International Journals of Combined Research & Development
eISSN:2321-225X,pISSN:2321-2241,Vol:5;Issue:6 ,June- 2016

An Efficient Data Upload Mechanism Using Wi-Fi Offloading

Ankit Shaw [1], Dhanya Vishwanath Hegde [2], Chaitra C Mouli [3], Chamundeshwari S [4], Mrs. Poornima G J [5]

[1,2,3,4] BE, Final year,Dept. Of CSE, Sapthagiri College of Engineering, Bangalore, India
[5] Asst. Professor , Final year,Dept. Of CSE, Sapthagiri College of Engineering,Bangalore, India

Abstract : The inventions and developments of smart phones and other portable devices, the amount of digital data being generated is increasing at a faster rate. This results in an overhead for the server to maintain and handle the requests for these large data. Cellular internet connectivity allows mobile users to upload and download their data onto the server but at the cost of draining their battery and overloading the service providers. Also an end-to-end connection has to be established from the mobile device to the server for data transfer. This causes a delay in the upload/download rate of the user data sometimes also creating a bottleneck at various access points of the network leading to longer waiting time.The proposed work uses the storage capabilities of various devices located on the Wi-Fi access points to offload the upload tasks. The system proposed plans to decrease the delay between data transfer time in the end-to-end connection using Wi-Fi offloading techniques.

Keywords: Wi-Fi offloading; Web technologies; Delay-tolerant networking; store and forward; Hoop System

1. INTRODUCTION

We ask that authors follow some simple guidelines.In essence, we ask you to make your paper look exactly likethisdocument.Theeasiestwaytodothisissimplytodownload the template, and replace the content with your ownmaterial.

When mobile data was introduced in the early 2000s, operators unsuccessfully looked for applications that would instigate subscribers to use slow networks on their voice-centric phones. It was the e-mail application on the first data-centric smartphones that started to reverse the situation. The appearance of iPhone in 2007 finally changed everything and exposed users to rich data services, such as mobile video.

In today’s world mobile data traffic is growing at an unprecedented rate.The main drive behind this explosive growth in traffic demand is rapid increase in the number of smart phones and tablets that offer ubiquitous Internet access and proliferation of traffic intensive applications for such smart devices.For example, they take photos and videos with their smartphones and produce or edit possibly large documents on their tablets and laptops. The data is then uploaded to online services, typically through web applications, native apps or system services. They do so for various purposes ranging from social sharing (e.g., sharing photos on Facebook , posting images on Instagram or videos on YouTube) to increasing availability and backup of data (e.g., uploading all sorts of documents to a cloud storage service such as Dropbox or Google Drives etc).

There are several solutions to this explosive traffic growth problem. The first is to scale the network capacity by building out more cell towers and base stations of smaller cell sizes (e.g., picocell, femtocell) or upgrading the network to the next-generation networks such as Long Term Evolution (LTE) and WiMAX. However, this is not a winning strategy, especially under a flat price structure where revenue is independent of data usage.[1]

The second is to adopt a usage-based price plan that limits heavy data usages. While price restructuring is rather inevitable, pure usage-based plans are likely to backfire by singling out a particular sector of user groups, e.g., smartphone users, which have the highest potential for future revenue growth.

WiFi offloading seems the most viable solution at the moment.Mobile data offloading, often known as WiFi offloading, is the use of complementary network technologies for delivering data originally targeted for cellular networks. Offloading reduces the amount of data being carried on the cellular bands, freeing bandwidth for other users. It is also used in situations where local cell reception may be poor, allowing the user to connect via wired services with better connectivity.

Building more WiFi hotspots is significantly cheaper than network upgrades and build-out. Many users are also installing their own WiFi access points(APs) at homes and work. If a majority of traffic is redirected through WiFi networks, carriers can accommodate the traffic growth only at a far lower cost.

There are two types of offloading :

- On-the-spot
- Delayed.

On-the-spot offloading is to use spontaneous connectivity to WiFi and transfer data on the spot. Most of the current smartphones support on-the-spot offloading by default. In delayed offloading, each data transfer is associated with a deadline, and the data transfer is resumed whenever getting in the coverage of WiFi until the transfer is complete. If the transfer does not finish within its deadline, cellular networks finally complete the transfer.This technology benefits in relieving the infrastructure network load, shifting data to a complementary wireless technology that leads to a number of other improvements, like increasing the throughput, better energy efficiency, extension of network coverage.

