Life of a Security Middlebox

Challenges with Emerging Protocols and Technologies

Zeeshan Afzal

Life of a Security Middlebox

Challenges with Emerging Protocols and Technologies

Zeeshan Afzal

Faculty of Health, Science and Technology

Computer Science

DOCTORAL THESIS | Karlstad University Studies | 2020:10

Life of a Security Middlebox

Challenges with Emerging Protocols and Technologies

Zeeshan Afzal

DOCTORAL THESIS | Karlstad University Studies | 2020:10

Life of a Security Middlebox - Challenges with Emerging Protocols and Technologies

Zeeshan Afzal

DOCTORAL THESIS

Karlstad University Studies | 2020:10

urn:nbn:se:kau:diva-76291

ISSN 1403-8099

ISBN 978-91-7867-093-2 (print)

ISBN 978-91-7867-103-8 (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 2020

WWW.KAU.SE

Life of a Security Middlebox: Challenges with Emerging Protocols and Technologies

Zeeshan Afzal
Department of Mathematics and Computer Science
Karlstad University

Abstract

The Internet of today has intermediary devices known as middleboxes that perform more functions than the normal packet forwarding function of a router. Security middleboxes are a subset of these middleboxes and face an increasingly difficult task to perform their functions correctly in the wake of emerging protocols and technologies on the Internet. Security middleboxes make many assumptions about the traffic, e.g., they assume that traffic from a single connection always arrives over the same path and they often expect to observe plaintext data. These along with many other assumptions may not hold true any longer with the advent of new protocols such as MPTCP and technologies like end-to-end encryption.

The work in this thesis focuses on security middleboxes and the challenges they face in performing their functions in an evolving Internet where new networking protocols and technologies are regularly introduced. We develop methods and solutions to help these security middleboxes continue to function correctly. In particular, we investigate the case of using MPTCP over traditional security infrastructure as well as the case of end-to-end encryption.

We study how practical it is to evade a security middlebox by fragmenting and sending traffic across multiple paths using MPTCP. Attack traffic that is generated from a self-developed tool is used to evaluate such attacks to show that these attacks are feasible. We then go on to propose possible solutions to detect such attacks and implement them. The potential MPTCP scenario where security middleboxes only have access to part of the traffic is also investigated. Furthermore, we propose and implement an algorithm to perform intrusion detection in such situations. Moreover, the thesis contributes a machine learning based approach to help security middleboxes detect malware in encrypted traffic without decryption.

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

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 pursue 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. I offer my gratitude to my wife Khadija for her support and our daughter Dua for all the happiness she brings to our lives. Thank you to my family in Norway for providing unwavering support and making it easier to stay away from home. 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 let's make many more. You will never walk alone!

Finally, I would like to thank all my colleagues at the department for our 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, December 11, 2019 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, Stefan Lindskog, and Anna Brunstrom. Slice Distance: An Insert-Only Levenshtein Distance with a Focus on Security Applications. In Proceedings of the 9th IFIP International Conference on New Technologies Mobility and Security (NTMS), Paris, France, February 26–28, 2018.

IV. **Zeeshan Afzal**, Johan Garcia, Stefan Lindskog, and Anna Brunstrom. Using Partial Signatures in Intrusion Detection for Multipath TCP. In Proceedings of the 24th Nordic Conference on Secure IT Systems (NordSec), Aalborg, Denmark, November 18–20, 2019.

V. **Zeeshan Afzal**, Anna Brunstrom, Stefan Lindskog, and Johan Garcia. Using Features of Encrypted Network Traffic to Detect Malware. Under Submission.

VI. **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 editorial changes.

Comments on my Participation

For all the papers, the ideas were developed together with my co-authors. Throughout the work process, Stefan Lindskog provided guidance from a security point of view while Anna Brunstrom provided valuable networking insights. However, I did most of the implementation, evaluation, and writing work. Comments about the contributions of my other co-authors are given below:

- In Paper II, Anders Lidén proposed the different MPTCP proxy scenarios and suggested where the proxy should be placed.
- In Paper III, Johan Garcia and I jointly came up with the idea of the slice distance algorithm.

- In Paper IV, Johan Garcia authored the subsection describing the MPTCP-aware multi-pattern approximate string matching algorithm.
- In Paper V, Johan Garcia provided guidance on the design of the machine learning experiments.

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.
- **Zeeshan Afzal.** Towards Secure Multipath TCP Communication. Licentiate thesis, Karlstad University, Karlstad, Sweden, 2017.

Contents

Introductory Summary

1 Introduction

We live in an age where kids do their homework using virtual assistants such as Alexa. Laundry machines are connected to smartphones with an application. We consult with a doctor over live video around the clock. This wave of digital revolution has completely engulfed us and has made our lives more convenient and complex at the same time. At the core of all this revolution lies the Internet. Initially designed to enable a limited group of people to communicate, the Internet has grown substantially to a mass medium of more than four billion users. The proliferation of smartphones together with cheap data plans have further contributed towards this trend.

Over the years, as the Internet has taken a central role in our lives, its design has moved away from the original end-to-end principle [51]. At its inception, the goal was to build new application functions at the endpoints and not into the core network. However, the Internet with its best-effort delivery of data and no time guarantee, was not ready to meet the performance demands of an increasing number of applications such as live streaming that necessitate timely delivery of data. Additionally, the classic Internet was developed with an assumption that endpoints are trustworthy, an assumption that is no longer true. Thus the design principle of the Internet was not ideal to face the increasing challenges of the digital age. Consequently, these reasons led to the introduction of devices known as "middleboxes" in the middle of the communication paths. Some of these devices provide functionality to overcome limitations in the original Internet protocol suite, e.g., Network Address Translation (NAT) boxes solve the address depletion problem, while others offer a range of additional functions. To solve security related issues, security middleboxes such as firewalls and Intrusion Detection Systems (IDSs) were introduced. This thesis focuses on the life of these so called security middleboxes and the challenges they face in performing their function in an evolving Internet where new networking protocols and technologies are regularly introduced.

TCP [15] is the most commonly used transport protocol on the Internet to deliver end-to-end services. However, under the increased application demands as mentioned above, TCP has fallen short for a number of reasons. One such reason is the dependency of a TCP connection on the same pair of IP addresses and port numbers throughout the life of a connection. Different proposals have been suggested to overcome such shortcomings [31, 53]. Multipath TCP (MPTCP) [26] is one such important proposal that was specified by IETF as an experimental standard in early 2013. MPTCP is on its way to become a standard that will 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 [4, 10, 11, 21, 44] and researchers have already shown how MPTCP can outperform TCP in a number

of situations [16, 47].

