5G SECURITY TEST BED™

Securing 5G:

CSRIC VII 5G Standalone Network Test Report

Q4 2023

Table of Contents

Introduction 3
Scope of Report 5
Background 5
Why CSRIC VII 5
CSRIC VII Working Group 3's Report and Recommendations for 5G Standalone Architecture ... 6
Definition of CSRIC Test Cases 7
Test Results 9
Introduction 9
IPsec Configuration 11
SIM Card Profiles 12
Test Case 1: CSRIC 7 WG 3 – NAS Signaling Confidentiality 12
Part 2: Using NEA2 Encryption 21
Test Case 2: CSRIC 7 WG 3 – RRC Signaling Confidentiality 24
Test Case 3: CSRIC 7 WG 3 – Access Stratum User Plane Confidentiality 31
Test Case 4: CSRIC 7 WG 3 – Access Stratum User Plane Integrity 33
Test Case 5: CSRIC 7 WG 3 – SUPI/SUCI Privacy Enabled 36
Test Case 6: CSRIC 7 WG 3 – IPsec on Transport Links 39
Test Case 7: CSRIC 7 WG 3 – Transport Layer Security for SBA Interfaces 47
Conclusions and Next Steps 60
Appendix: Acronyms 63

Introduction

The 5G Security Test Bed Is the Latest Industry Initiative to Advance 5G Security

The wireless industry prioritizes stronger security and reliability with every generation of its mobile networks. With 5G in particular, secure connectivity is the foundation that supports and enhances the many benefits these networks provide. The wireless industry devotes significant resources to 5G security and has expanded its efforts through the 5G Security Test Bed.

Formally launched in 2022, the 5G Security Test Bed is a unique collaborative endeavor between wireless providers, equipment manufacturers, cybersecurity experts, academia, and government agencies, created with a sole focus on testing and validating 5G security recommendations and use cases from government groups, wireless operators, and others. It is the only initiative that uses commercial-grade network equipment and facilities to demonstrate and validate how 5G security standards recommendations will work in practical, real-world conditions.

The 5G Security Test Bed reflects the industry's collaborative approach to 5G security—it was created by the Cybersecurity Working Group (CSWG), an industry initiative that convenes the world's leading telecom and tech companies to assess and address the present and future of cybersecurity. Its members are wireless providers AT&T, T-Mobile, and UScellular; industry partners Ericsson, the MITRE Group, SecureG, and Intel; and academic partners the University of Maryland and Virgina Tech Advanced Research Corporation (VT-ARC).

The 5G Security Test Bed has a Technical Advisory Committee (TAC) made up of its members and the Test Bed Administrator. The TAC advises the Test Bed Administrator on the day-to-day technical and operational activities and decisions related to the Test Bed, including but not limited to: development of use cases to be tested, test plan development and review, raw test data analysis, test result and report generation, and development of recommendations to standards bodies based on results.

The 5G Security Test Bed further works with a broad array of government agencies, policymakers, international standards bodies, thought leaders, and partners in the telecommunications and information technology sectors. These groups include the 3rd Generation Partnership Project (3GPP), the International Telecommunication Union (ITU), the Department of Homeland Security (DHS), the National Institute of Standards and Technology (NIST), and the Federal Communications Commission (FCC), among others.

The 5G Security Test Bed Uses Real-World Equipment, Validating Real-World Applications

One of the 5G Security Test Bed's core values lies in its ability to validate 5G security use cases in a real-world environment, using an actual 5G network architecture. Leveraging a significant investment and in-kind contributions, the Test Bed's founding members built this state-of-the-art, private 5G network from scratch for the singular purpose of evaluating 5G network security.

The 5G Security Test Bed's previous testing activities have worked to validate the recommendations of the FCC's Communications Security, Reliability, and Interoperability Council (CSRIC) advisory group, for both non-standalone and standalone network configurations. In addition, the Test Bed draws on recommendations from its own Technical Advisory Committee to address emerging vulnerability research. The first report in this series focused on the validation of the CSRIC non-standalone configurations, while this report addresses the CSRIC standalone configuration recommendations and network slicing. The 5G Security Test Bed will continue evaluating additional recommendations and use cases from CSRIC and other entities in future tests. It is not set up to be a platform for identifying vulnerabilities or conducting penetration testing of networks or equipment.

Real-World Testing

The 5G Security Test Bed advances wireless security by:

- Conducting real-world tests in a rigorous, transparent, and replicable manner that can assess and validate theoretical and policy concerns and overcome hypothetical laboratory testing limitations.
- Drawing on the expertise of government, wireless providers, and equipment manufactures to evaluate specific use cases and support new equipment development.
- Testing security functionality in different scenarios, enabling industry and government to identify, mitigate, and respond to evolving threats while protecting consumers, businesses, and government agencies.

Real-World Applications

The 5G Security Test Bed's tests and outcomes support several applications that can drive new technology and transform cities, government, and industries. Use cases include government and enterprise applications, general network security protections, and smart city applications such as:

- Primary Use Cases: Network Security
    - Protecting Information in Transit
    - Roaming Security
    - Subscriber Privacy
    - Zero Trust Network Security
    - False Base Station Detection and Protection
    - 5G Cloud Network Security

- Secondary Use Cases: Devices and Applications
    - High-Resolution Video Surveillance (e.g. Smart Cities, Large Venues)
    - LTE/5G Drones with High-Resolution Video Feedback (e.g. Smart Cities)
    - Dynamic Supply Chain Verification (Real-Time Monitoring and Logistics)
    - Automated, Reconfigurable Factories
    - Autonomous Vehicles
    - Immersive AR/VR

The 5G standalone network architecture tested for this report makes up key components of these applications because they enable service to be customized to diverse needs and requirements. The test cases outlined here show how these new and evolving uses can successfully adopt enhanced security capabilities while improving performance and capability.

Scope of Report

This report addresses recommendations derived from the FCC's Communications Security, Reliability, and Interoperability Council VII March 2021 report, *Report on Recommendations for Identifying Optional Security Features That Can Diminish the Effectiveness of 5G Security.*[1] The report focused on the implementation of security protections in 5G "standalone" (SA) networks (that is, networks designed and built specifically for 5G) by assessing security features from 3GPP TS 33.501, the primary technical standard for 5G SA. (By contrast, non-standalone networks offer 5G service together with 4G LTE over shared infrastructure.) The first report from the 5G Security Test Bed focused on NSA networks supporting both 5G and 4G traffic.

This 5G STB report's scope is to evaluate and verify CSRIC VII's recommendations for SA architecture by investigating the security features associated with 5G network infrastructure and the devices that can access a 5G SA network.

Background

Why CSRIC VII

The Communications Security, Reliability, and Interoperability Council is a federal advisory committee that provides the Federal Communications Commission with recommendations to enhance the security, reliability, and interoperability of communications systems. CSRIC provides a forum for industry and government technical experts to assess developing technology and analyze complex issues. It is a leading venue for stakeholders in and outside of government to share ideas and best practices, and to help the FCC stay abreast of cutting-edge technology

[1] CSRIC VII WG3, Report on Recommendations for Identifying Optional Security Features That Can Diminish the Effectiveness of 5G Security (Mar. 2021), https://www.fcc.gov/file/20606/download.

and security issues affecting the communications sector. CSRIC's work continues to influence government and industry agendas and activities.

The FCC charters CSRIC every two years. CSRIC VII's charter was from March 2019 to March 2021, and it focused on a range of public safety and homeland security-related communications matters, including issues related to 5G network evolution. 5G offers significant and novel capabilities compared with previous generations of wireless networks, but new capabilities, infrastructure, and equipment can also introduce security risks. The FCC tasked CSRIC VII with examining these security risks and making recommendations associated with the evolving standards' optional security features. Because 5G standards and specifications continue to develop, CSRIC VII's work offered an opportunity to update future standards.

Likewise, the 5G Security Test Bed's work in testing CSRIC's recommendations can be used both to inform network architecture and operation, and to enhance future 5G standards.

CSRIC VII Working Group 3's Report and Recommendations for 5G Standalone Architecture

CSRIC VII's Recommendations

CSRIC VII worked to identify and evaluate optional features in the 3GPP standards that would potentially cause security gaps in 5G if not implemented. In March 2021, CSRIC's Working Group 3 (WG3, "Managing Security Risk in Emerging 5G Implementations") released a report, *Report on Recommendations for Identifying Optional Security Features That Can Diminish the Effectiveness of 5G Security.*[2] The report focused on identifying optional features in proposed 3GPP standards that might diminish the effectiveness of 5G security, and made recommendations to address these gaps.

Several security features outlined in 3GPP TS 33.501 releases 15 and 16 were mandatory for equipment vendors to implement, but optional for 5G network operators to deploy. CSRIC VII WG3 looked at the optional security features and conducted a risk assessment and analysis on those measures, including: confidentiality for Non-Access Stratum (NAS) signaling,[3] user plane confidentiality and integrity, radio resource control signaling confidentiality, Subscription Permanent Identifier (SUPI)/International Mobile Subscriber Identity (IMSI) privacy, and network security, including IP security (IPsec) and transport layer security (TLS).

[2] *Id.*
[3] "NAS signaling" carries the user data from the user equipment to the MME through the S1 pathway.

Based on its assessment, CSRIC VII WG3 made eight recommendations:

- **Previous CSRIC Recommendations:** Communications sector members and stakeholders should adopt CSRIC-recommended 5G SA threat mitigations from previous CSRIC VI, V, and IV reports.[4]
- **NAS Signaling Confidentiality:** Operators should convey only non-user identity related information until security context is established. (CSRIC noted that 3GPP TS 33.501 encrypts all NAS messages after security context is established.)
- **User Plane Confidentiality:** Operators should apply user plane (UP) confidentiality protections at the Packet Data Convergence Protocol (PDCP) layer.
- **User Plane Integrity:** OEM and network infrastructure vendors should support, and operators should implement, the 3GPP TS 33.501 Release 16 and 128-NIA3 capabilities of supporting integrity protection and user data replay protection at the full data rate available to the user equipment. (Release 15 required only 64kbps.)
- **RRC Signaling Confidentiality:** Operators should protect RRC-signaling (Radio Resource Control) confidentiality and convey only non-identity related information prior to establishing security context.
- **SUPI/IMSI Privacy:** Devices and networks in the U.S. should use IMSI privacy, and permit the null encryption only for making emergency services calls (i.e. 9-1-1).
- **Network Security—IPsec:** Operators should apply IPsec or a tunneling technology such as VPN tunnels for transport.
- **Core Network Security—Transport Layer Security (TLS):** Operators should apply TLS for Service-Based Architecture (SBA) interfaces.

Definition of CSRIC Test Cases

Based on the CSRIC VII WG3 recommendations, the 5G STB established and executed seven test cases described in this report, as follows:

1. **NAS Signaling Confidentiality:**
    a. <u>CSRIC VII WG3 Recommendation:</u> Operators should convey only non-user identity related information until security context is established.

[4] *See* CSRIC VI WG3, Report on Best Practices and Recommendations to Mitigate Security Risks to Emerging 5G Wireless Networks (Sept. 2018), https://www.fcc.gov/file/14500/download; CSRIC V WG6, Best Practices Recommendations for Hardware and Software Critical to the Security of the Core Communications Network (making recommendations for security-by-design principles in the core communications network) (March 2016), https://transition.fcc.gov/bureaus/pshs/advisory/csric5/WG6_FINAL_%20wAppendix_0316.pdf; and CSRIC IV WG4, Wireless Segment Cybersecurity Risk Management and Best Practices (March 2015), https://transition.fcc.gov/pshs/advisory/csric4/CSRIC_IV_WG4_Final_Report_031815.pdf.

   b. 5G STB Test Case 1: Demonstrate how user identity related information can be transmitted confidentially by testing the implementation of NAS Signaling encryption. Once an encrypted channel is established, only non-user identity related information should be observable.

2. **RRC Signaling Confidentiality:**

   a. CSRIC VII WG3 Recommendation: Operators should protect RRC-signaling confidentiality and convey only non-identity related information prior to establishing security context.

   b. 5G STB Test Case 2: Demonstrate that the PDCP provides RRC signaling confidentiality between the user equipment and NG-RAN (Next Generation Radio Access Network) using 128-bit NEA algorithms.

3. **Access Stratum User Plane (Payload Data) Confidentiality:**

   a. CSRIC VII WG3 Recommendation: Operators should apply user plane confidentiality protections at the PDCP layer.

   b. 5G STB Test Case 3: To demonstrate that the PDCP provides user plane data confidentiality between the user equipment and NG-RAN using 128-bit NEA algorithms.

4. **Access Stratum User Plane (Payload Data) Integrity:**

   a. CSRIC VII WG3 Recommendation: Operators should apply user plane confidentiality protections at the PDCP layer.

   b. 5G STB Test Case 4-1: Demonstrate that the PDCP provides user plane data integrity protection at the full rate.

5. **SUPI/IMSI User Privacy:**

   a. CSRIC VII WG3 Recommendation: Devices and networks in the U.S. should use IMSI privacy.

   b. 5G STB Test Case 5-1: Register a device on the test network by exchanging identity information using the subscription concealed identifier (SUCI) to encrypt the SUPI.

6. **Network Security:**

   a. CSRIC VII WG3 Recommendation: Apply IPsec or tunneling technology to protect network security during transport.

   b. 5G STB Test Case 6: Use IPsec to transmit user plane and control plane (CP) messaging while protecting confidentiality, integrity, and replay.

7. **Core Network Security (Transport Link Encryption):**
    a. CSRIC VII WG3 Recommendation: Use TLS for SBA interfaces and tunneling technology for transport when not using the SBA.
    b. 5G STB Test Case 7: Demonstrate TLS encryption to protect SBA interfaces in the 5G core.

Test Results

Introduction

This document presents test results based on use cases derived from the FCC's Communications Security, Reliability, and Interoperability Council (CSRIC) VII Working Group 3 (WG3) Report 2 recommendations for securing 5G standalone networks based upon its analysis of optional security requirements in 3GPP TS 33.501.

The configuration used for these tests comprises radio access network (RAN) equipment hosted at the University of Maryland (UMD) and a dual-mode core (DMC), that provides both 4G LTE and 5G functionality hosted at the MITRE Corporation. The core is the Ericsson DMC, PCC version 1.19. The connection between the RAN at UMD and the DMC at MITRE goes over the internet and, for the scenarios considered here, is treated as an untrusted link.[5] Error! Reference source not found. shows the relevant components of the Test Bed, including available test points. Not all of the test points shown in the diagram were used for these tests.

The routers shown at each location are Ericsson 6672 routers (referred to as R6672, or R6K). The switches shown are each Pluribus Freedom 9372-X switches. For the tests implemented here, the two switches are considered part of the "untrusted" backhaul link. The core is configured to support two network slices. The first slice, referred to as Slice 1 in this report, is considered the default enhanced mobile broadband (eMBB) network slice. The second slice, Slice 2, emulates a private network and includes the ability to form an IPsec tunnel to create a highly secure slice. The IPsec tunnel is configured with one endpoint at the baseband unit (BBU) and the other at the core-side R6672 router. On the server on the core side, there are two virtual web servers instantiated, one for each slice, and isolated from each other. All tests for the test cases discussed in this report were executed on Slice 2.

[5] In the actual implementation, there are additional security measures implemented, including an IPsec tunnel between the UMD and MITRE campus/corporate networks. For the purposes of these tests, this tunnel is considered part of the untrusted link and therefore, any encryption implemented for the tests is in addition to these measures.

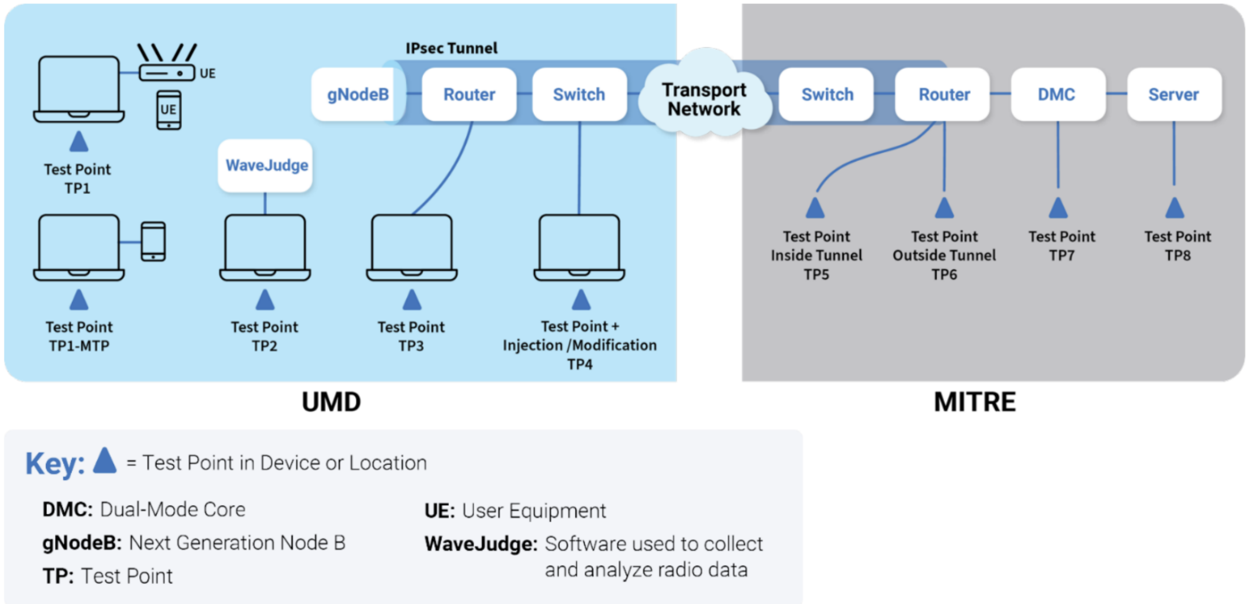


Figure 1: 5G STB Lab Component Block Diagram and Test Points

Tests were run with band N41 for the new radio (NR) using a Sierra Wireless EM9190 card connected to a laptop by USB as a cellular modem, as well as a Qualcomm Mobile Test Platform (MTP) device. For the purposes here, we will refer to the combination of that laptop and the cellular modem as the UE.

Packets are captured on each of the identified test points in Figure 1: at the UE(s) (TP1), on the RAN-side Pluribus switch (TP4), on the Core-side R6K router (TP6), and at the DMC (TP7). These test points are identified with numbers as shown in the figure and described in more detail in Table 1.

Table 1: Test Point Descriptions

Test Point	Description and Use
TP1-S	Laptop connected to Sierra Wireless card and/or software-defined radio (SDR); Wireshark captures packets originating at and destined to UE laptop; other tools access SDR controls and data
TP1-MTP	Laptop connected to Qualcomm MTP; QXDM allows access to low-level data
TP2	WaveJudge interface
TP3	Wireshark running on laptop connected to RAN-side R6K router; can capture packets inside the tunnel (encrypted packets when IPsec tunnel is enabled)
TP4	tcpdump running on laptop connected to port of RAN-side Pluribus switch used to capture, modify, and inject packets on the “untrusted link”
TP5	tcpdump running on port of core-side R6K router inside the IPsec tunnel (encrypted packets when IPsec tunnel is enabled) used to monitor packets on the “untrusted link”
TP6	tcpdump running on port of core-side R6K router outside the IPsec tunnel (i.e., before IPsec encryption or after IPsec decryption) used to monitor packets at the interface to the DMC
TP7	CNOM tool accessing DMC messages and command line interface on core
TP8	Applications running on application server in MITRE facility

IPsec Configuration

3GPP TS 33.401 requires IPsec, when used, to support ESP and IKEv2 with certificates-based authentication. The security gateway (SEG) is optional to use. The following requirements are from 33.401, section 12, Backhaul link user plane protection:

In order to protect the S1 and X2 user plane as required by clause 5.3.4, it is required to implement **IPsec ESP** according to RFC 4303 [7] as profiled by TS 33.210 [5], with confidentiality, integrity and replay protection.

Tunnel mode IPsec is mandatory to implement on the eNB for X2-U and S1-U.
On the X2-U and S1-U, transport mode IPsec is optional for implementation. NOTE 1: Transport mode can be used for reducing the protocol overhead added by IPsec.
On the core network side a **SEG may be used** to terminate the IPsec tunnel.

For both S1 and X2 user plane, **IKEv2 with certificates based authentication shall be implemented**. The certificates shall be implemented according to the profile described by TS 33.310 [6]. IKEv2 shall be implemented conforming to the IKEv2 profile described in TS 33.310 [6].

3GPP TS 33.501 retains these IPsec requirements for 5G SA and NSA, when IPsec is used.

CSRIC 7 WG 2 Report 2 recommends IPsec on untrusted links to provide confidentiality and integrity protection over the S1-MME, S1-U, and management interfaces.

IPsec is implemented on Slice 2, with tunnel endpoints at the RAN and at the core-side R6K.

SIM Card Profiles

Some tests require different SIM card profiles to tests the desired functionality. Table 2 lists the different profiles that were used during each test.

Table 2: SIM Card Profiles

ID	IMSI	Profile
N	310 014 791 791 001	N (NULL)
A	310 014 791 791 011	A
B	310 014 791 791 021	B

Test Case 1: CSRIC 7 WG 3 – NAS Signaling Confidentiality

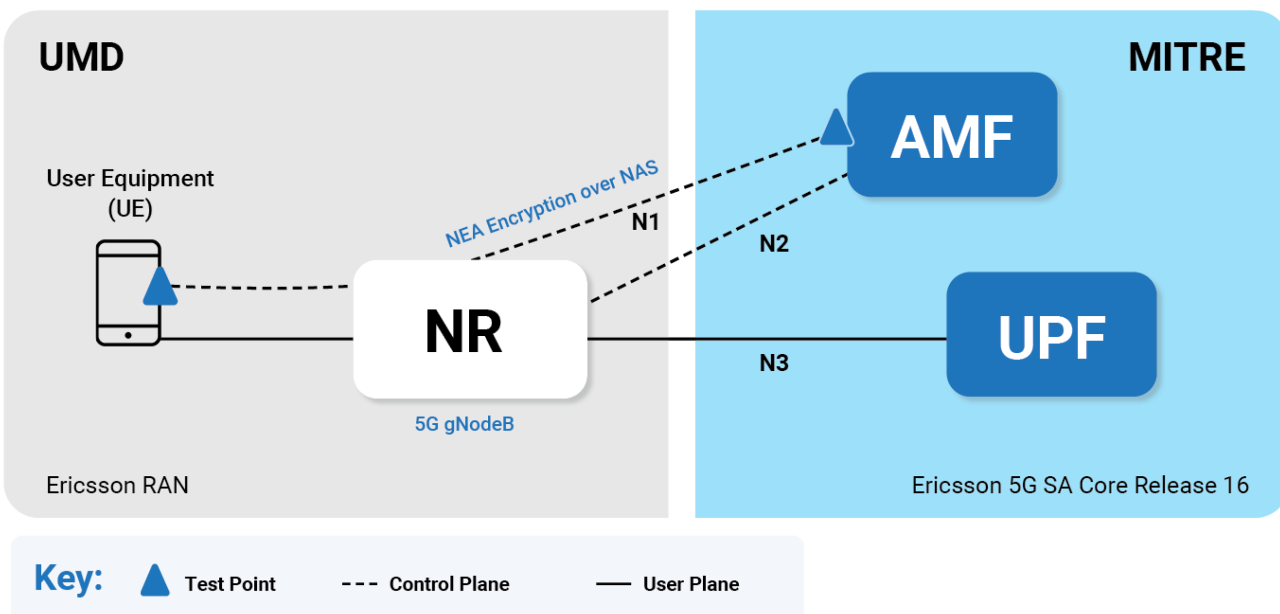


Figure 2: Test Case SA-01 Configuration

Test Case ID: TC-SA-01
Description:
3GPP TS 33.501 specifies mandatory (e.g., requires vendor implementation) support for protection of the NAS signaling confidentiality, but optional for service providers to use.

Given this standards requirement, CSRIC VII recommends only non-user identity related information shall be conveyed prior to security context being established. Note, after security context is established, all NAS messages are encrypted according to 3GPP TS 33.501.

This test involves implementation of NAS signaling encryption on the N1 interface. Once encrypted channels are established, user identity info may be securely exchanged.

