Vivisecting WhatsApp in Cellular Networks: Servers, Flows, and Quality of Experience

Pierdomenico Fiadino, Mirko Schiavone, Pedro Casas

Vivisecting WhatsApp in Cellular Networks: Servers, Flows, and Quality of Experience

Pierdomenico Fiadino, Mirko Schiavone, and Pedro Casas

Telecommunications Research Center Vienna - FTW
Email: {surname}@ftw.at

Abstract. Instant Multimedia Messaging (IMM) applications are increasing their popularity in cellular networks, rapidly taking over the traditional SMS and MMS messaging service. This paper presents the first large-scale characterization of WhatsApp, the new giant in IMM. Understanding how it works is paramount for cellular operators and service providers, both to assess its impact on the network as well as gaining know how for tracking its growing usage. Through the combined analysis of passive measurements at the core of a European national-wide cellular network, geo-distributed active measurements using RIPE Atlas, live traffic captures at end devices, and subjective Quality of Experience (QoE) lab tests, our study shows that: (i) the WhatsApp hosting architecture is highly centralized and exclusively located in the US; (ii) multimedia sharing covers about 75% of the total WhatsApp traffic volume, with 36% of it being video content; (iii) flow characteristics depend on the OS of the end device; (iv) despite achieving download throughputs as high as 1.5 Mbps, about 35% of the total file downloads are potentially badly perceived by the users, showing the impacts of the long latencies to WhatsApp servers. Our analysis additionally overviews the worldwide WhatsApp outage occurred in February 2014.

Keywords: WhatsApp, Large-Scale Measurements, Cellular Networks, Traffic Characterization, Quality Of Experience

1 Introduction

WhatsApp is doubtlessly the leading instant multimedia messaging service in cellular networks. WhatsApp is a cross-platform mobile application which allows users worldwide to instantly exchange text messages and multimedia contents such as photos, audio and videos. It currently handles more than 64 billion messages per day, including 700 million photos and 100 million videos [16]. With half a billion of active users, it has become the fastest-growing company in history in terms of users [15]. Such an astonishing popularity does not only have a major impact on the traditional SMS/MMS business, but might also have a remarked impact on the traffic, especially due to the sharing of multimedia messages.

The goal of this paper is to provide the first large-scale characterization of the WhatsApp service. By analyzing a week of cellular traffic flows collected in February 2014 at the cellular network of a major European ISP, we shed light on the WhatsApp hosting network architecture, the characteristics of the generated

traffic, and the performance of media transfers, specially as perceived by the end users. As WhatsApp runs on top of encrypted connections, our measurements are complemented with a dissection of the WhatsApp protocol through hybrid measurements, enabling a subsequent passive monitoring at the large-scale. In addition, due to its large worldwide popularity, the WhatsApp dataset is augmented with geo-distributed DNS active measurements using more than 600 RIPE Atlas boxes distributed around the globe [14]. As we shall see next, this paper it is not just about finding which flows belong to the WhatsApp service and analyze them. Indeed, there are many measurement challenges associated to the characterization of such a service: the data gathering, the processing and the interpretation are already very complex per se, given the number of different measurement sources and datasets.

Recent papers have partially addressed the characterization of the WhatsApp traffic [2, 3], but using very limited datasets (i.e., no more than 50 devices) and considering an energy-consumption perspective. Our study follows previous papers characterizing popular Internet services such as YouTube [6], Facebook [5], Google+ [8], Skype [4], and Dropbox [7] among others. This paper represents an extended version of a recently presented abstract on the topic [1].

Our main findings are the following: (i) Despite its worldwide popularity, **WhatsApp is a fully centralized service hosted by the cloud provider SoftLayer at servers located in the US**. (ii) While the application is mainly used as a text-messaging service in terms of transmitted flows (more than 93%), **video-sharing accounts for about 36% of the exchanged volume** in uplink and downlink, and **photo-sharing/audio-messaging for about 38%**. (iii) Despite achieving **flow download throughputs of 1.5 Mbps on average**, about **35% of the total file downloads are potentially badly perceived by users**. (iv) **Flow duration characteristics depend on the device OS**. In particular, different platforms employ different app-level timeouts.

Besides these contributions, our study also provides an overview on the worldwide WhatsApp outage reported on February the 22nd of 2014 [17], characterizing the event as observed from the analyzed dataset. The measurements are complemented with external Online Social Networks (OSNs) feeds (Twitter in this case) to verify that the outage was negatively perceived by the users, immediately at the time were the event occurred, additionally demonstrating the feasibility of using OSNs data to provide near real-time evidence of user quality impairments in large scale service outages. We believe that the information provided in this paper is highly useful for cellular operators to better understand how WhatsApp works and performs, and specially to provide means for analyzing and tracking its evolution inside their networks, including a Quality of Experience (QoE) perspective. To the best of our knowledge, we are the first to provide a large-scale characterization of the complete WhatsApp service running on its live environment.

