THE UNIVERSITY *of* EDINBURGH

An Empirical Study of the Transport Layer Performance and Security in Mobile Networks

Rupendra Nath Mitra

Master of Philosophy

School of Informatics
University of Edinburgh

2022

Abstract

Enabling ultra-low latency, and high-speed reliable connectivity, the latest generations of mobile networks are equipped to cater to diverse business and application requirements. Technologies such as software-defined networking (SDN), control and user plane separation (CUPS), deep-learning-based AI solutions along with newer networking protocols such as QUIC play critical roles in mobile networks to ensure state-of-the-art network performance.

This thesis thoroughly reviews the evolving security threat landscape of the modern mobile networks, empirically investigate the impacts of CUPS hijacking of a radio access network (RAN) slicing system on the overall network performances, and finally presents an experimental evaluation and characterization of QUIC performance over commercial 5G network.

In emerging mobile networks, CUPS plays a critical role in scaling the control-plane and user-plane functions independently and enables network virtualization through network slicing. However, a CUPS hijacking attack on a mobile network slicing system and the resulting network performance degradation are yet to be studied.

We investigate the consequences of CUPS hijacking of a RAN slicing system on the overall network performance. We quantify the impacts of CUPS hijacking by designing an Impact Factor metric I, prototype a real-world RAN slicing use case on an end-to-end mobile network test-bed, and systematically analyze the empirical results to reveal the impacts of CUPS hijacking on the network performance. We show a successful CUPS hijacking by a rogue slice owner in a RAN slicing system increases the RAN slice control-plane signaling delay above 2ms, the operational upper-bound of our system, to disrupt the control plane operations by injecting low rate DoS (LDoS) traffic in user-plane. The naive hijacking can degrade the throughput performances of the rogue slice as well as a co-located victim slice down to 0 Mbps. We further show that carefully crafted user-plane traffic by the attacker can regain 92% of its original user-plane packet delivery success rate while other slices are under the denial of service.

On the other hand, the new transport layer protocol QUIC is being adopted by major internet application providers such as Google and Facebook signifying a paradigm shift from the de-facto transport layer protocol TCP. However, the interactions between 5G as a network data link layer for QUIC transport protocol to deliver modern web browsing, video streaming, and file downloading applications remain unexplored.

We conduct an in-depth study of 5G and QUIC interactions and their impact on application performances by collecting measurements taken in the wild involving commercial 5G networks and production-grade QUIC application servers. Results from the study reveal that end-user experience of accessing popular web services such as 4K video streaming on YouTube, file downloading from the Google cloud server, Google Search, and Maps on QUIC remain similar to the experience when accessed on TCP, over mobile networks.

Lay Summary of Thesis

5G mobile networks rely on sophisticated underlying software technologies and newly adopted network protocols. However, how a mobile network's performance is impacted by cyberattacks that exploit security loopholes in a key mechanism that separates user data traffic from network control messaging, and how recent network transport protocols, whose primarily goal is reliable data transfer, perform in conjunction with 5G remain unexplored.

This thesis aims to answer these questions by first reviewing the evolving security threat landscape in mobile networks, empirically investigating the impacts of a special kind of cyberattack on a radio access network on the overall network performances. Secondly, it characterizes the performance of popular web apps (YouTube video streaming, Google Maps, Search, file download) with QUIC, a transport protocol increasingly adopted by cloud providers, over commercial 5G networks.

The results presented in the thesis reveal that a successful cyberattack can completely disrupt mobile network operation. In addition, end-user experience of accessing popular web services such as 4K video streaming on YouTube, file downloading from Google cloud, Google Search, and Maps over QUIC remains similar to that when these services are accessed using TCP over the 4G and 5G mobile networks. We believe that the research findings presented in the thesis motivate further research endeavours in mobile network security and optimisation of applications for secured and enhanced end-user experience.

Acknowledgements

I would first like to thank my supervisor Dr. Paul Patras for all his support and guidance throughout my MPhil. I am also thankful to my second supervisor, Prof. Nishanth Sastry for his guidance and for all the helpful discussions with him during my tenure. Special thanks go to Dr. M.M.M. Kassem for his invaluable brainstorming sessions with me.

I am grateful to my thesis examiners Dr. Antonio Barbalace and Dr. Posco Tso for their time and effort as well as for their valuable feedback.

My deepest gratitude goes to my parents. This Journey of completing the MPhil would have been impossible without the unconditional love and support they have given me throughout these years.

Declaration

I declare that this thesis is composed by myself and that the works and scientific results contained herein are produced by myself except where explicitly stated otherwise in the text and that this work has not been submitted for any other degree or professional qualification except as specified. Part of the material used for the contributions made by my thesis has been published in the papers listed below.

- **Mitra, Rupendra Nath**, and Mahesh K. Marina. "5G Mobile Networks Security Landscape and Major Risks." Wiley 5G Ref: The Essential 5G Reference Online (2019): 1-23.
- Usama, Muhammad, Inaam Ilahi, Junaid Qadir, **Rupendra Nath Mitra**, and Mahesh K. Marina. "Examining machine learning for 5g and beyond through an adversarial lens." IEEE Internet Computing 25, no. 2 (2021): 26-34.
- **Mitra, Rupendra Nath**, Mohamed M. Kassem, Jon Larrea, and Mahesh K. Marina. "CUPS Hijacking in Mobile RAN Slicing: Modeling, Prototyping, and Analysis." In 2021 IEEE Conference on Communications and Network Security (CNS), pp. 38-46. IEEE, 2021.

List of Figures

List of Tables

Table of Contents

Chapter 1

Introduction

Emerging 5G mobile networks along with technological advances in communication networking in general, such as standardization of QUIC transport layer protocol, and proliferation of high-end smartphones and IoT devices assisted by ubiquitous satellite coverage enable new-age cloud-native applications anytime anywhere.

However, if a catastrophe like a well-crafted cybersecurity attack hits a future mobile network, how will this impact the network performance, and what difference can a new transport paradigm like the QUIC protocol make to end-user application performance while accessed over a real-life commercial 5G network? This thesis empirically studies the impacts of a security attack on modern mobile radio access network systems and also characterizes production-grade application performance over QUIC accessed over commercial 5G networks.

The results presented in this thesis will motivate further research in making future networks more robust to withstand the evolving attack vectors and also provides timely insights into the interactions of QUIC and 5G networks in the wild.

1.1 Context

1.1.1 Emerging Mobile Network Architecture: 5G

Mobile communication networks have been rapidly transforming our everyday life. Beginning with a first-generation analog voice communication system on the go, the fourth generation of mobile networks currently guarantees high-speed Internet and ubiquitous connectivity among numerous smart devices. The present-day cellular networks cater to diversified services such as banking, health, governance, e-commerce,

education, and mobile TV, to name a few [76]. Yet today's mobile network infrastructure requires another leap to its 5th generation (5G), to fulfill the growing demand of higher data rates than those observed in existing 4G mobile Internet (which offers approximately 100Mbps) and the need for extremely low-latency (less than a millisecond) communication links, enabling futuristic applications such as mission-critical communications, augmented/virtual reality, self-driving cars, and so on [99].

The 5G mobile networks will not just be a technologically advanced version of currently deployed 4th generation (4G) systems, but an utterly new telecommunication paradigm powered by software-defined networking (SDN), network function virtualization (NFV), and cloud computing. The adoption of these new technologies in telecommunication system design brings unprecedented network agility and offers service providers the ability to implement 5G infrastructures incredibly fast. However, along with the unmatched network flexibility and advantages mentioned above, 5G networks bring an increased concern for security and privacy. The unrestricted usage of open source software and machine learning techniques, provision of virtualized services over shared physical infrastructure, connecting almost everything, and finally, the softwarization of the telecommunication network functions increase the cyberattack surface in 5G networks [5].

5G will play a pivotal role in building a networked society because critical services such as public transportation, (i.e., railway, buses, etc.), energy distribution, national security, and healthcare, which currently operate on exclusive networks, will heavily rely on the pervasive 5G infrastructures. Therefore, any adversary may launch malicious attacks on the 5G networks and, if successful, can bring severe disruptions to society [69]. Therefore, the adoption of open-source software, authentication protocols, machine learning (ML) models, and multi-tenancy public cloud platform operations needs to be overseen and scrutinized formally before being integrated into the 5G infrastructure. A systematic study on privacy-preserving techniques and policies in 5G to secure it from innumerable cyber threats is essential and timely [56].

5G Network Architecture: A Paradigm shift from 4G Today's cyber world is well connected through physical transoceanic cables, numerous satellite links, long-haul optical or unguided media, wide-area connectivity, and local wireless and wired networks. Traditionally, end-users connect to the worldwide cyber networks using user equipment, typically a general-purpose computer or a smartphone, through the mobile network's radio access network or the locally available access network. With the ad-

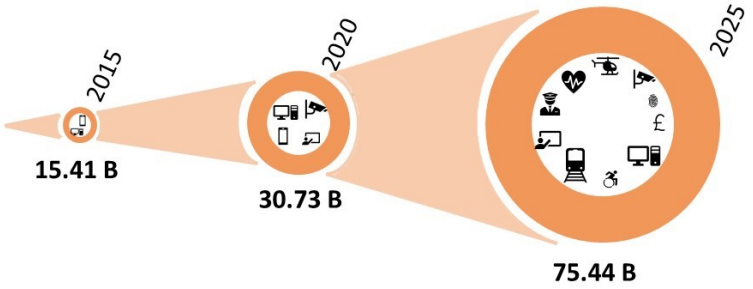


Figure 1.1: The projected boom in the number of connected gadgets [101]

vent of smart devices, exponential growth has been identified in the projected number of connected Internet of Things (IoT) in the coming years, as illustrated in Figure 1.1. But with the growing demand for staying always connected everywhere and accessing high-volume traffic on the go, cellular connection emerged as the predominant way to provide ubiquitous connectivity to users. Thus 5G network needs to be designed in a very different way than its 4G counterpart so that it can cater to the massive amount of connections to numerous futuristic services and many smart IoT devices for billions of its subscribers, providing high-speed cellular broadband [7].

5G networks' architecture is meant to be fundamentally different than that of 4G, because 5G necessitates backward compatibility and agile, modular architecture for an expeditious low-cost deployment and smooth technical augmentation in the future. The present-day 4G network inherits architectural features from its previous generations of traditional mobile networks. The 4G network has three segments, the user equipment (UE), the Radio Access Network (RAN) that connects to the UE over the air interface, and Evolved Packet Core (EPC) networks that connect to the RAN through transmission networks. However, 5G system (5GS) architecture has UE, multifaceted radio access networks called New Radio (NR), and the cloud-native 5G core networks (5G-CN) [5,8,25]. A piece of Mobile equipment (ME) equipped with a universal subscriber identity module (USIM) constitutes the UE through which the users can access the 5G network over the air interface.

The important entity of NR is the gNB, which hosts three main functional units called Radio Unit (RU), Distributed Unit (DU), and Centralized Unit (CU). Thus NR becomes a flexible logical architecture, unlike 4G RAN, which has a monolithic ar-

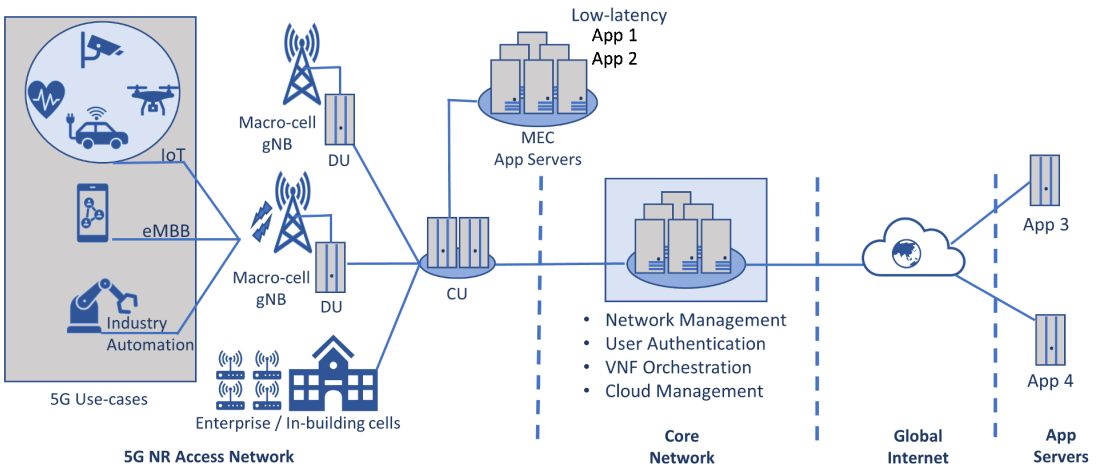


Figure 1.2: A schematic diagram of the 5G network architecture illustrating the disaggregated RAN architecture with the distributed unit (DU) and centralized unit (CU) components; the MEC for improved latency; and the cloud-native core network and system orchestration components.

chitecture. A 5G radio network (RN) can be deployed over the serving cell site and mobile edge server premises in different combinations of functional units, as required by the application scenario. For example, only RU can be installed in the cell site with minimum installation cost and complexity, and DU-CU can be installed on the edge site (in a local exchange office) to build a latency-tolerant access network. In contrast, a DU split from the CU can be installed in the cell site associated with RU for improved latency but at a higher installation cost and complexity. The latter is a lower-layer split of RN functions, and the prior topology is an example of a higher-layer functional split. DU-CU could also be integrated with the RU in the cell site for ULLC in case of industrial automation.

A schematic diagram of the 5G network architecture is depicted in Figure 1.2. Apart from the user equipment (UE), the 5G system features a cloud-native core network, a flexible and disaggregated radio access network (RAN), and a provision for a multi-access edge (MEC) cloud for reduced latency. The RAN comprises of gNodeB (gNB) access nodes, split into distributed and centralized units (DU and CU), to efficiently handle evolved network requirements. The gNB connects to the MEC to significantly reduce the network latency for selected applications by leveraging edge server computing at the MEC cloud, which is close to the radio service cells. For instance, to cater to the ultra-reliable low-latency communication (URLLC) use-case of industry automation, the RAN radio unit along with the DU, CU, and MEC can be

installed onsite. Thus, 5G network architecture enables applications to be deployed remotely (App 3 and App 4) or near the edge (App 1 and App 2), later when low latency is a requirement. The provision of MEC also reduces the aggregated traffic load on the transport networks responsible for connecting RAN to the core network. The 5G core network (5G-CN) is a cloud-native network that stores subscriber databases and hosts essential virtualized network functions for network operations and management. Although the network management and control functions are shown to be co-located with the core in the figure, they can be flexibly deployed at the edge as needed.

5G is meant to co-exist with legacy networks as well as have a stand-alone network architecture. Therefore, the 5G-CN has two different implementable provisions: Standalone (SA) and non-Standalone (NSA) mode. Besides, 5G-CN implements the network functions much like a 4G EPC does [10]. Still, 5G-CN is defined as a Service-Based Architecture (SBA) framework for increased modularity and function reusability. 5G-CN hosts various network functions (NFs) interfaced among them instead of various core network entities of a traditional mobile network. SBA architecture applies only to interfaces among control plane (CP) functions within the 5GC. The control plane network functions of the 5G-CN interact using service-based interfaces (SBI). This functional core architecture helps an implemented technology in the system to evolve and makes it seamlessly replaceable by its suitable future counterparts. 5G-CN brings more scalability, flexibility, and upgradability in the network architecture enabling the operators to build and maintain the network at a low cost. Cloud-based core network functionalities and NFV become the significant enablers of 5G mobile network architecture because of the SBA and the network slicing.

Figure 1.3 looks into a 5G network from a functional perspective, where the cloud-native 5GC NFs interact among themselves over defined interfaces. The main NFs are shown in the diagram and discussed below.

The Network Functions Repository Function (*NRF*) registers and discovers NF services. An NF identifies other NFs through NRF. The Unified Data Management (*UDM*) performs significant functionality of the Home Subscriber Server (HSS) from EPC. It is responsible for user identity management, subscription management, and the generation of AKA credentials. The Authentication credential Repository and Processing Function (ARPF) retains the authentication credentials. Only the Subscription Identifier De-concealing Function (SIDF) decrypts a Subscription Concealed Identifier (SUCI) to obtain its long-term identity, namely the Subscription Permanent Identifier (SUPI).

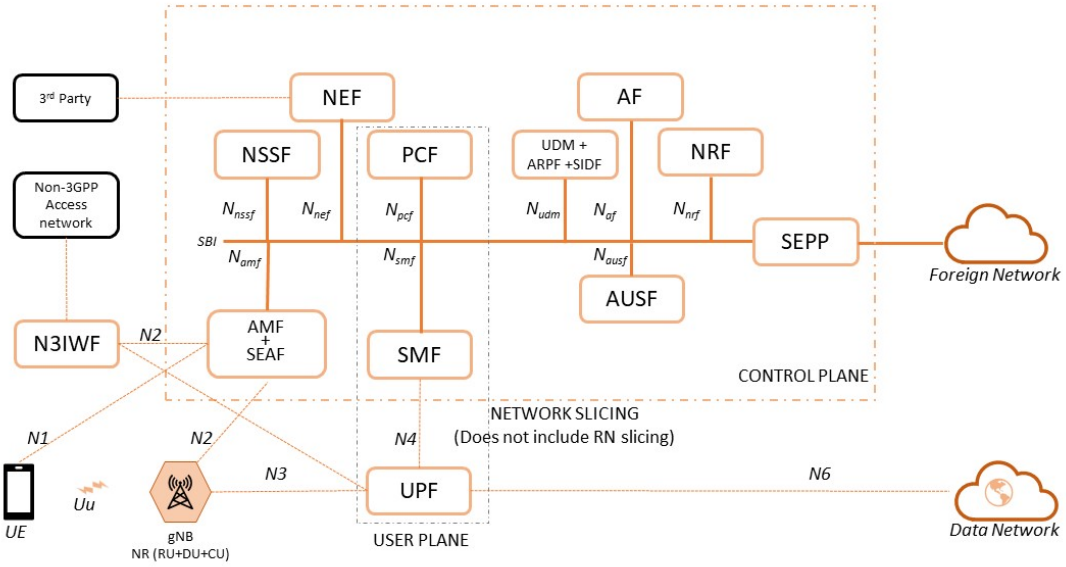


Figure 1.3: 5G-CN SBA representation as specified by 3rd Generation Partnership Project (3GPP) [9]

The Policy Control Function (*PCF*) is synonymous with the PCRF of the EPC framework. It provides policy decisions to control plane functions.

The Network Exposure Function (*NEF*) is a new network function created for 5G-CN and was not present in 4G EPC. It ensures secure information exchange to and from an external application to the 3GPP core and works like an exposure function.

The Application Function (*AF*) replicates the functionalities of AF from EPC architecture. The Authentication Server Function (*AUSF*) is the authentication server that completes the HSS's functionalities of an EPC along with UDM. The AUSF keeps a key that is derived after authentication for reuse in case of simultaneous registration of a UE in different access networks.

The Access and Mobility Function (*AMF*) is a sub-function of the Mobility Management Entity (MME) of 4G EPC. AMF tackles all the connection and session information from UEs or RAN and performs connection and mobility management tasks. The Session Management Function (SMF) takes care of the session management tasks. The AMF is collocated with the SEcurity Anchor Function (SEAF) which permits the secure introduction of future functions.

The *SMF* performs the interaction with the decoupled data frame and also acts as

an Internet Protocol (IP) address manager and a Dynamic Host Configuration Protocol (DHCP) server.

The Network Slice Selection Function (*NSSF*) offers services to the AMF and NSSF in a different Public Land Mobile Network (PLMN) via the Nnssf service-based interface within 5G-CN. NSSF is responsible for network slice selection and management.

The Non-3GPP Interworking Function (*N3IWF*) is similar to the ePDG (for Evolved Packet Data Gateway) in the 4G EPC. N3IWF allows the UE to access 5G-CN through untrusted non-3GPP access networks. It is responsible for establishing secure communication between UE and 5G-CN and routing messages outside the 5G RAN. However, 3GPP does not categorize any non-3GPP technologies as untrusted. This decision is left on the operator.

The User Plane Function (*UPF*) is the functional point between the mobile network and the Data Network (DN). UPF facilitates Control and User Plane Separation (CUPS) for non-Standalone (NSA) 5GS which allows scalable and efficient management of packet traffic in the mobile edge.

The Security Edge Protection Proxy (*SEPP*) is a new NF called Security Edge Protection Proxy introduced in 5GS by 3GPP SA3 to ensure secure signaling traffic between different operator networks. The N32 interface between the SEPPs is designed to protect sensitive data flowing through it and to ensure secure authentication between SEPPs.

The NSA mode is an intermediate step toward a full 5G network from the legacy network. In this configuration, 5G gNB facilitates the access network by carrying user plane (UP) traffic along with 4G LTE eNB but connects to the 4G EPC (and not 5G-CN) over the standard S1-U interface. The control plane traffic still routes through the 4G network. In other words, in NSA mode 5G access network increases the capacity of the existing 4G services as a secondary serving cell being a slave to the 4G network which controls the 5G access network as the master. Since NSA mode does not demand the all-new 5G-CN but is still capable of offering a 5G experience to the subscribers, telecom operators adopt NSA configuration as their first step to commercial 5G network.

On the other hand, SA mode demands 5G NR be connected to the 5GC hence it is the true 5G-only network capable of offering low latency mobile broadband service. 5G-CN is seen as the common core to 5G NR, 4G RAN, and any other type of access network in the future.

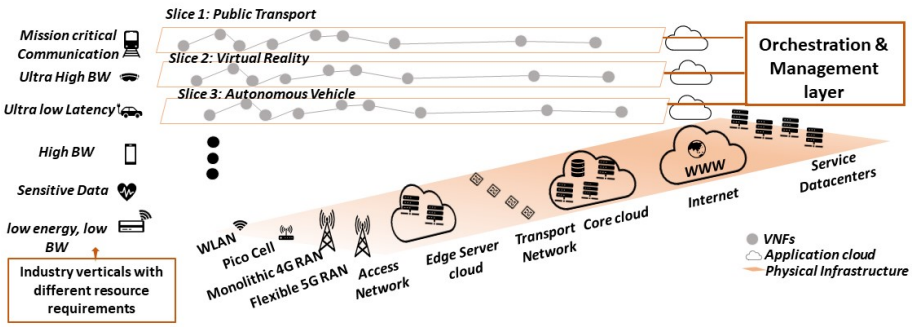


Figure 1.4: Schematic diagram of network slicing in an evolved network architecture of 5G coexisting with legacy networks

From the above description, we can understand that a majority of the 5G-CN NFs are created by refactoring the 4G EPC functions, MME, SPWG, HSS, and PCRF. At the same time, a set of brand new NFs such as NSSF, N3IWF, SEPP, or the SBA itself is introduced for enhanced network programmability, agility, and security. A full CP and UP separation in the 5G-CN is another unique trait where UPF processes UP data, and all other nodes tackle CP functionalities enabling separate scale-up of CP and UP operations.

Salient Features of 5G Network 5GS is a step toward the converged network of IT and telecommunication infrastructure for the future. Thus 5GS has several new features that the previous generations of telecom networks did not have. From a security perspective, 5G being a telecom network more similar to the IT infrastructure opens up a wider boundary of cyberattacks on the new unfortified sides of the 5G network [111]. The unique architectural amendments are briefly introduced in the following [21, 56].

- **New physical layer:** 5GS physical layer is significantly different from that of LTE. The use of a massive number of active antennae for efficient beamforming, high-frequency millimeter-wave signals for higher data rate over the air, and non-orthogonal multiple access for higher spectral efficiency make 5G new radio conspicuous from other radio communication systems [15].

- **Multi-access Edge Computing:** 5G enables operators to host latency-critical services closer to the end-user through the Multi-access Edge Computing (MEC) capability. MEC helps to reduce the traffic load on the transport network and meet the Service Level Agreements (SLA) of an ultra-low latency service by local hosting.

- **Network Slicing:** E2E network slicing is a significant feature of the 5G network. A network slice is an optimized set of network resources that enables a specialized service [51]. In principle, any given kind of slice of an operator can interconnect with the same kind of slice from another operator if the slice-sharing policy requirements are met. The NSSF enables the slice selection procedure. Figure 1.4 schematically describes the E2E network slicing over a 5G network infrastructure layer. Network slicing can be described as a horizontal network virtualization technique that offers flexibility to mobile operators to logically separate their different users with different SLAs. Mobile virtual operators can borrow their virtual network infrastructure in a pay-as-you-go fashion where E2E network slicing is in operation.

- **Cloud-native architecture:** In principle, cloud-native architecture promotes software services that are broken down into compact, more convenient pieces of constituent software which are achieved by using a microservice architecture. 5G-CN adopts a could-native architecture so that each piece of such micro technologies can be individually scaled, reconfigured, and upgraded. The actual cloud-native core network is a unique feature of 5G.

- **Multi-RAT dual connectivity:** 5G supports a UE to stay connected to more than one 5G eNB, and that also includes different Radio Access Technology (RAT) such as Wi-Fi and satellite networks. In such scenarios, the connected gNB will act as a master node and the other as a secondary node. This unique feature of 5G requires new security mechanisms and procedures to protect user traffic between UE and master and secondary nodes.

- **Support for non-3GPP access:** Apart from 3GPP access networks, 5G-CN opens its door to the non-3GPP access networks, such as satellite or Wi-Fi networks, with enhanced security functions defined by N3IWF. A UE can be served through Wi-Fi-like access networks directly connected to the 5GC for reduced latency. The provision of such many non-3GPP accesses to the 5GC requires

Table 1.1: Definitions of some common cybersecurity terms

Vulnerability	A weakness or lacuna which can be potentially exploited to damage the system or to have unauthorized access to the system.
Threat	Anything that can cash in on the vulnerabilities to destroy or paralyze the system. A threat can result from unintentional accidents as well as from a planned attack.
Attack	A planned activity that exploits security vulnerabilities and threatens the system by obtaining unauthorized access or causing disruptions in system operation.
Risk	A composite factor which is a function of vulnerability, threat, and the amount of damage caused. If a threat breaks in exploiting an existing vulnerability then how much potential damage it may cause to the system or in general to the business is called the risk associated with the vulnerability and threat. The likelihood of occurrence of the event is also another constituting factor of risk.
Trust	Psychological confidence in our general life but in the digital world, trust is often described as a binary model. Either one has complete trust or no trust in a digital system.
Privacy	The confidentiality of the user identity and concealment of sensitive data and personal information that is often sent over a network to which a user is connected.

more robust authentication than that of 4G ePDG.

