A Systematic Study for Understanding the Security Risks in 5G Core Network

Unnati Dixit, Shwetha Vittal, Antony Franklin A
Department of Computer Science and Engineering, Indian Institute of Technology Hyderabad
Email: {cs21mtech12012, cs19resch01001, antony.franklin}@iith.ac.in

2024 16th International Conference on COMmunication Systems & NETworkS (COMSNETS) | 979-8-3503-8311-9/24/$31.00 ©2024 IEEE | DOI: 10.1109/COMSNETS59351.2024.10427440

***Abstract*—The widespread adoption of 5G networks has revolutionized communication, providing great connectivity and services across various verticals. Specifically, on the 5G Core (5GC) this has been made possible by leveraging network slicing and Network Function Virtualization (NFV) where multiple Network Functions (NFs) are involved to serve different user requests on the control plane and a variety of service traffic on the data plane. However, as 5G networks embrace zero trust and multi-tenant architectures, this rapid technological advancement has also exposed the 5GC to various security risks such as Distributed Denial of Service (DDoS), Man in The Middle (MITM), cross-slice disruptions, and side-channel attacks. In this regard, this paper presents a comprehensive approach to enhance the security of 5GC through a systematic study of dependency graphs, Betweenness Centrality (BC), and attack graphs. Potential attack scenarios are carefully analyzed by modeling the 3GPP-defined 5GC, performing vulnerability assessments, building attack graphs, analyzing them, and presenting the key inferences from the study.**

I. Introduction

The 5G system comprises interconnected elements, including the Radio Access Network (RAN) and 5G Core (5GC), delivering end-to-end services to various types of User Equipments (UEs). Specifically, the 5GC involves Network Functions (NFs) like Access and Mobility Management Function (AMF), Authentication Server Function (AuSF), Network Repository Function (NRF), Session Management Function (SMF), Unified Data Management (UDM) on the control plane. The 5GC adopts network slicing for supporting various applications on a common physical infrastructure and containerization for scalability, flexibility, and reliability. However, these advancements increase security risks, demanding proactive approaches to address potential vulnerabilities. Securing 5GC is crucial for resilient and reliable service provision across all sectors.

In the dynamic landscape of 5GC, safeguarding against cyber threats is a paramount concern. In this context, this paper presents a systematic and comprehensive study of security risks in the 3GPP-defined 5GC by leveraging the dependency graphs [1], Betweenness Centrality (BC) [2], and the attack graphs [3]. First, the critical vulnerabilities are identified and prioritized as hot spots, by leveraging dependency graphs and BC in the native 5GC and then the attack graphs are generated on those vulnerabilities using appropriate open-source software like MulVAL [4]. Further, the attack graphs are carefully analyzed, where new attack paths are identified, which otherwise are not possible manually. Finally, the possible prevention and mitigation techniques for these attack paths are proposed. Moreover, the work presents several observations and inferences in every stage of this entire systematic study and experimentation. To summarize, the key contributions of this work are:

- A systematic approach analyzing the security vulnerabilities in 3GPP-defined 5GC using dependency graph, Betweenness Centrality (BC), and attack graph, thereby forming the basis for enhanced security measures.
- Attack graph generation using the MulVAL software to provide a visual representation of potential attack paths and aid in threat assessment.
- Security strategies to prevent or mitigate the identified vulnerabilities.
- Presenting the key insights realized in every stage of the study.

II. Motivation

Within the 5GC, several key security concerns have risen to the forefront.

- **Increased Complexity:** The 5GC architecture is notably more intricate than its predecessors. This complexity arises from the integration of various technologies like slicing with NFV, Multi-access Edge Computing (MEC), and other orchestration and management needs, like scalability, resilience, and High Availability (HA) [5, 6], thereby creating a vast network ecosystem with numerous interacting NFs [7]. Though this complexity offers greater functionality, it also introduces potential vulnerabilities that require thorough evaluation and mitigation.
- **New Attack Surfaces:** The proliferation of services by the support of slicing and MEC, devices, and interfaces by the support of microservices, within the 5GC possibly introduces new potential entry points for malicious actors. These expanded attack surfaces offer adversaries a wider range of avenues to explore potential vulnerabilities, emphasizing the necessity of proactive security measures.
- **Multi-tenant Architecture and Dependency on Third-Party Vendors:** In the end-to-end functioning of the 5G system, multiple tenants could inter-operate to save the cost [8]. Here, the ecosystem often welcomes third-party vendors like Application Functions (AFs) and services in MEC deployments. While these collaborations enhance innovation, they also introduce potential security risks.

