Exploring Mobile/WiFi Handover with Multipath TCP

Gokhan AY

Neşet doğanalp 114205

Ans Mohamed A Elbousiffi 109541

Undergraduate Project Report

submitted in partial fulfillment of

the requirements for the

degree of Bachelor of Science (B.S.)

in

Electrical and Electronic Engineering Department

Eastern Mediterranean University

June 2015

Approval of the Electrical and Electronic Engineering Department

Prof. Dr. Hasan Demirel
Chairman

This is to certify that we have read this thesis and that in our opinion it is fully adequate, in cope and quality, as an Undergraduate Project.

……………………………..	…………………………….. Co-
Supervisor	Supervisor

Members of the examining committee

Name	Signature
1.	……………………………………..
2.	……………………………………..
3.	……………………………………..

Date: ……………………………………..

Exploring Mobile/WiFi Handover with Multipath TCP

By

Gokhan AY

Neşet doğanalp 114205

Ans Mohamed A Elbousiffi 109541

Electrical and Electronic Engineering Department
Eastern Mediterranean University

Supervisor: Prof. Dr. Hasan AMCA

Keywords: Multipath TCP, Join capability, HMAC, Middlebox, 3G, 4G, Handover.

ABSTRACT The everlasting fact that data traffic always increases due to the expanding number of subscribers. Users always want to have a fast and reliable network speeds. As this growth in demand led data traffic to increase, operators started dedicating their work on some new and efficient ways that is different from 3G or 4G networks,one idea has been considered here offloading to Wi-Fi. Multipath TCP (MPTCP) which is an evolution of TCP that allows the simultaneous use of multiple interfaces for a single connection while standard TCP is kept the same for socket API to application. Multipath TCP allows to smooth handover from WiFi to 3G back and forth.In our experiment,we observe the workability of MPTCP over real WiFi / 3G or 4G networks and use our linux kernel implementation of Multipath TCP that results in smooth handover without lose of data.We observe that Multipath TCP energy consumption and performance depends on the modes of the kernel by performing perfect results in applications such as VoIP.

Acknowledgments

This work was supported by Comtech Ticaret LTD for addressing the issue. And technically by Computer center hardware hospital for providing us with the appropriate PC and also for IEEE club.

Table of Contents

List of Figures

1.Introduction

Nowadays, smartphones and tablets technologies are keep developing and the increase number of devices rapidly give rise to the number of subscribers in mobile data networks. Most of these devices nowadays are able to connect to multiple interfaces, most of the users seeking to have fast data network from their devices, it is natural that they are looking for the most apropriate mobile network to provide this kind of feature and maybe speed is number one on the list . This will result in need of more bandwith So some investigation predicted that in near future the deployment of 4G or LTE technology may not be sufficient to retain the growing demand of users.

However most of places like schools, cafeterias, houses, companies , hospitals...etc had started to equip these places with Wi-Fi or ADSL services to their costumers in order to make them able to connect internet at any time. This widespread avalibility of WiFi, ADSL or mobile data networks creates obvious opportunities for end-users and operators alike. Most of the devices are designed to have the ability to connect to more than one interface but unfortunately not all of them are utilized, only one at a time is used, if the user want to switch to another network or interface the switching operation(i.e. Handover) will lead to session termination and starting a new connection which in contrast will result in data loss like in regular TCP connection re-establishment. If the users devices use multiple links which is the case with tablets and smartphones, and if multiple links are used either simultaneously or alternatively the customers experience will increase while moving.

The devices uses either 3G/4G or WiFi network anywhere and at any time that user desires. So that these devices will be able to decide either selection of 3G or WiFi by smooth handover between these decisions. MPTCP works fine as regular TCP with unmodified applications, To illustrate more, no need to make some changes to applications because of this extention. MPTCP runs over Internet and this extension is standard at the IETF(i.e Internet Engineering Task Force) that enable a single TCP to use multiple interfaces on both ends clients and servers.

In this project, we show how MPTCP works by its advantages and places where it is used. Our aim is to understand how MPTCP works in theoritically and testing it in practically.

To experience practically how it works, we had to optimize the Linux MPTCP stack in order to make better observation and experiment WiFi/3G or 4G handover.