In the wake of new evolutionary networking protocols such as MPTCP, the security middleboxes face an increasingly challenging task. Deep packet inspection (DPI) is one of the technologies employed by these middleboxes to ensure the security of the network. 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 security middleboxes performing DPI can not recognize and thus analyze MPTCP traffic. Many of the basic assumptions about the traffic made by middleboxes are no longer true with MPTCP [45]. For instance, middleboxes that track connections and classify traffic based only on their five-tuple will see the subflows of an MPTCP connection as independent TCP connections with no correlation. Thus, they can not reassemble MPTCP traffic correctly. MPTCP also allows a sender to use all available subflows simultaneously. This enables the fragmentation of data among the subflows in a way such that there is not enough information on any of the subflows for a security device to recognize whether the data being sent are malicious.

This thesis takes a deep look into the auxiliary security impacts of MPTCP and the resulting attacks caused by the non-conformance of traffic to basic assumptions. The work makes an effort to investigate the feasibility of such attacks and then proposes solutions to defend against them. Specifically, we investigate the attacks made possible by fragmenting a data-stream among multiple active subflows and their impact on security middleboxes. We show that *cross-path data fragmentation* attacks are practical (Paper I). Furthermore, we differentiate between two scenarios where the simple case is when a middlebox 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 (Paper II) to ease the protocol deployment during the transition stage and extend secure benefits to more hosts. The other scenario is when the MPTCP operation causes a middlebox to observe only parts of the traffic from connections and it has to make decisions based on that alone. We investigate this problem in detail and propose a new metric (Paper III) to help solve the problem. As a next step, the metric is implemented in a middlebox to perform signature-matching when only parts of the traffic are observed (Paper IV).

Apart from emerging protocols, security middleboxes also have to deal with other novel technologies on the Internet. One such landmark that has challenged the way they operate is end-to-end encryption. The increasing need to secure our private data from any third party has resulted in encryption becoming ubiquitous. While this appears to improve communication security and enhance privacy, at the same time and not surprisingly, security reports [17] show that bad actors are also increasingly leveraging the same benefits to their advantage in order to avoid detection. Already in 2019, over half of the malware attacks use encryption in some way. It is expected that even more malware (more than 70%) will be using encryption by 2020 and

the majority of organizations will have no possibility to detect it [17]. This trend is not a good sign and calls for a need to investigate novel encrypted malware detection methods.

The response to encryption from the operators of security middleboxes was to develop tools to decrypt the traffic using end-point collaboration. This allows them to regain visibility and perform security functionality as usual. The approach has obvious security and ethical drawbacks, not to mention that it will be harder if not impossible to do as we approach newer versions of encryption protocols. From the perspective of a middlebox, if there is no application data in the plaintext to consider, all the functionality that it provides based on traffic content will be absent. The only thing the middlebox can make use of is the addressing and metadata information from the traffic. Thus it is important for the future of these middleboxes to find ways to detect malicious attempts without relying on decrypting the traffic. Instead of allowing security middleboxes to turn a blind eye towards threats using encryption, as part of this thesis new novel techniques based on machine learning are proposed that enable a security middlebox to detect malware using metadata information and without any decryption (Paper V).

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 the work. Section 5 relates the research and the methods used in this thesis 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 discussed in this thesis. The first part of the section provides an overview of the history of the Internet and the need for security middleboxes. Some background on the MPTCP protocol is then detailed followed by a subsection that provides a backdrop to end-to-end encryption. The last part of the section provides a synthesis and a discussion on how to move forward.

2.1 The Internet

The Internet is an experiment that escaped the lab [36]. After the success of ARPANET (a resource-sharing network), the National Science Foundation (NSF) in the US decided to create a communication network (NSFNET) that will serve as a medium to allow scientists and researchers to share data amongst each other. Little did NSF or anyone else know that the network they built would lead to the Internet of today. When it came to the task of selecting a protocol to inter-connect different heterogeneous networks, NSF adopted the TCP/IP protocol suite [15] that was used in ARPANET and developed by Kahn and Cerf. The core design principles of the Internet have

remained unchanged from ARPANET and NSFNET. One of the most important principles that guided the design of the Internet was the end-to-end principle [51], which suggests that a new application-level function normally can not and preferably should not be built in the network. This is because a function can only be correctly implemented with the knowledge and help of the application in the end-host.

Today, the reality is that the Internet has moved a bit away from the pure end-to-end principle. With the rapid expansion of Internet users with more than half of the world's population using it [1], the Internet is not just expected to provide general data transfer. The Internet of today is facing new application requirements that are almost impossible to meet by relying solely on end-hosts and without moving away from the end-to-end principle. One example is real-time streaming applications. The Internet was designed with a "best-effort" data transfer strategy with no throughput assurances. While this was and still is acceptable for some applications such as web-browsing or email (which itself relies on intermediate relays in the network), it is difficult to see how the classic Internet data transfer can meet the demands of these new applications [9]. Nowadays, these types of requirements are usually met through a multi-stage delivery by placing the content on an intermediate server closer to the end-hosts.

Another major concern with the end-to-end principle of the Internet that is more relevant for this thesis is the security aspect. Security of the network was not even a consideration when the Internet was being built, mainly, due to the network being so small and the expectation that every end-host knows everybody else and thus trust was inherent. Today, things are different and there is no reason to assume or trust an end-point to behave as expected. A malicious end-point can launch attacks on other end-hosts or even attack the network itself. This situation is not a surprise considering such a big influx in the number of Internet users coupled with the increased motivation to cause malice. After all, the Internet is today used to perform many sensitive tasks other than simple web-browsing and email. Therefore, it is imperative to detect malware and attacks. Every day, thousands of security breaches happen across the world causing damage to reputation and costing millions of dollars [46]. The number of security incidents per day is on the rise, and so is the cost incurred by each incident and the security budget dedicated by organizations.

2.2 Security Middleboxes

As discussed above, security was not really considered when the Internet was built. The TCP/IP protocol suite that has become the de facto standard for the Internet was also not developed with security in mind. This led to many security innovations over the years, most of which were deployed as part of security middleboxes in the communication paths. The following are some examples of security middleboxes and their function.