Malicious entities could potentially exploit vulnerabilities within third-party components to compromise the security of the 5GC.

In response to these challenges, the primary goal of this work is to construct a comprehensive security study that can aid in building a framework capable of safeguarding the 5GC against a wide array of security threats. By dissecting the intricate interactions of different NFs in the 5GC and assessing their potential consequences, our study aims to provide actionable insights that can guide the formulation of effective security strategies.

III. Related Work

5G system security is on continuous research and improvement both by 3GPP and researchers. In [9], the authors claim that the complex, multi-stage attacks that are on the rise cannot be adequately addressed by the vulnerability scanners and patching techniques now in use. To assist security administrators in better understanding and safeguarding against different assaults, they suggest a novel strategy based on attack graphs and recursive composition algebra. The same authors also classified vulnerabilities into different categories in [10], which is helpful to the Mobile Network Operator (MNO) when designing the mitigation techniques. While these works motivated us to initiate the security study on the 5GC, authors in [7] review the existing security architecture and trust models defined by 3GPP and highlight the security implications of the 5GC SBA while advocating the use of the recent advances in the field of machine learning to detect anomalies. Additionally, in close relation to our work, authors in [11] discover about 119 new potential exploits by representing 113 documented Software Defined Networking (SDN) and NFV attack vectors in the form of concise attack graphs. The authors here also employ Machine Learning (ML) and Constraint Satisfaction Problem (CSP) formulation to predict the possible connections of a new node in the graph to the existing nodes. This work is quite useful for knowing all possible generic and system-level vulnerabilities as 5GC leverages SDN and NFV architectures.

Overall, these works highlight the possible generic threats, which can be very much applicable to the 5GC at the system level and therefore are quite motivating. However, none of these works do a systematic study with practical experiments on the 3GPP-defined 5GC. Hence, we dive deep into studying the 5GC-specific architectural, application, and protocol vulnerabilities. To the best of our knowledge, this work is the first attempt to reveal the possible vulnerabilities in the 5GC by a systematic study and presenting the key inferences at every stage. Fig. 1 depicts the complete flow of three important stages (see Section I) in this process.

IV. Identification of Critical Network Function(s)

This section details the study of various graph-based models to aid in identifying the critical NFs that could potentially pave the way for unforeseen vulnerabilities in the 5GC.

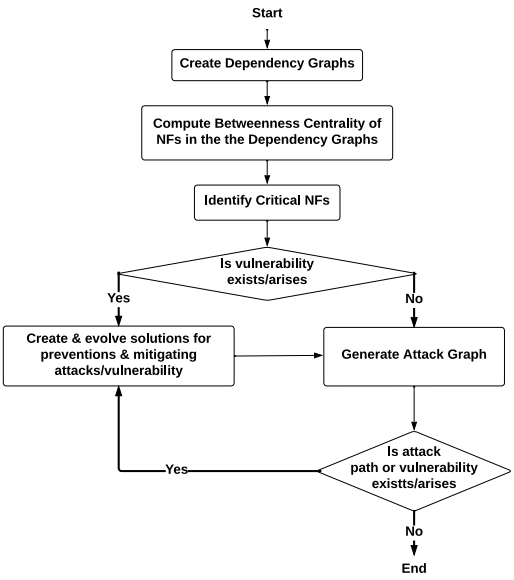


Fig. 1: 5GC security study through various stages.

