

TCP Anomalies: identification and analysis

Marco Mellia, Michela Meo and Luca Muscariello *

Dipartimento di Elettronica, Politecnico di Torino, Torino, Italy
{mellia,meo,muscariello}@mail.tlc.polito.it

Abstract. Passive measurements have recently received large attention from the scientific community as a mean, not only for traffic characterization, but also to infer critical protocol behaviors and network working conditions. In this paper we focus on passive measurements of TCP traffic, main component of nowadays traffic. In particular, we propose a heuristic technique for the classification of the anomalies that may occur during the lifetime of a connection. Since TCP is a closed-loop protocol that infers network conditions and reacts accordingly by means of losses, the possibility of carefully distinguishing the causes of anomalies in TCP traffic is very appealing and may be instrumental to the deep understanding of TCP behavior in real environments and the protocol engineering.

1 Introduction

In the last ten years, the interest in data collection, measurement and analysis to characterize Internet traffic behavior increased steadily. Indeed, by acknowledging the failure of traditional modeling paradigms, the research community focused on the analysis of the traffic characteristics with the twofold objective of understanding the dynamics of traffic and its impact on the network elements, and of finding simple, yet satisfactory, models, like the Erlang teletraffic theory in telephone networks, for designing and planning packet-switched data networks.

The task of measuring Internet traffic is particularly difficult for a number of reasons. First, traffic analysis is made very hard by the strong correlations both in space and time due to the closed-loop behavior of TCP, the TCP/IP client-server communication paradigm, and the fact that the highly variable quality provided to the end user influences the user behavior. Second, the complexity of the involved protocols, and of TCP in particular, is such that a number of phenomena can be studied only if a deep knowledge of the protocol details is exploited. And, finally, some of the traffic dynamics can be understood only if the forward and backward directions of flows are jointly analyzed.

From what mentioned above, it is clear that, since TCP plays a central role in the generation of Internet traffic, measurement tools should be equipped with modules for the analysis of TCP traffic, which do not neglect the occurrence of

* This work was founded by the European Community “euroNGI” Network of Excellence.

all those anomalies that strongly influence TCP behavior. In this context, the objective of this paper is to propose a new heuristic classification technique of anomalies that may occur during the lifetime of a TCP connection. In [1] a simple but efficient classification algorithm for out-of-sequence TCP segments is presented. The classification proposed in this paper is a modification and extension of that classification which allows the identification of a number of phenomena which were not previously considered, such as unneeded retransmissions and flow control mechanisms.

The proposed classification technique is applied to a set of real traces collected at our institution. The results show that a number of interesting phenomena can be observed through the proposed classification, such as the impact of the use of TCP SACK on the occurrence of unnecessary retransmissions, the relative small impact of the daily variation of the load on the occurrence of anomalies, the quite large amount of network reordering.

2 Methodology

The methodology adopted in this paper is a modification and extension of the one proposed for the first time in [1], in which authors proposed a simple but efficient classification algorithm for out-of-sequence packets in TCP connections and presented measurement results within the Sprint IP backbone. Similarly to what proposed by them, we adopt a passive measurement technique rather than using active probe traffic. The classification in [1] identifies out-of-sequence events due to i) necessary or unnecessary segment retransmissions by the TCP sender, ii) network duplicates, or iii) network reordering. Building over the same idea, we complete the classification by distinguishing other possible causes of out-of-sequence or duplicate packets. In particular, we also focus on the *cause* that triggered the segment retransmission by the sender. We analyze packet traces which record packets in both directions of a TCP connection: both data segments and ACKs are recorded.

Figure 1 sketches the evolution of a TCP connection: connection setup, data transfer, and connection tear down phases are highlighted. The measurement point (sniffer) is located in some point in the path between the Client and the Server. Both the IP layer and TCP layer overhead are observed by the sniffer, so that the TCP sender status can be tracked. A TCP connection starts when the first SYN from the client is observed, and terminates after either the tear-down sequence (the FIN/ACK or RST messages), or when no segment is observed for an amount of time larger than a given threshold ¹.

¹ Given that a tear-down sequence of a flow under analysis is never observed, a timer is needed to avoid memory starvation; the timer is set to 15 minutes, which is sufficiently large according to the findings in [2].