- **Firewalls** Network-based firewalls are often placed at the edge of a net-

work to inspect passing traffic using its addressing information and either forward it or drop it based on a pre-defined rule-set [43]. This is useful in protecting a network or some parts of it from an untrusted external network.

- **Intrusion Detection and Prevention Systems (IDPSs)** IDPSs provide another layer of protection by inspecting the payload of packets. They analyze the traffic and try to detect and possibly prevent attack or malware instances based on either identifying abnormal behavior and/or pre-defined attack patterns [7]. The systems 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 IDPS detects an attack instance, it generates an alert and either logs the intrusion attempt or takes some preventive action as configured by the administrator.

- **Fraud Monitoring Systems (FDSs)** Another important security function especially for the financial industry is to monitor the network for suspicious activities such as identity theft [35, 40]. Through the use of special algorithms and databases, FDSs protect businesses from potentially severe problems.

2.3 The Challenges of Multipath Protocols

The security middleboxes on the Internet face a constant challenge in performing their correct function as new protocols emerge on the Internet. Over the years these middleboxes have been developed to work with TCP as it is the most commonly used transport layer protocol to reliably deliver in-order data from one application to another. However, networking and applications are evolving. Modern applications require more than just reliable in-order delivery because the networking between end-hosts allows for multiple paths. There is a need to combine connections for a higher bandwidth. Some applications require additional resilience where end-hosts are always expected to stay connected. TCP has no means to support these use-cases. This was identified as early as 1995 [31]. Different protocols such as SCTP [53], MPTCP [26], and QUIC [32] have been suggested to resolve some of the problems. SCTP was originally a signaling protocol that evolved into a general-purpose transport protocol to offer a similar function as TCP, but with the added possibility of multi-homing (that brought its own risks [6]). QUIC was originally introduced by Google and aims to enhance performance of web traffic while introducing encryption by-default. This thesis focuses on the challenges faced by security middleboxes due to the MPTCP protocol [26].

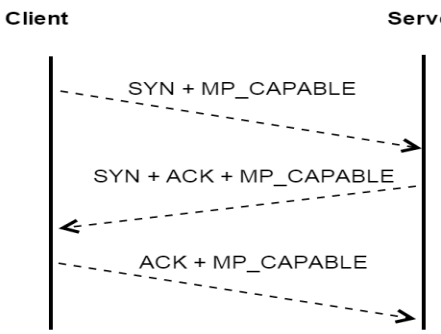

Figure 1: MP_CAPABLE handshake.

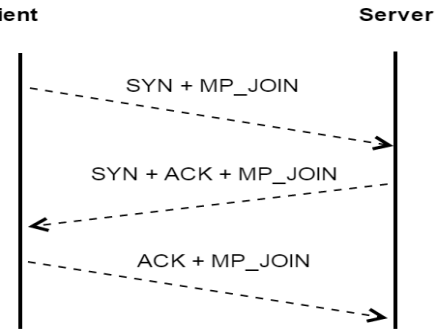

Figure 2: MP_JOIN handshake.

2.3.1 Multipath TCP

MPTCP [26] is an experimental extension to standard TCP that is soon to become a proposed standard. 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 with TCP. 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 (referred to as subflows in MPTCP). Each of these subflows is a proper TCP connection, but with additional TCP options that allow every subflow to be linked to an MPTCP connection. Some key concepts of MPTCP that are most relevant for this thesis are discussed in the subsequent text. See [26] for further details.

MPTCP connections and subflows An MPTCP connection consists of a number of TCP flows that are linked together. The establishment of a connection takes place 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 can 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.

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 uses all available paths (subflows) simultaneously [26] 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 re-order the data received over different subflows before passing them on to the application layer in the correct order.

Deployment MPTCP was designed with multi-homed devices such as smartphones in mind [19]. It has already found use-cases in many diverse areas. One such use-case is in 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 [13, 47]. Additionally, research has shown MPTCP's effectiveness in reducing download times and latencies for mobile users [13, 16]. Korean Telecom has utilized MPTCP to enable users to reach bandwidth of up to 1 Gbps [12]. There exist many implementations of MPTCP on a number of operating systems. It is available for Linux [44], BSD [4] and Android [21]. Commercially, Apple implemented MPTCP starting with iOS7 [10] for Siri and later allowed any application to use MPTCP from iOS11 [11].

2.3.2 Security Implications of MPTCP

The security infrastructure such as the security middleboxes on the Internet are used to analyze TCP. Therefore, it is natural to consider TCP as a reference point when discussing the security implications of MPTCP. 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 is a protocol security assessment of MPTCP [8] 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 extensively explored.

Indeed, MPTCP can be substantially different from TCP from a network security point of view. A study conducted by Pearce and Zeadally [45] outlined the main network security implications of MPTCP and the key security differences between TCP and MPTCP. They suggested at least four differ-

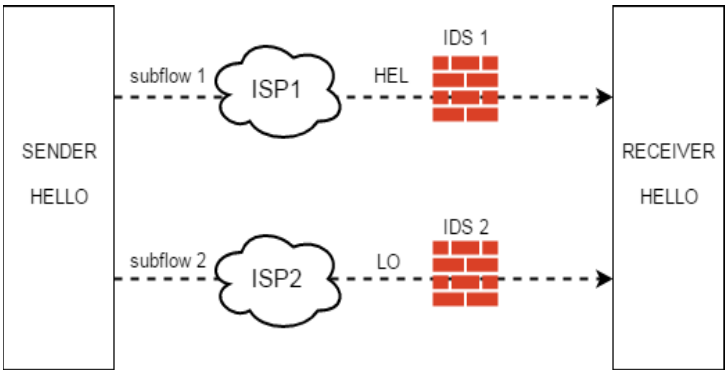

Figure 3: Data transfer using MPTCP [45].

ent security impacts of MPTCP on current network security. These impacts 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 of of Pearce and Zeadally. It should be noted that the reference to MPTCP in the security context assumes an MPTCP connection with at least two subflows.

Data fragmentation As discussed in Section 2.3.1, 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 by the sender. In such an instance, assuming that the security middleboxes 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 middleboxes 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 Internet Service Providers (ISPs). The middleboxes on each path of these subflows can no longer observe all traffic from a given connection. Figure 3 depicts such a scenario recreated from [45]. 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 remain-

ing 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 security middleboxes might have to make decisions based on partial traffic in the presence of MPTCP unless both ISPs share their observed data with each other.