Test points used:

Used	Test Point	Description and Use
X	TP1-S	Wireshark running on laptop connected to Sierra Wireless card; captures packets originating at and destined to UE laptop
	TP1-MTP	Laptop connected to Qualcomm MTP; QXDM allows access to low-level data
	TP2	WaveJudge interface
	TP3	Wireshark running on laptop connected to RAN-side R6K router; can capture packets inside the tunnel (encrypted packets when IPsec tunnel is enabled)
	TP4	tcpdump running on laptop connected to port of RAN-side Pluribus switch used to capture, modify, and inject packets on the “untrusted link”
	TP5	tcpdump running on port of core-side R6K router inside the IPsec tunnel (encrypted packets when IPsec tunnel is enabled) used to monitor packets on the “untrusted link”
X	TP6	tcpdump running on port of core-side R6K router outside the IPsec tunnel (i.e., before IPsec encryption or after IPsec decryption) used to monitor packets at the interface to the DMC
	TP7	CNOM tool accessing DMC messages
	TP8	Applications running on application server in MITRE facility

Slice	IP pool	SIM LABEL	SIM LABEL IMSI	DNN	DN SERVERS
Slice 2	172.24.1.0/24	N21	3100147917910021	dnn-embb-stb2.mitre.net	192.168.59.146/28

Network Slice 2 and a UE with Profile B SIM were used throughout the tests. The UE used for Slice 2 was a Sierra Wireless Modem, which is connected and controlled by a laptop outside the Faraday Cage. IPsec for for Slice 2 and control traffic was turned on/off as and when required.

To ensure that the core did not retain the UE state, we deleted the UE context from the core (Figure 3), and ensured that the IPsec tunnel for the transport channel between the RAN and core was up (Figure 4).

```
=== mtrdmcamf01 erv@eric-pc-mm-controller-0 ANCB ~ # gsh delete_subscriber -imsi 310014791791021
=== mtrdmcamf01 erv@eric-pc-mm-controller-0 ANCB ~ # gsh delete_subscriber -imsi 310014791791001
Subscriber identity: "310014791791001" is not registered.
=== mtrdmcamf01 erv@eric-pc-mm-controller-0 ANCB ~ #
=== mtrdmcamf01 erv@eric-pc-mm-controller-0 ANCB ~ #
```

Figure 3: Deleting UE Context from Core

```
MUMD02AVW> st ipsec

221118-16:31:13 169.254.2.2 22.0h MSRBS_NODE_MODEL_22.Q2_566.28125.116_3317 stopfile=/tmp/2593350
===================================================================================
Proxy  Adm State     Op. State     MO
===================================================================================
14282                1 (ENABLED)   Transport=1,Router=NRCUCP,IpsecTunnel=1
===================================================================================
Total: 1 MOs
```

Figure 4: IPsec enabled between BBU and Core-side R6K router

This test has two parts; 1) with NEA0 (no encryption) and 2) with NEA2 activated to encrypt NAS signaling. For Part 1, on the core side, we used a command line interface command to set priority for the NAS encryption algorithm making NEA0 (null algorithm) the highest priority (specifically, priority 1). See Figure 5 for the initial NEA settings, Figure 6 for the commands changing the priority settings, and

```
=== mtrdmcamf01 erv@eric-pc-mm-controller-0 ANCB ~ # gsh get_nea_algorithm -name nea0
Parameter                           Active Data     Planned Data
-------------------------------------------------------------------
timestamp                           20221118164429  _
planState                           _               _
prio (N1SecurityAlgorithmPriority)  1               _
=== mtrdmcamf01 erv@eric-pc-mm-controller-0 ANCB ~ # gsh get_nea_algorithm -name nea1
Parameter                           Active Data     Planned Data
-------------------------------------------------------------------
timestamp                           20221109120424  _
planState                           _               _
prio (N1SecurityAlgorithmPriority)  2               _
=== mtrdmcamf01 erv@eric-pc-mm-controller-0 ANCB ~ # gsh get_nea_algorithm -name nea2
Parameter                           Active Data     Planned Data
-------------------------------------------------------------------
timestamp                           20221118164429  _
planState                           _               _
prio (N1SecurityAlgorithmPriority)  3               _
=== mtrdmcamf01 erv@eric-pc-mm-controller-0 ANCB ~ # gsh get_nea_algorithm -name nea3
Parameter                           Active Data     Planned Data
-------------------------------------------------------------------
timestamp                           20220727201309  _
planState                           _               _
prio (N1SecurityAlgorithmPriority)  0
```

Figure 7 displaying the encryption setting to NULL (NEA0). For the second part, we set encryption back to NEA2 with the highest priority.

```
=== mtrdmcamf01 erv@eric-pc-mm-controller-0 ANCB ~ # gsh get_nea_algorithm -name nea0
Parameter                           Active Data     Planned Data
------------------------------------------------------------------
timestamp                           20221118100928  _
planState                           _               _
prio (N1SecurityAlgorithmPriority)  1               _
=== mtrdmcamf01 erv@eric-pc-mm-controller-0 ANCB ~ # gsh get_nea_algorithm -name nea1
Parameter                           Active Data     Planned Data
------------------------------------------------------------------
timestamp                           20221109120424  _
planState                           _               _
prio (N1SecurityAlgorithmPriority)  2               _
=== mtrdmcamf01 erv@eric-pc-mm-controller-0 ANCB ~ # gsh get_nea_algorithm -name nea2
Parameter                           Active Data     Planned Data
------------------------------------------------------------------
timestamp                           20221118100928  _
planState                           _               _
prio (N1SecurityAlgorithmPriority)  3               _
=== mtrdmcamf01 erv@eric-pc-mm-controller-0 ANCB ~ # gsh get_nea_algorithm -name nea3
Parameter                           Active Data     Planned Data
------------------------------------------------------------------
timestamp                           20220727201309  _
planState                           _               _
prio (N1SecurityAlgorithmPriority)  0               _
```

Figure 5: Confidentially Core Setting - NEA2 (128-NEA2 cipher algorithm) before configuration change

```
== mtrdmcamf01 erv@eric-pc-mm-controller-0 ANCB ~ # gsh modify_nea_algorithm -name nea2 -prio 1
== mtrdmcamf01 erv@eric-pc-mm-controller-0 ANCB ~ # gsh modify_nea_algorithm -name nea0 -prio 3
== mtrdmcamf01 erv@eric-pc-mm-controller-0 ANCB ~ #
== mtrdmcamf01 erv@eric-pc-mm-controller-0 ANCB ~ #
== mtrdmcamf01 erv@eric-pc-mm-controller-0 ANCB ~ #
== mtrdmcamf01 erv@eric-pc-mm-controller-0 ANCB ~ # gsh get_nea_algorithm -name nea0
'arameter                           Active Data     Planned Data
------------------------------------------------------------------
imestamp                            20221118100928  20221118161841
lanState                            _               Modified
rio (N1SecurityAlgorithmPriority)   1               3
== mtrdmcamf01 erv@eric-pc-mm-controller-0 ANCB ~ # gsh get_nea_algorithm -name nea1
'arameter                           Active Data     Planned Data
------------------------------------------------------------------
imestamp                            20221109120424  _
lanState                            _               _
rio (N1SecurityAlgorithmPriority)   2               _
== mtrdmcamf01 erv@eric-pc-mm-controller-0 ANCB ~ # gsh get_nea_algorithm -name nea2
'arameter                           Active Data     Planned Data
------------------------------------------------------------------
imestamp                            20221118100928  20221118161826
```

Figure 6: Confidentially cipher algorithm setting changes

```
=== mtrdmcamf01 erv@eric-pc-mm-controller-0 ANCB ~ # gsh get_nea_algorithm -name nea0
Parameter                          Active Data     Planned Data
-----------------------------------------------------------------
timestamp                          20221118164429  _
planState                          _               _
prio (N1SecurityAlgorithmPriority)  1              _
=== mtrdmcamf01 erv@eric-pc-mm-controller-0 ANCB ~ # gsh get_nea_algorithm -name nea1
Parameter                          Active Data     Planned Data
-----------------------------------------------------------------
timestamp                          20221109120424  _
planState                          _               _
prio (N1SecurityAlgorithmPriority)  2              _
=== mtrdmcamf01 erv@eric-pc-mm-controller-0 ANCB ~ # gsh get_nea_algorithm -name nea2
Parameter                          Active Data     Planned Data
-----------------------------------------------------------------
timestamp                          20221118164429  _
planState                          _               _
prio (N1SecurityAlgorithmPriority)  3              _
=== mtrdmcamf01 erv@eric-pc-mm-controller-0 ANCB ~ # gsh get_nea_algorithm -name nea3
Parameter                          Active Data     Planned Data
-----------------------------------------------------------------
timestamp                          20220727201309  _
planState                          _               _
prio (N1SecurityAlgorithmPriority)  0
```

Figure 7: Confidentially Core Setting changes – NEA0 (no encryption) now has the highest priority (priority 1)

```
=== mtrdmcamf01 erv@eric-pc-mm-controller-0 ANCB ~ # gsh get_nea_algorithm -name nea0
Parameter                          Active Data     Planned Data
-----------------------------------------------------------------
timestamp                          20221118171329  _
planState                          _               _
prio (N1SecurityAlgorithmPriority)  3              _
=== mtrdmcamf01 erv@eric-pc-mm-controller-0 ANCB ~ # gsh get_nea_algorithm -name nea1
Parameter                          Active Data     Planned Data
-----------------------------------------------------------------
timestamp                          20221109120424  _
planState                          _               _
prio (N1SecurityAlgorithmPriority)  2              _
=== mtrdmcamf01 erv@eric-pc-mm-controller-0 ANCB ~ # gsh get_nea_algorithm -name nea2
Parameter                          Active Data     Planned Data
-----------------------------------------------------------------
timestamp                          20221118171329  _
planState                          _               _
prio (N1SecurityAlgorithmPriority)  1              _
=== mtrdmcamf01 erv@eric-pc-mm-controller-0 ANCB ~ # gsh get_nea_algorithm -name nea3
Parameter                          Active Data     Planned Data
-----------------------------------------------------------------
timestamp                          20220727201309  _
planState                          _               _
prio (N1SecurityAlgorithmPriority)  0              _
=== mtrdmcamf01 erv@eric-pc-mm-controller-0 ANCB ~ #
```

Figure 8: Confidentially Core Setting changes – NEA2 (no encryption) now has the highest priority (priority 1)

Part 1: Using NEA0 Encryption

Upon starting the UE, it sends an initial message with a registration request. Figure 9 shows the Wireshark interpretation of the captured message from the UE indicating that for the 5G mobile identity, the type of identity is SUCI, the concealed identifier. Subsequently, the AMF requests authentication (Figure 10) and the UE responds (Figure 11). After authentication, the core indicates the ciphering algorithm to be used, in this case NEA0, as shown in Figure 12. Subsequent transmissions are processed with the NEA0 (NULL) algorithm, resulting in decipherable messages, as shown in Figure 13 and Figure 14, in which details of the messages are visible such as the UE's IMEISV.