Offloading is often described as a win-winstrategy since they benefit both users and cellular operators. This, however, comes at the cost of constrained mobility and/or significant delays for users.[2] Indeed, the users are required to stay in the vicinity of the access point while the data is being uploaded. In the case of personal or corporate access points, the data is uploaded only when the user reaches the corresponding location (i.e., home and work place respectively).

One of the most important constraint that is delay can be reduced by implementing HOOP , which offloads upload tasks onto devices located on the wifi access point's LAN, typically residential gateways or NAS units to decrease the waiting time.Here there will be an overview of creating html upload form, uploading the data to the server, creating a local network and sharing files through that network.

2. RELATED WORK

The problem of mobile data upload has received a great deal of attention from the research community over thelast few years. More specifically, Balasubramanian et al.first proposed [16] to augment the 3G capacity in mobilescenarios by exploiting Wi-Fi access points. They implementa software solution for delaying data exchanges andfast-switching between 3G and Wi-Fi, and they assess thepotential of their approach. In [3], Lee et al. perform alarge-scale experimental performance evaluation of dataoffloading over Wi-Fi that demonstrates the benefits of thisapproach, both in terms of the amount of data offloadedfrom 3G and of battery power. In [6], Trestian et al. Studythe data generation and upload patterns of mobile usersand advocate the use of cells with disproportionally upgradedbandwidth, called Drop Zones, for offloading thecontent generated by mobile users while on the go. Inaddition, they tackle the problem of the optimal placementand of the dimensioning of the Drop Zones. In [13], Goet al. propose a practical implementation of mobile Wi-Fi offloading where they use a transmission protocol that enables mobile devices to maintain an end-to-end connectionwith a server despite network disruption. As the proposedsolution operates at the transport layer, it is transparent tothe applications, and thus it is quite generic. However, itrequires modifications of the network interface at the clientside. Beyond academic contributions, some companies (e.g.,GoNet2) have successfully designed and deployed Wi-Finetworks for 3G offloading. Finally, some pieces of work,such as [17], complement data offloading with device-todevicecommunications. In [18], Han et al. survey existingoffloading techniques. In all these works, it is assumed thatthe data is offloaded directly over Wi-Fi, at the speed of theaccess point's connection to the Internet, which constitutesa bottleneck. Although HOOP relies on the same approach,i.e., offloading traffic at Wi-Fi access points, it goes beyondby exploiting the storage capacity at the access points tofully take advantage of the Wi-Fi connectivity for delaytolerantuploads. In [19], the authors study the trade-offbetween data downloading delays and user satisfaction inthe case of 3G offloading; they show that, by predicting theusers' offloading potential and by using appropriate incentives,data downloads can be efficiently delayed withoutsacrificing the users' satisfaction. Finally, in [20], Kim et al.propose an analytical framework to study the performanceof mobile data offloading. One of their main findings isthat existing Wi-Fi infrastructures deployed in metropolitanareas are sufficient to offload, within reasonable delays,most of the mobile user traffic.Several works, e.g., [11], [12], advocate the use of thestorage capacity of gateways–and other always-on deviceswith storage capacity–to offload data transfer from userdevices. Technical solutions have been proposed and implementedon gateways, set-top boxes and networked areastorage units. For instance, many such devices offer HTTPdownload services and run BitTorrent clients (e.g., SynologyNAS). Closer to our work, the Fonera [21] enables users toasynchronously upload files to a number of online services(including YouTube, flickr, and Facebook) by simply copyingthem over, e.g., ftp, to specific folders. Unlike HOOP,such solutions have major drawbacks that prevent wideadoption in the public domain: The device is trusted withthe users' credentials for these online services; the device isgiven the users' data, in clear, which it can alter; the solutionis dependent on the online service (as it relies on theirproprietary APIs) and it requires explicit user interactions,as opposed to HOOP that is generic and seamless.

3. SYSTEM MODEL AND ARCHITECTURE

3.1 Architecture

The whole system is composed of four major components:
1. A local area network
2. A mobile device, controlled by the user
2. An online web service
3. Storage device like NASas described in Figure 3.1.