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 lifetime. 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 and old can be removed while an MPTCP connection is still active. This makes it problematic for the middleboxes to keep track of connections. Furthermore, the middleboxes can not rely on the four-tuples any more 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.

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 middleboxes 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 as a backup) are closed.

In addition, the middleboxes 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 so-called reverse connections back to a client in the event that a new network interface becomes available.

2.4 The Challenges of End-to-end Encryption

New technologies are constantly integrated on the Internet. End-to-end encryption is one such technology that encodes messages to prevent eavesdropping of communications by any third party. End-to-end encryption can be implemented at different layers in the network. IP Security (IPSec) [5] encrypts application data transparently at the IP layer. Other protocols such as Transport Layer Security (TLS) [48] and Secure Shell (SSH) [39] operate at the transport layer and the application layer, respectively. In general, TLS is the preferred mechanism to achieve end-to-end encryption on the Internet. TLS is implemented on top of protocols such as HTTP for web browsing or SMTP for email. Initially defined in 1999 as an upgrade to its predecessor Secure Socket Layer (SSL), the TLS specification has been improved over the last two decades to its current version 1.3 [48] that was defined in 2018.

TLS 1.3 made a number of changes to make the protocol faster and more secure. This involves removing obsolete and insecure features. The aim of TLS is to provide confidentiality, integrity, and authenticity to any Internet-based communication. Moving forward, when we refer to encryption, we refer to end-to-end encryption protocols such as TLS.

Despite the obvious advantages, the public usage of encryption on the Internet was fairly limited up until 2013. The revelations of Edward Snowden about the surveillance capabilities of the National Security Agency (NSA) in the US changed the public perception and created an increased demand to encrypt communications. As a result, the computer industry worldwide started to take security and privacy seriously and took steps to better protect it. According to Google [27], the percentage of encrypted traffic across their services has increased from 50% in 2014 to over 94% in 2019. Today, more than 80% of enterprises' web traffic is encrypted [17]. These numbers show a significant trend towards ubiquitous end-to-end encryption on the Internet. However, it should be noted that encryption is just one important tool to improve the security of the Internet. Encryption also raises a few concerns for different stakeholders, as we discuss next.

2.4.1 Encryption and the Security Landscape

The properties of encryption that make communication security strong can also affect the functionality of some middleboxes in the Internet. At the network level, encrypting communications of end-hosts ensures that no third party can eavesdrop on the communication content directly, but at the cost of taking away the ability of network-wide monitoring to detect virus and malware or contain new vulnerabilities as well as a number of other functions. This is attractive for threat actors that are leveraging encryption to evade detection and to secure their malicious activities. With over half of all malware attacks carried out in 2019 using encryption and the expectation that more than 70% of malware will use encryption to hide its operation by 2020 [17], this is an important concern. Apart from the security aspect, taking away the visibility of traffic impacts other important day-to-day operational functions in enterprise networks and data centers [25,28]. These use-cases include troubleshooting, performance analysis, fraud monitoring, audit, regulatory compliance, and customer experience monitoring.

Network operators agree that strong encryption should be used to improve the privacy and security for all Internet sessions from client browsers to the edge of their networks [41]. Their concerns relate to the use of encryption within their networks. Enterprises encrypt data within a data center as a best practice to protect against inside threats and in some cases to comply with regulations. In doing so, the operators wish to maintain visibility of traffic within their network, where they control all the end-points to perform a number of important operational and security functions. In response to earlier versions of encryption protocols, the operators developed different approaches that perform decryption of traffic to give them the desired visibil-

ity. One approach known as passive or out-of-band decryption is based on a static private key used to decrypt captured encrypted traffic when necessary. Other solutions use an active man-in-the-middle (MitM) approach to analyze encrypted communications [3]. The traffic traversing a middlebox can be decrypted to gain the required visibility to perform security functionality. To achieve this, a middlebox generates a local root certificate that is installed on all internal devices for which the TLS traffic needs to be inspected. During a subsequent TLS handshake by the inspected device, the middlebox modifies the certificate provided by the server and signs it with its own private key from the root certificate, allowing it visibility into all further exchanges on that TLS session. For end-hosts, the process is completely transparent. Despite the ethical questions and the degraded end-to-end security [22] that can arise as a result of TLS inspection by decryption, it enables certain use-cases [3] and is utilized massively in enterprise networks. However, the latest version of TLS (v1.3) makes some of these approaches harder if not impossible. The operators stand on the brink of losing the visibility again and, in their opinion, this creates a number of short term problems for them that are not easily fixable [50]. They demand new protocols or extensions of existing encryption protocols [28,30] such that they are also suitable for their use-cases in addition to the main use-case of protecting the Internet-based sessions.

One might argue that the steps taken in TLS 1.3 to thwart TLS decryption are intended, as the use-cases satisfied by TLS decryption should instead be solely implemented at the end-points. Focusing only on the network security aspects, some argue that putting all the network security monitoring functionality in the end-points without affecting their performance is not feasible [28]. Enterprises also argue that end-points can not be solely trusted for security monitoring as a malware can delete logs or disable security monitoring. The increasing diversity of end-points, e.g., the IoT devices, also makes it a challenge to implement security monitoring only at the end-points [25]. Therefore, although some operators acknowledge that moving security monitoring services to end-points is a reasonable goal for the long-term, they believe that there is a short-term need of alternative solutions in the network to augment end-host security solutions that are still not adequate [50].

At a national level, law enforcement agencies believe that encryption makes their task much harder. The investigation methods that depend on lawful access to communication content through wiretapping are no longer useful, with strong encryption becoming ubiquitous. Two relatively recent examples where authorities complained about encryption coming in the way of their investigations are the 2016 Apple vs FBI case [34] and the 2017 shootings in the UK [52] where the encryption capabilities of WhatsApp were allegedly used by the terrorist involved. The consequence of these struggles of agencies has come in the form of them asking for a backdoor to encrypted communications. They ask for "exceptional access" [14, 18], which requires that a master key should be available for authorities to be able to decrypt communications in exceptional circumstances when national security is at risk.

