IP Forwarding Anomalies and Improving their Detection Using Multiple Data Sources

Matthew Roughan* Tim Griffin† Z. Morley Mao‡ Albert Greenberg§ Brian Freeman¶

ABSTRACT

IP forwarding anomalies, triggered by equipment failures, implementation bugs, or configuration errors, can significantly disrupt and degrade network service. Robust and reliable detection of such anomalies is essential to rapid problem diagnosis, problem mitigation, and repair. We propose a simple, robust method that integrates routing and traffic data streams to reliably detect forwarding anomalies, and report on the evaluation of the method in a tier-1 ISP backbone. First, we transform each data stream separately, to produce informative alarm indicators. A forwarding anomaly is then signalled only if the indicators for both streams indicate anomalous behavior concurrently. The overall method is scalable, automated and self-training. We find this technique effectively identifies forwarding anomalies, while avoiding the high false alarms rate that would otherwise result if either stream were used unilaterally.

Categories and Subject Descriptors:
C.2.3 Network Monitoring, C.4 Reliability, availability, and serviceability.

General Terms: Algorithms, Management, Reliability.

Keywords: Network anomaly detection, routing, BGP, traffic, SNMP.

1. INTRODUCTION

Anomaly detection is useful in network management for a range of applications, from detecting security threats (e.g. DoS attacks), to detecting vendor implementation bugs, network misconfigurations or faults. One wishes to detect times where the network is behaving abnormally, as action may then be required to correct a problem. Anomaly detection can be particularly useful in the context of reliability. Reliability is a critical objective in large IP networks, but many factors (for instance code bugs) are outside of an operator's ability to control. An alternative to preventing outages is to rapidly recover from these — for instance see the arguments presented in [1, 2]. In order to recover quickly, one must detect and localize a problem quickly.

However, while detection and alarming on real problems is important, it is equally important to keep the rate of false alarms low. A high false alarm rate results in genuine events being lost in the "snow" of false events. Statistical anomaly detection tests are run often (e.g., every five minutes), on large networks (with ten's of thousands of links), and so even a seemingly low false alarm rate may result in enough false alarms to overwhelm network operations staff. In the worst case, false alarms undermine anomaly detection, as operations staff tire of reacting to false alarms, and ignore or turn the system off entirely.

IP forwarding anomalies represent a large class of network anomalies, that relate to problems in forwarding packets to their destinations. More precisely, a *forwarding anomaly* is a period during which a significant number of packets fail to successfully exit the network at an appropriate point. Network component failures (line card, optical amplifier, or router outages, and fiber cuts) are not usually within the class of such anomalies. During such events, IP traffic is rerouted along alternate paths, resulting in at most a short transient anomaly while the routing protocols reconverge. Also, such failures are typically isolated, and easily detectable via other means, e.g. the Simple Network Management Protocol (SNMP). However, as we note below, component failures may trigger some larger network error, or occur simultaneously.

Forwarding anomalies can be the result of several problems:

- **Bugs:** Bugs in router software may cause forwarding problems that do not register via any hardware alerts, or may further be related to bugs in the instrumentation itself.
- **Misconfigurations:** The IP control plane — the distributed protocols that coordinate the building of forwarding tables throughout the network — is very complex. In such systems it is hard for an operator to understand the state of the system, and therefore the possible impacts of their actions [2]. The result is that routers may be configured in such a way that packets do not reach their destinations, even though the network may appear to be working normally. The distributed nature of the Internet means that such misconfigurations are not entirely under the control of a single operator. BGP allows network operators to misconfigure their systems in a way that may impact another network. Note that in [3] the authors found that operator errors in the form of misconfiguration was the major cause of service affecting outages, and BGP in particular has been seen to be hard to administer, so we may expect occurrences such as this to be a major form of forwarding anomaly.
- **Cascading network failures:** In a cascading network failure, a simple failure (such as that of a line card) results in widespread disruption of the network. Although, the possibility of such a collapse is not anticipated in the Internet, such failures are difficult to predict, or control, and certainly have been observed in other

*School of Mathematical Sciences, University of Adelaide, <matthew.roughan@adelaide.edu.au>
†Intel Research Cambridge <tim.griffin@intel.com>
‡University of Michigan <zmao@eecs.umich.edu>
§AT&T Research, <albert@research.att.com>
¶AT&T Labs, <bdfreeman@att.com>

SIGCOMM'04 Workshops, Aug. 30+Sept. 3, 2004, Portland, Oregon, USA.

network systems, for instance the power grid in North America in August 2003, and in the telephone network [4].

