Towards Secure Multipath TCP Communication

Zeeshan Afzal

Faculty of Health, Science and Technology

Computer Science

LICENTIATE THESIS | Karlstad University Studies | 2017:12

Towards Secure Multipath TCP Communication

Zeeshan Afzal

LICENTIATE THESIS | Karlstad University Studies | 2017:12

Towards Secure Multipath TCP Communication

Zeeshan Afzal

LICENTIATE THESIS

Karlstad University Studies | 2017:12

urn:nbn:se:kau:diva-48172

ISSN 1403-8099

ISBN 978-91-7063-763-6 (print)

ISBN 978-91-7063-764-3 (pdf)

Distribution:
Karlstad University
Faculty of Health, Science and Technology
Department of Mathematics and Computer Science
SE-651 88 Karlstad, Sweden
+46 54 700 10 00

Print: Universitetstryckeriet, Karlstad 2017

WWW.KAU.SE

Towards Secure Multipath TCP Communication

Zeeshan Afzal
Department of Mathematics and Computer Science
Karlstad University

Abstract

The evolution in networking coupled with an increasing demand to improve user experience has led to different proposals to extend standard TCP. Multipath TCP (MPTCP) is one such extension that has the potential to overcome a few inherent limitations in standard TCP. The design and deployment of the protocol is progressing, but most of the focus has so far been on its compatibility. The security aspect is limited to making sure that the MPTCP protocol itself offers the same security level as standard TCP.

The topic of this thesis is to investigate the unexpected security implications raised by using MPTCP in the traditional networking environment. The Internet of today has security middleboxes that perform traffic analysis to detect intrusions and attacks. Such middleboxes make use of different assumptions about the traffic, e.g., traffic from a single connection always arrives along the same path. This along with many other assumptions may not be true anymore with the advent of MPTCP, as traffic can be fragmented and sent over multiple paths simultaneously.

We investigate how practical it is to evade a security middlebox by fragmenting and sending traffic across multiple paths using MPTCP. Realistic attack traffic is used to evaluate such attacks against the Snort IDS to show that these attacks are feasible. We then go on to propose possible solutions to detect such attacks and implement them in an MPTCP proxy. The proxy aims to extend the MPTCP performance advantages to servers that only support standard TCP, while ensuring that intrusions can be detected as before. We also investigate the potential MPTCP scenario where security middleboxes only have access to some of the traffic. We propose and implement an algorithm to perform intrusion detection in such situations and achieve a nearly 90% detection accuracy. Another contribution of this work is a tool that converts IDS rules into synthetic attack traffic.

Keywords: network security, TCP, MPTCP, IDS, Snort, edit-distance

Acknowledgements

I would like to start by thanking the people who have enabled me to be in the position I am today. I will be forever grateful to Magnus Almgren, who I met as a student at Chalmers University of Technology. He took me under his guidance and inspired me to pursue an academic career. Judith Rossebø, who I met at ABB in Oslo during the work for my Master's thesis, deserves a special mention for her guidance. I am thankful to my main advisor Stefan Lindskog, who trusted a young Master's student with no previous research experience and provided valuable feedback and support throughout. I am grateful to Anna Brunstrom, Johan Garcia and all of my other co-authors for their brilliant ideas, helpful critique, and feedback.

This thesis is part of a journey that started with my grandfather's wish for me to secure a doctoral degree. Grandpa, I wish you could be here today to witness how far I have come to fulfill your dream. A special mention goes to my parents and siblings back in Pakistan for their love and support throughout my life. My family in Norway has provided unwavering support and made it easier for me to stay away from home and undertake a research career. Thank you to Noor, Ismail and Hadi for being constant sources of happiness in my life. I offer my gratitude to my fiancée Khadija for being my good luck charm and a blessing in my life. I can not help but mention Liverpool FC here. You have been there with me through both the good and bad times. Thanks for all the memories and lets make many more. You will never walk alone!

Finally, I would like to thank all my colleagues at the department for the technical and non-technical discussions. The "innebandy" squad has made me appreciate another sport than football. Thanks for that.

The work in this thesis was carried out in the High Quality Networked Services in a Mobile World (HITS) project, funded partly by the Knowledge Foundation of Sweden.

Karlstad University, March 14, 2017 Zeeshan Afzal

List of Appended Papers

This thesis is based on the work reported in the following appended papers.

I. Zeeshan Afzal and Stefan Lindskog. Multipath TCP IDS Evasion and Mitigation. In Proceedings of the 18th Information Security Conference (ISC), Trondheim, Norway, September 9–11, 2015.

II. Zeeshan Afzal, Stefan Lindskog, Anna Brunstrom, and Anders Lidén. Towards Multipath TCP Aware Security Technologies. In Proceedings of the 8th IFIP International Conference on New Technologies Mobility and Security (NTMS), Larnaca, Cyprus, November 21–23, 2016.

III. Zeeshan Afzal, Johan Garcia, and Stefan Lindskog. Partial Signature Matching in an MPTCP World using Insert-only Levenshtein Distance. Under Submission.