The remainder of this paper is organized as follows: §2 briefly describes the client application work-flow in terms of exchanged messages and server roles as identified from hybrid end-device measurements. §3 explains the procedure used

domain	protocol (port)	type
cX,eX,dX	XMPP (5222, 443)	chat & control
mmiXYZ,mmsXYZ	HTTPS (443)	media (photo/audio)
mmvXYZ	HTTPS (443)	media (video)

Table 1. Third level domain names used by `whatsapp.net` and communication types.

to detect the WhatsApp flows in the large-scale cellular passive measurements, and characterizes the underlying hosting network. The analysis of the WhatsApp traffic is presented in §4, including both the flow characteristics per communication and end-device types and the performance in terms of transfer throughputs. §5 presents the results of subjective QoE tests performed with customers downloading multimedia files through WhatsApp, and applies them to the analyzed large-scale traffic dataset. §6 provides an overview of the WhatsApp outage. Finally, §7 concludes this work.

2 An Overview on WhatsApp

WhatsApp uses encrypted communications, therefore the first step to analyze its functioning in the wild is to better understand its inner working. To this end, we rely on the manual inspection of hybrid measurements. We actively generate WhatsApp text and media flows at end devices (both Android and iOS), and passively observe them at two instrumented access gateways. We especially paid attention to the DNS traffic generated by the devices.

WhatsApp uses a customized version of the open eXtensible Messaging and Presence Protocol (XMPP) [20]. XMPP is a protocol for message oriented communications based on XML. Not surprising, our measurements revealed that WhatsApp servers are associated to the domain names `whatsapp.net` (for supporting the service) and `whatsapp.com` (for the company website). As indicated in table 1, different third level domain names are used to handle different types of traffic (control, text messages, and multimedia messages). When the client application starts, it contacts a *messaging* or *chat server* {e|c|d}X.whatsapp.net listening on port 5222, where X is an integer changing for load balancing. This port is assigned by IANA to clear-text XMPP sessions. Nevertheless, the connection is SSL-encrypted. This connection is used for text messages as well as control channel, and is kept up while the application is active or in background. If the connection is dropped, a new one with the same or another messaging server is immediately re-established. In case the application client is not running, the message notification is delivered through the OS push APIs.

The application also offers the capability of multimedia contents transfer, including photos, audio and video. Transfers are managed by HTTPS *multimedia (mm) servers* listening on port 443. Those servers are associated to different domain names depending on their specific task: `mmsXYZ.whatsapp.net` and `mmiXYZ.whatsapp.net` are both used for audio and photo transfers, while `mmvXYZ.whatsapp.net` are exclusively reserved for videos. For each object, a ded-

icated TLS-encrypted connection towards a *mm server* is established. Uploads are started immediately, while downloads of large objects need to be manually triggered by the receiving user to avoid undesired traffic. These servers do not perform any transcoding. As we shall see in §4, the two server classes have very different network footprints. While connections to chat servers are characterized by low data-rate and long duration (specially due to the control messages), media transfers are carried by short and heavy flows.

3 Hosting Infrastructure

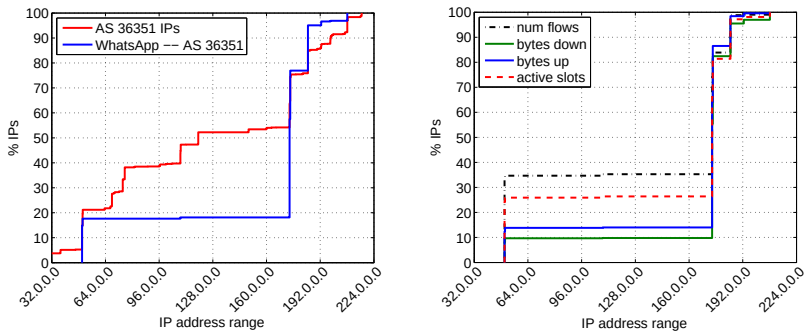
The first part of the study focuses on discovering where the servers are located. For doing so, we rely on the analysis of a complete week of WhatsApp traffic traces, consisting of more than 150 million flows collected at the core of a European national-wide cellular network, from 18.02 till 25.02. Flows are captured at the well-known Gn interface [22], using the METAWIN cellular network monitoring system [12]. To preserve user privacy, any user related data are anonymized, while packets' payload is removed on the fly. Using the MaxMind GeoIP databases [13], the ASes serving the corresponding flows are included in the dataset. Traffic flows are continuously imported and analyzed through DBStream [11], a data stream warehouse tailored for large-scale traffic monitoring applications. In the following analysis, volume and flow counts are normalized to preserve business privacy, and time-series are constructed with 10-min time slots resolution.

3.1 Methodology

WhatsApp communications are encrypted, thus we firstly devised a classification approach to identify WhatsApp flows. The approach is based on the HTTPTag classification system [10] running on DBStream. HTTPTag classification consists in applying pattern matching techniques to the `hostname` field of the HTTP requests. Given the usage of encryption, and the need to also classify non-HTTP traffic, the approach was extended to consider the analysis of DNS requests, similar to [9]. Every time a user issues a DNS request for the Fully Qualified Domain Name (FQDN) `*.whatsapp.net`, HTTPTag creates an entry mapping this user to the server IPs provided in the DNS reply. Each entry is time stamped and contains the TTL replied by the DNS server. Using these mappings, all the subsequent flows between this user and the identified servers are assumed to be WhatsApp flows. The approach also allows for a finer-grained classification, using more specific patterns, i.e. `(c|d|e)*.whatsapp.net` for chat flows and `mm*.whatsapp.net` for media flows. To avoid miss-classifications due to out-of-date mappings, every entry expires after a TTL-based time-out. To increase the robustness of the approach, the list of IPs is augmented by adding the list of server IPs signing the TLS/SSL certificates with the string `*.whatsapp.net`. Indeed, our hybrid measurements revealed that WhatsApp uses this string to sign all its communications. Finally, we use reverse DNS queries to verify that the list of filtered IPs actually corresponds to a WhatsApp domain.