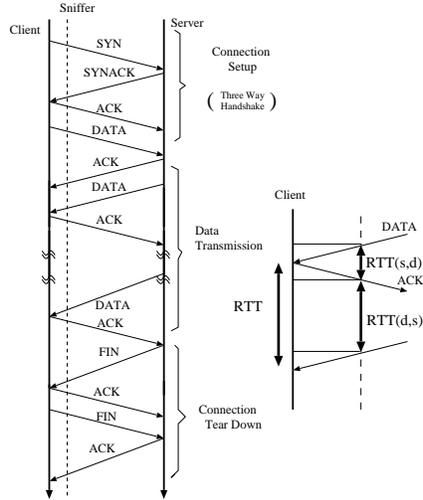


Fig. 1. Evolution of a TCP connection and the RTT estimation process.

By tracking the segment trace of a connection in both directions², the sniffer correlates the sequence number of TCP data segments to the ACK number of backward receiver acknowledgments and classifies the segments as,

- *In-sequence*: if the sender initial sequence number of the current segment corresponds to the expected one;
- *Duplicate*: if the data carried by the segment have already been observed before;
- *Out-of-sequence*: if the sender initial sequence number is not the expected one, and the data carried by the segment have never been observed before.

The last two classifications refer to an *anomaly* during the data transfer, that can have been caused by several reasons. We propose a fine heuristic classification of the anomalies. The classification has been implemented in [6, 7], and then used to analyze collected data. During the decision process that is followed to classify anomalies, several variables are used:

RTT_{min} Minimum RTT: is the estimated minimum RTT observed since the flow started;

RT Recovery Time: is the difference between the time the current anomalous segment has been observed and the time the segment with the *largest* sequence number has been seen;

ΔT Inverted-packet gap: is the difference between the observation time of the current anomalous segment and the previously received one;

RTO Retransmission Timeout: is estimated sender retransmission timer value (in seconds) according to [3] as $RTO = \max(1, E[RTT] + 4std(RTT))$;

² In case only one direction of traffic is observed by the sniffer, the heuristic will not be applicable.

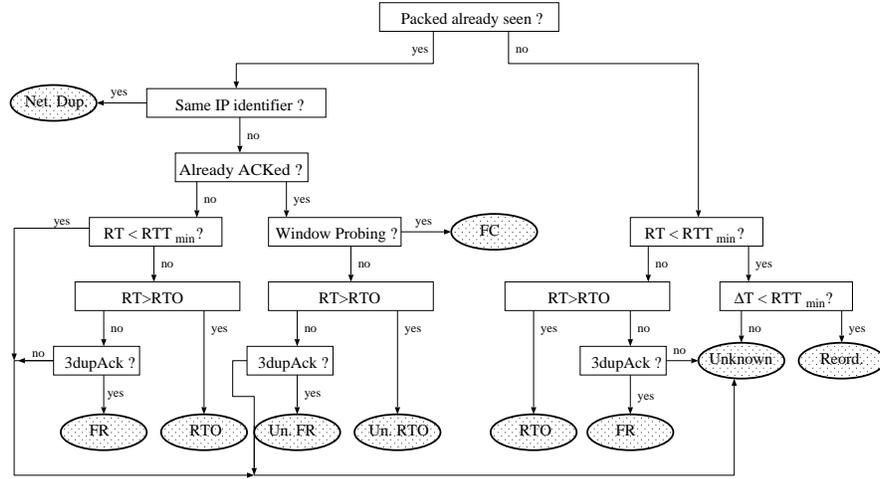


Fig. 2. Decision process of the classification of anomalous segments.

DupAck Number of DupAcks: is the number of duplicate ACKs observed on the reverse path.

Following the flow diagram of Figure 2, we describe the classification heuristic. Given an anomalous segment, the process starts by checking if the segment has already been seen by comparing the TCP sequence number of the current segment with the one carried by segments observed so far. Thus, the segment can be classified as either duplicate or out-of-sequence. In the first case (left branch of the decision process), the IP identifier field of the current packet is compared with the same field of the original copy of the packet. If they are the same, then the packet is classified as **Network Duplicate** (Net. Dup.). Network duplicates may stem from malfunctioning apparatuses, routing loops, mis-configured networks (e.g., Ethernet LANs with diameter larger than the collision domain size), or, finally, by unnecessary retransmissions at the link layer (e.g., when a MAC layer ACK is lost in a Wireless LAN forcing the sender to retransmit the frame). Compared with the decision process adopted in [1], we classify as duplicate segments all those duplicate packets with the same IP identifier, regardless of the ΔT value. Indeed, there is no reason to exclude that a network duplicate may be observed at any time, and there is no relation between the RTT and the time a network can produce some duplicate packets.