2.5 The Way Forward

The merits of introducing middleboxes in the Internet can be debated, but it is a reality that communication paths on the Internet of today consist of a high number of middleboxes [29, 42]. Therefore, it is wise to consider these middleboxes as a part of today's Internet whether we like it or not. Only then can the new innovations in the Internet successfully take place. The future of the Internet will likely go in the same direction where the end-to-end model will continue to co-exist while other new functions will be introduced in the network. The focus should be on considering the end-to-end concept during the design of each new application and only deploying functions in the network when it is not possible to do it correctly on the end-points.

In a similar manner, it is important to understand that, although encryption brings new challenges for different stakeholders, it fulfills an essential need and is a net-gain for the security [23, 37] of our world today. From our personal affairs to business, most communication is conducted digitally. While this digitization has changed things for the good and brought amazing benefits to our society, it also creates extra possibilities for foul play and increased motivation to cause harm.

Network operators may have a need to gain visibility within the networks they own to perform a number of important functions. Once the traffic enters a data center, it is no longer in transit and they own it. The communications from client browsers to the edge of data centers are still secured with strong encryption and thus the clients are protected. However, relying on decryption or asking for extending TLS 1.3 such that it provides them visibility in the data center is not the solution [50]. If new functionality is added in the encryption protocols that allows access to unencrypted data by certain parties such as the operators, then there is no guarantee that this access may not be abused by bad actors [37]. Ultimately, more work must be done to find a way that allows operators to perform their day-to-day functions in the short-term, while end-point monitoring solutions are improved, which is yet another challenge. Similarly, law enforcement authorities need to adapt and find new ways to conduct their investigations instead of asking for impractical backdoors [2]. The digital age provides them with new forms of data, e.g., social media, location, and cloud data [37]. They need to make use of this new data and combine it with traditional investigation techniques to find alternate ways to do their job without encryption getting in their way.

In this thesis, a practical perspective towards middleboxes and encryption is considered. It is a fact that middleboxes are part of today's Internet, irrespective of their merit. Encryption is also becoming ubiquitous on the Internet and as such with TLS 1.3, decrypting TLS in the middle is not an option any more. Therefore, it is time to focus our attention on restoring the functionality of middleboxes that is lost due to the emerging protocols and technologies.

3 Objectives and Research Questions

The research in this thesis is focused on helping security middleboxes and making them as useful as possible in the wake of emerging protocols and technologies. The work focuses on MPTCP as its potential security impacts need to be thoroughly investigated and addressed in order for the security middleboxes to correctly perform their functionality. This will also go a long way to promoting wide-scale deployment of the protocol. Moreover, it is important to study the impact of end-to-end encryption on the security middleboxes and propose ways to restore their functionality moving forward. The thesis deals with the following main objectives:

- Explore the potential MPTCP security-related issues and propose, implement, and make publicly available solutions to address the identified issues.
- Investigate non-benign usage of encryption by a malware and propose novel methods to allow a security middlebox to detect such instances without decryption.

The thesis addresses the following three research questions to reach the above mentioned objectives.

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

   The first step is to verify the problem of *cross-path data fragmentation*. MPTCP security implications make sense in theory, but this is not enough. There is a need to know how practical it is to exploit them and launch attacks that can degrade the security of a network. In particular, an effort is made to find out whether it is possible to exploit the way MPTCP allows a sender to fragment the data stream to initiate *cross-path data fragmentation* attacks. Such attacks can evade the detection capabilities in security middleboxes since they expect the data stream from a connection to come along the same path.

2. *How can we design solutions to detect cross-path data fragmentation attacks?*

   Once the problem is verified to exist, we undertake the task of identifying different solutions to solve it. Moreover, it is valuable to not just devise solutions but also implement them to evaluate their effectiveness and verify the proposed ideas. We divide the problem into two scenarios and address them separately. The first scenario relates to incorrect interpretation of the MPTCP protocol by the security middleboxes. This results in them not being able to correctly reassemble the MPTCP network traffic even when they can observe all of it. Apart from the possibility of an incorrect understanding of the protocol, MPTCP can cause a scenario where the security middleboxes have to make decisions based only on some parts or fragments of the traffic. It is also important

to investigate solutions for this problem as it will exist in an MPTCP world and will require re-thinking of a number of traditional security solutions.

3. *How is encryption used by malware and can encrypted malware be detected without decryption?*

   This question deals with the second research objective. As more and more malware is using encryption to hide its operation, it is important to devise ways to detect it without decryption. We investigate how malware uses encryption to hide its operation and evade detection, and whether it is possible to detect it. This also involves studying how encryption is used for normal or benign purposes. The solution is ensured to be deployable and not affect encryption in any way.

4 Contributions

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

1. *Insights into how an IDS can be evaded using MPTCP*

   The first contribution of this work are insights into how a security middlebox such as an IDS reacts under *cross-path data fragmentation* attacks using MPTCP. In particular, the Snort IDS [49] was selected as the IDS to investigate as it is open-source and deployed on a wide scale. We used Snort rules to generate synthetic attack traffic and then fragmented it across up to five paths to investigate how the Snort IDS reacts. The evaluation confirmed the initial concern, as the attacks were successfully able to transfer attack signatures while evading the IDS.

2. *An MPTCP proxy that can mitigate cross-path data fragmentation attacks*

   This contribution relates to proposing and implementing a solution to mitigate *cross-path data fragmentation* attacks in the scenario when an IDS can observe MPTCP traffic from all paths but can not make sense of it. The solution assumes that evasion occurs because of the IDS not understanding the MPTCP protocol semantics and treating an MPTCP subflow as a TCP connection. Our solution is implemented in a proxy to demonstrate how MPTCP performance advantages can be extended in a secure way to TCP hosts. The proxy provides different MPTCP aware security services including detection of intrusions in an MPTCP setting. The services utilize an MPTCP adapted connection tracker that mimics the Linux connection tracker.

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

   To help security middleboxes such as IDSs in the scenario when they only see parts of the traffic, a novel variation to the well known Levenshtein distance [38] is proposed. This variation ensures that only insert

operations are considered in pattern matching as that is the only possibility in a *cross-path data fragmentation* attack. Moreover, this proposed metric is implemented in a methodology and integrated into the Snort3 IDS to evaluate how such an approach will work in practice. As protocols such as MPTCP become widely deployed, such approaches will be required since the basic expectation of a security middlebox to observe traffic from all connections can no longer be guaranteed.