- **Latent errors:** It is possible to have latent errors: problems that are not significant until another error triggers them. A simple example might be a backup path that has been misconfigured, so that it does not work. Without careful testing such a problem might not come to light until after failure of the primary link. The failure of the primary link would be dealt with using normal procedures for detection and re-mediation, but without anyone realizing that the backup path was also failed, they might not give this task the priority it requires.

- **Exogenous factors:** It seems very unlikely that exogenous environmental effects can cause large scale failures. Networking equipment is generally held in tightly controlled environments, with redundant power supplies, and A/C. However, in rare cases, for instance the Sept 11 attack on the World Trade Center, the scale of external factors was large enough to affect a large part of the network. Although critical locations often have connections to multiple grids, and backup power supplies, very large scale power outages (such as the North-Eastern American blackout of August 2003) might also impact services in some networks if they do not have sufficient redundant power provisions for a long-term, large-scale outage. Obviously, detection of such impacts will be secondary to the event in question, but may none-the-less be useful in order to quickly assess the scale of the impact.

- **Simultaneous failures:** Most networks plan for single component failures, for instance, by providing pairs of redundant links. Given independent failures (consideration must be given to shared risk link groups when making such assumptions [5]) simultaneous failures should be unlikely. However, occasionally failures occur simultaneously. For example in May 2000 Optus (a major provider in Australia) had simultaneous, independent failures of the two redundant links between a pair of major cities [6].

A consistent property in these problems is that the standard methods for detecting network problems, for instance SNMP traps, syslog messages, etc., either do not detect such events, or are not likely to see the true extent of the problem. Understanding the extent of a problem quickly is important to prioritize the event appropriately.

Forwarding anomalies (by definition) have severe or network-wide impact, resulting in dropping large quantities of packets. For instance, on October 3rd 2002, a forwarding anomaly resulting from a router software bug caused a major tier-1 provider to lose a large volume of traffic [7], network-wide, over several hours. During this period, there were large drops in traffic on peering links (as measured by SNMP) and noticeable effects on Border Gateway Protocol (BGP) routing (as seen, for example, from customer and public viewpoints, such as Routeviews [8]). While this problem is near the extreme end of the spectrum there have been other, smaller scale forwarding problems in many ISPs (see, for example, email archives posted to the North American Network Operators' Group, NANOG, `http://www.nanog.org`). A major outage (several hours long covering a large proportion of the network) of a tier-1 ISP [7] motivated this work, in particular, the data sets used here, because the event showed up particularly clearly in both SNMP, and BGP data.

We investigate an approach for reliable detection with a low false alarm rate, which integrates multiple data sources. Specifically, this paper presents an analysis of using SNMP data, in conjunction with BGP data, to detect and localize forwarding anomalies in a large tier-1 ISP environment. Each data source provides a different view of such anomalies. The SNMP data provides traffic volumes, while the BGP data concerns routing between autonomous systems. Both individual data sources have problems in data quality and in missing causality information that lead naturally to false alarms. On the other hand, if such problems are suitably uncorrelated in two data sources, then the false alarm rate can effectively be diminished by correlating the two. This is precisely the intuition behind our choice of data sources and our method.

First, we transform each data set individually to create useful anomaly metrics. Though SNMP usage data is relatively simple — the number of packets or bytes that traverse an interface between successive polling intervals — operational measurements for large networks can be relatively complex and noisy. We use two methods to extract the anomaly indicators from this data: a standard technique called Holt-Winters, and a second method based on a decomposition of the traffic. BGP dynamic updates, on the other hand, provide a rich, high-dimensional data source, with considerable volatility. Here, to extract a useful anomaly indicator, we transform the raw data to simulate and track BGP tables at locations throughout the network. We then use the dynamic count of the number of routes satisfying a given predicate, and use a modified exponentially weighted moving average to signal anomalies. Last, we correlate the SNMP and BGP anomalies to produce a combined indicator.

In this investigation we have principally looked for rapid detection and diagnosis of larger scale events — that is, those that concern more than one router or link. We note that forwarding anomalies sometimes self-repair relatively quickly, so that no remedial action is possible or necessary. However, these anomalies can still cause transient disruptions and degradations of service quality for sensitive applications, such as Voice over IP or interactive gaming. Reliable detection is still important for tuning network protocols and processes to track and reduce the occurrences of transients.