Security Considerations 5G network will entice an unprecedented number of cyber attacks due to its pervasive reachability into high-value targets like governmental and financial organizations, its capability to serve a wide variety of industry verticals, and its accessibility to a massive amount of user data. Previous generations of the mobile network have seen several malicious attacks [57], [102], but 5G will face more aggravated threats. The attackers have continued to grow with time with inexpensive yet smarter AI-based attack tools, innovative side-channel attack techniques, and a plethora of malicious software capable of evading intrusion detection systems. The number of attacks that 5GS will experience is an order of magnitude higher than that of its previous generation counterparts [78].

Before we further explore the different aspects of the security threats on 5G, a few closely related technical vocabularies need to be clarified for a better understanding of the reader. Table 1.1 defines some basic cybersecurity terms which have occurred frequently in this thesis after that.

In the last few years, a broad set of attacks on GSM, UMTS, and LTE systems have been reported, which threatened user privacy, data secrecy, and business integrity. Rupprecht *et al.* analyzed the LTE layer two security vulnerabilities and designed

attack vectors that exploited the fact that LTE user payload is encrypted in counter mode (AES-CTR) but not integrity protected to modify user data [103]. Hussain *et al.* have shown the security loophole in downlink paging occasion and exploited the weakness to leak victim's soft identities and coarse-grained location [24]. So, the 5G defense mechanism needs further strengthening by underlying E2E network monitoring, reimplementing Artificial Intelligence (AI)-driven intrusion detection systems, updating authentication protocols, and fixing security bugs in its previous versions [49]. In this thesis, we present a pragmatic view of the evolving 5G threat space by zooming in on the transfigured network architecture and interfaces.

E2E security and privacy standards are a vital necessity for a hyper-scale, pervasive, and high-performance telecommunication network like 5G so that it can withstand the dynamic landscape of future threat vectors and build a profoundly trusted network architecture by securing customers' private data. 3GPP formally approaches 5G security and privacy, however, many aspects of it remain unattended by the security standardization bodies. We argue that this is the right time to consider 5G security and privacy as a design obligation and not to see it as an overhead to a faster system standardization because it is often not adequate and economical to annex security and privacy features afterward.

5G infrastructures revamp the traditional cellular network architecture radically to cater to the exponentially growing demand for high-speed, low-latency mobile Internet. Hence, the initial standardization process should include security and privacy concerns to thwart early attacks on the 5GS. Standardized security procedures should test the reliability of ML systems before we adopt them in the networks. cross-discipline research areas including cryptographic security, formal device testing, systematic and exhaustive testing of security procedures of 5G, and cyberattacks designing for 5G network components are necessary to bridge the security gaps that currently exist in present-day telecommunication systems.

The beneficiaries of a disruptive technology like 5G are people from all cross-sections of society. At the same time, any security defect can impact a significant section of society that depends on technology. Therefore, a part of this thesis highlights the emerging threat landscape and major risks associated with the advent of the 5G technology, anticipated impacts of unresolved security weaknesses, and future research directions to establish 5G as the most trusted network architecture for the years to come.

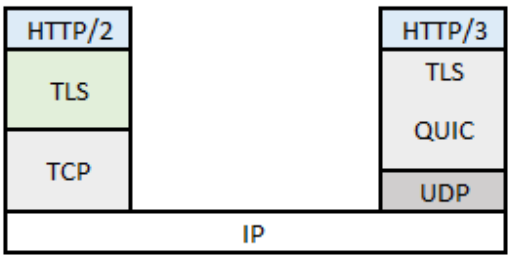


Figure 1.5: A Schematic diagram showing the basic differences between TCP and QUIC stacks

1.1.2 Emerging Transport Protocol: QUIC

Google first developed QUIC as a new transport protocol that leverages UDP under the hood to design an end-to-end encrypted, highly reliable, and faster layer-4 protocol to replace the decade-old TCP. QUIC was first introduced in 2013 and has undergone rapid development and adoption by Google and other big Internet companies [72]. Our study found, by early 2022, 1441 out of Tranco[1] top 10K websites deployed QUIC. Recently, the Internet Engineering Task Force (IETF) has published QUIC as RFC 9000 which officially formalized QUIC version 1 to be widely adopted throughout the Internet ecosystem.

The design of QUIC anticipated and overcame the ossification risk due to the introduction of a new protocol at the Internet scale (leading to a mass upgrade of existing operating systems kernels' transport protocols) by making the protocol an application-layer implementation on top of existing UDP transport protocol. Another aspect of QUIC protocol design has been the security consideration. QUIC is an end-to-end encrypted transport protocol that ensures users' privacy and data security over the Internet [65].

To achieve faster data transfer, reliable connection management, and improved application performance over rapidly evolving underlying network infrastructures, QUIC embraces a set of unique features that make QUIC conspicuous among its older counterparts TCP and UDP. We discuss those unique features of QUIC below.

Salient Features of QUIC As illustrated in figure 1.5, QUIC implements transport layer security (TLS) by design and combines TLS into the transport protocol instead of as an add-on security layer on top of TCP. Moreover, with end-to-end encryption,

[1] https://tranco-list.eu/

QUIC achieves secure and reliable application data delivery with proper flow control. QUIC also implements 0-round trip time (0-RTT) connection reestablishment, connection migration for mobile and heterogeneous network handover scenarios, and stream multiplexing for removal of head-of-line blocking that often slows down applications running over TCP [60].

- ***0-RTT Connection establishment:*** For a QUIC client that has had a connection in the recent past with a QUIC server, the connection can be reestablished right away without requiring a TCP-like 3-way handshake. That means, from the very first response the server can transmit application data to the client. This feature is known as 0-RTT connection establishment, which is designed to reduce TCP+TLS+HTTP/2 handover overheads. Moreover, QUIC implements TLS as an integral part. Thus QUIC always saves 1 RTT by having TLS negotiation built-in. In other words, QUIC always takes at least one RTT less to accomplish an HTTPS request when compared to the TCP.

  This smart connection reestablishment approach saves multiple RTTs from TCP-like connection establishment which we empirically quantify in our work to be a significant improvement for application performance metrics such as time to the first byte (TTBF).

- ***Connection Migration:*** Connection migration is a novel feature of QUIC. QUIC connections are not strictly bound to the network path through which the connection was first established between the server and the client. QUIC introduces a connection id to uniquely identify a connection between a server and a client and use the same connection id to resume back the connection if it gets broken due to a change in the network path.

  For example, unlike TCP, if a QUIC client establishes the connection from a Wi-Fi network but later moved to a mobile network, the new IP and the network path to the server will not force the client to re-initiate another fresh handshaking with the server. Rather the connection will be identified through the connection ID from the previous connection to resume the application packet transmission. Thus, the connection ID allows a QUIC connection to survive changes to endpoint addresses (IP address and port) and the network path.

  The QUIC endpoint that initiates the migration is termed as the the connection migration initiator and the peer is termed as the connection migration responder.

- ***Overcoming TCP's head-of-line blocking by Multiplexing streams:*** To overcome the head-of-line (HOL) blocking problem with TCP, where a lost packet containing data from one stream prevents the receiver from processing subsequent packets containing data from other streams, QUIC adopts stream multiplexing technique. Stream multiplexing allows data to be delivered in order only at the stream level, not at the connection level.

Thus, QUIC is designed and standardized to be a transmission protocol that can surpass the performance, robustness, and reliability of the de facto transport protocol TCP. However, an extensive real-life performance characterization and validation of QUIC against TCP over 5G networks in wild are yet to be reported in the literature.

1.2 Problem Description

In order to study the robustness and performance of the newly evolved mobile network infrastructure with new-age transport protocol in action, we split the bigger problem space into two separate studies containing specific research questions.

Problem description 1: Although mobile network security has been explored from protocol verification aspects, we recognize that there is a research gap in analyzing the system aspects of mobile networking that is rapidly merging with the IT and cloud infrastructure for increased operational agility. To be reasonably thorough in our study, we zeroed in on analyzing security vulnerabilities that emerge from newly adopted CUPS techniques in mobile RAN slicing systems. We take a real-world scenario of a RAN slicing system running on a typical neutral host and multi-operator indoor small cell (NHMO) as a use case.

In an NHMO set up the radio equipment and the associated IT infrastructure such as computing facilities and networks are owned and managed by the neutral host. The mobile operators can rent the infrastructure as needed, from the host and run their RAN controllers to manage their virtual RANs.

Cyberattacks like CUPS hijacking, where an adversary disrupts the control plane from the user plane, are well-studied in SDN-like cloud systems (a background on CUPS hijacking attacks is given in Chapter 3). However, the following research question remains open in the emerging amalgamated networking solution that combines cloud networks with mobile networks.

Research Question: What are the impacts on the overall mobile network performances under a CUPS hijacking attack on the SDN-based mobile RAN slicing system?

A detailed study through empirical evidence and quantifiable impact metrics is essential for obtaining insights into such disruptions, to be able to withstand similar attacks in the future.

Problem description 2: On another hand, the newly standardized QUIC transport protocol is being adopted by the most popular Internet application providers aiming for enhanced application performance over next-generation network infrastructure. However, a true validation and characterization of the applications' performance over commercial 5G networks with production-grade QUIC servers are yet to be studied. A systematic measurement study is to be performed to answer the following broad research question.

Research Question: How to characterize the 5G-QUIC interactions in terms of application performance gains at the user level, with real-life commercial 5G and production-grade, large-scale applications delivered over QUIC?

A city-wide large-scale application measurement collection on QUIC and TCP over commercial 5G networks is a first step toward characterizing the benefits of QUIC-5G interactions and providing further insights on how to optimize the future developments of applications and networks.

1.3 Challenges

Investigating research questions in the area of CUPS hijacking attacks on the mobile RAN and also in the area of QUIC performance characterization over 5G is particularly challenging for a set of reasons.

The dearth of studies in the literature on CUPS hijacking in mobile networks from a systems perspective can be explained by at least the following two challenges:

(i) Most of the real-world end-to-end mobile network slicing solutions are commercially developed and proprietary products that are not openly accessible to the research community;

(ii) Designing a testbed with a real-life mobile network running with modern CUPS mechanism and slicing techniques to conduct empirical studies on CUPS hijacking often demand enormous engineering efforts, human hours, and sophisticated hardware and network configurations;

On the other hand, conducting a large-scale network measurement study to characterize QUIC performance is challenging because

(i) The varied heterogeneity of the 5G ecosystem in terms of spectrum bandwidths and network coverage among the various operators makes it non-trivial to zero in on a proper experimental setup for the measurement study.

(ii) Maintaining consistency among long-term measurement collections requires automated scripts for different use cases and applications under the consideration in the study.

(iii) Apart from the technical challenges, there exist non-technical obstacles too. For instance, the amount of travel to collect field measurements to include both mobile and static experimental conditions and insure diverse geolocations is non-negligible and often expensive to carry out.

The experimental results reported in this thesis are achieved by reasonably addressing the above-mentioned challenges through systematic research methodologies that are illustrated in the following chapters of the thesis.

1.4 Importance

Commercial 5G is still in its infancy in terms of the roll-out of new cells, stand-alone 5G infrastructure, and integration of non-terrestrial and other heterogeneous technologies such as Open-RAN in a multi-vendor scenario. However, 3GPP is progressing through the standardization process of 5G and IETF has recently standardized the QUIC as a transport layer protocol [4]. A majority of the Internet companies like Google, and Facebook have already adopted QUIC as their alternate transmission protocol. Hence this is the right time to explore the security vulnerabilities in modern mobile systems and study the interactions of QUIC as a transmission protocol for next-generation application access over 5G networks.

Moreover, 5G is going to be a pervasive network to cater to diverse use cases with varied users and stakeholders. From private 5G for enterprise-specific networks to general networks for providing connectivity for public health emergency services, mass transport signaling, massive machine-type communications, and enhanced mobile broadband to the end-users, 5G become to be the de facto underlying infrastructure

in the near future. Thus the study of the security of such a network and that of QUIC-5G interactions for a faster and reliable mobile internet would benefit a wide spectrum of stakeholders such as governmental agencies, the general public, business owners, mobile operators, and mobile equipment vendors for further optimization of network and application performance.

Thus, this thesis seeks to provide insightful empirical results addressing a set of ***timely, non-trivial, and important*** research questions and remains ***beneficial*** to a wider group of technology stakeholders.

1.5 Thesis Contributions

(i) CUPS Hijacking in Mobile RAN Slicing

In emerging mobile networks, control and user plane separation (CUPS) plays a critical role in scaling the control-plane and user-plane functions independently and enables network virtualization through network slicing. However, a CUPS hijacking attack on a mobile network slicing system and the resulting network performance degradation are yet to be studied. This thesis contributes to expanding the current know-how on the impacts of CUPS hijacking attacks on an SDN-like RAN slicing system in a mobile network. In particular, our ***key contributions*** are threefold:

(a) we **model** the control plane (CP) behavior of an SDN-based RAN slicing system and quantify the impact of CUPS hijacking on the network performance in terms of a novel 'Impact Factor *I*' metric;

(b) we **prototype** a real-world use case of NHMO infrastructure, where RAN slicing plays an integral role, on the OpenAirInterface-based end-to-end mobile network running on a lab test-bed, to empirically show that CUPS hijacking is feasible under reasonable assumptions;

(c) we **analyze** the empirical results that reveal the following three-way impacts of a CUPS hijacking attack on the overall mobile network performance: *(1) cross-plane impact:* CUPS hijacking increases the RAN slice control-plane signaling delay above 2ms, the operational upper-bound of our system, to completely disrupt the control plane operations by injecting low rate denial of service (LDoS) traffic in user-plane; *(2) cross-slice impact:* CUPS hijacking degrades the throughput performance of a co-located victim slice down to 0 Mbps; *(3) in-slice impact:* a naive hijacking may completely diminish the throughput of the adversary slice itself. From the metric

formulation process, we further infer and demonstrate experimentally that a carefully crafted user-plane traffic by the adversary can regain 92% of its original user-plane packet delivery success rate while keeping other slices under the denial of service. Finally, we show that the impact factor metric I as modeled in section 4.3.2 is effective in quantifying the effects of CUPS hijacking by showing high correlations between I and network performance metrics such as throughput and latency.

(ii) Characterization of QUIC Performance over Commercial 5G

A set of 5G network performance studies have been reported in different countries, while a variety of papers provide early insights into QUIC performance. However, how QUIC performs over 5G networks in the wild remains unexplored. In this work, we make an attempt to fill the gap that exists in the literature by providing a real-life validation of QUIC's performance gains over TCP from an end-user's perspective, over a commercial 5G network.

Precisely, our **key contributions** are:

(a) For the first time, we report QUIC-5G interplay in terms of production-grade applications delivered on QUIC over a commercial 5G network in an urban setting. The results reported are fairly representative of application performance at the user end.

(b) Our empirical evidence shows that, although the existing measurement studies, either with earlier open-sourced QUIC on a wired network or collected at the global application server-end, highlight potential performance gains of QUIC, in a real-life setting at the user-end, QUIC performance without 0-RTT, is still on par with that of TCP.

(c) Our study shows, the 0-RTT feature of QUIC provides an obvious gain at the transport layer Time to The First Byte (TTFB) by reducing the median of TTFB by 70% on 5G over TCP 3-way handshake.

Chapter 2

Related Work

In this chapter, we provide a survey of recent developments in different areas of 5G security and 5G network measurements overlapping with QUIC performance analysis. This chapter on the literature survey helps in identifying the research gaps and how this thesis contributes toward closing those gaps and motivating further research.

2.1 5G Security Landscape

The 5G is all set to create a paradigm shift in the telecommunication network architecture by introducing a millimeter-wave-based customizable radio network, and a redesigned cloud-native core network to offer high-speed, low-latency mobile broadband anytime anywhere. Key enabling technologies of the 5G are Massive-MIMO, Beamforming, Software Defined Networking (SDN), Network Functions Virtualization (NFV), and Mobile Edge Computing (MEC) [84]. However, the network agility, connection to billions of smart things, and blind adoption of machine learning models and open-source software in the networked systems bring in a unique set of security threats and privacy concerns to the network and its stakeholders.

In this section, we carry out a comprehensive literature survey to provide fresh insights into the end-to-end (E2E) 5G security threat landscape, analyze the recommendations to secure the 5GS, identify evolved attack vectors, recognize the vulnerabilities, and categorize the major risks to highlight the crucial research gaps currently prevail in the way of 5G to become a trustworthy ecosystem. We also introduce the research groups working towards a safe and secure 5G network infrastructure and briefly review their lines of work [86].

Before we further explore the different aspects of the security threats on 5G, a few

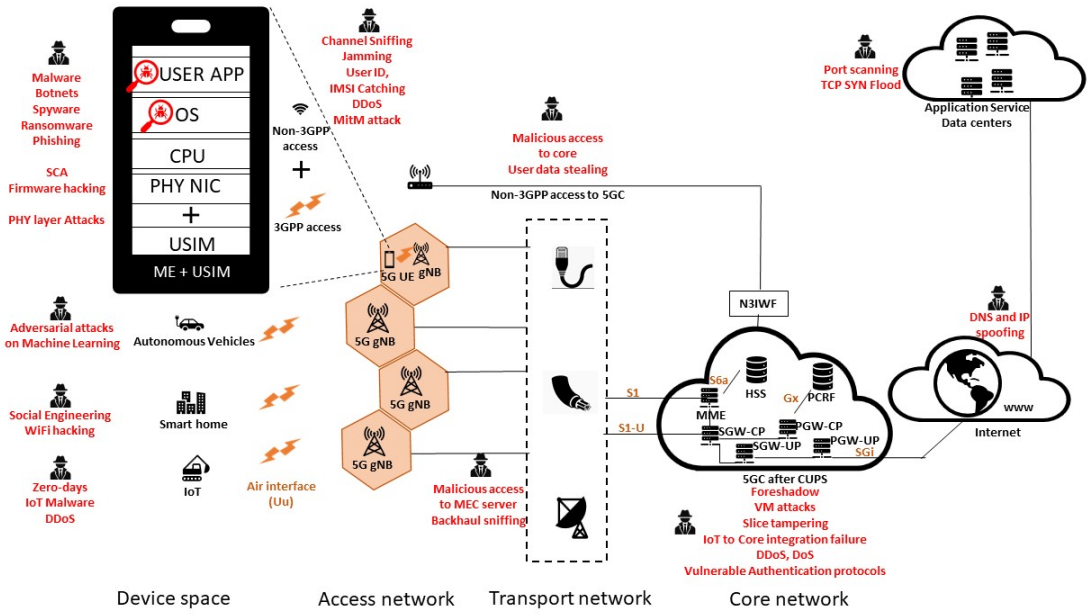


Figure 2.1: Evolved security threat landscape for 5G network

closely related technical vocabularies need to be clarified for a better understanding of readers of the thesis. Table 1.1 defines some basic cybersecurity terms which have occurred frequently in this thesis.

2.1.1 5G Network Threat Surface

Vulnerabilities of the 5G telecommunication ecosystem are spread all over the network [16]. As elucidated in Figure 2.1, the UE, the air interface between the UE and the base stations, the physical and application interfaces, and the core networks all can be quietly scanned for exploitable vulnerabilities by hackers.

Security Threats to the UE: The UE has been a soft target by hackers for quite a long time. Mobile operating systems, for instance, Android, have been reported with stealthy inter-application data leakage. In the 5G scenario, a plethora of malware along with countless phishing websites and nearby Bluetooth devices target end-users devices with vicious objectives [22]. Wu *et al.* reported an extensive analysis of android open ports that revealed vulnerability patterns in open ports of popular apps, including Skype, Instagram, and Samsung Gear [125]. Hardware chipsets and sophisticated sensor arrays can often be tampered with to steal user data, and the high-speed 5G Internet connectivity to the devices expedites the unauthorized data transfer leaving the user clueless about the harm to her

The 5G architecture accommodates various use cases and business verticals with a wide variety of SLAs. The threat surface for the use cases which includes self-driving vehicles, low-energy low bandwidth IoTs, ultra-low latency tactile Internet applications, and high bandwidth Virtual Reality (VR) applications to name a few, is unique in many ways. For instance, the machine learning models that control the behavior of an autonomous car are vulnerable to the plentiful adversarial attacks [32, 98, 117, 118]. Billions of IoT apparatus are often compromised to create a large-scale network of infected machines called botnets, which can then perform baleful activities like Distributed Denial of Service (DDoS) attacks. Shaik *et al.* demonstrated DoS attacks against UEs by exploiting the fact that certain tracking area update messages sent from the LTE network are accepted by UEs without any integrity protection [112].

Security Threats to the Air Interface: The air interface being open to everyone undergoes a massive number of maleficent attacks day in and day out. Hence, securing communication over the air is the most challenging. The Man in the Middle (MitM) attack is one of the commonly found attacks that happen in the air interface by intercepting or decrypting the victim's data surreptitiously. Another common threat is intentionally jamming the physical channel with a transmission of a high-power signal of the same frequency as that of the cellphones, which outmuscles the phone's signal causing a communication lapse. International Mobile Subscriber Identity (IMSI) catching has been a severe privacy threat to the subscribers of the previous generation telecom networks where the downlink paging messages are sniffed for side-channel data leakage and identify or track a connected device [57]. Jover reported a DoS attack against the LTE EPC using a compromised UE or another eNodeB that sends an enormous number of "attach-request" messages to the EPC to eventually chock the service [63].

However, the new 5G standard enhanced the privacy safeguards against such attacks by frequently refreshing and encrypting the user identifier called Subscription Concealed Identifier (SUCI) and dynamically adjusting the paging timings. Although fundamental security mechanisms of the entire telecommunication system depend on cryptographic algorithms and security protocols in the bit level in the digital domain, the need for another layer of secrecy of the propagation channel is often required from the information-theoretic perspective which is called the security of the physical layer.

The RAN and the mobile edge servers for 5G are crucial for ensuring ultra-low latency services. A rouge base station attack, where a false base station convinces the end-users to get attached to it instead of a bonafide gNB, can severely downgrade the

service experienced by the users. On the other hand, the transmission networks that are responsible for bringing the aggregated mobile traffic to the 5GC can be passively sniffed for possible UP or CP meta-data.

Security Threats to the Core Networks: 5G packet core which deploys and manages large-scale virtual machines for network functions virtualizations and slice management is susceptible to information leakage through covert channels between various instances of virtual machines and hence vulnerable to attacks trying to gain access to the system-level information of the whole infrastructure [83]. Since network slicing does not require strict hardware separations, shared hardware resources like cache memory can be exploited to gain side-channel access.

5G-enabled end-user devices can connect to the 5G-CN through the non-3GPP access network via a network function called N3IWF. The securely encrypted tunnel between the UE and the N3IWF is vulnerable to cyber threats if the lightweight encryption failed to encounter brute-force attacks often faced by Virtual Private Network (VPN) or Secure Shell (SSH) tunnels over insecure Internet.

It has been well known that Signalling System 7 (SS7) and Diameter protocol, which have been widely used in wireless and wired telecommunication networks for signaling in call processing, value-added services, traffic routing, and information exchange have exploitable vulnerabilities [100]. The major weakness of SS7, which has been in use until 3G network systems, had a lack of encryption. Current 4G systems use an IP-based improved signaling protocol called Diameter, where the encryption is compulsory in principle but often not practiced by the service providers within the network. 5GS specifications proposed enhancement in the procedure of accepting the resource reservation requests by using an HTTP/XML-based interface from other services. However, the PCF maintains the traditional Diameter protocol to exchange information with AF keeping the vulnerability alive in the 5GS.

Security Threats to the Application Servers: The conventional Internet and the application servers, external to the 5GS, remain vulnerable to the full range of cyber threats ranging from Domain Name Server (DNS) tunneling to DDoS attacks like Transmission Control Protocol (TCP) Synchronize (SYN) flood attack [34]. A 5G subscriber will experience poor application responsiveness or no service at all if the application server is under such an attack, no matter how ultra-high-speed 5G connection she is subscribed to [126], [119].

5G network architecture has a new set of threat vectors, which is exceptionally challenging to be tackled by 5G system designers. Network softwarization has merged

telecommunication networks with Information Technology (IT) infrastructures and thus become a great contributor to the broad range of 5G security threats that we are facing today. To mitigate the threats to 5GS, 3GPP has published recommendations for security and privacy procedures that are open to all for further inspection. In the following section, we investigate the 3GPP security recommendations to secure 5G networks.

2.1.2 Inspecting 3GPP Security Proposals

The 3GPP comprises seven Standards Development Organizations (SDO) from across the world are responsible for developing the technical specifications for mobile communication systems starting from 3G onwards. The relevant partner SDOs then convert the specifications into proper deliverables, which become the standards. According to 3GPP announcements, 5G will be spelled out in two or more phases starting from the Rel-15, which is the first phase of technical documentation of 5G [9]. Technological Specification Group-Service and System Aspects 3 (TSG-SA3), a technological specification working group within 3GPP that specifies and standardizes the security architecture, privacy requirements, and relevant protocols for 5G, has recently published specifications where they mentioned the security procedures for the 5G core and the 5G NR and in general 5G security architecture [48]. 3GPP Release 15 (R-15) depicts 5G security as a domain-based architecture that serves both SA and NSA modes of 5G operations.

- **Network access security** domain includes the set of security requirements that ensures a UE to authenticate and access the 5G network through both 3GPP and non-3GPP access networks. Access security measures are responsible for securing the most vulnerable part, the RAN, where a UE exchanges all forms of data with the network over the air interface.
- **Network domain security** is the set of security features that help the network nodes to communicate with the user and control plane data securely among themselves.
- **User domain security** ensures a user has secure access to mobile equipment.
- **Application domain security** offers reliability so that user applications and providers' applications interact securely.

- **SBA domain Security** is the set of security procedures that enable the NFs of the SBA architecture to exchange messages.
- **Visibility and configurability of security** is vital to let the user know if a security feature is turned on and in use or not.