Service/AS	**# IPs**	**#/24**	**#/16**	**#/8**
WhatsApp	386	51	30	24
SoftLayer (AS 36351)	1,364,480	5330	106	42

Table 2. Number of server IPs and prefixes used by WhatsApp.



(a) WhatsApp/SoftLayer IP ranges.

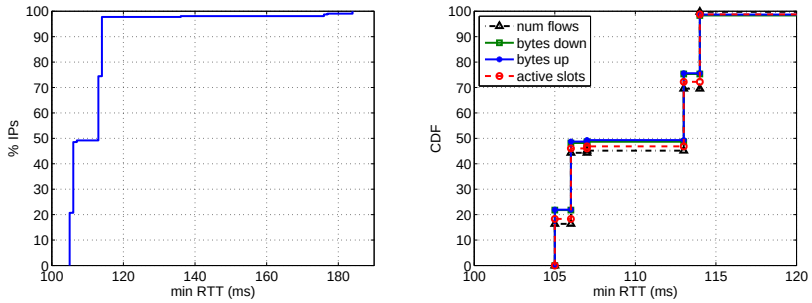
(b) Weighted IP ranges.

Fig. 1. Ranges of IPs hosting WhatsApp.

3.2 Measurements Analysis

The complete one-week server IP mappings resulted in a total of 386 IPs identified as hosting the service, belonging to a single AS called SoftLayer (AS number 36351) [19]. To avoid biased conclusions about the set of identified IPs from a single vantage point, we performed an active measurements campaign using the RIPE Atlas measurement network [14], where we analyzed which IPs were obtained when resolving the same FQDNs from 600 different boxes around the globe during multiple days. These active measurements confirmed that the same set of IPs is always replied, regardless of the geographical location of the requester. SoftLayer is a US-based cloud infrastructure provider consisting of 13 data centers and 17 Points of Presence (PoPs) distributed worldwide. Using MaxMind geoloacalization capabilities, we observed that despite its geographical distribution, WhatsApp traffic is handled mainly by the data centers in Dallas and Houston. Given that the city-location accuracy of public GeoIP databases such as MaxMind is questionable [21], we confirmed through traceroutes and active RTT measurements that the servers are indeed located in the US.

Tab. 2 reports the different number of prefixes covered by the identified IPs in SoftLayer. Note that we consider different netmasks (e.g., /24, /16, /8) for simple counting and aggregation purposes, i.e., we do not claim that the prefixes are fully covered/own by the ASes. The range of server IPs is highly distributed, covering 51 different /24 prefixes and 24 /8 ones. The table additionally shows the total number of SoftLayer IPv4 IPs. Fig. 1(a) shows the intersection of both IP address ranges. As depicted in Fig. 1(b), when weighting the IP ranges by volume, the majority of the traffic corresponds to IPs falling in 3 /16 ranges.



(a) min RTT per WhatsApp IP. (b) Weighted min RTTs.

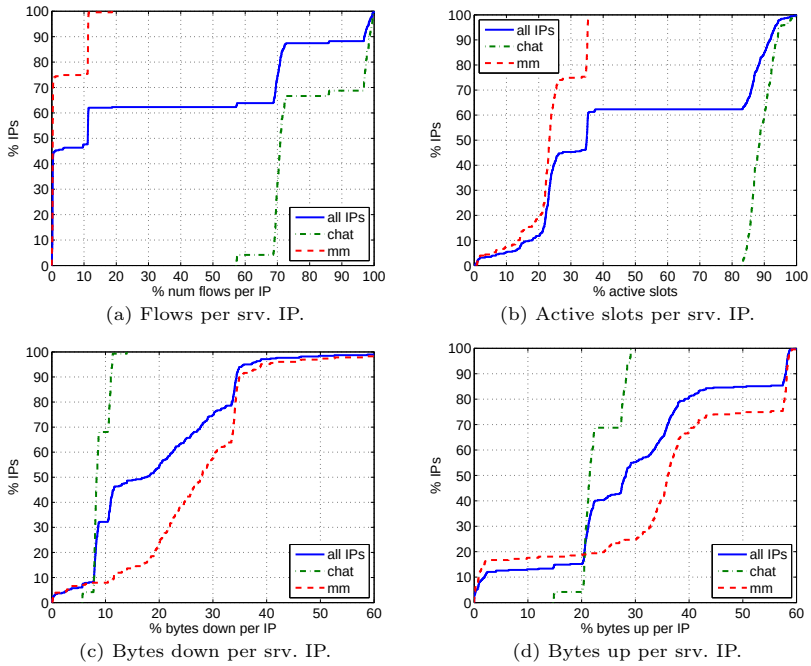
Fig. 2. min RTT to WhatsApp server IPs.

