for each subsequent message, we can identify a router reboot when we see the sequence number goes back to a small integer. We then temporally correlate the iBGP peer router's reboot with the local router's session failure using a time-window. We found that 72.5% of iBGP session failures were correlated in time with iBGP peer router reboots.

Our data shows that 80% of the failures took more than $T1$ seconds (the absolute number is not revealed due to proprietary reasons) to recover, because a router usually takes at least some fixed amount of time to boot up. We will combine this signature with the previous one to do our inference. Note that there is a certain amount of delay for a BGP router to detect a session failure, so the session down time can be shorter than the reboot time, which may be the reason why we see some extremely short down times.

C. Session Downtime vs. Failure Triggers

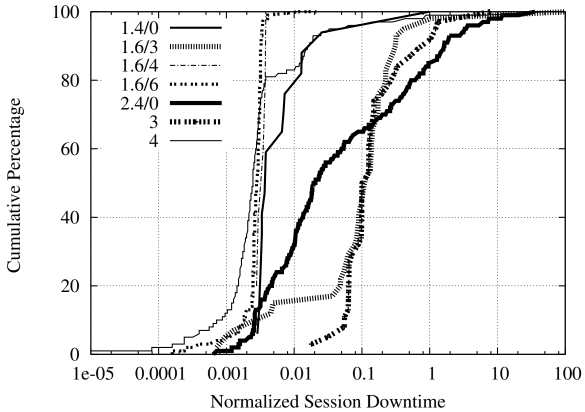


Fig. 9. Down Time of Single-Link eBGP Sessions vs. Failure Triggers

Figure 9 shows that different failure triggers lead to very different down time distribution (the down times are normalized by the maximum down time of all session failures in our data set). Note that (1, 3/1) and (1, 6/1) (both due to BGP errors detected by the peer routers) contain only very few data points, thus are not plotted.

First, one can observe that the curves can be divided into four groups. The first group contains the two curves on the left, i.e. "admin reset by peer router" (1, 6/4) and "other config changes by remote router"(1, 6/6). They have very short down times – 100% of the sessions recovered within $0.48 * T1$ seconds for (1, 6/4) and 99% recovered within $0.54 * T1$ seconds for (1, 6/6). This is not surprising since these two error codes indicate that the peer router is resetting the session due to configuration changes or for some other administrative purpose. The second group contains the two curves on the right, i.e. "local router admin shutdown"(1, 6/3) and "peer router de-configured the local router" (3). Their down times are much longer than the first group. More specifically, 68% of (1, 6/3) and 71% of (3) are longer than $T2 = 10 * T1$ seconds. These two error codes are similar: (1, 6/3) means that the peer de-configured the local router and (3) means that the administrator of the local router shut down the BGP session. The down time is longer perhaps because they correspond to maintenance events at the peer or the local router.

The third group contains "peer's hold timer expired" (1, 4/0) and "peer closed session" (4). Most of these failures recover very fast, but there are some cases with a long down time. This suggests that their root causes are mostly similar to those for group 1 with a few exceptions. The fourth group contains the middle curve "local router's hold timer expired" (2, 4/0), which spans a wide range of down time. This indicates that a variety of root causes, possibly including administrative reset and shutdown of the session by the peer, contributed to the local router's hold timer expiration.

In summary, the above results show that there is a correlation between the root causes and the subsequent session down time. Administrative session resets usually correspond to shorter down times (less than $T1$ seconds), while peer de-configuration and administrative shutdown of BGP session correspond to longer down times (longer than $T2$ seconds). Router reboots usually take more than $T1$ seconds but not very long. These signatures are used in our inference algorithm.