IV. Zeeshan Afzal and Stefan Lindskog. IDS Rule Management Made Easy. In Proceedings of the 4th International Workshop on Systems Safety and Security (IWSSS), Ploiesti, Romania, June 30–02 July, 2016.

The papers have been subjected to minor editorial changes.

Comments on my Participation

For all the papers, I came up with the initial idea and did most of the writing. The work was conducted together with my co-authors. Stefan Lindskog has been most influential for all the papers.

- In Paper II, Anders Lidén proposed the different MPTCP proxy scenarios and suggested where the proxy should be placed. Anna Brunstrom provided networking and protocol support.
- In Paper III, Johan Garcia proposed the insert-only Levenshtein distance algorithm and wrote about it in the paper.

Other Publications

- Zeeshan Afzal and Stefan Lindskog. Automated Testing of IDS Rules. In Proceedings of the 6th International Workshop on Security Testing (SECTEST), Graz, Austria, April 13, 2015.
- Zeeshan Afzal, Stefan Lindskog and Anders Lidén. A Multipath TCP Proxy. In Proceedings of the 11th Swedish National Computer Networking Workshop (SNCNW), Karlstad, Sweden, May 28–29, 2015.
- Zeeshan Afzal, Judith Rossebo, Batool Talha, and Mohammad Chowdhury. A Wireless Intrusion Detection System for 802.11 Networks. In Proceedings of the IEEE International Conference on Wireless Communications, Signal Processing and Networking (WiSPNET), Chennai, India, March 23–25, 2016.

Contents

Introductory Summary

1 Introduction

As technology is advancing, applications demand more and more from the network, such as a higher bandwidth and an enhanced availability of the connections in addition to reliable transmission. Gone are the days when peers were hard linked with single paths. Today's peers often boast multiple paths between them. At the transport layer, TCP [7] is the most commonly used protocol on the Internet to deliver end-to-end services. However, under these increased demands, TCP has fallen short for a number of reasons. TCP communication is limited to a single path per connection. During the life-time of a TCP connection, the pair of IP addresses and port numbers involved can not change. Thus, the protocol does not have the ability to utilize multiple paths available between end-hosts or to have a fail safe function. If a TCP connection fails, the communication stops.

While many different proposals have been suggested to overcome the shortcomings [13, 22], none has the same potential to succeed as Multipath TCP (MPTCP) [12]. Specified by IETF as an experimental standard in early 2013, MPTCP is expected to overcome the inherent weaknesses in single-path TCP by making it possible for multi-homed end-hosts to use multiple interfaces together for a higher throughput and/or availability. The whole design of MPTCP is evolutionary rather than revolutionary to ensure its operating feasibility over the existing Internet and applications. Since its specification, many independent MPTCP implementations exist [2, 4, 10, 17, 18] and researchers have already shown how MPTCP can outperform TCP in a number of situations [8, 20].

In the wake of new evolutionary networking protocols such as MPTCP, ensuring the security of a network is becoming an increasingly challenging task. The traditional approach to security may no longer be sufficient. One main reason for this is that the basic assumptions and expectations that the traditional security devices have from the network and observed traffic are not always true anymore. The same is the case with MPTCP. It changes the things that are taken for granted by security devices that are used to analyze standard TCP and know what to expect from TCP flows. The focus of this thesis is on the auxiliary security impacts of MPTCP caused by the non-conformance of traffic to basic assumptions. These are not the security issues for MPTCP as a protocol itself but the security implications of using MPTCP over the existing Internet, i.e., an Internet that consists of millions of middleboxes that are used to enforce security.

Deep packet inspection (DPI) is one of the technologies employed to ensure the security of networks. These on-path security middleboxes have worked quite well until the introduction of MPTCP. Despite the successful efforts of developers to design a protocol that works over the existing networking infrastructure, MPTCP has far-reaching and somewhat unexpected implications for network security. Most of the existing network security technologies based on DPI can not recognize and thus analyze MPTCP traffic. Many of the basic assumptions made by these technologies are no longer true

with MPTCP [19]. For instance, security devices that track connections and classify traffic based only on their five-tuple will see the paths (sub-flows) of an MPTCP connection as independent TCP connections with no correlation. Thus, they can not reassemble MPTCP traffic correctly. The fact that MPTCP allows a sender to use all available paths simultaneously enables the fragmentation of data among the paths in a way such that there is not enough information on any of the flows for an IDS to recognize whether the data being sent are malicious. This opens the possibility of cross-path data fragmentation attacks as described in Paper I.