4. *A machine learning approach to detect encrypted malware*

   Another contribution of this thesis is understanding the behavior of malware that uses encryption to hide its operation. Using large malware and benign traffic datasets, an investigation is conducted to learn how malware makes use of encryption. In the next step, properties of encrypted malware that are distinct to how encryption is used for benign purposes are identified and used in a machine learning approach. The proposed approach can detect the presence of malware within encrypted traffic with a high accuracy and without requiring any decryption.

5. *A tool that translates IDS rules into attack traffic*

   In the quest to address the research questions of this work, there was a 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 [24] is the science of computers. It is typically summarized into two main categories: theoretical and experimental. Theoretical computer science, as the name hints, is abstract and mathematical in nature. On the other hand, experimental computer science is concerned with applied areas of computer science. This thesis is concerned with experimental computer science.

The well known cycle of the *scientific method* is utilized to conduct the research work. This method is generally divided into four steps [24]. To start with, an observation is made. From the observation, a concrete question and a hypothesis are formulated. Experiments are conducted to verify the hypothesis, 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 which in turn can lead to *cross-path data fragmentation* attacks. Experiments were designed and conducted to verify the hypothesis, and conclusions were drawn. Paper V also follows the scientific method of cycling from the observation and hypothesis that encrypted

malware uses encryption in a distinct way that allows it to be detected, to conducting machine learning based experiments to test it. The difference is that, instead of "gathering data" for experiments, we make use of already available datasets.

However, a major part of experimental 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 [20, 24]. This engineering inspired method is used in some papers, where we apply existing scientific knowledge (gained by us and others) to propose novel systems. In Paper II, we propose and implement a novel heuristic based solution to solve one aspect of the *cross-path data fragmentation* issue identified in Paper I. In Paper III, we draw inspiration from the information theory fields to propose a variation to a well known algorithm. This enabled us to implement the proposed variation in Paper IV and evaluate it. In Paper VI, a novel tool to translate IDS rules into corresponding payload traffic was proposed, implemented, and presented to fill the gap in the community 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. The datasets used in Paper V include one private dataset, whereas the remaining datasets are freely available and can be used to reproduce the ideas of the paper.

6 Summary of Appended Papers

A summary of the appended papers is given below.

Paper I – Multipath TCP IDS Evasion and Mitigation

In this paper, we address the first research question. In particular, the practicality and severity of *cross-path data 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 a *MPTCP Linker* tool. The paper outlines the importance of MPTCP support in future security middleboxes.

Paper II – Towards Multipath TCP Aware Security Technologies

In this paper, we focus on the second research question and design solutions to tackle the scenario when cross-path data fragmentation attacks are made possible because of an inability to understand MPTCP semantics. We 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 (that can detect cross-path data fragmentation) and prevention as well as threshold rules to prevent DoS attacks.

Paper III – Slice Distance: An Insert-Only Levenshtein Distance with a Focus on Security Applications

This paper also addresses the second research question for the scenario when security middleboxes such as IDSs have access to only some parts of the connection traffic. We propose an algorithm based on an insert-only variation of the Levenshtein distance to enable comparison of two strings in pattern matching for the case in which differences occur only because of missing bytes. The proposed metric is formally presented, and its computational complexity is discussed.

Paper IV – Using Partial Signatures in Intrusion Detection for Multipath TCP

This paper builds on the slice distance metric proposed in Paper III. We show that, by using a specially tailored partial signature matcher and knowledge about MPTCP semantics, the Snort3 IDS can be empowered with partial signature detection. Experimental results show a low false positive rate for benign traffic and high detection coverage for attack traffic.

Paper V – Using Features of Encrypted Network Traffic to Detect Malware

In this paper, we address the third research question of this thesis and propose a way to detect encrypted malware without decryption. As encrypted traffic on the Internet is increasing, unfortunately, at the same time malware is also increasingly using encryption to hide its operation. Using large datasets of benign and malware traffic, we study and extract features that are distinct to how malware makes use of encryption. These features are used to build machine learning classifiers to enable accurate detection of encrypted malware without decryption.

Paper VI – IDS Rule Management Made Easy

This paper provides an indirect contribution of this thesis. We propose a tool to translate the Snort IDS ruleset into corresponding traffic. The tool can be

used in a number of applications, e.g., in IDS rule management to optimize a ruleset. The tool is used in Paper I, II, and IV to generate synthetic attack traffic.

7 Concluding Remarks and Outlook

As the Internet has grown in size and complexity, the need to support new applications and functions has led to it moving away from the end-to-end design principles [51]. Today, there are several types of middleboxes that are an important part of the Internet. Some of these middleboxes were introduced to solve problems, such as NAT boxes that solve the address depletion problem, while others perform additional functions to enhance the performance and/or security of the network. The subject of this thesis are the security middleboxes used in the Internet and the challenges they encounter as a result of emerging protocols and technologies. MPTCP is the protocol considered in this work as it has the potential for large scale deployment. 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 [16, 47]. The security of the protocol itself, in isolation to its environment, is also under the microscope [8, 33]. However, the unexpected network security implications of the protocol on existing security middleboxes such as IDSs have not yet been thoroughly investigated. This thesis takes a step towards that.

The first objective of this work deals with exploring potential risks that can be raised by using MPTCP over unaware security middleboxes. To do that, the work explores one potential security implication of MPTCP and examines its feasibility. Using a pragmatic methodology, it is shown that the risk for IDS evasion is real and the current security middleboxes are vulnerable to similar attacks. Furthermore, the work proposes and implements possible solutions to the identified problems. Different solutions are proposed by distinguishing between two scenarios depending on whether or not the security middleboxes can observe all the traffic needed to perform their function. For the first scenario, we propose a way to perform intrusion detection when an IDS device can observe all traffic, but not understand its semantics. Additionally, as for the time being most devices on the Internet do not support MPTCP themselves, the solution is integrated in an MPTCP proxy to enable the use of the protocol even when the server is not MPTCP capable. The security services on the proxy ensure that security functions are not affected. Another possibility that can arise with MPTCP is security middleboxes having to perform their functionality based on only parts of the traffic from an MPTCP connection. In this case, only understanding the protocol semantics is not enough for a middlebox to perform its function correctly. To deal with such a scenario, a method is proposed in the thesis to detect intrusions from partial traffic. The detection uses a metric based on a proposed variation to Levenshtein [38] distance and is shown to be promising.

