Digital Forensic Readiness Model for 5G Core Networks

Nathan Opperman

, , , , ,

Abstract

The introduction of 5G technology represents a significant milestone for cellular network technology, with unprecedented speeds, reduced latency, better energy efficiency, and higher data rates. However, these advancements also introduce new challenges, particularly in cybersecurity and digital forensics. Traditional forensic methods, designed for static, hardware-dependent environments, are no longer adequate for the dynamic, virtualized infrastructure of 5G networks. This research attempted to address these challenges by exploring and developing an agent-based approach to Digital Forensic Readiness (DFR) based on an existing model for DFR in NFV and a 5G Core Testbed to facilitate the solution's development and evaluation. The prototype, **5G Digital Forensic Readiness Tool** (5GDFRT), utilizes widely used tools such as Zeek for network security monitoring, Elasticsearch for data storage and search, Filebeat for log collection and preprocessing, Logstash for customizable data ingestion pipelines, and Kibana for user interaction and data visualization, enabling comprehensive customization and integration. The developed 5GDFRT demonstrates the capability to facilitate both real-time detection and event reconstruction through secure and comprehensive data collection.

Keywords: digital forensics, digital forensic readiness, security, 5G technology, 5G networks, NFV, Network Function Virtualization, centralized log collection

1. Introduction

The introduction of 5G technology marks a significant milestone in the evolution of telecommunications technology, promising unprecedented speeds, reduced latency, better energy efficiency, and higher data rates (Hajlaoui et al., 2020). This new generation of cellular technology is expected to impact various industries, from healthcare with remote surgery to autonomous driving with vehicle-to-vehicle (V2V) communication (Dangi et al., 2021).

Unfortunately, with these advancements come new challenges and problems in the realm of cybersecurity. The increased complexity and the decentralized nature of 5G technology has created new hurdles, particularly in the field of digital forensics, where previous methods and implementations are no longer adequate or practical (Sharevski, 2018). Given the increasing adoption of 5G technologies across various industries, it is important to address the associated security issues promptly. The importance of this research is motivated by the increasing reliance on 5G for critical infrastructure (Jover,

2019), and so ensuring digital forensic readiness is essential for maintaining security, trust, and compliance in 5G.

Digital forensics is a forensic science that forms the processes of identifying, acquiring, processing, analyzing, and reporting on data that is stored electronically with the goals of legal compliance, improved incident response, recovering lost data, supporting internal investigations, and aiding in criminal investigations. Digital forensics is crucial in modern large-scale organizations, as it forms part of the overall security of the organization, and in certain organizations it is required for legal compliance. Traditionally the processes of digital forensics have been reactive however, in recent years there has been a shift to what is known as Digital Forensic Readiness (DFR). The term 'digital forensic readiness' was first introduced in (Tan, 2001) and is described as setting up digital forensics in organizations with the aim to minimize costs while maximizing the capability of an organization to collect digital evidence. In simple terms, it is a proactive rather than a reactive approach to digital forensics.

5G is the latest generation of cellular network technology and has been built off the many advancements of the previous few generations (1-4G, LTE, LTE advanced) and now with the most recent generation: 5G. A 5G cellular network is comprised of two main components, the 5G Radio Access Network (RAN) and the 5G Core Network (CN) (Cardoso et al., 2020). The RAN connects 5G-enabled devices to the CN through antennas, base stations, and other receiving equipment. The RAN introduces advancements that enable it to manage significantly higher traffic loads and support far more simultaneous connections. Some of these technologies include massive MIMO (Multiple Input, Multiple Output), which enhances capacity and efficiency, mmWave technology that provides ultra-high-frequency communication, enabling faster data speeds, and beamforming which focuses signals directly toward devices, improving reliability and coverage (Smith and Doe, 2023). The 5G Core Network has also had many advancements with the introduction of a Service-Based Architecture (SBA), Network Function Virtualization (NFV), and Software-Defined Networking (SDN) that have improved efficiency, flexibility, and scalability in comparison to previous generations (Condoluci and Mahmoodi, 2018).

As mentioned, Network Function Virtualization is one of the technologies utilized in modern 5G Core Networks and was first introduced during the 4th generation of cellular networks. NFV is the virtualization of network functions that traditionally were performed using dedicated hardware appliances such as switches or load balancers. With NFV these network functions are now implemented as software applications known as Virtual Network Functions (VNF) running on commercial off the shelf (COTS) hardware (Sharevski, 2018). This virtualization reduces the dependency on hardware platforms and improves flexibility, scalability, and efficiency of the 5G Core network. Another important advantage that NFV has over dedicated hardware is the reduced development cycle, where hardware requires longer periods to be developed and manufactured, whereas software updates and patches can be deployed more quickly and efficiently (Condoluci and Mahmoodi, 2018). This leads to faster innovation and adaptation to future requirements.

Unfortunately, with these new advancements come new challenges for digital forensics and digital forensic readiness. Aside from the increased complexity of these new advanced

technologies, there is also the issue of NFV. With NFV, there are several new challenges for digital forensics, particularly for data management and recovery processes. Virtual environments can obscure where and how data is stored, making recovery processes more difficult. Some of these challenges discussed in (Sharevski, 2018) are that VNFs sometimes reside in different jurisdictions with different laws and regulations regarding digital forensics. Another issue with NFV is that it is volatile, with the network being dynamically altered in reaction to new conditions, such as increased traffic volumes. This dynamic nature can complicate the collection and analysis of data. Identifying the correct data at the right time for forensic purposes can be challenging.

Fortunately, these new technologies can also promote the integration of DFR, the main enabler being the virtualization of network functions in NFV and Software-Defined Networking (SDN). Since many of the functions of the 5G Core network are now independent of hardware platforms, it is much easier to develop a software-based solution for 5G by essentially adding digital forensic readiness functions to existing software. Something like this would be far more difficult if network functions were still using dedicated hardware, and would likely require new or additional hardware. DFR can also take advantage of the adaptability and scalability of NFV by reacting to the requirements of the network and scaling up forensic functions depending on traffic volumes. We can also take advantage of the flexibility of NFV by deploying "DFR enabled" VNFs to only suspected areas of the 5G network. This level of adaptability and targeted deployment is a significant advantage over traditional hardware-based solutions, which lack the flexibility and responsiveness of network function virtualization.

1.1. Problem Statement

5G technology marks a significant milestone in the evolution of telecommunications, promising unprecedented speeds, reduced latency, better energy efficiency, and higher data rates (Hajlaoui et al., 2020). With these advancements, the fifth generation of cellular technology is expected to revolutionize many industries. However, with these advancements, there are new challenges, particularly in cybersecurity. The increased complexity, scale, dynamic nature, and virtualized infrastructure are significant hurdles for digital forensics and digital forensic readiness. Previous strategies are no longer adequate or capable of addressing the complicated and dynamic nature of 5G networks with more complicated network functions, legality issues around jurisdiction, and the dynamic nature of 5G networks(Sharevski, 2018).

1.2. Research Questions

The research problem can be expanded using the following research questions. These questions are used to assess whether or not the solution is a successful digital forensic readiness solution.

1.2.1. What are the specific challenges associated with implementing digital forensic readiness in a 5G network environment?

This question is intended to address what the specific challenges are for DFR in 5G networks. Included in this question we want to find out what has been researched in the past and what these research papers have concluded about the DFR in 5G. By addressing

this question, we seek to provide a clear understanding of what needs to be done for a DFR solution to be effective in a 5G network environment.

1.2.2. What are the key requirements for a Digital Forensic Readiness solution in 5G Networks?

This question aims to address what the key components/requirements are for DFR in 5G networks. With this question we need to determine if our solution has the characteristics of a true DFR solution.

1.2.3. How can a DFR solution be tested and validated?

This question aims to address the most difficult challenge for developing a DFR solution for 5G networks. This research question aims to address whether or not we can set up a testing environment and solution that will be able to validate our solution. This is difficult since, as a student, we obviously do not have such access to an existing 5G network.

1.2.4. Can a DFR Prototype be developed to detect attacks in real time?

This question is with regard to the ability of a DFR solution in 5G networks and aims to address whether or not we can develop a prototype that can reliably detect common attack types in real-time.

1.3. Motivation

The motivation for this research is the lack of effective digital forensic readiness (DFR) solutions for 5G which could lead to significant security vulnerabilities, reduce incident response efforts, compromise legal compliance (Elyas et al., 2015), and result in substantial damage for organizations relying on 5G technology. This is especially important when you consider the dependence of critical infrastructure on 5G networks, (Dangi et al., 2021) where a vulnerability could potentially cause tremendous issues in existing industries such as healthcare and jeopardize new potential industries such as "self driving cars". This is more than possible where new vulnerabilities, from previous generations and 5G, are still being found (Shaik et al., 2019).

1.4. Research Objectives

The overall goal of this research is to develop and evaluate an agent-based digital forensic readiness solution to support digital forensics in 5G networks. This solution should enable real-time incident detection, efficient log collection, and effective forensic analysis, thus enhancing security and reliability without significantly affecting the performance, scalability, flexibility and efficiency of 5G networks.

1.5. Aims and Limitations

Unfortunately, the resources required to fully realize a 5G network to allow testing and development of a digital forensic solution are substantial. This is especially true for the RAN, where as a student I do not have access to technologies such as massive MIMO enabled towers. Well, it is impossible for me to develop a DFR solution for any existing 5G network. There do exist open-source projects such as Open-Air Interface, Open5GS, and UERANSIM that provide software that virtualizes the 5G RAN and provides 5G

Core network function implementations for research and testing purposes. However, even with this, it will still be difficult to fully realize a dynamic and feature-rich 5G network, due to the large resource requirements for hosting the many functions involved in a typical 5G network. Hence this research will focus on the 5G Core network with a simple configuration and a fully simulated RAN. Another limitation is that while a prototype should be able to comply with given legal requirements (with some configuration), we will not be incorporating any specific legal requirement as it is beyond the scope of this project due to the expertise and extensive resources needed. Instead, we will discuss how the prototype may be made legally compliant based on its configuration. As mentioned, the main aim of this project is to develop a DFR solution for the 5G Core Network, however on top of this, is the goal to have my solution be independent of the network function software used. This means that the solution should be general in that it can be applied to existing implementations without any drastic changes.

1.6. Methodology

The research methodology utilized is that of a combination of multiple research methodologies. This hybrid approach is required to address the complex problem being addressed, where there are many obstacles regarding limited resources and the complex fields that are digital forensics and cellular networking.

The following sections specify and explain the research methodologies we used throughout this research project.

1.6.1. Literature Study

In order to approach this highly complex issue that deals with the intersection of two different complex fields. I believe that a literature study will be needed to build a strong understanding of the requirements of digital forensics and digital forensic readiness. This literature survey includes researching current implementations of digital forensics in cellular networks, DFR requirements, 5G technology, and other solutions to take inspiration from.

1.6.2. Prototype

For the second research methodology, a prototype will be created. This will likely be a general prototype that can monitor and analyze data from the network functions used in core networks. By using a prototype we can demonstrate its function as proof, through testing. It will likely also provide a base for future research, perhaps using this prototype.

1.6.3. Experimentation and Evaluation

Simulation and testing will be conducted to evaluate the performance and effectiveness of the prototype DFR solution. By creating a controlled environment that attempts to mimic real-world 5G network scenarios, we can test the prototype under certain conditions to verify its functionality and identify its weaknesses. We will then analyze the results through critical evaluation to determine whether our solution was successful and where there is potential for future improvement.

1.7. Layout

This small section discusses the layout of the remaining sections and briefly explains their focus.

1.7.1. Part 1: Introduction

In this chapter we provided an introduction with research questions, methodologies, objectives and now the layout of the research project.

1.7.2. Part 2: Background

In the second chapter, we provide a literature survey on the current state of research. This chapter explores 5G technology, DFR and modern digital forensic methods used in 5G.

1.7.3. Part 3: Prototype Design, Implementation and Experimentation

Chapter 3: Prototype Design and Implementation Details - In this chapter we discuss details regarding the software, tools and concepts used in the prototype, the 5G Core Network and RAN.

Chapter 4: Prototype Experimentation and Insights - In this chapter we will discuss the final results of the experimentation that was done with the prototype and the 5G Core Network setup.

1.7.4. Part 4: Conclusion

Chapter 5: Evaluation - In this section we will provide an analysis of the prototype and its ability for real time detection, forensic evidence collection, and event reconstruction.

Chapter 5: Conclusion - This chapter provides a final summary of the research project and discusses some of the many possible additions that can be made in terms of future research.