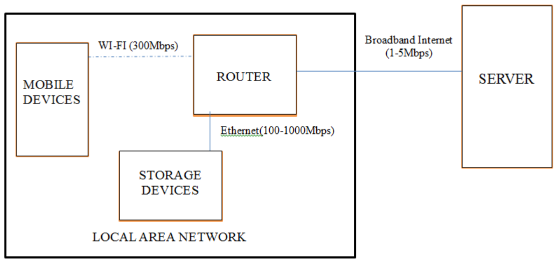
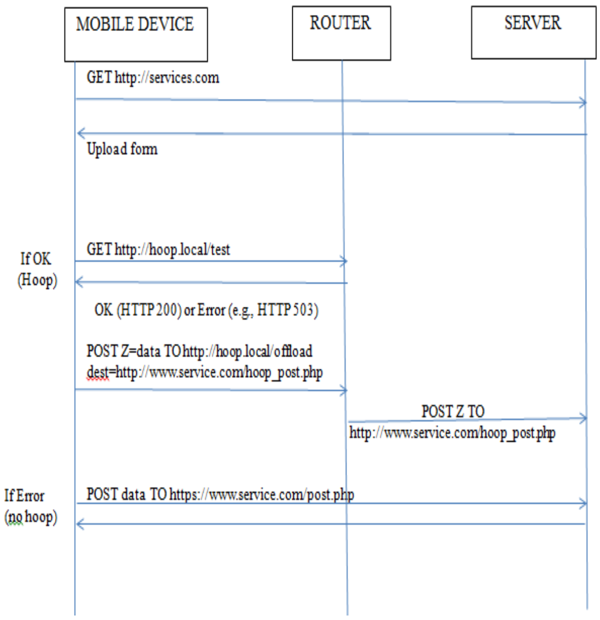


Fig. 1 System Architecture

The local area network is composed of a router that may be a gateway which connects the different devices to the Internet, a device with computational and storage capabilities to run Hoop (typically a set-top box or the gateway), and an access point that enables users with wireless-equipped devices to connect to the local network. At first user needs to connect to the Internet through the local network with a wireless-equipped mobile device and then makes use of web services through the browser installed in the device or any native apps. Here it has been considered an online web services that enables users to upload data through HTTP POST operations, from an HTML form (potentially with AJAX), a Flash uploader, or a native app. Here the main focus is on HTML forms.

4. SYSTEM DESIGN AND IMPLEMENTATION

4.1 Sequence Diagram



User connects to the web service and requests the upload page,through http, from the browser installed on the mobile device. The web service returns HTML webpage including a form to select the data to be uploaded, some extra information(a caption),an authentication token, and the target page to which the data will be posted. The user selects the file or files to upload and submit the form by clicking on the corresponding button.

A system for offloading upload tasks onto devices is located on the same LAN as the user's mobile device in a store-and-forward fashion. HOOP involves three different entities, as described in the system model: a software component on the device running HOOP (say a gateway), the application running on the user's mobile device, and the web service. We describe the functioning of HOOP by listing and explaining the different operations performed by each of the three aforementioned entities. HOOP operates as follows: The mobile device (be it a script executed by the browser or a native app) searches for a device running HOOP on the local network and, if any such device is found, it processes (i.e., re-formats and encrypts) the data to be sent and directs the upload to this device (instead of to the web service). The device running HOOP stores the data received from the mobile device and asynchronously uploads it to the web service that handles the data as for a regular upload. The system ensures the data transfer to the server after temporarily storing the user data to be uploaded and then forwarded on the server without user interference.If HOOP is not running on the router then the file is directly uploaded to server and the client user is acknowledged.

4.2 Algorithm

Step1: User connects to the web service and request for the upload page from the browser installed mobile device.

Step 2: Web server will return a html page including a form that contains at least one form element to select the data to be uploaded and some extra information like captions.

Step 3: The user selects the file to upload and submits the form using submit button.

Step 4: The mobile device searches for a device running HOOP on the local network by sending an HTTP request to http://hoop.local/test.

Step 5: If the request returns successfully (i.e., the host hoop.local is resolved and found, and the request returns the HTTP success code 200)

Step 6: If the request returns successfully then offload the upload data to the device running HOOP at http://hoop.local/offload.

Step 7: When the upload terminates, the user is redirected to a dedicated web page by changing the location header.

Step 8: Else the gateway returns the HTTP service unavailable 503 code , and the file is directly uploaded to sever.