2.MPTCP HANDOVER

Multipath TCP (MPTCP) automatically distribute the traffic away from the congested links while being fair to single path TCP. One of the most important advantage of MPTCP is the choice of its backward compability with both the applications and the network.Today's networks are multipath: mobile devices have a number of different wireless interfaces, datacenters have excessive number of paths between servers, and multihoming has become the idealistic for big server farms. Meanwhile, TCP is essentially a single-path protocol, when a TCP connection is established, since transport layer is still not fully separated from network layer, the connection is tied to the IP addresses of the two communicating hosts. If one of these addresses changes, for whatever reason, the connection will fail. In fact, a TCP connection load cannot even be balanced across more than one path within the network, because this results in packet reordering, and TCP misinterprets this reordering as congestion and slows down. MPTCP preserves the standard socket API that is used by most internet applications.

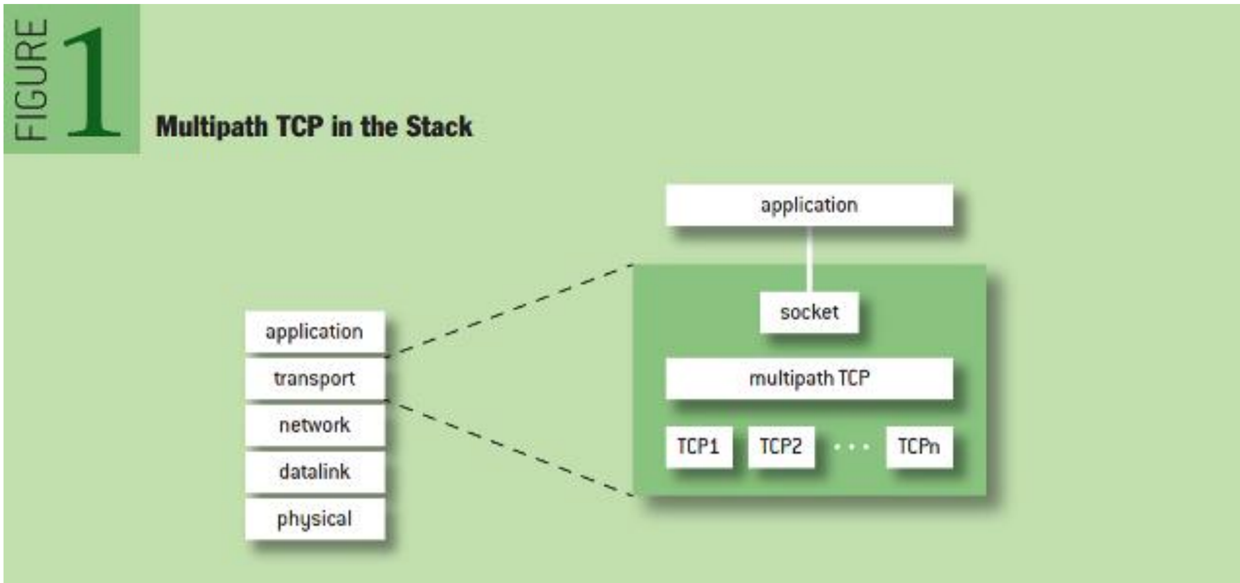
MPTCP connections contains at least one or more subflows and these subflows looks like a regular TCP connections to network. When the connection occurs, the options for MPTCP are included in SYN segments to verify the destination is MPTCP compatible or not by sending a MP_CAPABLE option in the SYN segment indicates that the client supports MPTCP. If it is compatible its connection will be uniquely identify by a token. Additional TCP subflow for each aditional interface will be joined or associated to this MPTCP connection by carrying the previously exchanged token in their three-way handshake. MPTCP breaks apart the applications's streams of bytes among all the established subflows to get the maximum benefit that the user can obtain of all the avaliable resources. There will be one subflow used on each interface which in term will improve the reaction to failures by maximizing the throughput.

In a mobile type of devices, mobiles usually have one or more wireless interface to be able to get connection to the internet (i.e Wi-Fi and 3G or 4G). Such a mobile architecture can support Multipath TCP's capabilities to use multiple interfaces in a manner to increase the performance of a data transmission but moreover to maintain the established multipath TCP connection while roaming around. In terms of keeping the connection alive while moving around regular TCP can not provide such a feature as any change occures in IP address of an end forces the hosts to restart all their established TCP connections. This type of problem is no more existing, with multipath TCP any change in any end's IP address will not force the MPTCP connection to be restared.

An MPTCP connection is associated to a set of underlying TCP subflows. TCP subflows can be added or removed from this set without impacting the MPTCP connection or the application. This ability to add and remove TCP subflows is key in MPTCP ability to support nodes.