When the IP identifiers are different, the TCP sender may have performed a retransmission. If all the bytes carried by the segment have already been acknowledged, then the receiver has already received the segment, and therefore this is an unneeded retransmission. The flow control mechanism adopted by TCP uses false unneeded retransmissions to perform *window probing*, i.e., to force the receiver to immediately send an ACK so as to probe if the receiver window $RWND$ (which was announced to be zero on a previous ACK) is now

larger than zero. Therefore we classify as **Flow Control** (FC) retransmissions the retransmitted segments for which the following three conditions hold: i) the sequence number is equal to the expected sequence number decreased by one, ii) the segment size is of zero length, and iii) the last announced *RWND* in the ACK flow was equal to zero. This is a new possible cause of unneeded retransmissions which was previously neglected in [1].

If the anomaly is not classified as flow control, then it must be an unnecessary retransmission, which could have been triggered because of either a Retransmission Timer (*RTO*) has fired, or the fast retransmit mechanism has been triggered, i.e., three or more duplicate ACKs have been received for the segment before the retransmitted one. We identify three situations: i) if the recovery time is larger than the retransmission timer ($RT > RTO$) the segment is classified as an **Unneeded Retransmission by RTO** (Un. RTO); ii) if 3 duplicate ACKs have been observed, the segment is classified as an **Unneeded Retransmission by Fast Retransmit** (Un. FR); iii) otherwise, if none of the previous conditions holds, we do not know how to classify this segment, and therefore we label it as **Unknown** (Unk.). Unneeded retransmissions may be due to a misbehaving sender, a wrong estimation of the *RTO* at the sender, or, finally, to ACKs lost on the reverse path. However, distinguishing among these causes is impossible by means of passive measurements.

Let us now consider the case of segments that have already been seen but have not been ACKed yet. This is possibly the case of a retransmission following a packet loss. Given the recovery mechanism adopted by TCP, a retransmission can occur only after at least a *RTT*, since duplicate ACKs have to traverse the reverse path and trigger the Fast Retransmit mechanism. Therefore, if the recovery time is smaller than RTT_{min} , the anomalous segment can only be classified as **Unknown**³. Otherwise, it can either be a **Retransmission by Fast Retransmit** (FR) or **Retransmission by RTO** (RTO); the classification being based on the same criteria adopted previously for unneeded retransmissions. Retransmissions of already observed segments may be due to i) data segments lost on the path from the measurement point to the receiver, and to ii) ACK segments delayed or lost before the measurement point.

Consider now the right branch of the decision process, which refers to out-of-sequence anomalous segments. In this case, the classification criterion is simpler. Indeed, out-of-sequence segments can be due to either the retransmission of lost segments, or to network reordering. Again, since retransmissions can only occur if the recovery time *RT* is larger than RTT_{min} , by double checking the number of observed duplicate ACKs and by comparing the recovery time with the estimated *RTO*, we can distinguish retransmissions triggered by RTO, by FR or we can classify the segment as an unknown anomaly. On the contrary, if *RT* is smaller than RTT_{min} , then a **Network Reordering** (Reord) is identified if the inverted-packet gap is smaller than RTT_{min} . Network reordering can be

³ In [1] the authors use the average *RTT*. However, being each RTT possibly different than the average *RTT* (and in particular smaller), we believe that using the average *RTT* forces a larger amount of misclassification.

due to either load balancing on parallel paths, or to route changes, or to parallel switching architectures which do not ensure in-sequence delivery of packets [4].

2.1 Dealing with wrong estimates

The classification algorithm uses some thresholds whose values must be estimated from the packet trace itself, which may not be very accurate or even valid when classifying the anomalous event. Indeed, all the measurements related to the RTT estimation are particularly critical, since they are used to determine the RTT_{min} and the RTO estimation. RTT measurement is updated during the flow evolution according to the moving average estimator standardized in [3]. Given a new measurement m of the RTT, we update the estimate of the average RTT by mean of a low pass filter $E[RTT]_{new} = (1 - \alpha)E[RTT]_{old} + \alpha m$ where α ($0 < \alpha < 1$) is equal to $1/8$.