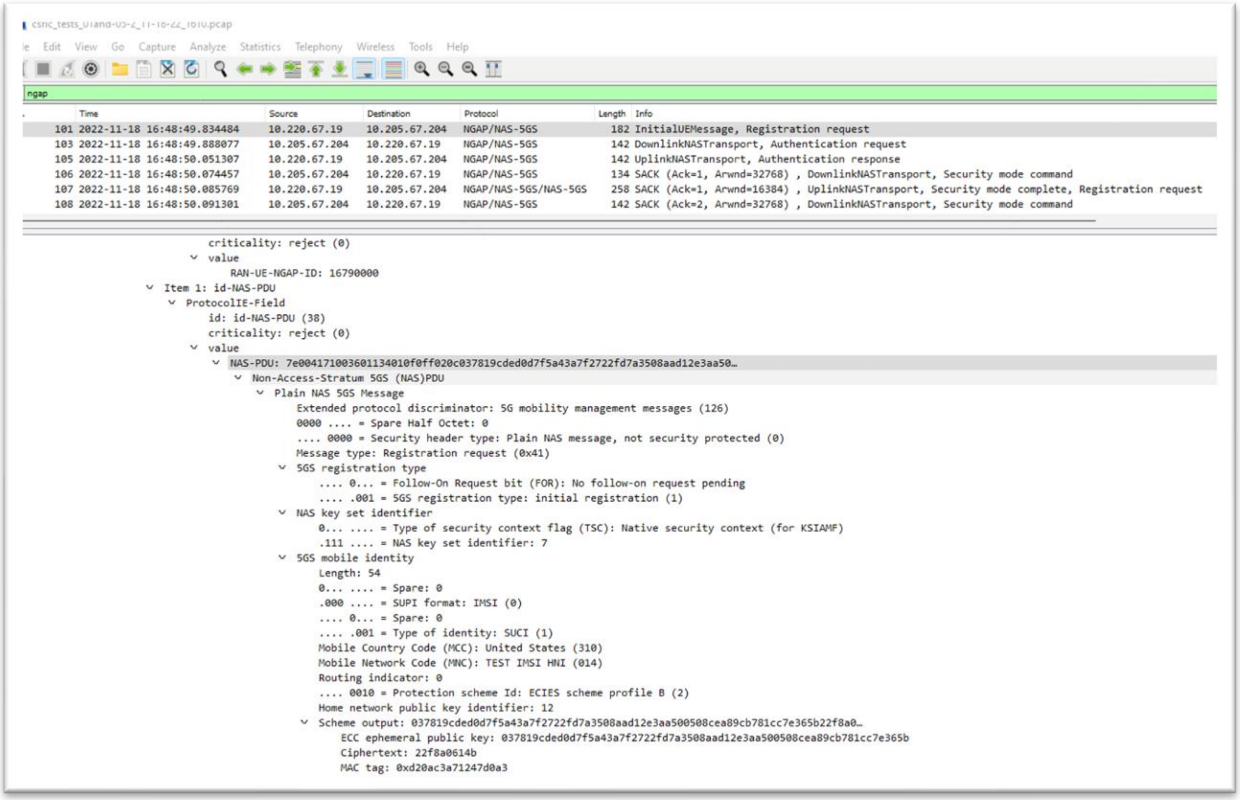


Figure 9: Test Case SA-01 Initial UE Message

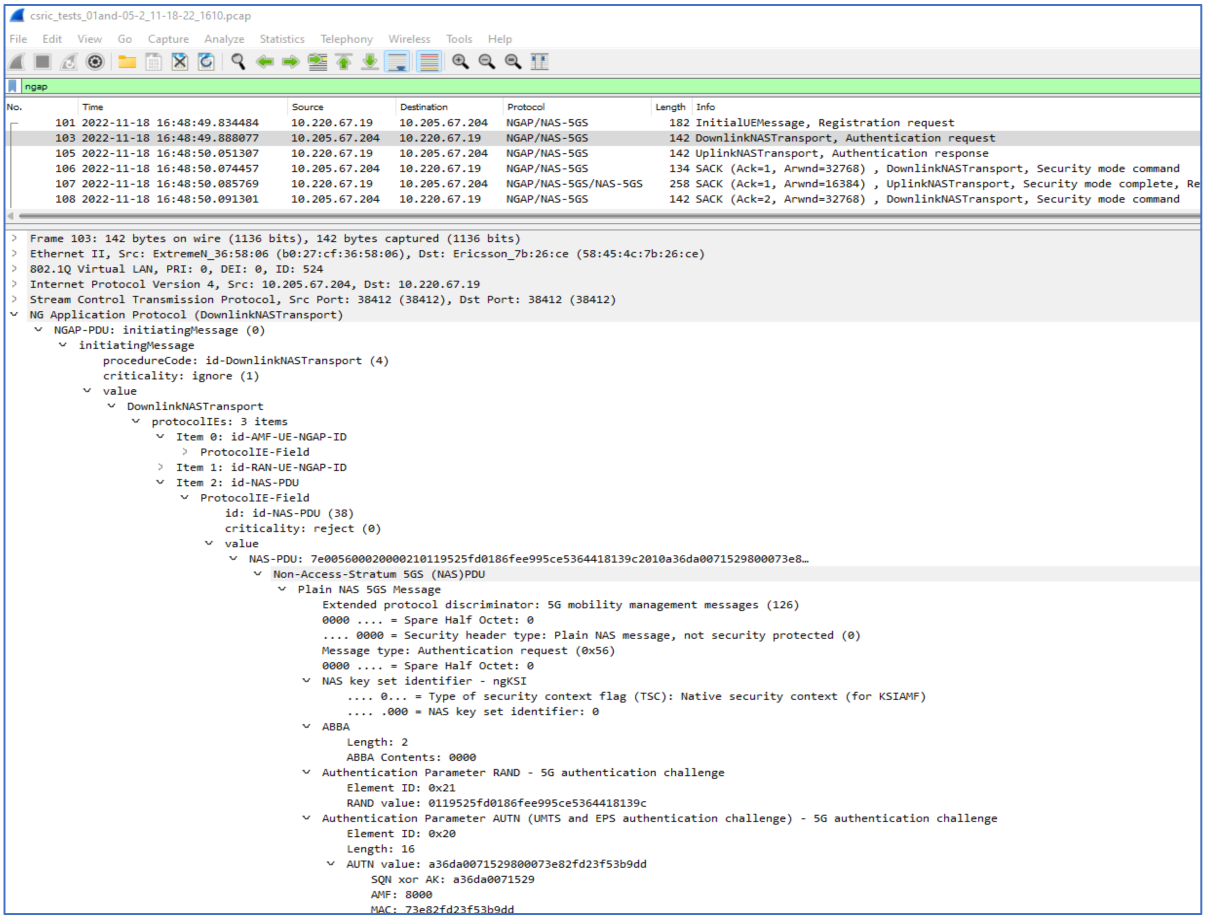


Figure 10: AMF/Core Authentication Request

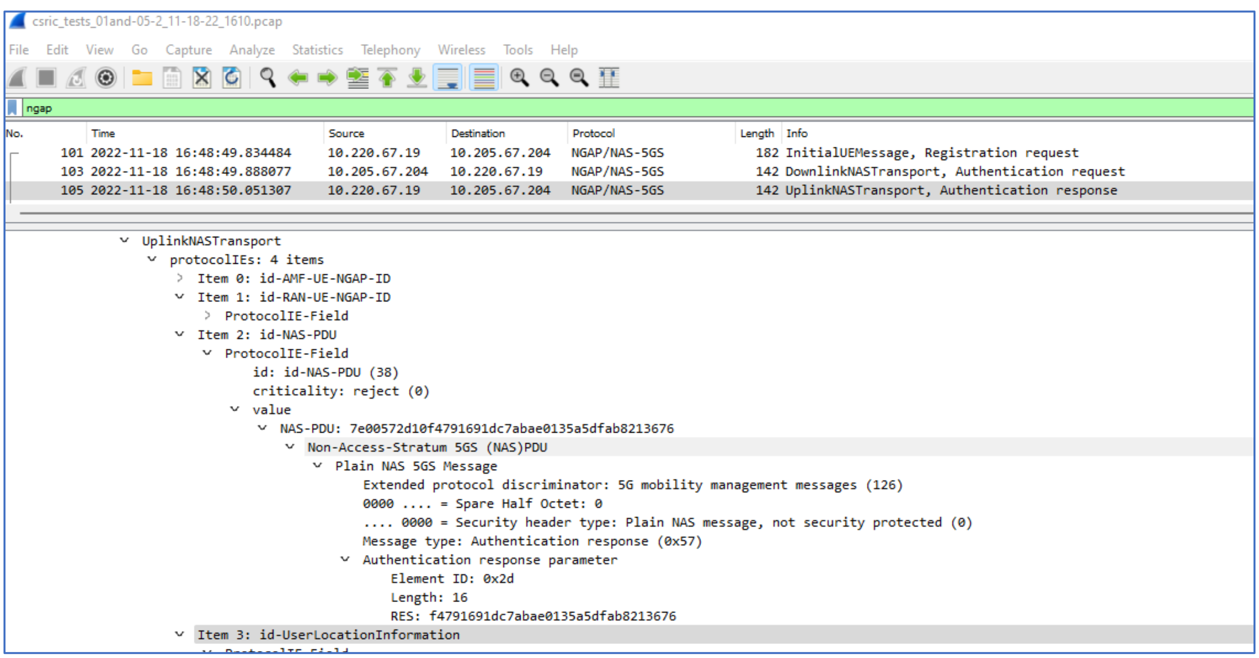


Figure 11: UE Authentication Response

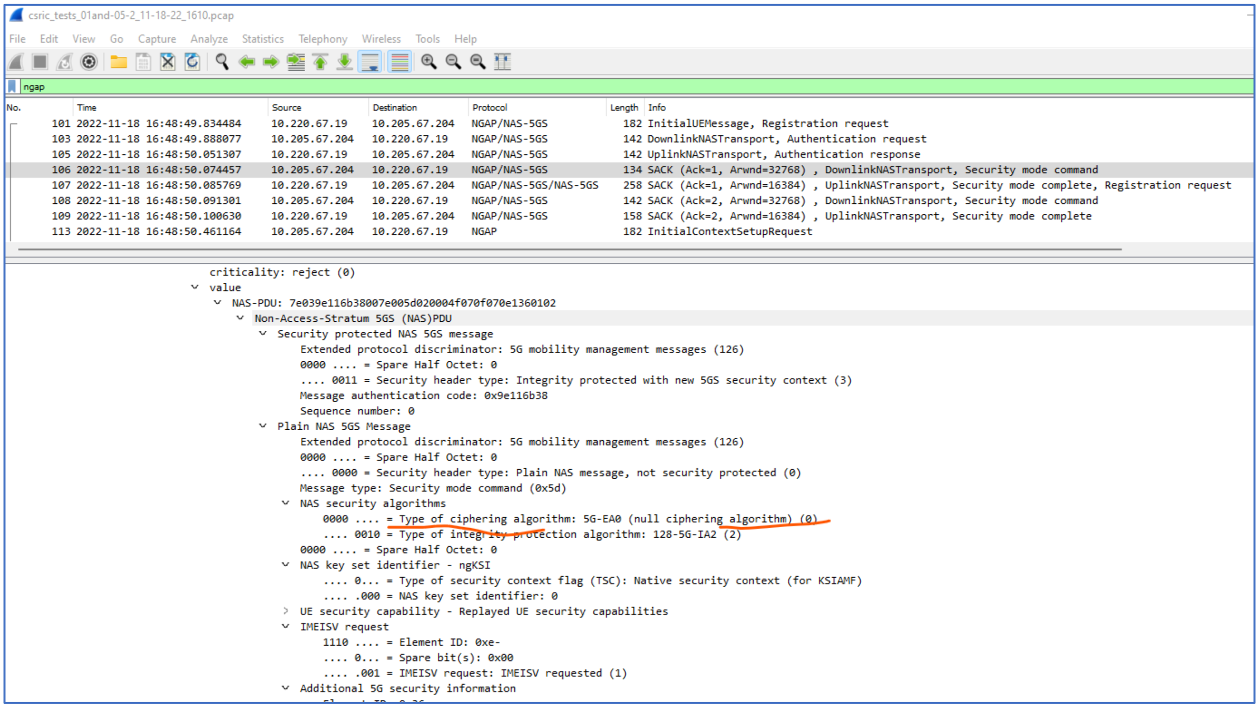


Figure 12: Core Ciphering Algorithm in use – NULL ciphering

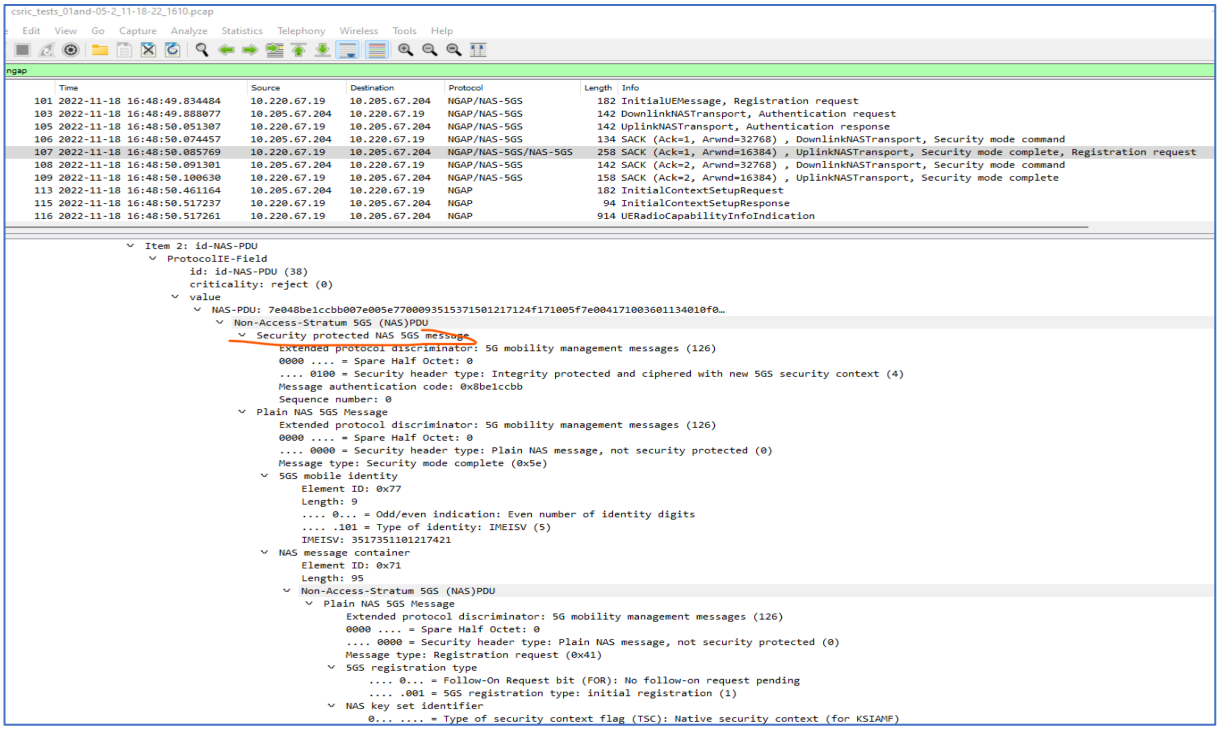


Figure 13: Core Ciphering Algorithm in use –Uplink NAS Transport NULL ciphering

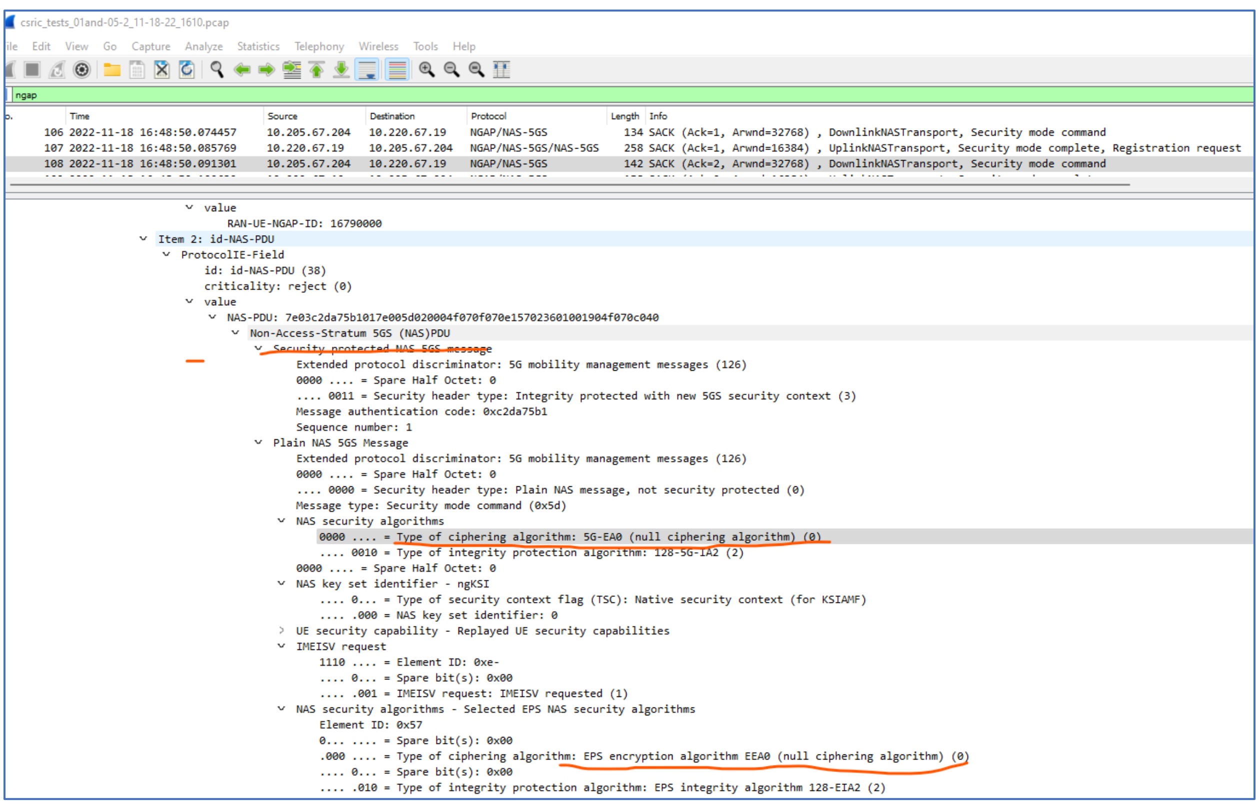


Figure 14: Core Ciphering Algorithm in use – Downlink NAS Transport NULL ciphering

Part 2: Using NEA2 Encryption

In the second part of the experiment, we changed the core setting to use NEA2, the 128-NEA2 cipher algorithm as shown in Figure 8 above. As above, the core and UE exchange an authentication request and response (Figure 15 and Figure 16). Figure 17 shows the downlink message indicating the NEA2 cipher algorithm is to be used. Subsequently, all information transmitted is encrypted, as indicated by the inability of the messages to be deciphered in Figure 18 and Figure 19.

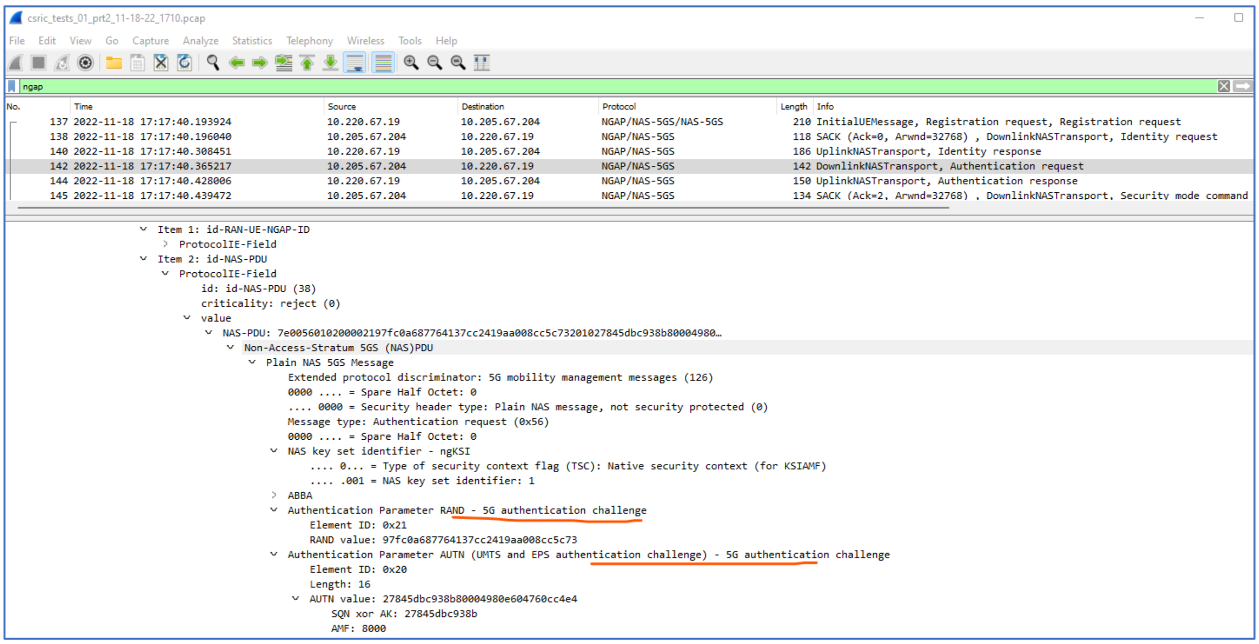


Figure 15: NEA2 AMF/CORE Authentication Request

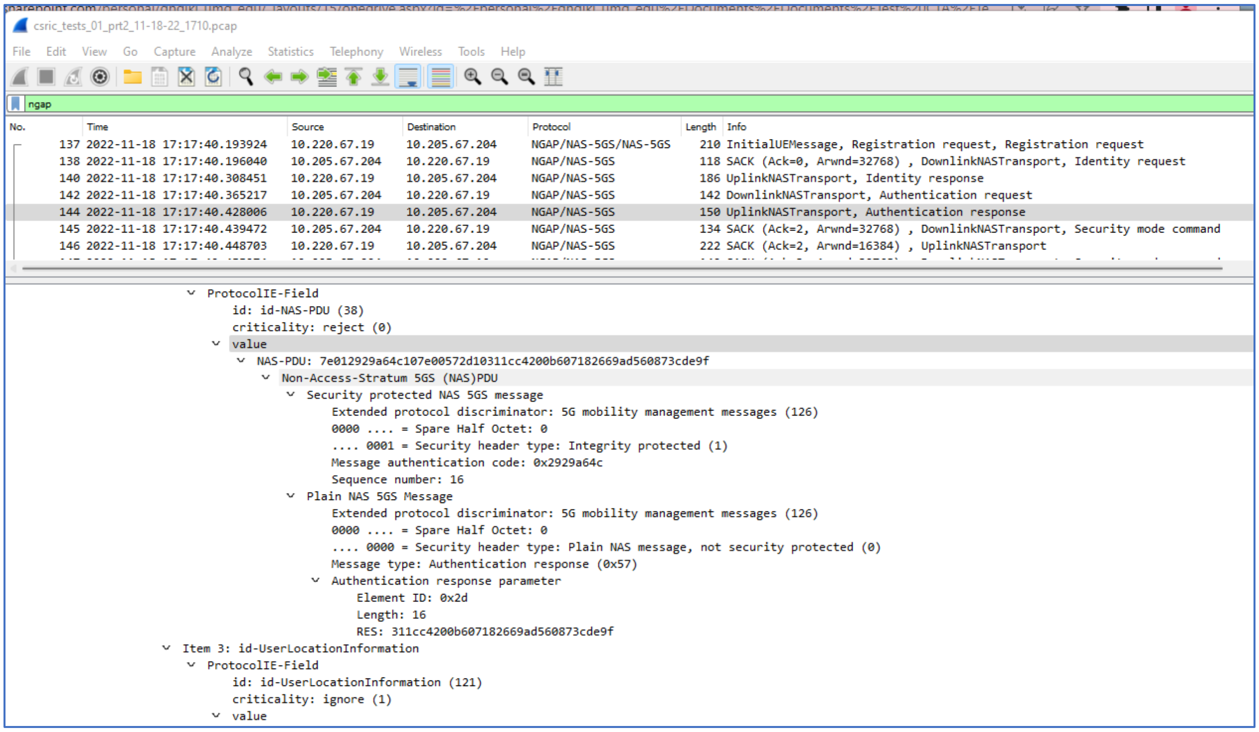


Figure 16: NEA2 AMF/UE Authentication Response

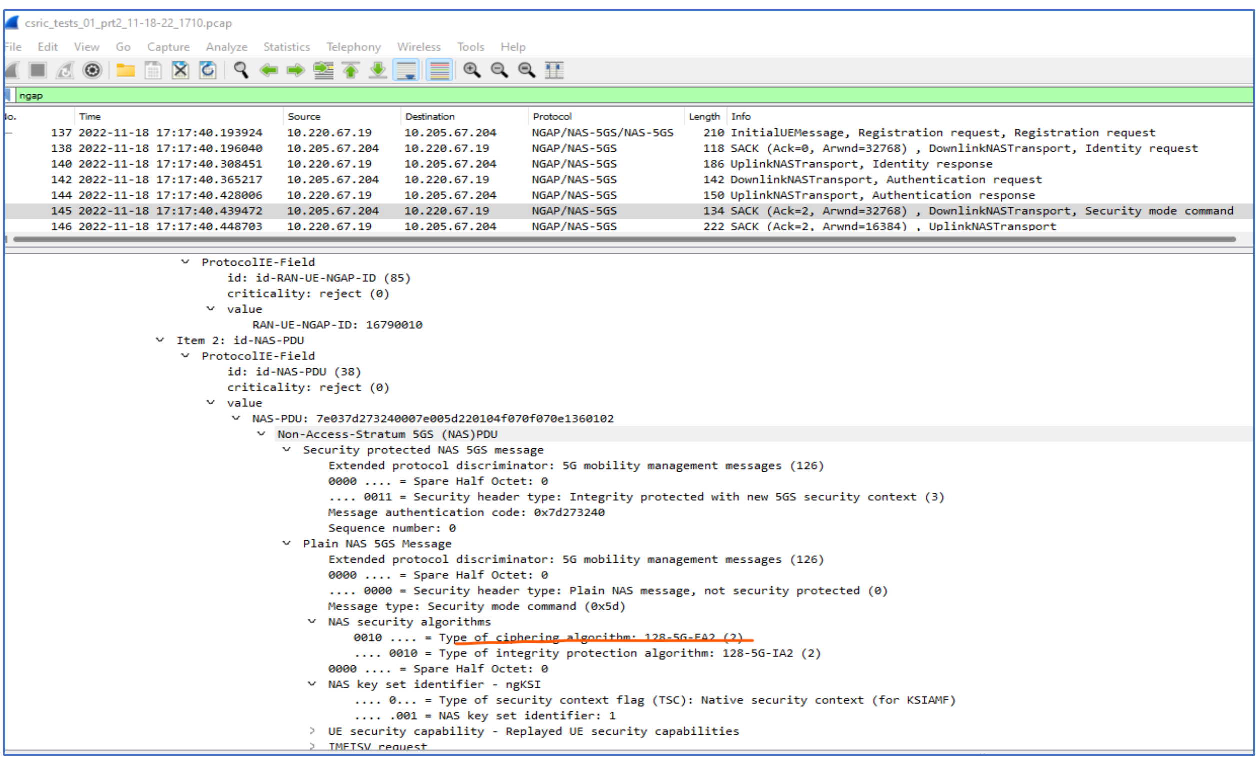


Figure 17: NEA2 Ciphering Command

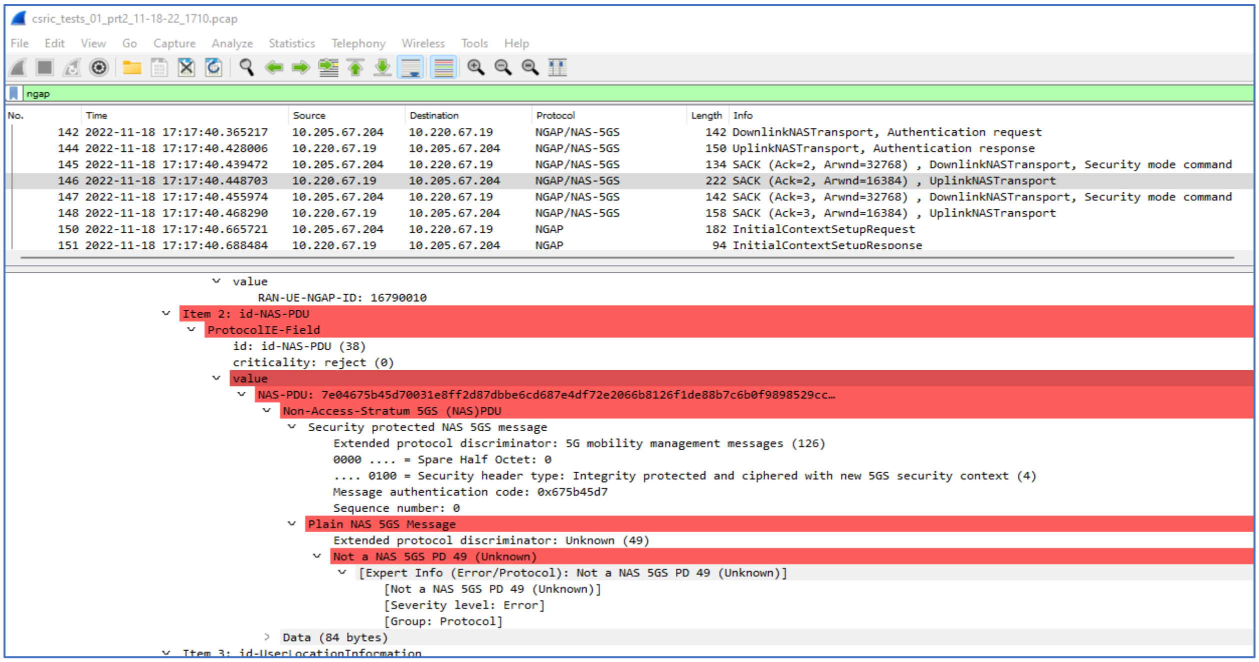


Figure 18: NEA2 Uplink NAS Transport Ciphering – Uplink Transport Data is encrypted

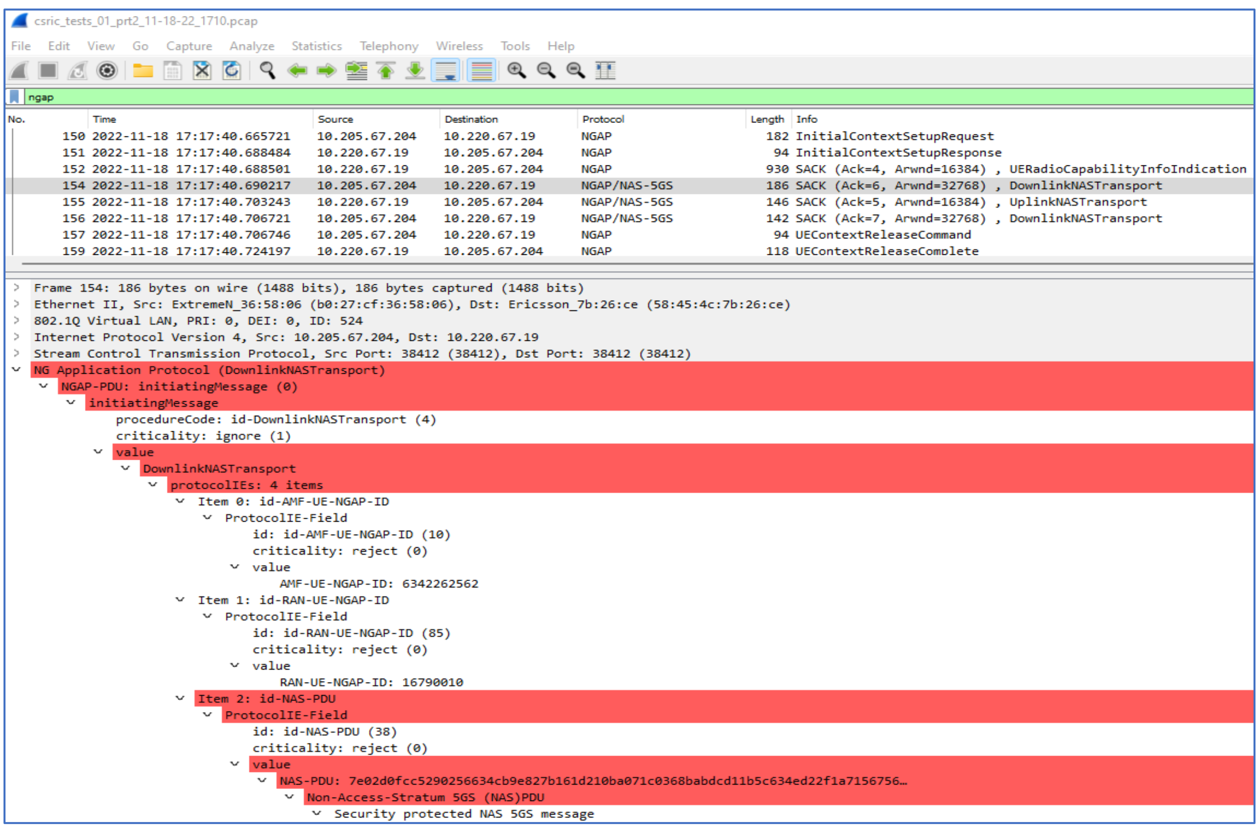


Figure 19: NEA2 Downlink NAS Transport Ciphering – Downlink Transport Data is encrypted

Success Criteria:
Only NAS messages without user identities (e.g. SUPI, IMEI, etc) are exchanged between smartphone and AMF prior to establishing an encrypted channel. These NAS messages may contain temporary identifiers (TMSI, GUTI, etc.).

After establishment of the encrypted channel, all NAS messages are encrypted, enabling user identity information to be safely exchanged. If the encrypted channel is disabled, messages containing user identities are exchanged between the UE and AMF without encryption.

Results

Condition	Status
Only non-user information is observable prior to NAS encryption	Only SUCI is transmitted
After NAS encryption, all NAS messages are encrypted	Messages are encrypted with NEA2
If the encrypted channel is disabled, messages containing user identities are exchanged between UE and AMF without encryption	Message details are visible with NEA0
Overall Test	Success

Test Case 2: CSRIC 7 WG 3 – RRC Signaling Confidentiality

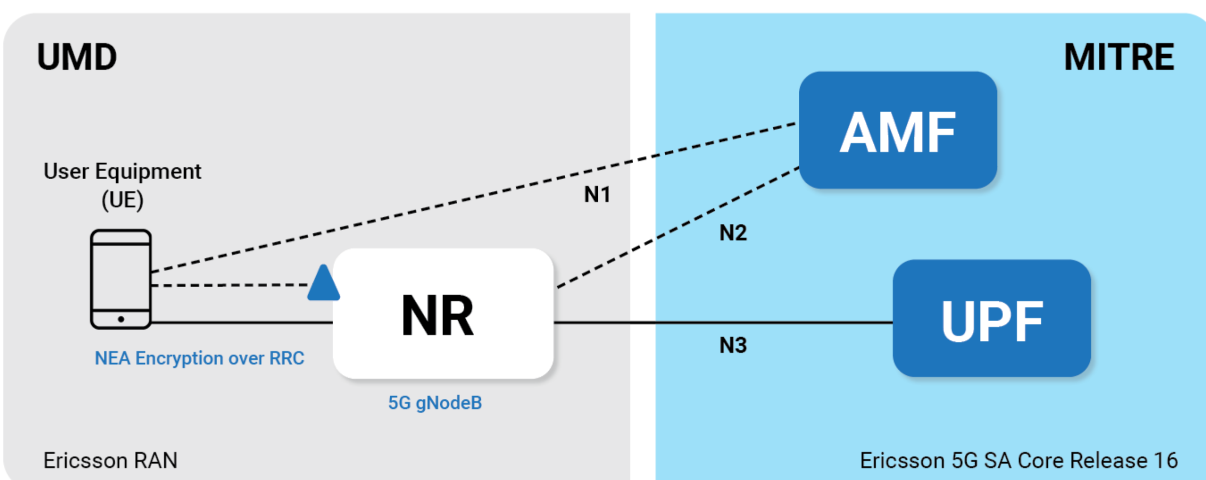


Figure 20: Test Case SA-02 Configuration

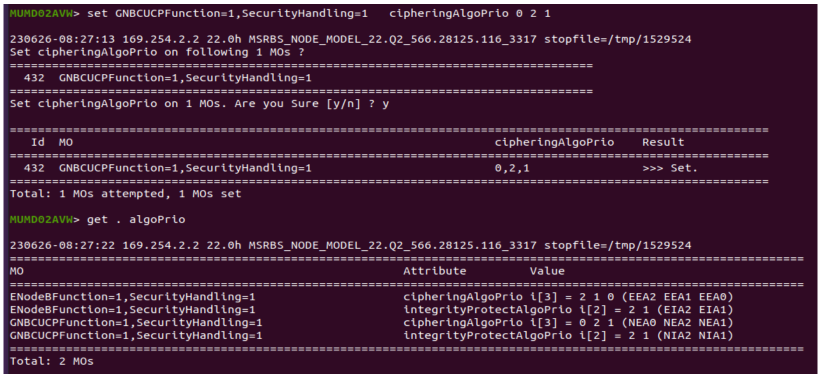
Test Case ID: TC-SA-02
Description:
3GPP TS 33.501 specifies mandatory (e.g., requires vendor implementation) support for protection of the RRC signaling confidentiality, but optional for service providers to use. Given this standards requirement, CSRIC VII recommends protection of the RRC-signaling confidentiality. Only non-user identity related information shall be conveyed prior to security context being established.

This test involves first demonstrating the visibility of identity-related data when no encryption (NULL scheme) is used and then subsequently demonstrating the concealment of that data when RRC encryption is enabled.