2.1 What is Multipath TCP protocol ?

Multipath TCP (MPTCP) is new extension of TCP that allows the simultaneous use of multiple interfaces for a single connection while still presenting a standard TCP socket API to the application. The protocol specification of Multipath TCP has foreseen the different building blocks to allow transparent handover from Wi-Fi to 3G back and forth. Since Wi-Fi is widespread everywhere usually limited within a certain wireless radio range, and cellular services might decay or disappear from region to region, when uploading contents to a social site with traditional TCP, one may suffer stalled connections in mobile scenarios.



[11] Multipath TCP in Transport layer

2.2 Goles of Multipath TCP :

In supporting the use of multiple paths, Multipath TCP has the following two functional aims. Improving the Throughput: Multipath TCP must support the simultaneous use of multiple paths. To meet the minimum performance encouragement for deployment, a Multipath TCP connection over multiple paths should achieve no worse throughput than a single TCP connection over the best essential path.

Improve Resilience: Multipath TCP must support the use of multiple paths in changeable way for resilience reasons, by permitting segments to be sent and re-sent on any available path. It follows that, in the worst case, the protocol must be not less resilient than regular single-path TCP.

As spreading of traffic among available paths and reactions are done to congestion in accordance with resource principles, another important effect of meeting these goals is that the widespread of using Multipath TCP over the Internet shall improve overall network usefulness by moving the load away from congested bottlenecks and by taking advantage of supplementary capacity wherever possible. Furthermore, Multipath TCP should feature an automatic negotiation of its use. A host supporting Multipath TCP that requires the other host to do the same, must be able to detect surely whether this host does in reality support the required extensions, using them if so, and otherwise automatically falling back to single-path TCP as long as Multipath TCP is compatible with regular TCP.

In the normal Internet architecture, network devices works at the network layer and lower layers, with the layers above the network layer incorporate only at the end hosts. While this architecture was initially largely adhered to, showed in the following figure, this layering no longer reflects the "reality" in the Internet with the present of middleboxes. Middleboxes routinely intermediate on the transport layer; sometimes even completely terminating transport connections, thus leaving the application layer as the first real end-to-end layer, as shown in the second figure.

```
+-------------+                                       +-------------+
| Application |<------------ end-to-end ------------->| Application |
+-------------+                                       +-------------+
|  Transport  |<------------ end-to-end ------------->|  Transport  |
+-------------+   +-------------+   +-------------+   +-------------+
|   Network   |<->|   Network   |<->|   Network   |<->|   Network   |
+-------------+   +-------------+   +-------------+   +-------------+
   End Host           Router            Router           End Host
```

[7] Traditional Internet Architecture

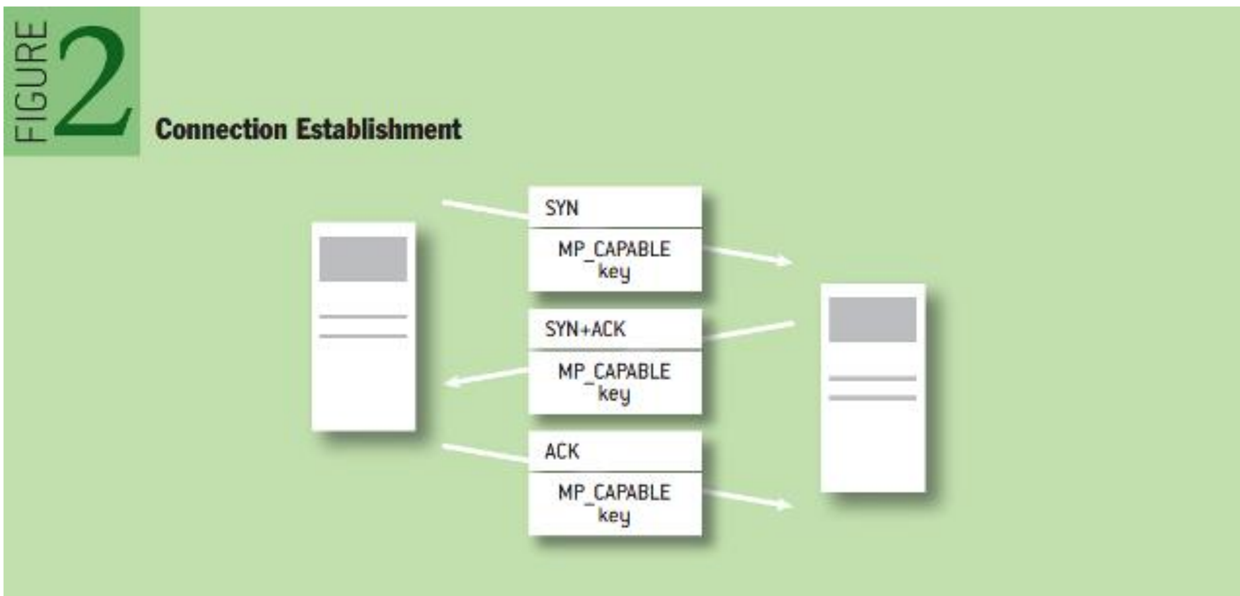
```
+-------------+                                       +-------------+
| Application |<------------ end-to-end ------------->| Application |
+-------------+                     +-------------+   +-------------+
|  Transport  |<------------------->|  Transport  |<->|  Transport  |
+-------------+   +-------------+   +-------------+   +-------------+
|   Network   |<->|   Network   |<->|   Network   |<->|   Network   |
+-------------+   +-------------+   +-------------+   +-------------+
                                       Firewall,
   End Host           Router         NAT, or Proxy       End Host
```