In this thesis, we make an effort to investigate the feasibility of attacks based on security implications of MPTCP and then propose solutions to defend against such attacks. Specifically, we investigate the attacks made possible by fragmenting a data-stream among multiple active paths and their impact on security middleboxes. We differentiate between two scenarios where the simple case is when security devices can observe all MPTCP traffic, but can not recognize it and thus can not reassemble correctly. We propose and implement a solution to perform correct correlation and reassembly of MPTCP traffic in that case. The solution is implemented in an MPTCP proxy solution (Paper II) to ease the protocol deployment during the transition stage and extend secure benefits to more hosts. The other scenario is when MPTCP operation causes security devices to observe some of the traffic from connections only and they have to make decisions based on that alone. We investigate this problem in detail and propose an algorithmic solution to the problem. See Paper III for further details.

The rest of the introductory summary is structured as follows. Section 2 provides some background and discusses related work. The main objectives of this thesis and the research questions addressed are outlined in Section 3. Section 4 summarizes the main contributions of this work. Section 5 relates the work to the field of computer science. A short summary of the appended papers is presented in Section 6. Finally, Section 7 provides concluding remarks and an outlook.

2 Background

This section provides the necessary background to understand the topics under discussion in this thesis. The first part of the section emphasizes the need for network security in the Internet. Then, we discuss some background on the MPTCP protocol while the last part of the section provides related work and context on the security impacts of MPTCP.

2.1 Network Security

The number of devices connected to the Internet are increasing everyday. Such an influx was not foreseen when the inception of the Internet took place. At the time, the Internet was supposed to be a medium that would allow scientists to share data amongst each other. Hence, the protocols that

soon became the de-facto language for the Internet were not developed with security in mind. Today, the Internet has grown substantially with nearly half of the world's population using it, according to the latest estimate [1]. Its usage varies from doing simple tasks such as browsing the web to sensitive tasks such as online bank transactions via internet banking. More and more information is being made available online. The sensitivity of data has increased many times. Therefore, it is imperative to detect malware and attacks in the traffic. If not, sensitive information can be stolen via attacks that can deter security. Everyday, thousands of security breaches happen across the world causing damage to reputation and costing millions of dollars [16]. The number of security incidents per day are on the rise, and so is the cost incurred by each incident and the security budget dedicated by organizations.

Since TCP, the main transport protocol on the Internet, had no inherent security mechanism, the research community has proposed many extensions of the protocol and different architectures to make the Internet more secure. One common security architecture is the utilization of intrusion detection systems (IDSs) as middleboxes in the network. Such IDSs are deployed on paths where network traffic traverses between hosts. They analyze the traffic passing through them and try to detect attack instances based on either identifying abnormal behavior and/or pre-defined attack patterns. The IDSs that rely on pre-defined attack patterns or signatures compare observed traffic against a given database of attack signatures and try to establish whether the conditions in any of the rules are met. The signature database consists of thousands of signatures that are devised by security experts to detect well known attacks and malware. If an IDS detects an attack instance, it generates an alert and logs the intrusion attempt for the administrator.

2.2 MPTCP

TCP is the most commonly used transport layer protocol to reliably deliver in-order data from one application to another. However, networking and applications have evolved since its proposal. Modern applications require more than just reliable in-order delivery because the networking between hosts allows for multiple paths. There is a need to combine connections for a higher bandwidth. Some applications require additional resilience where hosts are always expected to stay connected. TCP has no means to support these use cases. This was identified as early as 1995 [13]. Different proposals have been suggested to resolve the problem; however, none of them provide the same promise as MPTCP.

MPTCP [12] is an extension to standard TCP that is close to an official IETF standardization. It enables a TCP connection to operate across multiple paths at the same time. This brings the support to a number of use cases, which was not possible before. It is designed to run on top of today's Internet infrastructure and has a fallback mechanism that allows it to be backward compatible. MPTCP seems the same as standard TCP to a network. From a higher level view, an MPTCP connection consists of one or more TCP flows

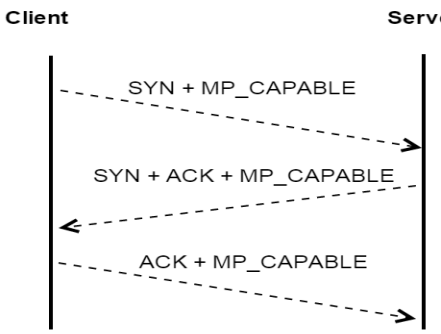

Figure 1: MP_CAPABLE handshake.

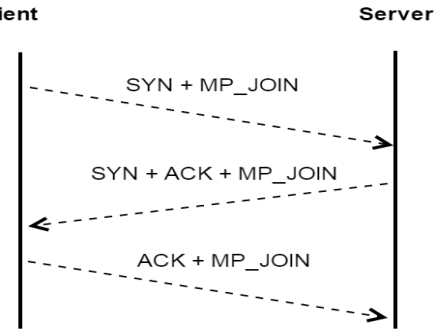

Figure 2: MP_JOIN handshake.

(referred to as subflows in MPTCP). Each of these subflows is a proper TCP connection, but with additional MPTCP options that allow every subflow to be linked to an MPTCP connection. A detailed technical discussion of the complete protocol is beyond the scope of this work. However, we discuss some key concepts of MPTCP that are most relevant for this thesis in the subsequent text. See [12] for further details.