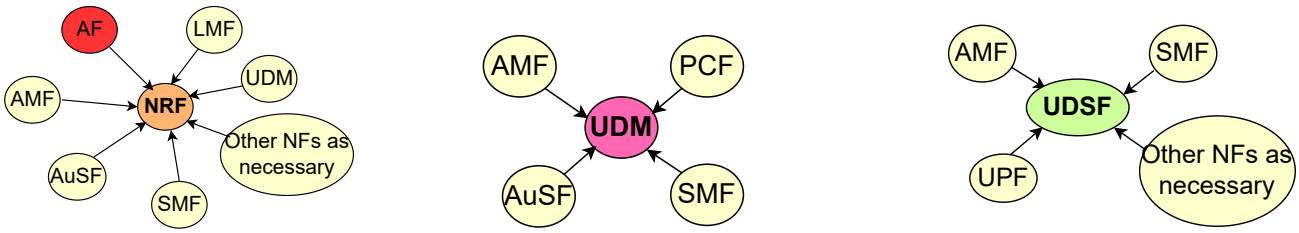
A. Dependency Graph

A dependency graph [1] is a directed graph that shows the inter-dependencies between multiple entities. Fig. 2 depicts the dependency graphs for NRF (in Fig. 2a), UDM (in Fig. 2b), and the global Unstructured Data Storage Function (UDSF) (in Fig. 2c). Here, in Fig. 2a, we can observe that most of the NFs are dependent on NRF for NF service registration and discovery. In Fig. 2b, we observe that AMF, AuSF, and SMF are dependent on UDM for subscription and authentication data [12]. Overall, these dependency graphs depict the relationship between NFs and possibly hint at the potential mutual attack issues like the 5G Authentication and Key Agreement (AKA) problem explained in the work [11]. Therefore, the **First insight** we obtain here is that sketching the dependency graphs is needed as a preliminary step in the study of the security of 5GC to know the mutual relationship among the different NFs involved. However, these graphs do not directly identify the critical NFs nor indicate the possible attack paths. Hence, we further explore Betweenness Centrality (BC).

B. Betweenness Centrality

Betweenness Centrality (BC) is a measure of centrality [2] in a graph based on shortest paths. Given the shortest path between any pair of vertices in a connected graph, BC captures the least number of edges that the path passes through (for unweighted graphs) or the minimum sum of the weights of the edges (for weighted graphs). We rely on computing BC in the 5GC to discover the NFs that have more influence. Therefore, we compute the BC values for individual NFs of the 5GC and then focus on the NFs having the highest BC values.

The computation of BC in the 5GC mainly goes through two steps.



(a) (b) (c)

Fig. 2: Dependency graph of a) NRF b) UDM c) UDSF in the SBA of 5GC.

TABLE I: Betweenness Centrality of the NFs in the connected graphs shown in Fig. 3 and Fig. 2

NF	BC(NF)	Description
NRF	At least 6	NRF is used for NF service registration and discovery. For example, as shown in Fig. 2a AMF can reach AuSF only via NRF, as AMF discovers the registered service of AuSF from NRF.
UDSF	At least 3	UDSF is reached during the start and end of different UE procedures by AMF, SMF, UPF, and other NFs as necessary. Note that, here unlike NRF, which is needed only once in the beginning for the UE to discover an NF for particular service access, UDSF is accessed by all the dependent NFs like AMF, SMF, UPF, LMF, and so on, while handling each procedure of every UE as it is a data store to update and read every UE's context information like its state, NAS security context, number of PDU sessions currently established, etc. Hence, the BC of UDSF is high compared to other NFs.
UDM	4	UDM is reached during UE registration for fetching the UE subscription data by AMF, and authentication vector by AuSF. It is also accessed by SMF and PCF during PDU session establishment, to fetch the PDU session-related subscription and policy data.

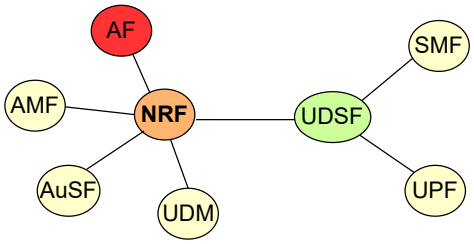


Fig. 3: 5GC for the study of betweenness centrality while handling UE registration and PDU session procedures.