However, in terms of flows and activity (measured as 10-min active slots), the range 50.22.225.0/24 captures a main share.

To complement the picture of the servers location, we investigate the distance to the vantage point in terms of RTTs, analyzing the minimum RTT values. RTTs are obtained from active ping measurements, performed during the span of the dataset from a single location in Europe. Every unique IP hosting WhatsApp is pinged with trains of 100 ICMP echo request packets every 10 minutes. Fig. 2 plots the distribution of the minimum RTT to (a) all the server IPs hosting WhatsApp, and (b) the same min RTT values, weighted by the previously considered 4 features (i.e., flows, active slots, and traffic volumes) to get a better understanding of the traffic sources. The distribution presents some clear steps indicating the existence of different data centers or hosting locations. The min RTT is always bigger than 100ms, confirming that WhatsApp servers are located outside Europe, where our vantage point is located. Fig. 2(b) shows that the service is evenly handled between two different yet potentially very close locations at about 106 ms and 114 ms, which is compatible with our previous findings.

To further understand how the hosting infrastructure of WhatsApp is structured, Fig. 3 depicts the distribution of server IPs over the same previous 4 features. The figures additionally depict chat and multimedia servers to discriminate their roles. Regarding (a) number of flows and (b) active time slots, we clearly observe how chat servers handle the biggest share of the flows, with a highly active set of server IPs. On the contrary, multimedia servers are much less active and handle a limited share of flows. In terms of volume, the picture is completely the opposite when considering traffic volumes in (c) downlink and (d) uplink directions.

Fig. 4 shows the dynamics of WhatsApp for 3 consecutive days, including the number of active server IPs, the fraction of flows and traffic volume shares, discriminating by chat and mm traffic. The mm category is further split into photos/audio (mmi and mms) and video (mmv). The time-series present a clear night/day pattern with two daily peaks at noon and 8pm. Fig. 4(a) indicates that more than 350 IPs serve WhatsApp flows during peak hours. Note that no less



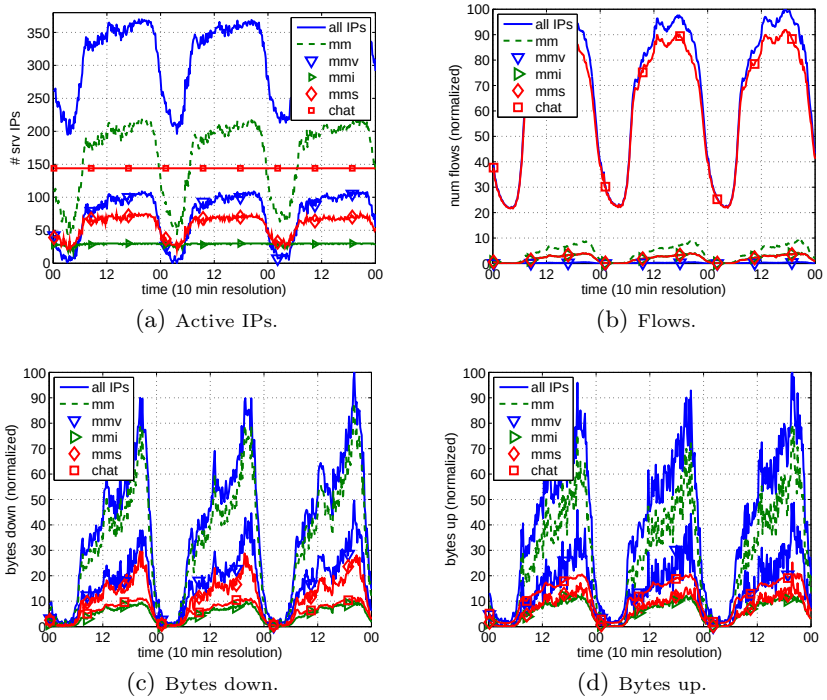
(a) Flows per srv. IP. (b) Active slots per srv. IP.

(c) Bytes down per srv. IP. (d) Bytes up per srv. IP.

Fig. 3. WhatsApp server IPs in terms of volume, flows, and activity shares.

than 200 IPs are active even in the lowest load hours. When analyzing the active IPs per traffic type, we see how chat servers are constantly active, as they keep the state of active devices to achieve an efficient and fast push of the messages to the device. Fig. 4(b) shows the flow count shares, revealing how chat flows are clearly dominating. Once again we stop in the mmi and mms servers, which seem to always handle the same share of flows, suggesting that both space names are used as a mean to balance the load in terms of photos and audio messages. Finally, Figs. 4(c) and 4(d) reveal that even if the mm volume is higher than the chat volume, the latter is comparable to the photos and audio messaging volume, specially in the uplink. Tab. 3 summarizes these shares of flows and traffic volume. The reader should note that our dataset does not include flows transmitted over WiFi, thus some of these results might be biased due to users potentially using WiFi for large file transfers. We are currently analyzing this potential bias as part of our ongoing work, and our first results confirm that our observations are still valid.