Since the measurement point is not co-located at the transmitter, nor at the receiver, the measure of RTT is not available. Therefore, in order to get an estimate of the RTT values, we build over the original proposal of [1]. In particular, denote by $RTT(s, d)$ the *Half path RTT sample*, which represents the delay at the measurement point between an observed data segment flowing from the transmitter, or source, to the receiver, or destination, and the corresponding ACK on the reverse path, and denote by $RTT(d, s)$ the delay between the ACK and the following segment, as shown in Figure 1. From the measurement of $RTT(s, d)$ and $RTT(d, s)$ it is possible to derive an estimate of the total round trip time RTT ,

$$RTT = RTT(s, d) + RTT(d, s)$$

The estimation of the average RTT is not biased, given the linearity of the expectation operators. Therefore, it is possible to estimate the *average RTT* by,

$$E[RTT] = E[RTT(s, d)] + E[RTT(d, s)]$$

Moreover, the standard deviation of the connection's RTT, $std(RTT)$ can be estimated following the same approach, given that $RTT(s, d)$ and $RTT(d, s)$ are independent measurements, which usually holds.

Finally, given $E[RTT]$ and $std(RTT)$, it is possible to estimate the sender retransmission timer as in [3]:

$$RTO = E[RTT] + 4std(RTT)$$

For what concerns the estimation of minimum RTT, RTT_{min} , we have

$$RTT_{min} = \min(RTT(s, d)) + \min(RTT(d, s))$$

In general, this estimator gives a conservative estimation of the real minimum RTT , as $RTT_{min} \leq \min(RTT)$ holds. This leads to a conservative classification algorithm, which increases the number of anomalies classified as unknown, rather than risking some misclassifications.

2.2 Handling Particular Cases

No RTT Sample Classification. There are some cases in which the *RTT* measurement is not available, but an anomalous event is detected. This happens in particular at the startup of a TCP connection, as no valid *RTT* samples may be available at the very beginning of the connection. Since most of TCP flows are very short [7], these events are quite frequent and cannot be neglected. Moreover, the choice of the initial values of *RTO* and *RTT_{min}* results to be critical, and inappropriate estimations of these variables may lead to wrong classifications. We adopt the following approach:

- if no valid RTT samples have been collected, the heuristic uses $RTO = 3s$ and $RTT_{min} = 5ms$ as default values
- the *RTO* estimation is forced to assume values larger or equal to 1s, according to [3]
- the *RTT_{min}* estimation is forced to be larger than 1ms

Batch Classification. Given that TCP can transmit more than one segment per RTT, it may happen that more than one anomalous segments are detected back-to-back. This occurs, for example, when the TCP sender adopts the SACK extension and retransmits more than one segment per RTT, or, when packets belonging to the same window on a path in which packets are reordered, arrive with “strange” patterns difficult to be identified. In such cases, the measurement of *RT* and ΔT may be wrong and lead to incorrect classifications. We therefore implement a filter in the classification heuristic that correlates the classification of the current anomaly with the classification of the previous segment. In particular, if the current recovery time *RT* is smaller than $E[RTT]$ (suggesting that the segment is belonging to the same window as the previous one) and the previous segment was not classified as in sequence, we then classify the current anomalous segment as the previous one.

For example, consider a simultaneous SACK retransmission of two segments triggered by a Fast Retransmit. The first retransmitted segment is correctly classified given that three duplicate ACKs have been observed on the reverse path, and the *RT* is larger than *RTT_{min}*. However, the second retransmitted segment cannot be correctly classified, given that no duplicate ACK has ever been observed, and $RT < RTT$. By explicitly considering the classification of the first segment, it is possible to correctly identify this segment as a retransmission triggered by Fast Retransmit.