1) **Step 1:** Representation of 5GC in the form of a connected graph G with a set of vertices and edges.
2) **Step 2:** For each vertex v in the graph G, compute the BC value by computing the shortest path between every pair of vertices where v is not the starting or the ending point. The final value shall be the sum of the individual values obtained for each edge.

BC of a vertex v in a connected graph is given by

$$\sum_{s \neq v \neq t} \frac{\sigma_{st}(v)}{\sigma_{st}} \tag{1}$$

where $\sigma_(st)$ is the total number of shortest paths from the vertex s to vertex t and $\sigma_{st}(v)$ is the number of those shortest paths between vertex s to vertex t that pass through v (not where v is an endpoint).

Step 1: In the first step, we represent the 5GC in the form of a connected graph and list the criteria influencing the calculation of BC of an NF.

- Role of the NF whose BC has to be calculated.
- Successful handling of UE procedures involves different NFs. Note that 3GPP [12] defines many UE procedures like registration, PDU session establishment, modification, release, and so on. However, we consider only UE registration and PDU session establishment here as that is sufficient to show the computation of BC.

From the description of dependency graphs in Section IV-A, we realize that every NF involved during the first UE registration discovers the other required NF through NRF and then can communicate with each other. So, NRF is an important NF in the 5GC. Next, for handling the PDU session establishment procedure, an SMF is discovered, and then the appropriate UPF. However, since all the NFs work in the stateless mode, they need to fetch the UE context (on which the PDU session request is supposed to be handled) from the UDSF. So, using all these observations we build a connected graph in Fig. 3 and that completes the first step listed in 1.

Step 2: In the second step, we delve into computing the BC of every NF (represented as a vertex) shown in the connected graph 3. Given the path between two NFs, the BC of an NF represented by a vertex say v is computed by measuring its influence and role in the successful working of two other NFs (in whose path v appears) in handling different UE procedures. It is needed so that two NFs considered for calculating the shortest path (s and t in (1)) can handle these requests successfully. Using this process, Table I lists the final BC of important NFs in the connected graph shown in Fig. 3. We can observe that NRF has the highest BC followed by the UDM and UDSF. Note that, other NFs are not listed in this table as their BC value becomes 0 as per the BC calculations [2].

So, **the Second inference** we obtain from this is that

dependency graphs and BC serve as the prerequisites to realize the possible points of attacks. Yet, these methods do not reveal the possible attack paths by themselves. Therefore, we dive deep into studying the attack graphs next.

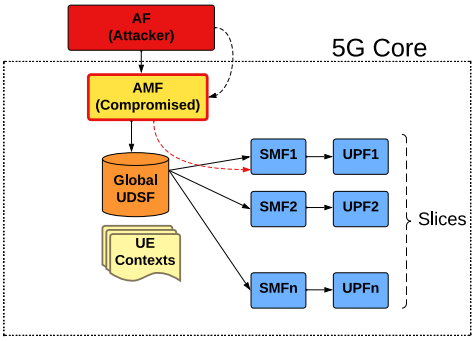


Fig. 4: AF-AMF vulnerability in 5GC reaching to SMF(s), via the global UDSF.

V. Attack Graph Generation & Analysis

To discover the possible new vulnerabilities in the distributed 5GC, we leverage the two-stage process of generating attack graphs and analyzing them systematically.

```
attackerLocated(af).
attackGoal(execCode(smftarget,_)).

hacl(af, amf, tcp, 8080).
hacl(amf, _,  _, _).
hacl(udsf, _, _, _).
hacl(smftarget, _, _, _).
hacl(H,H,_,_).

/* configuration information of udsf */
networkServiceInfo(udsf, mountd, http, 8080, root).
nfsExportInfo(udsf, '/userdata-sm', _anyAccess, smftarget).
nfsExportInfo(udsf, '/userdata-mm', _anyAccess, amf).
vulExists(udsf, vulID, mountd).
vulProperty(vulID, remoteExploit, privEscalation).
localFileProtection(udsf, root, _, _).

/* configuration information of amf */
vulExists(amf, 'CVE-2002-0392', mm).
vulProperty('CVE-2002-0392', remoteExploit, privEscalation).
networkServiceInfo(amf , mm, tcp , 8080 , root).

/* configuration information of smftarget */
nfsMounted(smftarget, '/usr/local/share', udsf, '/userdata-sm', read).
```

Fig. 5: Vulnerability input code to MulVAL [4].