2. Chapter 2: Literature Study

This literature study aims to explore and briefly explain referenced papers and other existing research on Digital Forensic Readiness (DFR) within previous and current generations of cellular networks. First, we will discuss Digital Forensics and Digital Forensic Readiness (DFR) and the requirements for DFR. Next, we examine current digital forensic processes in LTE and LTE-Advanced (4G). Lastly, we look at an existing model for DFR in NFV.

2.1. Digital Forensics

Digital Forensics (DF) is a specialized branch of forensic science and focuses on digital evidence stored electronically. The Book "Digital Forensics" by André Årnes (Årnes, 2017) defines digital forensics as the "identification, analysis, interpretation, documentation and presentation of digital information derived from digital sources" to facilitate or further event reconstruction. Event reconstruction is a crucial component of forensic science and is defined as "the process of identifying the underlying conditions and reconstructing the sequence of events that led to a security incident" (Jeyaraman and Atallah, 2006). By analyzing digital traces, artifacts, and other evidence, digital forensic experts are able to reliably create a narrative that aids in identifying causes, attributing responsibility, and establishing timelines or event windows. This process is especially essential in digital forensics, where data from various sources must be correlated to provide insights into complex events, security incidents, or cyber-attacks. In a 5G environment, this can be difficult due to the dynamic nature of NFV and the large, and complex amount of data that could be involved in particularly large deployments.

2.2. Digital Forensic Readiness

Digital Forensics Readiness (DFR) is essentially a proactive approach to Digital Forensics (DF), where the goal is to minimize the costs of DF by ensuring that organizations are forensically ready/prepared to respond to potential security incidents in a manner that is both effective and legally compliant.

2.2.1. DFR Requirements

Digital Forensic Readiness (DFR) is defined as having two objectives. The first is to maximize the usefulness of collected data, and the second is to minimize the costs of digital forensics when responding to an incident (Tan, 2001). These objectives highlight the overall goal of DFR but, can be broken down into a partial set of DFR requirements. DFR encompasses the capabilities needed for efficient and secure evidence collection, storage, and analysis in response to potential incidents. Based on (Tan, 2001) and (Makura et al., 2020), some of the essential DFR requirements we have identified include:

2.2.1 **Monitoring:** Monitoring corresponds to the first DFR objective of maximizing the usefulness of collected data by ensuring that relevant information is captured and retained. Monitoring is an important aspect of any DFR solution. By proactively monitoring various aspects of a system you collect crucial information that can be utilized in forensic investigations as evidence. However, it is also important the data collected is useful and formatted such that it can be analyzed quickly and effectively. Therefore, what information is collected should depend on the

environment (Tan, 2001). In the case of Networking, it may be best to collect information from firewalls, IDS systems, or in the case of 5G Networks, VNFs.

2.2.2 **Assurance:** Information assurance involves security measures such as encryption, access controls, and authentication protocols to protect data from unauthorized access or tampering. Information assurance is essential to ensure that information collected is admissible in court (Makura et al., 2020) and forms part of the first objective of maximizing the usefulness of the collected data. Furthermore, a chain of custody should be established to document the handling of evidence throughout its lifecycle. This includes documentation and timestamping of its creation, processing, and storage up until its potential presentation as evidence in court. This ensures that any evidence presented in court can be proven to be unaltered and reliable, thereby enhancing its credibility (Tan, 2001).

2.2.3 **Cost:** Cost is likely the most important aspect since one of the objectives mentioned is to minimize the costs of digital forensics. If a DFR solution does not provide the value to justify its cost then it is a failed solution. This takes into account the usefulness/value of the collected data and the resource costs of collecting such data. The resource costs may include the actual costs of the hardware/software used, the computational costs of running such hardware/software, and the effect that the solution has on other components of the business. Another important aspect of cost is time, the "Honeypot Project" referenced by John Tan (Tan, 2001) showed that investigators spent around 80 hours analyzing a compromised system. However, the attackers only took 2 hours to compromise the system. It is clear that a DFR solution should be lightweight and that DFR processes should avoid negatively affecting typical business operations while simultaneously reducing the time required to analyze the collected data.

2.2.4 **Compliance:** Regulatory Compliance is an especially important requirement for a DFR solution in 5G networks. There are many aspects to regulatory compliance that encompass privacy law, and government-mandated data retention requirements that vary based on jurisdiction. For example, Section 14: "Retention and restriction of records" of the Protection of the Personal Information Act (POPIA) states that "records of personal information must not be retained any longer than is necessary unless the responsible party reasonably requires the records for lawful purposes related to its functions or activities" (of South Africa, 2019). However, there are many other aspects such as disclosure and transparency. Personally identifiable information (PII) is one such aspect that needs to be considered, in particular IP addresses. According to the General Data Protection Regulation (GDPR), IP addresses are considered online identifiers and are classified as PII (Regulation, 2016). PII requires further consideration regarding its collection, use, and disclosure. Any DFR solution should be created such that it complies with any regulations that are applicable or can be configured to do so.

2.2.2. ISO/IEC 27043

When considering DFR Requirements we should also consider the ISO/IEC 27043 standard (Valjarević et al., 2016) (International Organization for Standardization, 2015). The standard provides many guidelines based on idealized models for conducting incident

investigations involving digital evidence. These guidelines can be used to attain DFR in any environment. While this standard is not specified for 5G Networks, it does however provide a systematic approach for DFR that can be applied to 5G Networks. This standard consists 5 classes of digital forensic investigation processes:

1. Readiness Processes
2. Initialization Processes
3. Acquisitive Processes
4. Investigative Processes
5. Concurrent Processes

The class that is relevant for DFR is the class of Readiness Processes which consist of activities designed to ensure an organization is well-prepared to collect and preserve digital evidence when a security incident arises in a cost-effective manner. Hence a DFR solution should both maximise the potential of data collected and minimize the cost associated with security incidents and investigative processes.

In this next section, we discuss one of the current mechanisms for digital forensics in previous generations of cellular technology.

2.3. Digital forensics in LTE and LTE-Advanced

LTE (3.9G) and LTE-Advanced (4G) are the two standards for cellular networks that came before and were built upon by the current fifth generation (5G) (Hajlaoui et al., 2020). The main mechanisms utilized for digital forensics in LTE and LTE-Advanced networks are "Lawful Interception" (LI) and "Lawful Access Location Services" (LALS) (Sharevski, 2018). LI is legally authorised process by which a network operator gives a "Law Enforcement Agency" (LEA) access to communications flowing through the network, with the purpose of investigating criminal activities (Thorogood and Brookson, 2007; Sharevski, 2018). LALS is the legally provisioned action performed by a network operator(s) to make location-based information available to a LEA (Sharevski, 2018).

2.3.1. Lawful Interception (LI)

The LI process in LTE and LTE-Advanced typically flows as such. First the "target identity" is established for the LI by the LEA, this is typically a victim or a suspected user/service. The LEA then sends an LI request, with the target identity, time period and delivery information to the "Administrator Function" (ADMF) of the network. The ADMF then provisions the LI to the "Interception Control Elements" (ICEs) and then configures the IP addresses of the interfaces specified in the delivery information of the LI request. The ICEs then intercept the specified traffic to forward it a delivery function that delivers the intercepted traffic to the LEA (Sharevski, 2018). This allows the LEA to monitor and analyse the traffic of the target identity as part of a potential digital forensic investigation.

2.3.2. Lawful Access Location Services (LALS)

Cellular networks also provide location services that require the location information of the user. This information can be very useful for the LEA conducting a digital forensic investigation. This is why cellular networks have "Lawful Access Location Services" that provide locational data to LEAs. The process is very similar to LI, where an LEA specifies a target identity, time period and delivery information. Then the information is intercepted from localization requests made by the target identity to location services which in turn forward the request to the LEA. LALS can be crucial for a digital forensic investigation (Sharevski, 2018).

2.4. LI and LALS in 5G

Many of the technologies that bring about the advantages of 5G also present challenges for digital forensics, this is true for both LI and LALS which needed to be adapted for 5G. These next subsections discuss how LI and LALS are affected by the 5G technologies of NFV and network slicing.

2.4.1. LI and LALS with NFV

NFV replaces specialized hardware with "virtualized network functions" (VNFs), offering flexibility, scalability, and efficiency. However, NFV introduces challenges for both LI and LALS. The distributed nature of NFV allows VNFs to be located across various regions, complicating jurisdictional issues for LEAs (Sharevski, 2018). Additionally, NFV's shared architecture requires sensitive functions involved in LI/LALS to be trusted and properly isolated to prevent "cross virtual network side-channel attacks" (Farahmandian and Hoang, 2016).

2.4.2. LI and LALS with network slicing

Network slicing is one of the new technologies utilized in 5G networks, and is enabled by NFV. Network slicing is the partitioning or "slicing" of a single physical network into multiple virtual networks that can be specialized for certain applications, services or users(Zhang, 2019). Some of these virtual networks can be provisioned for users as private networks managed by third parties, this introduces issues for LI and LALS in regards to trust and authorization as these third parties are not subject to national regulation like network operators are. This requires the LEAs to establish trust relationships with these private third parties to ensure they can authenticate and verify the integrity of intercepted data (Sharevski, 2018).

2.5. Limitations of LI and LALS

Well LI and LALS are effective and have been adapted for 5G, they are still a reactive approach for digital forensic investigations. This is due to the fact that a target identity must be specified and interception must be requested by the LEA, which is a reactive approach. Well there do exist more proactive approaches such as anomaly detection systems, they however are not standardised for cellular networks.

2.6. Potential for DFR in 5G

Fortunately, there is the potential for a digital forensic readiness solution, by adapting a documented "Intrusion Detection System"(IDS) that was modeled for an NFV architecture in (Behnke et al., 2019). An IDS is a system that is used in cybersecurity to automate the process of monitoring the events occurring in a computer/network and analyzing them for potential violations (Scarfone et al., 2007).

The model presented in (Behnke et al., 2019) utilizes an agent-based approach whereby an IDS system, in this case, "Suricata" (an open-source based IDS), is attached to network interfaces (could potentially be VNFs). This IDS would then monitor and analyze traffic flowing through each interface. Attached to each IDS is a data shipper called "Filebeat" which collects the logs generated by the IDS and ships them to a central database collection endpoint. Both the IDS and the data shipper comprise a single VNF that is attached to an interface. Once the data is filtered and stored in the database, it is analyzed for unexpected messages and, if found, will trigger an alert/action based on the message. Another similar agent-based approach for DFR has also been researched for cloud computing (Kebande and Venter, 2018), whereby agents are installed on cloud-hosted virtual machines. Like the previous case, these agents forward data to a central location for analysis.

2.7. Existing DFR Model

In this section, we discuss an existing model for Digital Forensic Readiness (DFR) in Network Function Virtualization (NFV) presented in (Makura and Venter, 2024) that serves as the basis for this research. The paper discusses a model that involves an agent-based approach for the forensic collection and storage of Potential in a database. The high-level model is comprised of six phases:

1. Prototype deployment: In this phase the prototype is deployed of to VNFs operating on top of NFV, the prototype then monitors and proactively collects and ships PDE.

2. Digital evidence collection: In this phase the PDE is collected in a proactive manner, in real time, from the 5G VNF instances.

3. Digital evidence preservation: This phase involves ensuring the integrity of collected data through hashing.

4. Digital evidence storage: Storing the collected data securely and systematically, allowing for efficient access and retrieval during investigation.

5. Digital evidence storage: This phase involves storing the collected hashed digital data in a forensic database for later analysis.

6. Digital evidence analysis: In this phase, the data is validated (using the hash values) and analyzed using various techniques.

7. Digital evidence reporting: This last phase, involves the documentation, reporting and presentation of findings produced in phase 5 in a legally admissible manner.

While this model does provide a basis for this research, no prototype was fully implemented, hence the focus of this research is on the development of such a prototype that addresses the first four proactive phases: deployment, collection, preservation, and storage.

3. Chapter 3: Prototype Design and Implementation Details

In this chapter, we discuss the design and implementation details of the **5G Digital Forensic Readiness Tool** (5GDFRT), the prototype developed to meet our Digital Forensic Readiness (DFR) requirements. Firstly, we discuss the technologies used to host the 5GDFRT, this discussion includes how network functions are deployed and managed, along with the options available for managing DFR components. We then cover the software and configurations used for the 5GDFRT, including, the main components and their roles, and how each component contributes to fulfilling DFR requirements. Following this we will cover the 5G Core Network software used as part of our 5G Core Testbed.

3.1. Platform

In this section, we outline the underlying platform used to support the development, deployment, and management of our 5G Core Network and the 5GDFRT. Below, we discuss the specific components of OpenStack—DevStack, Tacker, and the way in which they may support our DFR requirements.