Test points used:

Used	Test Point	Description and Use
	TP1-S	Wireshark running on laptop connected to Sierra Wireless card; captures packets originating at and destined to UE laptop
	TP1-MTP	Laptop connected to Qualcomm MTP; QXDM allows access to low-level data
X	TP2	WaveJudge interface
	TP3	Wireshark running on laptop connected to RAN-side R6K router; can capture packets inside the tunnel (encrypted packets when IPsec tunnel is enabled)
	TP4	tcpdump running on laptop connected to port of RAN-side Pluribus switch used to capture, modify, and inject packets on the “untrusted link”
	TP5	tcpdump running on port of core-side R6K router inside the IPsec tunnel (encrypted packets when IPsec tunnel is enabled) used to monitor packets on the “untrusted link”
	TP6	tcpdump running on port of core-side R6K router outside the IPsec tunnel (i.e., before IPsec encryption or after IPsec decryption) used to monitor packets at the interface to the DMC
	TP7	CNOM tool accessing DMC messages
	TP8	Applications running on application server in MITRE facility

This test has two parts: 1) with NEA0 (no encryption) and 2) with NEA2 activated to encrypt RRC signaling. Figure 21 shows the command line interface setting the RRC encryption on the RAN to use NULL (NEA0) as the first priority.



```
MUMD02AVW> set GNBCUCPFunction=1,SecurityHandling=1   cipheringAlgoPrio 0 2 1

230626-08:27:13 169.254.2.2 22.0h MSRBS_NODE_MODEL_22.Q2_566.28125.116_3317 stopfile=/tmp/1529524
Set cipheringAlgoPrio on following 1 MOs ?
===================================================================================
  432  GNBCUCPFunction=1,SecurityHandling=1
===================================================================================
Set cipheringAlgoPrio on 1 MOs. Are you Sure [y/n] ? y

=============================================================================================================
   Id  MO                                                                   cipheringAlgoPrio    Result
=============================================================================================================
  432  GNBCUCPFunction=1,SecurityHandling=1                                 0,2,1                >>> Set.
=============================================================================================================
Total: 1 MOs attempted, 1 MOs set

MUMD02AVW> get . algoPrio

230626-08:27:22 169.254.2.2 22.0h MSRBS_NODE_MODEL_22.Q2_566.28125.116_3317 stopfile=/tmp/1529524
==================================================================================================================
MO                                                       Attribute        Value
==================================================================================================================
ENodeBFunction=1,SecurityHandling=1                      cipheringAlgoPrio i[3] = 2 1 0 (EEA2 EEA1 EEA0)
ENodeBFunction=1,SecurityHandling=1                      integrityProtectAlgoPrio i[2] = 2 1 (EIA2 EIA1)
GNBCUCPFunction=1,SecurityHandling=1                     cipheringAlgoPrio i[3] = 0 2 1 (NEA0 NEA2 NEA1)
GNBCUCPFunction=1,SecurityHandling=1                     integrityProtectAlgoPrio i[2] = 2 1 (NIA2 NIA1)
==================================================================================================================
Total: 2 MOs
```

Figure 21: Setting RAN RRC encryption to NEA0

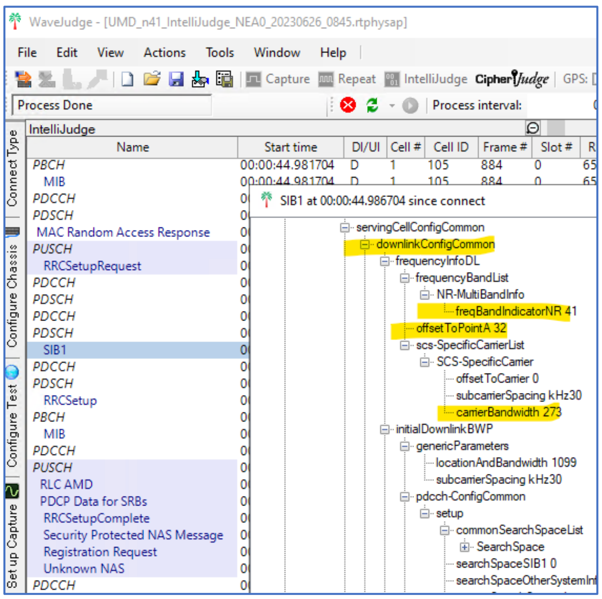


Figure 22: WaveJudge/IntelliJudge capture of N41 specifications

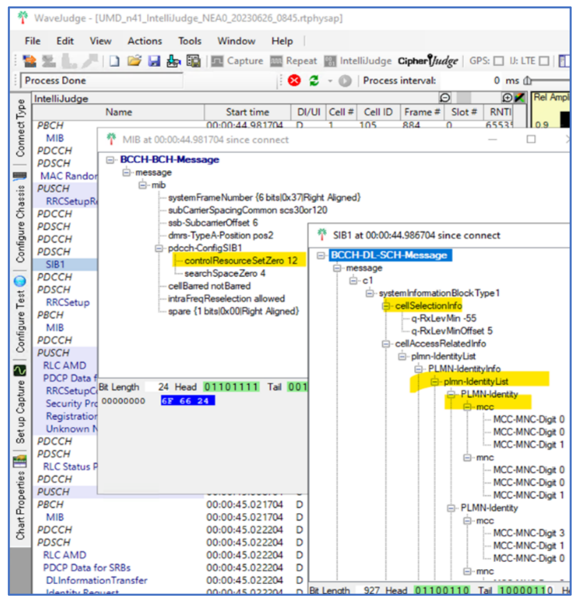


Figure 23: WaveJudge/IntelliJudge capture of PLMN information

Figure 22 and Figure 23 show the UE attach process as captured by the WaveJudge/IntelliJudge tool. Figure 24 and Figure 25 show WaveJudge captures of messages processed with the NEA0 (NULL) algorithm. Because the ciphering algorithm in use is NEA0, all the “security protected” messages can be read in clear text.

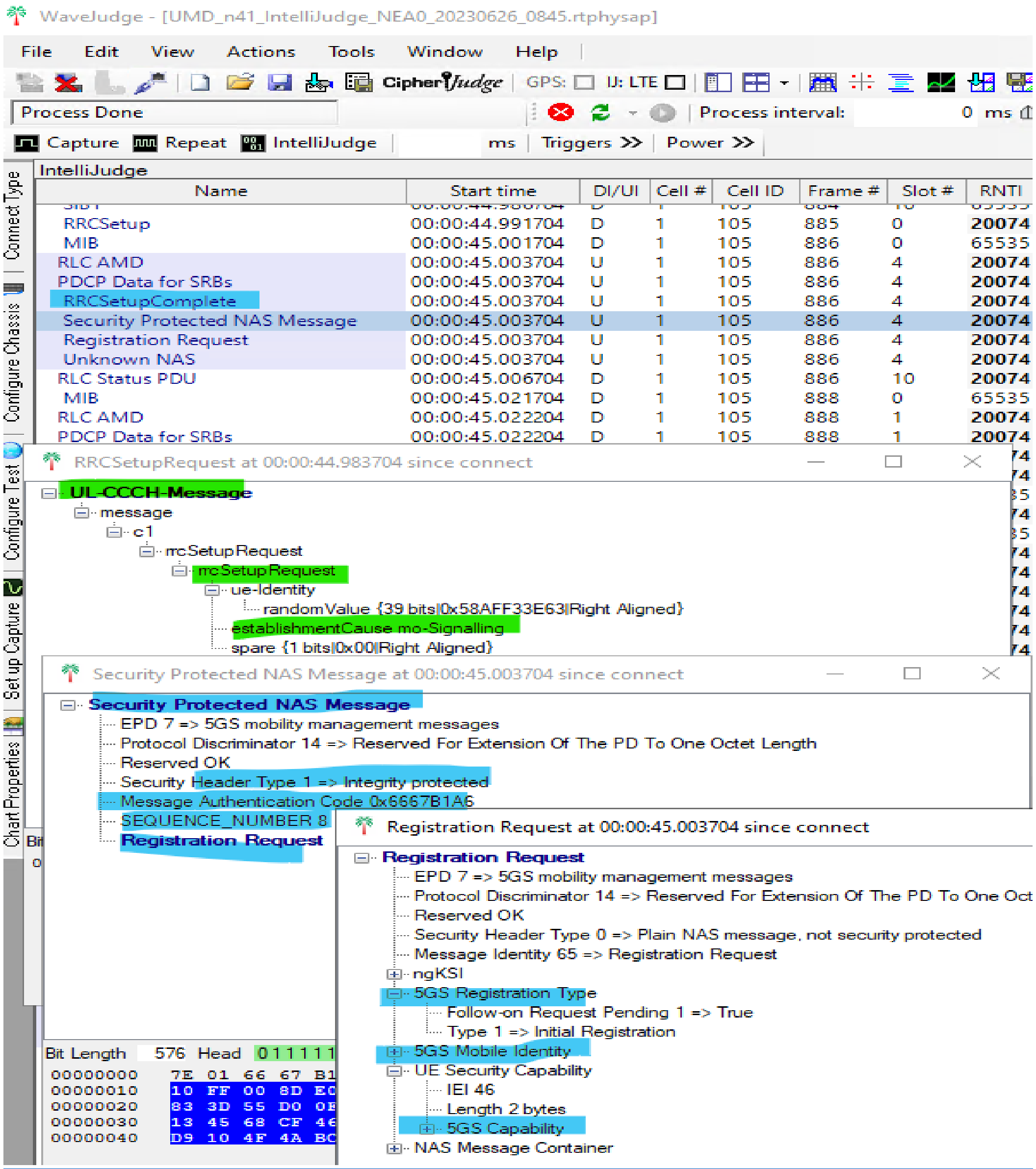


Figure 24: Unencrypted RRC messages

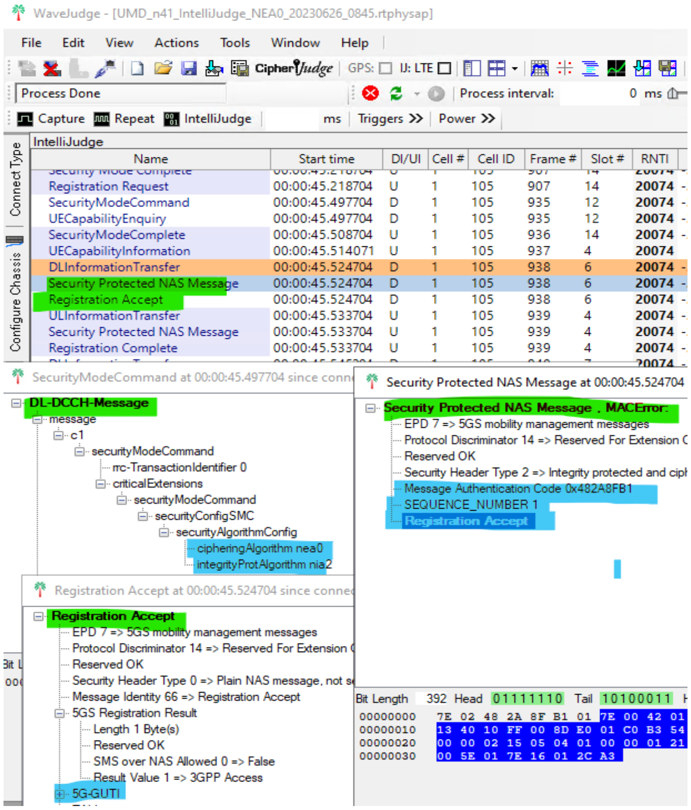


Figure 25: Decipherable RRC messages when using NEA0

For the second part of the test, we set NEA2 as the first priority for RRC encryption for the RAN, as shown in Figure 26. Figure 27 and Figure 28 show WaveJudge windows indicating that, because the ciphering algorithm in use is NEA2, all the RRC and upper layer are security protected and the contents cannot be deciphered.

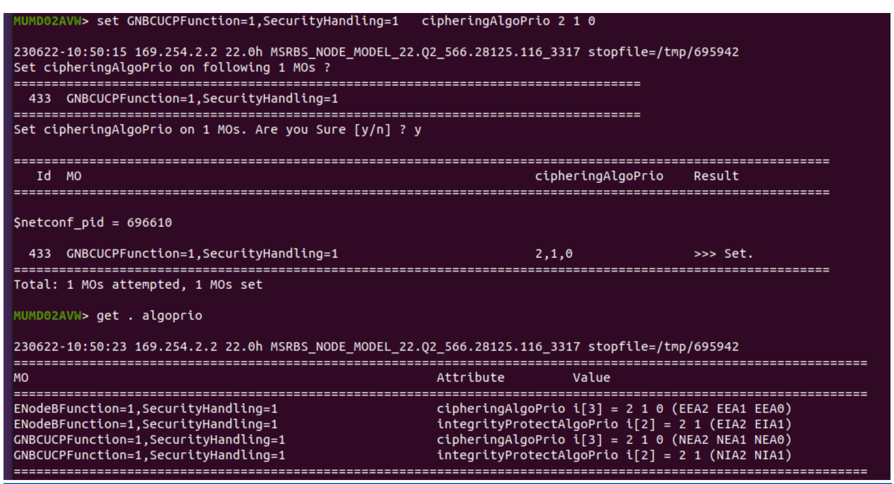


Figure 26: Setting RAN RRC encryption to NEA2

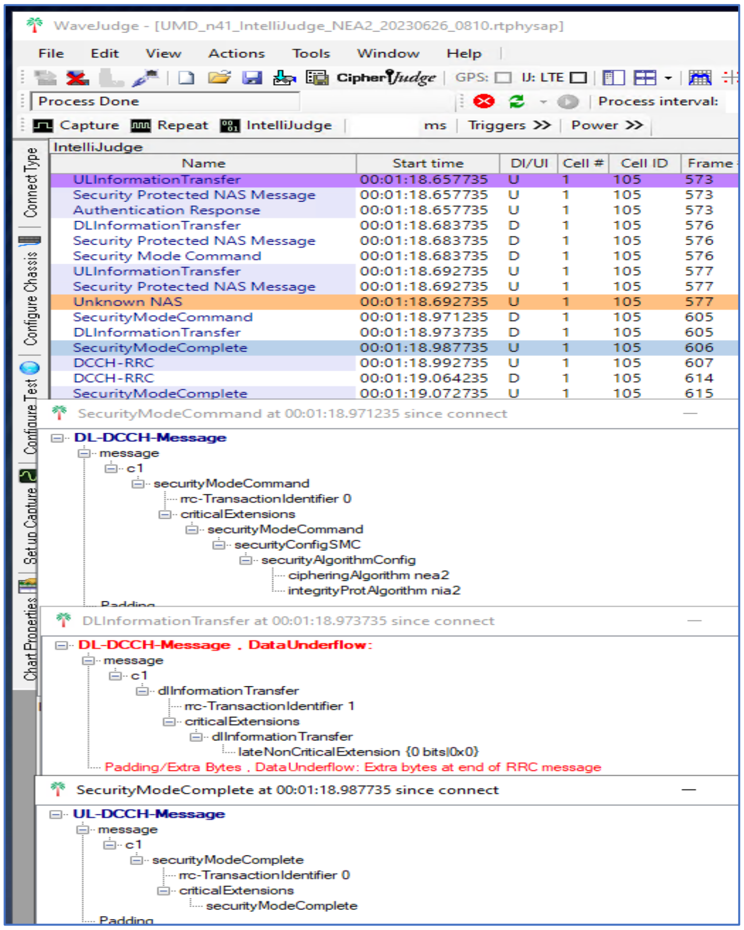


Figure 27: Protected RRC Messages

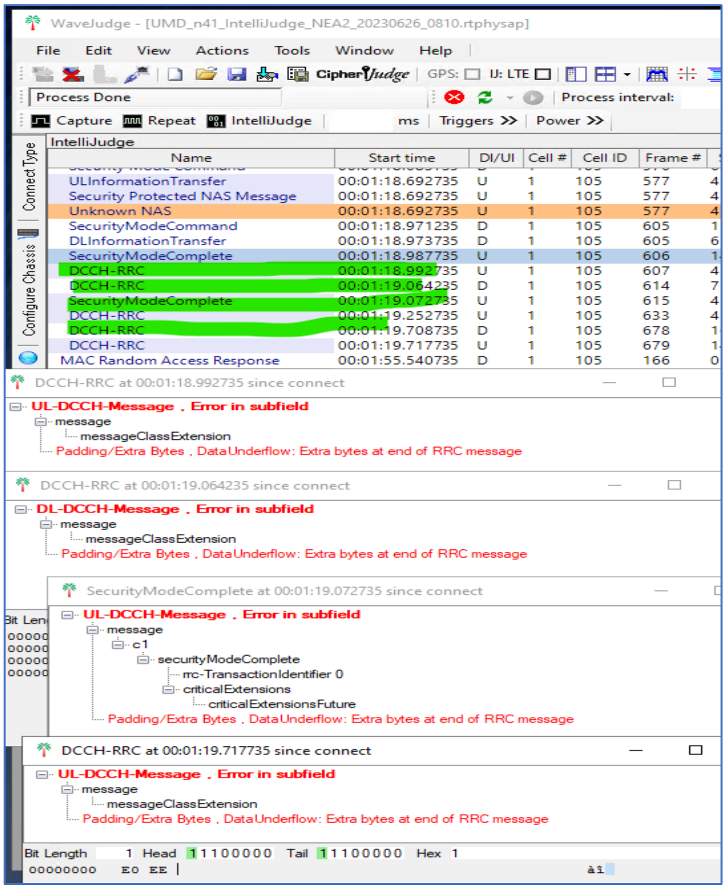


Figure 28: Protected RRC Messages

Success Criteria:

RRC messages are observable over the RF channel when RRC encryption is disabled, and RRC messages are no longer observable when RRC encryption is enabled.

Results

Condition	Status
RRC messages are observable over the RF channel when RRC encryption is disabled (through use of NEA0 algorithm)	Contents of RRC messages are fully decipherable by WaveJudge
RRC messages are no longer observable when RRC encryption is enabled (through use of NEA2 algorithm)	WaveJudge shows contents of encrypted messages as “Extra bytes at end of RRC message”
Overall Test	Success

Test Case 3: CSRIC 7 WG 3 – Access Stratum User Plane Confidentiality

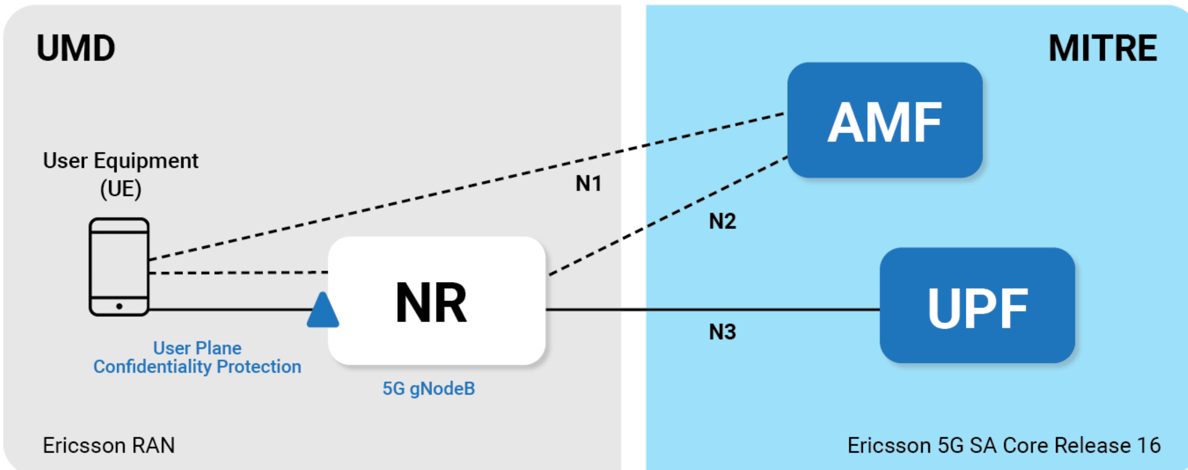


Figure 29: Test Case SA-03 Configuration

Test Case ID: TC-SA-03
Description:
3GPP TS 33.501 specifies mandatory (e.g., requires vendor implementation) support for protection of the user plane confidentiality, but optional for service providers to use. Given this standards requirement, CSRIC VII recommends user plane confidentiality protection over the access stratum be done at PDCP layer.

This test involves demonstrating that when confidentiality protection for the user plane is applied at the PDCP layer, no layers below PDCP are confidentiality-protected. User data sent via the UPF may be confidentiality protected.

This test also involves implementing and confirming user plane confidentiality protection over the access stratum at the PDCP layer. Layers below PDCP are not confidentiality-protected.

Test points used:

Used	Test Point	Description and Use
	TP1-S	Wireshark running on laptop connected to Sierra Wireless card; captures packets originating at and destined to UE laptop
	TP1-MTP	Laptop connected to Qualcomm MTP; QXDM allows access to low-level data
X	TP2	WaveJudge interface
	TP3	Wireshark running on laptop connected to RAN-side R6K router; can capture packets inside the tunnel (encrypted packets when IPsec tunnel is enabled)
	TP4	tcpdump running on laptop connected to port of RAN-side Pluribus switch used to capture, modify, and inject packets on the "untrusted link"
	TP5	tcpdump running on port of core-side R6K router inside the IPsec tunnel (encrypted packets when IPsec tunnel is enabled) used to monitor packets on the "untrusted link"
	TP6	tcpdump running on port of core-side R6K router outside the IPsec tunnel (i.e., before IPsec encryption or after IPsec decryption) used to monitor packets at the interface to the DMC
	TP7	CNOM tool accessing DMC messages
	TP8	Applications running on application server in MITRE facility

We were unable to capture user plane messages on the WaveJudge and, as a result, were unable to verify over-the-air encryption of user plane data.

Success Criteria:
Once encrypted channel is established, user plane data will not be observable in the recorded data.

Confidentiality protection for the user plane is applied at the PDCP layer via 128-bit NEA algorithms.

Results

Condition	Status
When encryption is disabled, all UP data will be observable	Unable to capture user plane messages with WaveJudge
When the encrypted channel is established, UP data will not be observable in the recorded data	Unable to capture user plane messages with WaveJudge
Overall Test	Limited by Test Capability

Test Case 4: CSRIC 7 WG 3 – Access Stratum User Plane Integrity

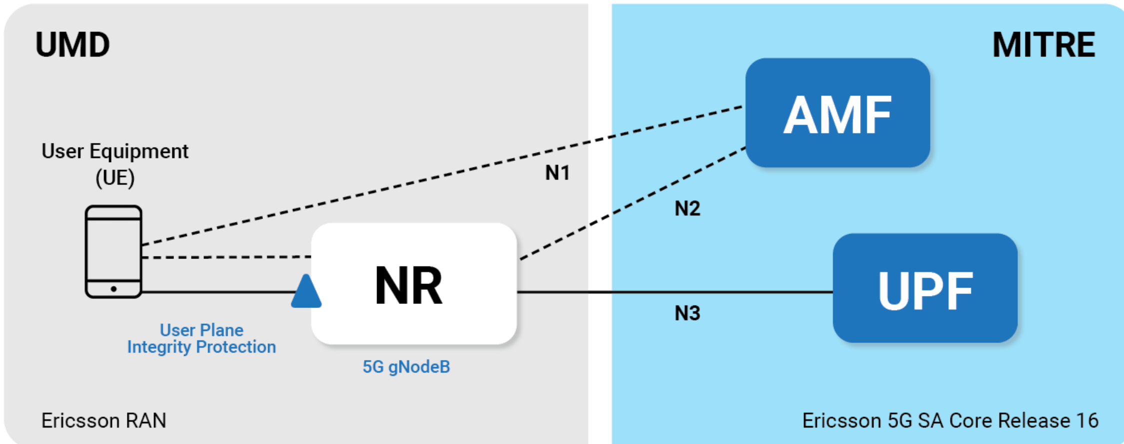


Figure 30: Test Case SA-04 Configuration

Test Case ID: TC-SA-04
Description:
3GPP TS 33.501 requires UE to support integrity protection and replay protection of user data between the UE and the gNB, but the data rates at which it is supported are different between releases 15 and 16, and it is optional for service providers to use this feature.

CSRIC VII recommends that device OEMs and network infrastructure vendors support the Release 16 full rate capability, along with 128-NIA3 as defined in Annex D of 3GPP TS 33.501, and for operators to implement according to the service requirement.
CSRIC VII recommends that user data integrity is mandatory for Release 16 U.S. deployments.

The Packet Data Convergence Protocol, as specified in TS 38.323 as between the UE and the NG-RAN, is responsible for user plane data integrity protection.

Test points used:

Used	Test Point	Description and Use
	TP1-S	Wireshark running on laptop connected to Sierra Wireless card; captures packets originating at and destined to UE laptop
	TP1-MTP	Laptop connected to Qualcomm MTP; QXDM allows access to low-level data
X	TP2	WaveJudge interface
	TP3	Wireshark running on laptop connected to RAN-side R6K router; can capture packets inside the tunnel (encrypted packets when IPsec tunnel is enabled)
	TP4	tcpdump running on laptop connected to port of RAN-side Pluribus switch used to capture, modify, and inject packets on the "untrusted link"
	TP5	tcpdump running on port of core-side R6K router inside the IPsec tunnel (encrypted packets when IPsec tunnel is enabled) used to monitor packets on the "untrusted link"
	TP6	tcpdump running on port of core-side R6K router outside the IPsec tunnel (i.e., before IPsec encryption or after IPsec decryption) used to monitor packets at the interface to the DMC
	TP7	CNOM tool accessing DMC messages
	TP8	Applications running on application server in MITRE facility

As with TC-SA-03, the WaveJudge/IntelliJudge was unable to capture user plane messages. As a result, we were unable to confirm over-the-air integrity protection. However, the PDU Establishment message, as shown in Figure 31, does indicate the intent to apply integrity protection at the full rate.