A. Generation of Attack Graph

Understanding the complex relationship of NFs in the 5GC is necessary for determining the impact of possible vulnerabilities in its security context. Since there is no recent open-source software available that can be used to create the attack graph in our work, we leverage a decade-old MulVAL[4], an open-source framework for multi-host, multi-stage vulnerability analysis. It converts vulnerability databases and scanning tool outputs into Datalog [13], making it possible for networks with many entities to be quickly analyzed.

B. Primary Vulnerability

By leveraging the dependency graph and BC detailed in Section IV, we now identify a primary vulnerability that could potentially span across different NFs in the 5GC. Here, the attacker is at the AF initially and then gains access to the AMF, via NRF (see Fig. 2a). So, we call this the primary vulnerability existing at AMF and use this as an example for the study of the attack graph in the rest of the paper. The view of this vulnerability is shown in Fig. 4. For the attack graph generation using MulVAL in the 5GC, we rely on the National Vulnerability Database (NVD) [14] which is a repository of standards-based vulnerability management data named Common Vulnerabilities and Exposures (CVE). Here, we specifically leverage CVE ID CAN-2002-0392 [1], as the objective here is to explore the possible consequences of AMF being compromised eventually modifying the users' data by exploiting the global UDSF. We feed this vulnerability to the MulVAL. Fig. 5 shows the vulnerability code fed to the MulVAL as an input file[2]. The MulVAL then generates the respective attack graph.

C. Analyzing Attack Graph:

To understand the generated attack graph in this work, the conventions followed by MulVAL are briefly listed below.

- Diamond Nodes: These are the privilege nodes, representing *logical OR* relation.
- Oval Nodes: These are the exploit nodes, representing *logical AND* relation.
- Rectangle Nodes: These are the configuration nodes.

Due to space limitations, the complete attack graph generated from MulVAL, the syntax, and the semantics of respective functions are not shown in this paper[2]. But, we focus on one of the attack paths generated there as shown in Fig. 6, and analyze it to identify the possible new attack paths. The attack path shown in Fig. 6, corresponds to one of the paths the attacker could take from AMF up to the SMF via the global UDSF. As shown in this path, the attacker could modify the data of already registered UEs (like UE QoS policy information) at the global UDSF, with a Trojan Horse installed. Due to this, SMF when fetching the corresponding UE's data for establishing the PDU session(s) or modifying them, eventually fetches the modified data. During this process, the related tunnel data and QoS policy for the PDU session are updated to the UDSF by the UPF and SMF and are further conveyed to the RAN for data plane communication. So, the compromised vulnerability at AMF results in a new vulnerability that eventually disturbs the respective UEs' data plane services. The situation is worse, when the 5GC supports multiple slices on the data plane as shown in Fig. 4. One such problem has already been discussed in [8]. Overall, from this analysis of the generated attack graph, we realize that the vulnerability at AMF initially traversed up to UDSF and SMF and then landed on the data plane too. The complete attack graph generated for the example case here can

[1]This vulnerability is defined for web servers like Apache, which allows remote attackers to cause a denial of service and execute arbitrary code via a chunk-encoded HTTP request that causes Apache to use an incorrect size.

[2]More details on syntax and semantics of the code in MulVAL can be availed from the work in [4].