3 Measurement Results

Our measurements have been gathered from the external Internet edge link of our institution. Our campus network behaves like an Internet stub, because the access router is the sole gate to the external network. Our institution counts more than 7,000 hosts, whose great majority is constituted by clients. Some servers are regularly accessed from outside as well. We collected all packets flowing into the

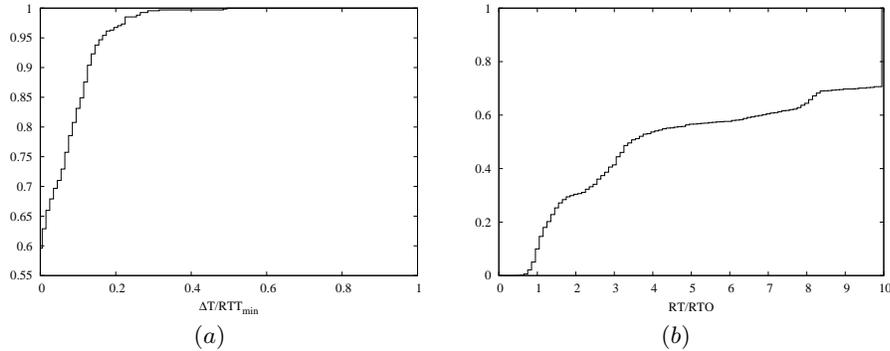


Fig. 3. (a) CDF of the ratio between the inverted packet gap ΔT and RTT_{min} . (b) CDF of the ratio between the recovery time RT and the actual estimation of the RTO.

access link that connects the campus border router to the GARR network [5], the nation-wide ISP for research and educational centers. We measured for several months during several time periods and gathered the most interesting statistics related to the anomalous traffic. We present only a subset of results, and in particular:

- from the 6th to the 7th of February 2001. The bandwidth of the access link was 14 Mbit/s;
- from the 29th of April to the 5th of May 2004. The bandwidth of the access link was 28 Mbit/s.

3.1 Impact of RTT_{min} and RTO

We first lead a set of measurements to double-check the impact of the choices described in Sec 2.1. In particular, we are interested in the impact of the measurement of RTT_{min} and RTO . The first one is involved on the classification of the network reordering anomalies that may occur when identifying two out-of-sequence segments separated by a time gap smaller than RTT_{min} . Figure 3 (a) plots Cumulative Distribution Function (CDF) of the ratio between the inverted packet gap ΔT and the value of the RTT_{min} considering only TCP anomalies classified as network reordering. Measurements referring to 2004 are reported, and similar results are obtained considering the 2001 dataset. The CDF clearly shows that ΔT is much smaller than the RTT_{min} . This suggests that the initial choice of RTT_{min} is appropriate, and the conservative estimation of RTT_{min} does not affect the classification.

Figure 3 (b), reports part of the CDF of the ratio between the actual Recovery Time RT and the corresponding estimation of the RTO when considering anomalous events classified as retransmissions by RTO. Also in this case we report results referring to the 2004 dataset. The CDF shows that $RT > RTO$ holds, which is a clear indication that the estimation of the RTO is not critical.

Moreover, it can be noted that about 50% of the cases have a recovery time which is more than 5 times larger than the estimated RTO. This apparently counterintuitive result is due to the RTO back-off mechanism implemented in TCP but not considered by the heuristic which doubles the RTO value at every retransmission of the same segment. Not considering the back-off mechanism during the classification lead to a robust and conservative approach.

3.2 Aggregate Results

In the following we report results obtained by running the classification heuristic over the two datasets we selected and by measuring the average number of occurred anomalies during the whole time period. The objective is twofold. First, we quantify the different causes that generated anomalous segment delivery; second, we double check the heuristic classification. Indeed, it is impossible to test the validity of the classification algorithm, given that the real causes of the anomaly are unknown. We therefore run the classification over real traces, and try to underline some expected and intuitive results that confirm the validity of the heuristic design.

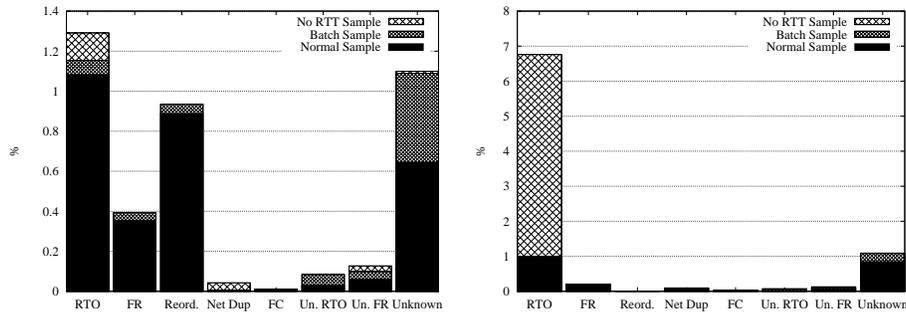


Fig. 4. Classification of anomalous events: incoming traffic on the left , outgoing traffic on the right.