[7] Internet Reality nowadays

Middleboxes that interpose on the transport layer result in loss of "holding together ", that is, they often hold "hard" state that, when lost or corrupted, results in loss or corruption of the end-to-end transport connection.

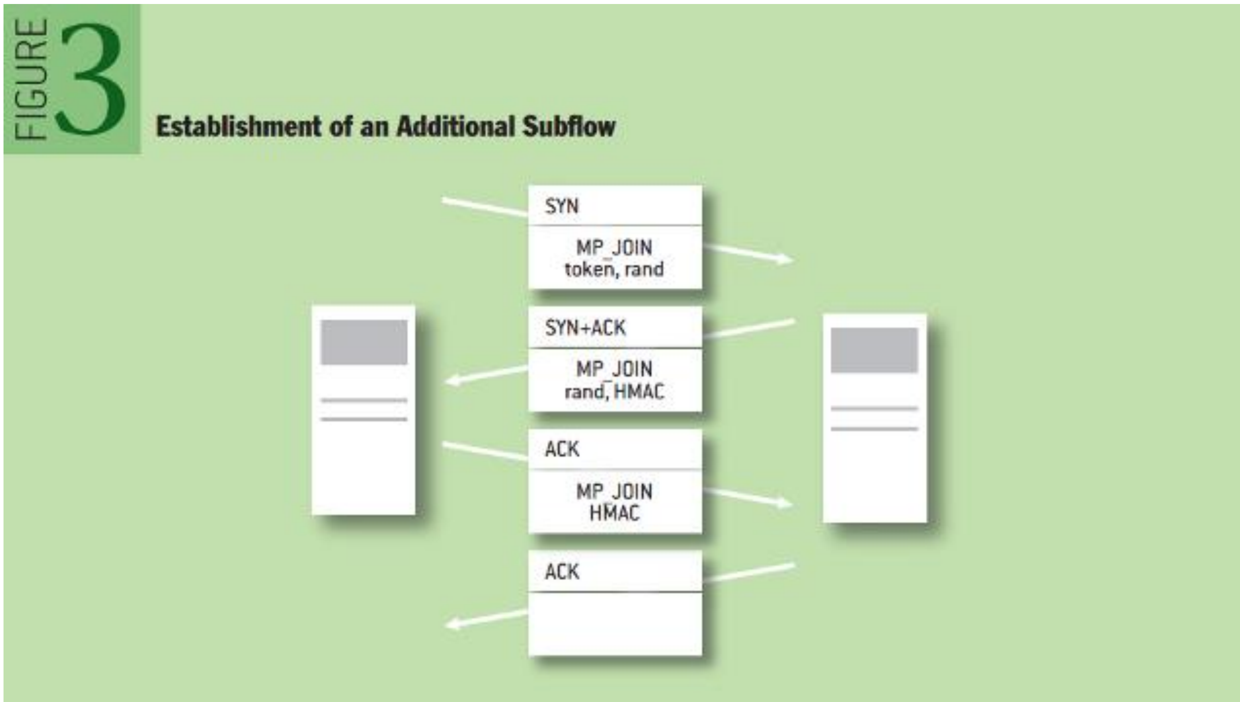
2.3 Adding and Removing TCP Subflows:

This is the same way as for initiating a regular TCP connection, but the SYN, SYN/ACK, and ACK packets the three handshake packets also carry the MP_CAPABLE option to indicate that the host is capable. This is variable length and serves multiple aims. Firstly, it verifies whether the remote host supports Multipath TCP; secondly, this choice allows the hosts to exchange some information to authenticate the establishment of additional subflows.