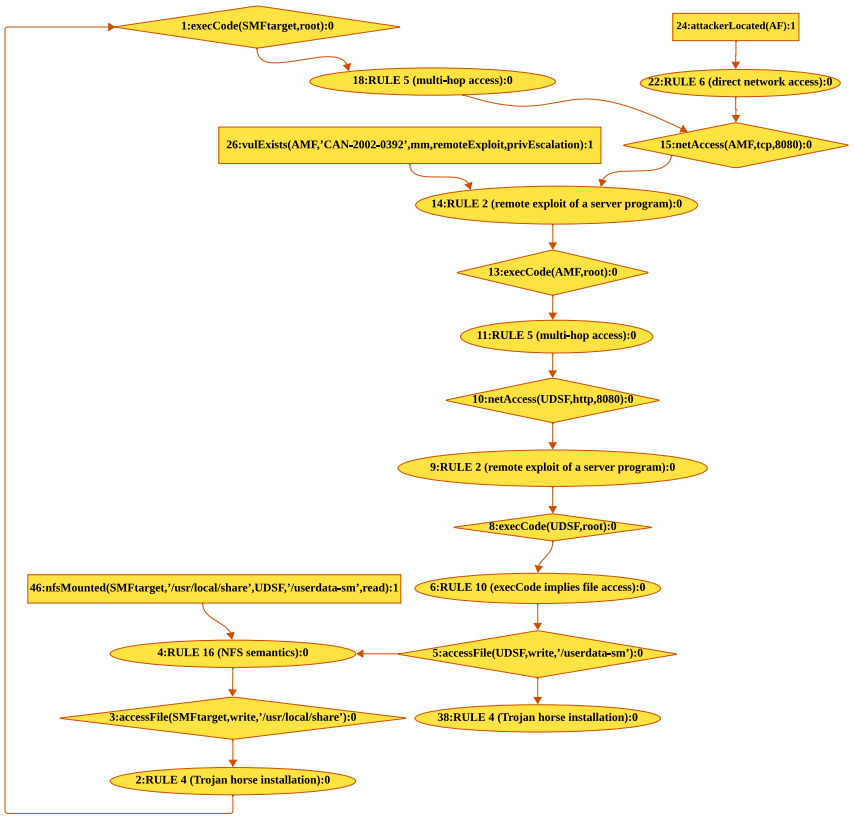


Fig. 6: Attack path from the MulVAL [4] generated attack graph of the vulnerability described in Section V-B.

be further refined using the work detailed in [15]. However, that is outside the scope of this work.

So, **the Third insight** we obtain from this is that generating attack graphs for any suspected or known vulnerability becomes a prerequisite in the 5GC to realize the possible severe problems and side effects it could potentially give rise to. **The Fourth insight** we obtain from this is that it is also important to be aware of the downlink and interaction points between the NFs like AF to AMF interaction, UPF to SMF, and SMF to AMF which can be very dangerous too if no proper safety checks are incorporated in the NF design. **The Fifth inference** we obtain from this is that though control plane and data plane operations are independent in their functions, secure control plane working is very important for the data plane operation to be successful, as otherwise, the HA of the data plane service will eventually deteriorate.

VI. Mitigation Techniques for Attack Paths

By analyzing the attack path shown in Fig. 6, it was observed that the attacker can modify the UEs' data on a large scale by accessing to global UDSF, eventually reaching SMF(s). Hence, as a mitigation technique, we evaluate if the tight inter-dependency between different NFs can be removed or loosened. As a possible solution, we localize the data stored and accessed by AMF and SMF(s) for this example shown in Fig. 4. i.e., we deploy the local UDSF individually

```
attackerLocated(af).
attackGoal(execCode(smftarget,_)).

hacl(af, amf, tcp, 8080).
hacl(amf, _, _, _).
hacl(amf-udsf, _, _, _).
hacl(smf-udsf, _, _, _).
hacl(smftarget, _, _, _).
hacl(H,H,_,_).

/* configuration information of amf-udsf */
networkServiceInfo(amf-udsf, mountd, http, 8080, root).
nfsExportInfo(amf-udsf, '/userdata-mm', _anyAccess, amf).
vulExists(amf-udsf, vulID, mountd).
vulProperty(vulID, remoteExploit, privEscalation).
localFileProtection(amf-udsf, root, _, _).

/* configuration information of smf-udsf */
networkServiceInfo(smf-udsf, mountd, http, 8080, root).
nfsExportInfo(smf-udsf, '/userdata-sm', _anyAccess, smftarget).
vulExists(smf-udsf, vulId, mountd).
vulProperty(vulId, remoteExploit, privEscalation).

/* configuration information of amf */
vulExists(amf, 'CVE-2002-0392', mm).
vulProperty('CVE-2002-0392', remoteExploit, privEscalation).
networkServiceInfo(amf , mm, tcp , 8080 , root).
nfsMounted(amf, '/usr/local/share', amf-udsf, '/userdata-mm', read).

/* configuration information of smftarget */
nfsMounted(smftarget, '/usr/local/share', smf-udsf, '/userdata-sm', read).
```