Figure 4 reports the percentage of identified anomalous events during the 2004 period over the total amount of observed segments. Left plot refers to incoming traffic, i.e., traffic whose destination host is inside our campus LAN; right plot reports measurements on outgoing traffic, i.e., traffic whose destination host is outside our campus LAN. Each bar in the plot explicitly underlines the impact of the batch classification and of the lack of RTT samples, as described in Section 2.1: solid black blocks report the anomalies classified by a normal classification, while dark pattern blocks report the impact of the batch classification, and, finally, light pattern blocks report the classification obtained when no RTT sample was available, i.e., at the very beginning of each flow.

Considering the incoming anomalies classification (left plot), we observe that there is a large dominance of retransmissions due to RTO expiration and re-ordering. Fast Retransmit occurs only for a very small portion of the total retransmissions. This is related to the characteristics of today data traffic, which is mainly composed of very small file transfers that cannot trigger Fast Retransmit. This effect is further stressed by the fact that our campus LAN traffic is mainly made of web browsing applications, whose (short) HTTP requests travel on the outgoing directions.

A small percentage of unnecessary retransmissions is also present. A rather large percentage of anomalies that could not be classified but unknown is collected. Inspecting further, we observed that for most of them the recovery time is smaller than the estimated RTO (therefore missing the retransmission by RTO classification), but larger than the RTT_{min} (therefore missing the reordering classification) and the number of duplicate ACKs is smaller than 3 (therefore missing the retransmission by FR classification). We suspect that they may be due to either i) transmitters that trigger the Fast Retransmit with just 1 or 2 duplicate ACKs, or ii) servers with aggressive RTO estimation that triggers the retransmission earlier than the RTO estimation at the measurement point. Given the conservative approach that guided the classification heuristic, we prefer to classify them as unknown rather than misclassifying them.

For what concerns the impact of the batch classification and of the lack of a valid RTT sample, observe that the first is evenly distributed among all classification cases, while the latter one has a large impact on the identification of retransmissions by RTO. This is due to the lack of valid RTT samples at the very beginning of the TCP connection, when the sender can only detect packet losses by RTO.

When considering the outgoing anomalies (right plot), the heuristic correctly identifies the anomalies, and neither Network Reordering nor Duplicates are identified. Indeed, this is quite obvious given that our institution LAN is a switched Ethernet LAN. In this network, IP packets can be duplicated or re-ordered only in case of malfunctioning. This confirms the validity and robustness of the classification heuristic we developed.

Considering the average percentage of total anomalies identified, we have that about 4% and 8% of incoming and outgoing traffic respectively is affected by an anomaly. We will see in the next section that the average values is not representative at all, given the non-stationaries of the anomalies.

Finally, in order to double check the validity of our heuristic, we split the flow into three different classes based on their segment length. *Short* flows (also called mice in the literature) have payload size (in segments) no longer than 5 segments. *Long* flows (the so called elephants) have payload size (in segments) larger than 20 segments, and the *middle* length flows payload size is larger than 5 segments, but shorter or equal to 20 segments. Figure 5 reports the classification of the anomalous events split among the three different classes. Incoming segment classification is considered for the 2004 time period. Solid black refers to short flows, dark gray pattern refers to middle flows, and light gray pattern refers to

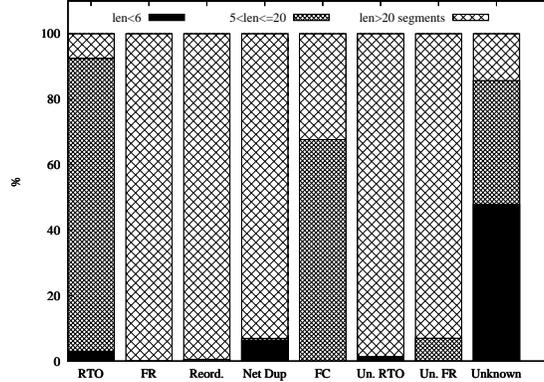


Fig. 5. Classification of anomalous events for different flow length; incoming traffic for 2004 measurements.

long flows. For the sake of clarity, we omit the further batch or no rtt sample classification.