3.1.1. OpenStack

In order to host the 5G Core testbed, we utilized OpenStack. OpenStack is an open-source cloud platform used for deploying and managing infrastructure. It is ideal for managing the virtualized resources needed for a 5G Core network. Specifically, OpenStack offers compute (Nova), networking (Neutron), and storage (Cinder) services which operate as our Virtual Infrastructure Manager (VIM), enabling the deployment and management of virtualized network functions (VNFs) with OpenStack acting as our NFV Infrastructure (NFVi).

However, since this is a small-scale project with limited resources we decided to utilize Devstack. DevStack is a series of scripts and tools used to spin up a simple yet complete OpenStack environment based on the latest versions of every service specified from the git master. DevStack is intended for development, and research purposes, which makes it an appropriate choice for this research project. It provides the necessary infrastructure without the resources, overhead, and complexity required for a typical large-scale or production 5G Network.

The host machine used to host Openstack for this project has the following specifications:

- **CPU**: 11^{th} Generation Intel(R) Core(TM) i9-11900F 2.5 GHz
- **RAM**: 32GB
- **Storage**: 500 GB SSD
- **Operating System**: Ubuntu 22.05 LTS

Typically, a distributed environment with many machines and a significant amount of resources that span multiple regions would be used to host a full-scale, production 5G Network. However, for the purpose of this project, this setup should be adequate, given the limited resources available.

3.1.2. Tacker

With OpenStack as our NFVi, we utilized Tacker to manage the VNFs. Tacker is an OpenStack project that provides a VNF Manager (VNFM) and an NFV Orchestrator (NFVO) that can be used in combination with Neuron, Nova, and Cinder (VIM) to deploy, manage, and operate Network Services and Virtual Network Functions (VNFs) on an NFVI platform like OpenStack.

For this project, we created a single Virtual Network Function Descriptor (VNFD) for each network function as well as for some of the other necessary components. A VNFD is an information model that describes how a Virtual Network Function (VNF) should be deployed, configured, and managed within an NFV environment. Tacker VNFDs are based on the ETSI NFV-SOL001 v2.6.1 standard (ETSI, 2023). These VNFDs allow for multiple versions or "flavors" for alternate definitions of each network function. For this project, we created two variants: DFR enabled (with DFR processes) and DFR disabled (without DFR processes). This feature is particularly useful for addressing the issues mentioned in Chapter 2 regarding jurisdiction by essentially only deploying enabled variants in areas/jurisdictions with the required authorizations for DFR.

In this next section, we discuss the implementation details of the 5GDFRT, including the details of the tools and software used and how each is configured to meet DFR requirements.

3.2. 5GDFRT Implementation

The goal of the 5GDFRT is to proactively retrieve and store useful information in such a way that it can be effectively utilized for digital forensic investigations and real-time detection.

The 5GDFRT itself is rather simple and uses many popular tools, concepts, and protocols to securely collect information that is both useful and does not require integration with existing 5G Core software. This simple design has two main components: the **Agent** and the **Log Server** each with its own smaller components. These next two sections discuss each component in detail and how they contribute to our DFR solution. After that, the security features implemented for information assurance and data protection are discussed.

Figure 1 provides a basic visualization of the prototype and its components:

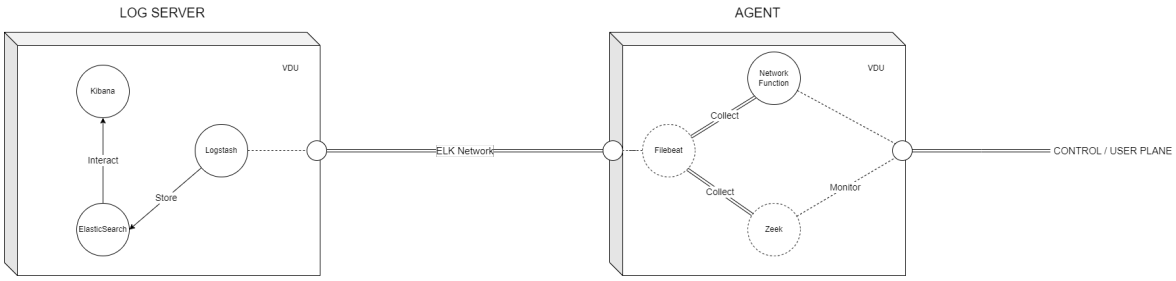


Figure 1: 5GDFRT Prototype

Figure 1 illustrates only one Log Server and Agent, however, you would typically have many agents all connected to one (or potentially many) log servers.

3.3. Log Server - ELK Stack

The role of the Log Server is to securely ingest, store, manage, analyze, and visualize the collected log data. To facilitate this, we made use of the popular ELK stack. The ELK stack is a collection of three open-source projects, namely Elasticsearch, Logstash and Kibana.

The ELK Stack is a centralized logging solution that is designed to manage and provide analytical tools for logs from distributed sources. Below, the role of each component is discussed in detail:

3.3.1. Elasticsearch

Elasticsearch is the core search and analytics engine in the ELK Stack and is based on Apache Lucene (Białecki et al., 2012). Elasticsearch is a document-oriented database, a specific type of NoSQL database designed to store, retrieve, and manage documents, in this case, JSON documents. Which is perfect for handling the many different and varying types of logs that can be produced by network monitoring software. Elasticsearch stores and indexes all log data, making it easily searchable. Elasticsearch is also designed to be distributed with many nodes, providing high availability and data replication through data sharding. However, for the purposes of this project, we only utilized a single instance.

3.3.2. Logstash

Logstash is responsible for ingesting the data collected and forwarded by agents. Once received, it then transforms or filters the data if needed, and then forwards it to Elasticsearch. Logstash can be used to preprocess and filter data such that data can be searched more effectively and efficiently. By offloading the preprocessing workload, Logstash helps minimize the processing burden on both the agents and Elasticsearch, enabling smoother and more scalable operation, especially in environments with high data throughput like 5G Networks. For this project, we utilized Logstash to create a pipeline that separates data collected by each agent into separate indices as well as adding fields that indicate when the data was received. This provides a useful mechanism to efficiently and effectively query data collected by each agent.

3.3.3. Kibana

The role of Kibana is to provide a user-friendly platform for interacting with the search and analytics engine (Elasticsearch). Kibana provides a web-based interface for visualizing and interacting with data. Key features include creating customizable dashboards and analyzing trends. Kibana also facilitates some of the more advanced features of Elasticsearch such as configuring machine learning tasks and setting up alerts. For this project, we utilized Kibana primarily for evaluation purposes and used many of the analytical features to search for patterns and create dashboards that capture trends.

The ELK Stack (Elasticsearch, Logstash, and Kibana) is widely regarded as one of the most advanced and comprehensive centralized log collection and analysis solutions available today. In this section, we have only scratched the surface of its capabilities, which include powerful full-text search, real-time analytics, flexible data visualizations, advanced machine learning with anomaly detection, and many other features.

The real benefit of the ELK stack is that it is designed for distributed environments. While, for this project, we only utilized a single Elasticsearch and Logstash instance, ELK can support multiple instances that are specialized for different tasks and can be scaled according to such tasks. This distributed approach could be leveraged to efficiently handle the high volume of information that may be collected from a complex 5G network.

3.4. Agent

The Agent is responsible for monitoring, harvesting, and shipping potentially useful log data to support the storage and analysis of network events, including interactions between network functions and user traffic. Each Agent is installed on a Virtual Deployable Unit (VDU) hosting a network function and is configured to monitor both inbound and outbound traffic associated with that network function. The next few sections discuss the software components that make up the Agent.

3.4.1. Zeek

Zeek is an open-source software network analysis tool that is commonly used as a network security monitor (NSM) to support investigations of suspicious or malicious activity. In this case, Zeek is positioned to analyze and monitor traffic flowing through a network interface, specifically, the network interface utilized by the network function that it is monitoring. Zeek is capable of monitoring and capturing detailed information related to TCP/UDP/ICMP connections. It then stores this information in various detailed, highly structured JSON log files.

Zeek produces many log files that capture multiple aspects of different types of traffic, however, the log files that were deemed most important are:

(a) **conn.log**
Description: The conn.log or connection log is one of the more important logs produced by Zeek. It captures metadata from each observed connection, including originating/responding IP addresses and port numbers, timestamps, connection state, protocol (TCP, UDP, ICMP), duration, and the service involved (e.g., HTTP, SSL,

DNS, other). This log is essential for understanding network behavior and is particularly useful for forensic analysis, incident response, and identifying anomalies in connection patterns. By analyzing the connection log, you can detect unauthorized access attempts, track the flow of data, understand the interactions between VNFs, and correlate events across different logs to establish a comprehensive picture of network activities.

(b) **http2.log**
Description: The HTTP/2 log captures metadata from HTTP/2 connections. Each log entry includes request/response headers, URI, URI parameters, and status codes. This log is one of the most useful logs, as 5G Core network functions that are 3GPP-Compliant communicate using HTTP/2. So by extracting the URI and URI values from HTTP/2 requests/responses, it can provide tremendous value with contextual information that can be used for event reconstruction.

(c) **http.log**
Description: The HTTP log collects metadata from HTTP/1.1 requests. As stated, 3GPP-Compliant network functions communicate using HTTP/2 or HTTP/3. However, collecting HTTP/1.1 logs can still be useful for non-standardized VNFs or perhaps providing more details on user traffic. Like http2.log, each log entry includes request/response headers, URI, URI parameters, and status codes.

(d) **stats.log**
Description: This log provides information regarding the performance of Zeek itself but is still useful, particularly for providing an overview of the current state of the network. Each log entry includes total packets processed/dropped, capture loss (packets that Zeek decided to, or could not process), and CPU/memory usage, total TCP/UDP/ICMP connections (alive/not). Unlike the other log types, the stats.log is generated at a configured frequency (every 300s). This log is particularly useful for detecting DoS attacks as it provides an overview of the total traffic that Zeek has processed within a specified period. So if there is a much larger than normal amount of packets being dropped we might be able to infer a DoS attack, by looking at the number of active connections for each traffic type we may even be able to conclude the type of DoS attack.

(e) **ssl.log**
Description: The ssl.log captures detailed information related to SSL/TLS handshakes and connections. It can be used to help analysts understand encrypted communications, and detect possible malicious activity. Each entry corresponds to a specific SSL/TLS connection or handshake event and includes several important fields that describe the session including, timestamp, the origin/responder IP address, the cipher algorithm used, and the server name (e.g example.com).

(f) **dns.log**
Description: The Domain Name System (DNS) log is one of the most important sources of data generated by Zeek. The log captures application-level name resolution activity. DNS logs can help identify suspicious or malicious domains that may be associated with malware, command and control servers, or phishing sites. By

correlating DNS queries with other logs, investigators can piece together the timeline and nature of an incident, providing context for suspicious behavior. This log is particularly useful for analyzing user traffic. The information this log captures includes timestamp, origin/responder IP address, response code, and the query name.

(g) **notice.log**
Description: This log captures alerts and notifications generated by Zeek when certain predefined conditions or patterns are met. This log is useful for highlighting potentially suspicious activities and security incidents that may require further investigation. Each entry typically includes information such as the timestamp, the origin/responder IP addresses, event type, and a description of the alert or message. This information can help forensic investigators to analyse security incidents effectively.

(h) **weird.log**
Description: This log captures unusual or anomalous events that may not fit typical traffic patterns or behaviors. These unusual or anomalous events could indicate potential issues, misconfigurations, or even malicious activity. Each entry includes information such as the timestamp, the origin/responder IP addresses, a UID that references the connection, and a short description of the unusual behavior. The information can serve as a valuable resource for investigators looking to identify anomalies that may warrant further investigation.

An important feature of Zeek is that it generates unique identifiers (UID) for each connection/event logged. This provides a useful mechanism for forensic investigators to analyze and correlate data across multiple different log entries and types. Effectively speeding up the forensic process, thus reducing cost. For example, a UID can be used to link HTTP requests in the http.log with its corresponding connection in the conn.log or perhaps a weird.log entry which is particularly useful. Each ID can serve as a reference point that links multiple log entries related to a single session or event recorded by a particular agent, which can aid in event reconstruction.

The highly structured and well-labeled log data produced by Zeek enables more effective and efficient real-time analysis, offering capabilities that standard application logs generated by network functions typically cannot provide. Zeek's logs are highly organized, with clearly defined fields for timestamps, source and destination IP addresses, protocol types, and unique identifiers for each network event. This structured format allows for effective parsing and indexing, making it possible to detect anomalies, track session flows, and identify suspicious patterns almost immediately as data is ingested. In contrast, application logs generated by network functions are often not standardized, lack uniformity and require additional complex preprocessing. Application logs also often require the context provided by surrounding log entries to be understood, making them even less reliable for real-time analysis. In comparison, Zeek logs provide consistent, self-contained details of each network event in a single log entry, making them far more reliable, actionable, and useful for effective real-time analysis.

