Detection and Localization of DDoS Attack During Inter-Slice Handover in 5G Network Slicing

Himanshu Bisht‡, Moumita Patra‡ and Sathish Kumar*
‡Indian Institute of Technology Guwahati, Assam, India
*Cleveland State University, Cleveland, OH, USA
{himanshu.bisht, moumita.patra}@iitg.ac.in, s.kumar13@csuohio.edu

2023 IEEE 20th Consumer Communications & Networking Conference (CCNC) | 978-1-6654-9734-3/23/$31.00 ©2023 IEEE | DOI: 10.1109/CCNC51644.2023.10060402

***Abstract*—Network slicing plays a crucial role in supporting Fifth Generation (5G) mobile network, which is designed to efficiently accommodate a diverse range of services with varying service level requirements. In this work, our efforts are largely aimed at exposing security flaws in 5G network slicing from a Distributed Denial of Service (DDoS) attack perspective. Time consuming authentication process during the inter-slice handover procedure is exploited to launch a DDoS attack. To address this issue, we offer novel attack detection and localization algorithms. We have compared results for various combinations of average waiting time and average switching rate to detect the attack and localize compromised user equipments. As per experimentation results, our approach resulted in an accuracy of 91% for detecting an attack and 96% for identifying compromised users.**

***Index Terms*—Network Slicing, Inter-Slice Handover, DDoS attack**

I. Introduction

Network slicing enables the transition from a network-as-a-infrastructure to a network-as-a-service architecture, enabling various 5G smart services with varying requirements. In the smart transportation scenario, autonomous car accident reporting has severe latency limitations that must be met by the network in order to enable quick accident reporting and minimize damage. On the other hand, smart agriculture has lower latency requirements than smart transportation systems. Network slicing enables 5G to provide network services based on these specific requirements by various applications [1]. Network slicing allows the building of customized end-to-end logical networks (network slices) on top of shared network infrastructure in a flexible and efficient manner [1], [2].

In a network-sliced environment, user mobility must be handled not only between different base stations or access methods but also across different slices. The availability of a communication service across multiple slices allows users to change their slice if they desire. If a User Equipment's (UE's) requirements change over time, it may seek to modify its slices. Slice owners might also want to move users out of a slice, causing them to seek out other slices to reconnect. As a result, in a network-sliced environment, handovers between different slices (i.e., inter-slice handovers) are expected in addition to usual horizontal (i.e., inter-cell/base-station handovers) and vertical (i.e., inter-technology handovers) [3]. Network slicing presents a wide range of security challenges in addition to many implementation challenges [1], [2]. Vertical and horizontal handovers from the previous generation have been thoroughly studied and reviewed, but not the inter-slice handover. During this handover, mutual authentication occurs between the UE and the network slice [3]. This is a time-consuming and resource-intensive process that can be triggered for a variety of reasons [3]. An attacker can utilize this to launch a DDoS attack. Following are the contributions of our work:

- We show the effects of DDoS attack during inter-slice handover in a 5G network sliced scenario.
- We propose a detection and localization algorithm to limit a potential DDoS attack.

The rest of the paper is organized as follows: Related work is discussed in Section II. System model is presented in Section III followed by Threat Model in Section IV. Section V presents the proposed detection and localization solution. Section VI discusses results and finally, Section VII concludes our paper.

II. Related Works

Survey papers such as [1], [2], [4] provide an extensive study on network slicing, implementation challenges, security challenges, and research directions. In [3], the authors provide a comprehensive explanation of the inter-slice handover process, important 3GPP standards, and various scenarios that lead to it.

A. Denial of service attacks in cellular networks

Authentication and initial registration processes are necessary for privacy and security, but these procedures take time and resources. This fact has already been exploited in the form of a DDoS attack. In these works [5]–[7], authors have given examples of such attacks where they use signaling amplification to launch a Denial-of-Service (DoS) attack. In [5] and [6], the authors have focused on attacks on the Third and Fourth generations. Therefore, the solutions provided by them are not effective in 5G network-sliced environments. Although the attack described in [7] is carried out in a network sliced environment, it ignores the fact that Authentication and Key Agreement (AKA) and Extensible Authentication Protocol (EAP) is carried out while switching between network slices and focuses only on the initial authentication of UEs. The only work that

considers that an inter-slice handover is a resource intensive and time consuming process and considers a DDoS attack using this vulnerability is [8]. However, it does not provide a solution for the attack and just mentions defense strategies such as building backup network slices or establishing threshold based detection of attack.