As expected, retransmissions due to RTO expiration are distributed among all flows, while retransmissions due to Fast Recovery are only triggered for long flows. This is intuitive, as already said, because of the limit in triggering Fast Retransmit by short flows. The majority of packet reordering affects long flows, for which the chance to suffer from a reordering is much larger. Neglecting the network duplicates, flow control and unnecessary retransmissions as they are very marginal, we observe that the anomalies classified as unknown are largely related to short flows. Indeed, this is an hint that the $E[RTT]$ estimation is affected by a larger error for short flows, while long flows have the chance to get a better estimation of the RTT and therefore to better classify the anomaly. This confirms the intuition that the unknown classification is related to possible different estimation of the RTO at the transmitter and at the measurement point.

Results relatively to outgoing flows and to the 2001 dataset are very similar and therefore not reported here.

3.3 Behavior in Time

We report in this section the results of the occurrence of anomalies in time. Again, we omit the sub-classification due to batch or no RTT samples for the sake of clarity. Figures 6 and 7 depict the time evolution of the volume (in percentage, normalized on the total flowed traffic in each direction) of anomalous measured segments classified by the proposed heuristic. Measurements aggregate anomalies over an interval of time equal to 15 minutes. The detailed classification is outlined in colored slices whose size is proportional to the percentage of that particular event. Top plot refers to the incoming traffic; bottom plot reports measurements considering outgoing traffic. Figure 6 refers to the three

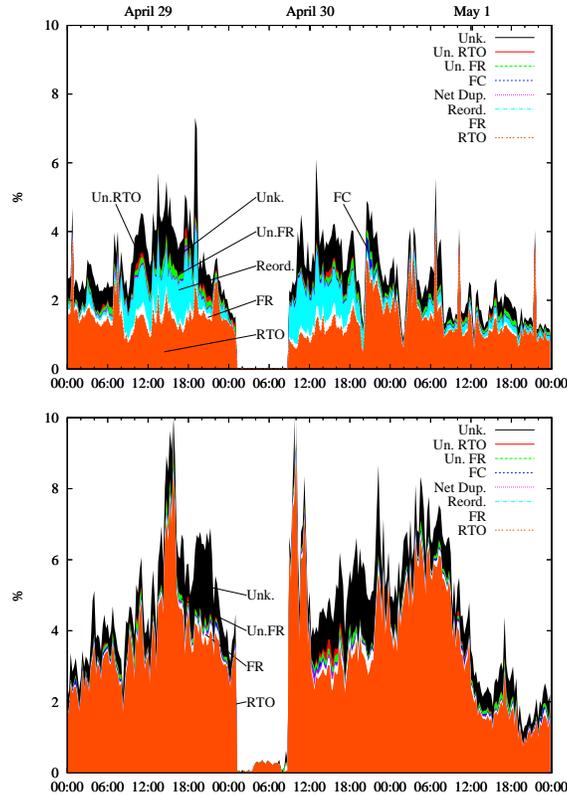


Fig. 6. Classification of anomalous events versus time: incoming traffic on the left, outgoing traffic on the right for 2004 measurements.

days evolution of such traffic continuously monitored and classified during the 2004 period, while, similarly, Figure 7 refers to the two day long subset of measurements in 2001.

Apart from the network outage that is evident, the first unambiguous results is that TCP anomalies are highly non stationary over several time scales. There are some peaks of very significant magnitude that reach 10% during 2004 and 15% during 2001. Considering the incoming segments, Retransmissions by RTO, Network Reordering and Unknowns are the largest part of the anomalies, while TCP Flow control seldom kicks in, and negligible Unnecessary Retransmissions are identified. Surprisingly, the typical night and day effect, which is commonly present on the total traffic volume (and is valid also in the considered link), is not anymore visible when considering TCP anomalies. Notice also that the last two measurement days are weekend-days. In particular, the Labour Day is celebrated on Sunday the 1st of May in Italy. Therefore the link load during that weekend was particular low. Nonetheless, the RTO fraction is almost equal to the one observed during busy hours of weekdays. Only the Network Reordering