One of Zeek's most valuable features is its extensibility. It offers a powerful scripting

language that enables developers to create custom rules and detection mechanisms. This customizability can be leveraged to meet specific needs and adapt to evolving network environments. A prime example of this extensibility is the HTTP/2 log. Although HTTP/2 is not supported by the base version of Zeek, a plugin called bro-http2, which uses the nghttp2 C library, extends Zeek's capabilities to monitor the more complex HTTP/2 traffic. This highlights how Zeek may be customized to add advanced features tailored to individual Virtual Network Functions (VNFs).

Furthermore, this flexibility and customizability can be leveraged to address jurisdiction and regulatory compliance challenges mentioned in Chapter 2. By configuring Zeek scripts for each agent, you can ensure that they comply with any applicable legal requirements. For example, in a highly distributed 5G Core Network with network functions deployed across different regions, Zeek scripts can be defined to align with the specific regulations of each region. This approach also extends to tools like Elasticsearch and Logstash, where different data retention, disclosure, and access policies may be set for each Log Server, ensuring that compliance is maintained across the entire system. This region-based configuration can be facilitated by Tacker's VNFD "flavors" by having a dedicated flavor for each network function for each region.

3.4.2. Filebeat

Filebeat is a lightweight, open-source data shipper that is part of the Beats family and is specifically designed for harvesting and forwarding log entries from log files. It integrates seamlessly with the Elastic Stack (ELK), making it a great choice for this DFR solution. Filebeat is responsible for securely harvesting, parsing, and hashing logs collected from both Zeek and the network function . Filebeat reads log files line by line and ships them to a designated output, however, it also allows for additional preprocessing steps. Recognizing the importance of data integrity in any DFR solution, we configured Filebeat to utilize a hashing function (SHA-256) that generates a digest based on the original log entry. This digest serves as a fingerprint of the log entry, allowing for later verification that the logs have not been tampered with. This mechanism for validation is crucial for forensic investigations, where the integrity and authenticity of the logs must be maintained. Filebeat was also configured to decode and parse the log entry into a format that is suitable for real time analysis.

The ID and type of each VNF are also included in the log data to be transported. This will provide the necessary context for the documents and facilitate simple correlation during analysis, effectively speeding up an investigation. The basic structure of the final document shipped by Filebeat is shown in figure 2.

```
{
    "fields": {
      "digest": <DIGEST>, % Digest of original log entry
      "source": <LOG TYPE>, % Log source type (connection/dns/...)
      "vnf": <VNF_TYPE>, % Type of the VNF
      "id": <VNF_ID> % Unique ID for VNF instance
    },
    "input": {
      "type": "log" % Type of input
    },
    "agent": {
      "ephemeral_id": <AGENT_ID>, % Unique ID for agent
      "name": "filebeat", % Name of the agent
      "version": "8.15.0", % Version of the agent
      "type": "filebeat", % Type of the agent
      "id": <AGENT_ID> % Unique agent ID
    },
    "log": {
      "offset": 2500, % Offset in the log file
      "file": {
        "path": "path/to/log/file.log" % Path to the log file
      }
    },
    "message": "{...}", % Log entry (truncated)
    "@version": "1", % Version of the log format
    "@timestamp": "YYYY-MM-DDTHH:MM:SSZ", % Timestamp (When Filebeat processed it)
    "event": {
      "original": "{...}" % Original log entry (truncated)
    },
    "zeek/application": {
      "timestamp/ts": 1729950669 % Timestamp (When Zeek/NF generated the original log)
      "field": "value" % Decoded/Parsed JSON field:value pairs
       ...
    },
    "host": {
      "name": <HOST_NAME> % Host name of the agent
    }
}
```

Figure 2: Document Structure

3.5. Data Flow

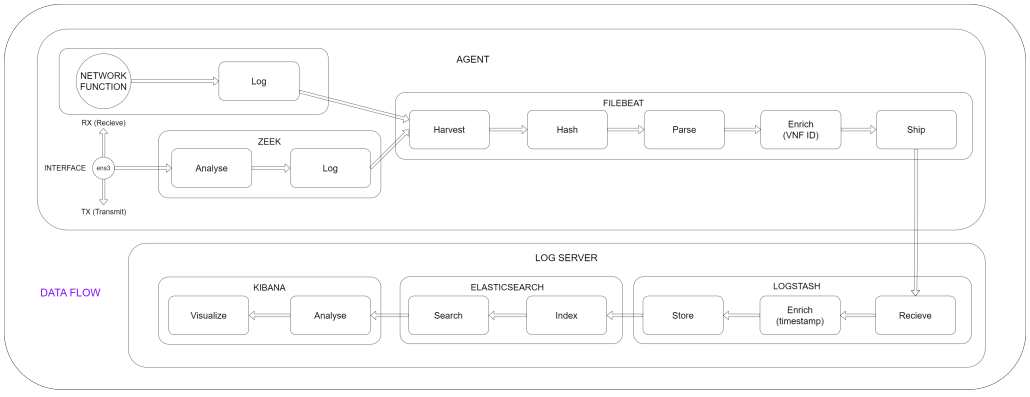


Figure 3: Log Data Flow

Figure 3 illustrates the basic flow of log data in the 5GDFRT. Firstly, the data is generated by Zeek or the network function, it is then harvested, hashed, decoded, and shipped by Filebeat. Next, it is received, timestamped, and sorted/stored by Logstash. Elasticsearch then indexes and facilitates analytical tasks. Finally, Kibana provides the mechanism to interact, analyze, and visualize the data. Some steps, such as authentication and encryption, discussed in the coming sections are not included in figure 3 for brevity.

3.6. Chain of Custody

The use of identifying information and timestamping shown in 2 establishes a chain of custody for each document. This process includes several critical components:

- "zeek.ts" and "zeek.uid": Indicate the original timestamp when the log was generated by Zeek and provide a unique identifier for the specific connection.
- "agent.id" (Filebeat agent ID) and "fields.id" (VNF ID): Identify which Virtual Network Function (VNF) and which Filebeat agent harvested and processed the log entry.
- "@timestamp": Marks the time when the log entry was processed by the Agent.
- "@received_at": Records the time the log data was received and stored by Logstash. (Not shown in 2)

The combination of these fields effectively forms a detailed history of the log data that documents its entire lifecycle, from its generation to its final storage. By linking each log entry to specific components, we can effectively establish a chain of custody, ensuring that the integrity and authenticity of the data can be verified and traced.

3.7. Security

To meet both information assurance and possible regulatory requirements, multiple security features have been implemented that ensure Authentication, Access Control, Integrity, and Confidentiality of log data.

3.7.1. Authentication and Access Control

The ELK stack utilizes a Role-Based Access Control (RBAC) system to manage user credentials and permissions. For this project, we defined five users: logstash_internal for Logstash, kibana_system for Kibana, filebeat_internal for Filebeat, and elastic (superuser) each with unique credentials required to authenticate access. This ensures that all incoming log data, from Agents, is coming from authenticated sources. It also ensures that all users (elastic) are also authenticated before they can access the log data.

Additionally, two custom roles, logstash_writer (assigned to logstash_internal) and filebeat-writer (assigned to filebeat_internal), grant specialized privileges, such as restricted write access to designated log data indices, an important aspect highlighted in (Tan, 2001). This ensures that filebeat_internal (Filebeat) and logstash_internal (Logstash) are the only users allowed to store log data in specific indexes. However, communication and the transfer of log data between Filebeat and Logstash uses a more advanced and secure Public Key Infrastructure (PKI) (Boldyreva et al., 2007) for authentication, ensuring a secure exchange with mutual certificate verification.

3.7.2. Data in Transit - PKI

As mentioned, Public Key Infrastructure (PKI) is used for authentication and communication between Filebeat and Logstash and ensures the security of data in transit between Filebeat and Logstash. PKI facilitates both mutual authentication between actors and the encryption of log data in transit. PKI utilizes asymmetric encryption to secure the data transmitted between Filebeat and Logstash. This is done by encrypting the log data with Logstash's public key, this way only Logstash can decrypt it using its private key. This prevents unauthorized actors from intercepting and reading sensitive log information during transmission, an important aspect of regulatory compliance (preventing PII disclosure). Using PKI allows for mutual authentication, where both Filebeat and Logstash verify each other's identities through the exchange of digitally signed certificates, ensuring that each party is communicating with a legitimate and trusted counterpart. This significantly mitigates the risk of man-in-the-middle attacks.

3.7.3. Integrity - Hashing (SHA256)

As mentioned, Filebeat uses a hashing function (SHA-256) to generate a digest of the original log entry before shipping it, along with various other identifying fields to Logstash. Hashing is an important part of any forensic investigation, as it provides a way to ensure that the logs remain immutable. Any modification to the log entry would result in a different hash value, alerting system administrators to potential tampering or data corruption, essentially assuring that the data has not been changed. This is particularly crucial in any forensic investigation, where maintaining the integrity of log data is essential for legal compliance and trustworthiness.

3.8. Basic 5G Core Testbed

In this section, we will discuss the software, tools, and implementation details of the 5G Core Testbed used to evaluate our 5GDFRT. First, we discuss the software used for our 5G Core Network functions, followed by the details of the software used for simulating the Radio Access Network (RAN) and User Equipment (UE).

3.8.1. 5G Core Network - OAI

For this project, we decided to use network functions developed by the Open Air Interface 5G (OAI-5G) Core Network (CN) project. The goal of the OAI-5G project is to provide 3GPP-Compliant 5G Standalone (SA) core network implementations that are both flexible and extensible.

With OAI, the 5G Core is designed with a service-oriented architecture through the adoption of the new 3GPP defined SBA. In this architecture, a set of 5G Core NFs provide services to other NFs. For the interaction between network functions, one of these acts as a Service Consumer (Client), and the other as a Service Producer (Server).

For our implementation of the 5G Core Network we decided to utilize a basic configuration with a single network slice and minimal resource requirements.

1. **AMF:** The Access and Mobility Management Function (AMF)
2. **SMF:** The Session Management Function (SMF)

3. **UPF:** The User Plane Function (UPF)
4. **NRF:** The Network Repository Function (NRF)
5. **AUSF:** The Authentication Server Function (AUSF)
6. **UDM:** Unified Data Management (UDM)
7. **UDR:** Unified Data Repository (UDR)

Each of these network functions handles a specific aspect of the 5G Core Network, and work together to handle UE registration and authentication (AMF, AUSF), PDU session management (AMF,SMF), network function discovery, user data storage and querying (UDM,UDR), policy enforcement (SMF), user traffic handling (UPF), along with many other functions.

3.8.2. NF Intercommunication

One important detail that should be highlighted is the manner in which Network Functions communicate and how we can capture this. As mentioned, in this architecture, network functions operate as client and/or server when interacting with one another. For this communication, network functions often utilize HTTP(S), HTTP/2, and HTTP/3 protocols, as defined by the 3GPP. OAI Network functions specifically utilize HTTP/2 to communicate.

Capturing these interactions may provide useful information that can facilitate event reconstruction and by utilizing both Zeek (specifically the http2.log file) and the application logs, produced by the network function, we can effectively capture these interactions and events in great detail. Some of these events include:

- Network Function Registration
- Data Access (UDR)
- UE Registration
- UE Authentication
- PDU Session Establishment/Modification/Termination

These are only a few of the events that we can capture and by securely collecting and storing this information, we should be able to effectively facilitate event reconstruction in the case of a security incident.

3.8.3. RAN + UE Simulation

As mentioned, the focus of this research is DFR in 5G Core Networks. However, to test and evaluate our 5GDFRT, we need to simulate RAN and UE interaction with the 5G Core Network. Therefore, we utilized three tools: UERANSIM, Ostinato, and SOCKS5 to simulate both RAN/UE interactions and to generate traffic.