D. Inference Algorithm

We use the following algorithm to infer the root cause of a session failure (the six root causes are described at the beginning of this section). $T1$ is a time threshold that separates *short failures due to administrative reset and congestion* from *medium-duration failures due to router reboots*. Similarly, $T2$ is a threshold that separates *medium-duration failures* from *long-lasting failures that are due to maintenance*.

```
if $((trigger == (*, 6/4)) \lor (trigger == (*, 6/6)))$
  then $rootcause =$ admin reset;
elsif $((trigger == 3) \lor (trigger == (*, 6/3)))$
    then if $(downtime <$T1$)$
           then $rootcause =$ admin reset;
            else $rootcause =$ maintenance;
         fi
elsif $((trigger == (*, 3/1)) \lor (trigger == (*, 6/1)))$
    then $rootcause =$ BGP errors;
elsif $(linkfailure == true)$
    then $rootcause =$ link failure;
elsif $(downtime <$T1$)$
    then if $(trafficlevel \geq 85\%)$
           then $rootcause =$ link congestion;
```

```
        else $rootcause =$ admin reset;
      fi
elsif (($downtime \geq$T1) $\wedge$ ($downtime <$T2))
    then $rootcause =$ router reboot;
    else $rootcause =$ maintenance;
fi
```

E. Results

We apply the above algorithm to the single-link eBGP session failures, and the results are shown in Table III. We found that the two biggest factors are administrative session resets and link failures, each contributing to 46.1% and 30.4% of the session failures respectively. Maintenance events and router reboots account for 13.6% and 9.4% of the session failures. BGP errors and congestion are associated with less than 1% of the session failures. In other words, various administrative causes (admin. reset, router reboot, and maintenance) constitute 69.1% of the inferred root causes.

TABLE III
ROOT CAUSE RESULTS

root cause	percentage	root cause	percentage
adimin. reset	46.1%	router reboot	9.4%
link failure	30.4%	BGP error	0.5%
maintenance event	13.6%	link congestion	0%

Link failures are a considerable contributor to session failures. The implication is that one cannot assume that the physical link is functioning during a BGP session failure, and we need to be very careful in deploying the Graceful Restart mechanism [17]. Since Graceful Restart uses the old forwarding table during a session failure instead of flushing the obsolete routes, packets will be dropped if the link is down.

VIII. CONCLUSION

We have presented a first systematic study on the failures of hundreds of operational BGP sessions in a large ISP. We observed that iBGP sessions and multi-link eBGP sessions are more stable than single-link eBGP sessions. This demonstrates both the need and effectiveness of adding redundancy to important components in the global routing infrastructure. We also found that "Peer closed session" and "Local router's hold timer expired" were the top two session failure triggers (they make up more than 70% of all triggers). Using several heuristics including the correlation between the session downtime and their causes, we inferred the root causes that led to the single-link eBGP session failures. We found that the major root causes are administrative session resets and link failures, contributing to 46.1% and 30.4% of the session failures, respectively.

Our work can be extended in several directions. First, we plan to study the failures of more sessions than those covered in this work. Second, we plan to develop empirical models for session failures based on the failure characteristics learned from this study. Finally, we will use this model to evaluate the effectiveness of the proposed mechanisms such as Graceful Restart [17], FRTR [18], and protection tunnel [12] in mitigating the negative effects of session failures.

ACKNOWLEDGMENT

We are indebted to Jake Qualls for his help with revision of the paper. We thank Alex Gerber, Aman Shaikh, Carsten Lund, Chris Chase, Jay Borkenhagen, Jen Yates for their help with the data we used in this study. We also thank Aman Shaikh, Jia Wang, Olivier Bonaventure, Tom Scholl, and anonymous INFOCOM reviewers for their helpful comments on an earlier version of this paper.

REFERENCES

[1] Y. Rekhter, T. Li, and S. Hares, "A Border Gateway Protocol (BGP-4)," *RFC 4271*, Jan. 2006.

[2] C. Labovitz, G. R. Malan, and F. Jahanian, "Internet routing instability," in *Proceedings of the ACM SIGCOMM '97*, Cannes, France, Sep. 1997, pp. 115–26.