Apart from testing several anomaly detection algorithms on individual data sources, the main contribution of this paper is the finding that using traffic and routing data together significantly reduces the false alarm rates for forwarding anomaly detection. The reason is that during a forwarding anomaly, traffic fails to reach its correct exit point from the network. A large part of the traffic on a major ISP exits the ISP at its peering points, and so a major forwarding anomaly will be noted by a change in traffic along this edge of the network. Similarly, the routing to peers is controlled via BGP, and so large scale forwarding anomalies will appear in this data source as well.

1.1 Related work

Previous work on network anomaly detection has primarily focused on security tasks (detecting DoS attacks, worms, or other intrusions) and has often been signature based. We seek to find anomalies which may never have occurred previously, and so do not have a known signature. In many cases providers use very simple techniques for anomaly detection, such as fixed thresholds, but such techniques are quite limited. There has however been some more sophisticated work in the detection and analysis of network anomalies. Instances are [9, 10, 11, 12, 13, 14]. Of these, the most directly relevant to this paper is [12] which tests the use of the Holt-Winters forecasting technique for for network anomaly detection. Also of note is [13] which proposes a wavelet based method with great potential, but is most noteworthy because it has the strongest set of data used for testing these algorithms. [13] contains one of the first large scale, quantitative tests of algorithms for network anomaly detection.

In addition to the papers describe above, there are a few works on correlating alarms or various sorts via various means, for instance see [15, 16, 9, 10]. Such work is relevant here in the sense that we are performing a type of alarm correlation, although the method we use here is very simple, despite its good performance. One might apply techniques such as those suggested in these papers to improve performance in the future.

2. BACKGROUND

2.1 SNMP

In this paper, we analyze Simple Network Management Protocol (SNMP) traffic data extracted from a large archive drawn from a

large tier-1 ISP's backbone network. SNMP is unique in that it is supported by essentially every device in an IP network. The SNMP data that is available on a device is defined in a abstract data structure known as a Management Information Base (MIB). An SNMP *poller* periodically requests the appropriate SNMP MIB data from a router (or other device). Since every router maintains a cyclic counter of the number of bytes transmitted and received on each of its interfaces, we can obtain basic traffic statistics for the entire network with little additional infrastructure.

SNMP data has many practical limitations – for instance missing data, incorrect data, and a coarse sampling interval (typically five minutes, though one could easily collect data at somewhat finer intervals), and an off-set in the measurement times between different devices. A further limitation is that SNMP only provides aggregate link statistics – we cannot determine anything about the type of traffic using the link, nor its source or destination.

In the network in question we have more than one year's SNMP data, gathered at five minute intervals. The data contains the MIB-II counters, which include the number of bytes in and out for each interface in the network, and this is the data we shall examine here. We first aggregate the data onto one hour intervals, using nearest neighbour interpolation of the samples. This transforms the problems of off-set and missing data, and sample jitter into a small amount of noise in the measurements. We then aggregate this data to examine the total traffic along the peering edge at each Point-of-Presence (PoP) at which there are inter-peer connections.

2.2 BGP

The Border Gateway Protocol (BGP) has the very important role of exchanging the reachability information between the Autonomous Systems or ASes. Monitoring BGP updates can reveal all the changes in the best route used by a given router and can in turn indicate routing problems in the network. Route monitors, supporting passive BGP peering sessions, can be set up to receive all the routing changes. Such route monitors archive all the update messages and regularly dump all the routes in the BGP table. The Oregon Route-views [8] project and the RIPE NCC RIS project [17] have both set up such routing sessions with numerous ISPs to obtain the best routes used by these networks.

Inside a service provider's backbone network, it is also very valuable to monitor the best routes used by various routers to study potential network problems. Such data allow one to differentiate the problems within one's network from those in neighbouring networks. In our experiments, we make use of measurements from route monitoring sessions to route reflectors in all major PoPs of the backbone under study. The data include all the BGP updates, i.e., the routing changes, as well as daily table dumps of all the best routes used. The monitoring sessions see all the available routes. Given a BGP table which contains all the best routes used, one can apply the updates to obtain the best routes at any given time.

BGP updates comprise an extremely rich, high dimensional, and relatively volatile data source. The volatility arises from the fact that BGP sees routing changes over a large segment of the total Internet. The challenge is to define metrics that allow for rapid calculation and prediction of forwarding anomalies. In general, we have found metrics that count the number of routes that satisfy a given predicate to be very useful in detecting and explaining routing phenomena and various forwarding anomalies. In the particular case here, we shall focus on the distribution of routes via different egress points. A router typically has several choices of where to send traffic for most of the prefixes in the table. A sudden shift in the distribution of routes gives a good indication that the egress point that is less preferred may experience some network outages. In this paper, we will investigate metrics that track (for each route reflector under observation), the number of routes that exit a given PoP. These metrics are easily calculated by combining BGP updates with configuration data that maps BGP next hops to BGP routers.