As a conclusion, our measurements confirmed that WhatsApp is a centralized and fully US-based service. This is likely to change in the near future after Facebook's WhatsApp acquisition. As for now, all messages among users outside the US are routed through the core network. Being Brazil, India, Mexico and Russia the fastest growing countries in terms of users [16], such a centralized hosting infrastructure is likely to become a problematic bottleneck. Indeed, as



(a) Active IPs. (b) Flows.

(c) Bytes down. (d) Bytes up.

Fig. 4. WhatsApp dynamics. More than 350 IPs serve WhatsApp during peak hours.

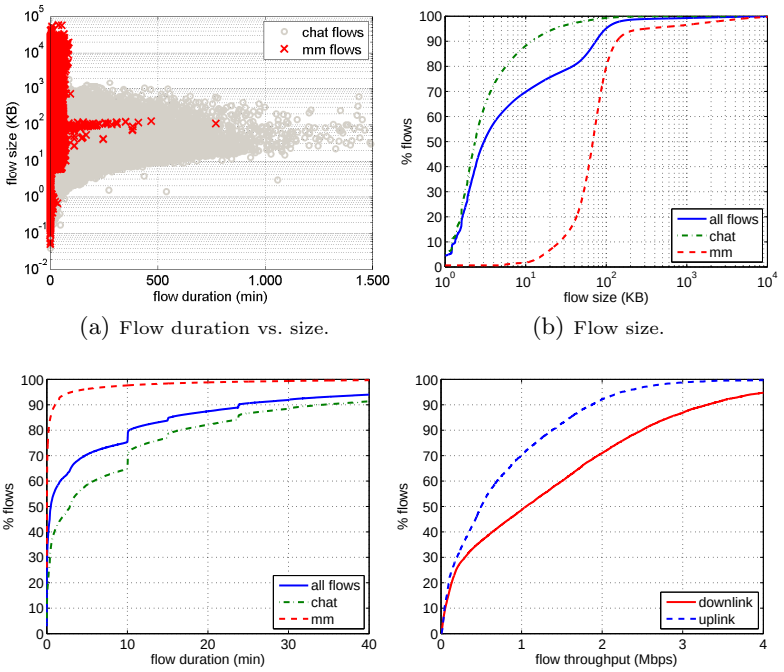
features	chat	mm	mmv	mmi	mms
# $\text{bytes}_{\text{down}}$	16.6%	83.0%	38.8%	12.8%	29.8%
# bytes_{up}	29.5%	70.2%	35.2%	15.0%	17.9%
# flows	93.4%	6.2%	0.3%	2.9%	2.9%
$\frac{\#\ \text{bytes}_{\text{down}}}{\#\ \text{bytes}_{\text{down+up}}}$	60.6%	76.3%	75.1%	70.0%	81.9%

Table 3. Volume and flows per traffic category.

we show in §5, the high latencies to US servers are a potential cause of bad QoE for users downloading multimedia files, due to an increased download time and a reduced TCP throughput.

4 Traffic Analysis

We study now the characteristics of the WhatsApp traffic in terms of size and duration. Additionally, we evaluate the performance of the service, computing the transfer throughputs as the Key Performance Indicator (KPI). Flow durations are measured with a coarse-grained resolution of one second (this is a limitation of the monitoring system, given the large amount of processed traffic), considering the time-stamps of the first and the last packet of a standard 5-tuple



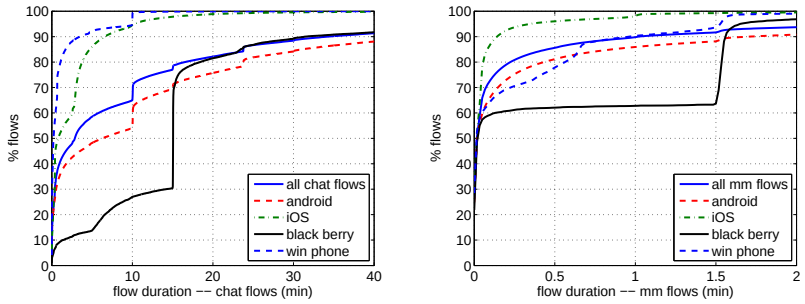
(a) Flow duration vs. size. (b) Flow size.

(c) Flow duration. (d) MM flow throughput (flow size > 1MB).

Fig. 5. WhatsApp flow characteristics and performance.

measured flow (note that flows are unidirectional) and adaptive flow time-outs, see [12] for additional details. Flow throughput is estimated as the ratio between the total transferred bytes and the flow duration. Note that given the one second resolution, throughput values are somehow an underestimate of the real throughput. Still, the results obtained in the paper about flow duration allows us to claim that the absolute errors are marginal.

Fig. 5(a) shows a scatter plot reporting the flow duration vs. the flow size, discriminating by chat and mm flows. Whereas mm messages are sent over dedicated connections, resulting in short-lived flows, text messages are sent over the same connection used for control data, resulting in much longer flows. For example, some chat flows are active for as much as 62 hours. The protrusion at around 100KB is due to the fact that the client perform compression of images and most of media flows are close to that size. Fig. 5(b) indicates that more than 50% of the mm flows are bigger than 70 KB, with an average flow size of 225 KB. More than 90% of the chat flows are smaller than 10 KB, with an average size of 6.7 KB. In terms of duration, Fig. 5(c) shows that more than 90% of the mm flows last less than 1 min (mean duration of 1.8 min), whereas chat flows last on average as much as 17 minutes. The flow duration CDF additionally reveals some clear steps at exactly 10, 15 and 24 minutes, suggesting the usage of an application time-out to terminate long idle connections. This behavior is actually dictated by the operating system of the device. To better understand it,



(a) chat flows. (b) mm flows.