Fig. 7: Solver code written in MulVAL [4] for mitigating the vulnerability shown in Fig. 4.

for AMF and SMF. For the proposed mitigation technique, we once again write the solution code and feed it to the MulVAL. Fig. 7 shows this written solution code. The code has individual UDSFs for AMF and SMF, namely, AMF-UDSF

and SMF-UDSF with their respective configurations colored in cyan and violet. We see that this time, the MulVAL does not generate any attack graph. So, this confirms that the proposed mitigation technique is one of the solutions to prevent this vulnerability from spreading to the other NFs (SMFs, UPFs) and thereby prevent further problems on the data plane. While the possible attack paths to the SMF via the global UDSF are prevented, note that, here, the AF-AMF vulnerability still exists. However, we limit our study here, as our primary goal to generate attack graphs, analyze them, and identify the possible new attack paths from there, is satisfied. This process can be repeated for all the identified attack paths from the attack graph (see the flow chart shown in Fig. 1).

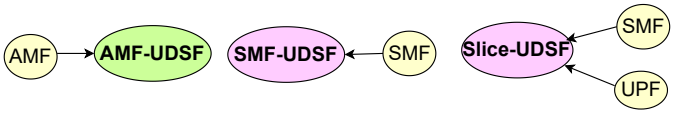


Fig. 8: Dependency graph(s) as per the modified topology.

Further, as an additional verification, we also create the dependency and connected graphs on the modified topology proposed. We then attempt measuring BC on it. Fig. 8 shows the new dependency graphs for modified topology, where the AMF is connected to the AMF-UDSF only, and SMF is connected to the SMF-UDSF or there is a slice-specific UDSF, with SMF and UPF relying on the relevant UE contexts stored there. Here, we can realize that for AMF to AMF-UDSF, there is no BC as there is no NF in between them. Similarly, for SMF-UDSF and SMF. Overall, they are individually connected components, but not a sole connected graph where AMF could modify the UEs' PDU session data for SMF(s) via the global UDSF as earlier (shown in Fig. 4).

So, the **Sixth insight**, we gain, is that shared NFs like UDM and global UDSF can be misused unintentionally or intentionally, paving the way to data poisoning and eventually leading to service disturbance in both control and the data planes. Therefore, the MNO or the slice tenant must ensure to localize the data with an additional local NF like local UDSF for AMF (AMF-UDSF), another for SMF (SMF-UDSF), and so on as per the requirement, but at the additional cost for it.

Finally, we realize that it becomes inevitable for the slice tenant to continuously monitor the interactions with NFs of the other slice irrespective of the control plane or data plane, to ensure that the data communicated or exchanged is zero or very minimal. Whenever, the slice tenants interoperate and communicate across UE, RAN, 5GC NFs, MEC, and the external data network, they need to ensure to have secure communication with Confidentiality, Integrity, and Availability (CIA) triad mechanisms along with regular authentication and authorization. Thus, having anomaly detection built at the respective slice frontiers [8] can be an additional bonus.