[3] ——, "Origins of Internet routing instability," in *Proceedings of the IEEE INFOCOM '99*, New York, NY, Mar. 1999, pp. 218–26.

[4] C. Labovitz, A. Ahuja, and F. Jahanian, "Experimental study of internet stability and wide-area backbone failures," in *Proceedings of the 29th Annual International Symposium on Fault-Tolerant Computing*, Madison, WI, Jun. 1999, pp. 278–85.

[5] O. Maennel and A. Feldmann, "Realistic BGP traffic for test labs," in *Proceedings of ACM SIGCOMM 2002*, Aug. 2002.

[6] C. Labovitz, A. Ahuja, A. Bose, and F. Jahanian, "Delayed Internet routing convergence," in *Proceedings of the ACM SIGCOMM*, Aug. 2000.

[7] C. Labovitz, A. Ahuja, R. Wattenhofer, and S. Venkatachary, "The impact of Internet policy and topology on delayed routing convergence," in *Proceedings of the IEEE Infocom*, Jul. 2001.

[8] A. Feldmann, O. Maennel, Z. M. Mao, A. Berger, and B. Maggs, "Locating Internet routing instabilities," in *Proceedings of ACM SIGCOMM 2004*, Aug. 2004.

[9] A. Shaikh, L. Kalampoukas, R. Dube, and A. Varma, "Routing stability in congested networks: Experimentation and analysis," in *Proceedings of the ACM SIGCOMM*, Stockholm, Sweden, Sep. 2000, pp. 163–74.

[10] L. Xiao and K. Nahrstedt, "Reliability models and evaluation of internal BGP networks," in *Proceedings of the IEEE INFOCOM*, March 2004.

[11] L. Wang, X. Zhao, D. Pei, R. Bush, D. Massey, A. Mankin, S. Wu, and L. Zhang, "Observation and analysis of BGP behavior under stress," in *Proceedings of the ACM SIGCOMM Internet Measurement Workshop 2002*, Nov. 2002.

[12] O. Bonaventure, C. Filsfils, and P. Francois, "Achieving sub-50 milliseconds recovery upon bgp peering link failures," in *Proceedings of the Co-Next 2005*, October 2005.

[13] R. Gerhards, "The syslog protocol," *Work in Progress, draft-ietf-syslog-protocol-19.txt*, Nov. 2006.

[14] "BGP fast external failover." [Online]. Available: http://www.cisco.com

[15] J. Wu, Z. M. Mao, J. Rexford, and J. Wang, "Finding a needle in a haystack: Pinpointing significant BGP routing changes in an ip network," in *Proceedings of 2nd Symposium on Networked Systems Design and Implementation (NSDI)*, Boston, MA, May 2005.

[16] B. Zhang, V. Kambhampati, M. Lad, D. Massey, and L. Zhang, "Identifying BGP Routing Table Transfers," in *ACM SIGCOMM Mining the Network Data (MineNet) Workshop*, August 2005.

[17] S. Sangli, Y. Rekhter, R. Fernando, J. Scudder, and E. Chen, "Graceful restart mechanism for BGP," *Work in Progress*, July 2006.

[18] L. Wang, D. Massey, K. Patel, and L. Zhang, "FRTR: A scalable mechanism for global routing table consistency," in *Proceedings of the International Conference on Dependable Systems and Networks*, Jun. 2004.

[19] L. Gao, "On inferring automonous system relationships in the internet," *IEEE/ACM Transactions on Networks*, vol. 9, no. 6, 2001.

[20] B. Norton, "Peering BOF IX," in *May 2005 NANOG*, May 2005.

[21] E. Chen and V. Gillet, "Subcodes for BGP cease notification message," *RFC 4486*, Apr. 2006.

[22] S. Siegel and N. J. C. Jr., *Nonparametric Statistics for The Behavioral Sciences.* McGraw-Hill, 1988.

[23] S. Hares and A. Retana, "BGP-4 implementation report," *RFC 4276*, Jan. 2006.