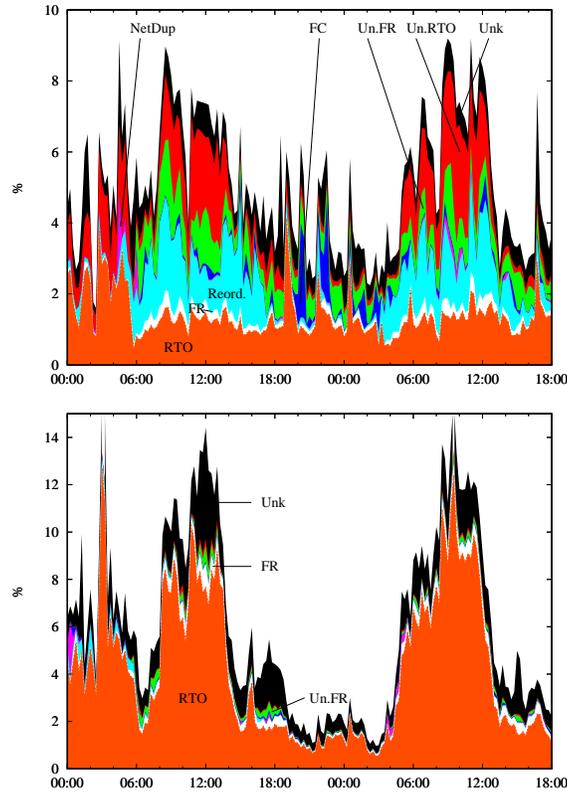


Fig. 7. Classification of anomalous events versus time: incoming traffic on the top, outgoing traffic on the bottom for 2001 measurements.

seems to disappear. This hints to a weak correlation between link load and TCP anomalies.

Considering the outgoing traffic (bottom plots of Figure 6 and 7), observe that the heuristic correctly identifies the anomalies as retransmission by RTO. Given that hosts in our campus LAN are mainly clients of TCP connections, the outgoing flow size is very short, and therefore in case of a packet loss, the only way to recover is to fire the RTO.

If we compare 2004 and 2001 incoming time plots, it can be noted that after three years the number of Unnecessary Retransmissions almost disappears. This fact is explained by the vast popularity of TCP SACK flows that were only the 21% of total flows during 2001 while it increased to 90% of total flows during 2004. At the same time, a reduction of the fraction of TCP anomalies is noticeable comparing 2001 and 2004 measurements. This could be related to the corresponding increase in the access link capacity, which doubled in 2004.

4 Conclusions

In this paper, we proposed a heuristic technique for the classification of TCP anomalies. The classification identifies seven possible causes of anomalies and extends previous techniques already proposed in the literature. We were also able to quantify the quality of our classification by studying the sensitivity of our approach in estimating the RTT which strongly influences the proposed methodology which gathers every flow state information from passive measurements and the quantitative analysis show its effectiveness in the identification procedure.

The technique was implemented and tested on the external Internet link of our institution and allowed the observation of a number of interesting phenomena such as the impact of the use of TCP SACK on the occurrence of unnecessary retransmissions, the relative small impact of the daily variation of the load on the occurrence of anomalies, the quite large amount of network reordering anomalies.

References

1. S. Jaiswal, G. Iannaccone, C. Diot, J. Kurose, D. Towsley, "Measurement and Classification of Out-of-Sequence Packets in a Tier-1 IP Backbone", IEEE Infocom, San Francisco, March 2003.
2. G.Iannaccone, C.Diot, I.Graham, and N.McKeown, "Monitoring very high speed links," *ACM Internet Measurement Workshop, San Francisco*, November 2001.
3. V. Paxson, M. Allman, "Computing TCP's Retransmission Timer", *RFC 2988*, November 2000.
4. J.C.R.Bennett,C.C.Partridge, N.Shectman, "Packet reordering is not pathological network behavior" em *IEEE/ACM Transactions on Networking*, Vol. 7, N. 6, pp.789-798, December 1999.
5. "GARR Network" <http://www.noc.garr.it/mrtg/RT.TO1.garr.net/polito.garr.net.html>
6. "Tstat Web Page" <http://tstat.tlc.polito.it/>
7. M. Mellia, R. Lo Cigno, F. Neri, "Measuring IP and TCP behavior on edge nodes with Tstat", *Computer Networks* 47(3): 1-21, Jan. 2005.