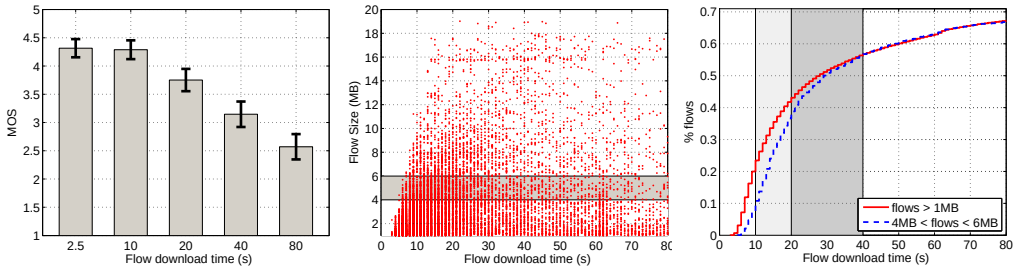
Fig. 6. Flow duration per different OS.

we performed a device OS classification based on manual labeling of each device based on its IMEI, covering more than 90% of the observed flows. Note that the device IMEI is not contained in the WhatsApp messages, but comes from other monitoring sources in METAWIN. Fig. 6(a) splits the analysis of the chat flow duration per device OS. The figure clearly shows that the aforementioned time-out is mainly OS-dependent, as different platforms show different values. Three different time-outs are visible for Android devices at 10, 15 and 24 mins; iOS uses a very short time-out of 3 mins, BlackBerry devices have 15 mins. long time-outs, whereas Windows Mobile phones favor 10 mins. time-outs. On the contrary, in the case of mm flows in Fig. 6(b), all the different OS show a similar behavior, with the exception of BlackBerry and Windows Phone, using a 90 secs. time-out. These observations might have a major impact on the performance of the Radio Access Network, due to different OS synchronization times and uneven resources reservation requests. Indeed, it has been recently shown that applications that provide continuous online presence such as WhatsApp can generate a significant burden on the signaling plane in cellular networks [3].

Considering flow throughput, Fig. 5(d) depicts the uplink and downlink throughputs for flows bigger than 1 MB. This filtering is performed as a means to improve the throughput estimations. A-priori, one might expect that the long RTTs involved in the communications to the US servers might heavily impact the achieved performance. This is confirmed for about 30% of the transmitted flows, which achieve a throughput smaller than 250 kbps. However, higher throughputs are obtained for the largest shares of flows, achieving an average per flow downlink/uplink throughput of 1.5 Mbps/800 kbps. Still, as we show next, a big share of the file downloads can actually result in a very poor quality of experience for the users.

5 Quality of Experience in WhatsApp

In the previous section we considered the transfer throughput as the main KPI reflecting service performance. However, in order to better understand the impacts of transfer throughputs on the experience of the users, we performed a

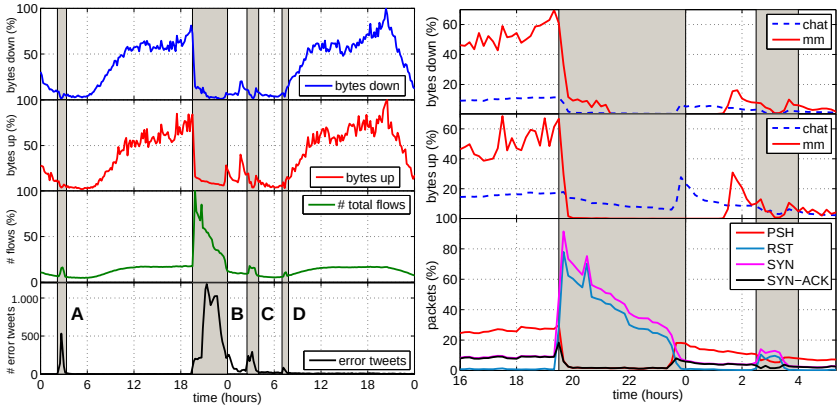


(a) WhatsApp overall quality. (b) Flow size vs. download time. (c) Flow download time distribution.

Fig. 7. QoE in WhatsApp, considering flows bigger than 1MB.

QoE-based study of WhatsApp, relying on subjective QoE tests performed in the lab, following well defined standards for realizing the tests and analyzing the results [23, 24]. In a nutshell, 50 participants (45%/55% male/female, 23 average age, 60%/40% students/employees) provided their feedback in terms of Mean Opinion Scores (MOSs), reflecting their experienced quality while using WhatsApp for transferring video and music files. The study consisted of users receiving a multimedia file of 5MB to download on their smartphones as a WhatsApp shared file. Different network conditions were emulated by connecting the phones to a network emulator, introducing different download throughput profiles via traffic shaping. At the end of each download, the user rates the overall quality in a 1-to-5 MOS scale, where 5 means excellent experience and 1 means a very bad one. Note that the file size of 5MB has a clear motivation behind: mp3 music files and short videos have a similar size. While it is clear that the 5MB flow size reflects only a fraction of the total flows (as depicted in Fig. 5(b)), the performed study permits to have some rough ideas of what the users perceive of the service in terms of quality in this case. A deeper WhatsApp QoE-based study is part of our current work.