For each PoP we shall store a time series constructed from the number of best routes seen exiting the PoP by the local route reflector. We further aggregate this data into time bins so that we store the minimum and maximum number of routes seen during each bin. For the purpose of this study we use one hour bins, though this number is arbitrary, and can easily be reduced to bins as small as a minute with very little additional computational load.

This data can contain artifacts as a result of resetting the BGP session between the route reflector and the route monitor. These artifacts have been removed using data archived at the BGP monitor, but other artifacts may persist.

3. ANOMALY DETECTION

Anomaly detection can be thought of as an attempt to detect data that shows evidence of emerging from a different process or mechanism than typical data. There are far too many possible approaches to anomaly detection to list here. We shall take the approach of using several very simple, but reasonable detectors for individual data sets, and rely on the combination of data sets to provide good performance. One could obviously improve the quality of the detectors on individual data sets using more sophisticated algorithms, but this paper shows that such an approach is not needed, given the correct combination of data sets. Furthermore, the false alarm rate for any one data set will be intrinsically limited by that data set, and so one must use some combination of data in order to get useful performance.

Many simple approaches to anomaly detection are basically outlier detection — detecting observations that differ by a large amount from *normal* observations. The method by which you measure the normal behavior of the observations is still of interest. There are a number of alternatives available, depending on the properties of the data in question:

- **EWMA:** The Exponentially Weighted Moving Average (EWMA) chart is a text-book method applied in quality processes (for instance see [18]). The method is broadly applicable to data with a stable, stationary mean, and independent observations. We will adapt this approach for use on the BGP data analyzed here. Exponential smoothing is a method for computing a prediction of the value of some measured quantity, where the measurements include noise. Given the prediction, we can assess how far the next measured value is from its prediction, and thus decide if the measurement is an outlier. One assumes that the process can change (slowly) over time, and so more recent measurements are more relevant, but may be overridden by the body of preceding data.

- **Holt-Winters:** The Holt-Winters method is a generalization of the EWMA for data which shows both periodic variations (both in mean, and variance), and long term trends, in addition to stochastic variations. Holt-winters has been perhaps the most commonly tested algorithm in the context of Internet anomaly detection [12, 14, 13] because of the fact that while not optimal, it does not make many assumptions and is therefore quite robust.

- **Decomposition:** In addition to Holt-Winters we apply a recent method using a model developed in the context of backbone Internet traffic [19]. The method, like Holt-Winters, is based on the idea that traffic is composed of a long-term trend, a periodic component, and noise, from which we wish to extract anomalies. The method has a number of advantages due to a more appropriate model for traffic data.

There are many alternative methods, for instance, the wavelet techniques explored in [13] are particularly appealing. Other methods include ARIMA modeling, as used in [14], or Bayesian techniques, for instance see [9]. These methods might allow for improvements in false-alarm rates in themselves. However, the properties of the data itself suggest that there will always be false alarms for any detector of the required sensitivity.

3.1 Anomaly detection in SNMP traffic data

Internet traffic shows strong non-stationarity both in the mean, and the variance [12]. This non-stationarity has two major components – a long term trend, and a periodic, or seasonal component. What's more, as noted in [12] the periodic component can gradually evolve, for instance, as the number of evening daylight hours changes from summer to winter. Thus, we apply Holt-Winters, and the decomposition technique to anomaly detection in SNMP data, because they are designed to work on this type of data.

3.2 Anomaly detection in BGP data

As with the SNMP data, the BGP data is non-stationary. It exhibits trend over long time intervals (typically the number of routes in the Internet has grown over time), and also dramatic changes where changes in routing policy (for instance introduction of prefix aggregation) have resulted in shifts of a large number of routes. BGP data is, as noted above, relatively volatile. However our metric – the number of routes matching a given predicate – shows good stationary properties over moderate time scales (days), and little periodic time-of-day, or day-of-week variation. Hence EWMA seems applicable. In the algorithms here we use as our observations the minimum number of routes in any time interval, which provides a faster method of alerting to a drop in the number of routes.