2.2.1 MPTCP Connection and Subflows

An MPTCP connection consists of a number of TCP flows that are linked together. The establishment of a connection takes places via a three-way handshake with an *MP_CAPABLE* option attached to all the exchanged messages. This option announces to the remote host that the sender supports MPTCP and wants to use it. It also carries information, e.g., random keys, that could be used later in the connection's lifetime. Figure 1 shows the interaction between an MPTCP capable client and server to successfully complete the MPTCP handshake. This initial handshake is known as the *MP_CAPABLE* handshake.

Once an MPTCP connection is established, additional subflows can be added or removed from the connection on the fly as required. This is achieved in the same way as initiating a new MPTCP connection, but instead making use of the *MP_JOIN* option. The option informs the remote host that the connection request is not for a new connection but relates to an existing one. Figure 2 shows the handshake involved when a new subflow is added to an established MPTCP connection. This handshake is called the *MP_JOIN* handshake. If a subflow is removed from a connection that has more than one subflow, then the overall connection still survives and keeps operating as normal.

2.2.2 Transfer of Data

MPTCP ensures reliable and in-order delivery of the data across all subflows of an MPTCP connection using a data sequence number. Every subflow has its own transmission window (sequence number space), and the data sequence signal (DSS) option of MPTCP is used to map the subflow sequence space to

the overall MPTCP connection space. This enables data to be retransmitted on different subflows in the event of failure. On the receiver side, MPTCP uses a single receive window across all subflows. The MPTCP standard enables the sender to decide how exactly to send the data among the available subflows or paths. The common use case for an increased throughput is using all available paths (subflows) simultaneously [12] as long as enough data are available. The sender tells the receiver how the data are scheduled among the subflows using the DSS option. The receiver uses this information to reorder the data received over different subflows before passing them on to the application layer in the correct order.

2.2.3 Deployment

MPTCP was mainly designed with mobile devices in mind. However, it has already found use-cases in many unexpected areas. One such use case is in the data centers. Today, the large server farms in data centers provide content to end-users. The network topology within a data center is designed to allow for multiple paths between hosts to ensure redundancy. In such a setting, MPTCP can be used to enhance performance in the data center [6, 20]. Additionally, research has shown MPTCP's effectiveness in reducing download times and latencies for mobile users [6, 8]. Korean Telecom has utilized MPTCP to enable users to reach bandwidth of up to 1 Gbps [5]. There exist many implementations of MPTCP on a number of operating systems. It is available for Linux [17], BSD [2] and Android [10]. Commercially, Apple has implemented it in iOS7 [4] for Siri and OS X Yosemite [18]. See [6] for further details.

2.3 Security Implications of MPTCP

Despite its single-path nature, years of research have meant that TCP ensures a specific level of security. The security infrastructure such as the middleboxes on the Internet are used to it. Therefore, it is natural to consider TCP as a reference point when discussing the security of a network. With MPTCP, the design goal was to ensure that there are no new vulnerabilities and the security level provided by MPTCP is at least the same as TCP. There has been a protocol security assessment of MPTCP [3] that investigated possible attacks on the protocol and proposed some solutions, which have since been slowly integrated in the later specifications of the protocol. However, the security aspects in the MPTCP design have been considered with the protocol in isolation. The unexpected and auxiliary security impacts caused by MPTCP by its operation in the current networking environment have not been explored.

Indeed, MPTCP can be substantially different from TCP from a security point of view. A study conducted by Pearce and Zeadally [19] outlined the main network security implications of MPTCP and the key security differences between TCP and MPTCP. They suggested at least four different security impacts of MPTCP on current network security. These impacts

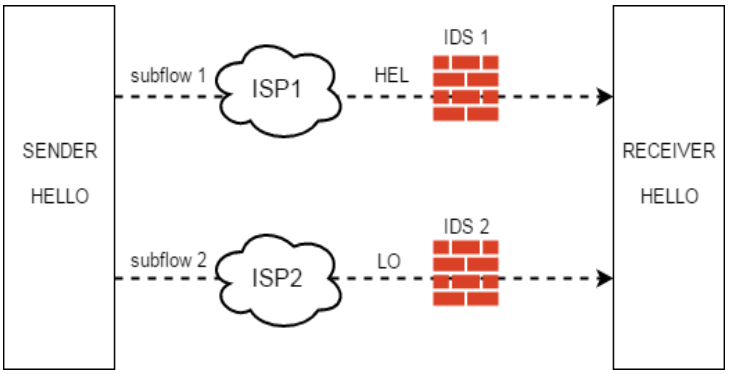

Figure 3: Data transfer using MPTCP [19].

include *broken correlation*, *moving targets*, *split traffic paths*, and *active control avoidance*. Some of these security impacts are mainly applicable only in the transitional stage where the majority of devices do not support MPTCP. However, other security impacts will still apply even when MPTCP is widely deployed. These impacts will require a change in the way network security is conducted. Below we summarize some of the key dissimilarities between MPTCP and TCP that can have a security impact. The discussion is inspired by the work in [19]. It should be noted that the reference to MPTCP in the security context assumes an MPTCP connection with at least two subflows.