3GPP documented the security requirements on the UE, gNB, and the core networks toward a more transparent and trustworthy ecosystem [115]. Table 2.1 highlights a few key security requirements as prescribed by the 3GPP. Security requirements of the 5G physical layer are not specified in the R-15 security proposals and security of narrowband IoT devices, vehicle-to-everything services, codec, and streaming techniques are to be specified in release 16 due in June 2020.

2.1.3 Security Research Groups and Visions

5G security enablers are proposed, developed, delivered, and maintained by a handful of technical working groups. Within the 3GPP, Technical Specification Group Service and System Aspects (TSG-SA) takes care of the overall system architecture, and service specifications based on the 3GPP recommendations and the SA3 working group is exclusively responsible for the security and privacy of 5G. Similarly, the Technical Committee (TC) Cyber is the technical group within ETSI responsible for designing reliable cybersecurity solutions and is an active contributor to 5G security and privacy deliverables. Another such group is 5G-Ensure which belongs to the 5G Infrastructure Public-Private Partnership (5GPPP), which is a joint initiative between the European Commission (EC) and the European Information and Communications Technology (ICT) industry to deliver 5G solutions. Table 4.1 briefs the security goals of the above-mentioned working groups along with other prominent technical working groups working in 5G security and privacy area and their primary responsibilities to enable a safe and trustworthy 5G network.

2.1.4 Why Securing 5G is Challenging?

Identifying potential cybersecurity threats and providing prudent solutions to resolve them is not a whole new area of research in computer networking. However, with the advent of 5GS and its pervasive revolutionary nature to connect anything anywhere, the scope of possible risk has gotten more extensive. The evolved threat space brings a fresh and unique set of challenges for cybersecurity experts in securing the 5GS and

Table 2.1: Key 5G security recommendations by 3GPP

Requirements of UE: • UE should be capable of exchanging encrypted messages with gNB using the ciphering algorithms NEA0, 128-NEA1, 128-NEA2, as presented in R-15. • The UE must support 5G-GUTI, an 80 bit long unique identifier allocated by the AMF to the UE during network registration to keep the subscriber's IMSI safe
Requirements of gNB: • gNB must support the encryption and ciphering algorithms a UE is required to have. • Commissioning gNB should be done through secured O & M systems so that no attacker can stealthily modify the gNB configuration. • Clear text key storing is strongly discouraged. A secured and trusted environment should be used to store sensitive user data and execute privacy-preserving algorithms.
Requirements of core: • Network operators are required to divide the network into several trust zones so that the sub-networks of different operators lie in the different trust zones. • Message exchanges among SBA NFs should be confidential and done only after successful authentication. • E2E core network interconnection security solution should be activated.

Table 2.2: A synopsis of 5G security research groups and deliverables

3GPP-SA3	SA WG3 is responsible for specifying security and privacy in 3GPP systems architectures and protocols and defining Cryptographic algorithms for specifications.
ETSI TC Cyber	TC Cyber works on quantum-safe cryptography, cybersecurity for national safety and ensures privacy for enterprise and individual Data.
5G-ENSURE Security	5G-ENSURE sec is responsible for realizing the 5G security specification, development, and documentation of the challenges and needs for 5G security [87].
NGMN 5G SCT	SCT focuses on security issues raised by NGMN, revision, and updation of deliverables by security work stream
ETSI ISG MEC	Industry Specification Group on Multi-access Edge Computing creates a standardized, open environment that allows flawless integration of various applications from third parties across different vendors' Multi-RAT Edge Computing (MEC) platforms.
STIX Community	STIX Community develops a structured language called STIX for a formal description of new cyber threats for efficient information sharing and analysis in a standard way.
GSMA FAS group	The FAS group analyzes the global fraud and security threat landscape and predicts risks for network operators and subscribers.
Trusted Computing Group	TCG is a non-profit organization that develops, defines, and promotes vendor-independent, open industry specifications for interoperable trusted computing platforms.

establishing it as a trustworthy ecosystem [114]. This section summarizes the vital challenges of securing a hyper-scale pervasive network like 5G.

The Convergence of IT and Telecom Infrastructure 5GS inspired the telecom network to merge with IT infrastructure by virtualizing high-level network functions traditionally carried out by dedicated physical entities. Network functions are pushed to the cloud of connected servers throughout the network. SDN, NFV, and cloud computing are the key enablers of the 5G network toward a programmable, scalable, all-IP infrastructure for voice and broadband [29]. Schneider *et al.* pointed out that due to a flat IP-based network architecture cyberattacks such as IP spoofing, and port scanning can harm the 5G network if not appropriately secured from the very beginning [108]. Moreover, in cloud-native computing architecture, user information is stored, processed, and shared by many colocated services following techniques like replication, distributed file synchronization, and controlled data flow, to name a few. However, a few of these techniques only have intrinsic security features [18, 55]. A recent contribution in network systems research explicates that the public cloud implementation of 5GC functionalities is feasible [92], although another set of security research reveals that data leaks among the virtual machines are quite possible due to the lack of hardware isolation.

Open-source Softwarization 5GS standardization is largely implemented by open-source software alliances such as OpenAirInterface [59] along with many proprietary vendor-specific implementations. Small-scale implementations often opt for open-source software implementation to keep costs low. The advantage of using an open-source software suite is that anybody from the community can review the codes and flag potential bugs. Many reviewers can report those bugs so that the vulnerabilities can be detected early for patching.

On the other hand, the National Vulnerability Database [11] publicly lists recent exploits, which could be potential targets by attackers. Another risk of using open-source software modules at an enterprise scale is sluggish development practices (such as copying code from unreliable sources) and the slow process of security patching. Hence, it is indeed a challenge to secure 5GS, which mainly adopts several open-source software [75].

Real-time E2E Security Monitoring at Scale: Securing the 5G network requires intrinsic security features to be developed as an integral part of the 5GS development process. Every module should have secure communication protocols and interfaces over which they transfer user and control data. We can not envisage 5G security as a separate layer of shield that can save the whole 5G system after its deployment [77]. This stand-alone security idea is not suitable mainly because of the highly modular and agile network architecture of 5GS, providing non-3GPP access to the 5G core network, and hosting highly sensitive public health and national security data services [70]. But there is no ultimate security by design guaranteed for a complicated, evolving, and a multi-vendor system like 5G [76]. Hence, continuous monitoring for incoming attacks is more appropriate than just securing the system during its development.

2.1.5 Can 3GPP Recommendations Solely Secure the 5GS?

5G security architecture, as proposed by 3GPP, throws light only on a selected aspect of network security procedures and privacy-preserving protocols [28]. However, cybersecurity research continues to identify weaknesses in existing defense mechanisms and new protocols [54]. In this section, we highlight a few research needs and corner cases that 3GPP security proposals left out of scope.

Government requirements: Lawful interception of user data is often a requirement in scenarios where law enforcement authority wants to track an individual for criminal offenses committed. Hence it is mandatory not to altogether abolish support for null encryption or unencrypted mode of communications from the 3GPP specifications. This requirement from a government leaves a vulnerable corner case that may potentially lead to an infrequent occasion of security threat to the networks [17].

Co-existence with vulnerable LTE network: Various security research works have recently disclosed several loopholes in LTE's privacy and authentication procedures [111]. 5G network will initially co-exists with the LTE counterpart, where a set of known vulnerabilities, including RNTI tracking and DNS traffic hijacking in layer 2, exists without being adequately addressed by current safety proposals. Thus the existing loopholes of LTE may become a potential weakness to the new 5G networks, and there is no countermeasure recommended by 3GPP.

Vendor specific hardware security: 3GPP security proposal assumes that the user-specific sensitive data such as credentials and long-term keys and authentication algorithms are stored and executed in secure hardware. However, the security auditing scheme to ensure the tamper-resistant secure hardware component falls outside the scope of the specifications [64].

Thus, 3GPP security recommendations have limitations, presumptions, and a handful of protocol corner cases that can leave 5G networks vulnerable to future cyberattacks [45]. Since services like public health, and mass transit signaling To make 5GS as a trusted mobile network ecosystem, cross-domain research endeavors are essential along with the efforts from dedicated telecom standardization bodies represented by 3GPP.

2.1.6 Current State-of-the-art

The widespread security threat landscape and shortcomings in existing systems offer exciting opportunities for cybersecurity and communication system experts to indulge in innovative research that resolves the current security flaws and establishes 5G as a ubiquitous mode of reliable communication. In this section, we highlight a few decisive aspects of 5G security and state-of-the-art systems currently available to secure the network. We also throw light on possible future research endeavors for enhanced confidence in using 5GS.

Threat models against Virtual Network Components: Recent progress in the amalgamation of the 5G networking with SDN architecture inspires security researchers to innovate threat models that can exploit the security vulnerabilities of the evolved mobile architecture. A state-of-the-art such attach exploits the inter-slice mobility of 5G networks. Inter-slice mobility that enables users' session mobility across the network slices is exploited by novel distributed slice mobility (DSM) attack [107]. ML-assisted attacks targeting 5G RAN slices are developed where the adversary consumes the 5G RAN resources by flooding fake slicing requests [113].

Recent attacks such as SSH over robust cache covert channels in the cloud, divulge the inherent weaknesses of cloud computing on which a majority of the SDN solutions depend [79]. Through inference attacks, an adversary can fingerprint SDN controllers. Information like switches flow-table size, traffic patterns, etc. can be estimated and further analysis of this information can help an adversary to launch a more

aggravated attack [67] [27] [109] [74]. The current state-of-the-art SDN security tool is ForenGuard which is capable of recording runtime dependencies and activities involving both the SDN's control and data plane and detecting any control plane attack on the SDN [122]. Dixit *et al.* developed AIM-SDN that discloses the vulnerabilities of widely used SDN datastores by novel fuzzing approaches and addressed the weaknesses [46]. Veriflow is another popular SDN-based real-time bugs checker that checks for network-scale invariant violations by scanning the forwarding rules [68]. However, a security study on the 5G virtualized network resources is needed to facilitate the secure and rapid deployment of 5G. The above-mentioned threat models recently designed against virtual networks and shared compute components constitute the background for further attacks such as CUPS hijacking, which has been examined in this thesis.

2.2 Interactions of 5G Networks with the QUIC Protocol

With ultra-low latency high speed reliable connectivity, 5G emerges as the latest cellular technology to cater to diverse business and application requirements. On the other hand, the new transport layer protocol QUIC is being adopted by major internet application providers such as Google and Facebook signifying a paradigm shift from the de-facto transport layer protocol TCP. However, the interactions between 5G, as a network data link layer, and QUIC, as a transport protocol to deliver modern web browsing, file downloading, and 4K video streaming, remain unexplored.

In this section, we survey related works in the two most relevant research areas, characterization of 5G and characterization of QUIC, to explore the existing literature and identify research gaps in understanding QUIC-5G interactions.

2.2.1 Characterization of 5G

With the advent of commercial 5G roll-out, a set of empirical studies [73, 91, 127] aim at early characterization of the network performance. These studies provide early insights into the mmWave 5G network performance as perceived by COTS smartphones and conduct performance analysis of the apps delivered over TCP/HTTP2 through 5G networks. Xu et al. analyze TCP performance over 5G under various radio conditions and handover scenarios and also measure 5G smartphone energy consumption [127]. In a series of measurement studies [95–97] Poorzare et al. investigate the TCP perfor-

mance over mmWave 5G networks. In [44, 94], the authors perform measurements in busy urban railway stations to measure 5G Quality of Service (QoS) and reveal that the 5G connectivity icon and notifications on COTS Android smartphones are often misleading and do not guarantee that the phone is attached to a 5G service.

2.2.2 Characterization of QUIC

Google developed QUIC striving to overcome the performance limitations of the de facto transport layer protocol TCP that still spurs research endeavors for possible improvements [47, 81, 131]. The initial benchmarks of QUIC performance were reported by Google in [72], claiming that QUIC improves YouTube re-buffering by 15.3-18% and lowers the search latency by 3.6–8%. However, more recent studies [26,31,33,41,65,66,80,82,104,105,110] show the empirical measurements of open-source versions of QUIC performance are often implementation dependent [37,40,90] and not always a true reflection of the protocol's behavior. In [128], the authors identify the paradox of QUIC performance optimization and empirically prove that even the performance of production-grade QUIC heavily depends on design choices such as congestion-control algorithms, and network corner cases. The authors of [20] and [129] report test-bed-based QUIC measurement studies and conclude that QUIC is advantageous over TCP in a network with limited loss and the key benefit of QUIC arises from the stream multiplexing mechanism.

Very recently, [23, 71, 88] extended the QUIC measurement studies onto wireless networks. Kunze et al. [71] comment on improving the passive measurability of heavily encrypted QUIC packets over wireless networks where network packet sniffing is essential in monitoring frequent bursty losses.

The existing body of QUIC measurement studies conducts experiments on wired networks or test beds. However, QUIC performance on commercial cellular networks in the wild is yet to be understood. This thesis complements the existing body of work by providing a first broad assessment of QUIC performance over commercial 5G from the end-users perspective.

Chapter 3

Background

This chapter provides a brief technical background useful for delving deep into the further chapters of the thesis. This thesis deals with 5G system security aspect by examining a specific type of attack, called CUPS hijacking attack, on the modern 5G RAN slicing system. Hence a background on CUPS hijacking attacks is necessary at this point, to appreciate the system-level design and empirical analysis of the results discussed in the following chapter.

Nonetheless, a more detailed comparative perspective of QUIC and TCP is also presented in this chapter. This thesis contains a detailed 5G network measurement data analysis that reveals interesting insights into the performance of QUIC and TCP in real-life settings. The background of the QUIC and TCP is hence provided in this chapter to appreciate the results discussed in Chapter 5.

3.1 CUPS Hijacking attack on 5G RAN Slicing

In modern mobile networks, CUPS refers to the logical separation between *control-plane functions* such as user authentication, connection management, etc., and *user-plane functions* such as user data traffic forwarding [38]. The key advantage of CUPS is that it enables mobile operators to scale the user plane (UP) and control plane (UP) independently of each other. CUPS is an integral part of virtualized mobile networks that aim to support diverse services through network slicing. However, the CUPS mechanism can be vulnerable to cyberattacks like CUPS hijacking, where the adversary disrupts the CP communications from the UP, obscuring the logical separation between the two planes.

The CUPS is not a 5G-specific concept, rather 5G benefits from modern SDN-

based cloud network architecture hence the threat models of the SDN architecture now become relevant to 5G. Virtualization is the key technique to ensure seamless cloud networking operation, however, it poses many security threats. One such threat is attack on the hypervisor, the critical component to achieve the vitualization.

Similarly, in 5G, SDN-like virtual network components called network slices are tailored by means of slicing hypervisor. An end-to-end 5G network slicing provisions on-demand virtualization of the RAN, the transport network, and the 5G core network. Specifically, a RAN slicing system is a radio network virtualization system that enables the dynamic allocation of radio resources and management of virtual RANs. For example, Orion, a popular RAN slicing system, constitutes with its hypervisor and the slice controllers [50].

The radio network slicing system relies on the SDN-architecture and achieves the virtualization by means of a decoupled CP from its UP. The signalling messages flowing between the hypervisor and the controllers belongs to the CP. The user traffic flows through the UP. The hypervisor sits between the CP and the UP of the system and is allocates radio physical resource blocks (PRBs), to the slices. In other words, the hypervisor provides an abstraction of the available physical resources as virtualized resources and assigns them to the controllers.

Ideally, a RAN slicing system should guarantee functional isolation between two co-located slices and logical separation between CP and UP functionalities. Which means, ideally, the control-plane functions cannot be interfered by the user-plane traffic flows and vice versa.

However, the CUPS hijacking attack tries to blur the isolation between the CP and UP by exploiting the shared physical link between the CP and the UP. A similar attack on SDN-based cloud system is reported in [116].

In the following chapter, we have shown that CUPS hijacking attack is viable in the mobile RAN slicing use case of NHMO if industry best practices for cloud security is not followed where physical isolation between CP and UP might not have ensured.

3.2 QUIC as a viable alternative to TCP

QUIC, as introduced in Chapter 1 section 1.1.2, emerges as a viable alternative to the decade-old TCP. The motivation behind designing the QUIC protocol was manifold. Security aspect being on of the prime consideration, QUIC is designed, unlike TCP, as an encrypted transport layer protocol that embrace TLS security by default. An

end-to-end encryption of QUIC ensures that the protocol header information will be immutable by the network, unlike TCP.

To be faster than the TCP in connection establishment QUIC implements novel 0-RTT and connection migration, and stream multiplexing techniques as illustrated in Chapter 1. However, to understand the ground reality if a production grade QUIC with those salient features were able to outperform the existing TCP when accessed over 4G or 5G connectivity, we presented the real-life application performance measurements in Chapter 5. To complement the results reported in the Chapter 5, we presented a brief comparison on the differences of the design philosophies of the two protocol in this chapter.

TCP was no way designed to be implemented over mobile networks and for specific applications such as streaming. However, QUIC is designed keeping in mind exclusively the nature of modern web, bursty streaming traffic, and the mobile wireless connections with high user mobility. For example, QUIC is designed differently to tackle packet-loss than TCP. QUIC embraces a monotonically-increasing packet numbers, unlike TCP, envisaging simplified retransmission calculation and RTT measurements. The stream multiplexing feature along with quick connection reestablishment envision to deliver batch traffic for web browsing faster to the application layer by eradicating the TCP's HOL blocking.

However, it remains unclear that even after considering the advanced design choices if QUIC really outperforms TCP in real-life in achieving higher application performances. To explore further and present the ground truth, a city wide 5G network measurement results are reported in the Chapter 5 that motivates further research in the measurement studies overlapping transport layer performances and 5G network performances.

Chapter 4

CUPS Hijacking in Mobile RAN Slicing

4.1 Introduction

CUPS hijacking attack, as elucidated in the previous chapter, is well-studied in the cloud-computing context. However, the impact of the CUPS hijacking on the mobile network performance is underexamined. For the first time, we aim to empirically show the effects of a CUPS hijacking attack on a mobile network's performance. In order to demonstrate the feasibility of CUPS hijacking and study its effects on network performance we develop a threat model, present quantitative modeling of a RAN slicing system, and perform experiments on a mobile network testbed.

Our work also aligns with the UK government's cloud security guidance that reinforces a complete separation between control and data planes to achieve robust operational security in vulnerability management [1] [3]. In principle, the control and data connections may share the same physical connections or not. Security agencies urge, like the guidelines provided by the UK government, to follow industry best practices and completely separate control and user planes and also the resources of different tenants. However, recent work [35] shows that there is a prevalent tendency among mobile operators to defy these recommendations. Our work shows, CUPS hijacking can be achieved in case of such security lapse due to shared control and user plane links. Subsequently, we provide empirical insights into the impacts of such an attack and means to quantify the impacts.

The CUPS mechanism, as explained in the previous chapter, is an integral part of today's LTE and 5G mobile networks. CUPS-based network architecture was borrowed from the Software-Defined Networking (SDN) and was introduced to the telecom paradigm through the 3GPP Release-14 for the LTE core networks, and later it was

adopted in the 5G service-based core network architecture by 3GPP Release-15 [6] [9].

Beyond the service-based core network functionalities, mobile network virtualization that is achieved by the network slicing technique is another key application area where CUPS plays a critical role. Network slices are independent, logically separated, virtualized set of network resources exclusively orchestrated to cater to different performance requirements of network latency, security, and throughput for various vertical industries such as eHealth, automotive, smart factories, etc. An end-to-end mobile network slicing is achieved by combining the RAN slicing, transport network slicing, and core network slicing. CUPS plays a pivotal role in a network slicing system as it allows a fully customizable CP for each slice so that a slice owner (often a mobile operator) can tailor the slice on the fly according to the service requirements (i.e., Quality of Services (QoS)).

Despite the paramount importance and timeliness of the CUPS technique in emerging mobile networking, studies on CUPS hijacking in mobile networks from a security perspective are lacking in the literature because of at least the two following challenges *(i)* most of the real-world end-to-end mobile network slicing techniques are closed-sourced and/or running on commercial networks that are not accessible to the research community; *(ii)* setting up a realistic mobile network running with modern CUPS mechanism and slicing techniques to conduct empirical studies on CUPS hijacking often demands enormous engineering efforts, human-hours, and sophisticated hardware and network configurations.

In this work, we attempt to change the status quo through a study on CUPS hijacking on mobile network slicing systems and its impacts on network performance [85]. To be reasonably detailed in our study, we restricted the scope of this endeavor only to the CUPS hijacking of a RAN slicing system instead of an end-to-end slicing system. We consider a real-world scenario of a RAN slicing system running on a typical neutral host and multi-operator indoor small cell (NHMO) as a use case (see Figure 4.1). We also set up OpenAirInterface[1] driven end-to-end mobile network testbed built on commercially available off-the-shelf (COTS) computing devices and peripherals in a controlled lab environment. Finally, we replicate the NHMO setup on the testbed to perform a thorough empirical analysis of the CUPS hijacking on the mobile network performance.

[1] https://www.openairinterface.org/

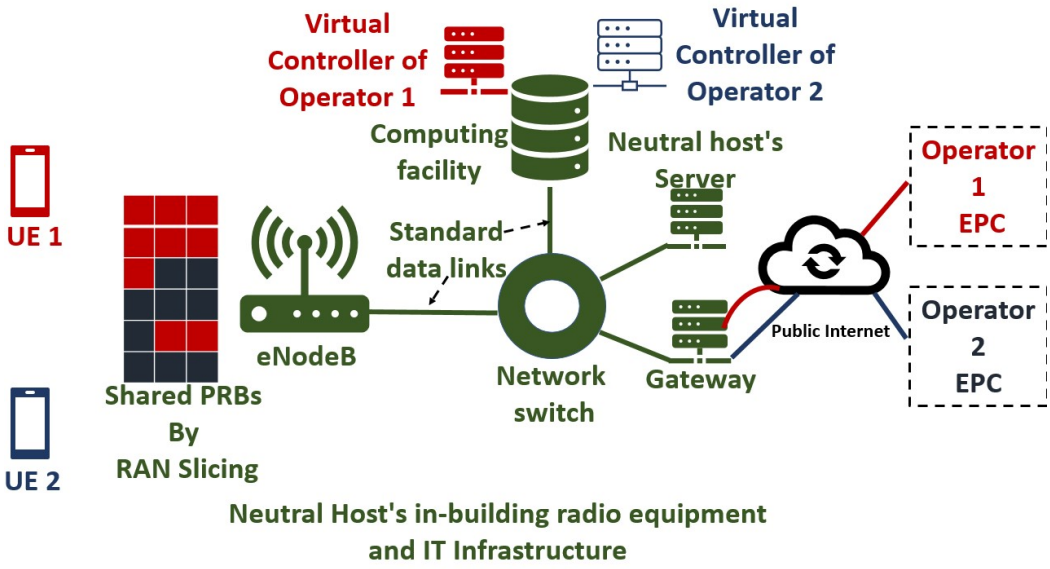


Figure 4.1: A schematic diagram of a typical neutral host and multi-operator small indoor cell use-case where the host's in-building radio equipment and IT infrastructures are shared between the operators through RAN slicing [89]. Here a two-operator scenario is illustrated.

4.2 System overview

4.2.1 RAN Slicing Primer

As the mobile network is all set to advance from the LTE/4G network which is essentially a "best-effort" network to 5G, a flexible, service-oriented mobile network, the notion of a "one size fits all" network becomes unsuitable. To host multiple services with diverse quality of service (QoS) requirements, it is required to segregate the physical infrastructure into multiple virtual networks called slices. An end-to-end network slicing envisages on-demand virtualization of all three major network segments of a mobile network: the core network, the transport network, and the RAN. Thus, RAN slicing is essentially a network virtualization technique to enable the dynamic allocation of radio resources and management of virtual RANs [42, 52, 106, 130]. Each virtualized RAN created on top of common radio hardware and network resources can be individually customized to meet different levels of QoS requirements for different slices' service level agreements (SLAs).

Orion [50] is a state-of-the-art RAN slicing system that relies on decoupled CP from its UP, thereby complying with the SDN approach of network virtualization. From a system perspective, a hypervisor is the core component of a RAN slicing sys-

tem. The hypervisor acts as the middleman between the CP and the UP of the system and is responsible for allocating radio resources, more precisely physical resource blocks (PRBs), to the slices as well as representing the available PRBs as virtualized resources to the controller. Ideally, a RAN slicing system is expected to provide a twofold assurance such as *functional isolation* between co-located slices so that no slice can affect other slices, and *separation between CP and UP functionalities* so that control-plane functions cannot be disrupted by the user-plane traffic flows and vice versa.

4.2.2 A RAN slicing use case: NHMO

In this section, we introduce NHMO, which is a highly practical use case for RAN slicing systems. The conventional RAN currently deployed worldwide assumes that the outdoor macro-cells cater "well" to the in-building consumers. However, in reality, as many as 43% of mobile subscribers face regular coverage issues inside their offices. The exponential growth in the demand for indoor mobile data usage and adoption of the higher frequency spectrum in RAN motivate a high-capacity addition to in-building mobile infrastructures. There is a wide consensus around the idea of *neutral-host* operator to achieve a faster indoor small-cells deployment. The central concept is the property owner (of commercial buildings such as urban shopping malls or offices) builds and manages the indoor radio access network, local computing, and IT facilities as a part of smart building infrastructures and offers it to multiple mobile operators to come and share for a fee [62] [2].

Figure 4.1 depicts a schematic diagram of a typical NHMO setup where the radio equipment and the associated IT infrastructure (data communication links, typically Ethernet links, network switch, computing node, server (not shared, exclusive to the neutral-host), and gateway) are owned and managed by the neutral-host. Operators can rent the infrastructure, the radio resource in terms of Physical Resource Blocks (PRBs) from the host and run their *controller* to manage their virtual RANs. The controller communicates with the eNodeB (eNB) through the slicing system's CP. A similar concept called *"Bring your own controller"* is already proposed in infrastructure-as-a-service (IaaS) clouds context to enable enterprise-level tenant sharing more flexible and affordable [121].

Sharing network resources and computing facilities has evolved as a successful business model in the IT industry in the past few years. Mobile networks are going

to experience a similar evolution by merging a majority of their network functions running on general-purpose IT infrastructure and sharing the physical infrastructures at a scale for the first time with the 5G deployment. For example, core network implementation in a public cloud facility is successfully achieved paving the way for further progress toward a merged IT and telecom infrastructure [92]. Techniques like SDN that ensures CUPS, and Network Function Virtualization (NFV) that offers easy network reconfiguration on the fly, have been adopted in designing mobile networks. Thus a significant fraction of the emerging mobile network becomes deployable on the general-purpose data network and IT infrastructures achieving unprecedented acceleration in network implementation, reconfiguration, and sharability among multiple operators.