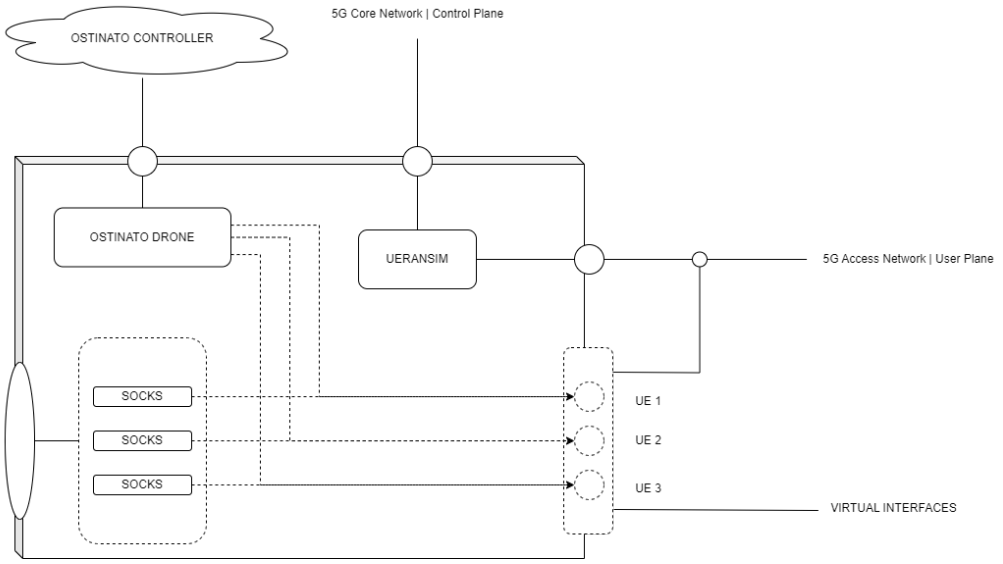


Figure 4: Simulating the RAN

As illustrated in 4, all these tools are hosted on a single VDU deployed via OpenStack through Tacker. The VDU is directly connected to the control and user plane through Neuron.

UERANSIM. UERANSIM is an open-source, state-of-the-art 5G UE and RAN simulator. It implements 5G Standalone UEs and a 5G Standalone RAN. It simulates a 5G-enabled device and a gNodeB as a base station. UERANSIM uses configuration files for the gNodeB and each UE to register each of them with the 5G Core Network (AMF, AUSF). UERANSIM provides an interface for managing each UE that allows the establishment and management of PDU sessions. A PDU (Protocol Data Unit) session is essentially a logical connection between a UE (a user device) and the data network. In order to provide access to the 5G Core network using the PDU sessions, UERANSIM creates virtual interfaces, each of which is connected to the data network. These virtual interfaces serve as networking endpoints, effectively allowing our traffic generation tools to utilize a UE's PDU session.

Ostinato Drone + Controller. Ostinato is a popular open source (up to version 1.3) traffic generation tool that can emulate and generate traffic for multiple devices using network interfaces, in this case, it uses the virtual interfaces created by UERANSIM to utilize a UE's PDU session. Ostinato provides detailed low-level control over the packets generated for each UE which allows us to generate many different types of traffic and control the amount and rate of packet generation. Ostinato is used extensively to evaluate our 5GDFRT prototype by generating many different types of malicious traffic. In Ostinato, multiple streams can be configured for each device. A stream is essentially a sequence of packets that you define and control for transmission over a network, it allows you to define the protocols used the number of packets transmitted, and the packet

transmission rate. Ostinato also has a "Drone" feature that allowed us to control packet generation externally, simplifying the process.

SOCKS Proxy. SOCKS5 Proxy server provides an additional method for traffic generation using externally hosted tools and applications. In this case, it was used in conjunction with Firefox and Python traffic generation libraries (Scapy) to generate typical browser traffic.

3.9. DFR enabled 5G Core configuration

A final view of the 5G Core network is shown in figures 5 and 6. Figure 5 illustrates a simplistic view of the 5G Core Network configuration, indicating which network functions interact with each other. The area highlighted in green represents the network slice, with the UPF, SMF, and NRF. The user and control planes are indicated by the green and yellow lines respectively. The red is used to represent the interaction and transmission of logs between the Agents installed at each VNF and the ELK stack/Log Server. Figure 6 illustrates a lower-level view of how the 5G Core Network, as defined in OpenStack. Here we can see that there are three separate Neuron networks: Control_Network, N3_Network, and the ELK network. Each Agent is connected to the Log Server (ELK stack) through the ELK network, and each VNF is connected to the 5G Control/N3 network. The figure also illustrates that Zeek is monitoring the interfaces utilized for the 5G Control/N3 networks.

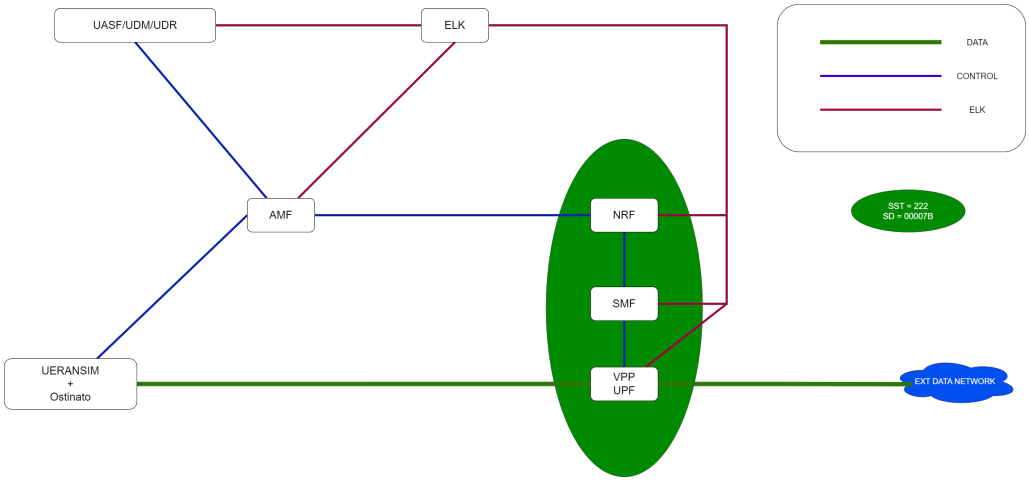


Figure 5: High level view

It is important to note the VDU marked EXT_DN (External Data Network) in figure 6, this VDU essentially forwards traffic between the UPF and the external network using IP tables. This node also served as the target for our simulated attacks during experimentation and evaluation.

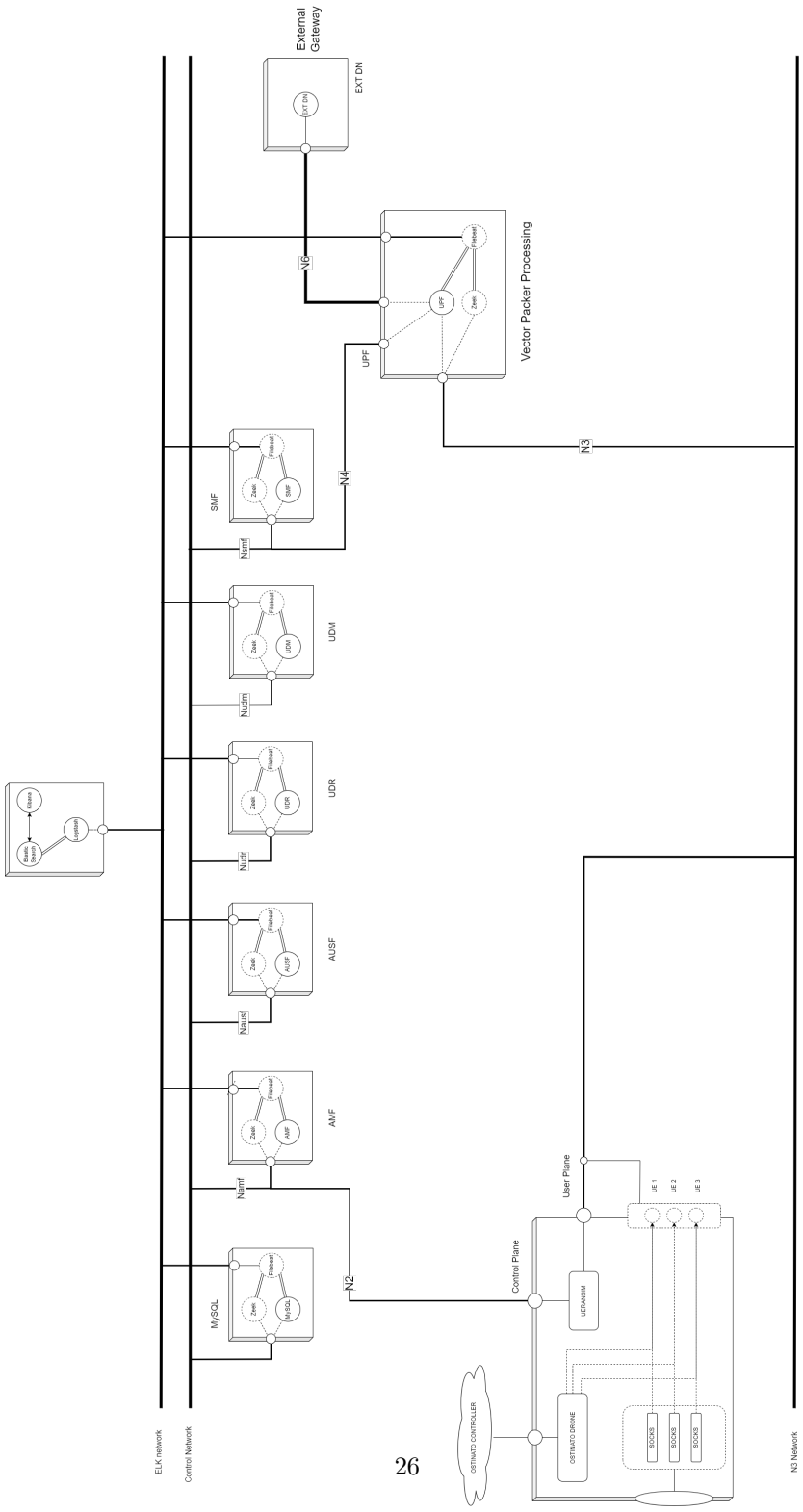


Figure 6: Low level view

4. Chapter 4: Prototype Experimentation and Insights

In this chapter, we discuss the experiments that have been conducted to evaluate our 5G Digital Forensics Readiness Tool (5GDFRT). These experiments are rather simple but should provide the results needed to evaluate the 5GDFRT's ability to monitor our basic 5G Core Network. There are two main aspects in which the 5GDFRT needs to be evaluated. Firstly, there is its ability to detect malicious behavior in real time, and the second is event reconstruction. This chapter is constructed as follows, first, we discuss the experiments and results collected to evaluate real-time detection capabilities. Then, we discuss event reconstruction, with details related to NF communication and application logs.

4.1. Real-time Detection

Two distinct scenarios were simulated using Ostinato. Each scenario models a common attack that the 5GDFRT should be able to detect and identify in near real-time. The target used for each attack is the EXT_DN VDU referenced in figure 6, it has been preconfigured with the IP address 10.7.0.4.

4.2. DoS Ping Flood Attack

Denial of Service (DoS) attacks are a major issue for the networks of today and have been utilized to extort businesses and prevent access to legitimate users. DoS attacks overwhelm their target's resources and render them unavailable to legitimate users, which can have serious implications for the business operations of an organization. One such kind of attack is a Ping flooding attack. A Ping Flood Attack utilizes Internet Control Message Protocol (ICMP) echo requests to overwhelm the target. Each time the target receives the echo request, it will use a little computing power to process it and generate an ICMP echo reply. A Ping Flood Flood Attack takes advantage of this by generating tons of malicious ICMP echo requests, often from multiple devices (Distributed DoS), in order to overwhelm the target by consuming bandwidth and computing resources (Kumar, 2006).

4.2.1. Experimental Setup

This Ping Flood Attack was simulated using Ostinato with 11 simulated UE devices. Figures 9 show the configurations used in Ostinato. It depicts the 11 virtual interfaces (uesimtunx) associated with each UE PDU session.

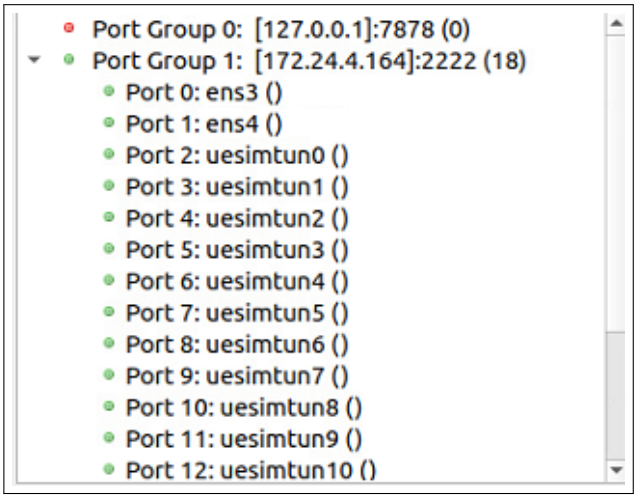


Figure 7: UE interfaces

A stream is configured for each interface (UE PDU session). The stream is configured to use 10.7.0.4 as the destination address and the source address is set to the corresponding PDU session IP allocated to each device. In the ICMP configuration, the ECHO request flag option is set, indicating that each packet is an echo request.