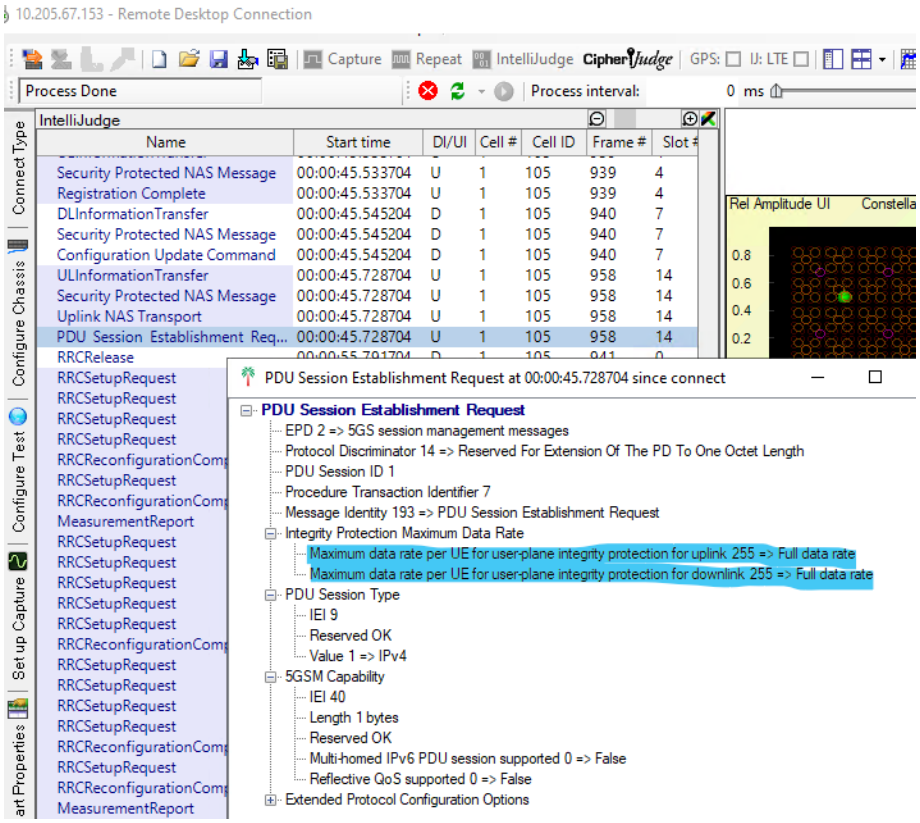


Figure 31: WaveJudge capture of PDU Establishment message

Success Criteria:

Integrity protection for the user plane is applied full-rate at the PDCP layer.

Examine data collected by the RF monitoring tool for the user plane messages. Confirm the integrity check fields are populated.

Results

Condition	Status
Integrity protection for the user plane is applied full-rate at the PDCP layer	PDU Establishment message indicates integrity protection applied at full rate
Integrity check fields are populated	Unable to capture and read user plane messages over the air with WaveJudge
Overall Test	Limited by Test Capability

Test Case 5: CSRIC 7 WG 3 – SUPI/SUCI Privacy Enabled

Test Case ID: TC-SA-05-1

Description:

3GPP TS 33.501 specifies mandatory (e.g., requires vendor implementation) support for protection of the SUPI/IMSI privacy; however, 3GPP allows for some exceptions where the Subscription Concealed Identifier (SUCI) may use the null scheme (i.e., the identity is not protected).

CSRIC VII recommends that devices and networks in the U.S. use IMSI privacy (SUCI) and not use the null scheme, except when the UE is requesting emergency services.

It is recommended that no other exceptions allowed by 3GPP in Release 16 (for null scheme SUCI) be used by devices or networks in the U.S. This may result in roaming 5G devices configured by operators from outside the U.S. being unable to connect to 5G SA networks, but that they use 4G LTE networks instead.

To avoid identifying a handset by its Subscription Permanent Identifier (SUPI), 5G uses the subscription concealed identifier (SUCI) to encrypt the SUPI to exchange identity information between the UE and 5G NR. SUPI/SUCI privacy is used for all services, except emergency services and non-authenticated roaming emergency calls.

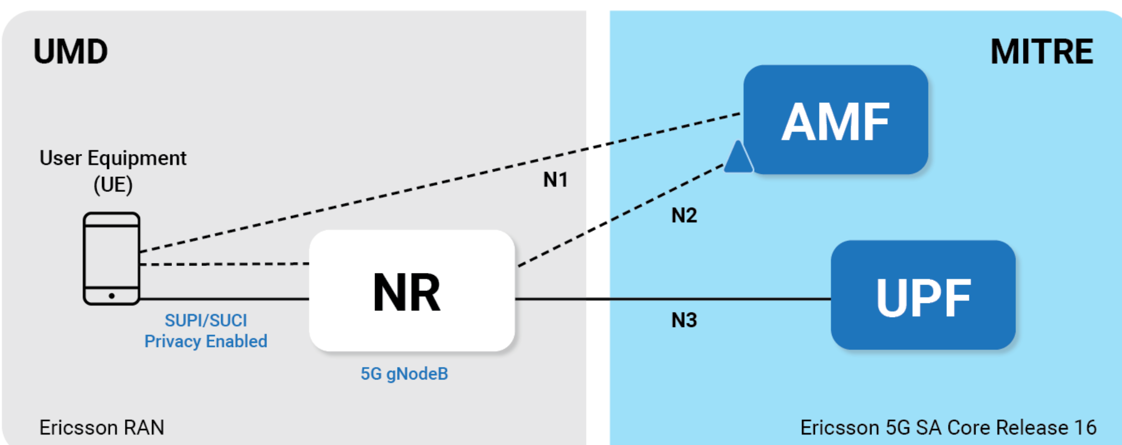


Figure 32: Test Case SA-05 Configuration

We used logs from CSRIC Test TC-SA-01 above, with highest priority set for NEA0 (null algorithm).

Test points used:

Used	Test Point	Description and Use
	TP1-SW	Wireshark running on laptop connected to Sierra Wireless card; captures packets originating at and destined to UE laptop
	TP1-MTP	Laptop connected to Qualcomm MTP; QXDM allows access to low-level data
	TP2	WaveJudge interface
	TP3	Wireshark running on laptop connected to RAN-side R6K router; can capture packets inside the tunnel (encrypted packets when IPsec tunnel is enabled)
	TP4	tcpdump running on laptop connected to port of RAN-side Pluribus switch used to capture, modify, and inject packets on the "untrusted link"
	TP5	tcpdump running on port of core-side R6K router inside the IPsec tunnel (encrypted packets when IPsec tunnel is enabled) used to monitor packets on the "untrusted link"
X	TP6	tcpdump running on port of core-side R6K router outside the IPsec tunnel (i.e., before IPsec encryption or after IPsec decryption) used to monitor packets at the interface to the DMC
	TP7	CNOM tool accessing DMC messages
	TP8	Applications running on application server in MITRE facility

Figure 33 shows the initial UE registration message in which the type of identity is SUCI. Figure 34 shows the uplink NAS transport message with registration request that also provides the type of identity as SUCI. Lastly, Figure 35 shows the downlink NAS transport message indicating the encryption algorithm to use is NEA0.

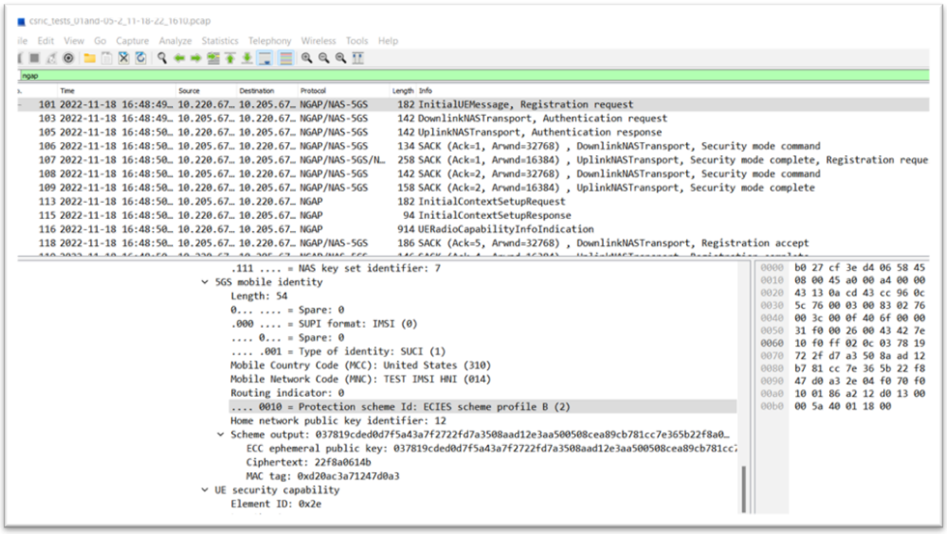


Figure 33: UE with profile B – source core-side R6K

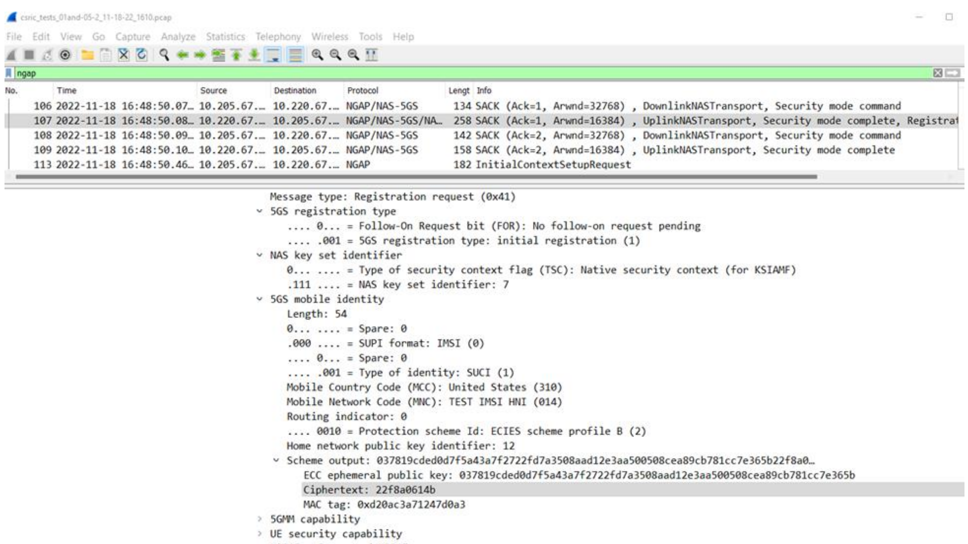


Figure 34: UE Registration Complete Message- Source Core-side R6k

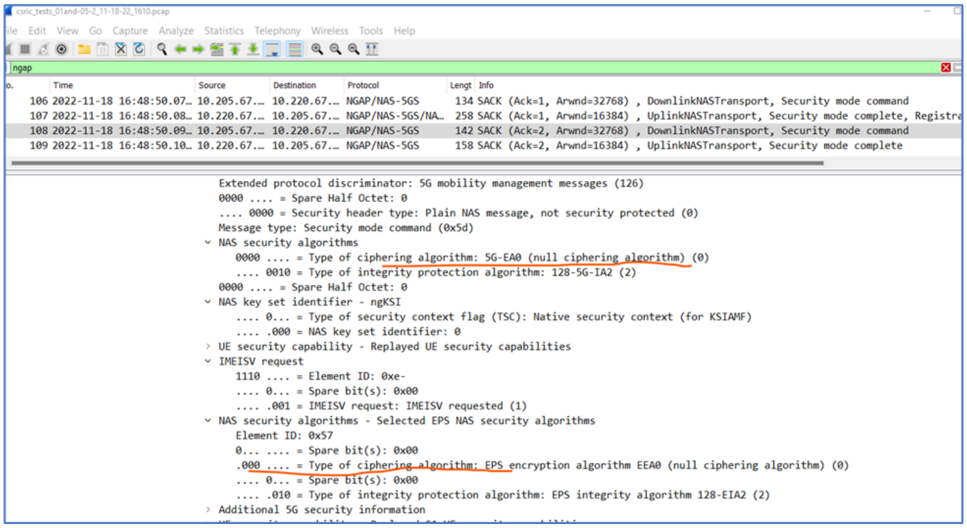


Figure 35: NULL scheme in use - Source Core-side R6K

Success Criteria:
Successful registration with encrypted SUPI.

Results

Condition	Status
Use of SUCI by UE in registration process	Success
Successful registration	Success
Overall Test	Success

Test Case 6: CSRIC 7 WG 3 – IPsec on Transport Links

Test Case ID: TC-SA-06
Description:
3GPP TS 33.501 specifies mandatory (e.g., requires vendor support for) network security protection such as IPsec, but optional for service providers to use. Given this standards requirement, CSRIC VII recommends the use of IPsec or use of a tunneling technology for transport (e.g., VPN tunnels) for protection of network security.

This test involves demonstrating that when IPsec is used for confidentiality and integrity protection of user plane and signaling on the N1, N2, and N3 interfaces across the transport link, UP and CP traffic cannot be captured, modified, or injected with new packets.

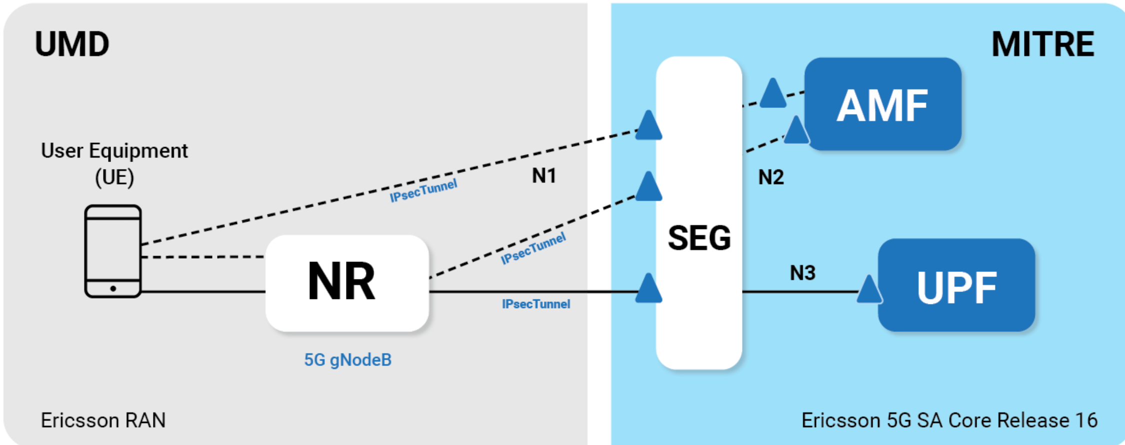


Figure 36: Test Case SA-06 Configuration

Test points used:

Used	Test Point	Description and Use
	TP1-SW	Wireshark running on laptop connected to Sierra Wireless card; captures packets originating at and destined to UE laptop
	TP1-MTP	Laptop connected to Qualcomm MTP; QXDM allows access to low-level data
	TP2	WaveJudge interface
X	TP3	Wireshark running on laptop connected to RAN-side R6K router; can capture packets inside the tunnel (encrypted packets when IPsec tunnel is enabled)
X	TP4	tcpdump running on laptop connected to port of RAN-side Pluribus switch used to capture, modify, and inject packets on the “untrusted link”
X	TP5	tcpdump running on port of core-side R6K router inside the IPsec tunnel (encrypted packets when IPsec tunnel is enabled) used to monitor packets on the “untrusted link”
X	TP6	tcpdump running on port of core-side R6K router outside the IPsec tunnel (i.e., before IPsec encryption or after IPsec decryption) used to monitor packets at the interface to the DMC
	TP7	CNOM tool accessing DMC messages
	TP8	Applications running on application server in MITRE facility

Similar to the previous tests, Network Slice 2 and UE2 with profile B were used for this test. The UE used for Slice 2 was a Sierra Wireless Modem, which is connected and controlled by a laptop outside the Faraday Cage. IPsec for for Slice 2 and control traffic was turned on/off as and when required.

This test has two parts, attempting to capture, modify, and inject user and control plane traffic: 1) without an IPsec tunnel and 2) with an IPsec tunnel.

Part 1: Traffic Modification without Enabled IPsec Tunnel between BBU and Core-Side R6K

For the first part of the test, we ensured that the IPsec tunnel for the transport channel between the RAN and the core was off.

The UE was then restarted and attached to the 5G core, as shown in Figure 37. The larger window in the figure shows the contents of an initial context setup request message captured on the untrusted transport between the RAN and core. The inset figure shows a message within the core regarding the registration of the UE with IMSI used for this test. Figure 38 shows logs of both control plane and user plane messages after UE registration and ping started to the DN server (192.168.59.146). These packets are clearly visible on the transport channel when the IPsec tunnel is down and traffic is unencrypted.

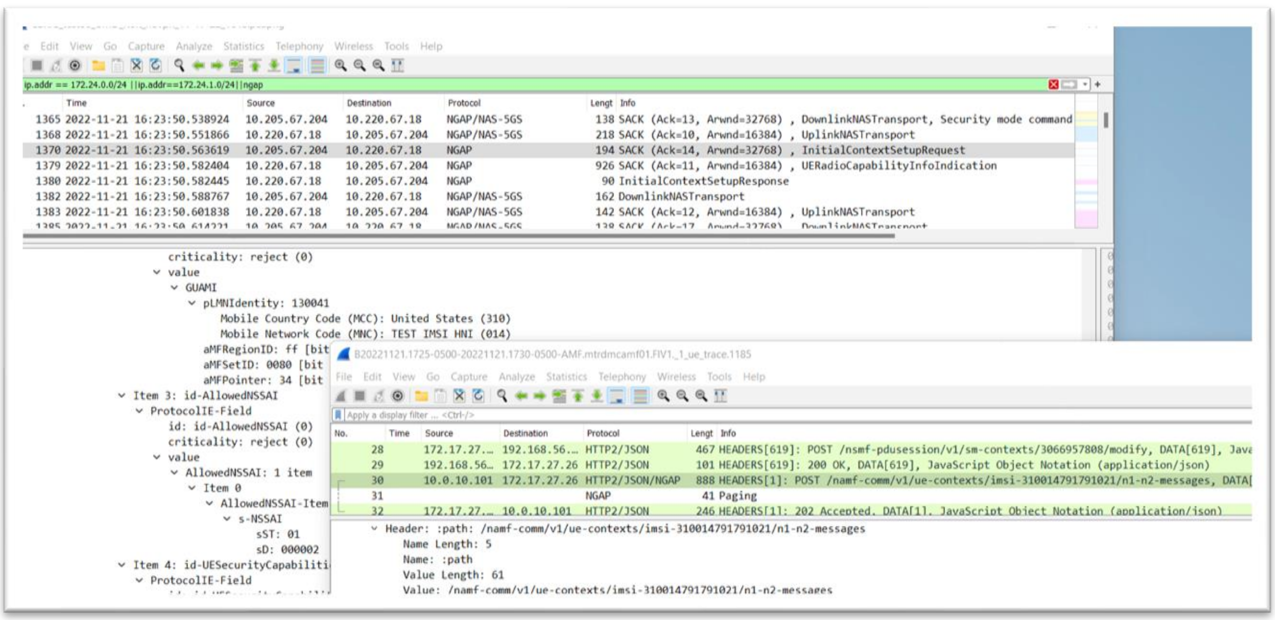


Figure 37: Packet captures at RAN-side switch and at the core showing UE registration over unencrypted transport

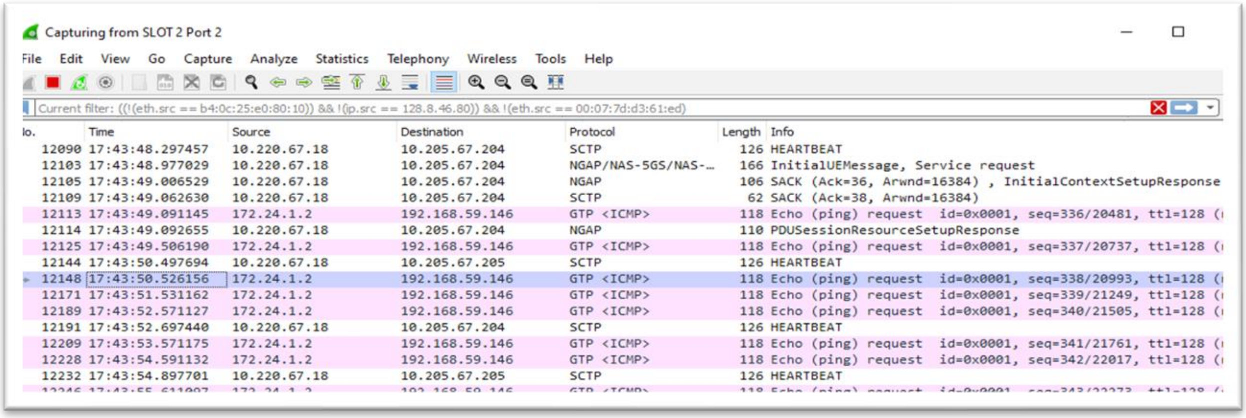


Figure 38: Ping messages captured on the untrusted transport link between the RAN and core

Subsequently, we used TP4, a laptop connected to the transport channel, to capture user plane traffic, modify that traffic, and inject it into the transport channel. The data captured was saved to a file (ping_111822.pcap), and we stopped pings from the UE. The modified data was injected using a laptop connected to the transport channel at the RAN-side switch (TP4), as shown in Figure 39Figure 39. To enhance the injection, the captured traffic is looped and replayed as quickly as possible using the loop and topspeed commands. Logs for the injected traffic are captured on the outgoing interface for the inject laptop (Figure 40), on the outgoing core-side R6K interface (Figure 41Figure 41), and in the ping responses displayed at the laptop connected to the UE (Figure 42).