2.3.1 Data Fragmentation

As discussed in Section 2.2.2, MPTCP enables a sender to utilize all available subflows simultaneously and fragment a data stream across them. This is unlike TCP, where a data stream from a connection is transferred over the same path. Although it can be fragmented along the same path, the data can not be distributed across multiple paths in TCP. In such an instance, assuming that the security devices such as IDSs can observe traffic from all paths, they have to understand the MPTCP protocol and reassemble data correctly in the correct order before performing their functions.

Furthermore, with TCP, the security devices assume observation of all traffic for a given connection. In the case of MPTCP, it is likely that the subflows used in an MPTCP connection belong to different networks owned by different ISPs. The security devices on each path of these subflows can not observe all traffic from a given connection anymore. Figure 3 depicts such a scenario recreated from [19]. The sender has established an MPTCP connection with two subflows to the receiver. The first subflow is established over a path owned by ISP 1 while the second subflow is established over a path owned by ISP 2. Both ISPs can be independent and competitors. Each subflow has an IDS scanning for an attack signature “Hello”. The sender wishes to send the same 5-byte message “Hello” to the receiver. The sender can utilize both flows together and send 3 bytes on subflow 1 and the remaining 2 bytes on subflow 2. The end result will be that the receiver will collect data from both subflows, re-order the bytes, and pass the 5-byte message to the

application. Meanwhile, neither IDS 1 nor IDS 2 will get a match even if they are MPTCP-aware as they only observed 3 and 2 bytes of the signature respectively. This possibility means that the network security devices might have to make decisions based on partial traffic in the presence of MPTCP unless both ISPs share their observed data with each other.

2.3.2 Independence from a Fixed Four-tuple

A TCP connection always uses a fixed four-tuple of addresses and ports on sender and receiver during its life-time. If any of these four parameters has to change, the connection has to be re-established. In contrast, an MPTCP connection can survive changes in network addresses. New subflows can be added or old can be removed while an MPTCP connection is still active. This makes it problematic for security devices to keep track of connections. Furthermore, the security devices can not rely on four-tuples anymore as that will lead them to consider each subflow of an MPTCP connection as an independent TCP connection and thus not be able to properly reassemble the data from those subflows.

2.3.3 Durable and Reverse Connections

MPTCP can lead to incorrect functioning of at least two more security techniques based on TCP. First, it is common for security devices to close down malicious TCP connections if required. Previously, this has been as easy as inserting the TCP RST packet with the correct sequence number in the communication. However, MPTCP brings additional levels of resilience. An MPTCP connection will not terminate until all subflows (some of which might be passive for a backup) are closed.

In addition, the security devices that assume that the sender of a SYN in a handshake is always the client, and make decisions based on the direction of traffic might fail in their operation. MPTCP allows a server to open reverse connections back to a client in the event that a new network interface becomes available.

3 Research Objectives and Questions

The potential security impacts of MPTCP need to be thoroughly investigated and addressed in order to enable wide-scale deployment of the protocol. This work takes one step towards achieving that. To reach our goal, we have devised the following two research objectives:

- Increase the awareness of potential MPTCP security issues in the community by investigating the feasibility of attacks on existing network security infrastructure, i.e., IDSs.
- Propose, implement, and make publicly available proposed solutions to address the investigated MPTCP security issues.

The thesis deals with the following research questions to achieve its objectives.

1. *Are MPTCP cross-path data fragmentation attacks possible?*

   The first step is to verify the problem. MPTCP security implications make sense in theory, but this is not enough. We need to know how practical it is to exploit them and launch attacks that can degrade the security of a network. In particular, we try to answer whether it is possible to exploit the way MPTCP allows a sender to fragment the data stream to initiate cross-path data fragmentation attacks that can evade the existing security devices since they expect the data stream from a connection to come along the same path. We undertake this task by using attack traffic that is representative of actual real world malware and attacks.

2. *How can we detect such attacks if the devices can observe all traffic?*

   Once the problem is verified to exist, we undertake the task of proposing different heuristics to solve it. The first problem relates to incorrect understanding of the MPTCP protocol by current network security devices. This leads them to not being able to correctly reassemble the MPTCP network traffic even when they can observe all of it. Moreover, it is valuable to not just propose solutions, but implement them to evaluate their effectiveness and verify the proposed ideas. This is precisely how we answer this question.

3. *How can interim MPTCP aware security solutions be developed?*

   The MPTCP protocol is as of yet not common on the Internet. The protocol needs interim solutions to expedite its deployment in the transitional stage as the network infrastructure slowly starts to support MPTCP. We explore such interim solutions with the idea to extend MPTCP performance advantages to all hosts in a setting where network security performs as it does with TCP. An improved performance using MPTCP coupled with security assurances can go a long way to enable the adoption of the protocol.