To assess the impact of new technologies on the security middleboxes, the second objective of this work concentrates on end-to-end encryption and makes an attempt to restore lost middlebox functionality. The work investigates the manner in which encryption is used for malicious purposes and whether there is a way to develop new functionality in security middleboxes such that they can detect these events without having to decrypt the traffic. A quantitative study is conducted using comprehensive malware and benign datasets to outline specific features that can be used by a machine learning model to distinguish between malware and benign traffic. Experiments with those features show that it is possible to detect encrypted malware with a high precision using only the unencrypted metadata of the traffic.

It is also worth mentioning some limitations of this thesis work. Although one of the objectives of the work was to explore different MPTCP security related issues, the work mainly focuses on and addresses the problem of *cross-path data fragmentation* attacks. Other possibilities in which MPTCP can affect security middleboxes, e.g., possibility of reverse connections, also need to be addressed and call for a number of future research directions. It will not be feasible to promote the deployment of the protocol at a wide scale unless it can be used in a secure way. The solutions proposed in this thesis are also mainly prototypes and proofs-of-concepts. We evaluated the solutions using the best possible means (often using synthetic traffic) and, while they show good promise, they are not ready for production systems as they are. These solutions should ideally be further tested using more realistic traffic and modified according to the deployment needs. Furthermore, encryption is already ubiquitous on the Internet. Although we make an effort to address the problem of accurate encrypted malware detection without decryption, the possibility of false positives still exists and requires further attention. There is also a need for more work to help middleboxes restore other functionality they lost due to the lack of visibility. Such solutions are important to fill the gap for the short-term where end-host-based-only security monitoring solutions are not yet up to par.

References

[1] Internet live stats. `http://internetlivestats.com`. Accessed: 2019-10-27.

[2] H. Abelson, R. J. Anderson, S. M. Bellovin, J. Benaloh, M. Blaze, W. Diffie, J. Gilmore, M. Green, S. Landau, P. G. Neumann, R. L. Rivest, J. I. Schiller, B. Schneier, M. A. Specter, and D. J. Weitzner. Keys under doormats: mandating insecurity by requiring government access to all data and communications. *J. Cybersecurity*, 1(1):69–79, 2015.

[3] F. Andreasen, N. Cam-Winget, and E. Wang. TLS 1.3 Impact on Network-Based Security. Internet-Draft draft-camwinget-tls-use-cases-05, Internet Engineering Task Force, July 2019. Work in Progress.

[4] G. Armitage, N. Williams, et al. FreeBSD kernel patch to enable Multipath TCP. `https://bitbucket.org/nw-swin/caia-mptcp-freebsd`. Accessed: 2019-10-27.

[5] R. Atkinson and S. Kent. Security Architecture for the Internet Protocol. RFC 2401, Nov. 1998.

[6] T. Aura, P. Nikander, and G. Camarillo. Effects of mobility and multihoming on transport-protocol security. In *Proceedings of the IEEE Symposium on Security and Privacy (S&P), 9-12 May 2004, Berkeley, CA, USA*, pages 12–26, 2004.

[7] S. Axelsson. Intrusion detection systems: A survey and taxonomy. Technical report, 2000.

[8] 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`.

[9] M. S. Blumenthal and D. D. Clark. Rethinking the design of the Internet: the end-to-end arguments vs. the brave new world. *ACM Transactions on Internet Technology*, 1(1):70–109, 2001.

[10] O. Bonaventure. Apple seems to also believe in Multipath TCP. `http://perso.uclouvain.be/olivier.bonaventure/blog/html/2013/09/18/mptcp.html`. Accessed: 2019-10-27.

[11] O. Bonaventure. Apple uses Multipath TCP. `http://blog.multipath-tcp.org/blog/html/2018/12/15/apple_and_multipath_tcp.html`. Accessed: 2019-10-27.

[12] O. Bonaventure. Multipath TCP is pronounced giga path in Korea. `http://blog.multipath-tcp.org/blog/html/2015/07/24/korea.html`. Accessed: 2019-10-27.

[13] 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`.

[14] D. Cameron. PM: spy agencies need more powers to protect britain. `https://www.theguardian.com/uk-news/video/2015/jan/12/david-cameron-spy-agencies-britain-video`, 2015. Accessed: 2019-10-29.

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

[16] 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.

[17] Cisco White paper. Cisco encrypted traffic analytics. `https://www.cisco.com/c/dam/en/us/solutions/collateral/enterprise-networks/enterprise-network-security/nb-09-encrytd-traf-anlytcs-wp-cte-en.pdf`. Accessed: 2019-10-29.

[18] J. B. Comey. Going dark: Are technology, privacy, and public safety on a collision course? `https://www.fbi.gov/news/speeches/going-dark-are-technology-privacy-and-public-safety-on-a-collision-course`, 2014. Accessed: 2019-10-29.

[19] Q. D. Coninck and O. Bonaventure. Tuning multipath TCP for interactive applications on smartphones. In *Proceedings of the 17th International IFIP TC6 Networking Conference, Zurich, Switzerland, May 14-16, 2018.*, pages 487–495, 2018.

[20] 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.

[21] G. Detal. MPTCP-enabled kernel for the Nexus 5. `https://github.com/gdetal/mptcp_nexus5`. Accessed: 2019-10-27.

[22] Z. Durumeric, Z. Ma, D. Springall, R. Barnes, N. Sullivan, E. Bursztein, M. Bailey, J. A. Halderman, and V. Paxson. The security impact of HTTPS interception. In *Proceedings of the 24th Annual Network and Distributed System Security Symposium, NDSS San Diego, California, USA, February 26 - March 1, 2017*, 2017.

[23] ENISA. Strong encryption safeguards our digital identity. `https://www.enisa.europa.eu/publications/enisa-position-papers-and-opinions/enisas-opinion-paper-on-encryption`. Accessed: 2019-10-27.

[24] 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.

[25] S. Fenter. Why Enterprises Need Out-of-Band TLS Decryption. Internet-Draft draft-fenter-tls-decryption-00, Internet Engineering Task Force, Mar. 2018. Work in Progress.

[26] A. Ford, C. Raiciu, M. J. Handley, O. Bonaventure, and C. Paasch. TCP Extensions for Multipath Operation with Multiple Addresses. Internet-Draft draft-ietf-mptcp-rfc6824bis-18, Internet Engineering Task Force, June 2019. Work in Progress.