We modify the EWMA technique somewhat to effectively allow for rapid reinitialization after a large level shift (a case which the EWMA algorithm does not deal with well). The method used here is to estimate the mean of the normal number of BGP routes using the EWMA. However, if the value falls outside our the threshold around the predicted value, indicating an anomaly, we shall not use it to update the EWMA. If we were to discount such outliers entirely, we would loose track of the process at any large step. We would mark the first point of the step as an outlier, and so discount it, and therefore the estimated mean would remain unchanged, causing us to mark the next and all future points as outliers. To avoid this pitfall, we allow a maximum gap in the data – if outliers persist for longer than this gap then we reset the value of the EWMA.

3.3 Correlating anomaly indications

We apply arguably the simplest approach possible for correlating the anomaly indications from the different data streams: for any one PoP if both BGP AND SNMP anomaly indicators report anomalies, we shall generate an alarm. This simple correlation allows us to focus on the hypothesis that the power of the approach derives principally from combining indicators whose false alarms will be uncorrelated, as the false alarms relate to independent glitches in data collection or natural variations of the individual data streams. The vast literature on pattern recognition provides more sophisticated correlation approaches, which remain open for improving the results – for instance the authors of [9] propose the use of Bayesian inference. However, note that the following tests demonstrate that the very simple approach used here already produces results of practical quality.

4. RESULTS

We evaluate the anomaly detection method using three data-sets:

- A long set of SNMP/BGP data covering eight large PoPs in the tier-1 ISP backbone IP network spanning an interval of more than one year.

- A second shorter set of data from February to May 2003, inclusive, consisting of detailed *fault tickets* from the network. Fault, or trouble tickets are the representation of a network problem in the database system used by operators to track a problem from detection to re-mediation. These tickets contain data on the start, and end time of a fault, and its root cause. However, this data is primarily text, entered by operators as they discover, and analyse a problem. Hence the times are often not precise, and the records must be processed laboriously by hand to obtain the relevant data. For instance, see [3] for more discussion of this type of data source.

- A list of the events that were considered by the IP operations group to be of the type that we wished to detect (using the methods above). This list is very short, as such events are rare, but they are of such scale that it is still important to detect them, even if they occur less than once a year.

The results given here are broken into two sections. First we shall consider two illustrative examples to see how the algorithm works. We shall then undertake a more systematic examination of the algorithm's performance.

4.1 Examples

We start by considering two illustrative examples of the algorithm in practice. The example shown in Figure 1 (a) covers the failure of a major network peer. During this failure, the peer dropped traffic along its peering links in a number of locations. In the figure, we focus on a PoP where nearly half of the traffic (and corresponding routes) arose from that peer, and so the failure stands out clearly. However, the failure was also detected in a number of other locations. While this example would not lead to a particular remedial action on the part of ISP, some operational actions may be called for to mitigate the impact of the outage. Moreover, the example is highly illustrative for two reasons. Firstly, it shows the nature of the SNMP and BGP time series used here. Secondly, it clearly shows the large deviations during the outage. A third feature to note are the presence of two false alarms in the BGP data – dips that do not correspond to noticeable traffic changes.

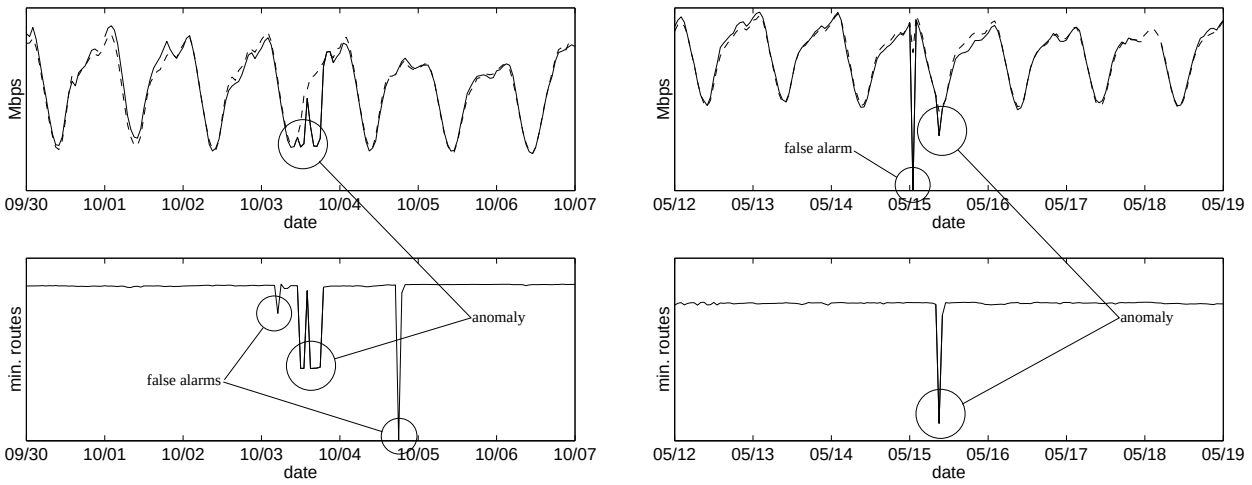
The second example, in Figure 1 (b), shows an example of what we see when a router with many peering links has an outage (in this case a planned outage). We can see early in the morning, local time (the date ticks occur at 00:00 GMT), a dip in both BGP, and SNMP. The plot should reinforce those points above, with the addition that we see a false alarm (near 00:00 5/15) in the SNMP due to missing data. Note that such a large drop in traffic *should* be detected by any anomaly detection algorithm one might build. However, it is an artifact in the SNMP data. The presence of such artifacts is a key reason why correlation between data sources is needed. The data sources are generated by completely different processes, and so artifacts are unlikely to occur in two data sets at the same time.