4. *How can we design security solutions for a fully MPTCP world?*

   Apart from the security implications that are only valid in the interim stage, MPTCP has other long-term implications that need to be investigated and addressed accordingly. One of these issues is the possibility of network security devices having to make decisions while observing only parts or fragments of the traffic. This problem will exist in an MPTCP world and requires re-thinking of how network security is approached today.

4 Contributions

While answering the research questions outlined in Section 3, this thesis makes the following contributions.

1. *Insight into how an IDS can be evaded using MPTCP and its mitigation*

   The first contribution of this work is an insight into how a traditional security device such as an IDS reacts under cross-path data fragmentation attacks using MPTCP. In particular, the Snort IDS [21] was selected as the IDS as it is open-source and deployed at a wide scale. We used Snort rules to generate realistic attack traffic and then fragmented it across up to five paths to investigate how the Snort IDS reacts. The evaluation confirmed the initial worry as the attacks were successfully able to transfer attack signatures while evading the IDS. We also propose and implement a possible solution to mitigate the problem when an IDS can observe traffic from all paths. The solution assumes that the evasion happens because of not understanding MPTCP protocol semantics and treating an MPTCP connection as TCP.

2. *An MPTCP proxy that can extend secure performance advantages*

   Interim solutions such as a proxy can help accelerate the deployment of a protocol. Another contribution of this work is an MPTCP proxy that demonstrates how MPTCP performance advantages can be extended in a secure way to TCP servers. The proxy hosts two MPTCP aware security services. The first service enables correct operation of denial of service (DoS) rules while the second performs detection of intrusions in an MPTCP setting. Both services utilize an MPTCP adapted connection tracker that mimics the Linux connection tracker. We evaluate the performance of security services and validate the functionality of the proxy to show its correctness.

3. *An algorithmic solution to detect signatures from fragments*

   To help evolve network security, we propose a methodology that employs a novel variation to the well known Levenshtein distance [15] to detect attack signatures from signature fragments. We believe that this is the approach that will have to be undertaken by the security devices when protocols such as MPTCP become widely deployed. In such a setting, the basic expectation of traditional security devices to observe traffic from all connections can not be guaranteed anymore.

4. *A tool that translate IDS rules into attack traffic*

   In our quest to address the research questions of this work, we discovered the need to develop a tool that can generate synthetic attack traffic. This tool is an indirect contribution of this thesis. It can translate the majority of the well known open-source Snort IDS ruleset into corresponding traffic. The tool is publicly available for the benefit of the research community.

5 Research Methodology

Computer science [11] is the science of computers. It is typically summarized into two main categories: Theoretical and Practical. Theoretical computer science, as the name hints, is abstract and mathematical in nature. On the other hand, practical computer science is concerned with applied areas of computer science. This thesis is concerned with practical computer science.

The well known cycle of *scientific method* is utilized to conduct the research work. This method is generally divided into four steps. To start with, an observation is made. From the observation, a concrete question and a hypothesis is formulated. To verify the hypothesis, experiments are conducted and conclusions are drawn. If the conclusions show that the hypothesis was incorrect, then a new hypothesis is formulated and the cycle goes on. For example, in Paper I, a hypothesis was formed based on the observation that MPTCP can divide data across paths. Experiments were designed and conducted to verify the hypothesis, and conclusions were drawn.

However, a major part of practical computer science involves development of new technologies and tools. This process involves a conceptually similar method with slightly different names of steps. The method starts with an idea which is utilized to design a system or tool in theory. Next, the system is practically implemented and evaluated [9, 11]. This engineering inspired method is used in all papers, where we apply existing scientific knowledge (gained by us and others) to propose novel systems. In Paper I, we propose and implement a novel heuristic based solution to solve one aspect of the cross-path data fragmentation issue. In Paper II, we propose and implement an MPTCP proxy solution. In Paper III, we draw inspiration from mathematical and information theory fields, to propose and implement a modification to a well known algorithm. This enabled us to propose novel ways of attack detection. Finally, in Paper IV, a novel tool to translate IDS rules into corresponding traffic was proposed, implemented, and presented to fill the gap in the literature where such a tool was missing.

During the course of this work, we have designed experiments to verify our hypotheses. We implemented software programs to perform the verification in all papers. In a networking context, simulations, emulations, and real world measurement can be used as methods to collect data to verify a hypothesis. In Paper II, we employ a method based on emulations to measure the effectiveness of the proposed solution in a networking context. We believe this method gives the best balance between feasibility and applicability.

The reproducibility of research is important to allow others to repeat the work. This can either result in verification and extension of results obtained or identification of mistakes. The research in this thesis is conducted with this in mind, and the source code of tools and solutions from all papers are freely available.

6 Summary of Appended Papers

Paper I – Multipath TCP IDS Evasion and Mitigation

In this paper, we examine the security risk that can arise if MPTCP is used over the existing Internet infrastructure consisting of security middleboxes. In particular, the practicality and severity of cross-path fragmentation attacks utilizing MPTCP against the signature-matching capability of the Snort IDS is investigated. Results reveal that the attack is realistic and opens the possibility to evade any signature-based IDS. To mitigate the attack, a heuristic-based solution is also proposed in the form of the *MPTCP Linker* tool. The paper outlines the importance of MPTCP support in future network security middleboxes.