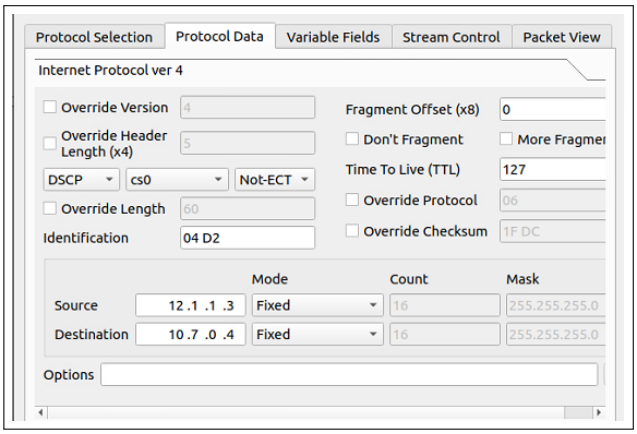


Figure 8: ICMP Setup

Finally, the number of packets and the packet rate are set, using 1 Million packets at a packet rate of 2000 thousand packets per second. Meaning each device/stream would run for a total of 500 seconds.

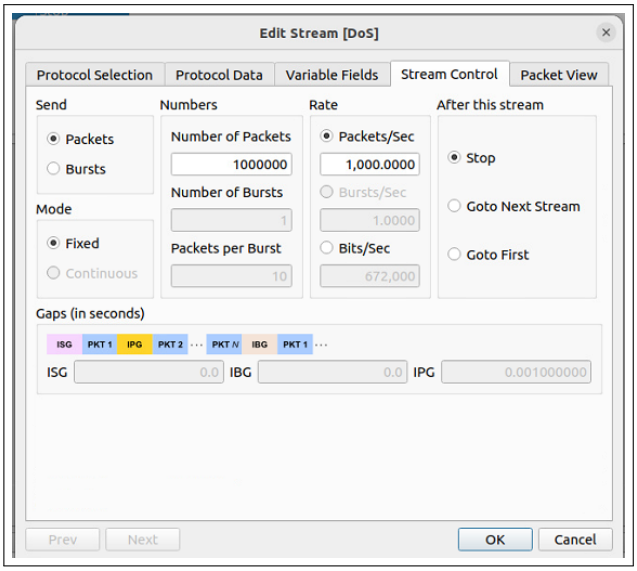


Figure 9: Packet Rate

4.2.2. Results

In order to best detect and visualize the collected log data we set up visualizations using Kibana. The best network function to analyze user-based attacks on other devices is the UPF function since it handles all user traffic in this configuration. However, if there are multiple network slices you would monitor each UPF allocated for each slice. For this visualization, we utilized the packets processed field of the stats.log produced by the Zeek.

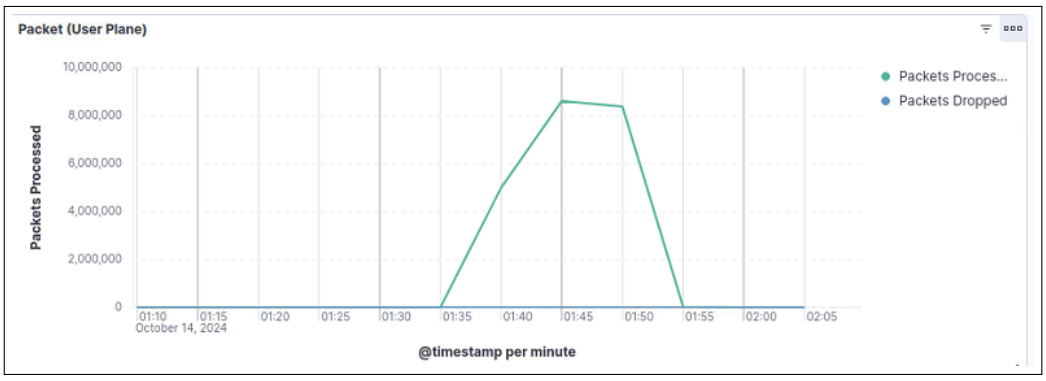


Figure 10: Packets Processed by UPF

As illustrated, in figure 10 the increase in traffic was clearly captured by the 5GDFRT, with almost 10 million packets being detected in the 300s intervals. You will notice a similar result by counting the number of ICMP connections generated in the conn.log.

This same strategy can be applied to other types of DoS attacks such as UDP flooding, and TCP SYN flooding (the connection state of connections in the conn.log may further indicate this type of attack).

4.3. IP Fragmentation Attack

The second attack scenario we simulated is an IP Fragmentation Attack. An IP fragmentation attack exploits how IP packets are fragmented and reassembled between devices. The attack involves manipulating offsets and sizes to trigger vulnerabilities of the victim/target host or even to evade security controls such as firewalls that would otherwise block traffic. IP fragmentation attacks avoid security controls by exploiting the fact that network devices and security systems often process each fragment independently without reassembling them for inspection, which can be used to avoid detection. They can also be used to exploit potential vulnerabilities in the target's IP packet processing mechanisms, potentially causing a crash, or some other undesired effect.

4.3.1. Experimental Setup

For this simulation we once again utilized Ostinato but, with a single UE device. We configured the stream to utilize a random offset value between 0 and 16 for each packet 11, essentially overlapping and creating gaps between the received fragments, simulating the behaviour of an IP fragmentation attack.

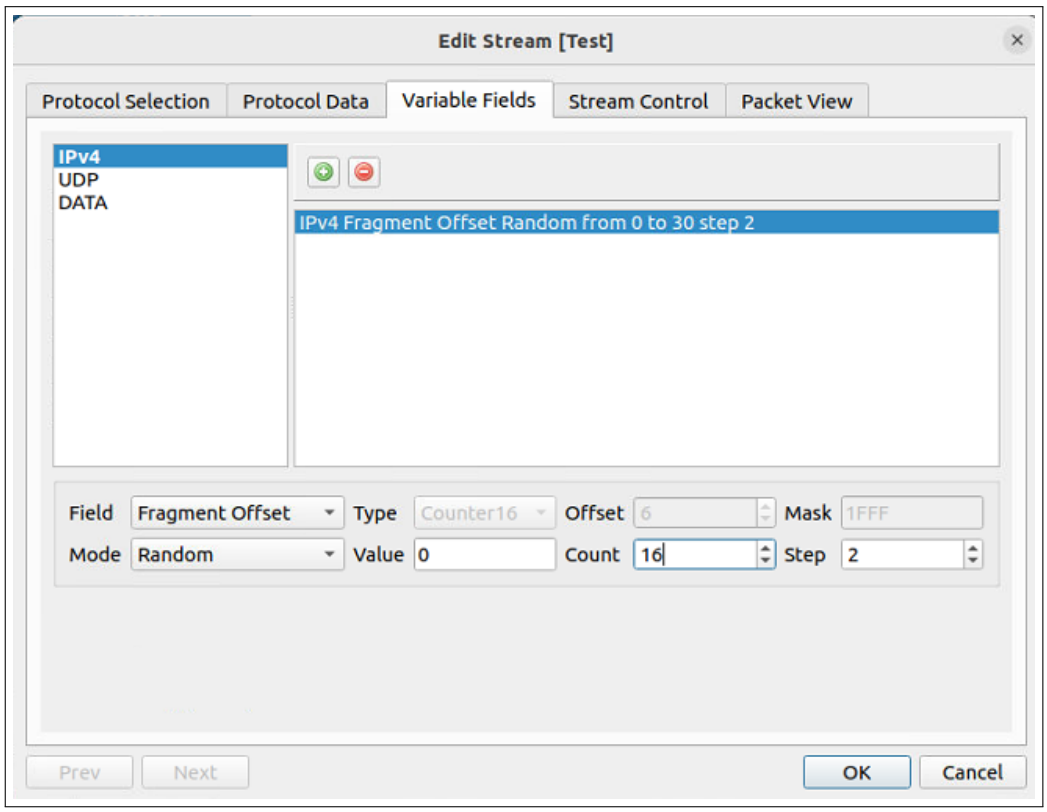
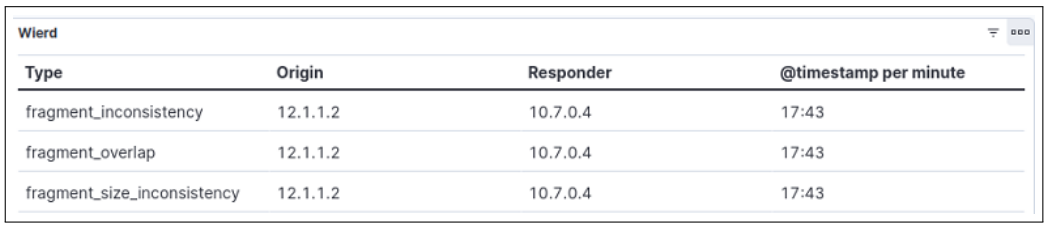


Figure 11: Random Fragment offset

4.3.2. Results

Once again, we can detect this attack by looking at the log data collected by the Agent positioned at the UPF, specifically the weird.log data. As mentioned, the weird.log captures unusual and potentially malicious behavior. This includes overlapping and gaps in the fragment values of IP packets. This is illustrated in 12, we can see that the weird log is reporting both fragment overlaps and fragment size inconsistencies. While this is not a definite indication that a fragmentation attack is occurring, if an unusual amount of such unusual behavior is present and continuous then it can be a definite indication that should require further investigation.



Wierd

Type	Origin	Responder	@timestamp per minute
fragment_inconsistency	12.1.1.2	10.7.0.4	17:43
fragment_overlap	12.1.1.2	10.7.0.4	17:43
fragment_size_inconsistency	12.1.1.2	10.7.0.4	17:43

Figure 12: IP Fragmentation alerts

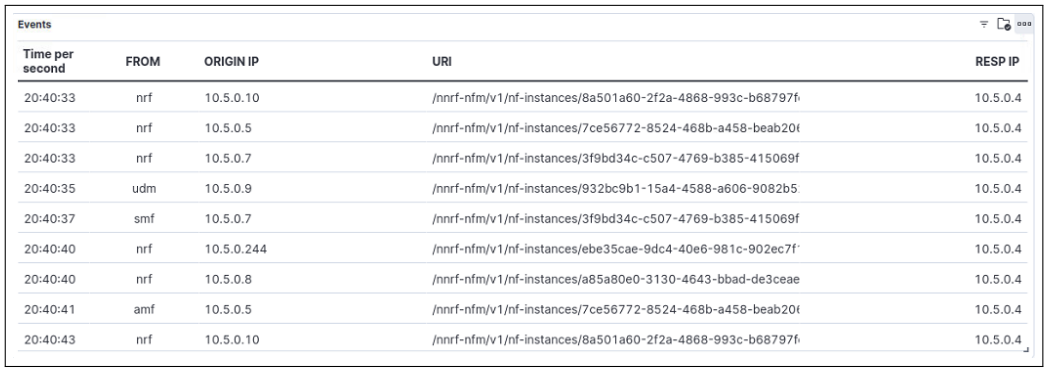
4.4. Event Reconstruction

As mentioned in Chapter 2, event reconstruction is one of the most vital processes in forensic investigations, and our 5GDFRT should be able to facilitate this. In 5G Core Networks, we can sort events into two main categories: Control Plane events, generated by interactions between network functions, and User Plane events, generated by user activity. Our 5GDFRT solution is able to capture both.

4.4.1. Control Plane Events

The Control Plane is responsible for managing signaling and orchestration between network functions, making Control Plane events a critical source of data for reconstructing events related to the state of the 5G Core Network. These Control Plane events we refer to are related to the interactions between these network functions. The most valuable sources of information for these events are the application and HTTP/2 logs (NF communication).

The HTTP/2 log data captured and shown in 13 provides essential details for tracing NF interactions. This includes fields like the origin and responder IP addresses, URI, and timestamps, which collectively offer insights into the timing and nature of these interactions.