4.3 Implementation

A local network is created and various devices are connected in the network. User needs to give valid username and password to get access to the registered account. The entered username and password is verified with the database and validated for a registered user. If it is a legitimate user, the user gets access to the registered account. If a user has no account, the user can sign up by creating an account and logging in with the same to the registered account.

```
session_start();
if($_SERVER["REQUEST_METHOD"] == "POST") {
$myusername=mysqli_real_escape_string($db,$_POST['email']);
$mypassword=mysqli_real_escape_string($db,$_POST['password
']);
$sql = "SELECT * FROM persons WHERE email =
'$myusername' and password = '$mypassword'";
$result = mysqli_query($db,$sql);
$row = mysqli_fetch_array($result,MYSQLI_ASSOC);
$count = mysqli_num_rows($result);
```

User selects a file(eg.photos or docs) from his device to be uploaded and submits the choice. The user can choose to upload directly to the server or through an intermediate hoop system.

```
<form method="post"
action="http://hoopsystem:9999UploadServlet"enctype="multipar
t/form-data">
<input type="file"
name="dataFile"id="fileChooser"/><br/><input type="submit"
value="Upload" /></form>
```

Once the data is successfully uploaded on to the intermediate device an acknowledgment is sent to the user. The user then can log out of the web page. The hoop system later uploads the file offline to the server and it receives an acknowledgment from the server. On receiving the file from the user directly, the server sends an acknowledgement to the user and then the user can log out.

4.4 Testing and Evaluation

Table 1. Table Performance of the upload time taken for various file types of different sizes

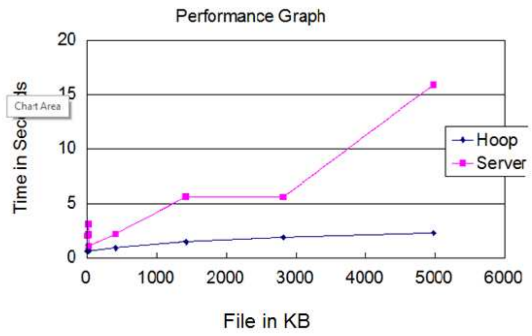


Fig 2 Performance graph of Hoop system versus Direct server upload

5. CONCLUSION AND FUTURE WORK

In this scheme, HOOP, a system for offloadingdata uploads on devices with storage capabilities,e.g., gateways, in a store-and-forward fashion has been presented. The systemenables mobile users to fully exploit the Wi-Fi link byrelaxing the speed constraints due to the link that connectsthe LAN to the Internet. Unlike existing systems, HOOPoperates transparently–from the stand point of the usersand provides aready-to-use, secure and generic solution todata uploads offloading. HOOP can run on devices with verylimited capabilities (e.g., MIPS processor at 400 MHz with32 MB of RAM) and decreases the waiting time of mobileusers. n the future the HOOP system can be extended for downloads.

6. ACKNOWLEDGMENTS

Our sincere thanks to our project guide Mrs. Poornima G J, Assistant Professor of Dept. Of CS&E, Sapthagiri College of Engineering.Our regards to the HOD of the CS&E dept Dr. C M Prashanth and the principal of the college Dr. Aswath Kumar. We are also greatful to all those who have supported us in any possible way.

7. REFERENCES

[1] K. Huguenin, E. Le Merrer, N. Le Scouarnec, and G. Straub, "Hoop: HTTP POST Offloading from User Devices onto Residential Gateways," in ICWS'14: Proc. of the 21st IEEE Int'l Conf. on Web Services, 2014, pp. 654–661.
[2] F. Guidec, D. Benferhat, and P. Quinton, "Biomedical monitoring of non-hospitalized subjects using disruption-tolerant wireless sensors," in MobiHealth'12: Proc. of the 3rd Conf. on Wireless Mobile Communication and Healthcare, vol. 61, 2012, pp. 11–19.
[3] K. Lee, J. Lee, Y. Yi, I. Rhee, and S. Chong, "Mobile Data Offloading: How Much Can WiFi Deliver?" IEEE/ACM Transactions on Networking, vol. 21, no. 2, pp. 536–550, 2013.
[4] N. Ristanovic, J.-Y. Le Boudec, A. Chaintreau, and V. Erramilli, "Energy Efficient Offloading of 3G Networks," in MASS'11: Proc. of the 2011 IEEE 8th Int'l Conf. on Mobile Ad-Hoc and Sensor Systems, 2011, pp. 202–211.
[5] I. IntelliNet Technologies, "Mobile data offload for 3G networks,"