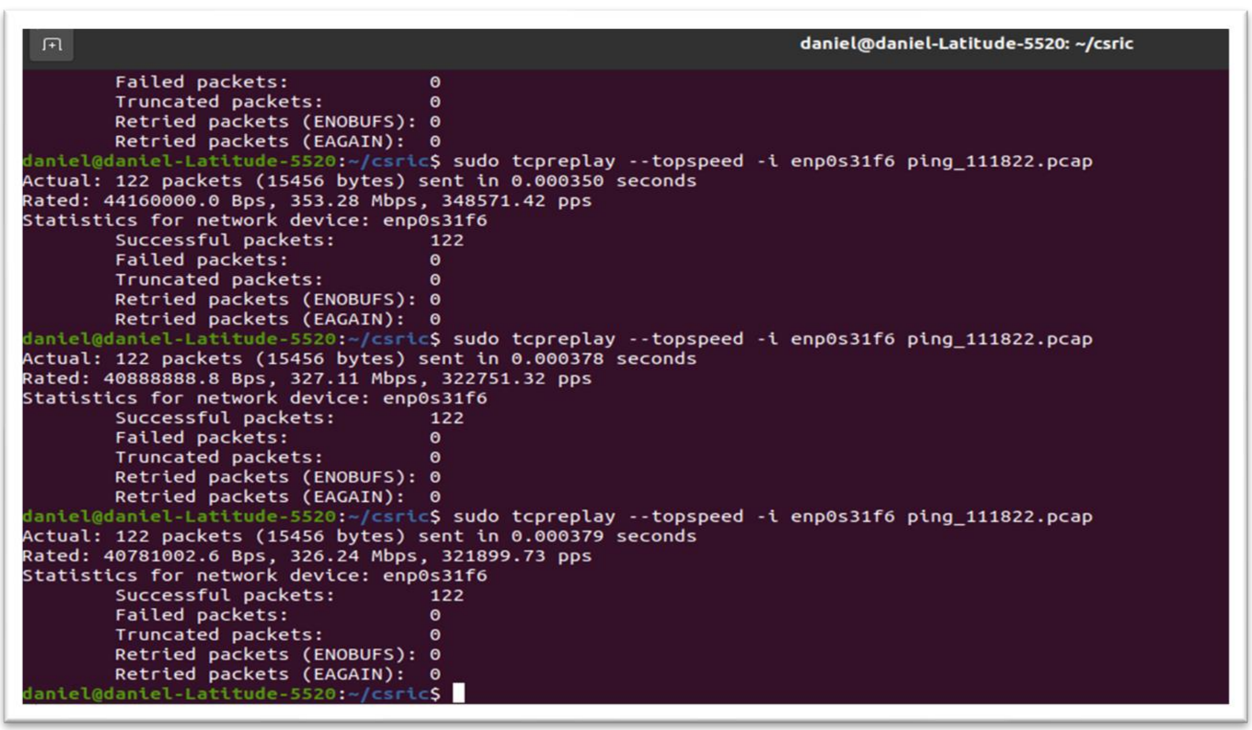


Figure 39: Injecting Unencrypted Captured Traffic – TP4 Laptop

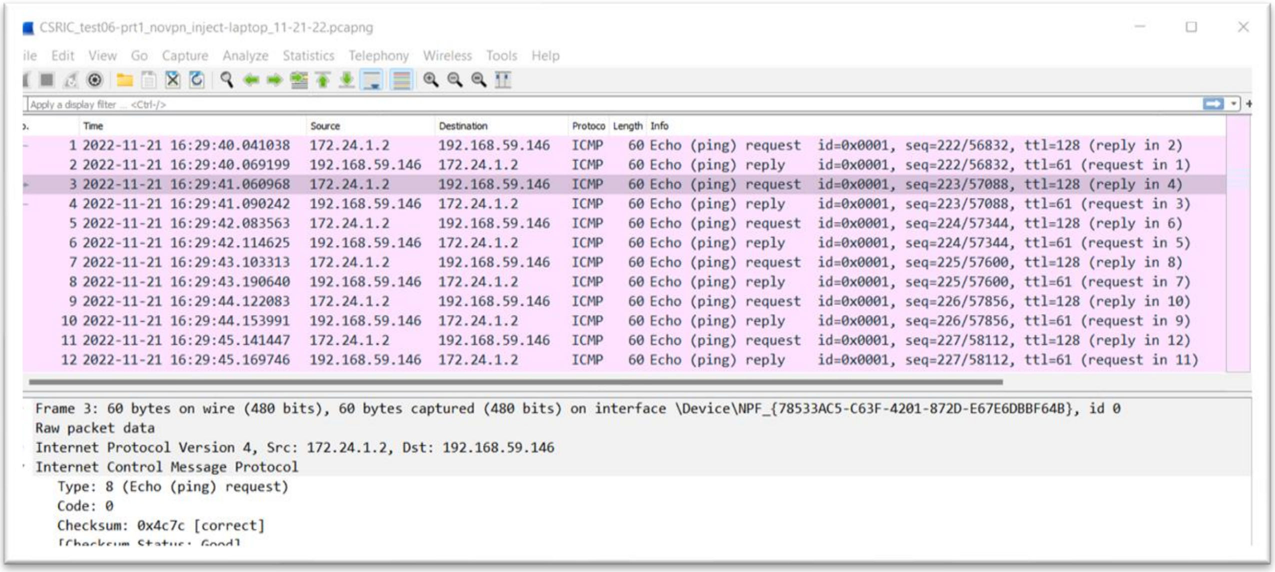


Figure 40: Injected outgoing traffic - Inject Laptop

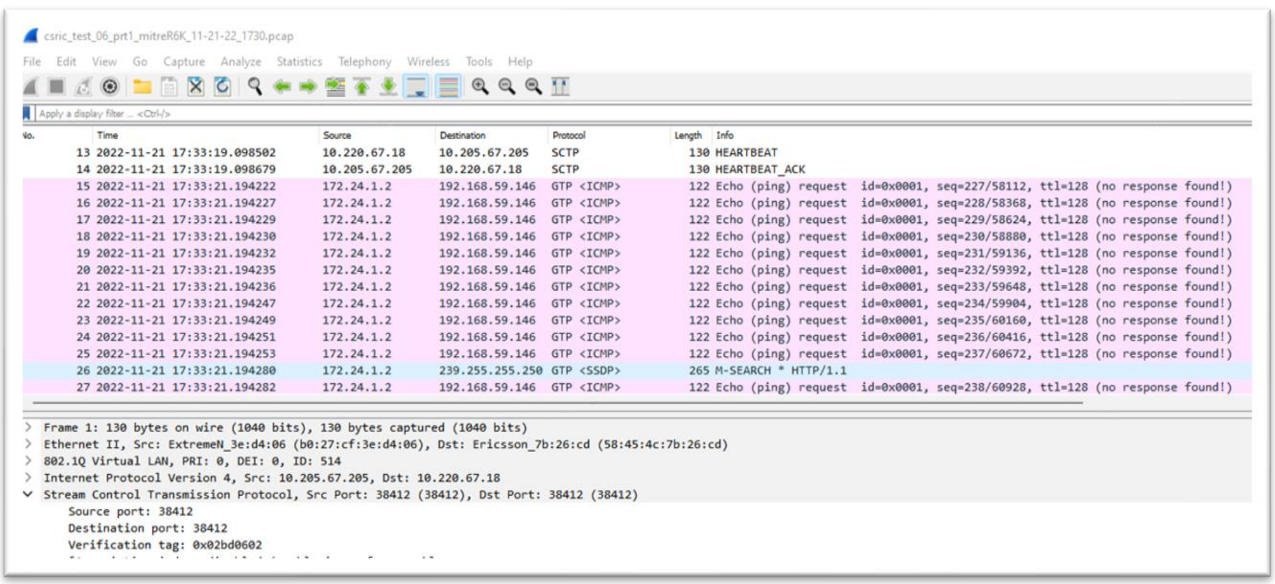


Figure 41: Injected received traffic at core-side R6K

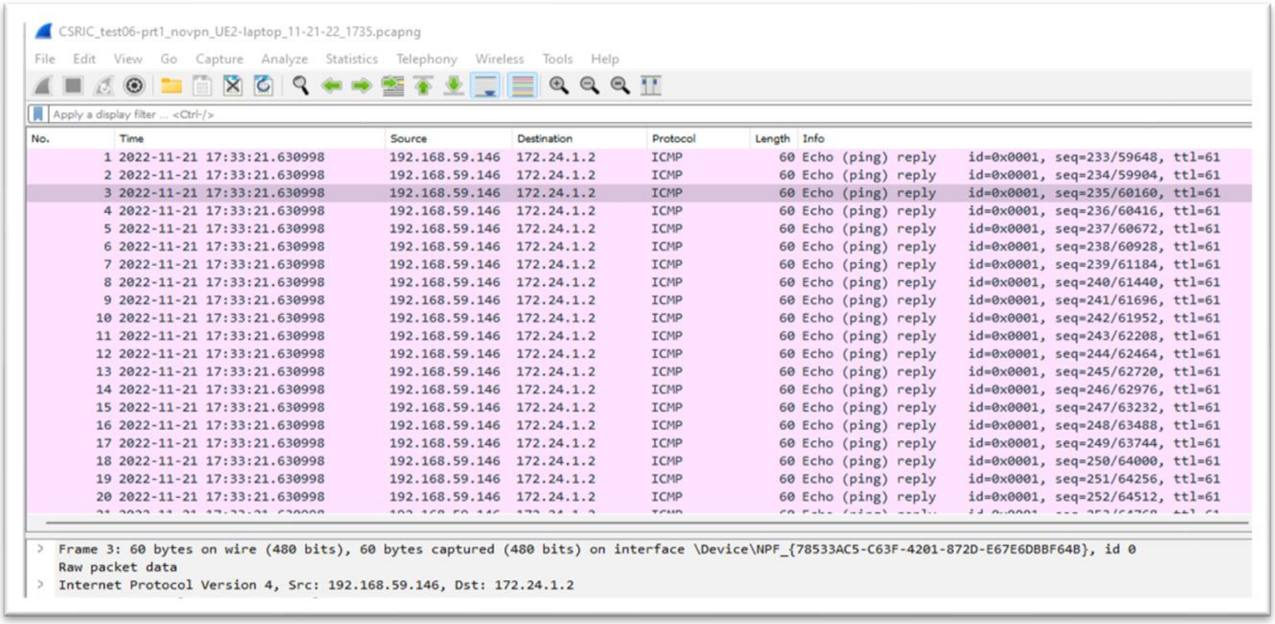


Figure 42: Injected traffic received at UE laptop

Part 2: Traffic Modification with Enabled IPsec Tunnel between BBU and Core-Side R6K

In the second part of the experiment, we turned on IPsec. As above, from the UE laptop, we issued continuous ping messages and captured packets on both the UE and the core-side R6K. Figure 44 shows packet captures of ping traffic outside the tunnel: at the UE and on the egress of the core-side R6K. These messages were not visible on the transport channel, only appearing as ESP packets. The encrypted packets were captured on the transport channel using the laptop connected to TP4. The messages were then modified and injected into the transport channel as shown in Figure 43.

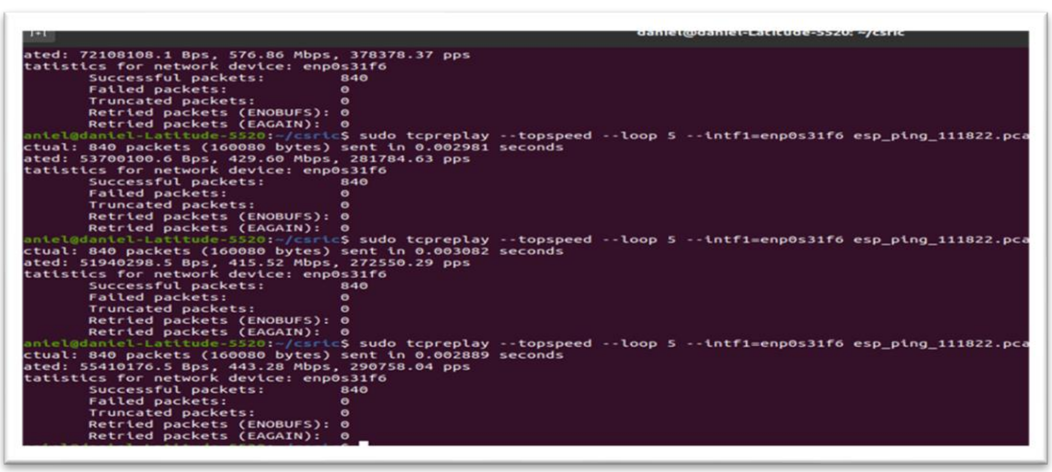


Figure 43: Injecting Encrypted Captured Traffic – TP4 Laptop

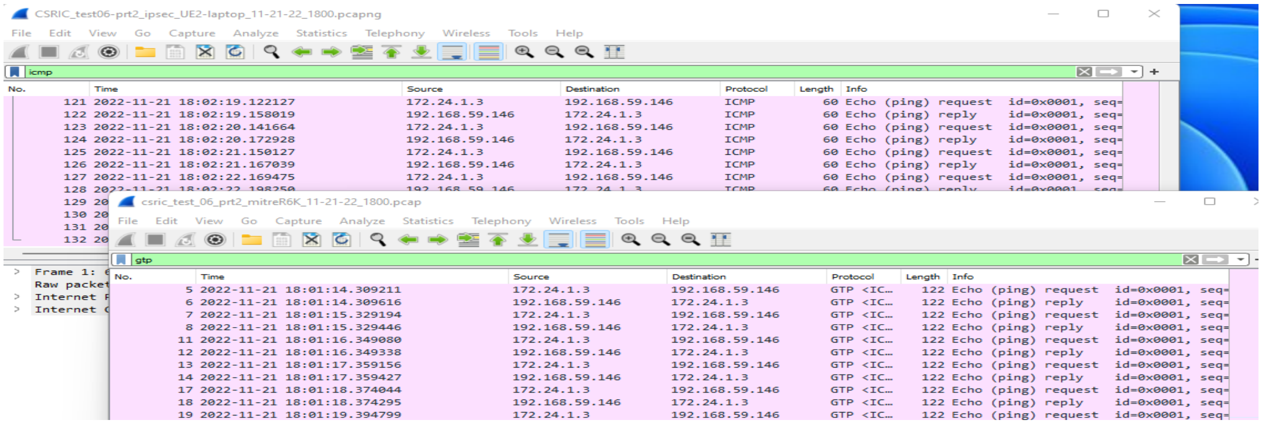


Figure 44: Ping traffic from UE laptop and egress of core-side R6K

The logs for injected packets were captured from the outgoing interface of the inject laptop at TP4 as shown in Figure 45. As observed, only ESP packets were captured. To distinguish between the actual ESP packets flowing through the encrypted tunnel from the modified injected ESP packets, we used the loop and topspeed commands to attempt to inject a high volume of ESP packets (2,490 packets) as quickly as possible.

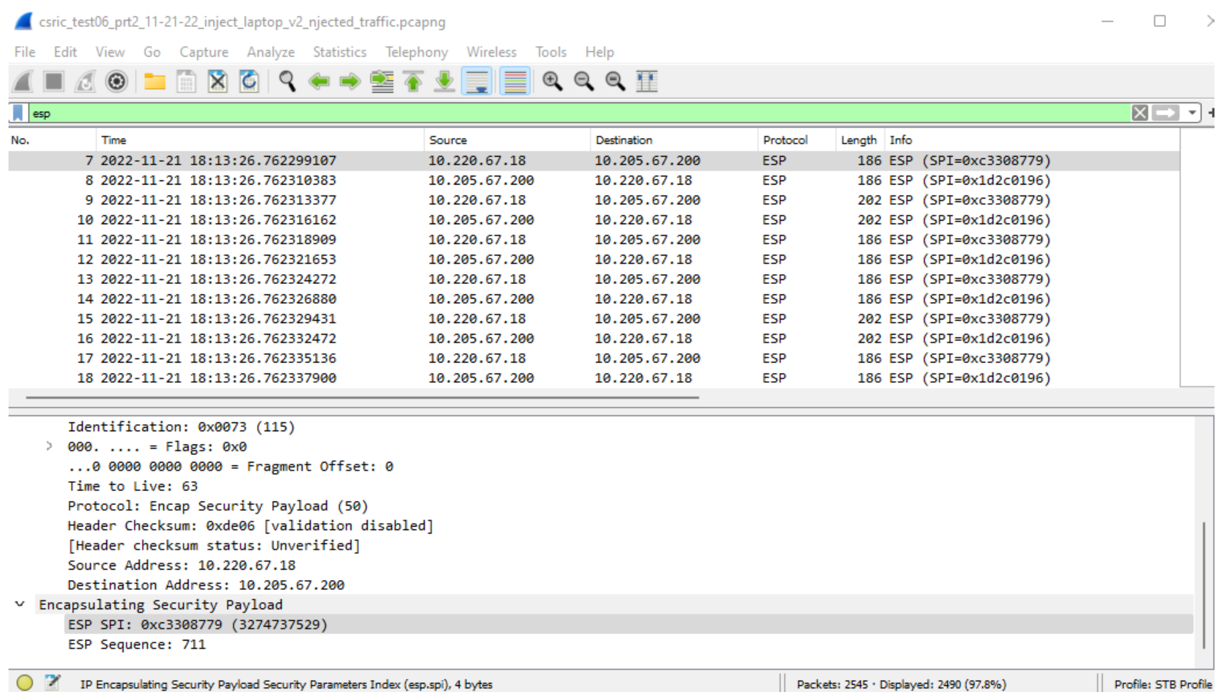


Figure 45: Injected packets from inject laptop at RAN-side switch

As shown in Figure 46, none of the injected packets or their decrypted version makes through to the UE or MITRE R6K during the test. Once the IPsec tunnel is established, traffic to and from the UE to the 5G core is encrypted, and it's not possible to see the contents of the messages. Even though it is possible to capture ESP packets, their contents are encrypted and unreadable, and when packets are modified and injected, they are dropped from either end of the tunnel end-points.

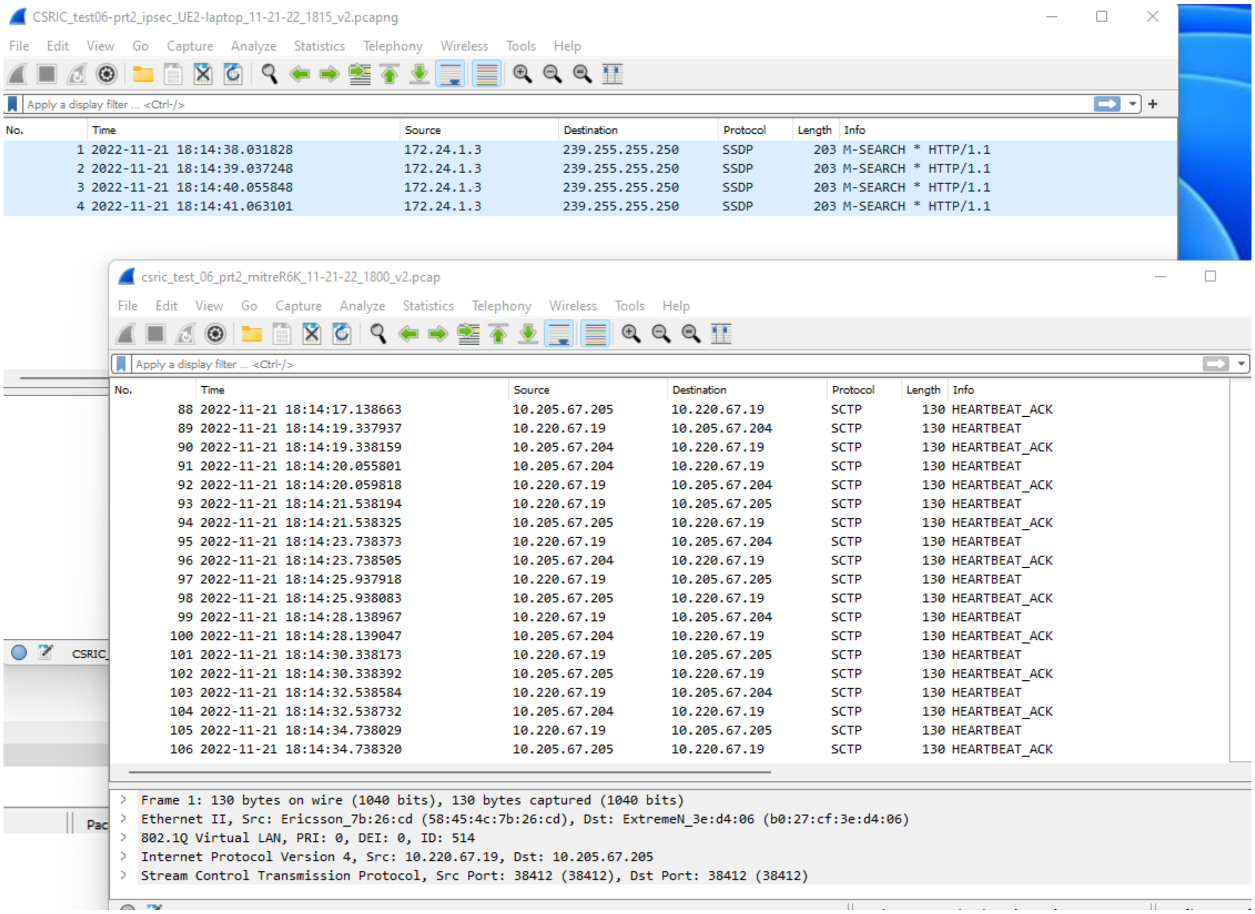


Figure 46: Observed Packets at UE and MITRE R6K after Injecting Packets on Encrypted Transport Channel

Success Criteria:

1. Unable to eavesdrop on UP and CP traffic across the transport link
2. Unable to modify UP and CP traffic across the transport link
3. Unable to inject UP and CP traffic across the transport link

Results

Condition	Status
Unable to eavesdrop on UP and CP traffic across the encrypted transport link	Success
Unable to modify UP and CP traffic across the encrypted transport link	Success
Unable to inject UP and CP traffic across the encrypted transport link	Success
Overall Test	Success

Test Case 7: CSRIC 7 WG 3 – Transport Layer Security for SBA Interfaces

Test Case ID: TC-SA-07
Description:
3GPP TS 33.501 specifies mandatory (e.g. requires vendor support for) transport layer security, but optional for service providers to use.

Given this standards requirement, CSRIC VII recommends the application of TLS for SBA interfaces and tunneling technology for transport when not using the SBA.

This test involves demonstrating that when TLS is used to provide protection for SBA interfaces in the 5G core, data packets cannot be captured, modified, or injected on the SBA interface.

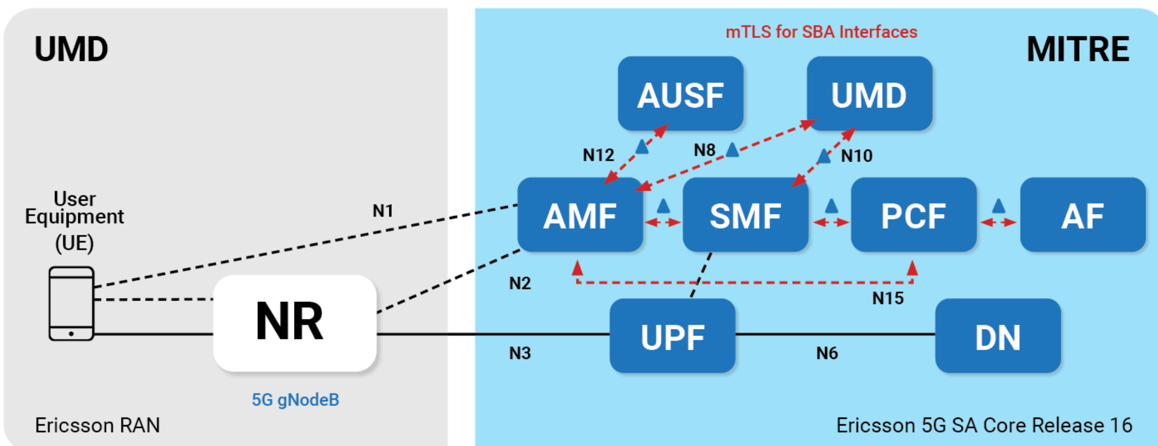


Figure 47: Test Case SA-07 Configuration

Test points used:

Used	Test Point	Description and Use
	TP1	Laptop connected to Sierra Wireless card and/or software-defined radio (SDR); Wireshark captures packets originating at and destined to UE laptop; other tools access SDR controls and data
	TP1-MTP	Laptop connected to Qualcomm MTP; QXDM allows access to low-level data
	TP2	WaveJudge interface
	TP3	Wireshark running on laptop connected to RAN-side R6K router; can be configured to capture packets outside the tunnel (i.e., before IPsec encryption or after IPsec decryption) or inside the tunnel (encrypted packets when IPsec tunnel is enabled)
	TP4	tcpdump running on laptop connected to port of RAN-side Pluribus switch used to capture, modify, and inject packets on the “untrusted link”
	TP5	tcpdump running on port of core-side R6K router inside the IPsec tunnel (encrypted packets when IPsec tunnel is enabled) used to monitor packets on the “untrusted link”
	TP6	tcpdump running on port of core-side R6K router outside the IPsec tunnel (i.e., before IPsec encryption or after IPsec decryption) used to monitor packets at the interface to the DMC
X	TP7	CNOM tool accessing DMC messages, Ericsson transparent TCP proxy tool
	TP8	Applications running on application server in MITRE facility

This test case is comprised of two parts. For Part 1, the 5G core SBA network function (NF) interfaces are not configured with mutual transport layer security (mTLS)—rather, they utilize HTTP2. In Part 2, the 5G core SBA interfaces are configured with mTLS. Table 3 lists the mapped IP addresses used by the various NFs used for the SBA interfaces. Due the nature of the 5G core setup, some NFs (e.g., AMF) communicated on multiple IP addresses.

Table 3: Network Function IP Addresses

AMF	NRF	AUSF	UDM	SMF	TCP Proxy
172.17.152.165	192.168.56.143	192.168.56.138	192.168.56.137	192.168.56.129	172.17.208.251
172.17.95.197	192.168.56.143			192.168.56.131	
172.17.27.33					
172.17.152.146					
172.17.13.136					

For modification and insertion of traffic on the SBA interfaces, we used an Ericsson-provided TCP Layer Proxy Tool. The Proxy Tool was inserted between the various NFs and had the ability to intercept messages from producer (NRF) and its client, as shown in

Figure 48.

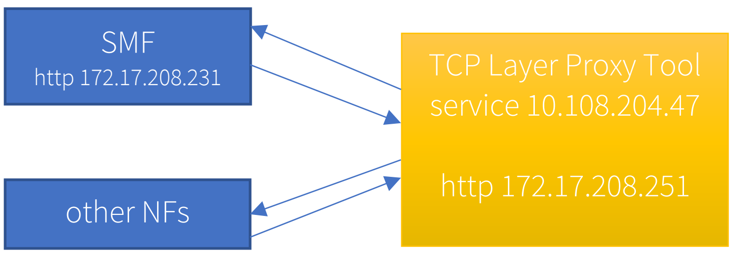


Figure 48: TCP Layer Proxy Setup

As shown in Figure 49, services requests (e.g., the GET operation) are made through the TCP Proxy Tool, where in this figure we have highlighted the AMF (172.17.152.146) requesting services from the TCP Proxy (172.17.208.251) using port 8080.