B. Authentication delay in network sliced environment

The authors of [9], [10], and [11] have highlighted the importance of AKA in a network slicing context, as well as the fact that authentication is a time and resource-intensive procedure. They present a novel mechanism each to reduce the delay associated with the AKA process. Reducing AKA delay may increase the amount of time and number of bots necessary to launch an attack, but it will not eliminate the DDoS attack threat because UEs can still initiate the inter-slice handover request. Furthermore, minimizing the latency during AKA results in an increase in resource utilization as per the solution provided.

C. Existing works on solution to DDoS attack in network sliced environment

A new privacy-preserving slice selection protocol is described in [12]. This protocol eliminates the possibility of users requesting slice switches, and the decision is made completely by the network slice itself. This restricts the capabilities and flexibility promised to users in the 3GPP network slicing definition [13]. The work in [14] describes implementing a machine learning model in 5G to defend against a wide range of attacks. The DDoS attack during inter-slice handover is not covered in [14]. The attacks described in [7] are countered by the authors with a suitable slice isolation strategy, but the DDoS attack during inter-slice handover cannot be countered with their solution because they do not consider inter-slice handover requests issued by UEs. Based on our literature review, as per our knowledge, no work in the literature implements DDoS threat model from 5G network slicing perspective, as explained in Section IV. Consequently, we believe our proposed solution is a first of its kind given the scenario.

III. System Model

Fig. 1 depicts our system model which consists of two network slices (slice 1, slice 2) which share some network functions such as the Access and Mobility Management Function (AMF) and the Network Repository Function (NRF). However, slice 3 has network functions such as AMF, NRF, User Plane Function (UPF), etc., reserved for itself (complete isolation). The attack focuses on preventing users of a particular network slice i.e., the target slice from receiving the service they desire. Aside from slice components, the 5G core also contains components that are not included in a specific slice, such as User Data Management (UDM), Network Slice Selection Function (NSSF), and so on.

UEs are divided into two groups. One group of UEs will connect to the target slice and the other group containing rest of the UEs will connect to all other slices

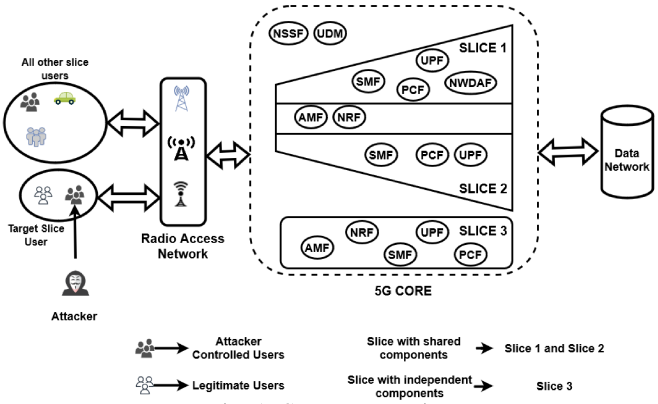

Fig. 1: System scenario

such as IoT, Vehicular Ad hoc Network (VANET), enhanced Mobile Broadband (eMBB), massive Machine Type Communications (mMTC) and personalized slices, etc. Target slice users are further subdivided into

- Legitimate users
- Compromised users / Attacker controlled users / Bot users

We use the terms inter-slice handover/ switching/ mobility interchangeably throughout this work.

IV. Threat Model

We present a threat model where we expose the vulnerability of inter-slice handover process, which enables a DDoS attack on a network slice.

A. Vulnerability in network slicing during inter-slice handover

We have identified the following vulnerability which an attacker can exploit.

- UE can initiate inter-slice handover requests due to the following reasons: the current network slice's access network conditions, end-to-end delay, reliability value, quality of service requirements, monetary costs, isolation and security policies [3].
- The authentication and re-authentication process is initiated during inter-slice handover.
- 5G Authentication and Key Agreement (AKA) is a time and resource consuming operation
- Re-authentication in 5G uses Enhanced Authentication Protocol (EAP) which is as time-consuming as initial AKA.

The time-consuming nature of the AKA process is evident because many signaling message exchanges occur between 5G core control plane functions and UE. Also, the process of mutual authentication and key derivation is based on challenge-response mechanisms and symmetric cryptography which are also time and resource-consuming [15]. Based on these vulnerabilities a DDoS attack is possible.