VII. Conclusion and Future Work

With the groundbreaking advancements in 5GC architecture comes an amplified risk of security issues. This paper presented a systematic approach to exhaustively cover the study of possible vulnerabilities on the 5GC, where primary vulnerable spots from different NFs are identified first with the help of dependency graphs and Betweenness Centrality (BC). Further, an attack graph is generated on one of the primary vulnerabilities to identify the new attack paths. With the detailed analysis of the generated attack graph and attack paths, we illustrated the serious problems that could eventually affect the ultimate service from the 5GC. Finally, we presented the appropriate mitigation techniques, to solve the newly identified vulnerability. In the future, we plan to review the security challenges in the inter-slice handover scenarios and possible extensions to our proposed study there.

Acknowledgment

This work was supported by the Department Of Science & Technology, India as part of the "Next Generation Wireless Research and Standardization on 5G and Beyond" project.

References

[1] Wikipedia, "Dependency Graph", https://en.wikipedia.org/wiki/Dependency_graph, 2023.
[2] Wikipedia, "Betweenness centrality", https://en.wikipedia.org/wiki/Betweenness_centrality, 2023.
[3] Kengo Zenitani, "Attack graph analysis: An explanatory guide", *Computers Security*, vol. 126, pp. 103081, 2023.
[4] Xinming Ou, Sudhakar Govindavajhala, Andrew W Appel, et al., "Mulval: A logic-based network security analyzer.", in *USENIX security symposium*. Baltimore, MD, 2005, vol. 8, pp. 113–128.
[5] Endri Goshi, Raffael Stahl, Hasanin Harkous, Mu He, Rastin Pries, and Wolfgang Kellerer, "Pp5gs—an efficient procedure-based and stateless architecture for next-generation core networks", *IEEE Transactions on Network and Service Management*, vol. 20, no. 3, pp. 3318–3333, 2023.
[6] Shwetha Vittal and A. Antony Franklin, "Harness: High availability supportive self reliant network slicing in 5g networks", *IEEE Transactions on Network and Service Management*, vol. 19, no. 3, pp. 1951–1964, 2022.
[7] Qiang Tang, Orhan Ermis, Cu D. Nguyen, Alexandre De Oliveira, and Alain Hirtzig, "A systematic analysis of 5g networks with a focus on 5g core security", *IEEE Access*, vol. 10, pp. 18298–18319, 2022.
[8] Shwetha Vittal, Unnati Dixit, Siddhesh Pratim Sovitkar, K Sowjanya, and A Antony Franklin, "Preventing cross network slice disruptions in a zero-trust and multi-tenant future 5g networks", in *2023 IEEE 9th International Conference on Network Softwarization (NetSoft)*, 2023, pp. 227–231.
[9] Ghanshyam S. Bopche, Gopal N. Rai, and Deepnarayan Tiwari, "Rcama - an recursive composition algebra-based framework for detection of multistage attacks", in *2023 15th International Conference on COMmunication Systems NETworkS (COMSNETS)*, 2023, pp. 48–53.
[10] Ghanshyam S. Bopche, Gopal N. Rai, B. M. Mehtre, and G. R. Gangadharan, "Modeling and analyzing multistage attacks using recursive composition algebra", in *Information Systems Security*, Vinod Ganapathy, Trent Jaeger, and R.K. Shyamasundar, Eds., Cham, 2018, pp. 67–87, Springer International Publishing.
[11] Tanujay Saha, Najwa Aaraj, and Niraj K Jha, "Machine learning assisted security analysis of 5g-network-connected systems", *IEEE Transactions on Emerging Topics in Computing*, vol. 10, no. 4, pp. 2006–2024, 2022.
[12] 3GPP, "Procedures for the 5G System", Tech. Rep. TS 23.502, 3GPP, 2022.
[13] S. Ceri, G. Gottlob, and L. Tanca, "What you always wanted to know about datalog (and never dared to ask)", *IEEE Transactions on Knowledge and Data Engineering*, vol. 1, no. 1, pp. 146–166, 1989.
[14] NIST, "National Vulnerability Database", https://nvd.nist.gov/, 2023.
[15] Mehdi Yousefi, Nhamo Mtetwa, Yan Zhang, and Huaglory Tianfield, "A novel approach for analysis of attack graph", in *2017 IEEE International Conference on Intelligence and Security Informatics (ISI)*, 2017, pp. 7–12.