However, sharing infrastructures among close competitors with limited physical isolation opens up unforeseen security threat that surfaces with significant concerns to the stakeholders. Moreover, the merging of telecom networks with IT networks makes the telecom network vulnerable to the attack vectors, such as an LDoS, which originally targets IT infrastructures.

In this work, we choose the NHMO as a real-world use case to implement it in a lab testbed environment to conduct the CUPS hijacking attack on it because CUPS-based RAN slicing is an integral part of the NHMO, and the use case is applicable in a dense-urban commercial indoor small-cell which is realistically replicable in a lab environment. We consider the study to be quite timely and motivating for further research to identify vulnerabilities in modern mobile network design, recognize potential drawbacks in operational practices by the mobile operators and vendors, and investigate the impact of unattended security loopholes on the stakeholders.

4.3 System Modeling

In this section, we first provide a detailed interpretation of the working principle of a state-of-the-art SDN-based RAN slicing system, Orion, deployed in the NHMO setting as depicted in Figure 4.1. Then, we quantify the impact of hijacking by introducing the impact factor metric I. We evaluate the usefulness of I in revealing the severity of CUPS hijacking on the network performance in section 4.6.

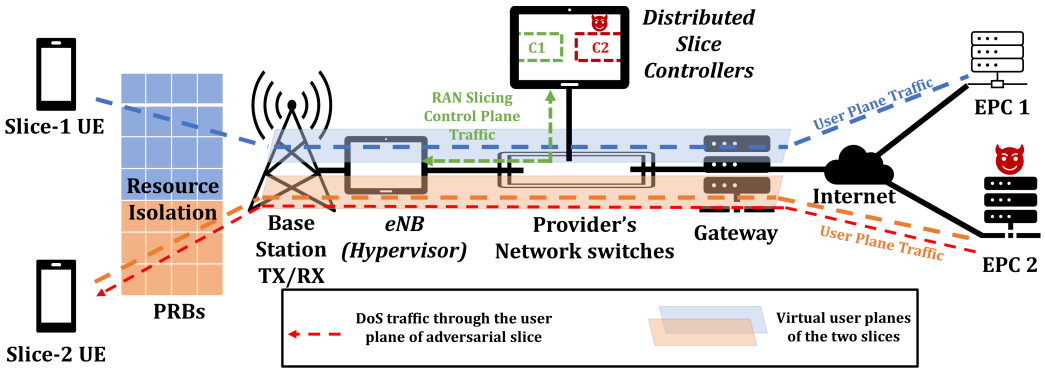


Figure 4.2: Virtual user-planes and the control-plane traffic-flows through the shared network links of an NHMO infrastructure considered as a RAN slicing use case in the study.

4.3.1 A RAN Slicing System Model

An SDN-based RAN slicing system, as introduced in Section 4.2, seeks to provide a strict isolation guarantee among slices while enabling efficient network resource sharing by the mobile operators. The hypervisor that lies on top of the physical layer is the key component of a slicing system like Orion, which we deploy in our testbed to prototype the infrastructure for the small-cell indoor NHMO use case. The Orion hypervisor is a critical component in virtualizing the RAN SDN networks. The hypervisor abstracts the underlying physical network into multiple logically isolated virtual networks each with its own RAN controller. Hence the Orion hypervisor is a classic centralized SDN hypervisor capable of running on a common computing node [30].

The hypervisor joins the isolated RAN slices to the PRBs and the shared physical infrastructure (i.e. eNB), by offering a virtual abstraction of the underlying PRBs and the UP states and accordingly updating any state changes in the physical UP by mapping virtual to physical resources. The hypervisor allocates the PRBs among slices after virtualizing them. In the NHMO setting, the hypervisor is part of the network infrastructure provider's software suite that facilitates RAN slicing.

Figure 4.2 shows the two RAN slices with user-planes and the control-plane traffic flows through the network switches, the eNB (where the hypervisor is running), the network server systems (where two RAN slicing controllers are running), and the gateway through which the user-plane traffic flows to and from the EPCs (evolved packet core networks) of the two operators. In our experimental setup on the test bed, as described in section 4.5, we considered a similar RAN slicing architecture with two RAN slices running on the NHMO infrastructure.

In our system, the mobile operators (e.g., mobile virtual network operators (MVNOs) or business verticals) can realize their RAN slices by instantiating virtual eNBs on top of the hypervisor. For each of the virtual eNB, a virtual control plane is created to manage the user-plane state which is virtually exposed to the eNB by the hypervisor. The virtual control plane of a RAN slice is essentially the RAN slice controller running separately on a remote computing facility (as shown in Figure 4.2) and is capable to tailor the functionality as if the slice was operating on its own dedicated infrastructure.

The slice controller communicates with the hypervisor in near real-time through logically independent communication channels. To be precise, here the computing response time from the controller is less than 2 ms as empirically found in [50], thus the term is near real-time, and not an instant or real-time requirement. However, any significant time delay introduced in the channel has critical implications on the performance of the hypervisor, because it can delay resource allocation, which is expected to happen in near real-time following the traffic demand of the slice.

4.3.2 Formulation of Impact Factor

In an NHMO network, there can be N RAN slicing controllers each owned by a mobile operator. A single controller may tailor a slice for a virtual mobile operator or for distinct QoS requirements. For instance, Services such as web browsing or non-interactive traffic are delay-tolerant and can be grouped together to be catered by a network slice. On the other hand, voice, video, or interactive services such as telnet or SSH can be catered by a different network slice for prioritized traffic to achieve the QoS [36].

In Figure 4.2, we have a set of two, (N=2), distributed virtual RAN slicing controllers, C_i, where i is in $\{1,..., N\}$. Each virtual RAN slicing controller has a control-plane signaling delay, δ_i, that is the latency control signals experienced through the network links between the hypervisor and the controller C_i.

To define the average controllers' response time (CRT), we consider the cumulative control-plane request traffic $\Sigma\phi$, that flows in between the hypervisor and a RAN slicing controller. The average CRT of the ith slice controller is denoted by σ_{ci}, where i is in $\{1,..., N\}$. Let's assume the ith slice controller's maximum capacity of processing requests from the hypervisor at a time is $\gamma_{i,max}$ and the cumulative request traffic flows to a particular controller is $\Sigma\phi$. We model the distributed controllers as M/M/1, and assume that the flow requests obey the Poisson distribution. Therefore, the average

Table 4.1: Definitions of the symbols used

N	Number of RAN Slicing Controllers
δ_i	Control-plane signalling delay which is the latency control signals experienced through the network links in between the hypervisor and the i-th controller C_i
δ_{break}	The upper limit of the control-plane signalling delay of a controller above which the hypervisor and the controller fall out of sync
σ_{ci}	The average CRT of the ith slice controller C_i
$\gamma_{i,max}$	The ith slice controller's maximum capacity of processing requests from the hypervisor at a time
$\Sigma\phi$	The cumulative control-plane request traffic that flows in between the hypervisor and a RAN slicing controller
Δ_{avg}	The average control-plane signalling delay between the hypervisor and the set of distributed slice controllers
$CPFW$	Control Plane Functional Window of a controller C_i. It is the total time taken by a control-plane packet to arrive in the controller from the hypervisor, to be processed by the controller, and the reply from the controller to reach to the hypervisor. CPFW under no attack is the above-mentioned cumulative time window when no attack is in place.
$CPFW_{under\_attack}$	CPFW under the influence of the DoS attack through the shared data plane
$CPFW_{max}$	The maximum value of the CPFW of a controller C_i. It represents a special case of $CPFW_{under\_attack}$. and is the target benchmark to achieve for successfully launching a CUPS hijacking attack.
I	The impact factor of the CUPS hijacking attack on the network. it is defined as the ration of $CPFW_{under\_attack}$ and $CPFW_{no\_attack}$.

CRT σ_{ci} of C_i, given its maximum capacity $\gamma_{i,max}$ and load in terms of the cumulative flows $\Sigma\phi$, can be defined using the Little's theory [120] as follows

$$\sigma_{ci} = \frac{1}{\gamma_{i,max} - \Sigma\phi} \tag{4.1}$$

On the other hand, the average control-plane signalling delay Δ_{avg} in between the hypervisor and the set of N distributed controllers, can be formulated as

$$\Delta_{avg} = \frac{1}{N}\Sigma_{i=1}^{N}\delta_i \tag{4.2}$$

In an ideal RAN slicing system, average CRT σ_{ci} is negligible because, usually in a small-cell setup, the maximum serving capacity $\gamma_{i,max}$ is designed larger than the peak cumulative flows of requests $\Sigma\phi$. However, a distributed deployment of the RAN slice controllers over a complex network may incur wide-spread control-plane signaling delays, δ_i, that in turn, can give rise to the average control-plane signaling delay Δ_{avg}.

In our setup, two controllers are co-located to each other are deployed away from the hypervisor (eNB). We empirically found Δ_{avg} is 0.3 ms in the experimental setup during normal operation (no attack) of the network. We also empirically validate the claim made in [50] that if the control-plane signaling delay δ_i of a controller C_i becomes 2ms or more, the Orion hypervisor and the controllers fall out of sync impacting the network services drastically. We denote this upper limit of the control-plane signaling delay as δ_{break}.

From the above formulation, we can infer if the user-plane traffic flow can introduce a delay of more than δ_{break} in the control-plane flows then the control-plane functions of the slicing systems are disrupted leading to a CUPS hijacking that violates the CP and UP separation.

We define the Control Plane Functional Window (CPFW) of a controller C_i as the total time taken by a control-plane packet to arrive in the controller from the hypervisor, to be processed by the controller, and the reply from the controller to reach to the hypervisor. Then from equation (5.1) and (4.2), CPFW (under no attack) can be defined as follows

$$CPFW = \delta_i + \sigma_{ci} \tag{4.3}$$

where δ_i denotes the control-plane signaling delay between the hypervisor and the controller C_i. In our setup, the two controllers are co-located to each other and

placed at an equal distance from the hypervisor. Hence, Δ_{avg}=δ_i. Clearly, CPFW (under attack) can be defined as $CPFW_{under\_attack} = \delta_{i,under\_attack} + \sigma_{ci,under\_attack}$. The maximum value of the CPFW of a controller C_i, $CPFW_{max}$, represents a special case of $CPFW_{under\_attack}$. $CPFW_{max}$, is the target benchmark to achieve for successfully launching a CUPS hijacking attack. When defining $CPFW_{max}$, we assume that σ_{ci} as a low-magnitude constant for an NHMO setup because of the design consideration of the high capacity $\gamma_{i,max}$. In this particular work, we assume that $\gamma_{i,max}$ cannot be compromised by an attacker, albeit, that might not be the case in a different threat model than ours which is explained in the next section. Thus, the upper bound of CPFW, $CPFW_{max}$, can be defined in terms δ_{break} as follows

$$CPFW_{max} = \delta_{break} + \sigma_{ci,under\_attack} \tag{4.4}$$

From equation (4.3) and (4.4), we define the impact factor of the CUPS hijacking on a RAN slicing controller C_i, I_{ci}, as a function of CPFW as follows

$$I_{ci} = \frac{CPFW_{under\_attack}}{CPFW_{no\_attack}} \tag{4.5}$$

Since, in our setup $\sigma_{ci} \approx 0$, hence, I_{ci}=($\delta_{i,under\_attack}$/δ_{i,no_attack}). Similarly, $I_{ci,max}$= (δ_{break}/δ_{i,no_attack}). As mentioned, we empirically find $\delta_{break} = 2ms$ and $\delta_{i,no\_attack} = 0.3ms$, in our setup $I_{ci,max} = 6.66$. When calculating the value of I, if found $\delta_{i,under\_attack}$ is greater than or equal to δ_{break}, then .$\delta_{i,under\_attack}$ can be replaced by δ_{break} because beyond δ_{break} the slicing system control plane is impacted the most and lost the CP and UP separations denoting the upper bound of the impacts. It is a good idea to represent the impact factor I in a normalized form. In this work, we use the widely adopted min-max normalization method and scaled the values of I so that I $[0,1]$.

CPFW and I provide important insights into CUPS hijacking. For instance, CUPS hijacking can be achieved when $\delta_i \approx \delta_{break}$. On the other hand, if σ_{ci} is sufficiently large then the control-plane functionality of C_i would also be disrupted. However, in a mobile network, σ_{ci} is sufficiently low to cater to the peak volume of the control request.

The formulation of *CPFW* and *I* helps us to infer that, CUPS hijacking of RAN slicing system can be possible by at least two possible ways (i) by $\delta_i \approx \delta_{break}$, for instance, a possible way to achieve this is to inject a large volume of DoS traffic in user plane to sufficiently congest the shared physical link so that the control-plane traffic suffers high delay (greater than δ_{break}); (ii) by affecting the σ_{ci}, for instance, a pos-

sible way to achieve this is a side-channel attack from a co-located adversary slice controller that generates impersonated control requests in a sufficiently larger volume than the victim controller is designed for to handle at a time, i.e., $\gamma_{i,max}$. However, in order to demonstrate CUPS hijacking, in this work, we do not consider a side-channel attack that compromises the capacity, $\gamma_{i,max}$, but a DoS attack that elevates $\delta_i \approx \delta_{break}$ satisfying the pre-condition of CUPS hijacking. In the following section 4.4, we design an attack model to achieve the pre-condition of CUPS hijacking.

A non-obvious insight is drawn from the CPFW is that an adversarial controller, C_k can achieve an intelligent CUPS hijacking if it can keep its own $\delta_k < \delta_{break}$ but can sufficiently increase the δ_is of the rest of the *(N-1)* slice controllers beyond the δ_{break}. In section 4.6, we leverage this insight to improve the primary threat model and in section 4.6 we empirically validate this notion to achieve the CUPS hijacking.

We also elaborate on the impact factor in section 4.6. We show that the impact factor is correlated with the two key network performance metrics - throughput and latency, proving I as a suitable metric to measure the severity of the CUPS hijacking on the network performance.

4.4 Threat Model and CUPS Hijacking Attack

Based on our modeling, we conjecture that under certain conditions the RAN slicing system becomes vulnerable to CUPS hijacking attack. To test our hypothesis we carry out the empirical study of CUPS hijacking attack in an SDN-based RAN slicing system by designing the following threat model with a reasonable set of assumptions.

Attack Scenario: We propose that in an NHMO setting, an operator with a legit tenancy can turn himself into an adversary and launch a CUPS hijacking attack by increasing the control-plane signaling delay by injecting malicious traffic in the user plane. Precisely, the adversary exploits the lack of physical isolation in the SDN-based RAN slicing system's CP and UP traffic flows to disrupt the network performance.

In the context of NHMO, RAN slicing is an integral part of the mobile network implementation enabling adaptive sharing of the common infrastructure among multiple operators. Due to shared physical links among operators and between CP and UP traffics, an abrupt increase of a slice's UP traffic has impacts on the CP traffic of the same slice and the other slices as well. An adversary slice controller can craft its CP traffic that causes cross-plane, cross-slice, and in-slice impacts because of the shared physical links. Thus an adversary can compromise the logical separation between the

control and user plane as modeled in the section 4.3.

In the experimental setup of this work, we consider the neutral host as having two operators sharing the radio equipment and the neutral host's in-building IT infrastructure (Figure 4.2). The shared physical data links, switches, and gateway cater to both operators.

Goal of the Attacker: The goals of the attacker, a malicious operator among the mobile operators, are threefold: (i) Fingerprint the shared links and have an estimate of how much UP traffic has a significant impact on control-plane signaling delay, δ_i, in between the eNB/hypervisor and the controller. (ii) Launch an LDoS attack through the UP to bring network functionalities under complete disruption. (iii) Keep its own traffic flowing while putting other slices under the denial of service through CUPS hijacking.

Challenges for the Attacker: The adversary aims to launch an LDoS to choke the shared links in the small cell data network. Thus it needs to estimate how much attack traffic can choke the link capacity. Therefore, the attacker needs to (1) learn the shared link capacity (2) since a continuous DoS kills traffic of the adversary slice as well, the adversary needs to let its own traffic flow while hindering others. Our chosen method of attack addresses both the challenges to successfully launch a CUPS hijacking attack exploiting the vulnerability that emerges from the multi-operator infrastructure sharing.

Assumptions: We assume that the attacker is a legal tenant in the small cell NHMO setup and can control its RAN slice through its controller. The controllers of the tenant mobile operators are deployed in a distributed computing facility connected to the hypervisor (eNB) through shared communication links. However, we do NOT assume that the attacker has compromised or has privileged root access to any of the neutral host's servers or equipment. The threat model also assumes the CP and UP traffics share common communication links in a distributed deployment of SDN-based RAN slicing systems on NHMO infrastructure which are reasonably realistic assumptions in the context of the NHMO use case.

Attack Approach: The attacker achieves the goals by the following three-step attack approach. The attacker first tries to estimate the approximate capacity of the shared physical links (especially the link between the eNB and the slice controllers) by injecting LDoS flows with an increasing rate in its user plane until it observes disruptive control-plane signaling delay. Once the attacker learns the shared link capacity, it launches the LDoS attack with the learned data rate to completely choke the network

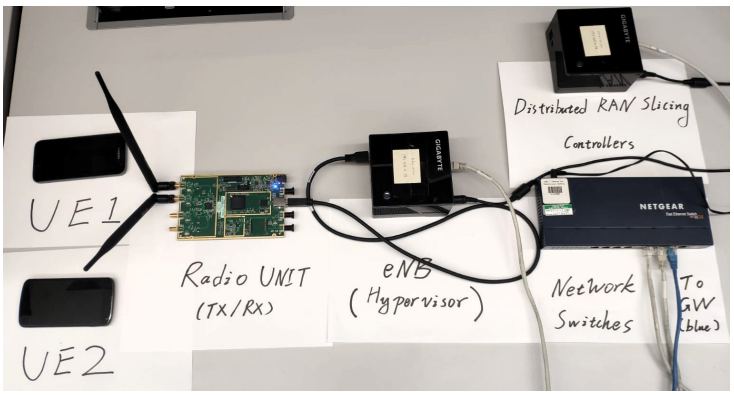


Figure 4.3: A photograph of the RAN segment of the lab testbed.

bandwidth. The attacker injects LDoS traffic in its UP to disrupt the CP functionalities of the overall RAN slicing system by leveraging the condition of CUPS hijacking modeled in section 4.3. Finally, the attacker carefully controls the attack traffic flow and its legit user traffic flow to regain its network performance while keeping other slices under a denial of service as pointed out during the formulation of the impact factor I, in section 4.3.

While designing an LDoS flow the attacker can customize its burst duration, burst magnitude, and inter-burst gap period. We choose the burst duration as 200 ms, the inter-burst period as 300 ms., and the maximum rate found as 100 Mbps which completely chokes the link in our setup. In a commercial deployment, the maximum speed may go higher than 1Gbps, given high-speed Ethernet connections between the small cell network nodes.

(i) Link capacity estimation with increasing rate of LDoS: In order to estimate the share link capacity, the attacker injects streams of an increasing rate of LDoS traffic in its user plane (in our experiment we used 10 Mbps, 30Mbps, 60Mbps, and 100Mbps) each of duration of 2 seconds in the network and measures its control-plane signaling delay. To get a finer estimation of the link capacity, the attacker can choose an off-peak time of the day when all the user traffic is expected to be very low or negligible. In an urban commercial smart building, it can be midnight. With increasing LDoS traffic through the UP, the RAN slicing CP packets experience an elevated delay due to shared links between CP and UP traffic. At a certain LDoS rate, the attacker observes that its controller has fallen out of sync with the hypervisor ($\delta_i >= \delta_{break}$) as discussed in section 4.3. The attacker may estimate this LDoS rate (in our case it is 100Mbps) as the link capacity and prepare for aggressive CUPS hijacking.

(ii) CUPS hijacking with constant rate LDoS: The attacker now launches the

LDoS flow through its user plane at the learned rate roughly equals the shared link capacity to completely disrupt the CP functionality. However, the increased CP signaling delay, as well as the congested shared user planes, lead to a denial of services for all the RAN slices including the attacker's own.

(iii) Regaining attacker's slice-performance under LDoS: The constant rate LDoS attack kills the UP and CP traffics of all the slices indiscriminately. So, the attacker should craft an intelligent UP traffic flow scheme to regain the network performance for itself but still keep other slices under the DoS attack. As outlined in section 4.3, we try to realize the notion of an intelligent CUPS hijacking such that the adversary keeps its own $\delta_k < \delta_{break}$ but sufficiently elevate the δ_i of the rest of the ***(N-1)*** slice controllers above δ_{break}. To achieve intelligent CUPS hijacking, the attacker now sends its legit user traffic only during the interburst gap period of the LDoS injection and holds it off during the burst duration.

A CUPS hijacking on the RAN slicing system may be achieved by a completely different or more robust threat model, but the above-described threat model sufficiently achieves the objective of enabling understanding of the three-way impacts of a CUPS hijacking attack on a RAN slicing system by allowing to empirically answer the following set of research questions (RQs):

RQ1. How much time does the adversary need to launch the CUPS hijacking attack?

RQ2. What are the impacts of a successful CUPS hijacking launched by a malicious slice owner on the network performance?

RQ3. How effective the attack is in achieving the adversary's objective of retaining its own traffic intact while diminishing traffic from other co-located slices?

We analyze the empirical results in section 4.6 to answer the above RQs. Thus, we demonstrate the viability of CUPS hijacking by quantitative modeling, threat model design, and performing experiments on a mobile network testbed.

4.5 Testbed Implementation

The testbed consists of two GIGABYTE Intel-based small form factor PCs (i7-4770R CPU @ 3.20GHz, 4GB of RAM) running Ubuntu 18.04 with a low-latency kernel. The two PCs are running the RAN slicing system, Orion, and its components (hypervisor and slice controllers). The core part of the network is deployed as virtual machines (VMs) on an additional Intel-based machine (i5-3230M CPU @ 2.60GHz, 16GB of

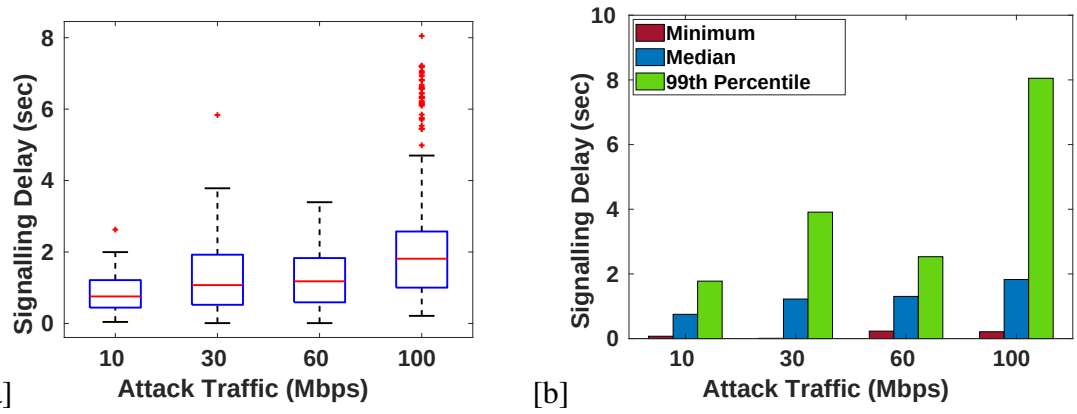


Figure 4.4: ***Cross-plane impact:*** [a] Control-plane signaling delay, δ_i, experienced by the adversary RAN slicing controller during link capacity estimation using LDoS with increasing rate [b] 99% of the control-signaling packets experience maximum 8 seconds of delay under 100Mbps attack traffic leading to a CUPS hijacking because the controllers fall out-of-sync with the hypervisor.

RAM). We leverage OpenAirInterface open-source EPC implementation where two mobile network operators are deployed on two different virtual machines. The two VMs are running Ubuntu 16.08 with Linux 4.7.7 kernel optimized for real-time operation i.e., disabled CPU C-states, low-latency Linux kernel, and with disabled CPU frequency scaling. For the front-end radio unit, we use Ettus USRP B210 Software-Defined Radio (SDR) board equipped with two omnidirectional 2.45GHz antennas. We use Samsung Galaxy Note 4 and Huawei E3372 LTE dongle as UEs. We set the default bandwidth of the 5MHz spectrum to be shared between the two slices in LTE band 7. The Orion base station is configured to use single input single output (SISO) transmission mode, which for 5MHz spectrum can provide up to a maximum throughput of 16Mbps. Figure 4.3 shows a real picture of the RAN segment of the testbed implementation of the NHMO use case in the lab environment.

4.6 Results

In this section, we report the experimental results and make an attempt to answer the **RQs**, framed at the end of section 4.4, by analyzing empirical evidences.

RQ1: Exploitation time by the attacker. We demonstrate with a limited number of attempts an adversary can guess the bandwidth of the shared network links and subsequently launch the LDoS attack leading to CUPS hijacking. We used 100 Mbps ethernet cable as the shared link capacity that an attacker can roughly estimate in a

period of 8 seconds in our threat model. Given the fact that the prediction of a 100Mbps link capacity is achieved in under 10 seconds, the capacity for higher speed links (1 Gbps or higher) is within a few minutes.

RQ2: Impacts of CUPS hijacking. We present a comprehensive ***threefold impacts*** of a CUPS hijacking attack on the network behavior as follows.

Cross-plane impact: Figure 3.4[a] shows that data traffic in the user-plane can affect the control-plane signaling delay, δ_i, resulting in a complete disruption of the slicing system's control operations when the condition, $\delta_i \mathcal{E} \delta_{break}$, is met. The attack introduces a delay in the RAN slice CP, above the operational upper bound of our system δ_{break}=2ms, to disrupt the control plane operations by injecting LDoS traffic in the UP. Figure 3.4[b], shows that 99% of the control-signal packets experience maximum of 8 seconds (far greater than $\delta_{break} = 2ms$) of delay under 100Mbps LDoS leading the RAN slicing controllers to fall out-of-sync with the hypervisor. This empirical evidence proves the feasibility of CUPS hijacking as modeled in section 4.3.2.