B. Prerequisite for DDoS attack

Before launching a DDoS attack, an attacker must have fulfilled the following requirements:

- The attacker has control over some UEs of the target slice.

- Identification of peak times when a significant number of users will request a specific slice, triggering the 5G AKA process.

C. Attack procedure

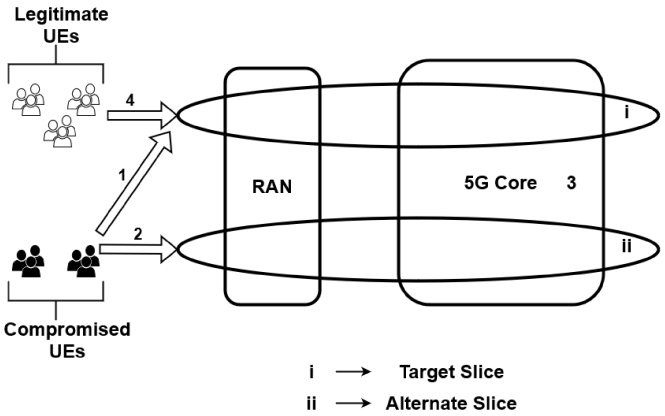

Fig. 2: Attack procedure

1) During the identified period of peak time when most of the users would connect to the target slice, the attacker coordinates the compromised UEs to connect to the target slice (slice (i) as shown in Fig. 2), and initiate the initial AKA.
2) The compromised UEs then initiate the EAP process for re-authentication by performing inter-slice handover between target and alternate slice (slice (ii) as shown in Fig. 2).
3) Each AKA and EAP will engage 5G core components such as AMF/SEAF, AUSF, User Data Function (UDF)/ User Data Management (UDM), Network Slice Selection Function (NSSF) and also reserve some resources for each request till it is completed.
4) When a legitimate user tries to connect to the target slice during the attack duration, it might face denial of service or delay in request completion.

V. Proposed Detection and Localization Model

We propose a three step process as shown in Fig. 3 for detecting an attack and further localizing (identifying) the bot users present in the network slice.

1) Detection:
   a) The monitoring system monitors the traffic flow related to slice switching requests by each user.
   b) Based on collected information, our model periodically checks for an anomaly in the parameter Average Waiting Time (AWT). This parameter is indicative of a possible attack.
   c) If this parameter contains a possible attack signature we activate the second step.
2) Localization:
   a) Once an attack is detected, the system localizes the devices which are under attacker's control (bot devices).
   b) The localization is done based on the value of parameter Average Switching Rate (ASR). A threshold value is calculated based on ASR.
   c) For each user, the number of switching requests per unit time is calculated, which shows how frequently the particular user is switching between slices. If this value is higher than the threshold value, then the system identifies that particular user as a bot user.
3) Reporting:
   a) After the localization of bot users, the list of identified bot users is then reported to 5G core.
   b) These bot users are then blocked by the 5G core and any further requests by these users are not served.

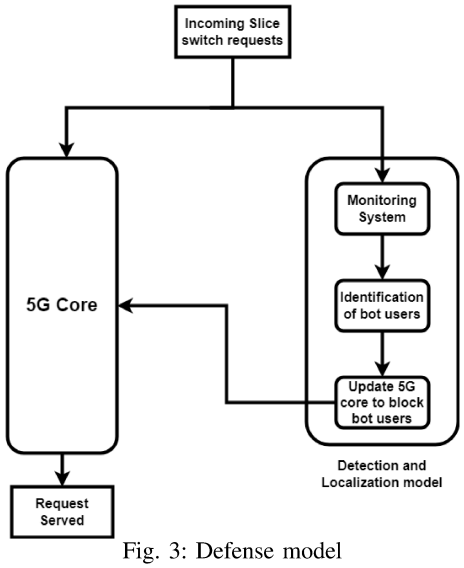

Fig. 3: Defense model

Algorithm 1 describes the attack detection procedure. The input are: *Users* which is list of users, *window_Size* which is the duration to analyze the user's pattern to predicting an anomaly, *threshold_WT* is a key-value mapping of the number of users to its corresponding threshold value of waiting time. The algorithm first calculates the average waiting time of active users denoted by *AWT*. Then the value of *AWT* is compared against the threshold value present in *threshold_WT*. If the algorithm detects an attack, it generates a call to the localization algorithm. Algorithm 2 describes the localization procedure. Input to this algorithm is *Users*. The algorithm first calculates the average switching rate of active users denoted by *ASR*. Based on this, a threshold value is calculated. Each active user's switching rate is then compared against this threshold, and if it exceeds the threshold value, the user id is added to *Bot_Users* and the user is labeled. The algorithm gives *Bot_Users* as output which contains the list of users identified as bot users by this algorithm.