File Type	Size	Time taken in seconds to upload	
		With Hoop	Without Hoop
Image (.png)	4KB	0.624	2.06
Image (.jif)	20KB	1.20	2.20
Word (.doc)	16KB	0.847	3.14
Music (.mp3)	2.75MB	1.90	5.61
Pdf file	1.38MB	1.49	5.63
Torrent file	24KB	0.666	1.12
Zip file	400KB	0.918	2.22
Execution (.exe)	4.85MB	2.30	15.91

White paper, 2009.
[6] I. Trestian, S. Ranjan, A. Kuzmanovic, and A. Nucci, "Taming the mobile data deluge with drop zones," IEEE/ACM Transactions on Networking, vol. 20, no. 4, pp. 1010–1023, 2012.
[7] "AT&T WiFi," http://www.att.com/gen/general?pid=5949, Last visited: Jan. 2015.
[8] "FON," http://www.fon.com, Last visited: Jan. 2015.
[9] M. Dischinger, A. Haeberlen, K. P. Gummadi, , and S. Saroiu., "Characterizing Residential Broadband Networks," in IMC'07: Proc. of the 7th ACM Int'l Conf. on Internet Measurement, 2007, pp. 43–56.
[10] OECD, "OECD Broadband portal," http://www.oecd.org/sti/broadband/oecdbroadbandportal.htm, Last visited: Jan. 2015.
[11] S. Defrance, A.-M. Kermarrec, E. Le Merrer, N. Le Scouarnec, G. Straub, and A. Van Kempen, "Efficient peer-to-peer backup services through buffering at the edge," in P2P'11: Proc. of the 11th IEEE Int'l Conf. on Peer-to-Peer Computing, 2011, pp. 142–151.
[12] V. Valancius, N. Laoutaris, L. Massoulié, C. Diot, and P. Rodriguez, "Greening the Internet with Nano Data Centers," in CoNext'09: Proc. of the 5th ACM Int'l Conf. on Emerging networking experiments and technologies, 2009, pp. 37–48.
[13] Y. Go, Y. Moon, G. Nam, and K. Park, "A disruption-tolerant transmission protocol for practical mobile data offloading," in MobiOpp'12: Proc. of the 3rd ACM International Workshop on Mobile Opportunistic Networks, 2012, pp. 61–68.
[14] "Gallery: open source web based photo album organizer," http://galleryproject.org/, Last visited: Jan. 2015.
[15] "ResourceSpace: an Open Source Digital Asset

Management,"
http://www.resourcespace.org/, Last visited: Jan. 2015.
[16] A. Balasubramanian, R. Mahajan, and A. Venkataramani, "Augmenting
mobile 3G using WiFi," in MobiSys'10: Proc. of the 8th ACM Int'l Conf. on Mobile systems, applications, and services, 2010,
pp. 209–222.
[17] X. Bao, Y. Lin, U. Lee, I. Rimac, and R. Choudhury, "Exploiting
naturally clustered mobile devices to offload cellular traffic," in
INFOCOM'13: Proc. of the 32nd IEEE International Conference on
Computer Communication, 2013.
[18] T. Han, N. Ansari, M.Wu, and H. Yu, "On accelerating content delivery
in mobile networks," IEEE Communications Surveys Tutorial,
vol. 15, no. 3, pp. 1314–1333, 2013.
[19] X. Zhuo,W. Gao, G. Cao, and S. Hua, "An incentive framework for
cellular traffic offloading," IEEE Transactions on Mobile Computing,
vol. 13, no. 3, pp. 541–555, 2014.
[20] Y. Kim, K. Lee, and N. B. Shroff, "An analytical framework to
characterize the efficiency and delay in a mobile data offloading
system," in MobiHoc'14: Proc. of the 15th ACM International Symposium
on Mobile Ad Hoc Networking and Computing, 2014, pp. 267–
276.
[21] "Fonera 2.0n," http://www.fon.com/en/product/
fonera2nFeatures, Last visited: Jun. 2014.
[22] M. Stapp, B. Volz, and Y. Rekther, "The Dynamic Host Configuration
Protocol (DHCP) Client Fully Qualified Domain Name
(FQDN) Option," IETF RFC 4702, October 2006.