Cross-slice impact: Isolation between any two RAN slices is a critical condition of a robust deployment of a slicing system so that no adversary can impact cross-slice network performance and encroach on virtualized network resources that are not allocated to it [19]. However, Figure 3.5[a] shows, a UE attached to the co-located victim slice starts experiencing poor user-throughput performances with an increasing rate of LDoS injection by the adversary in the user plane of the adversary slice. In this study, the LDoS lasts for 8 seconds, and the LDoS rate increases from 10Mbps to 30 Mbps, to 60Mbps, and to 100Mbps in every 2 seconds. As shown in Figure 3.5[b], under an 8-second-long continuous 100Mbps (the estimated maximum link capacity) LDoS injection from the adversary in its user plane, the same UE experiences a near-zero user-throughput during the entire attack period. Figures 5[a] and 5[b] prove that not only cross-plane but also the cross-slice functional isolation that a RAN slicing system guarantees can be voided by a successful CUPS hijacking.

However, the characteristics of gradual degradation perceived by the UE of a co-located slice can be used for early detection of the presence of an adversary in the pool of controllers.

in-slice impact: As shown in Figure 3.6[a], the user throughput of a UE attached to the adversary slice gradually decreases with time as the shared communication link accumulates LDoS traffic. The network performance significantly aggravates when the attacker injects 100Mbps LDoS through its UP leading to the CUPS hijacking. Figure 3.6[b] shows the user throughput of the same UE, completely diminished under

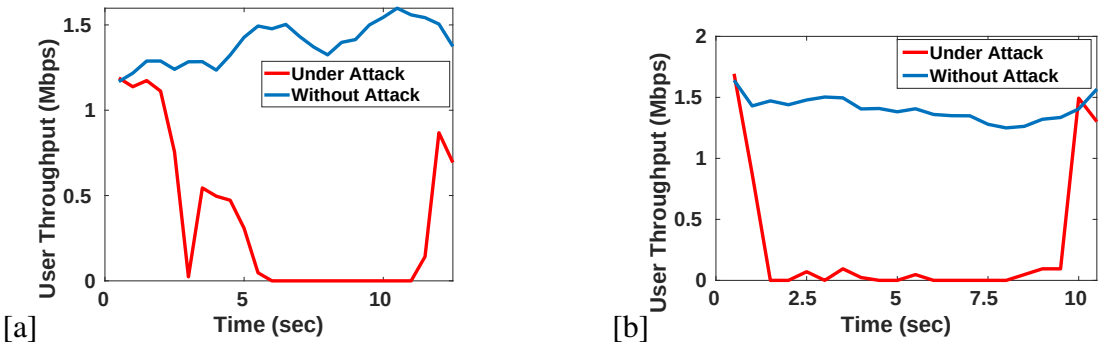


Figure 4.5: ***Cross-slice impact:*** [a] User throughput of the UE attached to the co-located victim slice under the increasing rate of LDoS and under no attack. [b] User throughput of the UE attached to the co-located victim slice under 100 Mbps LDoS and under no attack.

a continuous 100Mbps LDoS injection. Figures 6[a] and 6[b] show the throughput performance degradation perceived by a UE attached to the adversary slice under the increasing rate of LDoS and a continuous 100Mbps LDoS.

From figures 5 and 6 we infer that a naive CUPS hijacking attack has *similar diminishing impacts* on the network throughput in both co-located and the adversary slices motivating the adversary to regain its own slice performance.

RQ3. Attacker's Gain: The attacker can now leverage the insights learned in the section 4.3 to exclusively regain the performance of its own slice. To achieve efficient CUPS hijacking, as proposed in the threat model in section 4.4, the attacker sends its legit user traffic only during the interburst gap period of the LDoS injection and holds it off during the burst duration. As Figure 3.7[a] shows, instead of sending the user traffic under the influence of LDoS in a naive manner, if the adversary carefully crafts the user traffic to be sent only during the inter-burst gap period of the LDoS, it can regain his user-plane packet delivery success ratio upto 92% from 0% under a CUPS hijacking. We name this scheme intelligent DoS, reflecting an efficient CUPS hijacking.

Impact on the user-plane latency: Figure 3.8[a] shows the UP round-trip time (RTT) latencies perceived by a UE attached to the network under the attack and under no attack. The x-axis is the RTT in seconds computed from the TCP acknowledgments in the Wireshark traces, and the y-axis is the cumulative distribution of the RTT. We observe that under no attack, the RTT of a UE always remains below 0.25 seconds. However, under the CUPS hijacking attack, the RTT exceeds 0.25 seconds with a high probability of 75%.

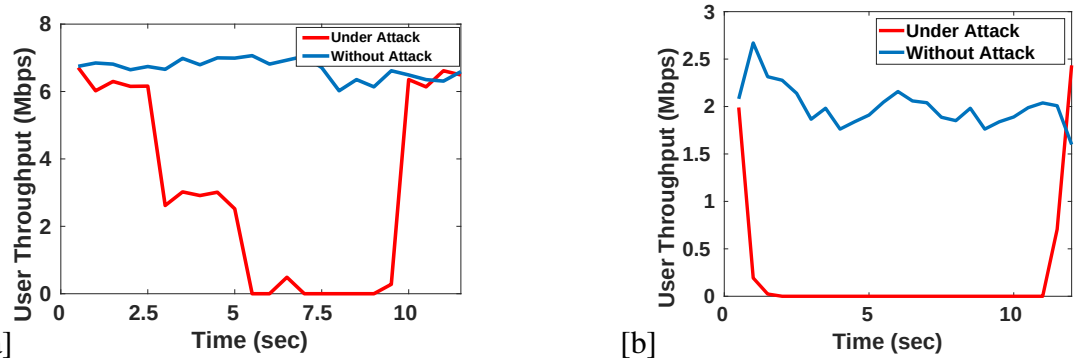


Figure 4.6: ***In-slice impact:*** [a] User throughput of the UE attached to the adversary slice under the increasing rate of LDoS and under no attack. [b] User throughput of the UE attached to the adversary slice under 100Mbps LDoS and under no attack.

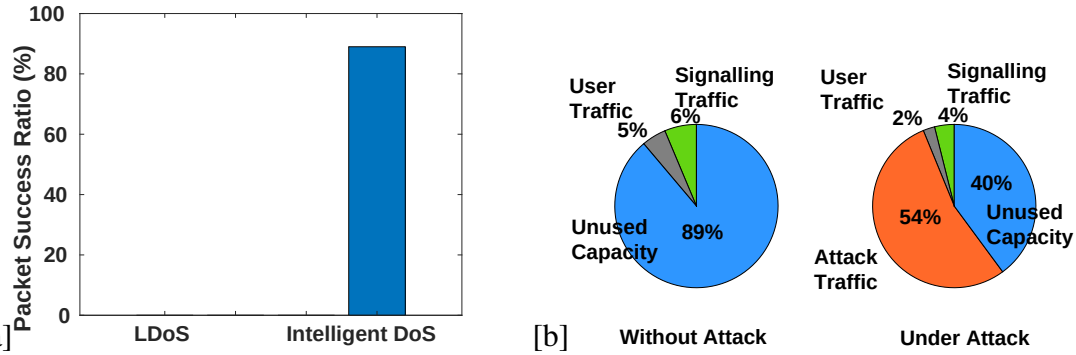
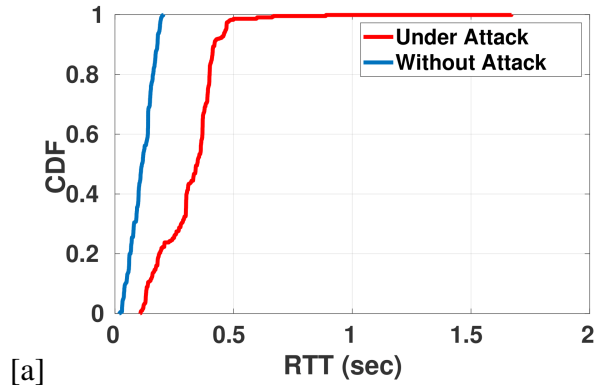


Figure 4.7: [a] User packet transmission success ratios under naive LDoS and intelligent DoS [b] Port utilization of the eNB without attack and under 100Mbps LDoS

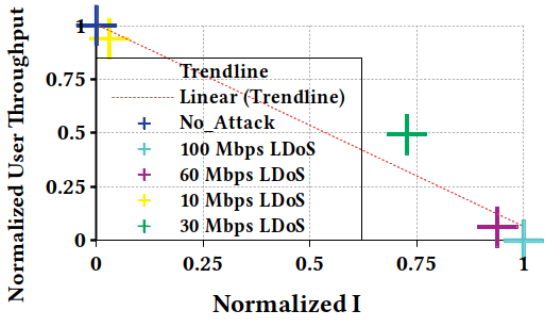
Impact on UE attachment procedure: We performed another experiment to investigate if the CUPS hijacking has any impact on the mobile network control plane and we found that under the 100Mbps LDoS injection, a UE could not attach to the network. The UE, in our setup, keep sending attach request to the EPC every 10.611 seconds. On the other hand, under no attack, a UE can get connected to the network for an average of 0.63 seconds.

eNB port utilization: Figure 3.7[b] elucidates the network port utilization of the eNB, under attack and without attack. Although, under attack, the eNB port is under-utilized still the user experiences zero throughput due to the TCP retransmission time out (RTO) mechanism that further delays the packet sending because too many acknowledgments have been missed due to the ongoing network congestion.

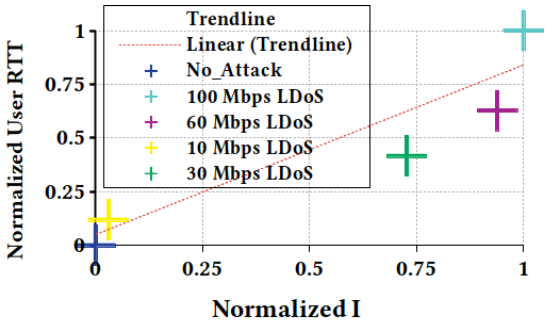
Correlation between *I* and network performances: Section 4.3.2 introduces the impact factor metric *I*, to quantify the severity of the CUPS hijacking attack. Figures 3.8[b] and 3.8[c] show the correlations between the impact factor I and the UP throughput and RTT of a UE attached to the network. We scaled the parameters by



[a]



[b]



[c]

Figure 4.8: [a] Distributions of the user-plane latencies (RTT) during the attack and under no attack. The x-axis is the round-trip time in seconds. The CUPS hijacking increases the UP latency. [b] and [c] Correlate the normalized impact factor *I* with the normalized user throughput [b] and normalized user RTT [c]. Different colors represent experiments with different rates of LDoS traffic. We see a negative (positive) correlation between *I* and user throughput (user RTT) that proves the impact factor *I*, is a useful metric to quantify the network performance degradation due to CUPS hijacking attack.

the min-max normalization to the interval of [0,1]. Each point in the plot represents experiments under the different rates of LDoS injection. We see a positive correlation between the impact factor and the network RTT latency and a negative correlation between the impact factor and the user throughput, which proves the impact factor I is a pertinent metric to quantify the severity of a CUPS hijacking attack.

4.7 Discussion

Although our work empirically investigates the effects of CUPS hijacking attacks on the performances of LTE/4G mobile networks equipped with SDN-based RAN slicing systems in the context of NHMO use case, the research findings have implications on the emerging 5G network design implementations. For instance, the notion of split eNB protocol stacks of ORAN[2] supports shared links between the Distributed Units (DU) and Centralized Unit (CU) CP and UP through F1-c and F1-u interfaces respectively, hence cross-plane impacts of a possible CUPS hijacking remains relevant in the 5G RAN sharing in O-RAN context [93] if implemented defying the security best practices.

Traffic isolation techniques as countermeasures: In the commercial space, operators can consider leveraging tagged VLAN and VPNs to ensure strict isolation between CP and UP [39, 132]. For the transport network, soft isolation through tunneling is necessary on top of physical isolation to ensure QoS among various services. However, our demonstration of CUPS hijacking shows, in a shared IT infrastructure, where the data communication links are having limited bandwidth capacity, soft-isolation alone not be able to mitigate the impacts of the CUPS hijacking unless a VPN with strict bandwidth policing is adopted.

To summarize, in this work for the first time, we empirically study the CUPS hijacking on the mobile RAN slicing system. Our work provides two key insights: (i) *need for traffic isolation:* without physical isolation or strict bandwidth-restricted soft-isolation (such as VPN service with constricted traffic regulations) between CP and UP traffic and among the UP traffics of co-located slices, an SDN-based RAN slicing system running on shared infrastructure becomes vulnerable to CUPS hijacking attacks; (ii) *feasibility of CUPS hijacking on RAN slicing:* Although RAN slicing system guarantees CUPS and inter-slice functional isolation, under the CUPS hijacking attack, both promises are voided with serious implications on network performance.

[2]https://www.o-ran.org/

A general principle of CUPS hijacking is known in cloud-computing systems with shared infrastructure, however, this work presents interesting findings like a failure of UE attachment or eNB network port utilization under CUPS hijacking which are specific to RAN slicing systems. We believe, our results stimulate the cybersecurity research community to take on further research endeavors envisioning secure slicing in 5G and impact RAN slicing design considerations when deployed on shared infrastructures. The results of this study drive us to initiate follow-up research in the direction of securing emerging RAN architecture such as O-RAN. Our endeavor highlights the need for systematic studies of security vulnerabilities in modern mobile network deployment on public clouds and multi-operator scenarios through designing sophisticated threat models involving side-channel attacks and CUPS hijacking to ensure a robust mobile network architecture that is rapidly merging with IT infrastructures.

Chapter 5

Characterization of QUIC Performance over Commercial 5G

5.1 Introduction

QUIC is being adopted as a transport protocol by major Internet companies such as Google, Facebook, etc. A plethora of research endeavors has examined 5G network performance, while other initiatives have provided early insights into application performance over QUIC when using wired and wireless networks, such as Wi-Fi, following measurement collection and analysis. However, the interaction between QUIC and 5G, the two major underpinning technologies to bring mobile broadband to the end-users, remains underexplored.

The 3GPP, the mobile network standardization body, has already published release-16 (R-16) standardizing the first 5G mobile networks and is in the process of finalizing R-17, R-18, and R-19 to fully mature the 5G technology by 2024 [13]. On the other hand, the IETF has released the QUIC version 1 (RFC 9000) specification [14]. Thus, this is high time to investigate how the new transport layer protocol QUIC performs with the newly rolled out 5G infrastructure.

The existing performance evaluation and characterization approaches for QUIC are either limited to wired networks or open-source implementations, which are not optimized for production-grade enterprise servers. Thus the existing results are not truly representative of real-life scenarios.

The lack of such an extensive study in the literature can be explained by at least the following three challenges:

(i) The varied heterogeneity of the commercial 5G ecosystem in terms of spectrum

usage and network coverage makes it non-trivial to zero in on a proper experimental setup for the measurement study.

(ii) Maintaining consistency among long-term measurement collections requires automated scripts to control a wide variety of different applications and should be deployed in different use-case scenarios such as under mobility and static environments. This makes an empirical study time-consuming and challenging.

(iii) Apart from the technical challenges, there exists non-technical obstacles too. For instance, the amount of time and travel required to collect field measurements to include both mobile and indoor static experimental conditions and insure diverse geolocations is significant and often expensive to carry out.

Thus, an empirical study to evaluate production-grade QUIC performance over commercial 5G in a real-life urban setup with different measurement scenarios is timely, non-trivial, and important. Since 5G is going to bring mobile broadband to millions of people across the globe and QUIC has already been adopted by the major Internet players, the indicative insights provided by the QUIC performance characterization study over commercial 5G will be beneficial to a wider spectrum of technology stakeholders such as mobile operators, research communities and the application developers. Our work undertakes an empirical characterization of production-grade QUIC over 5G in the wild, using TCP performance as a benchmark.

We summarise the **key findings** of our study below:

(i) Although measurements from the server side previously claimed that QUIC could improve the performances of applications such as Google search and YouTube streaming, our measurements from the user side in a real-life urban scenario with various applications show that QUIC performance without 0-RTT is similar to that of TCP.

(ii) From a transport layer's perspective, we see that QUIC's 0-RTT feature can reduce the median TTFB by 70% on 5G compared with the TCP 3-way handshake when loading a web page like the YouTube homepage.

(iii) Experiments on accessing Top Tranco 105 QUIC-enabled websites reveal that 61 out of these 105 websites load faster on TCP over 5G and the difference in page load time (PLT) on QUIC vs TCP for 100 out of 105 websites is less than 500 ms.

Table 5.1: A synopsis of the collected measurements

Cells Covered [Indoor static Measurement Collection]	8 [4 5G cells and 4 4G/LTE cells]
Duration of 4K YouTube Streaming	500+ minutes
Total amount of file size downloaded from Google Drive	250+ GB
Number of Google searches performed	400
Number of times Google maps accessed	392
The approximate distance traveled for mobility data collection	55+ km

Thus, this thesis provides insightful empirical evidence that, although QUIC is designed to outperform TCP for modern applications, in real-life application performance as perceived by an end user of 5G networks is on par with that of TCP.

5.2 Measurement Study Design

We collected measurements in the city of Edinburgh over the commercial 5G and 4G/LTE networks deployed by the O2 mobile operator. We present the comparative performance metrics from both the transport protocols over both the network access technologies, 4G/LTE and 5G, to provide a complete picture of performance differences from the users' perspective.

Currently, in the UK O2 operates 5G in NSA mode with n78 TDD (NR-ARFCN: 633696) sub-6GHz band. In our setup, the maximum number of downlink physical resource blocks (PRBs, 3GPP 38.104) available were 106 [12].

We used commercial off-the-shelf Samsung Galaxy Note20 5G mobile phones running Android version 11 (kernel version 4.19.87-22307827) as a modem. A Lenovo ThinkPad with 32 GB of memory and 1TB of SSD storage driven by a 2.8 GHz 11th generation Intel octa-core i7-1165G7 processor running the Microsoft Windows 10 operating systems was used to run applications accessing the Internet through the Samsung Galaxy Note20 5G mobile phone. Google Chrome version 104, a 64-bit browser was used to access the services considered for this study.

We focused on the following widely used applications to measure the performance of the transport protocol: 4K video streaming from YouTube, web browsing, Google Search, Google Maps, and File download from Google Drive. In our setup, the browser cache was disabled and the browser was opened in incognito or private mode before visiting every website.

To be reasonably comprehensive in collecting measurements, we considered both

static and mobile environments. In static environments, we chose 4 cells in 4 different locations in the city. The cells are geographically well apart from each other. To collect measurements in mobile settings, we traveled by tram through the city. Table 5.1 summarizes the measurement collection statistics. Most of the measurements were collected during business hours and in the evening when the highest network usage is expected.

5.3 Results

In this section, we report our findings following the performance measurement collection along with the metrics we used to quantify the performance and key takeaways for each application.

5.3.1 Google Search

Google search is available on both QUIC and TCP. The original QUIC paper shows that Google search has improved over QUIC [72]. We attempt to characterize the improvement from the user end.

Methodology: To measure the performance of Google Search, we automate the search operation from the Google Chrome browser with a python script using selenium[1] web browser automation tool. We searched a predefined set of 25 different search items in an iterative manner. As mentioned in Table 5.1, we performed 400 Google searches on TCP and QUIC over 5G and 4G from static environments.

Metric: We used PLT to assess the performance of the Google Search application. PLT is measured as the time elapsed since the search request is sent from the browser until the search result return page loads complete. In practice, we measure PLT using the Selenium web driver and Performance Timing API, which captures the page loading events and the amount of time elapsed in each event from the time the request was made. Among the various events, the navigationStart attribute is used to return the start time of the PLT and domComplete attribute is used to mark the end of the loading. domComplete refers that the parsing of the received web page is complete and all the chunks of data required to render the web page are downloaded. The smaller the PLT the better the performance.

[1] https://www.selenium.dev/

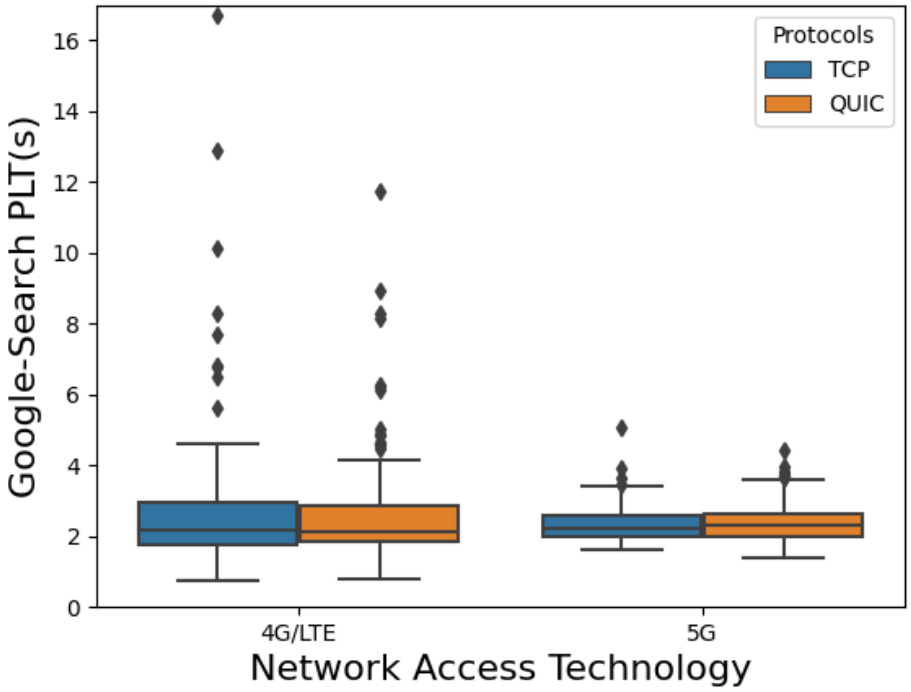


Figure 5.1: Comparison of PLTs of Google Search using QUIC and TCP over 4G/LTE and 5G networks.

Takeaway: Google search performance in terms of PLT measured from the user-end remains largely the same regardless of the wireless access technology (5G or 4G/LTE) and transport protocol (TCP or QUIC) in static environments, as illustrated in Figure 5.1. The p-value calculated for PLT of Google Search application on 5G was 0.2511 which is too high to accept an alternative hypothesis that Google Search performs better over QUIC than TCP. However, this could still be true, but in our measurement collection, we do not have the evidence to support such claims. A more scaled-up measurement collection would facilitate the experiments and thus left as the future work of this research as discussed in the concluding chapter of the thesis. However, the performance similarity of Google search on QUIC and TCP is mainly because the search application is highly optimized, leaving little room for performance improvement from transport or data link layers. The comparison of Google search on TCP and QUIC was fair because in both cases the amount of data transferred over the network remains similar (1.1MB for TCP and 1MB for QUIC) and the total number of HTTP requests sent over the network is also similar (108 for TCP and 109 for QUIC) because we performed the same set of searches over TCP and QUIC.

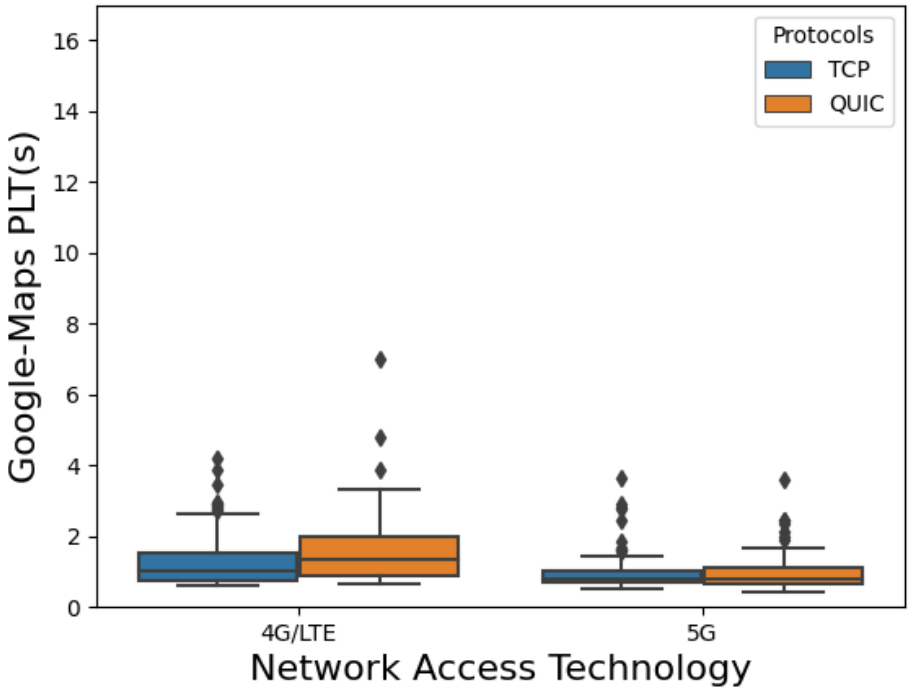


Figure 5.2: Comparisons of PLTs of Google Maps location rendition on QUIC and TCP over 4G/LTE and 5G networks

5.3.2 Google Maps

Google Maps is also available on both QUIC and TCP. How the Maps application performs is often overlooked in the literature. We attempt to characterize this performance using QUIC over the mobile network from the user end.

Methodology: To measure the performance of Google Maps, we automate the Maps operation from the Google Chrome browser using again the Selenium web browser automation tool, searching a set of 25 different locations in an iterative manner. As mentioned in Table 5.1, we have iteratively performed more than 390 Google Maps renditions on TCP and QUIC over 5G and 4G in static environments.

Metric: We used PLT, as described in 5.3.1, again to measure the performance for Google Maps rendering on the browser.

PLT is measured as the time elapsed since the location map request is sent from the browser until the Maps page loading completes.

Takeaway: Google Maps location rendition on chrome browser in terms of PLT measured from the user-end shows a median performance difference of 320 ms between 4G/LTE and 5G access technologies, as illustrated in Figure 5.2. Over 5G the

median difference between PLT performance between QUIC and TCP was similar (25ms) whereas over 4G/LTE this figure raises to 295ms. The p-value calculated for PLT of Google Maps application on 5G was 0.4516 which is too high to accept an alternative hypothesis that Google Maps performs better over QUIC than TCP. However, this could still be true, but we do not have the evidence to support that claim.