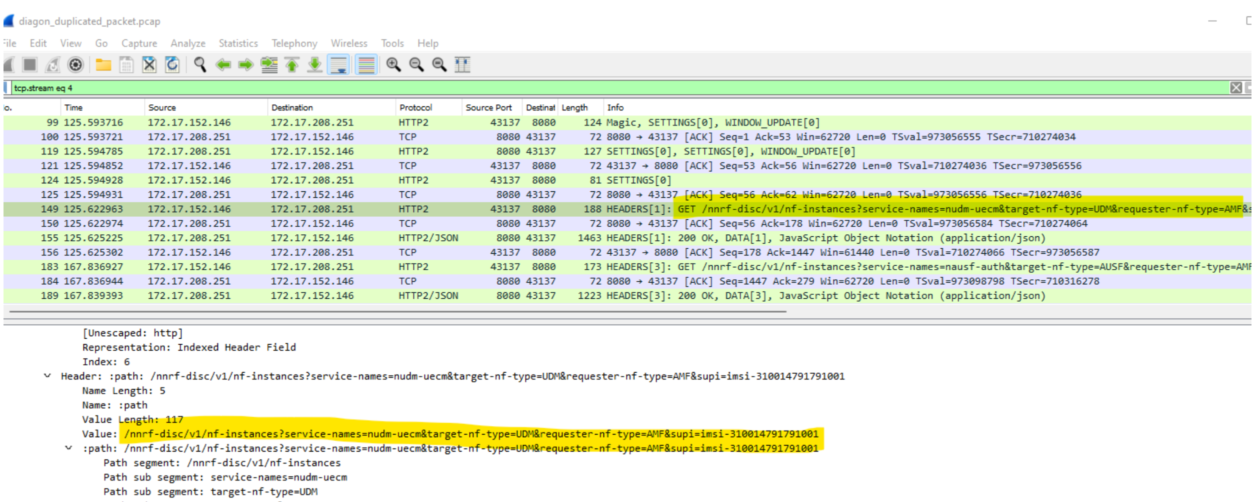


Figure 49: Wireshark Capture of TCP Layer Proxy Communication with Network Functions

Part 1: Unencrypted SBA Interfaces

Tests for Part 1 were performed on February 27, 2023. There are three subparts to this test: (1) eavesdropping on SBA interfaces, (2) packet modification, and (3) packet injection.

```
=== mtrdmcamf01 erv@eric-pc-mm-controller-0 ANCB ~ # gsh get_subscriber -imsi 310014791791001

Subscriber Data
----------------------------------------------------------------------
IMSI                                         : 310014791791001
Mobile Subscriber ISDN No.                   : 7917910001
IMEI                                         : 004403160428990
Radio Access Technology                      : NGRAN
Mobility Management State                    : RM-DEREGISTERED CM-IDLE
Time of Registration in AMF                  :
Time released                                :
RAT restrictions                             : [EUTRA]
Forbidden areas                              : []
Core network type restrictions               : Information not available
RFSP Index in Use                            : Information not available
Service area restriction
    Restriction type                         : NOT_ALLOWED_AREAS
    TACs                                     : []
    Max num. of TAs                          : Information not available
Security Context State                       : Security Context without Secure exchange
5GMM Capability                              : Information not available
In MICO Mode                                 : No
CN Assistance Info RRC INACTIVE to NG-RAN    : false
Paging Proceed Flag                          :
Last Visited Tracking Area [TAI]             :
Tracking Area List                           :
Latest NG-RAN Node List (MCC-MNC-Size-gNBId) : 001-001-24-100002
IMS VoPS                                     : Not supported
SMS over NAS Allowed                         : false
Subscribed S-NSSAIs
    Default S-NSSAIs                         : 1-1
    Non Default S-NSSAIs                     :
Registered S-NSSAIs                          :
5G-GUTI
    PLMN Id                                  : 310-014
    AMF Region Id                            : 255
    AMF Set Id                               : 2
    AMF Pointer                              : 13
    5G-TMSI                                  : 3758407710 (#E004C01E)
```

Figure 50: SA-07 Subscriber details from MITRE 5G Core

Figure 51 shows both the UE trace files and the combined ITC trace files including similar traffic flows with NAS messages and SBA interface messages. In addition, the ITC trace file shows TCP handshakes on the SBA interface: HTTP2 SETTINGs and DATA frame messages.

From these captures, we can clearly read the messages on the SBA interface. Specifically, as shown in Figure 52 and Figure 53, we can eavesdrop on the TCP handshakes and HTTP2 frames messages through, during, and after the UE initial registration process. Looking deeply into HTTP2 HEADER frame messages we see that at Packet 448, the AMF requests AUSF services from NRF through an HTTP2 HEADERS GET frame. These messages expose multiple AMF IP addresses (172.17.152.146, 172.17.95.197, 172.17.27.33, and 172.17.152.165) through the establishment of successful TCP handshakes between other NFs such as NRF (192.168.56.143), SMF (192.168.56.129), AUSF (192.168.56.138), and UDM (192.168.56.137), as well as UE details such as the SUPI (IMSI: 310014790791001), all visible in Figure 53.

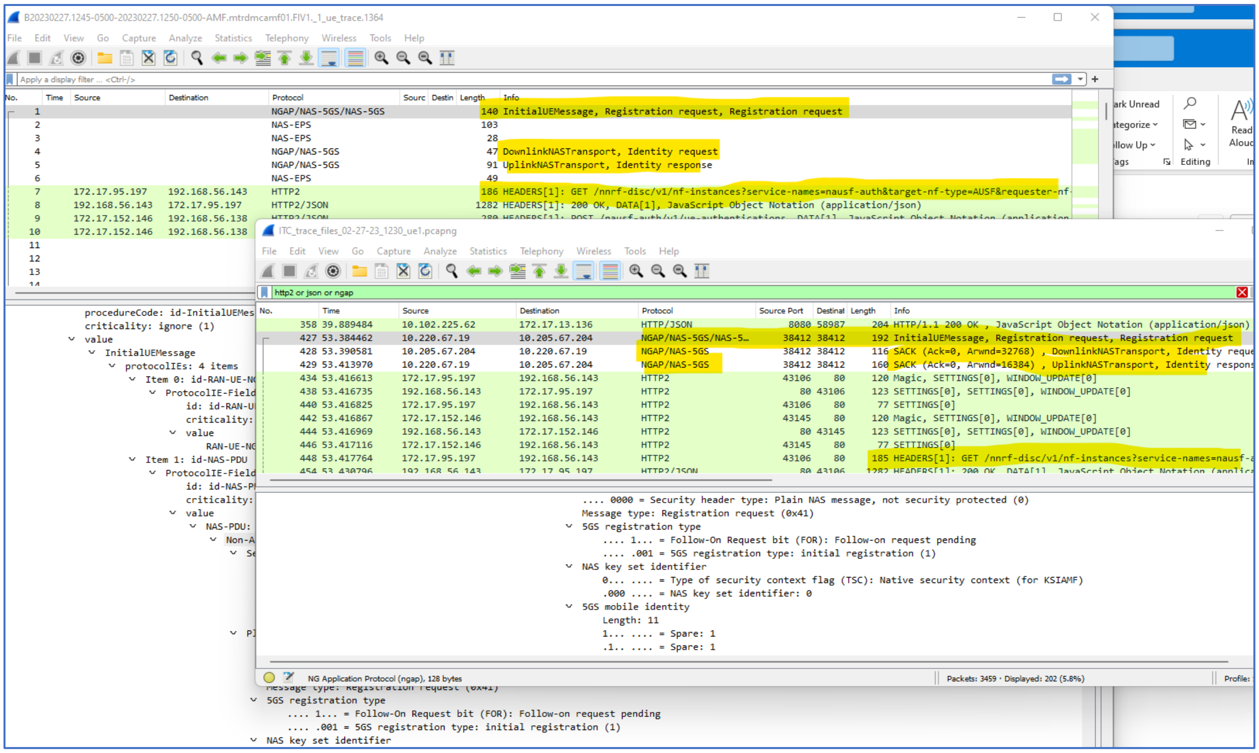


Figure 51: Test Case SA-07 traffic flows for combined ITC and UE trace files

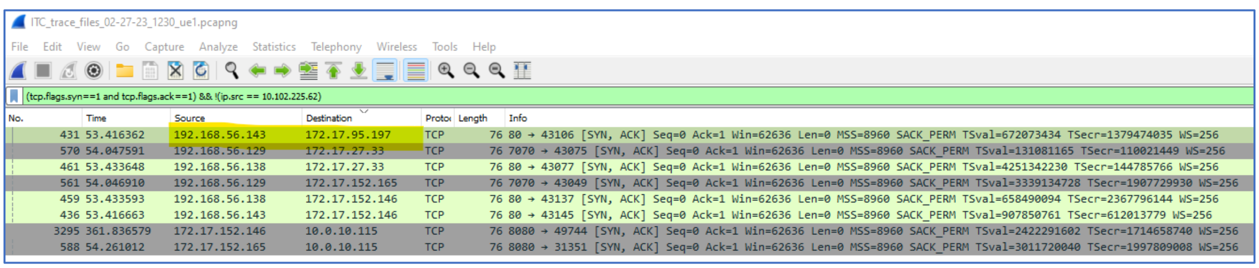


Figure 52: Test Case SA-07 with no mTLS – AMF and other SBA NFs TCP Handshake, Source combined ITC trace file

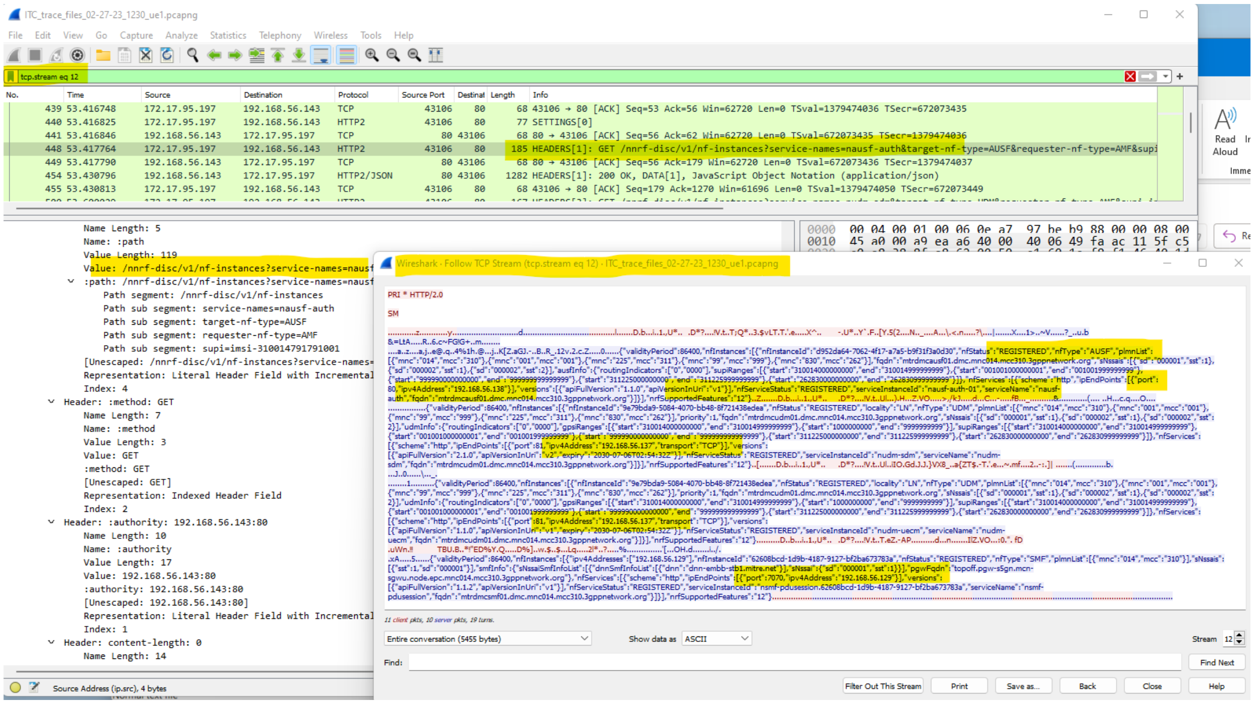


Figure 53: TCP stream showing SBA NFs' communication messages

Part 1.2: Modifying Traffic on the SBA Interfaces

For this part, we used the TCP Layer Proxy Tool to modify a message from one NF and transmit the modified packet to the client NF.

In Figure 54, the AMF requests SMF service through a GET frame via the TCP Proxy Tool (packet 229), and the tool relays this request to NRF (packet 231). Shown as an inset in the figure is the NRF response to the Proxy Tool containing the SMF IP address of 192.168.56.129 (packet 233).

However, as can be seen in Figure 55, rather than passing the message back to the AMF as-is, the Proxy Tool intercepts the message, modifies the SMF IP address to 192.168.56.131, and posts this modified HEADER frame to the AMF (packet 235). That message is received successfully by the AMF without generating an error.

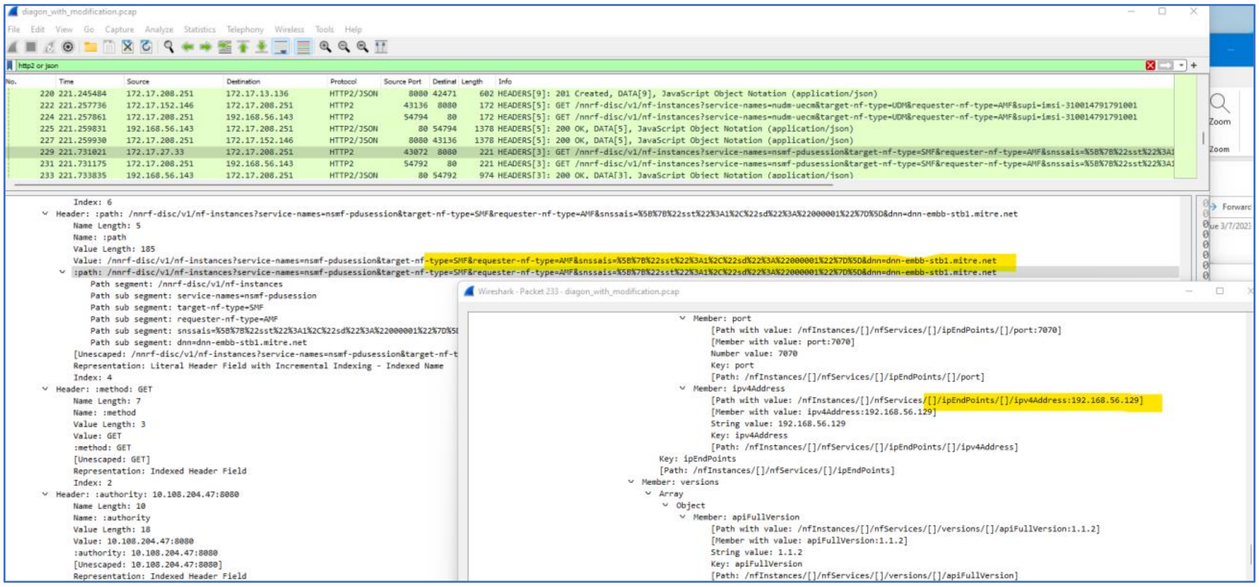


Figure 54: AMF request for SMF services via TCP Proxy Tool

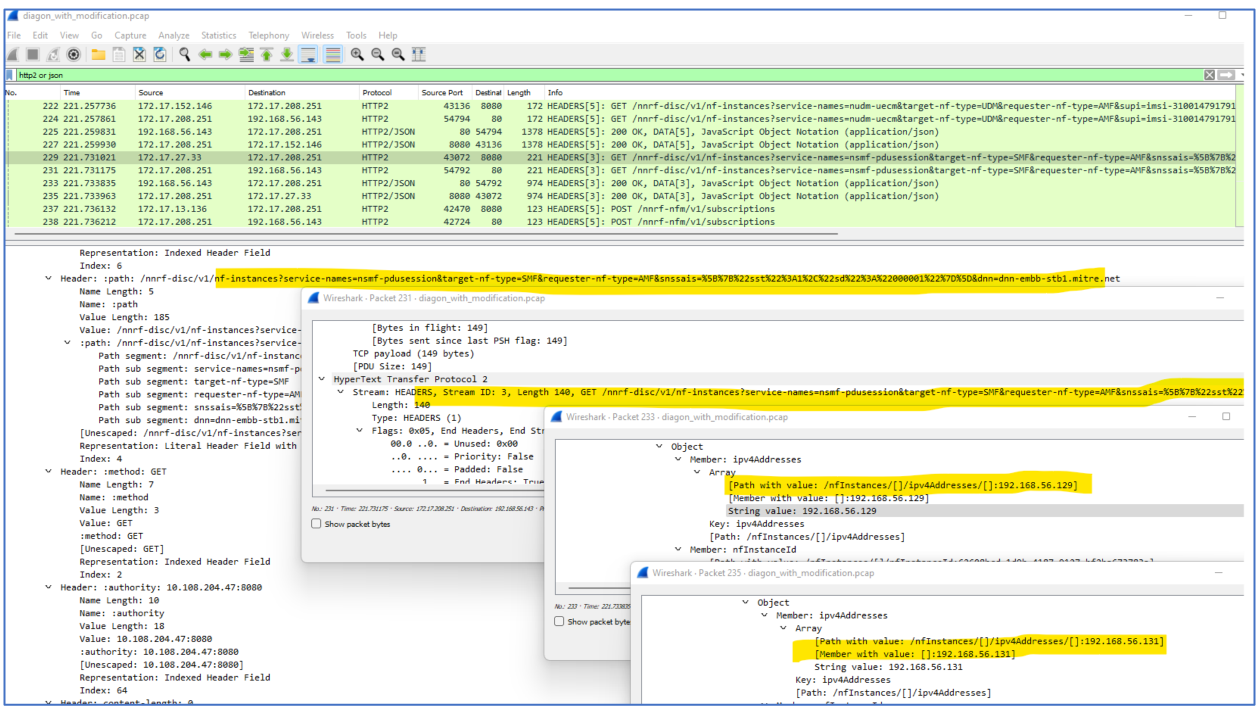


Figure 55: Proxy Tool modification of SMF IP address in service request response

Part 1.3: Injection of Traffic on the SBA Interface

Similar to Part 1.2 above, in Part 1.3 of the experiment, we again used the TCP Layer Proxy Tool, this time injecting a new packet into the SBA NFs' interface data stream. We attempted to insert additional packets into the data stream as additional TCP/HTTP2 messages.

Figure 56 shows the AMF (172.17.152.146) requesting SMF services from the NRF (192.168.56.143) by way of the TCP Proxy Tool (172.17.208.251), as seen in packets 219 and 220. The NRF replies to the Proxy Tool with the IP address for the SMF (packet 221), which is then passed on to the AMF (packet 223). However, in this instance, we saved the message relayed to the AMF, and inserted that duplicated packet into the SBA interface (packet 225). This message is successfully conveyed to the AMF. Because it is a duplicate packet, the AMF recognizes the extra packet and issues a GOAWAY frame message, telling the Proxy Tool and NRF to initiate a graceful shutdown of the HTTP2 connection.

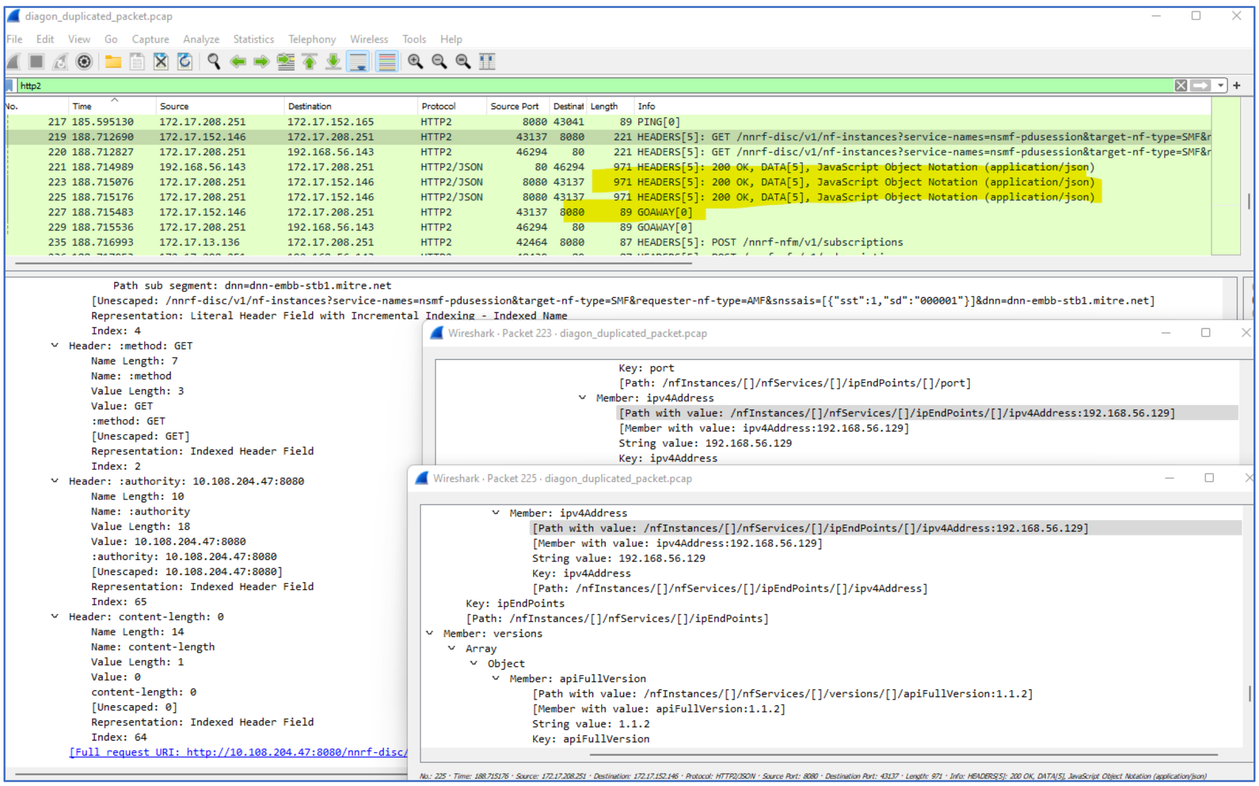


Figure 56: Inserting duplicate message on the SBA interface

Part 2: mTLS on the SBA Interface

For the second part of the test, we repeated the same basic tests as Part 1 of this test after configuring mTLS on the 5G core SBA interface. With mTLS, each network function mutually authenticates with the others to form encrypted connections among them. These latter tests were conducted on June 1, 2023, and also used the TCP Layer Proxy Tool.

Part 2.1: Inability to Eavesdrop on the SBA Interfaces

For this experiment, we focused on the traffic between the SMF (192.168.56.129), NRF (192.168.56.143), and AMF (172.17.27.33). As shown in Figure 57, two SBA interfaces at IPs 172.17.27.33 (AMF) and 192.168.56.129 (SMF) perform TLS handshake, including the Client Hello, Server Hello, and Key Exchanges and Verifications (packets 14664-14673). Thereafter, upon successful key exchange, the mTLS tunnel is established between the two network functions. All traffic between them is subsequently encrypted, and we can no longer see or tell the underlying messages, as seen in Figure 58 in which the Application Data is shown as encrypted and undecipherable by Wireshark.