Events

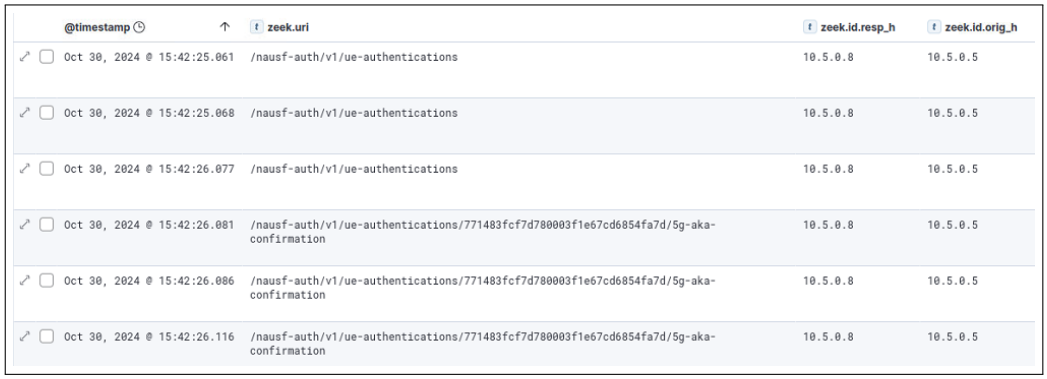
Time per second	FROM	ORIGIN IP	URI	RESP IP
20:40:33	nrf	10.5.0.10	/nnrf-nfm/v1/nf-instances/8a501a60-2f2a-4868-993c-b68797f	10.5.0.4
20:40:33	nrf	10.5.0.5	/nnrf-nfm/v1/nf-instances/7ce56772-8524-468b-a458-beab20	10.5.0.4
20:40:33	nrf	10.5.0.7	/nnrf-nfm/v1/nf-instances/3f9bd34c-c507-4769-b385-415069f	10.5.0.4
20:40:35	udm	10.5.0.9	/nnrf-nfm/v1/nf-instances/932bc9b1-15a4-4588-a606-9082b5	10.5.0.4
20:40:37	smf	10.5.0.7	/nnrf-nfm/v1/nf-instances/3f9bd34c-c507-4769-b385-415069f	10.5.0.4
20:40:40	nrf	10.5.0.244	/nnrf-nfm/v1/nf-instances/ebe35cae-9dc4-40e6-981c-902ec7f	10.5.0.4
20:40:40	nrf	10.5.0.8	/nnrf-nfm/v1/nf-instances/a85a80e0-3130-4643-bbad-de3ceae	10.5.0.4
20:40:41	amf	10.5.0.5	/nnrf-nfm/v1/nf-instances/7ce56772-8524-468b-a458-beab20	10.5.0.4
20:40:43	nrf	10.5.0.10	/nnrf-nfm/v1/nf-instances/8a501a60-2f2a-4868-993c-b68797f	10.5.0.4

Figure 13: NRF Events

When looking at the collected logs you may notice that all the entries have similar URI values and the same responder IP Address (10.5.0.4). This is because these are the

heartbeat signals that each network function sends periodically to the NRF to indicate its current status. These heartbeat signals are particularly useful for event reconstruction, as they establish a clear active lifetime for each network function.

However, if we filter these events out as in Figures 14 and 15, we notice some of the other important interactions between network functions.



@timestamp	zeek.uri	zeek.id.resp_h	zeek.id.orig_h
Oct 30, 2024 @ 15:42:25.061	/nausf-auth/v1/ue-authentications	10.5.0.8	10.5.0.5
Oct 30, 2024 @ 15:42:25.068	/nausf-auth/v1/ue-authentications	10.5.0.8	10.5.0.5
Oct 30, 2024 @ 15:42:26.077	/nausf-auth/v1/ue-authentications	10.5.0.8	10.5.0.5
Oct 30, 2024 @ 15:42:26.081	/nausf-auth/v1/ue-authentications/771483fcf7d780003f1e67cd6854fa7d/5g-aka-confirmation	10.5.0.8	10.5.0.5
Oct 30, 2024 @ 15:42:26.086	/nausf-auth/v1/ue-authentications/771483fcf7d780003f1e67cd6854fa7d/5g-aka-confirmation	10.5.0.8	10.5.0.5
Oct 30, 2024 @ 15:42:26.116	/nausf-auth/v1/ue-authentications/771483fcf7d780003f1e67cd6854fa7d/5g-aka-confirmation	10.5.0.8	10.5.0.5

Figure 14: HTTP/2 Events 1

Oct 30, 2024 @ 15:42:26.149	/nsmf-pdusession/v1/sm-contexts	10.5.0.7	10.5.0.5
Oct 30, 2024 @ 15:42:26.160	/namf-comm/v1/ue-contexts/imsi-208950000000034/n1-n2-messages	10.5.0.5	10.5.0.7
Oct 30, 2024 @ 15:42:26.165	/nsmf-pdusession/v1/sm-contexts	10.5.0.7	10.5.0.5
Oct 30, 2024 @ 15:42:26.169	/namf-comm/v1/ue-contexts/imsi-208950000000031/n1-n2-messages	10.5.0.5	10.5.0.7
Oct 30, 2024 @ 15:42:26.181	/nsmf-pdusession/v1/sm-contexts	10.5.0.7	10.5.0.5
Oct 30, 2024 @ 15:42:26.204	/nsmf-pdusession/v1/sm-contexts/1/modify	10.5.0.7	10.5.0.5
Oct 30, 2024 @ 15:42:26.242	/namf-comm/v1/ue-contexts/imsi-208950000000035/n1-n2-messages	10.5.0.5	10.5.0.7

Figure 15: HTTP/2 Events 2

In Figure 14 we are able to observe the UE registration and authentication process in action, specifically for three UEs. Like the heartbeat signals, this information can provide an indication of the window in which each UE is active. We can also see the PDU session establishment and modification requests in figure 15. This is useful for indicating the state of a UE's PDU session at a given point in time. What is even more interesting is the IMSI values depicted. IMSI values. IMSIs (International Mobile Subscriber Identifier) are unique IDs that identify mobile subscribers of networks by their SIM cards. These

identifiers are particularly useful as they can be used to identify the specific user involved in these requests.

The application logs collected also reflect these observed interactions. In figure 16 we can see the NF heartbeat signal request being processed and sent to the NRF. These logs provide even more context and value as they indicate the exact status (REGISTERED) of a given network function (AMF) at a given point in time (2024-10-29 18:56:42). By using timestamps we can even cross-reference these interactions between the Zeek and application logs, a particularly useful aspect for event reconstruction.

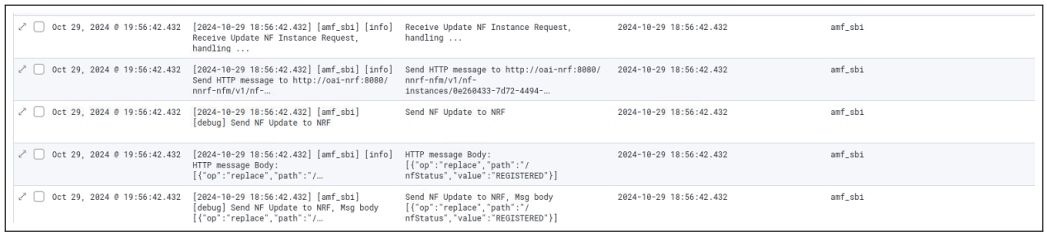


Figure 16: Application logs (AMF)

The application log data collected from the AMF shown in 17, lists all active UE (User Equipment) devices along with their details, including the IMSI, NGAP ID, AMF UE ID, Cell ID, and PLMN (Public Land Mobile Network) ID. The PLMN ID contains both the MCC (Mobile Country Code) and MNC (Mobile Network Code) for each UE. This is quite clearly a valuable source of information and provides a comprehensive view of each active UE's identity and location within the 5G core network. These attributes are highly valuable for event reconstruction, as they can be used by investigators to trace individual device activities and correlate network events. This level of detail is valuable in reconstructing the sequences of events related to UEs.

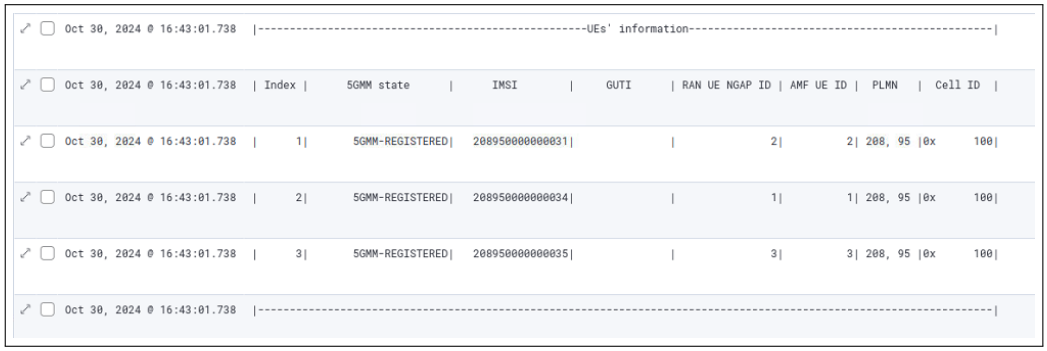


Figure 17: UE Device Table (AMF)

By combining the highly structured and well-labeled logs generated by Zeek with the detailed and valuable application logs collected from network functions, along with the

identifying and timing information provided by our 5GDFRT 2 3, we are able to provide a comprehensive and accurate framework for event reconstruction. This integration ensures that all relevant data including network traffic behaviors, user activity related to function-specific interaction, and network function interactions in general are captured with precise timestamps and unique identifiers (Zeek UID, IMSI, IP) enabling an efficient and seamless platform for the reconstruction of events across the 5G Core network.

4.4.2. User Plane Events

The User Plane events we refer to are activities related to the user's interaction with the network, such as data transmission, browsing, or application usage. These events are generated based on the data traffic exchanged between the User Equipment (UE) and the network, typically involving interactions with external services, websites, or perhaps other users. As discussed earlier, this user traffic flows through the User Plane Function(s) (UPF). Therefore, by analyzing the log data generated by Zeek at the UPF(s), we can capture and examine these User Plane events.

In order to test this we utilized the SOCKS5 proxy server, mentioned in 3.8.3.3, along with Firefox to generate browser traffic. It is important to note that the browser traffic was encrypted using HTTPS (a common situation), which meant that certain log types such as http.log and http2.log, which require unencrypted traffic, were not available. However, Zeek does provide a valuable log type specifically for this situation: the ssl.log. As mentioned in Chapter 3, this log captures information related to SSL/TLS sessions, allowing us to analyze secure communications and extract relevant data from encrypted traffic.

We can observe this in the tabularized ssl.log data collected from the UPF in figure 18, with the origin and responder IP addresses and the specific service involved in the TLS handshake or TLS session.

USER BROWSER

Time per second	ORIGIN IP	Service	RESP IP
21:20:19	12.1.1.2	www.google.com	142.251.47.164
21:20:26	12.1.1.2	incoming.telemetry.mozilla.org	34.120.208.123
21:20:27	12.1.1.2	fonts.googleapis.com	142.251.47.234
21:20:28	12.1.1.2	i.ytimg.com	142.251.47.150
21:20:29	12.1.1.2	accounts.google.com	66.102.1.84
21:20:30	12.1.1.2	googleads.g.doubleclick.net	142.251.47.98
21:20:31	12.1.1.2	fonts.gstatic.com	142.251.47.99
21:20:38	12.1.1.2	play.google.com	192.178.54.46
21:20:41	12.1.1.2	jnn-pa.googleapis.com	142.251.47.74
21:22:26	12.1.1.2	feeds.elastic.co	34.120.127.120
21:24:41	12.1.1.2	incoming.telemetry.mozilla.org	34.120.208.123
21:24:44	12.1.1.2	firefox.settings.services.mozilla.com	34.149.100.209
21:24:47	12.1.1.2	contile.services.mozilla.com	34.117.188.166
21:25:32	12.1.1.2	accounts.google.com	66.102.1.84
21:27:47	12.1.1.2	www.youtube.com	142.251.47.78

Figure 18: Browser SSL log data

Based on figure 18, we can see that the browser accessed different Google related services, including YouTube and Google fonts. It is clear that even for encrypted user traffic, the 5GDFRT solution was able to capture information that can aid in understanding what a user is doing at a given point in time. Furthermore, by combining and correlating this with the dns.log and conn.log data we can obtain an even greater understanding of a user's activities.

5. Chapter 5: Evaluation

This chapter focuses on the analysis of our 5GDFRT and how it meets the requirements presented in Chapter 2. First we will start with monitoring, followed by information assurance, followed by cost and regulatory compliance. After that, we will discuss some of the issues and potential solutions to these issues. Finally, we will discuss some potential enhancements that could be included in a future implementation.