Paper II – Towards Multipath TCP Aware Security Technologies

In this paper, we propose and implement two MPTCP aware security services and deploy them inside a proof of concept MPTCP proxy. The aim is to enable hosts, even those without native MPTCP support, to securely benefit from the MPTCP performance advantages. The evaluation shows that the security services that are implemented enable proper intrusion detection and prevention to thwart potential attacks as well as threshold rules to prevent DoS attacks.

Paper III – Partial Signature Matching in an MPTCP World using Insert-only Levenshtein Distance

In this paper, we propose a methodology consisting of a constrained version of the Levenshtein distance that can be used to detect signatures from partial traffic. This is particularly useful in an MPTCP scenario where a single IDS can observe only parts of the connection traffic. The proposed algorithm is formally presented, implemented, and tested using the latest available version of the Snort ruleset. The results show that the algorithm can successfully detect all partial signatures with nearly 90% accuracy.

Paper IV – IDS Rule Management Made Easy

In this paper, we propose a tool to help optimize IDS rulesets and make IDS rule management easier. Due to an ever changing threat landscape, the rulesets used by these IDSs have grown large. Such broad and non-optimized rulesets lead to false positives and an unnecessary burden on the IDS, resulting in possible degradation of the security. The presented tool can convert IDS rules into corresponding traffic that can be used to evaluate and optimize the ruleset. The same tool is used in Paper I to generate attack traffic to test cross-path data fragmentation attacks.

7 Concluding Remarks and Outlook

MPTCP has the potential to eventually succeed TCP by overcoming some of its limitations. However, the MPTCP protocol needs time and promotion before its adoption becomes universal. Both, performance and security provided by a protocol play a significant role in motivating its use. The performance advantages of MPTCP are increasingly investigated and communicated by researchers [8, 20]. The security of the protocol itself, in isolation to its environment, is also under the microscope [3, 14]. However, the unexpected network security implications of the protocol on existing security devices have not yet been thoroughly investigated. This thesis takes a step towards that. The first objective of this work is to increase the awareness of potential risks that can be raised by using MPTCP over unaware security infrastructure. To do that, the work explores one potential security implication of MPTCP and examines its feasibility. Using a realistic methodology, it is shown that the risk for IDS evasion is possible and the current network security infrastructures are vulnerable to similar attacks.

The second objective of this work is to propose and implement possible solutions to the investigated problems. Different solutions are proposed, for the current transitional stage where MPTCP is not widely adopted, and for a fully MPTCP world, to mitigate and overcome the investigated issues. For the current period, where most devices on the Internet do not recognize MPTCP, we propose a way to perform intrusion detection when an IDS device can observe all traffic, but can not understand its semantics. An MPTCP proxy is also proposed to enable the use of the protocol even when the server is not MPTCP capable. The security services on the proxy ensure that security is not affected. In the long term, if and when MPTCP becomes widely deployed, just understanding the protocol semantics is not enough for security devices to perform their function correctly. As shown in this work, network security will have to evolve. In our work, we propose a way to detect intrusions from partial signatures. We assume that a security device only observes some parts of the traffic from a connection and has to make decisions based on that. We show our methodology to be promising, although not perfectly accurate yet.

The significance of the issues raised in this thesis is high and calls for a number of possible future research directions. All in all, the work described here touches just one, i.e., *cross-path data fragmentation*, of many identified potential security implications of MPTCP. Once all aspects of this MPTCP implication have been further properly addressed, we aim to turn our attention to other possibilities in which MPTCP can affect network security, as discussed in Section 2. For instance, we aim to investigate how the possibility of reverse and durable connections made possible by MPTCP can affect network security and propose solutions if required. Ultimately, for MPTCP to be widely adopted, its use has to be secure for all the stakeholders including the network operators and end-hosts. It will not be feasible to promote the deployment of the protocol at a wide scale if it can degrade security.

References

[1] Internet live stats. http://internetlivestats.com. Accessed: 2017-03-14.

[2] G. Armitage, N. Williams, et al. FreeBSD kernel patch to enable Multipath TCP. https://bitbucket.org/nw-swin/caia-mptcp-freebsd. Accessed: 2017-03-14.

[3] M. Bagnulo, C. Paasch, F. Gont, O. Bonaventure, and C. Raiciu. Analysis of residual threats and possible fixes for Multipath TCP (MPTCP). RFC 7430, RFC Editor, July 2015. https://tools.ietf.org/html/rfc7430.

[4] O. Bonaventure. Apple seems to also believe in Multipath TCP. http://perso.uclouvain.be/olivier.bonaventure/blog/html/2013/09/18/mptcp.html. Accessed: 2017-03-14.