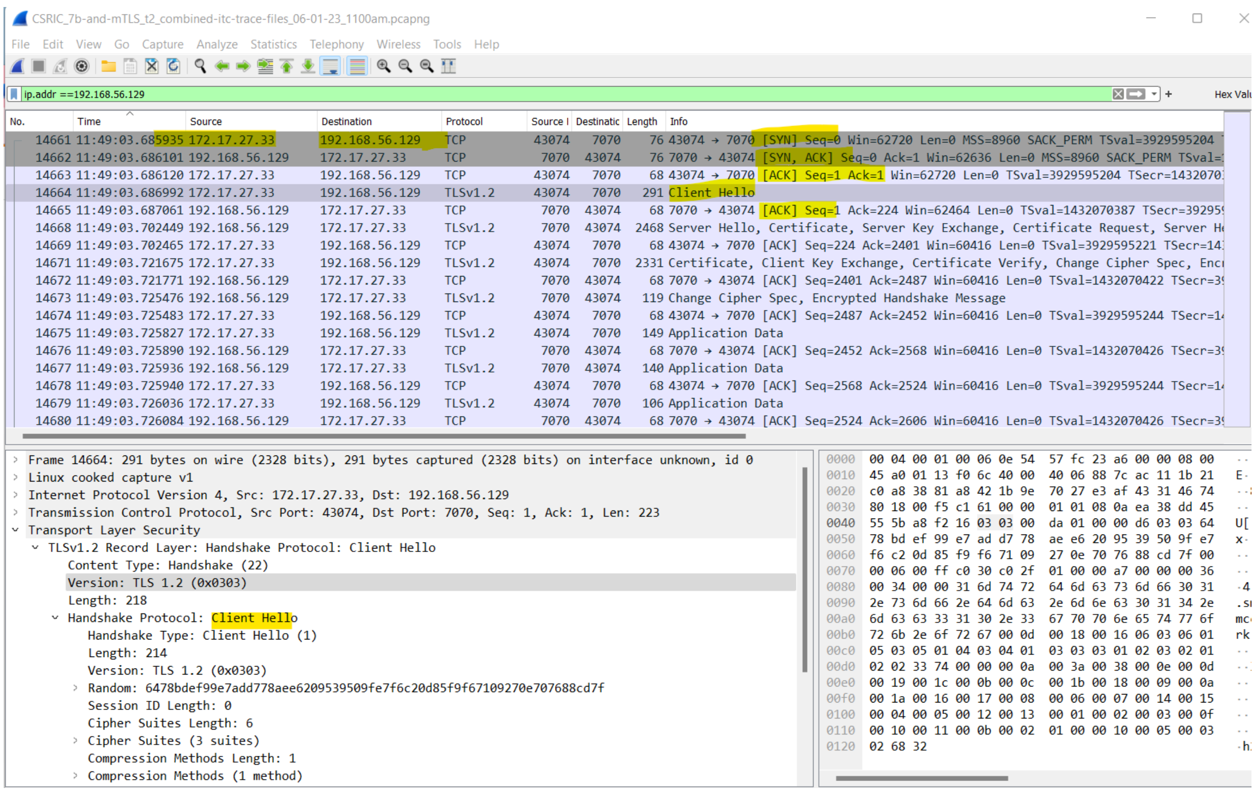


Figure 57: ITC trace file showing mTLS handshake

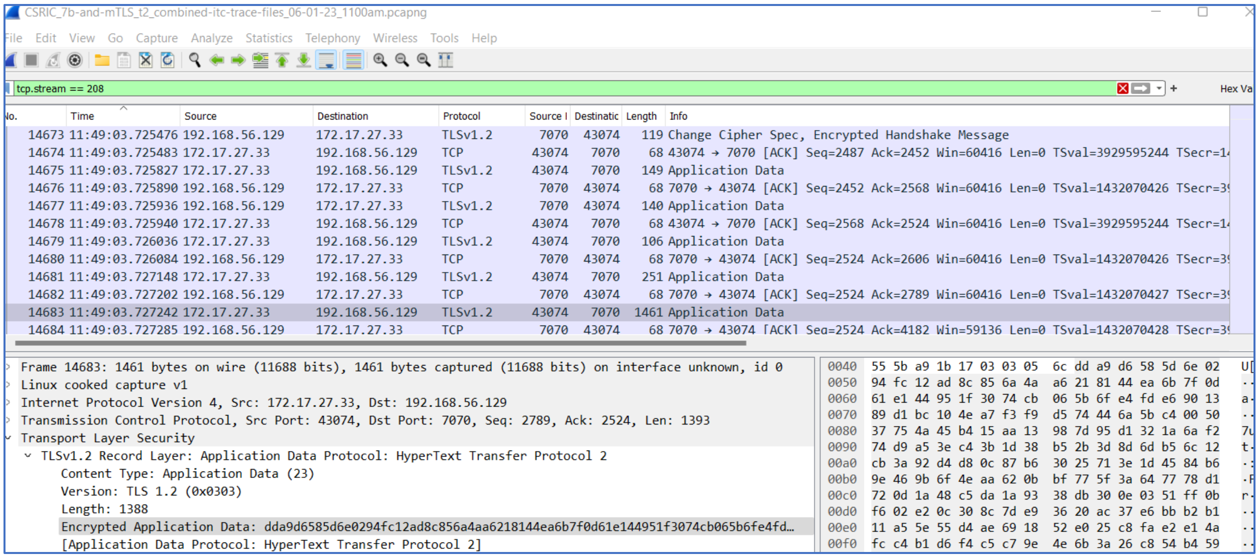


Figure 58: Encrypted SBI traffic with mTLS – source ITC trace file, MITRE 5G Core

Part 2.2: Modifying Traffic on the SBA Interface

In this part, as in Parts 1.2 and 1.3, we used the TCP Proxy Layer Tool to attempt to modify the traffic stream from the NRF (IP address 192.168.56.143) and transmit the traffic to the SMF (IP address 172.17.208.231). The tool is transparent to the SBA interfaces. Also, because traffic between the SBA NFs is encrypted, and the TCP Proxy Tool actions are transparent, the ITC trace files do not show any interactions for the TCP Proxy Tool service or HTTP pod IPs as illustrated in Figure 59, where no messages appear when filtering on the relevant IP addresses. The logs from these IPs are only visible from the logs taken by the Proxy Tool.

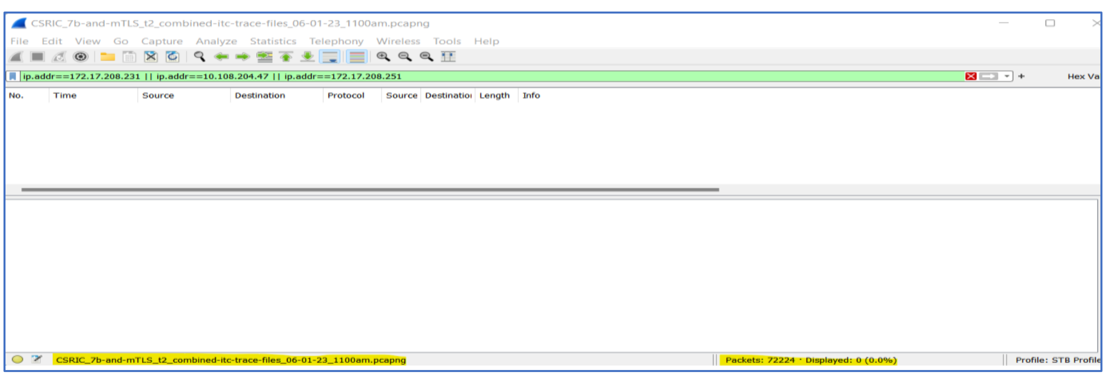


Figure 59: TCP Proxy Tool actions are transparent, source ITC trace file

Because data flowing through the TCP Proxy Tool is encrypted, the tool cannot identify what type of message any given packet corresponds to. Consequently, for data modification, the TCP Proxy Tool randomly selected packets to modify. The modification changed the last byte of data to 0x00 for the selected packet. An example output of the Proxy Tool is shown in Figure 60. The figure also shows how the receiving node disconnects the TLS/TCP traffic stream when the modified packet is received. Subsequently, a new TCP client traffic connection is initiated, starting a new TLS stream in order to complete the failed operation. Figure 61, Figure 62, and Figure 63 show additional cases of the TCP Proxy Tool modifying encrypted packets. Every time the tool modifies the data, a TCP reset (RST) is sent, closing the connection between the sender and the recipient device, and informing the sender to create another connection and resend the traffic.

```
root@diagon-d4df7fdb-mctv9:/app# python diagon_with_modification2.py
('172.17.208.231', 64802) has connected
Modifying data ('172.17.208.231', 64802)!
Original data: b'\x17\x03\x03\x00\x1a\x95\xee\xea\x81\x82\xf1\x06(\x11\xc9N\x07e\x9b\xb5W\xca\xf5\x16\x01\x95\xf9\xba\xf3(\xf8'
Modified data: b'\x17\x03\x03\x00\x1a\x95\xee\xea\x81\x82\xf1\x06(\x11\xc9N\x07e\x9b\xb5W\xca\xf5\x16\x01\x95\xf9\xba\xf3(\x00'
('172.17.208.231', 64802) has disconnected
('172.17.208.231', 65338) has connected
('172.17.208.231', 64582) has connected
('172.17.208.231', 64737) has connected
('172.17.208.231', 64587) has connected
Modifying data ('192.168.56.143', 443)!
Original data: b'\x16\x03\x03\x00z\x02\x00\x00v\x03\x03\x82\x86\xfa\xc2[\xe3\xb3\xef\x08Fh\x1f\xf1\xcfY\x8e!\x18B!-aqY4\x0e\x02\x83\r\xec\xcd] \xfa\x9a./\xc4
\xae\xb79\xb8R\xab\xb4\xd6\xe6\xd4\xb6\xf3\xfb\xbc\x1e\xb8\xcc\xe1\x0c\xe2TV:rF\xfe\x9b\x13\x02\x00\x00.\x003\x00$\x00\x1d\x00 \x1aVhvN\xb3\x19\xa8\x0b\x01
\xa0\xe5\x99\x8e$*\x18\xc4\xbd\xf8\xc4\xf9\xfc\xee\xfe\x8a\xfb\x98]<\x9db\x00+\x00\x02\x03\x04\x14\x03\x03\x00\x01\x01\x17\x03\x03\x05\x94\x08r\x1b9\nl\x95
```

Figure 60: Traffic modification using TCP Proxy Tool

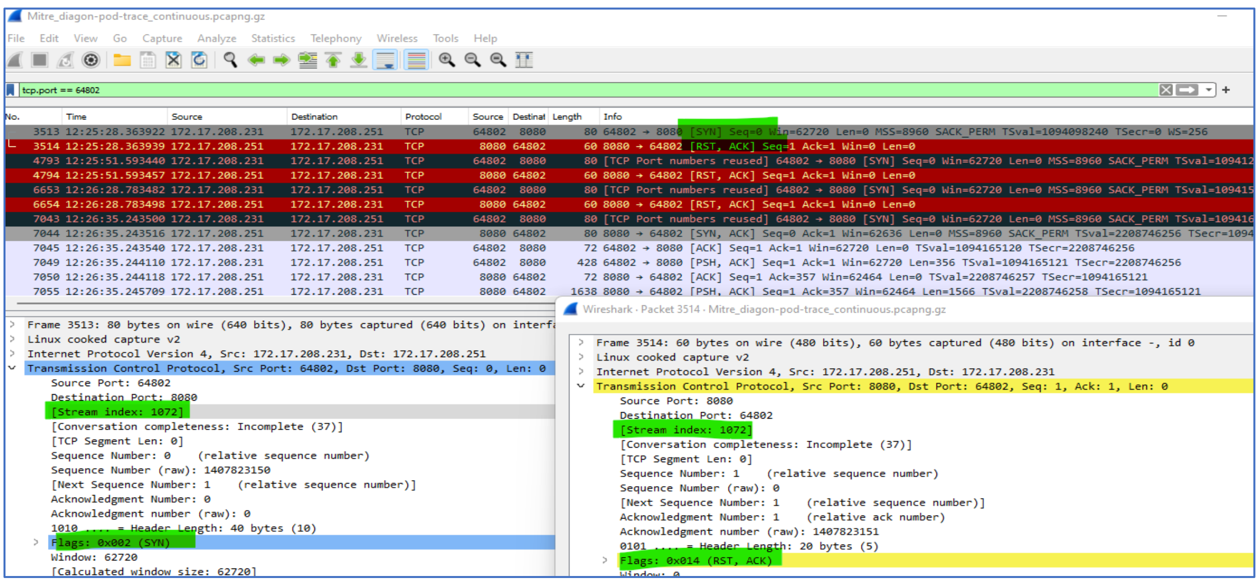


Figure 61: TCP packet reset after data is modified, source TCP Proxy Tool logs

Figure 62: Traffic modification using TCP Proxy Tool

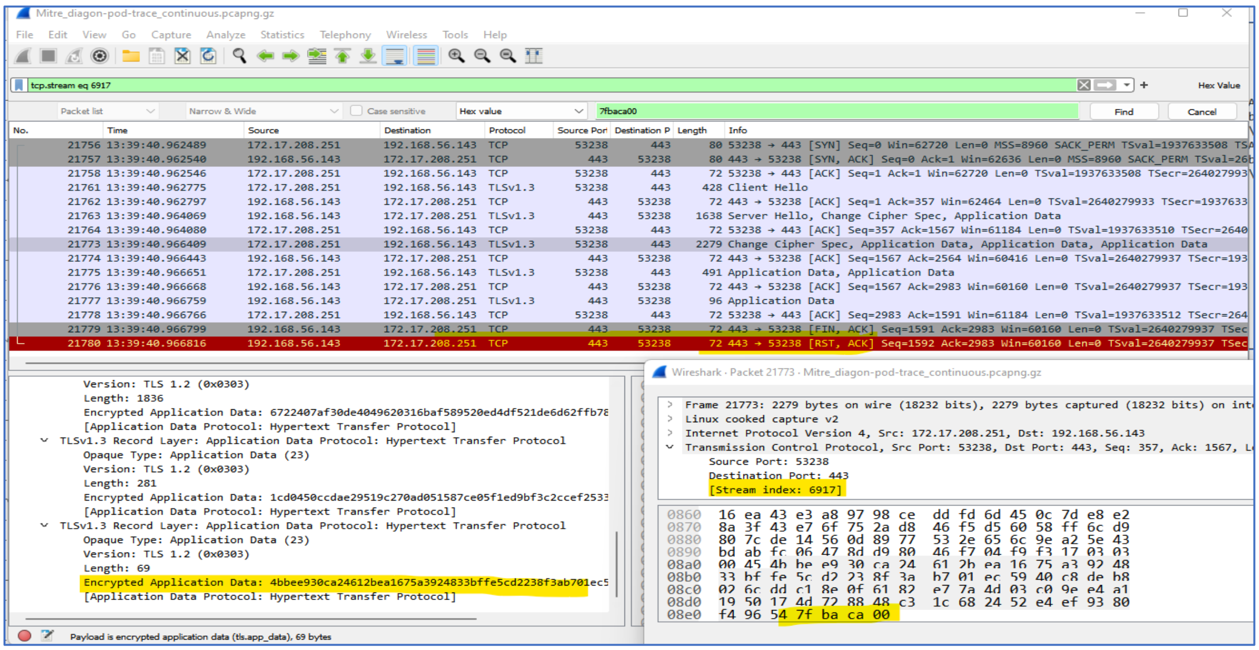


Figure 63: TCP packet data modification, source TCP Proxy Tool logs

Part 2.3: Inserting Packets into SBA Traffic Stream

Similar to Part 2.2 above, in Part 2.3 of the experiment we again used the TCP Layer Proxy Tool to try and inject a new packet into the SBA NFs' interface data stream. In this case, the tool was programmed to duplicate packets randomly. Each duplicate packet is then inserted into the traffic stream and transmitted. As shown in Figure 64 and Figure 65, the TCP Proxy Tool duplicates data on the fly. We see from the logs that the TCP stream is disconnected by issuance of a [FIN,ACK] whenever the remote NF notices a duplicate packet.

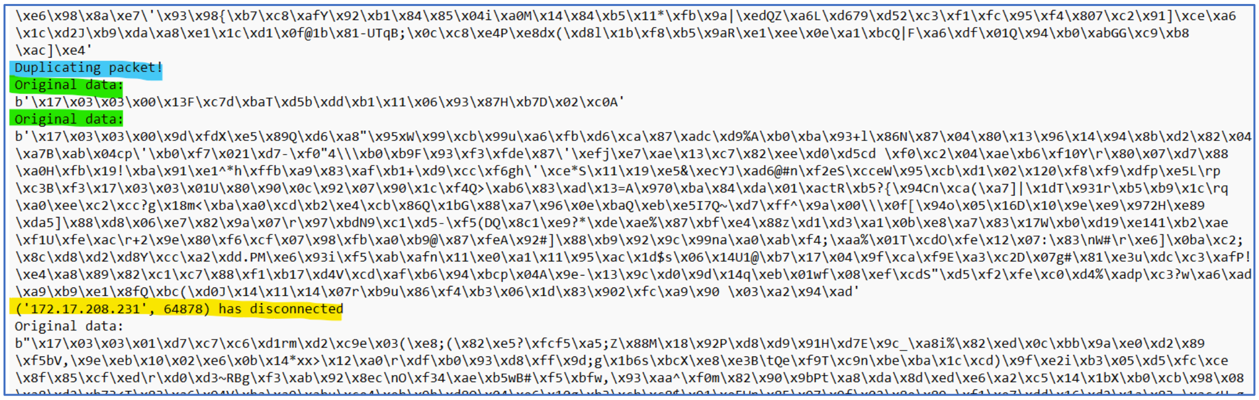


Figure 64: TCP packet data insertion, source TCP Proxy tool logs

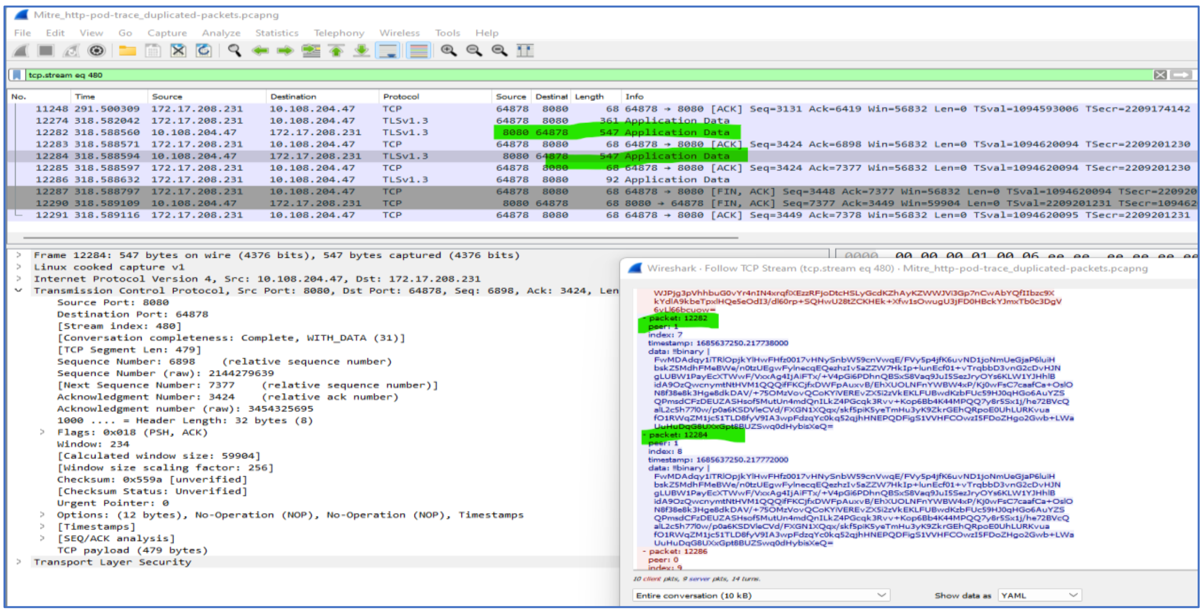


Figure 65: Inserted TCP packets, Wireshark view, source TCP Proxy Tool logs

Success Criteria:

1. Unable to eavesdrop on the SBA interfaces
2. Unable to modify traffic on the SBA interfaces
3. Unable to inject messages on the SBA interfaces

Results

Condition	Status
Able to eavesdrop on the SBA interface when mTLS is not implemented	Success: Able to identify IP addresses for AMF, AUSF, NRF, SMF, UDM; as well as IMSI/SUPI
Able to modify traffic on the SBA interface when mTLS is not implemented	Success: Able to intercept message between the NRF and AMF and modify the SMF IP address without producing error
Able to insert traffic on the SBA interface when mTLS is not implemented	Success: Able to insert duplicate packet on SBA interface, which is received successfully by the AMF, causing it to issue a GOAWAY command
Unable to eavesdrop on the SBA interface when mTLS is implemented	Success: After successful TLS handshake, all subsequent data is encrypted and undecipherable
Unable to modify traffic on the SBA interface when mTLS is implemented	Success: After successful TLS handshake, any attempt to modify encrypted traffic results in an error and reset, terminating the connection
Unable to insert traffic on the SBA interface when mTLS is implemented	Success: Inserting duplicate encrypted packet into the SBA interface causes error and disconnection of session between network functions
Overall Test	Success

Conclusions and Next Steps

This round of testing successfully verified the efficacy of employing security procedures recommended by the CSRIC VII WG3 report, implementing commercial hardware in a commercially-relevant SA configuration.

For each of the seven test cases described here, the tests successfully verified the efficacy of employing security procedures recommended by the CSRIC VII WG3 Report 2 recommendations for securing the 5G standalone network architecture. This verification of the CSRIC recommendations in a commercially-deployed environment is the first of its kind for 5G standalone networks. The test cases focused on confidentiality and integrity at multiple locations in the 5G system, including over-the-air between the UE and the RAN, for NAS signaling, for RRC signaling, over an untrusted backhaul, as well as on the Service-Based Architecture interface.

The first test case demonstrated that the implementation of NEA2 encryption on NAS messages enables user identity to be safely exchanged. With no encryption, as observed when setting the system to use the NULL NEA0 algorithm, messages containing user identities were exchanged between the UE and AMF in a way that message details were visible. However, when the NEA2 encryption algorithm was specified, all NAS messages were encrypted and undecipherable by an observer who does not have the correct encryption key. In addition, only non-user information was observable prior to NAS encryption, and user identity was transmitted via the SUCI.

The second test case considered confidentiality protection for RRC traffic. To test RRC confidentiality, Test Case 2 used an RF network monitoring tool to capture the messages transmitted over the air. First, this test demonstrated the visibility of identity-related data when no encryption (NULL scheme) was used for RRC messages. The captured data showed that the contents of RRC messages were fully decipherable by the RF monitoring tool. Second, the test demonstrated the concealment of the data when RRC encryption was enabled. In that test, the RF monitoring tool indicated the contents of the encrypted messages as “Extra bytes at end of RRC message,” implying that there was additional data present in the packets, but the tool was unable to make sense of it.

The fifth test case addressed the CSRIC VII recommendation that devices and networks in the U.S. use IMSI privacy (SUCI) and do not use the NULL scheme, which could expose the IMSI/SUPI to an unauthorized entity. The test run on the 5GSTB demonstrated the use of the SUCI by the UE in the registration process and resulted in a successful registration.

The next test case reported here replicated tests performed previously for the NSA architecture, demonstrating that the implementation of an IPsec tunnel over an untrusted backhaul link prevents eavesdropping on both user plane and control plane traffic, as well as preventing modification and injection of false traffic designed to appear as originating from or destined to a valid UE. As with the NSA tests, use of the IPsec tunnel resulted in all traffic on the untrusted link appearing as encrypted ESP packets with no ability to read the contents. In addition, when attempting to modify and inject traffic into the transport link, the IPsec tunnel prevented all of the injected packets, or decrypted versions of them, from making it out of the tunnel to either the UE or the core-side router.

The final tests performed for this effort addressed security on the SBA interface, illustrating the benefits of mTLS among the multiple network functions. As such, there were two main parts to the test case: highlighting vulnerabilities without encryption; and demonstrating the protection provided by encrypting traffic on the SBA interface using mTLS. For the first part, it was shown that, without encryption, we were able to identify IP addresses for several network functions (AMF, AUSF, NRF, SMF, and UDM) as well as extracting user identity through the IMSI/SUPI. In addition, the tests demonstrated the ability to intercept messages between the NRF and AMF and modify the SMF IP address without producing an error when encryption was not used. It was also possible to insert duplicate packets on the SBA interface, which were received successfully by the AMF and resulted in a GOAWAY command from the AMF. The second part, with mTLS enabled, encrypted traffic among the network functions. After being able to observe a successful TLS handshake between two network functions, all subsequently exchanged data were encrypted and undecipherable. Furthermore, attempts to modify and inject traffic on the SBA resulted in errors and tearing down the connection between the network functions.

The seven test cases summarized above validate a subset of the CSRIC VII WG3 recommendations. Validation of additional recommendations from WG3 Report 2 are anticipated when the available test tool capabilities are sufficient to run the appropriate tests and capture the required data. Some examples of required capabilities include the ability to alter a message after the integrity check is applied, as well as the ability to capture user plane traffic over the air, in order to demonstrate the efficacy of applying user plane integrity and of access stratum user plane confidentiality.

As new participants and the diversity of test cases grow in tandem, the 5G Security Test Bed will continue contributing to the evolving future of 5G network security, including additional phases of network slicing tests. For future tests, the 5G Security Test Bed is exploring additional aspects of network function security, false base stations, roaming security, and 5G cloud security that arise with use of the Network Exposure Function (NEF), the Application Function (AF), and Multi-access Edge Computing (MEC). The Test Bed is also exploring opportunities to test configurations of Open Radio Access Network (RAN) to verify security recommendations.

For more information, or to participate in the 5G Security Test Bed, please contact Harish Punjabi (hpunjabi@ctia.org; (202) 845 5701), or visit https://5gsecuritytestbed.com/.

Appendix: Acronyms

3GPP	3rd Generation Partnership Project
5G STB	5G Security Test Bed
AMF	Access and Mobility Management Function
AUSF	Authentication Server Function
BBU	Baseband Unit
CNOM	Core Network Operations Manager
CP	Control Plane
CSRIC	Communications Security, Reliability, and Interoperability Council
CSWG	Cybersecurity Working Group
DHS	Department of Homeland Security
DMC	Dual-Mode Core
eMBB	Enhanced Mobile Broadband
eNB	e-Node B
ESP	Encapsulating Security Payload
FCC	Federal Communications Commission
IKEv2	Internet Key Exchange Protocol Version 2
IMEISV	International Mobile Station Equipment Identity Software Version
IMSI	International Mobile Subscriber Identity
IPsec	Internet Protocol Security
ITC	Integrated Traffic Capture
ITU	International Telecommunications Union

MME	Mobility Management Entity
mTLS	Mutual Transport Layer Security
MTP	Mobile Test Platform
NAS	Non-Access Stratum
NG-RAN	Next-Generation Radio Access Network
NIST	National Institute of Standards and Technology
NR	New Radio
NRF	Network Repository Function
NSA	Non-Standalone
PDCP	Packet Data Convergence Protocol
RAN	Radio Access Network
RRC	Radio Resource Control
SA	Standalone
SBA	Service-Based Architecture
SBI	Service-Based Interface
SDR	Software-Defined Radio
SEG	Security Gateway
SUCI	Subscription Concealed Identifier
SUPI	Subscription Permanent Identifier
TAC	Technical Advisory Committee
TC	Test Case
TCP	Transmission Control Protocol
TLS	Transport Layer Security

TP	Test Point
TS	Technical Standards
UE	User Equipment
UP	User Plane
UPF	User Plane Function
VPN	Virtual Private Network
WG	Working Group