5.3.3 File Download from Google Drive

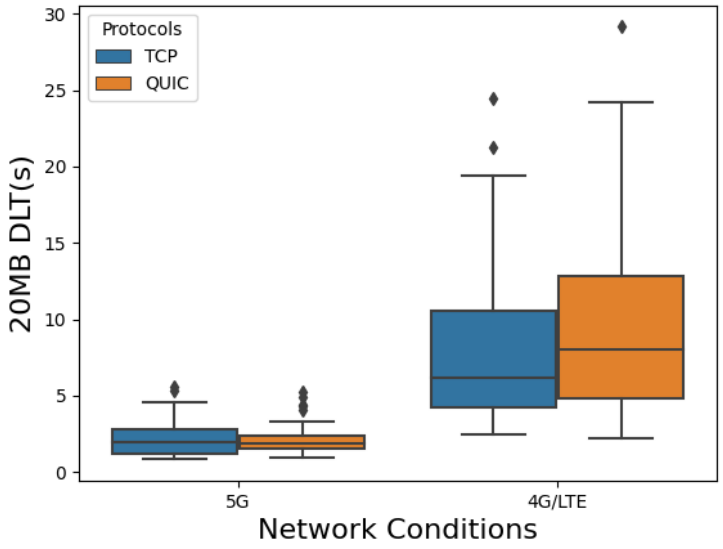
Next we checked the performance of file downloading from Google Drive on QUIC and TCP over both 5G and 4G/LTE, in a static environment. To quantify the performance, we used the following metrics.

- Download Time (DLT): Download time is measured as the time difference between the start of file download and the end time when the download is finished. DLT is measured in our setup in seconds and the smaller the DLT, the better the performance.
- Goodput: Goodput is defined as the amount of payload data, excluding the packet headers, transferred successfully from the Google Drive server to the client Laptop in a unit of time. Goodput can be simply calculated as file size divided by the DLT. Goodput is measured in Mbps and the greater the Goodput, the better the performance.

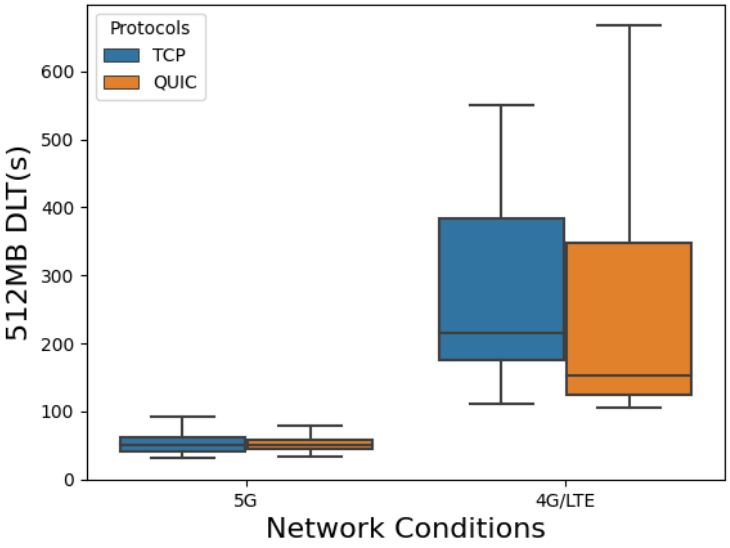
Methodology: We stored in the Google Drive copies of two files of sizes 20 MB and respectively 512 MB, and used a Python script with the Selenium automation tool to iteratively download the files on TCP and QUIC over both 5G and 4G/LTE. We measured the DLT, as defined above, and calculated the goodput of the downloading iteration by the script. To be noted, file downloading happens purely either on QUIC or TCP. When accessing the Google Drive on QUIC, the initial TCP handshaking and subsequent QUIC handover as an alt-serv happen at the beginning, before the client requests the file download from the Drive.

Takeaway: Irrespective of the transport protocols in use, small file (20 MB) and large file (512 MB) downloads perform similarly over a specific access technology.

Figures 5.3[a] and 5.3[b] show the comparisons of the DLTs of small files and the large files downloaded on TCP and QUIC over 5G and 4G/LTE technology. The differences in DLT for a given file size remain similar irrespective of the transport protocol used. However, DLT over 5G is significantly lower than that over 4G/LTE

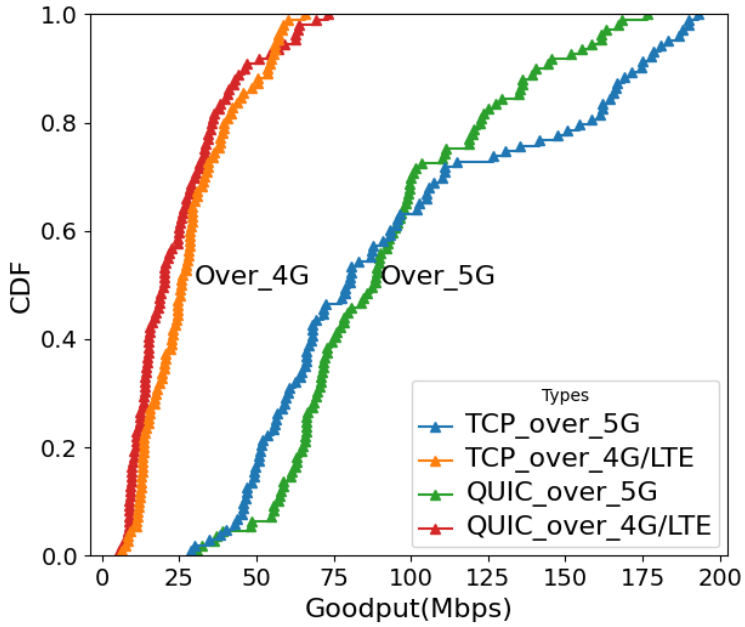


[a]

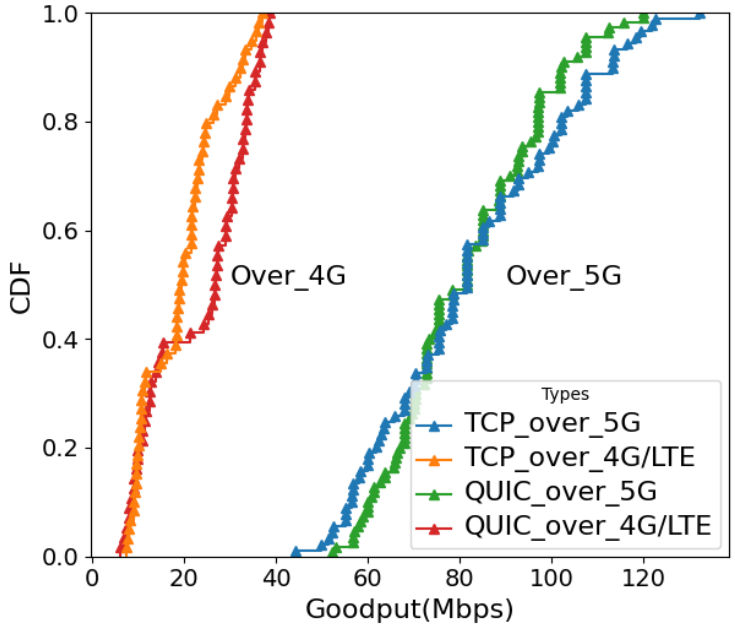


[b]

Figure 5.3: [a] Comparisons of DLTs of small file size (20 MB) downloaded on TCP and QUIC over 5G and 4G/LTE [b] Comparisons of DLTs of large file size (512 MB) downloaded on TCP and QUIC over 5G and 4G/LTE



[a]



[b]

Figure 5.4: Goodput comparison when downloading [a] small (20 MB) and [b] large (512 MB) files using TCP and QUIC over 5G and 4G/LTE.

because of the higher bandwidth available. Figures 5.4[a] and 5.4[b] show that the median goodput for a given file size (small and large) remains consistent irrespective of the transport protocol used. The median difference in the bandwidth of 5G and 4G/LTE was close to 75Mbps in our setup. The results contain a fairly large sample size of 528 file download instances overall.

5.3.4 Fairness in Multi-flow

Traffic flow competition is often encountered in practice. Ideally, the available network bandwidth should be fairly or equally shared among all the competing flows. In general, the end-host congestion control algorithm attempts to maximize utility per flow by ensuring that every flow through the same bottleneck link gets a fair share of the available bandwidth. To understand how a QUIC flow competes with another simultaneous TCP flow, we study the fairness when downloading from Google Drive.

The key objective of this experiment is to provide an initial comparison of how QUIC and TCP flows share the network bandwidth. Currently, TCP and QUIC are both in use widely, however, we do not yet have empirical evidence of how they co-exist.

There is a misconception that since QUIC is built on top of best-effort UDP, QUIC would outperform TCP because during network congestion the TCP congestion control reduces the packet transmission rate while UDP does not. However, this is misleading because QUIC is a stand-alone transport-layer protocol with reliable congestion control mechanisms just like TCP. Hence we present a comparison of two flows both from Google's server and on the same mobile operator.

However, our experimental result in its current form does not unpack the individual congestion control algorithms but rather measures their performance as they are in use by the production-grade Google servers.

We leave controlled lab measurement experiments to empirically test QUIC and TCP performance under various congestion control algorithms for future work as discussed in the concluding chapter.

Metric: To quantify the fairness between the QUIC and TCP flows we used the widely-known Jain's Fairness Index (JFI) [61]. The Jain's fairness index is defined as

$$FI_{Jain} = \frac{\left(\Sigma R_c\right)^2}{C\Sigma R_c^2} \tag{5.1}$$

where C is the set of all flows, i.e., in our case, $C = 1, 2$. R is the throughput for the c-th

connection. The values of JFI is bounded by (0, 1] where a unity JFI value indicates the ideal or highest fairness.

Methodology: We stored a copy of a very large size (1GB) file on Google Drive and used two different Python scripts to connect to the drive, one over QUIC and another over TCP, and start the download of the file simultaneously. Packet capturing at the client end through Wireshark was turned on. We iteratively performed this experiment 10 times under the 5G connection and calculated the JFI for each time window of 10 seconds.

Takeway: Downloading a large file over QUIC, when competing with another TCP flow, (both from Google's server and under the same 5G mobile operator) finishes earlier than file download over TCP. Figure 5.5[a] shows the distribution of the JFI over the 10-second windows. Clearly, JFI is mostly less than 0.9, proving that the QUIC flow captures most of the bandwidth share. Figure 5.5[b] shows the average throughput of the two competing flows. Note that we cannot access the congestion control algorithms at the end-host, yet we can see that currently, the TCP and QUIC deployed by Google might not be fair to each other [123].

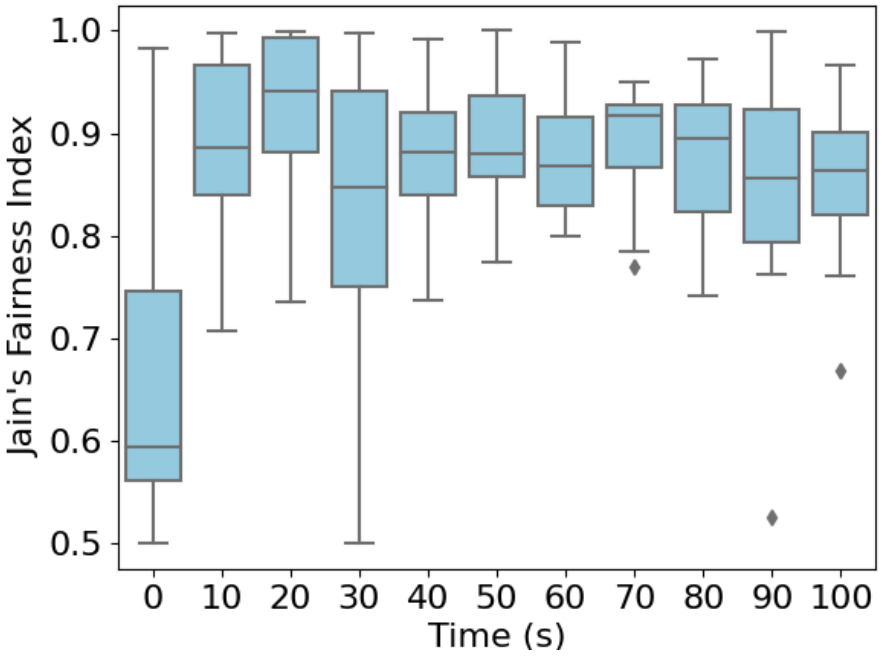
5.3.5 4K Video Streaming on YouTube

YouTube also adopted QUIC as their alternate transport protocol. Hence, we considered 4K video streaming on YouTube as one of the applications to measure the performance of QUIC. We considered the following metrics to quantify the performance:

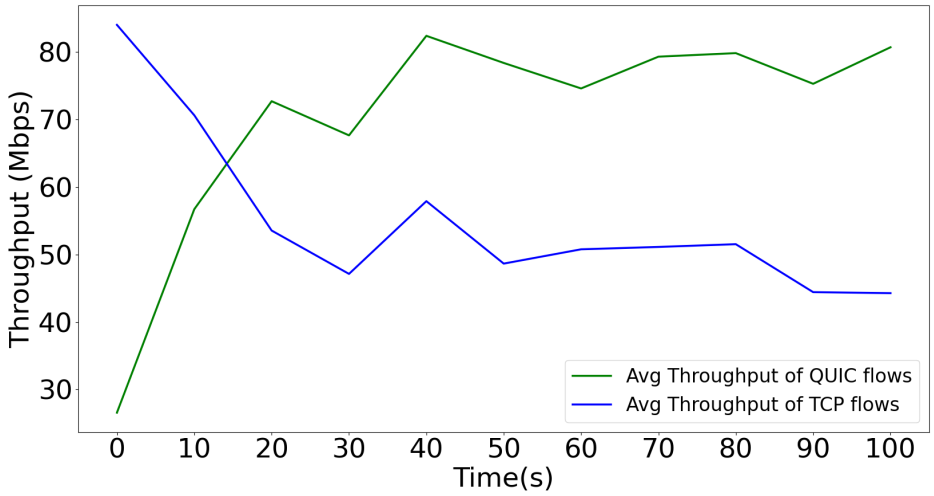
- Start-up delay: the difference between times from when the video playing request is made by the browser and when the video starts playing, measured in seconds. The lower the start-up delay the better the streaming experience performance.

- Video Quality: YouTube has a variety of video quality options available to play a requested video: 2160p (4K), 1440p (2K), 1080p, 720p, 480p, 360p. Although our study mainly focuses on the 4K quality of video streaming, during the measurements taken in mobile settings, we noticed a few video quality degradation instances (down to 720p). Modern streaming services choose dynamic adaptive streaming over HTTP (DASH) to modify the video quality on the fly and avoid playback stalling.

- Buffering events: The YouTube player can have 6 different states: unstarted,



[a]



[b]

Figure 5.5: [a] Distributions of JFIs computed in every 10 seconds of windows on samples from two competing flows of QUIC and TCP [b] Average throughput of two competing flows of QUIC and TCP

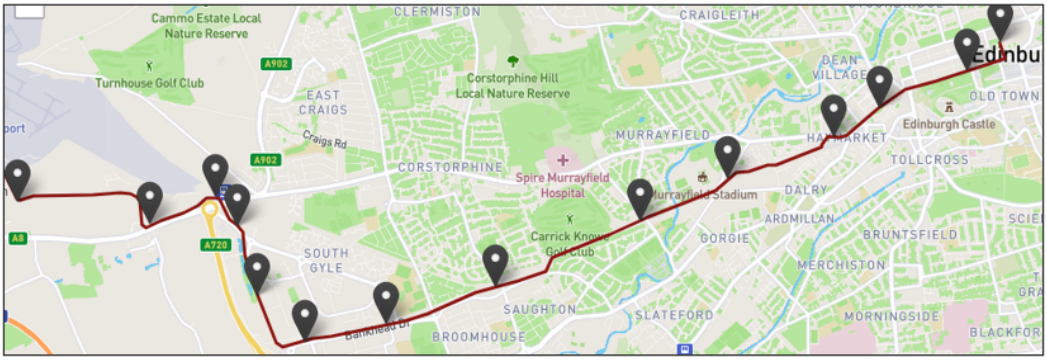


Figure 5.6: The tram route (approximately 14 km in one direction) on which the measurements with mobility have been performed in both directions. The pointers on the route (data from Edinburgh trams) show the tram stops during measurement collection.

ended, playing, paused, buffering, and video cued. Once the video started playing, it attains the playing state, and then after any changes in its state, until the complete video has been played (state-ended) is termed as *"buffering events"*. In our experiment, we keep track of the player states to monitor if any buffering events occurred.

- Fraction of content loaded: A requested video is downloaded in multiple chunks as the playback progresses. The fraction of content loaded is defined as the fraction of the entire video, that has been downloaded, buffered, and is ready for playback. The complete loading of the whole video content is marked with 1. A steady progression in the fraction of content loaded over time signifies smooth playback at the user's end.

  Any buffering events (maybe user-initiated or caused by any other reasons such as due to poor connections) have direct correlations with the fraction of content loaded. For instance, If the user paused the video the fraction of content may not be loaded further until the playback is resumed. If due to a lost network connection, the next video chunks are not downloaded, the "fraction of content loading" would not progress. As the user keeps playing back on the video, eventually the playback state gives raise to a *"buffering event"* until the playback resumes after the connection is reestablished.

Methodology: We automated the request to play a set of videos from the Chrome browser. For automation and streaming performance measurement purposes, we used Python scripts and Selenium browser automation tool along with YouTube IFrame

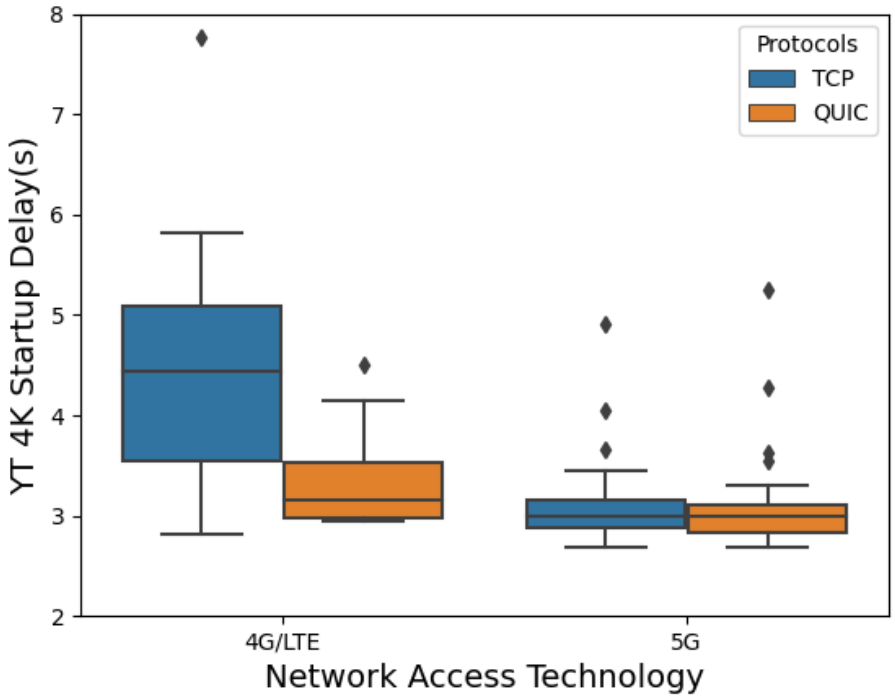


Figure 5.7: Comparisons of start-up delays of 4K YouTube streaming on QUIC and TCP over 4G/LTE and 5G networks

Player API.[2] The 4K videos were 5-minute long and we played both over TCP and QUIC.

For ***static environment*** measurement collection, we played the videos on both 5G and 4G connections in an iterative manner through automated scripts and captured the performance metrics. For the ***mobile environment***, we accessed only 5G coverage and played the video on both TCP and QUIC from within the tram compartment, to measure the streaming performance. Figure 5.6 depicts the tram route on which the mobile measurement collections were carried out in both directions multiple times over TCP and QUIC. As mentioned in Table 5.1, the approximate distance traveled was over 55 km and on average, more than 50 handovers in each direction of the trip were observed.

Takeaway1: In a static environment, YouTube start-up delay on QUIC shows similar behavior to that over TCP when accessed over 5G, as seen in Figure 5.7. However, the p-value calculated for the collected start-up delay data on 5G was 0.4567 which is too high to accept an alternative hypothesis that YouTube performs better on QUIC than TCP. This could still be true, but in our measurement collection, we do not have

[2]https://developers.google.com/youtube/iframe_api_reference

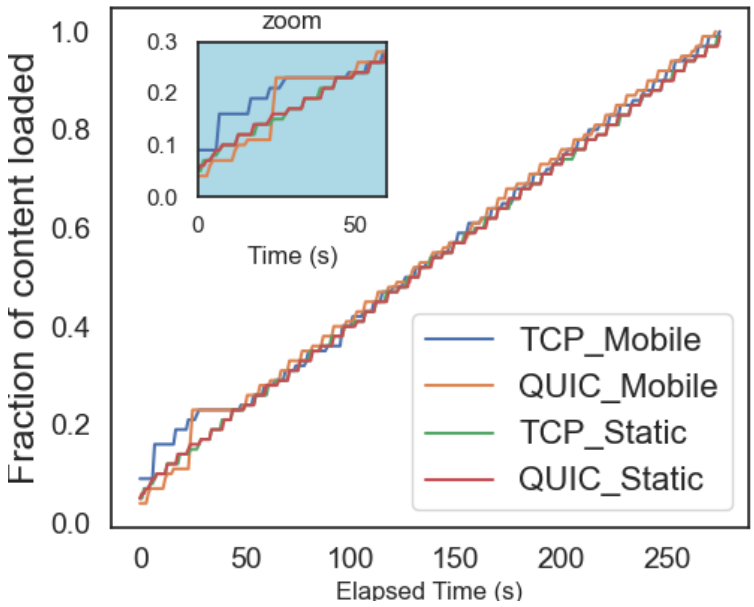


Figure 5.8: Comparisons of the fraction of content loaded over TCP and QUIC in the mobile and static environment for a given video accessed over 5G network. A further zoomed inset is provided, looking at the initial 60 seconds of the comparisons where QUIC streaming experienced a difference in video quality and rapid handovers than TCP.

sufficient evidence to support that claim.

The figure also shows that the 4K start-up delay on QUIC is lower than that of TCP over 4G/LTE. However, we noticed that on 4G/LTE, 4K start-up was often not possible (only 12 QUIC and 14 TCP instances when the video started playing on 4K).

Takeaway2: With mobility under 5G coverage, YouTube streaming start-up on 4K was a rare event on both TCP and QUIC. We noticed only one 4K start-up on each protocol over 5G during our 55 km of experimental tour with streaming measurement collections. Playback starts in lower quality on mobility and then upgrades itself to 4K.

Takeaway3: YouTube streaming adapts the video quality to the connection quality, as seen in Figure 5.8[a] where we observe the fraction of content loaded on QUIC and TCP over 5G under static and mobile conditions.

Takeaway4: Streaming over QUIC and TCP survives the mobile handovers equally efficiently due to its bursty nature of content loading. Figure 5.8[a] shows the handover has no visible impact on the fraction of content loaded as time progressed. In this figure, the number of handovers are captured only within the duration of a single video

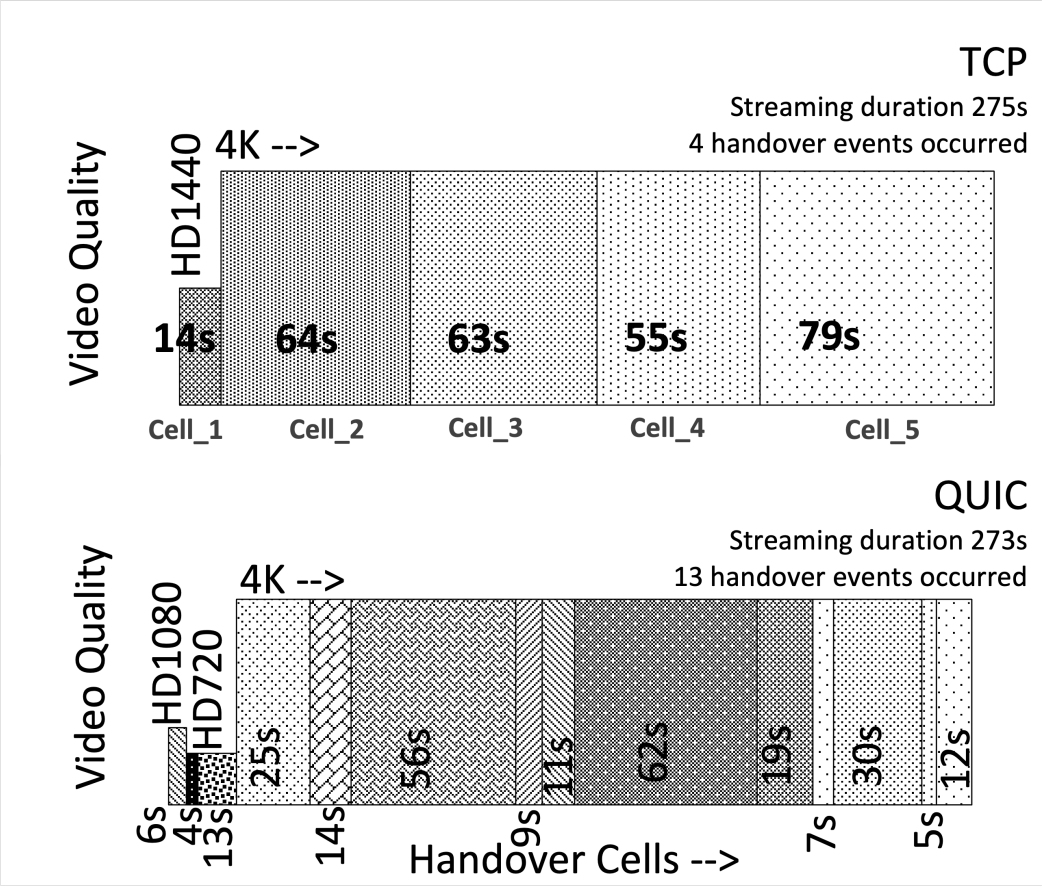


Figure 5.9: During mobility, the streaming experience the mobile handover events. A typical performance of a 5-minute-long video being loaded and played back (streamed) over TCP and QUIC. Time spent in every cell is also illustrated. In both cases, the video started in lower resolution and upgraded to 4K quality later.

streaming session.