5.1. Monitoring

As highlighted in Section 2.2 (Monitoring), the value of the data is crucial to DFR, and "What to log?" is an important question. For 5G Core Networks, it is clear that the data collected should be related to networking, traffic, and the state of VNFs. Zeek as a network security monitor combined with application logs proved to be perfect for this. Zeek managed to capture data that proved to be useful for both real-time detection and event reconstruction, while the application logs proved to be useful for capturing highly valuable and detailed information related to the state of both network functions and user devices, as illustrated in Chapter 4. Zeek's comprehensive logs allow analysts to easily correlate and identify unusual behavior effectively while the detailed application logs provided the missing application-level details and context that Zeek missed. And so by combining Zeek with application logs, we were able to collect and store useful data that facilitates a comprehensive and accurate framework for event reconstruction.

5.2. Information Assurance

The use of ELK user credentials and roles effectively handles access control and authentication between Elasticsearch and the other components/users, while PKI addresses authentication and encryption between Logstash and Filebeat. Hashing ensured the integrity of the original log entry, while the final stored documents effectively captured the chain of custody from the point the log was generated, to when it was harvested and shipped until it was finally received and stored with information indicating exactly which components were involved. It is clear that the 5GDFRT effectively meets the Information assurance requirement.

5.3. Cost and Regulatory Compliance

While both cost and regulatory compliance were considered out of scope for this project, the 5GDFRT was shown to have the capability to meet these requirements. With the lightweight and efficient nature of all the components involved, reducing costs and the extensibility and customisability of Zeek being able to operate under various regulations and performance requirements. While this does require further investigation, it is clear that a more comprehensive and abundant set of resources and expertise is required to effectively validate this.

5.4. Issues

5.4.1. Encrypted NF Communication

While OAI 5G CN currently does not support TLS for communication between network functions, its implementation is a part of the 3GPP standard. While Zeek does offer some limited TLS 1.2 decryption capabilities (given the required key material), it

is still not a supported feature. This degrades our ability to reconstruct events using HTTP/2 logs. That said Zeek is still able to collect useful information from both the ssl.log and the conn.log that are useful for capturing the interactions between network functions, however, they do not provide the same contextual value that the http2.log does.

However, there are ways to handle this issue, such as the use of man-in-the-middle (MitM) proxies that handle decryption allowing Zeek to analyze the decrypted HTTP/2 traffic. This unfortunately introduces additional resource costs and latency that may not be acceptable for 5G Core Networks.

Another approach demonstrated in (Wilkens et al., 2022), shows that TLS decryption using Zeek is feasible without disrupting live communications. This approach leverages client-side integration to selectively forward TLS session key material to Zeek, allowing for passive decryption. While this avoids the complexity and latency of MitM proxies, it introduces additional operational challenges. One issue is that key material must be forwarded by clients very quickly to ensure full decryption. This would then require network function integration and additional processing which may not be acceptable.

5.5. Potential Enhancements

5.5.1. Application Integration

While the logs generated by each network function do provide valuable information that can be used for event reconstruction, they do not provide the same detail and structure that allows for real-time detection and efficient analysis. This can be addressed if the 5G CN software is adapted to generate logs that are structured (JSON/XML), standardized, and rich in context. This in combination with the logs produced by Zeek would provide a more complete DFR solution.

5.5.2. Specialization

As mentioned in Chapter 3, Zeek provides a framework for creating customized scripts and configurations that can be applied based on the role and legal requirements of the network function it is monitoring. For example, the agent positioned to monitor the UPF may be under strict legal regulations that restrict the capture of certain types of user traffic. You can then configure Zeek to only capture information such as total packets processed/dropped and not HTTP or DNS requests. By specializing agents to comply with different legal requirements and VNF roles, we can effectively maximize the monitoring of the network while still complying with legal constraints.

6. Chapter 6: Conclusion

This chapter concludes the report by summarising the work done during the course of the research project. Firstly we start by revisiting the problem statement and research questions. Next, we will discuss the potential for future research with the 5GDFRT prototype in mind. Finally, the report will be concluded with a final summary of the research project.

6.1. Problem Statement and Research Questions Revisited

The goal of this research was to attempt to resolve the lack of Digital Forensic Readiness in 5G (Core) Networks. The problem was then expanded to include identifying the specific challenges associated with DFR in 5G. We answered this as part of the literature study and found that virtualization, jurisdiction issues, and high data volumes were the main challenges and that there was a lack of research in this area.

The second question asked what the requirements for DFR are. We once again answered this question as part of our literature study and found that DFR encompasses many requirements that can depend on the organization implementing 5G and the resources available. Ultimately, it was narrowed down to ensuring 4 requirements that could either be reasonably met or at the very least integrated with future research. While a 5GDFRT prototype was created that managed to address the data collection and information assurance requirements of DFR, some of the other requirements do require further research and incorporation for the 5GDFRT to be fully validated.

The third question was about how a prototype might be tested. We effectively answered this question by constructing our own basic 5G Core Testbed that allowed us to simulate various attack scenarios that allowed us to evaluate our 5GDFRT effectively. While this did effectively answer the question, given the resources available, it also became clear that further research requires a more complex and comprehensive 5G Core Network in order to validate some of the cost and regulatory compliance requirements. The final question was with regard to whether a prototype could be developed to detect common attacks. The answer to this is yes, even based on the smaller scale of this project, the data captured and the visualizations used demonstrated clear signs of DoS, and IP Fragmentation attacks.

Overall I do believe that we were able to achieve the desired research objective of developing a prototype that facilitates real-time incident detection, efficient log collection, and effective forensic analysis as demonstrated in Chapters 4 and 5. However, weather this solution significantly affects the performance and efficiency of 5G networks requires further research.

6.2. Further Research

This next section describes possible avenues for further research.

6.2.1. A more advanced 5G Testbed and greater resources

While admittedly, most of the effort of this project was focused on implementing a basic 5G Core Testbed, and while it allowed for a great deal of experimentation and evaluation, it is clear that a more advanced setup may be required for further research and testing, or at the very least more resources to allow a more advanced core network with multiple network slices with more network functions.

6.2.2. DFR in the Radio Access Network (RAN)

This research focused primarily on the core network, due to resource limitations. Therefore there is an oppurtunity for research to be conducted on the potential for DFR in the Radio Access Network (RAN).

6.2.3. Cost and Regulatory Compliance

While efforts were made to reduce the resource costs and allow for regulatory compliance in the design 5GDFRT discussed in this report, an in-depth analysis into the efficiency, performance, and legal requirements that a DFR solution for 5G Networks should adhere to requires further research, testing, and expertise.

6.3. Conclusion

To summarise, this research project conducted a literature review to determine the current state of digital forensics and digital forensic readiness in 5G, this review found that there is a lack of a proactive digital forensic readiness approach to 5G Networks. Following this, a prototype 5G Digital Forensic Readiness Tool (5GDFRT) was designed and implemented to securely collect and store useful data so that it can be analyzed and utilized as part of a potential forensic investigation. A 5G Testbed was then designed and implemented as part of the implementation and evaluation of the 5GDFRT. I consider both the basic 5G Core Testbed and the 5GDFRT as contributions of this research. The 5GDFRT was then tested and evaluated using the Testbed with attack scenarios to determine if it was able to capture and detect these attacks. We then evaluated the results as part of a critical evaluation and found that the 5GDFRT could in fact detect these attacks and that the collected data could effectively and efficiently be used for event reconstruction.

Overall, this research project showed that a DFR solution could at the very least satisfy monitoring and information assurance requirements, and explained how it could potentially meet the cost and regulatory compliance requirements.

References

Årnes, A., 2017. Digital forensics. John Wiley & Sons.

Behnke, D., Müller, M., Bök, P.B., Schneider, S., Peuster, M., Karl, H., Rocha, A., Mesquita, M., Bonnet, J., 2019. Nfv-driven intrusion detection for smart manufacturing , 1–6.

Białecki, A., Muir, R., Ingersoll, G., Imagination, L., 2012. Apache lucene 4, in: SIGIR 2012 workshop on open source information retrieval, p. 17.

Boldyreva, A., Fischlin, M., Palacio, A., Warinschi, B., 2007. A closer look at pki: Security and efficiency, in: Public Key Cryptography–PKC 2007: 10th International Conference on Practice and Theory in Public-Key Cryptography Beijing, China, April 16-20, 2007. Proceedings 10, Springer. pp. 458–475.

Cardoso, K.V., Both, C.B., Prade, L.R., Macedo, C.J., Lopes, V.H.L., 2020. A softwarized perspective of the 5g networks. arXiv preprint arXiv:2006.10409 arXiv:2006.10409.

Condoluci, M., Mahmoodi, T., 2018. Softwarization and virtualization in 5g mobile networks: Benefits, trends and challenges. Computer Networks 146, 65–84.

Dangi, R., Lalwani, P., Choudhary, G., You, I., Pau, G., 2021. Study and investigation on 5g technology: A systematic review. Sensors 22, 26.

Elyas, M., Ahmad, A., Maynard, S.B., Lonie, A., 2015. Digital forensic readiness: Expert perspectives on a theoretical framework. Computers & Security 52, 70–89.

ETSI, 2023. Nfv descriptors based on tosca specification. Available at: https://www.etsi.org/deliver/etsi_gs/NFV-SOL/001_099/001/02.06.01_60/gs_NFV-SOL001v020601p.pdf.

Farahmandian, S., Hoang, D.B., 2016. Security for software-defined (cloud, sdn and nfv) infrastructures–issues and challenges, in: Eight international conference on network and communications security.

Hajlaoui, E., Zaier, A., Khlifi, A., Ghodhbane, J., Ben Hamed, M., Sbita, L., 2020. 4g and 5g technologies: A comparative study, in: 2020 5th International Conference on Advanced Technologies for Signal and Image Processing (ATSIP), IEEE. pp. 1–6.

International Organization for Standardization, 2015. Iso/iec 27043:2015 - information technology – security techniques – incident investigation principles and processes. https://www.iso.org/standard/44407.html. Accessed: 2024-10-06.

Jeyaraman, S., Atallah, M.J., 2006. An empirical study of automatic event reconstruction systems. digital investigation 3, 108–115.

Jover, R.P., 2019. The current state of affairs in 5g security and the main remaining security challenges. arXiv preprint arXiv:1904.08394.

Kebande, V.R., Venter, H.S., 2018. On digital forensic readiness in the cloud using a distributed agent-based solution: issues and challenges. Australian Journal of Forensic Sciences 50, 209–238.

Kumar, S., 2006. Ping attack–how bad is it? Computers Security 25, 332–337.

Makura, S., Venter, H., 2024. Towards the development of a digital forensic readiness model for 5g nfv environments, in: 2024 IST-Africa Conference (IST-Africa), IEEE. pp. 1–12.

Makura, S.M., Venter, H.S., Ikuesan, R.A., Kebande, V.R., Karie, N.M., 2020. Proactive forensics: Keystroke logging from the cloud as potential digital evidence for forensic readiness purposes, in: 2020 IEEE International Conference on Informatics, IoT, and Enabling Technologies (ICIoT), pp. 200–205. doi:10.1109/ICIoT48696.2020.9089494.

Regulation, P., 2016. Regulation (eu) 2016/679 of the european parliament and of the council. Regulation (eu) 679, 2016.

Scarfone, K., Mell, P., et al., 2007. Guide to intrusion detection and prevention systems (idps). NIST special publication 800, 94.

Shaik, A., Borgaonkar, R., Park, S., Seifert, J.P., 2019. New vulnerabilities in 4g and 5g cellular access network protocols: exposing device capabilities, in: Proceedings of the 12th Conference on Security and Privacy in Wireless and Mobile Networks, Association for Computing Machinery. pp. 221–231. URL: https://doi.org/10.1145/3317549.3319728, doi:10.1145/3317549.3319728.

Sharevski, F., 2018. Towards 5g cellular network forensics. EURASIP Journal on Information Security 2018, 1–16.

Smith, J., Doe, J., 2023. An example of conference paper citation in bibtex, in: Proceedings of the International Conference on Sample Topics, IEEE. IEEE Press, New York, NY. pp. 123–130. doi:10.1109/ICST.2023.00001.

of South Africa, R., 2019. Protection of personal information act 2019. Government Printer. Available at: https://popia.co.za/section-14-retention-and-restriction-of-records/.

Tan, J., 2001. Forensic readiness. Cambridge, MA: Stake 1.

Thorogood, R., Brookson, C., 2007. Lawful interception. TELEKTRONIKK 103, 33.

Valjarević, A., Venter, H., Petrović, R., 2016. Iso/iec 27043: 2015—role and application, in: 2016 24th Telecommunications Forum (TELFOR), IEEE. pp. 1–4.
Wilkens, F., Haas, S., Amann, J., Fischer, M., 2022. Passive, transparent, and selective tls decryption for network security monitoring, in: IFIP International Conference on ICT Systems Security and Privacy Protection, Springer. pp. 87–105.
Zhang, S., 2019. An overview of network slicing for 5g. IEEE Wireless Communications 26, 111–117.