[11] Connection initiation

When the first MPTCP connection is established, each end knows the IP addresses of the other end. When an aditional interface is provided , SYN packet is sent with a JOIN option to the server known address. This JOIN option informs the server about which connection this subflow belongs to.



[11] Adding Subflow

To illustrate more At this point, the Multipath TCP connection is established and the both ends can exchange TCP segments via the 3G path for example. How could the smartphone or tablet also send streams of data through this Multipath TCP connection over another interface and let us assume it is Wi-Fi interface? Naively, the smartphone or tablet could simply send some of the packets over the Wi-Fi interface; however, most internet server providers will discard these packets, as they would rely on the source address of the first interface which is 3G interface. Maybe the client could tell the server the IP address of the Wi-Fi interface and use that address when it sends over Wi-Fi also. Unfortunately, this will rarely work because of the enemy of MPTCP, firewalls and similar middleboxes on the Wi-Fi path expect to see a SYN packet before they see data packets. The only solution for this problem is to perform a full SYN handshake on the Wi-Fi path before sending any packets over that way, so this is what Multipath TCP does. This SYN handshake carries the MP_JOIN TCP option, providing sufficient information to the server that it can securely identify the correct connection with which to associate this additional subflow. The server replies with MP_JOIN in the SYN+ACK, and the new subflow is established. The mobile client can not connect to a address if the later is behind a NAT or a proxy. In this case the server sends an Add Adress option to on existing subflow. The server will try to establish a new TCP subflow to newly received address and Remove Address option is similar to Add Address when one address is unavailable.

2.4 HANDOVER MODES :

According to users, there are three important factors which are really important for them. The first factor is the behavior of the data transfer ,i.e, how fast is it ?. Some user wants fastest possible data transfer. However some of the users will prefer longer battery life time which is the major work is going on nowadays, while some user will prefer a data transmition pricing.Because 3G networks are charged in terms of the number of transmitted bits or packets per time period. Some users will prefer cheaper networks. And these modes are designed to meet the needs for most of the users.

MPTCP kernel has three modes these are;

a) Full MPTCP Mode
b) Backup Mode
c) Single-Path Mode

a) Full MPTCP Mode:

In this mode, MPTCP uses all subflows between the two ends, where the full mesh of TCP subflows among the client's and the server's addresses is formed. This type of mode provides maximum throughput rate.

b) Backup Mode:

In this mode MPTCP opens TCP subflows over all interfaces but uses only a some of these to transport data segments. This will result in optimize cost and battery lifetime by sending data segments over the cheaper interfaces. That's why in nowadays networks it is obvious for MPTCP to prefare Wi-Fi over a cellular networks.

C) Single-Path Mode:

In this mode the MPTCP work similar to Backup Mode. However the only thing different from backup mode is a single subflow is established in any moment. When the interface goes down , Single-Path establishes a new TCP subflow over the other interface. This is possible due to break-before-make design of MPTCP. MPTCP is able to recover from this failure by establishing a new TCP subflow and continue the data transmission without loss of any data. Later on new MPTCP subflow can be established.