4.2 Statistical analysis

In this section we shall conduct a more systematic examination of the data. The first task is to determine the sensitivity, or detection probability – the probability that an anomaly is detected. For this we use the short list of known events, and the SNMP/BGP data set over the enclosing time interval. The algorithms above performed perfectly on this list, and further, indicated the location of the problem. Of course, with a small set of event data it is hard to determine precisely the detection probability, but it is likely that high detection probabilities are easily achievable because the events we wish to detect are large scale, with a significant impact on measurements.

The second, and more difficult task is to determine the selectivity, or false alarm rate. To assess the false alarm rate we take each detected event from the period of February to May 2003, and examine the detailed fault tickets to determine its root cause. The results of this investigation are shown in Table 1 — note we cannot report the actual number of events because these numbers are considered sensitive data, however the numbers are quite small.

The majority of events detected by either technique turn out to be caused by simple router or link outages on the peering edge of the network, such as shown in Figure 1 (b) (both methods detect exactly the same set of such events). For instance, if a router with many peering linkages is taken out of service, then the routes that would



(a) Example 1. (b) Example 2.

Figure 1: Examples showing SNMP traffic data (top plots), and BGP routes (bottom plots).

root cause	percentage of events	Holt-Winters events
edge node/link outage	67%	55%
simultaneous outages	11%	18%
unknown cause	22%	18%
error	0%	9%

Table 1: Root cause analysis of the detected events. The table shows the proportion of different root causes for the decomposition and Holt-Winters approaches.

normally exit the network at this point will be rerouted to alternative points, resulting in changes in the SNMP traffic, and BGP routes. The majority of such events are part of planned maintenance (for instance for software updates), and occur during the maintenance window early in the morning when not much traffic will be effected, if there is any impact at all. By correlation with operational work flow information, such false alarms may be filtered out. In addition, the vast majority of such events occur in isolation, and are easily detected through standard mechanisms such as SNMP traps/polls of the routers, and so, they are not actually of great interest here. Note also that we do not detect all such events, because in many cases the actual change in traffic volumes, and/or BGP routes may not be that significant (for instance the failure of a peering link to a small peer).

In the process of analyzing the above events we found a small percentage which were the result of simultaneous link/node outages. Such simultaneous outages might be more serious than single outages because the network is designed to deal with single component failures. Some scenarios involving multiple points failing simultaneously might result in excess loads being placed in parts of the network as traffic is rerouted. Note that once again, both Holt-Winters and the decomposition techniques came up with the same list of simultaneous outages.

Finally, we also found a small proportion of events that had no apparent cause in the trouble ticket data. The cause of these events remains a mystery, but they appear to be genuine anomalies (they are not the result of missing, or obviously erroneous data). Such events could have two causes: problems in the network that are not detected via other means, or with more likelihood network problems on adjacent peers. In either case the events are interesting to network operations. In the former, it would allow better network management, in the latter case it gives us a window onto the behavior of the greater Internet. Holt-Winters detected only half of these events, and in addition detected a small number of events that seem to be artifacts of the detection mechanism, and are not readily apparent in the data.

Given the above (the fact that the majority of events correspond to causes that we don't need to detect here), proportion of false alarms drawn from Table 1 seems quite large (64-67%), but note that the frequency or rate of events is very low. A false alarm rate of at most a couple of events detected per month (as here) is acceptable.