Fig. 7(a) shows the QoE results for different download throughput values, translated into waiting times. Download time is in fact the most relevant feature as perceived by the user when analyzing file transfers [25], as this is directly linked to anxiety and satisfaction. The figure shows that users tolerate transfers of up to 20s long with a good overall experience, whereas transfers lasting more than 80s are considered as very bad quality. A threshold of about 40s permits to approximately discriminate between good and bad experience. Fig. 7(b) plots the Flow Size vs. the Flow Download Time (FDT) for the large-scale dataset, considering only flows bigger than 1MB. If we focus on the range of flows with sizes around 5MB, we see that while the majority of the flows have a FDT below 40s, there are many downloads which highly exceed this threshold. Indeed, Fig. 7(c) shows the distribution of the FDTs, both for all the flows with size between 4MB and 6MB, as well as for all the flows bigger than 1MB. From these CDFs, one can say that almost 40% of the WhatsApp downloads with size between 4MB and 6MB have a FDT lower than 20s, resulting in good user experience. About 60% still result in an acceptable quality, and about 35% are potentially badly



(a) Aggregated traffic and OSN feedback. (b) A zoom-in of the TCP flag counts.

Fig. 8. The WhatsApp worldwide outage.

or very badly perceived. Finally, if we now assume that users are generally non experts and that file sizes are not taken into account into their quality expectations when downloading a video or a song through WhatsApp, we could say that similar results are observed for the complete dataset of downloaded flows bigger than 1MB. Of course this last observation is rather controversial, but still presents some notions on the experience of the end users. As a main conclusion, we see that the architectural design of WhatsApp, with servers centralized in the US, might actually have an impact on the experience of the users.

6 The WhatsApp Blackout

The last part of the study focuses on the analysis of the major WhatsApp worldwide outage reported since its beginning as observed in our traces. The outage occurred in February the 22nd of 2014, and had a strong attention in the medias worldwide. The event is not only clearly visible in our passive traces, but can also be correlated with the near real-time user reactions on social networks. Through the online downdetector application [18] we accessed the counts of tweeter feeds containing the keyword "whatsapp", coupled with keywords reflecting service impairments such as "outage", "is down", etc.. We refer to these tweets as *error tweets*.

Fig. 8(a) depicts the time series of the share of bytes exchanged with the servers, the share of flows, as well as the number of error tweets during two consecutive days encompassing the outage. The traffic drastically dropped on the 22nd at around 19:00 CEST (event `B`), and slowly started recovering after midnight, with some transient anomalous behaviors in the following hours (events `C` and `D`). Traffic volumes in both directions did not drop completely to zero but some non-negligible fraction of the traffic was still being exchanged, suggesting an overloading problem of the hosting infrastructure. In terms of number of flows, there is a clear ramp-up on the flow counts. Very interestingly, there is a

clear correlation between the events B, C and D and the number of WhatsApp-related error tweets. The users reacted on the social network immediately after the beginning of the outage, with the viral effect reaching its highest point after one hour. There is an additional outage event marked as A, which is clearly observable in the error tweet counts and has exactly the same signature of events B, C and D, i.e., a drop in the traffic volume and an increase in the flows count. As a take away of this social data analysis, one can use such information as ground truth for near real-time detection of QoE-relevant anomalies in popular services such as WhatsApp.

To better drill-down the anomaly, Fig. 8(b) depicts a 12-hour zoom-in of the traffic volume trends, split by chat and mm traffic, along with the counters of TCP flags. The bytes down counters show that the residual downlink traffic exchanged during the first part of the anomaly is due to previously queued mm transfers. In fact, while chat servers stopped working, media servers are still up and running at the beginning of the outage. We recall that connections to chat servers are also used for application control, hence they provide links to media contents. If such links have been delivered before the chat outage, the users might still be able to retrieve media objects. The chat traffic in the uplink direction does not drop to zero but slowly fades out, which actually corresponds to control flows trying to re-establish the lost connections. In particular, the TCP flags counters reveal an steeped increase of SYN packets, indicating that devices were repeatedly trying to reconnect after the servers abruptly flashed the connections (RST flags). This suggests that the servers were still reachable, thus the failure occurred at the application layer. The SYN and RST counters decrease gradually, revealing a back-off mechanism of the client application. These connection attempts explain the high increase in the flow counts during events A-D, as well as the persistence of uplink traffic to chat servers. This behaviour affected the whole WhatsApp addressing space.