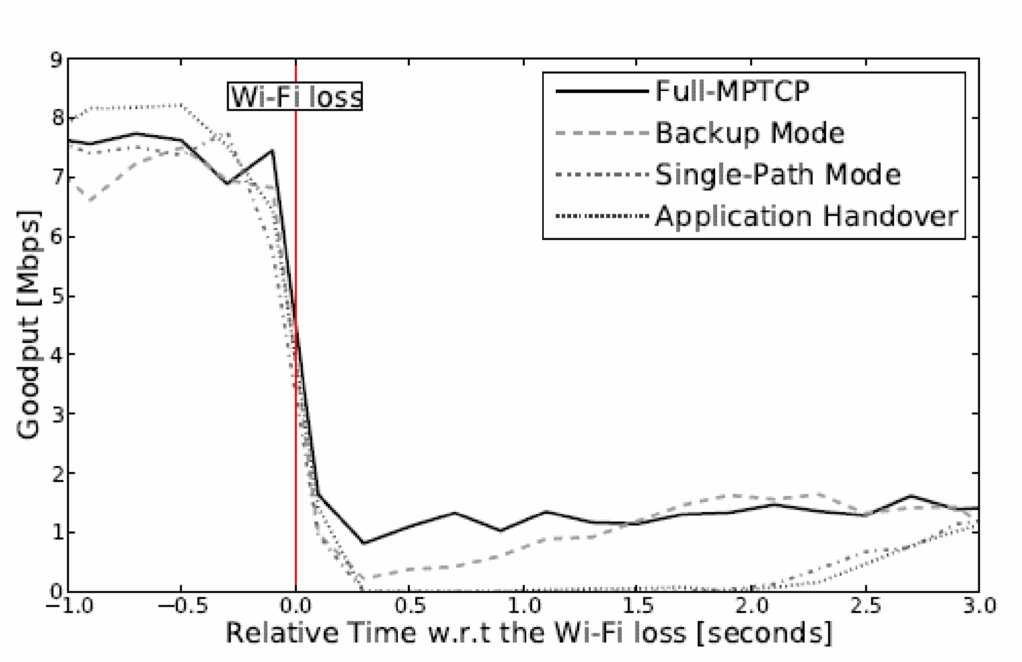
3.EVALUATİON BY MEASUREMENTS:

3.1 Download Goodput:

During this experiment we make a evaluation of the goodput for a simple HTTP application during vertical hangover. The figure which is below show the goodput averaged over 200ms intervals over 20 measurements by using different handover modes. The x-axis shows the offset compared to the time when the client lost its WiFi connection.

This graph also show the goodput of an Application-level Handover which is modified HTTP client (running over regular TCP) that monitors the changes in the routing table to detect the faikure of the WiFi interface. Upon detecting the collapse, our application restarts the HTTP download. Supporting such handover requires significant changes to the application, and is specific to the HTTP protocol; in contrast, MPTCP does not require any application-level modification.

We can see that from the graph when a loss happen to a specific network like a WiFi network in the following graph, it becomes unavailable for any reason like signal loss etc, the Backup Mode behaves similar to Full MPTCP mode which we expected. The TCP subflow on the 3G interface is already established in both cases. These differences between these modes come from the fact that in Backup mode only including three segments belonging to the three way handshake event which has been sent over 3G while the Full MPTCP mode data segments have been transmitted.



[1] Recovery from a loss

According to this graph, the Full MPTCP mode recovers quickly due to the congestion window on the 3G subflow larger than in the backup mode where the congestion window still has the initial value.

The single path mode and Application Level Handover are both need to perform a three way handshake and then reinject and send data into new subflow or into a new connection. After three seconds all modes obtain the 3G network average download speed.

We make our measuremts by forcing the 3G interface in order to remain it active and observe that there is not much differences between the Backup and Single path mode. In this case the effect of performance of three handshake is negligible.

3.2 Application Delay:

In most of the communication applications such as VoIP, the time of delay is an important factor. The delay of application is measured by sending blocks of data which is tagged with a timestamp for reorganizing the received packets in the receiver buffer. The time differences between the timestamps give us the variation in application delay which is called jitter.

The measurements are done by using 500Kbps as a download speed to the client. However, the WiFi and 3G are both activated and sent the data segments by choosing different handover modes. Then after five seconds we close the WiFi access point and traffic has to move to 3G interface. According to our measurements the Backup Mode and is behaving similar to Full-MPTCP mode. A sending speed 500kbps does not fill the pipe over the WiFi interface because the MPTCP chooses the WiFi path which has higher bandwith and lower RTT, no data is send over 3G network even in Full MPTCP mode.

In Single Path mode no subflow is established over 3G prior to the vertical handover event. The 3G interface is idle and thus first needs to come up before establisment of the new subflow. This will take up a two second to delay of an application.