Now compare these result with the false alarm rates for the individual data sources. Table 2 shows these results, based on the more than one years worth of data we have available. Because we don't have fault tickets for this entire period, we infer a false alarm from the fact that both data sources do not signal alarms. This is clearly only an upper bound on the false alarm rate, as some of these alarms may fall into the categories above (i.e., single edge node failures), but the vast majority of such events are false alarms (tested on a sizable sample of these events). We compute these false alarm rates based both on the proportion that happen per PoP per hour over all the eight PoPs considered here. We examine these false alarm rates for the two SNMP detection algorithms, and for the EWMA algorithm on the BGP data. Note that the latter algorithm has the lowest false alarm rate. Holt-Winters has a slightly higher false-alarm rate than the decomposition technique.

data set	algorithm	false alarm rate	expected false alarms per day
SNMP	Decomposition	3.4%	78
SNMP	Holt-Winters	4.3%	99
BGP	EWMA	0.5%	12

Table 2: False alarm rates for individual data sources.

Note that the rates in Table 2 are different from the percentages in Table 1. In the previous table we reported the percentage of reported events of each type. In Table 2 we report the proportion of *tests* that return a false alarm. That is, around three in one hundred decomposition test return a false alarm. Assuming only 8 PoPs, and five minutes measurements (288 per day) one would expect between 12 and 99 false alarms per day (see Table 2, column 4). These are supposed to major events, and given high priority treatment, and the amount of work required to determine the cause of each alarm is large. Such a large number of false alarms is unreasonably high, and would result in the anomaly detection being switched off, or ignored. It would be even worse for a larger scale network, or tests performed at a finer level of detail. Relative to the number of tests performed, the percentages in Table 1 would be *very* small, producing fewer than an event per week. This demonstrates this paper's main point — we can get a very large false alarm reduction using the combination of data sources. The reduction is greater than a factor of one hundred.

4.3 Discussion

The above results suggest that the decomposition technique is a little better than Holt-Winters — both have perfect sensitivity, however, the false alarm rate for the Holt-Winters technique is slightly higher when we use the SNMP data alone. Furthermore, although, in Table 1, they appear to have the same selectivity, notice that in the Holt-Winters test some of the nominal false alarms are generated by errors, whereas all of those in the decomposition technique are generated by real data events, even though some are not interesting. Further note that the decomposition found more of the interesting, unknown events. Hence we conclude that the decomposition technique is superior to the Holt-Winters approach in this context, though the improvement is minor in comparison with the benefit of combining the two data sources.

Of course one can consider improving the algorithms used for detection of both SNMP and BGP events to reduce the false alarm rate individually. However, there are fundamental problems with this approach. There are artifacts in the data, as well as genuine anomalies in both data sets that are the result of causes that we are not interested in detecting. For instance, the SNMP data may naturally vary for a number of reasons we do not consider to be network problems, at least in the sense that there is nothing we can do about them using current IP infrastructure: for instance traffic may vary from its normal patterns on holidays. Similarly BGP routes are not all responsible for equal traffic volumes. A large number of routes may change, but if these routes do not carry much traffic, there will be little impact on the network. An additional concern is problems in the data: SNMP data can contain erroneous or missing data, which could appear as untoward spikes, or dips in the data. The BGP data also contains artifacts – for instance, if a BGP session between peers is reset, the number of routes from this table drops temporarily to zero, even if they are quickly replaced, so that the event has only a small impact on traffic.

By combining multiple sources of data we gain a specific, and powerful way of detecting forwarding anomalies, which avoids any of the above pitfalls.

5. CONCLUSION

This paper has described an important class of network anomalies — forwarding anomalies — and specific methods for combining routing and traffic data to perform accurate forwarding anomaly detection. The choice of data sets used here was motivated by a specific example, but we found them to be a good choice — the method has a perfect detection rate, while dramatically reducing the false alarm rate. Moreover, the method is automated and self-training — essential characteristics for deployment in large operational networks.

Further, we found several events in the data, using this technique, that had not been previously detected through any other alarming mechanism, and are therefore worthy of further study in their own right. These events may be the result of undiagnosed internal network problems, but with more likelihood are the result of external network changes, outside the normal view of the single domain under investigation. Thus we may be able to gain a window into major external Internet changes.