7 Concluding Remarks

WhatsApp is the fastest-growing company in history in terms of users, and this paper presented the first large-scale characterization of the service from passive measurements collected at a national-wide cellular network. Our study fully dissected the well structured internal naming scheme used by WhatsApp to handle the different types of connections, which shall enable an easy way to monitor its traffic in the network. We discovered that WhatsApp is a centralized service, fully hosted in the US. We showed that such a centralized hosting infrastructure might negatively impact the experience of the end users. Even more, we believe that having a poorly geo-distributed network of servers might be highly harmful in terms of failures for such Internet-scale services, as revealed by the characterized worldwide outage. Finally, we showed that WhatsApp uses two different approaches for handling the messages exchanged among its users, keeping persistent connections to handle text messages and short-lived flows to send multimedia contents.

The datasets collected and analyzed in this paper correspond to a very interesting point in time in the history of WhatsApp, in which Facebook acquired the service. Given the highly distributed nature of the Facebook network [5], we expect a significant change in the WhatsApp network architecture in the next couple of years, and we are currently collecting a very large-scale WhatsApp dataset to further investigate such changes in an upcoming study.

Acknowledgements

This work has been performed in the framework of the EU-IP project mPlane, funded by the European Commission under the grant 318627

References

1. P. Fiadino et al., "Vivisecting WhatsApp through Large-Scale Measurements in Mobile Networks", extended abstract in *ACM SIGCOMM*, 2014.
2. E. Vergara, S. Andersson, S. Nadjm-Tehrani, "When Mice Consume Like Elephants: Instant Messaging Applications", in *ACM e-Energy*, 2014.
3. A. Aucinas, N. Vallina-Rodriguez, Y. Grunenberger, V. Erramilli, K. Papagiannaki, J. Crowcroft, D. Wetherall, "Staying Online While Mobile: The Hidden Costs", in *ACM CoNEXT*, 2013.
4. K. Chen, C. Huang, P. Huang, C. Lei, "Quantifying Skype User Satisfaction", in *ACM SIGCOMM*, 2006.
5. P. Fiadino, A. D'Alconzo, P. Casas, "Characterizing Web Services Provisioning via CDNs: The Case of Facebook", in *TRAC*, 2014.
6. A. Finamore, M. Mellia, M. Munafo, R. Torres, S. G. Rao, "YouTube Everywhere: Impact of Device and Infrastructure Synergies on User Experience", in *ACM IMC*, 2011.
7. I. Drago, M. Mellia, M. Munafo, A. Sperotto, R. Sadre, A. Pras, "Inside Dropbox: Understanding Personal Cloud Storage Services", in *ACM IMC*, 2012.
8. G. Magno, G. Comarela, D. Saez-Trumper, M. Cha, V. Almeida, "New Kid on the Block: Exploring the Google+ Social Graph", in *ACM IMC*, 2012.
9. I. Bermudez, M. Mellia, M. Munafo, R. Keralapura, A. Nucci, "DNS to the rescue: Discerning Content and Services in a Tangled Web", in *ACM IMC*, 2012.
10. P. Fiadino, A. Bär, P. Casas, "HTTPTag: A Flexible On-line HTTP Classification System for Operational 3G Networks", in *IEEE INFOCOM*, 2013.
11. A. Bär, P. Casas, L. Golab, A. Finamore, "DBStream: an Online Aggregation, Filtering and Processing System for Network Traffic Monitoring", in *TRAC*, 2014.
12. F. Ricciato, "Traffic Monitoring and Analysis for the Optimization of a 3G network", in *IEEE Wireless Communications*, vol. 13(6), 2006.
13. MaxMIND GeoIP Databases, in `http://www.maxmind.com`. [Accessed on 20-08-14].
14. The RIPE Atlas measurement network, in `https://atlas.ripe.net/`. [Accessed on 02-09-14].
15. H. Blodget, "Everyone Who Thinks Facebook Is Stupid To Buy WhatsApp For $19 Billion Should Think Again", in `http://www.businessinsider.com/why-facebook-buying-whatsapp-2014-2`. [Accessed on 15-08-14].
16. WhatsApp Blog, in `http://blog.whatsapp.com/`. [Accessed on 07-09-14].
17. WhatsApp Status in Twitter, in `https://twitter.com/wa_status`. [Accessed on 13-08-14].
18. Downdetector.com, in `http://downdetector.com/`. [Accessed on 07-10-14].
19. SoftLayer: Cloud Servers, in `http://www.softlayer.com`. [Accessed on 01-10-14].
20. P. Saint-Andre, "Extensible Messaging and Presence Protocol (XMPP): Instant Messaging and Presence", RFC-6121, March 2011.
21. I. Poese, S. Uhlig, M. Kaafar, B. Donnet, B. Gueye, "IP Geolocation Databases: Unreliable?", in *ACM SIGCOMM Computer Communication Review*, pp. 53-56, 2011.
22. J. Bannister, P. Mather, S. Coope, "Convergence Technologies for 3G Networks: IP, UMTS, EGPRS and ATM". Wiley, 2004.
23. International Telecommunication Union, "Methods for Subjective Determination of Transmission Quality", *ITU-T Rec. P.800*, 1996.
24. International Telecommunication Union, "Estimating End-to-End Performance in IP Networks for Data Applications", *ITU-T Rec. G.1030*, 2005.
25. P. Casas et al., "A First Look at Quality of Experience in Personal Cloud Storage Services", in *ICC Workshops*, 2013.