[27] Google. HTTPS encryption on the web. `https://transparencyreport.google.com/https/overview`. Accessed: 2019-10-29.

[28] M. Green, R. Droms, R. Housley, P. Turner, and S. Fenter. Data Center use of Static Diffie-Hellman in TLS 1.3. Internet-Draft draft-green-tls-static-dh-in-tls13-01, Internet Engineering Task Force, July 2017. Work in Progress.

[29] M. Honda, Y. Nishida, C. Raiciu, A. Greenhalgh, M. Handley, and H. Tokuda. Is it still possible to extend TCP? In *Proceedings of the 11th ACM SIGCOMM Internet Measurement Conference, IMC, Berlin, Germany*, pages 181–194, 2011.

[30] R. Housley and R. Droms. TLS 1.3 Option for Negotiation of Visibility in the Datacenter. Internet-Draft draft-rhrd-tls-tls13-visibility-01, Internet Engineering Task Force, Mar. 2018. Work in Progress.

[31] C. Huitema. Multi-homed TCP. Internet-Draft draft-huitema-multi-homed-01, Internet Engineering Task Force, Nov. 1995. Work in Progress.

[32] J. Iyengar and M. Thomson. QUIC: A UDP-Based Multiplexed and Secure Transport. Internet-Draft draft-ietf-quic-transport-24, Internet Engineering Task Force, Nov. 2019. Work in Progress.

[33] 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.

[34] A. Kharpal. Apple vs FBI: All you need to know. `https://www.cnbc.com/2016/03/29/apple-vs-fbi-all-you-need-to-know.html`. Accessed: 2019-10-29.

[35] Y. Kou, C.-T. Lu, S. Sirwongwattana, and Y.-P. Huang. Survey of fraud detection techniques. In *Proceedings of the IEEE International Conference on Networking, Sensing and Control*, volume 2, pages 749–754, 2004.

[36] S. Landau. *Surveillance or security?: The risks posed by new wiretapping technologies*. Mit Press, 2011.

[37] S. E. Landau. *Listening in: Cybersecurity in an insecure age*. Yale University Press, 2017.

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

[39] C. M. Lonvick and T. Ylonen. The Secure Shell (SSH) Protocol Architecture. RFC 4251, Jan. 2006.

[40] E. A. Lopez-Rojas and S. Axelsson. A review of computer simulation for fraud detection research in financial datasets. In *Proceedings of the 2016 Future Technologies Conference (FTC)*, pages 932–935, 2016.

[41] K. Moriarty and A. Morton. Effects of Pervasive Encryption on Operators. RFC 8404, July 2018.

[42] A. Müller. *Analysis and Control of Middleboxes in the Internet*. PhD thesis, Technische Universität München, 2013.

[43] R. Oppliger. Internet security: Firewalls and beyond. *Communications of the ACM*, 40(5):92–102, 1997.

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

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

[46] PwC International Limited. The global state of information security survey 2018. `https://www.pwc.com/us/en/services/consulting/cybersecurity/library/information-security-survey.html`. Accessed: 2019-10-27.

[47] 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.

[48] E. Rescorla. The Transport Layer Security (TLS) Protocol Version 1.3. RFC 8446, Aug. 2018.

[49] 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.

[50] RSA Conference 2018. Network monitoring is going away...now what? TLS, QUIC and beyond. `https://www.rsaconference.com/usa/us-2018/agenda/network-monitoring-is-going-awaynow-what-tls-quic-and-beyond-2`. Accessed: 2019-10-29.

[51] J. H. Saltzer, D. P. Reed, and D. D. Clark. End-to-end arguments in system design. *ACM Trans. Comput. Syst.*, 2(4):277–288, 1984.

[52] A. Sparrow. WhatsApp must be accessible to authorities. `https://www.theguardian.com/technology/2017/mar/26/intelligence-services-access-whatsapp-amber-rudd-westminster-attack-encrypted-messaging`. Accessed: 2019-10-29.

[53] R. R. Stewart. Stream Control Transmission Protocol. RFC 4960, Sept. 2007.

Life of a Security Middlebox

The Internet of today has intermediary devices known as middleboxes that perform more functions than the normal packet forwarding function of a router. Security middleboxes are a subset of these middleboxes and face an increasingly difficult task to perform their functions correctly. These middleboxes make many assumptions about the traffic that may not hold true any longer with the advent of new protocols such as MPTCP and technologies like end-to-end encryption.

The work in this thesis focuses on security middleboxes and the challenges they face. We develop methods and solutions to help these security middleboxes continue to function correctly. In particular, we investigate the case of using MPTCP over traditional security infrastructure as well as the case of end-to-end encryption. We study how practical it is to evade a security middlebox by fragmenting and sending traffic across multiple paths using MPTCP. We then go on to propose possible solutions to detect such attacks and implement them. The potential MPTCP scenario where security middleboxes only have access to part of the traffic is also investigated and addressed. Moreover, the thesis contributes a machine learning based approach to help security middleboxes detect malware in encrypted traffic without decryption.

ISBN 978-91-7867-093-2 (print)

ISBN 978-91-7867-103-8 (pdf)

ISSN 1403-8099

DOCTORAL THESIS | Karlstad University Studies | 2020:10

Life of a Security Middlebox

The Internet of today has intermediary devices known as middleboxes that perform more functions than the normal packet forwarding function of a router. Security middleboxes are a subset of these middleboxes and face an increasingly difficult task to perform their functions correctly. These middleboxes make many assumptions about the traffic that may not hold true any longer with the advent of new protocols such as MPTCP and technologies like end-to-end encryption.

The work in this thesis focuses on security middleboxes and the challenges they face. We develop methods and solutions to help these security middleboxes continue to function correctly. In particular, we investigate the case of using MPTCP over traditional security infrastructure as well as the case of end-to-end encryption. We study how practical it is to evade a security middlebox by fragmenting and sending traffic across multiple paths using MPTCP. We then go on to propose possible solutions to detect such attacks and implement them. The potential MPTCP scenario where security middleboxes only have access to part of the traffic is also investigated and addressed. Moreover, the thesis contributes a machine learning based approach to help security middleboxes detect malware in encrypted traffic without decryption.

ISSN 1403-8099 | ISBN 978-91-7867-093-2 (print) | ISBN 978-91-7867-103-8 (pdf)

DOCTORAL THESIS | Karlstad University Studies | 2020:10