The real-life mobility through the tram route as depicted in figure 5.6 is taken under consideration. In our setup, we did not encounter any buffering events.

Figure 5.9 illustrate a typical real-life streaming behavior on the move with handover events. In this sample, the video streaming of the 5 minutes long video was requested in 4K quality. Figures 5.8[a] and [b] show YouTube streaming performance of the same video in terms of the fraction of content loaded on QUIC and TCP over 5G under static and dynamic conditions. We can notice that in the beginning, the fraction of content loaded over TCP was better than QUIC until around 23 seconds. This can be explained in conjunction with figure 5.9 which says TCP streaming quality was better and stable to facilitate smoother progress of loading, however, due to quick handover events (within 6 and 4 seconds consecutively) streaming over QUIC had to lower the video quality.

In figure 5.9, on the mobility, in both cases with TCP and QUIC, the streaming started with lower HD qualities. The width of the rectangular boxes signifies the time spent in a particular cell and the height signifies the prevalent quality of the video streaming in the cell. In this instance, the streaming over TCP survives 4 handovers over a streaming duration of 275s (time to completely download the video content) and QUIC survives 13 handovers over a period of 273s. The handovers are cellular events that happened independently (due to the mobility of the user) of the transport protocol in use. In other words, the transport protocol (TCP or QUIC) used does not trigger or influence the handover events. The objective of the measurement collection was rather to understand if cellular handover events have different impacts on TCP and QUIC. Thus, the experimental results shown in the figure rather characterize the 4K streaming performance over TCP and QUIC.

Empirically, we can say from figures 5.9 and 5.8, that streaming application performance remains unaffected by handover events due to its bursty nature of data transfer and adaptive video quality selection schemes depending on the network connection quality.

5.3.6 Web Browsing

Web browsing is a basic means of retrieving information from the web and remains one of the key uses of mobile broadband connections. We thus study performance over QUIC and TCP when connected to 5G.

Methodology: To measure the performance of web browsing, we automate the browsing operation of the top 105 Tranco websites that are QUIC enabled. The automation script visits the websites on both TCP and QUIC. We used PLT as the measurement of the performance for web browsing applications. Like measuring the PLT for Google Searches and Google Maps, for general web browsing we also calculated PLT using the Selenium web driver and Performance Timing API.

Takeaway1: All requests from the browser are initiated on TCP, and then, if QUIC service is available, another QUIC connection is established after the QUIC handshake. Figure 5.10 shows TCP initiates the original connection and QUIC connection is established later providing dual connections to download the webpage contents. Hence, in real-life production-grade web applications, a part of a web page is always downloaded on TCP and HTTP/3 first, and then once the QUIC connection is negotiated the rest of the contents download on both QUIC-HTTP/3 and TCP-HTTP/2.

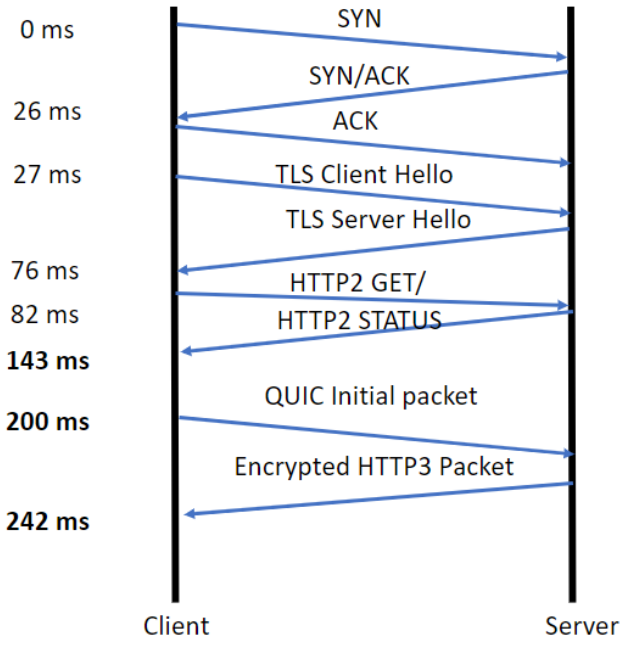


Figure 5.10: A typical TCP-QUIC connection setup timeline, as observed in our experimental setup, from Google Chrome to www.youtube.com.

Figure 5.10 shows a typical timeline from our setup to access the YouTube homepage on QUIC. The first connection establishment over TCP happens by the usual 3-way handshake followed by TLS handshake (76 ms). HTTP/2 response was first received after 143 ms. At 200 ms the alternative service (alt-serv) of QUIC kicks in and in the next 42 ms, the first encrypted QUIC packet arrives to the client.

Takeaway2: If the browser cache is warmed and the client has had a successful QUIC connection to the server, QUIC can leverage on its 0-RTT connection reestablishment. In Figure 5.11, empirical evidence in accessing the YouTube home page shows that the 0-RTT feature helps the median of TTFB to be reduced by 70% on 5G against TCP 3-way handshake.

0-RTT (zero round-trip time) is a salient feature of QUIC and for users, it seems to be beneficial because it allows them to reestablish a secure connection to a recently visited website faster. However, our experiments show that although 0-RTT improves the connection reestablishment time, this does not essentially translate to the performance gain such as lower PLT, which is measured at the application layer. PLT is still the most widely chosen Web-browsing metric to estimate the Quality of Experience (QoE) of Web users [43].

Takeaway3: PLT of a web page not only depends on factors like transport protocols or underlying access technology used but also on how the prioritization of resource

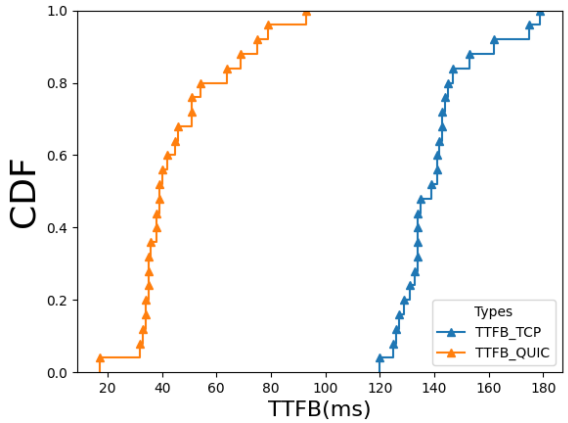


Figure 5.11: QUIC's gain due to 0-RTT quantified in terms of TTFB from the transport layer. The median of TTFB reduced by 70% on 5G against TCP 3-way handshake.

delivery is designed [124]. Moreover, at this point accessing a top production-grade website purely on QUIC is not possible because always a part of its content will be downloaded over TCP and HTTP/2 hence the reflection of PLT on QUIC will not necessarily be a true reflection of the performance of the combination of QUIC and HTTP/3 alone. Figure 5.10 shows TCP initiates the original connection and QUIC connection is established later providing dual connections to download the webpage contents. Moreover, a gain in transport layer TTBF may not be converted into application layer PLT ultimately if the front end is designed sub-optimally because PLT is calculated at the end of the webpage parsing process and TTBF denotes just how fast the first packet arrives.

Figure 5.12, shows the delta in PLTs of TOP 105 Tranco QUIC enabled websites, between TCP and QUIC as underlying transport protocols over a 5G static indoor environment. The figure shows that, in our setup, 61 out of 105 top QUIC-enabled websites load faster on TCP HTTP/2 over 5G. The median PLT difference between QUIC and TCP+HTTP/2 for 105 websites was 70 ms. The RTTs of the connections to the servers were less than 39 ms (which is representative of most real-life connections, measured through ICMP ping responses) for 102 websites. The difference in the amount of traffic transferred over the network between access over TCP and access over QUIC was less than 667 kb for 100 out of 105 websites.

5.4 Discussion

Our work shows that, although earlier measurement studies with QUIC have indicated potential performance gains from the server side and on wired connections, in the wild QUIC's performance is on par with that of TCP for major applications and web pages. Despite our attempt to present a comprehensive measurement study, due to budget and time constraints, there are a set of limitations. For instance, a nationwide multi-city measurement drive with multiple operators would be more interesting to compare various 5G networks' performance and QUIC's dependence on different underlying networks. The presented measurements are taken from commercial 5G networks and production-grade applications on QUIC protocol hence the results are fairly indicative of the true QUIC-5G interplay. However, controlled-lab environment measurements with various open-sourced implementations of QUIC could provide specific QUIC feature-wise performance analysis. For instance, we do not have access to the congestion control algorithms for QUIC used by various commercial application providers such as Google or Facebook to play around with their configuration and study the implications on the real-world performance from the user end. In a controlled lab environment, different parameters of QUIC can be tweaked to study their individual impacts on different applications. Future work will aim to address these limitations.

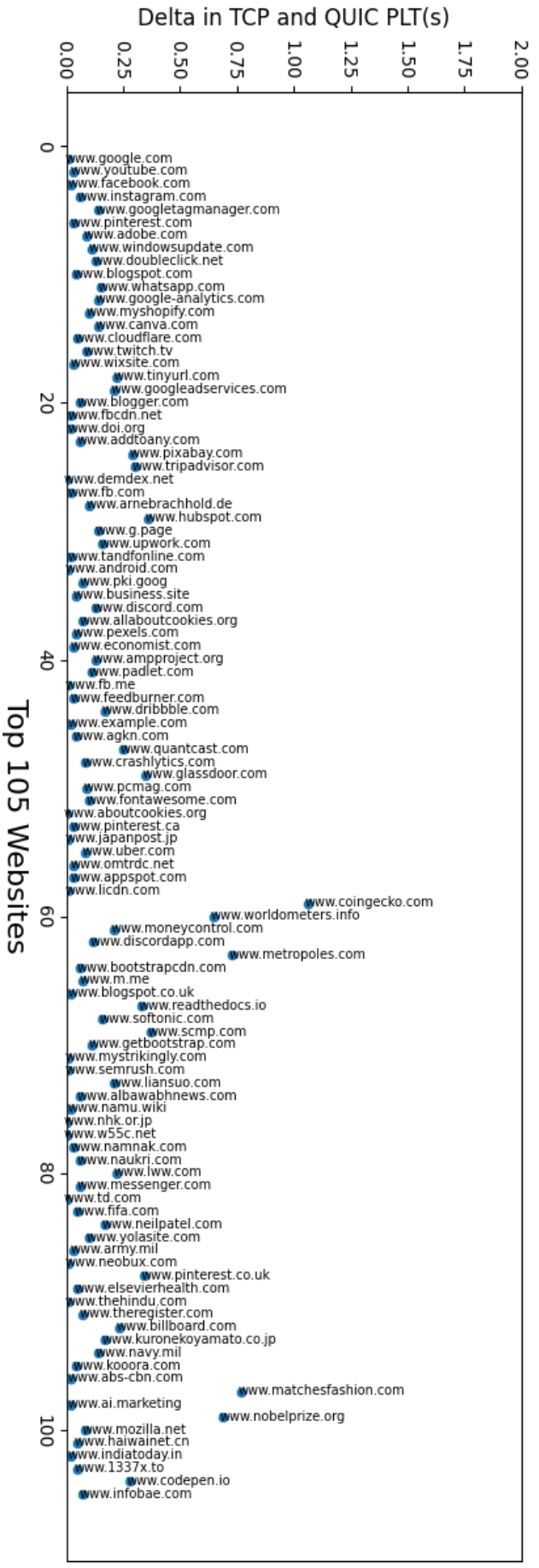


Figure 5.12: Delta in PLTs of top 105 Tranco QUIC enabled websites, between QUIC and TCP as the underlying transport protocol over 5G static environment

Chapter 6

Conclusions and Future Works

6.1 Conclusions

The 5G network is being standardized by the 3GPP with continued integration of heterogeneous networking technologies such as cloud computing and satellite communications, with a view to making the 5G and beyond mobile infrastructure more scalable and capable of delivering future application requirements. On the other hand, IETF standardized the QUIC transport protocol recently (May 2021) as a viable alternative to the de facto TCP [4]. Interactions of the next generations of mobile technologies, and protocols, such as multi-tenancy on mobile networks, and QUIC protocol, on top of the common computing nodes, are yet to be empirically understood in a real-life setting.

This thesis takes the first step to examine the security vulnerability of the future mobile network system and characterize application performances through commercial 5G network measurements aiming to better understand the nature of emerging mobile Internet from a system's perspective.

The thesis specifically focuses on uncovering how mobile network performance is affected by a cybersecurity attack called CUPS Hijacking on RAN slicing systems. In the second half of the thesis, we report on a 5G/transport network measurements collection campaign. This study characterizes how popular applications perform over the new transmission protocol QUIC when accessed over 5G, providing interesting insights into the 5G and QUIC interactions.

In order to systematically study the consequences of a CUPS hijacking attack on a RAN slicing system in terms of network performance, we prototype a real-world RAN slicing use case on an end-to-end mobile network test-bed, design a simple attack

model, and quantify the impacts of CUPS hijacking on the network performance. Our study demonstrates the plausibility of a successful CUPS hijacking attack by a rogue slice owner in a RAN slicing system and shows that such an attack can increase the RAN slice control-plane signaling delay above the operational upper-bound, to disrupt network operation.

In the city-wide 5G measurement study we analyze the performance of popular applications including file download, 4K video streaming, web-based navigation, and web search over QUIC, and compared that with the performance achieved over TCP when connected to a commercial 5G network. We conclude that, although the design of the QUIC protocol eliminates the TCP-like 3-way connection establishment to complete the initial handshake faster than TCP and, the multiple stream multiplexing feature of the QUIC eliminates TCP's notorious head-of-line blocking to achieve faster client data delivery, popular web applications achieve similar user experiences while accessing over QUIC and TCP. Hence today's mobile network infrastructures' transformations to fulfill the growing demand for higher data rates above 100Mbps) and the need for extremely low-latency (less than a millisecond) communication links enabling futuristic high bandwidth applications such as mission-critical communications, augmented/virtual reality, self-driving cars, and so on, needs to be studied and verified through network and application measurements from a users perspective.

We believe, the results presented in this thesis motivate immediate future research in the area of 5G system security and commercial network performance benchmarking, toward reliable and robust mobile Internet services.

6.2 Limitations and Future Work

This section summarizes the limitations and future research opportunities in relation to the contributions made in this thesis.

Limitations of the CUPS hijacking study: In this work, we attempt, to model an SDN-based RAN slicing system and analyze the impacts of a CUPS hijacking attack on it. However, the work in its current form bears certain limitations due to constraints stemming from issues such as lack of access to the codebase of proprietary RAN slicing systems and real network datasets. The results we presented are obtained from a prototyped mobile network testbed and can only be taken as indicative to infer the performance of a commercial NHMO network under a similar set of assumptions rather than an exact replication. Moreover, our threat model exclusively targets SDN-based

RAN slicing systems running on shared network infrastructure assuming no traffic shaping in action with no physical separations between CP and UP traffic flows. The threat model might not be extendable to RAN slicing systems designed with significant deviations from the SDN architecture and deployed in a setting where physical isolation is insured. However, other slicing systems in the literature, like in [53], are reported with a similar SDN-based architecture to that of Orion where CUPS plays a crucial role. Finally, this work focuses on RAN slicing system, and not an end-to-end mobile network slicing system. The latter is a natural target for follow-on work.

Limitations of the QUIC performance measurement study: Although we carried out extensive measurement collection throughout the city both in static environments and on public transports, our study could not be extended over multiple cities and among multiple operators for a comparative view, due to limited time and budget constraints. Moreover, current 5G deployments in the UK do not have rural coverage and also do not unleash mmWave spectrum with standalone (SA) network configuration. Hence, our study does not report the performance in rural scenarios or under an SA 5G NR coverage, but a single band coverage on the n78 band.

Despite the above-mentioned limitations, we believe, the insights we provide will certainly motivate future research initiatives in the field of 5G security and QUIC performance characterization over 5G and beyond mobile networks.

The following subsections provide concrete road maps toward future research avenues and a set of open research questions that can be taken into account for further research endeavors in the direction explored by this thesis.

Future work on 5G slicing security

O-RAN security: Our work on slicing security considers a 3GPP RAN architecture. However, O-RAN is another emerging RAN architecture, which enables multi-vendor interoperability with additional open interfaces broadening the threat landscape. Thus a natural future direction of research on slicing security would be to include O-RAN slicing security and extend a similar study that this thesis presents.

Slicing security analysis in the context of O-RAN architecture entices a set of immediate open research questions as follows.

Open RQ1: How to empirically show the impacts of cyberattacks on the ORAN architecture by exploiting the interoperable open interfaces of the O-RAN standard?

Open RQ2: How to conduct threat modelings with a view to multi-operator multi-

vendor O-RAN use-cases such as NHMO?

Open RQ3: Which RAN architecture, 3GPP and O-RAN, would be more robust to withstand innovative CUPS hijacking attacks?

Our results and approach of quantifying the attack impact on RAN slicing motivate the above-mentioned immediate open RQs to be examined in the future for robust RAN eco-system design.

End-to-end 5G Slicing security: We examined the impacts of the CUPS hijacking attack only on the RAN segment of mobile networks. However, end-to-end mobile network slicing, which includes RAN, transport network, and core network separation among multiple MVNOs, is another use case to which to extend the CUPS hijacking study. End-to-end slicing security is a less examined area in 5G security literature. We provide the following set of open research questions in the context of end-to-end slicing security.

Open RQ1: How to design and implement a threat model for vulnerability analysis of an end-to-end mobile slicing system?

Open RQ2: Can the cloud-native 5G core network security functions withstand DDoS attacks aiming to disrupt the control plane of end-to-end mobile slicing systems?

Open RQ3: How to mitigate the side channel security risks of cloud-native SBA 5G core that enables multi-operator hardware sharing with no physical separation for end-to-end network slicing?

The above-mentioned open questions would be crucial for future research toward a secure end-to-end 5G network slicing for multi-operator scenarios.

Future works on QUIC performance over 5G

To make the results of our QUIC performance study more generalizable, the following aspects can be considered for further study.

Correlations among QUIC parameters to the 5G PHY layer parameters: In this thesis, we present how the application performance varies with the underlying transport layer protocols performance, however, the correlations among the transport layer QUIC protocol's parameters such as packet length, initial maximum data length, etc with PHY layer parameters such as SINR, CQI, etc are correlated. Thus the following RQ would be interesting to explore in future work.

Open RQ1: What are the correlations between QUIC transport parameters to physical layer radio parameters?

To answer this important research question, future studies will include multi-city measurement collections with extensive physical layer statistics collected with reliable mobile drive test tools such as Infovista TEMS [58]. A statistically significant correlation between the QUIC performance and the radio network parameters will expedite further application and protocol performance optimization.

Futuristic use cases and applications: Another important future direction of the study would be to include emerging applications for analyzing performances on QUIC over 5G, such as online multiplayer gaming, and VR/AR. Thus, the following research question will be part of our future work.

Open RQ2: Which transport protocol, TCP or the UDP-based QUIC, will enable enhanced user experience of the emerging AR/VR applications in a real-life setting?

An extensive real-world measurement data-driven analysis paves the way forward for delivering optimized VR/AR or other emerging applications on emerging QUIC-5G network infrastructure.

Acronyms

4G	4th Generation
5G	5th Generation
5G-CN	5G- Core Network
5G-GUTI	5G- Global Unique Temporary Identifier
5GS	5G System
AF	Application Function
AI	Artificial Intelligence
AMF	Access and Mobility Management Function
AUSF	Authentication Server Function
CU	Centralized Unit
CUPS	Control and User Plane Separation
DDoS	Distributed Denial of Service
DHCP	Dynamic Host Configuration Protocol
DoS	Denial of Service
DU	Distributed Unit
E2E	End-to-End
EPC	Evolved Packet Core
GAN	Generative Adversarial Network
gNB	next generation NodeB
GSM	Global System for Mobile Communication
HSS	Home Subscriber Server
IoT	Internet of Things
IT	Information Technology
LTE	Long Term Evolution
ME	Mobile Equipment
MIMO	Multiple Input Multiple Output
ML	Machine Learning
MME	Mobility Management Entity
N31WF	Non-3GPP Internetworking Function
NEF	Network Exposure Function
NF	Network Function
ng-eNB	5G-eNodeB
NOMA	Non Orthogonal Multiple Access
NR	New Radio
NRF	NF Repository Function
NSA	Non-Standalone
NSSF	Network Slice Selection Function
NVF	Network Function Virtualization
O&M	Operations and Monitoring
PCF	Policy and Charging Function
PCRF	Policy and Charging Rule Function
PLMN	Public Land Mobile Network
R-15	3GPP Release-15
RN	Radio Network
RNTI	Radio Network Temporary Identifier
RU	Radio Unit
SA3	Service and System Aspects 3
SBA	Service-Based Architecture
SBI	Service-Based Interface
SDN	Software Defined Networking
SDO	Standards Development Organization
SEPP	Security Protection Proxy
SLA	Service level agreement
SSH	Secure Socket Shell
SUCI	SUbscription Concealed Identifier
TSG	Technological Specification Group
UDM	Unified Data Management
UE	Users Equipment
ULLC	Ultra-Low Latency Communication
UMTS	Universal Mobile Telecommunication System
USIM	Universal Subscriber Identity Module
VPN	Virtual Private Network

Bibliography

[1] *Cloud security guidance*. https://www.ncsc.gov.uk/collection/cloud/understanding-cloud-services/technically-enforced-separation-in-the-cloud.

[2] *Dense Air*. http://denseair.net/.

[3] *Operational security*. https://www.ncsc.gov.uk/collection/cloud/the-cloud-security-principles/principle-5-operational-security.

[4] *QUIC Standardization, IETF*. https://datatracker.ietf.org/doc/rfc9000/.

[5] Threat Landscape and Good Practice Guide for Software Defined Networks/5G. White paper (wp), HUAWEI TECHNOLOGIES, 2016.

[6] 3GPP;TSG Services and System Aspects; Rel 14 Description; (Release 14). Ts, 3GPP, 06 2017.

[7] 5G Whitepaper: 5G Security Overview. White paper (wp), University of Surrey, 2017.

[8] 5G security – enabling a trustworthy 5G system. White paper (wp), Ericsson, 2018.

[9] 3GPP; TSG Services and System Aspects; Rel 15 Description; (Release 15). Ts, 3GPP, 09 2019.

[10] 3rd Generation Partnership Project; Technical Specification Group Services and System Aspects; Release 16 Description; Summary of Rel-16 Work Items (Release 16). Technical report (tr), 3rd Generation Partnership Project (3GPP), 09 2019.

[11] National vulnerability database, 2019. [Online; accessed 26-November-2019].

[12] 3GPP TS 38.104 version 17.6.0 Release 17; Rel 17; (Release 17). Ts, 3GPP, 07 2022.

[13] 3GPP; TSG Services and System Aspects; Rel 16 Description; (Release 16). Ts, 3GPP, 04 2022.

[14] IETF; QUIC: A UDP-Based Multiplexed and Secure Transport; RFC 9000; (Version 1). Ts, IETF, 04 2022.

[15] 3GPP. 5G;NR;Physical layer;General description. Technical specification (ts), 3rd Generation Partnership Project (3GPP), 09 2018.

[16] Ijaz Ahmad, Tanesh Kumar, Madhusanka Liyanage, Jude Okwuibe, Mika Ylianttila, and Andrei Gurtov. 5g security: Analysis of threats and solutions. In *2017 IEEE Conference on Standards for Communications and Networking (CSCN)*, pages 193–199. IEEE, 2017.

[17] Ijaz Ahmad, Tanesh Kumar, Madhusanka Liyanage, Jude Okwuibe, Mika Ylianttila, and Andrei Gurtov. Overview of 5g security challenges and solutions. *IEEE Communications Standards Magazine*, 2(1):36–43, 2018.

[18] Adnan Akhunzada, Ejaz Ahmed, Abdullah Gani, Muhammad Khurram Khan, Muhammad Imran, and Sghaier Guizani. Securing software defined networks: taxonomy, requirements, and open issues. *IEEE Communications Magazine*, 53(4):36–44, 2015.

[19] NGMN Alliance. 5G security recommendations package# 2: Network slicing. *Ngmn*, pages 1–12, 2016.

[20] Sahel Alouneh et al. Quic transmission protocol: Test-bed design, implementation and experimental evaluation. *Journal of Electrical Engineering*, 72(1):20–28, 2021.

[21] 5G Americas. The Evolution of Security in 5G. White paper (wp), 5G Americas, 08 2019.

[22] Daniele Antonioli, Nils Ole Tippenhauer, and Kasper Rasmussen. Nearby threats: Reversing, analyzing, and attacking google's' nearby connections' on android. 2019.

[23] Nobuo Aoki, Kohei Okazaki, Hiroyasu Obata, and Junichi Funasaka. Performance evaluation on concurrent connecting quic and tcp nodes over wireless lan. In *2021 3rd International Conference on Computer Communication and the Internet (ICCCI)*, pages 109–116. IEEE, 2021.

[24] Myrto Arapinis, Loretta Mancini, Eike Ritter, Mark Ryan, Nico Golde, Kevin Redon, and Ravishankar Borgaonkar. New privacy issues in mobile telephony: Fix and verification. In *Proceedings of the 2012 ACM Conference on Computer and Communications Security*, CCS '12, pages 205–216, New York, NY, USA, 2012. ACM.

[25] Ghada Arfaoui, Pascal Bisson, Rolf Blom, Ravishankar Borgaonkar, Håkan Englund, Edith Félix, Felix Klaedtke, Prajwol Kumar Nakarmi, Mats Näslund, Piers O'Hanlon, et al. A security architecture for 5g networks. *IEEE Access*, 6:22466–22479, 2018.

[26] Divyashri Bhat, Amr Rizk, and Michael Zink. Not so quic: A performance study of dash over quic. In *Proceedings of the 27th workshop on network and operating systems support for digital audio and video*, pages 13–18, 2017.