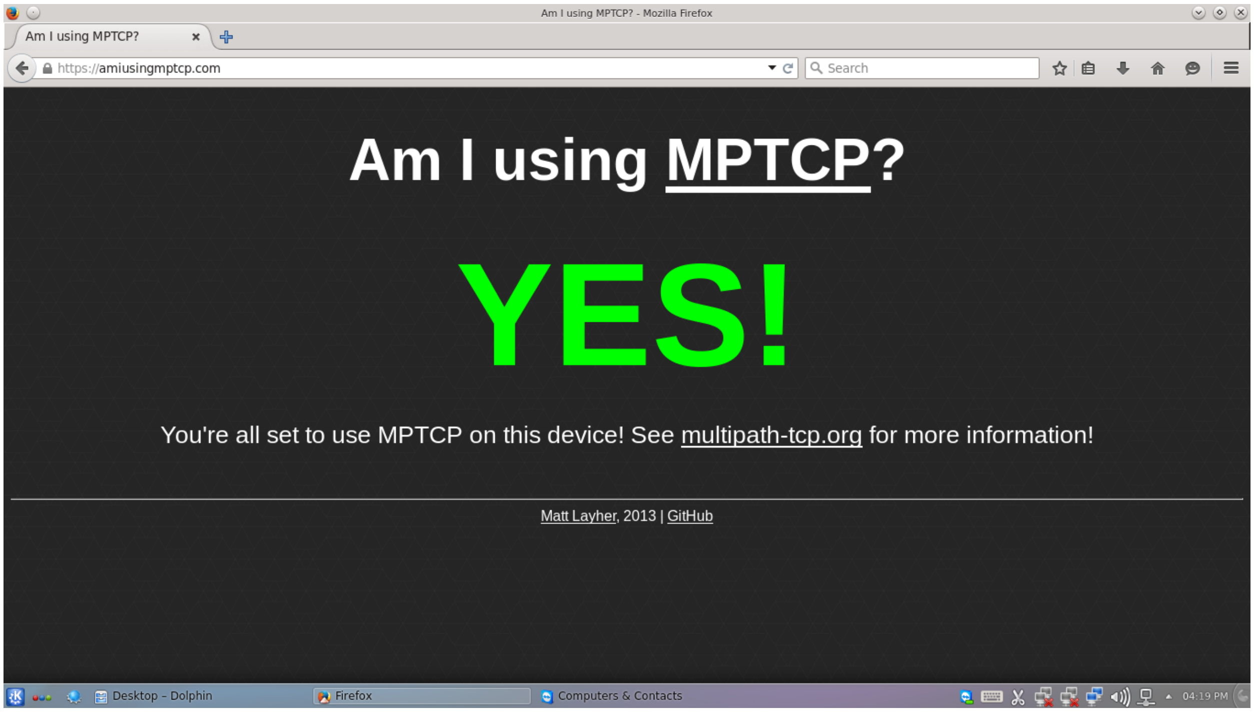
3.3 Impact on Existing Applications:

The API sockect remains same in MPTCP not like in SCTP so that no any change occurs and API is kept same which is a plus for MPTCP that will lead to a wide spreading of this new protocol in the future with our any huge changes in the network architecture . So this can be used by any TCP application. Skype is the commercial VoIP application which is able to be used over both TCP and UDP. By using Skype we observe that the application had very bounded restrictions on packet-level delays. İn order to show a worst case for MPTCP handover.

In order to test Skype over MPTCP, we force the application to pass through an MPTCP-enabled HTTP-proxy by blocking UDP and other TCP ports on the client firewall. Otherwise Skype will by default try to use other access points such as UDP or regular TCP over the public Skype servers. Finally we observe that Skype was seamlessy smooth handovering without any loss of data and connection.

3.4 Our Experement：

We have implemented Multipath TCP using [19] Linux Kernel Ubuntu system 14.04.2,it is implemented on 14.04 version but we implemented it on 14.04.2 and we configure it manually not automatically as it is shown clearly on the webpage of the implementation.



We have tested handover in various scenarios; using different cellular networks along with WiFi and ethernet connections for our aim handover was smooth and achieved from WiFi to 3G/4G and visa versa, what is more than that is the backup mode does resilience in moving between the two interfaces and cost avoidance or battery consumption avoidance by moving most of the load to WiFi network.

4. Summary of Experimnts:

We observe that the MPTCP allows a smooth handover between interfaces such as WiFi/3G or WiFi/4G and also unmodified applications such as Skype to continue to function while handover from WiFi to 3G takes place. MPTCP has three modes and these modes have different attribute, while one has faster data connection rate and the other has greater battery life. Every user can choose their mode according to their opportunity. The Full MPTCP mode offers the smoothest handover and the Backup Mode performs the performance mode over Single-Path Mode during handover. The Backup mode always gives the good performance at modest energy cost.