The method is very simple, but this should be seen as an advantage, as simplicity makes the method more scalable, and more easily extendible to include additional data sets, for instance, OSPF routing data, active network probes, router logs, or flow level data. A simple example might be the inclusion of network management information on planned outages so that we can exclude these from the detection algorithm. One could also improve the algorithms applied here to individual data sets, for instance by applying the wavelet techniques of [13], and there are also many different ways in which data from these various sources could be combined. We have used a very simple method for data fusion, but there is a great deal of literature in the areas of pattern recognition, detection and classification that is relevant to this task. Finally, we have considered only one class of network anomalies here — forwarding anomalies, but it is likely this sort of technique could be extended to other sorts of anomalies, for instance of application in security.

Acknowledgements

We would like to acknowledge many useful comments made by Yin Zhang, Jennifer Rexford, Michael Rumsewicz and others. We would also like to thank those involved in collecting some of the data used, in particular Fred True, Joel Gottlieb, and Carsten Lund.

6. REFERENCES

[1] A.Brown and D. A. Patterson, "To err is human," in *Proceedings of the First Workshop on Evaluating and Architecting System dependability (EASY '01)*, 2001.

[2] D. Patterson, A. Brown, P. Broadwell, G. Candea, M. Chen, J. Cutler, P. Enriquez, A. Fox, E. Kiciman, M. Merzbacher, D. Oppenheimer, N. Sastry, W. Tetzlaff, J. Traupman, and N. Treuhaft, "Recovery-oriented computing (roc): Motivation, definition, techniques, and case studies," Tech. Rep. UCB//CSD-02-1175, UC Berkeley Computer Science, 2002.

[3] D. Oppenheimer, A. Ganapathi, and D. A. Patterson, "Why do Internet services fail, and what can be done about it?," in *4th Usenix Symposium on Internet Technologies and Systems (USITS'03)*, 2003.

[4] D.J.Houck, K.S.Meier-Hellstern, F.Saheban, and R.A.Skoog, "Failure and congestion propagation through signalling control," in *Proceedings of the 14th International Teletraffic Congress (ITC-14)*, vol. 1a, pp. 367–376, Elsevier, 1994.

[5] J. Strand, A. Chiu, and R. Tkach, "Issues for routing in the optical layer," *IEEE Communications Magazine*, Feb., 2001.

[6] Nanog mailing list `http://www.cctec.com/maillists/nanog/historical/0005/msg00073.html`, 5th May 2000.

[7] Nanog mailing list: `http://www.cctec.com/maillists/nanog/historical/0210/msg00058.html`, 3rd Oct. 2002.

[8] "University of Oregon Route Views Archive Project." `www.routeviews.org`.

[9] C. Hood and C. Ji, "Proactive network fault detection," *IEEE Trans. Reliability*, vol. 46, no. 3, pp. 333–341, 1997.

[10] M. Thottan and C. Ji, "Proactive anomaly detection using distributed intelligent agents," *IEEE Network*, Sept/Oct 1998.

[11] A. Ward, P. Glynn, and K. Richardson, "Internet service performance failure detection," *ACM SIGMETRICS Performance Evaluation Review archive*, vol. 26, pp. 38–43, December 1998.

[12] J. D. Brutag, "Aberrant behavior detection and control in time series for network monitoring," in *Proceedings of the 14th Systems Administration Conference (LISA 2000)*, (New Orleans, LA, USA), USENIX, December 2000.

[13] P. Barford, J. Kline, D. Plonka, and A. Ron, "A signal analysis of network traffic anomalies," in *ACM SIGCOMM Internet Measurement Workshop*, (Marseilles, France), Nov., 2002.

[14] B. Krishnamurthy, S. Sen, Y. Zhang, and Y. Chen, "Sketch-based change detection: Methods, evaluation, and applications," in *ACM SIGCOMM Internet Measurement Conference*, (Miami, Florida, USA), October 2003.

[15] Katzela and Schwartz, "Schemes for fault identification in communication networks," *IEEE/ACM Transactions on Networking*, vol. 3, 1995.

[16] M. Grossglauser, N. Koudas, Y. Park, and A. Variot, "Falcon: Fault management via alarm warehousing and mining," in *NRDM 2001 workshop*, (Santa Barbara, CA), May 2001.

[17] R. NCC, "Routing Information Service Raw Data." `http://data.ris.ripe.net/`.

[18] S. H. Steiner, "Grouped data exponentially weighted moving average control charts," *Applied Statistics*, vol. 47, no. 2, 1998.

[19] M. Roughan, A. Greenberg, C. Kalmanek, M. Rumsewicz, J. Yates, and Y. Zhang, "Experience in measuring Internet backbone traffic variability: Models, metrics, measurements and meaning," in *Proceedings of the International Teletraffic Congress (ITC-18)*, 2003.