[27] Roberto Bifulco, Heng Cui, Ghassan O Karame, and Felix Klaedtke. Fingerprinting software-defined networks. In *2015 IEEE 23rd International Conference on Network Protocols (ICNP)*, pages 453–459. IEEE, 2015.

[28] Gregory Blanc, Nizar Kheir, Dhouha Ayed, Vincent Lefebvre, Edgardo Montes de Oca, and Pascal Bisson. Towards a 5g security architecture: Articulating software-defined security and security as a service. In *Proceedings of the 13th International Conference on Availability, Reliability and Security*, page 47. ACM, 2018.

[29] Bego Blanco, Jose Oscar Fajardo, Ioannis Giannoulakis, Emmanouil Kafetzakis, Shuping Peng, Jordi Pérez-Romero, Irena Trajkovska, Pouria S Khodashenas, Leonardo Goratti, Michele Paolino, et al. Technology pillars in the architecture of future 5g mobile networks: Nfv, mec and sdn. *Computer Standards & Interfaces*, 54:216–228, 2017.

[30] Andreas Blenk, Arsany Basta, Martin Reisslein, and Wolfgang Kellerer. Survey on network virtualization hypervisors for software defined networking. *IEEE Communications Surveys & Tutorials*, 18(1):655–685, 2015.

[31] Fabio Bulgarella, Mauro Cociglio, Giuseppe Fioccola, Guido Marchetto, and Riccardo Sisto. Performance measurements of quic communications. In *Proceedings of the Applied Networking Research Workshop*, pages 8–14, 2019.

[32] Yulong Cao, Chaowei Xiao, Benjamin Cyr, Yimeng Zhou, Won Park, Sara Rampazzi, Qi Alfred Chen, Kevin Fu, and Z Morley Mao. Adversarial sensor attack on lidar-based perception in autonomous driving. *arXiv preprint arXiv:1907.06826*, 2019.

[33] Gaetano Carlucci, Luca De Cicco, and Saverio Mascolo. Http over udp: an experimental investigation of quic. In *Proceedings of the 30th Annual ACM Symposium on Applied Computing*, pages 609–614, 2015.

[34] Min Chen, Yongfeng Qian, Shiwen Mao, Wan Tang, and Ximin Yang. Software-defined mobile networks security. *Mobile Networks and Applications*, 21(5):729–743, 2016.

[35] Merlin Chlosta, David Rupprecht, Thorsten Holz, and Christina Pöpper. LTE security disabled: misconfiguration in commercial networks. In *Proceedings of the 12th ACM WiSec*, pages 261–266, 2019.

[36] Ioannis P Chochliouros, Anastasia S Spiliopoulou, Pavlos Lazaridis, Athanassios Dardamanis, Zaharias Zaharis, and Alexandros Kostopoulos. Dynamic network slicing: Challenges and opportunities. In *IFIP International Conference on Artificial Intelligence Applications and Innovations*, pages 47–60. Springer, 2020.

[37] Chromium. QUIC, a multiplexed transport over UDP, 2021. Retrieved August, 2021 from https://www.chromium.org/quic.

[38] Cisco. Control Plane and User Plane Separation (CUPS). Technical report, 10 2018. Retrieved May, 2021 from https://tinyurl.com/c2x488xn.

[39] Cisco. Slicing the transport network for 5G, 2018. Retrieved June, 2020 from https://bit.ly/3eN6rEn.

[40] clouflare. Does my browser support HTTP/3 QUIC?, 2021. Retrieved August, 2021 from https://cloudflare-quic.com/.

[41] Sarah Cook, Bertrand Mathieu, Patrick Truong, and Isabelle Hamchaoui. Quic: Better for what and for whom? In *2017 IEEE International Conference on Communications (ICC)*, pages 1–6. IEEE, 2017.

[42] Xavier Costa-Pérez, Joerg Swetina, Tao Guo, Rajesh Mahindra, and Sampath Rangarajan. Radio access network virtualization for future mobile carrier networks. *IEEE Communications Magazine*, 51(7):27–35, 2013.

[43] Diego Neves da Hora, Alemnew Sheferaw Asrese, Vassilis Christophides, Renata Teixeira, and Dario Rossi. Narrowing the gap between qos metrics and web qoe using above-the-fold metrics. In *International Conference on Passive and Active Network Measurement*, pages 31–43. Springer, 2018.

[44] Therdpong Daengsi, Pana Ungkap, and Pongpisit Wuttidittachotti. A study of 5g network performance: A pilot field trial at the main skytrain stations in bangkok. In *2021 International Conference on Artificial Intelligence and Computer Science Technology (ICAICST)*, pages 191–195. IEEE, 2021.

[45] Martin Dehnel-Wild and Cas Cremers. Security vulnerability in 5g-aka draft. *Department of Computer Science, University of Oxford, Tech. Rep*, 2018.

[46] Vaibhav Hemant Dixit, Adam Doupé, Yan Shoshitaishvili, Ziming Zhao, and Gail-Joon Ahn. Aim-sdn: Attacking information mismanagement in sdn-datastores. In *Proceedings of the 2018 ACM SIGSAC Conference on Computer and Communications Security*, pages 664–676. ACM, 2018.

[47] Mo Dong, Qingxi Li, Doron Zarchy, P Brighten Godfrey, and Michael Schapira. ˆPCC‘: Re-architecting congestion control for consistent high performance. In *12th ˆUSENIX‘ Symposium on Networked Systems Design and Implementation (ˆNSDI‘ 15)*, pages 395–408, 2015.

[48] ETSI. System Architecture for the 5G System; (Release 15). Technical specification (ts), ETSI 3rd Generation Partnership Project (3GPP), 06 2019.

[49] Mohamed Amine Ferrag, Leandros Maglaras, Antonios Argyriou, Dimitrios Kosmanos, and Helge Janicke. Security for 4g and 5g cellular networks: A survey of existing authentication and privacy-preserving schemes. *Journal of Network and Computer Applications*, 101:55–82, 2018.

[50] Xenofon Foukas, Mahesh K Marina, and Kimon Kontovasilis. Orion: RAN slicing for a flexible and cost-effective multi-service mobile network architecture. In *Proceedings of the 23rd annual international conference on mobile computing and networking*, pages 127–140, 2017.

[51] Xenofon Foukas, Navid Nikaein, Mohamed M. Kassem, Mahesh K. Marina, and Kimon Kontovasilis. Flexran: A flexible and programmable platform for software-defined radio access networks. In *Proceedings of the 12th International on Conference on Emerging Networking EXperiments and Technologies*, CoNEXT '16, pages 427–441, New York, NY, USA, 2016. ACM.

[52] Thomas Frisanco, Paul Tafertshofer, Pierre Lurin, and Rachel Ang. Infrastructure sharing and shared operations for mobile network operators from a deployment and operations view. In *NOMS 2008-2008 IEEE Network Operations and Management Symposium*, pages 129–136. IEEE, 2008.

[53] Gines Garcia-Aviles, Marco Gramaglia, Pablo Serrano, and Albert Banchs. Posens: A practical open source solution for end-to-end network slicing. *IEEE Wireless Communications*, 25(5):30–37, 2018.

[54] Markus Gross. Eth researchers uncover security gaps in the 5g mobile communication standard, 2018.

[55] Daojing He, Sammy Chan, and Mohsen Guizani. Securing software defined wireless networks. *IEEE Communications Magazine*, 54(1):20–25, 2016.

[56] Jim Hodges. Heavy Reading's 2019 5G Security Survey. Research report (rr), F5 Networks, Fortinet, NetNumber, and Palo Alto Networks, 02 2019.

[57] Syed Rafiul Hussain, Mitziu Echeverria, Omar Chowdhury, Ninghui Li, and Elisa Bertino. Privacy attacks to the 4g and 5g cellular paging protocols using side channel information. In *NDSS*, 2019.

[58] Infovista. TEMS Investigation 5G, 2022. Retrieved Aug, 2022 from https://www.infovista.com/tems/investigation .

[59] Open Air Interface. 5g software alliance for democratising wireless innovation, 2017.

[60] Jana Iyengar and Martin Thomson. QUIC: A UDP-Based Multiplexed and Secure Transport. RFC 9000, May 2021.

[61] Raj Jain, Arjan Durresi, and Gojko Babic. Throughput fairness index: An explanation. In *ATM Forum contribution*, volume 99, 1999.

[62] James Lapham. Smal Cell: Neutral Hosting' is it the Future, 2016. Retrieved June, 2020 from https://tinyurl.com/yawfvtgr.

[63] Roger Piqueras Jover. Security attacks against the availability of lte mobility networks: Overview and research directions. In *2013 16th international symposium on wireless personal multimedia communications (WPMC)*, pages 1–9. IEEE, 2013.

[64] Roger Piqueras Jover and Vuk Marojevic. Security and protocol exploit analysis of the 5g specifications. *IEEE Access*, 7:24956–24963, 2019.

[65] Arash Molavi Kakhki, Samuel Jero, David Choffnes, Cristina Nita-Rotaru, and Alan Mislove. Taking a long look at quic: an approach for rigorous evaluation of rapidly evolving transport protocols. In *Proceedings of the 2017 Internet Measurement Conference*, pages 290–303, 2017.

[66] Prashant K Kharat, Aniket Rege, Aneesh Goel, and Muralidhar Kulkarni. Quic protocol performance in wireless networks. In *2018 International Conference on Communication and Signal Processing (ICCSP)*, pages 0472–0476. IEEE, 2018.

[67] Sajad Khorsandroo and Ali Saman Tosun. Time inference attacks on software defined networks: Challenges and countermeasures. In *2018 IEEE 11th International Conference on Cloud Computing (CLOUD)*, pages 342–349. IEEE, 2018.

[68] Ahmed Khurshid, Xuan Zou, Wenxuan Zhou, Matthew Caesar, and P Brighten Godfrey. Veriflow: Verifying network-wide invariants in real time. In *Presented as part of the 10th ˆUSENIX‘ Symposium on Networked Systems Design and Implementation (ˆNSDI‘ 13)*, pages 15–27, 2013.

[69] Tanesh Kumar, Madhusanka Liyanage, Ijaz Ahmad, An Braeken, and Mika Ylianttila. User privacy, identity and trust in 5g. *A Comprehensive Guide to 5G Security*, page 267, 2018.

[70] Anitha Kumari, Shymala Gowri, EG Radhika, et al. An approach for end-to-end (e2e) security of 5g applications. In *2018 IEEE 4th International Conference on Big Data Security on Cloud (BigDataSecurity), IEEE International Conference on High Performance and Smart Computing,(HPSC) and IEEE International Conference on Intelligent Data and Security (IDS)*, pages 133–138. IEEE, 2018.

[71] Ike Kunze, Klaus Wehrle, and Jan Rüth. L, q, r, and t–which spin bit cousin is here to stay? *arXiv preprint arXiv:2106.13710*, 2021.

[72] Adam Langley, Alistair Riddoch, Alyssa Wilk, Antonio Vicente, Charles Krasic, Dan Zhang, Fan Yang, Fedor Kouranov, Ian Swett, Janardhan Iyengar, et al. The quic transport protocol: Design and internet-scale deployment. In *Proceedings of the conference of the ACM special interest group on data communication*, pages 183–196, 2017.

[73] Yang Li, Hao Lin, Zhenhua Li, Yunhao Liu, Feng Qian, Liangyi Gong, Xianlong Xin, and Tianyin Xu. A nationwide study on cellular reliability: Measurement, analysis, and enhancements. 2021.

[74] Sheng Liu, Michael K Reiter, and Vyas Sekar. Flow reconnaissance via timing attacks on sdn switches. In *2017 IEEE 37th international conference on distributed computing systems (ICDCS)*, pages 196–206. IEEE, 2017.

[75] Madhusanka Liyanage, Ahmed Bux Abro, Mika Ylianttila, and Andrei Gurtov. Opportunities and challenges of software-defined mobile networks in network security. *IEEE Security & Privacy*, 14(4):34–44, 2016.

[76] Madhusanka Liyanage, Ijaz Ahmed, Jude Okwuibe, Mika Ylianttila, Hammad Kabir, Jesus Llorente Santos, Raimo Kantola, Oscar Lopez Perez, Mikel Uriarte Itzazelaia, and Edgardo Montes De Oca. Enhancing security of software defined mobile networks. *IEEE Access*, 5:9422–9438, 2017.

[77] Madhusanka Liyanage, Ijaz Ahmed, Mika Ylianttila, Jesus Llorente Santos, Raimo Kantola, Oscar Lopez Perez, Mikel Uriarte Itzazelaia, Edgardo Montes de Oca, Asier Valtierra, and Carlos Jimenez. Security for future software defined mobile networks. In *2015 9th International Conference on Next Generation Mobile Applications, Services and Technologies*, pages 256–264. IEEE, 2015.

[78] Adrian Belmonte Martin, Louis Marinos, Evangelos Rekleitis, George Spanoudakis, and Nikolaos Petroulakis. Threat Landscape and Good Practice Guide for Software Defined Networks/5G. Technical report (tr), European Union Agency for Network and Information Security (ENISA), 2016.

[79] Clémentine Maurice, Manuel Weber, Michael Schwarz, Lukas Giner, Daniel Gruss, Carlo Alberto Boano, Stefan Mangard, and Kay Römer. Hello from the other side: Ssh over robust cache covert channels in the cloud. In *NDSS*, volume 17, pages 8–11, 2017.

[80] M Hammad Mazhar and Zubair Shafiq. Real-time video quality of experience monitoring for https and quic. In *IEEE INFOCOM 2018-IEEE Conference on Computer Communications*, pages 1331–1339. IEEE, 2018.

[81] Stephen McQuistin, Colin Perkins, and Marwan Fayed. Tcp hollywood: An unordered, time-lined, tcp for networked multimedia applications. In *2016 IFIP Networking Conference (IFIP Networking) and Workshops*, pages 422–430. IEEE, 2016.

[82] Péter Megyesi, Zsolt Krämer, and Sándor Molnár. How quick is quic? In *2016 IEEE International Conference on Communications (ICC)*, pages 1–6. IEEE, 2016.

[83] Diomidis S Michalopoulos, Borislava Gajic, Beatriz Gallego-Nicasio Crespo, Aravinthan Gopalasingham, and Jakob Belschner. Network resilience in virtualized architectures. In *Interactive Mobile Communication, Technologies and Learning*, pages 824–839. Springer, 2017.

[84] Rupendra Nath Mitra and Dharma P Agrawal. 5g mobile technology: A survey. *Ict Express*, 1(3):132–137, 2015.

[85] Rupendra Nath Mitra, Mohamed M Kassem, Jon Larrea, and Mahesh K Marina. Cups hijacking in mobile ran slicing: Modeling, prototyping, and analysis. In *2021 IEEE Conference on Communications and Network Security (CNS)*, pages 38–46. IEEE, 2021.

[86] Rupendra Nath Mitra and Mahesh K Marina. 5g mobile networks security landscape and major risks. *Wiley 5G Ref: The Essential 5G Reference Online*, pages 1–23, 2019.

[87] Sergio Morant, Jean-Philippe Wary, and Esa Piri. 5G Enablers for Network and System Security and Resilience. Deliverable), EG-ENSURE, 06 2016.

[88] Mohamed Moulay, Fernando Díez Muñoz, Vincenzo Mancuso, et al. On the experimental assessment of quic and congestion control schemes in cellular networks. In *The 19th Mediterranean Communication and Computer Networking Conference (IEEE MedComNet 2021)*, 2021.

[89] Kyung Mun. Making Neutral Host a Reality with OnGo. Technical report, 12 2018.

[90] mvfst. facebookincubator/mvfst, 2021. Retrieved August, 2021 from https://github.com/facebookincubator/mvfst.

[91] Arvind Narayanan, Eman Ramadan, Jason Carpenter, Qingxu Liu, Yu Liu, Feng Qian, and Zhi-Li Zhang. A first look at commercial 5g performance on smartphones. In *Proceedings of The Web Conference 2020*, pages 894–905, 2020.

[92] Binh Nguyen, Tian Zhang, Bozidar Radunovic, Ryan Stutsman, Thomas Karagiannis, Jakub Kocur, and Jacobus Van der Merwe. Echo: A reliable distributed cellular core network for hyper-scale public clouds. In *Proceedings of the 24th Annual International Conference on Mobile Computing and Networking*, pages 163–178, 2018.

[93] O-RAN Alliance. O-RAN Use Cases and Deployment Scenarios , 2020. Retrieved June, 2021 from https://www.o-ran.org/resources.

[94] Emeka Obiodu, Abdullahi K Abubakar, and Nishanth Sastry. Is it 5g or not? investigating doubts about the 5g icon and network performance. In *IEEE INFOCOM 2021-IEEE Conference on Computer Communications Workshops (INFOCOM WKSHPS)*, pages 1–6. IEEE, 2021.

[95] Reza Poorzare and Anna Calveras Augé. Challenges on the way of implementing tcp over 5g networks. *IEEE access*, 8:176393–176415, 2020.

[96] Reza Poorzare and Anna Calveras Augé. How sufficient is tcp when deployed in 5g mmwave networks over the urban deployment? *IEEE Access*, 9:36342–36355, 2021.

[97] Reza Poorzare and Anna Calveras. Fb-tcp: a 5g mmwave friendly tcp for urban deployments. *IEEE Access*, 2021.

[98] Adnan Qayyum, Muhammad Usama, Junaid Qadir, and Ala Al-Fuqaha. Securing connected & autonomous vehicles: Challenges posed by adversarial machine learning and the way forward. *arXiv preprint arXiv:1905.12762*, 2019.

[99] Adnan Qayyum, Muhammad Usama, Junaid Qadir, and Ala I. Al-Fuqaha. Securing connected & autonomous vehicles: Challenges posed by adversarial machine learning and the way forward. *CoRR*, abs/1905.12762, 2019.

[100] Siddharth Prakash Rao, Silke Holtmanns, Ian Oliver, and Tuomas Aura. Unblocking stolen mobile devices using ss7-map vulnerabilities: Exploiting the relationship between imei and imsi for eir access. In *2015 IEEE Trustcom/BigDataSE/ISPA*, volume 1, pages 1171–1176. IEEE, 2015.

[101] Department Statista Research. iot-number-of-connected-devices-worldwide. 2016.

[102] David Rupprecht, Adrian Dabrowski, Thorsten Holz, Edgar Weippl, and Christina Pöpper. On security research towards future mobile network generations. *IEEE Communications Surveys & Tutorials*, 20(3):2518–2542, 2018.

[103] David Rupprecht, Katharina Kohls, Thorsten Holz, and Christina Pöpper. Breaking lte on layer two. In *2019 IEEE Symposium on Security and Privacy (SP)*, pages 1121–1136. IEEE, 2019.

[104] Jan Rüth, Ingmar Poese, Christoph Dietzel, and Oliver Hohlfeld. A first look at quic in the wild. In *International Conference on Passive and Active Network Measurement*, pages 255–268. Springer, 2018.

[105] Jan Rüth, Konrad Wolsing, Klaus Wehrle, and Oliver Hohlfeld. Perceiving quic: Do users notice or even care? In *Proceedings of the 15th International Conference on Emerging Networking Experiments And Technologies*, pages 144–150, 2019.

[106] Konstantinos Samdanis, Xavier Costa-Perez, and Vincenzo Sciancalepore. From network sharing to multi-tenancy: The 5G network slice broker. *IEEE Communications Magazine*, 54(7):32–39, 2016.

[107] Vipin N Sathi and C Siva Ram Murthy. Distributed slice mobility attack: A novel targeted attack against network slices of 5g networks. *IEEE Networking Letters*, 3(1):5–9, 2020.

[108] Peter Schneider and Günther Horn. Towards 5g security. In *2015 IEEE Trustcom/BigDataSE/ISPA*, volume 1, pages 1165–1170. IEEE, 2015.

[109] Minjae Seo, Jaehan Kim, Eduard Marin, Myoungsung You, Taejune Park, Seungsoo Lee, Seungwon Shin, and Jinwoo Kim. Heimdallr: Fingerprinting sd-wan control-plane architecture via encrypted control traffic. In *Proceedings of the 38th Annual Computer Security Applications Conference*, pages 949–963, 2022.

[110] Michael Seufert, Raimund Schatz, Nikolas Wehner, and Pedro Casas. Quicker or not?-an empirical analysis of quic vs tcp for video streaming qoe provisioning. In *2019 22nd Conference on Innovation in Clouds, Internet and Networks and Workshops (ICIN)*, pages 7–12. IEEE, 2019.

[111] Altaf Shaik, Ravishankar Borgaonkar, N. Asokan, Valtteri Niemi, and Jean-Pierre Seifert. Practical attacks against privacy and availability in 4g/lte mobile communication systems, 2015.

[112] Altaf Shaik, Ravishankar Borgaonkar, N Asokan, Valtteri Niemi, and Jean-Pierre Seifert. Practical attacks against privacy and availability in 4g/lte mobile communication systems. *arXiv preprint arXiv:1510.07563*, 2015.

[113] Yi Shi and Yalin E Sagduyu. Adversarial machine learning for flooding attacks on 5g radio access network slicing. In *2021 IEEE International Conference on Communications Workshops (ICC Workshops)*, pages 1–6. IEEE, 2021.

[114] Murtaza A Siddiqi, Mohammad Khoso, and Abdul Aziz. Security issues in 5g network. 2017.

[115] Mike Surridge, Gianluca Correndo, Ken Meacham, Juri Papay, Stephen C. Phillips, Stefanie Wiegand, and Toby Wilkinson. Trust modelling in 5g mobile networks. In *Proceedings of the 2018 Workshop on Security in Softwarized Networks: Prospects and Challenges*, SecSoN '18, pages 14–19, New York, NY, USA, 2018. ACM.

[116] Kashyap Thimmaraju, Bhargava Shastry, Tobias Fiebig, Felicitas Hetzelt, Jean-Pierre Seifert, Anja Feldmann, and Stefan Schmid. Taking control of SDN-based cloud systems via the data plane. In *Proceedings of the Symposium on SDN Research*, pages 1–15, 2018.

[117] Muhammad Usama, Muhammad Asim, Siddique Latif, Junaid Qadir, et al. Generative adversarial networks for launching and thwarting adversarial attacks on network intrusion detection systems. In *2019 15th International Wireless Communications & Mobile Computing Conference (IWCMC)*, pages 78–83. IEEE, 2019.

[118] Muhammad Usama, Junaid Qadir, Ala I. Al-Fuqaha, and Mounir Hamdi. The adversarial machine learning conundrum: Can the insecurity of ML become the achilles' heel of cognitive networks? *CoRR*, abs/1906.00679, 2019.

[119] Prasanna Karthik Vairam, Gargi Mitra, Vignesh Manoharan, Chester Rebeiro, Byrav Ramamurthy, et al. Towards measuring quality of service in untrusted multi-vendor service function chains: Balancing security and resource consumption. In *IEEE INFOCOM 2019-IEEE Conference on Computer Communications*, pages 163–171. IEEE, 2019.

[120] Haibo Wang, Hongli Xu, Liusheng Huang, Jianxin Wang, and Xuwei Yang. Load-balancing routing in software defined networks with multiple controllers. *Computer Networks*, 141:82–91, 2018.

[121] Haopei Wang, Abhinav Srivastava, Lei Xu, Sungmin Hong, and Guofei Gu. Bring your own controller: Enabling tenant-defined SDN apps in IaaS clouds. In *IEEE INFOCOM 2017-IEEE Conference on Computer Communications*, pages 1–9. IEEE, 2017.

[122] Haopei Wang, Guangliang Yang, Phakpoom Chinprutthiwong, Lei Xu, Yangyong Zhang, and Guofei Gu. Towards fine-grained network security forensics and diagnosis in the sdn era. In *Proceedings of the 2018 ACM SIGSAC Conference on Computer and Communications Security*, pages 3–16. ACM, 2018.

[123] Ranysha Ware, Matthew K Mukerjee, Srinivasan Seshan, and Justine Sherry. Beyond jain's fairness index: Setting the bar for the deployment of congestion control algorithms. In *Proceedings of the 18th ACM Workshop on Hot Topics in Networks*, pages 17–24, 2019.

[124] Maarten Wijnants, Robin Marx, Peter Quax, and Wim Lamotte. Http/2 prioritization and its impact on web performance. In *Proceedings of the 2018 World Wide Web Conference*, pages 1755–1764, 2018.

[125] Daoyuan Wu, Debin Gao, Rocky KC Chang, En He, Eric KT Cheng, and Robert H Deng. Understanding open ports in android applications: Discovery, diagnosis, and security assessment. 2019.

[126] Yongpeng Wu, Ashish Khisti, Chengshan Xiao, Giuseppe Caire, Kai-Kit Wong, and Xiqi Gao. A survey of physical layer security techniques for 5g wireless networks and challenges ahead. *IEEE Journal on Selected Areas in Communications*, 36(4):679–695, 2018.

[127] Dongzhu Xu, Anfu Zhou, Xinyu Zhang, Guixian Wang, Xi Liu, Congkai An, Yiming Shi, Liang Liu, and Huadong Ma. Understanding operational 5g: A first measurement study on its coverage, performance and energy consumption. In *Proceedings of the Annual conference of the ACM Special Interest Group on Data Communication on the applications, technologies, architectures, and protocols for computer communication*, pages 479–494, 2020.

[128] Alexander Yu and Theophilus A Benson. Dissecting performance of production quic. In *Proceedings of the Web Conference 2021*, pages 1157–1168, 2021.

[129] Yajun Yu, Mingwei Xu, and Yuan Yang. When quic meets tcp: An experimental study. In *2017 IEEE 36th International Performance Computing and Communications Conference (IPCCC)*, pages 1–8. IEEE, 2017.

[130] Liang Zhao, Ming Li, Yasir Zaki, Andreas Timm-Giel, and Carmelita Görg. LTE virtualization: From theoretical gain to practical solution. In *2011 23rd International Teletraffic Congress (ITC)*, pages 71–78. IEEE, 2011.

[131] Jianer Zhou, Zhenyu Li, Qinghua Wu, Peter Steenkiste, Steve Uhlig, Jun Li, and Gaogang Xie. Tcp stalls at the server side: Measurement and mitigation. *IEEE/ACM Transactions on Networking*, 27(1):272–287, 2018.

[132] ZTE. 5G Security White Paper Security Makes 5G Go Further, 2019. Retrieved June, 2020 from https://tinyurl.com/yy55sxsj.