5. Conclusion and Further Work:

This experiments are done by using both WiFi and 3G in order to prove the smooth handover of MPTCP. We have tested all three modes of MPTCP . These are Full MPTCP Mode, Single-Path Mode and Backup Mode. Our measurements show that MPTCP can quickly recover from WiFi loss in presence of a 3G interface with only a small impact on the application delay and goodput.

Our further work will be testing the MPTCP kernel performance and battery life by using different tablets and smartphones by proving that MPTCP works efficiently in these devices.

6. Referances:

1. FON WIRELESS, Ltd. FON website available at http://www.fon.com, April 2012.
2. A. Balasubramanian, R. Mahajan, and A. Venkataramani. Augmenting mobile 3G using WiFi. In Proc. Mobisys. ACM, 2010.
3. H. Falaki, D. Lymberopoulos, R. Mahajan, S. Kandula, and D. Estrin. A First Look at Tra_c on Smartphones. In Internet Measurement Conference (IMC'10), pages 281{287, 2010.
4. A. Ford, C. Raiciu, M. Handley, and O. Bonaventure. TCP Extensions for Multipath Operation with Multiple Addresses. Internet-Draft draft-ietf-mptcp-multiaddressed-06, Jan. 2012.
5. J. R. Iyengar, P. D. Amer, and R. R. Stewart.Concurrent Multipath Transfer using SCTP Multihoming over Independent End-to-End Paths.IEEE/ACM Transactions on Networking,14(5):951{964, 2006.
6. S. J. Koh, M. J. Chang, and M. Lee. mSCTP for Soft Handover in Transport Layer. Communications Letters, IEEE, 8(3):189 { 191, March 2004.
7. A. Ford, C. Raiciu, M. Handley, S. Barre, J. Iyengar. March 2011. Architectural Guidelines for Multipath TCP Development. RFC6182; https://tools.ietf.org/html/rfc6182#section-1.2
8. R. Koodli. Mobile IPv6 Fast Handovers. RFC 5568,RFC Editor, July 2009.
9. C. Pluntke, L. Eggert, and N. Kiukkonen. Saving Mobile Device Energy with Multipath TCP. In ACM Workshop on MobiArch, pages 1{6, 2011.
10. C. Raiciu, D. Niculescu, M. Bagnulo, and M. Handley. Opportunistic mobility with multipath TCP. In ACM Workshop on MobiArch, pages 7{12, 2011.
11. Christoph Paasch and Olivier Bonaventure. 2014.Multipath TCP : Decoupled from IP, TCP is at last able to support multihomed hosts. Volume 12, issue 2. Available at: https://queue.acm.org/detail.cfm?id=2591369
12. C. Raiciu, C. Paasch, S. Barre, A. Ford, M. Honda, F. Duchene, O. Bonaventure, and M. Handley. How Hard Can It Be? Designing and Implementing a Deployable Multipath TCP. In Networked Systems Design and Implementation (NSDI '12), 2012.
13. A. Snoeren and H. Balakrishnan. An end-to-end approach to host mobility. In Proc. Mobicom, pages 155{166. ACM, 2000.
14. R. Stewart, Q. Xie, M. Tuexen, S. Maruyama, and M. Kozuka. Stream Control Transmission Protocol (SCTP) Dynamic Address Reconfiguration. RFC 5061, RFC Editor, September 2007.

15. D. Wischik, M. Handley, and M. B. Braun. The Resource Pooling Principle. SIGCOMM Computer Communication Review, 38(5):47{52, Sept. 2008.
16. D. Wischik, C. Raiciu, A. Greenhalgh, and M. Handley. Design, Implementation and Evaluation of Congestion Control for Multipath TCP. In Networked Systems Design and Implementation (NSDI '11), 2011.
17. olivier.bonaventure, Sébastien Barré, Christoph Paasch, Grégory Detal, Mark Handley, Costin Raiciu, Alan Ford, Micchio Honda, Fabien Duchene and many others, March 2013.Decoupling TCP from IP with Multipath TCP Olivier Bonaventure http://inl.info.ucl.ac.be http://perso.uclouvain.be/olivier.
18. A. Ford, C. Raiciu, M. Handley, S. Barre, J. Iyengar. March 2011. Architectural Guidelines for Multipath TCP Development. RFC6182; https://tools.ietf.org/html/rfc6182#section-1.2
19. MultiPath TCP Linux kernel implementation. http://mptcp.info.ucl.ac.be/.