[5] O. Bonaventure. Multipath TCP is pronounced giga path in Korea. http://blog.multipath-tcp.org/blog/html/2015/07/24/korea.html. Accessed: 2017-03-14.

[6] O. Bonaventure, C. Paasch, and G. Detal. Use cases and operational experience with Multipath TCP. RFC 8041, RFC Editor, January 2017. https://tools.ietf.org/html/rfc8041.

[7] V. Cerf and R. Kahn. A protocol for packet network intercommunication. *Communications, IEEE Transactions on*, 22(5):637–648, May 1974.

[8] Y. Chen, Y. Lim, R. J. Gibbens, E. M. Nahum, R. Khalili, and D. Towsley. A measurement-based study of MultiPath TCP performance over wireless networks. In *Proceedings of the 2013 Internet Measurement Conference, IMC 2013, Barcelona, Spain, October 23-25, 2013*, pages 455–468, 2013.

[9] P. J. Denning. Performance evaluation: Experimental computer science at its best. In *Proceedings of the 1981 ACM SIGMETRICS conference on Measurement and modeling of computer systems, Las Vegas, Nevada, USA., September 14-16, 1981*, pages 106–109, 1981.

[10] G. Detal. MPTCP-enabled kernel for the Nexus 5. https://github.com/gdetal/mptcp_nexus5. Accessed: 2017-03-14.

[11] D. G. Feitelson. Experimental computer science: The need for a cultural change. *Internet version: hsttp://www.cs.huji.ac.il/~ feit/papers/exp05.pdf*, 2006.

[12] A. Ford, C. Raiciu, M. Handley, O. Bonaventure, and C. Paasch. TCP extensions for multipath operation with multiple addresses. RFC 6824, RFC Editor, 2016. https://tools.ietf.org/html/draft-ietf-mptcp-rfc6824bis-06.

[13] C. Huitema. Multi-homed TCP. Internet Draft draft-huitema-multi-homed-01, IETF, May 1995.

[14] M. Jadin, G. Tihon, O. Pereira, and O. Bonaventure. Securing Multi-Path TCP: Design & Implementation. In *Proceedings of the IEEE International Conference on Computer Communications, INFOCOM, Atlanta, USA*, 2017.

[15] V. I. Levenshtein. Binary codes capable of correcting deletions, insertions and reversals. *Soviet Physics Doklady*, 10(8):707–710, 1966.

[16] P. I. Limited. The global state of information security survey. `http://www.pwc.com/sg/en/publications/assets/pwc-global-state-of-information-security-survey-2016.pdf`. Accessed: 2017-03-14.

[17] C. Paasch, S. Barré, et al. Multipath TCP in the Linux kernel. Available from http://www.multipath-tcp.org, 2017.

[18] C. Pearce. MPTCP roams free. `http://labs.neohapsis.com/2014/10/20/mptcp-roams-free-by-default-os-x-yosemite/`. Accessed: 2017-03-14.

[19] C. Pearce and S. Zeadally. Ancillary impacts of Multipath TCP on current and future network security. *IEEE Internet Computing*, 19(5):58–65, 2015.

[20] C. Raiciu, S. Barre, C. Pluntke, A. Greenhalgh, D. Wischik, and M. Handley. Improving datacenter performance and robustness with Multipath TCP. In *Proceedings of the ACM SIGCOMM 2011 Conference*, SIGCOMM '11, pages 266–277. ACM, 2011.

[21] M. Roesch. Snort: Lightweight intrusion detection for networks. In *Proceedings of the 13th Conference on Systems Administration (LISA), Seattle, WA, November 7-12*, pages 229–238, 1999.

[22] R. Stewart. Stream control transmission protocol. RFC 4960, RFC Editor, September 2007. `http://www.rfc-editor.org/rfc/rfc4960.txt`.

Towards Secure Multipath TCP Communication

Multipath TCP (MPTCP) is an extension to standard TCP that is close to being standardized. The design of the protocol is progressing, but most of the focus has so far been on its compatibility. The security aspect is confined to making sure that the MPTCP protocol itself offers the same security level as standard TCP. The topic of this thesis is to investigate the unexpected security implications raised by using MPTCP in a traditional networking environment. Today, the security middleboxes make use of different assumptions that may not be true anymore with the advent of MPTCP.

We investigate how practical it is to evade a security middlebox by fragmenting and sending traffic across multiple paths using MPTCP. Realistic attack traffic generated from a tool that is also presented in this thesis is used to show that these attacks are feasible. We then go on to propose possible solutions to detect such attacks and implement them in an MPTCP proxy. The proxy aims to extend secure MPTCP performance advantages. We also investigate the MPTCP scenario where security middleboxes can only observe some of the traffic. We propose and implement an algorithm to perform intrusion detection in such situations and achieve a high detection accuracy.

ISBN 978-91-7063-763-6 (print)

ISBN 978-91-7063-764-3 (pdf)

ISSN 1403-8099

LICENTIATE THESIS | Karlstad University Studies | 2017:12