The division of our suggested model into three parts increases the system's robustness and ability to scale quickly. Because the identification of attacks and the localization of bot users are independent of one another, future improvements to any of the modules will be simple to implement. One of the challenges in developing this system was determining how to distinguish a network congestion from an actual attack. If the system recognises a congestion as an attack, it may designate normal users as bots and prevent them from being serviced. This problem is solved by

Algorithm 1 Algorithm for Detection: *DETECT(Users, threshold_WT, window_Size)*

Input : $Users$, $threshold\_WT$, $window\_Size$

repeat
if $(curr\_Time \bmod window\_Size == 0)$ **then**
  Initialize $active\_Users = 0$, $total\_WT = 0$
  for each $(u \in Users)$ **do**
    if $(u.is\_Active == True)$ **then**
      $total\_WT \leftarrow total\_WT + u.WT$
      $active\_Users$++
    end if
  end for
  if $(active\_Users == 0)$ **then**
    $AWT \leftarrow 0$
  else
    $AWT \leftarrow \frac{total\_WT}{active\_Users}$
  end if
  $index \leftarrow (\frac{active\_User}{10}) \times 10$
  if $(active\_Users \bmod 10 \geq 5)$ **then**
    $index \leftarrow index + 10$
  end if
  if $(AWT \geq threshold\_WT.get(index))$ **then**
    $detected++$
    if $(detected \geq 5)$ **then**
      $LOCALIZE(Users)$
      $detected \leftarrow 0$
    end if
  else
    if $(detected \geq 1)$ **then**
      $detected--$
    end if
  end if
end if
until Simulation Ends

combining two separate parameters at two different stages. The AWT used in detection module is indicative of an attack, but during congestion, the AWT of users is high, and our detection model may raise a false alarm. However, because we use ASR during localization, we are able to deal with this. Because a normal user will be waiting in the control plane during congestion and will not request a slice switch, whereas bot users will be frequently changing slices during an attack, increasing the ASR. Further, we consider an attack when the system reaches the threshold value for five windows to make the detection model more reliable for distinguishing between attack and congestion. Because during congestion, the control plane will receive burst of requests at the same instance, but the control plane will process requests sequentially. Eventually, the AWT for subsequent time windows will decrease, which is unlikely in the case of an actual attack, where the bot users will continuously generate slice switch requests. Therefore we examine five windows for detecting whether the system is under attack or not. As a result, our detection model can differentiate between regular congestion and an attack.

Algorithm 2 Algorithm for Localization: *LOCALIZE(Users)*

Input : $Users$
Output : Bot_Users

Initialize $Bot\_Users = []$
for each $(u \in Users)$ **do**
  if $(u.AT \geq curr\_Time)$ **then**
    $u.time\_Spent \leftarrow curr\_Time - u.AT$
    $u.curr\_Rate \leftarrow \frac{u.no\_Of\_Switches}{u.time\_Spent}$
    $total\_Rate = total\_Rate + u.curr\_Rate$
    $active\_Users$++
  end if
end for
$ASR \leftarrow \frac{total\_Rate}{active\_Users}$
if $(active\_Users \geq 1)$ **then**
  $threshold \leftarrow ASR \times 1.5$
  for each $(u \in Users)$ **do**
    if $(u.AT \geq curr\_Time)$ **then**
      if $(u.curr\_Rate \geq threshold)$ **then**
        $Bot\_Users \leftarrow u.id$
        $u.banned \leftarrow True$
      end if
    end if
  end for
end if

VI. Experimentation and Results

A. Experimental setup

We carried our simulation on a discrete event simulator based on java with three isolated slices representing three independent specification verticals, each with its own authentication server. The simulation involved 100 UEs, each of which was given an application request to fulfill. In the initial 1150 milliseconds, the users' arrival time in the environment is dispersed randomly. This is done to avoid congestion at the start of the simulation because in real world scenario, all users connecting at the same time is rare and could lead to false findings. The mobility for a normal user is given by a probability of 0.01.

B. Simulation results for DDoS attack

In this subsection, we show the effect of DDoS attack on the AWT and the number of dropped requests of users. As

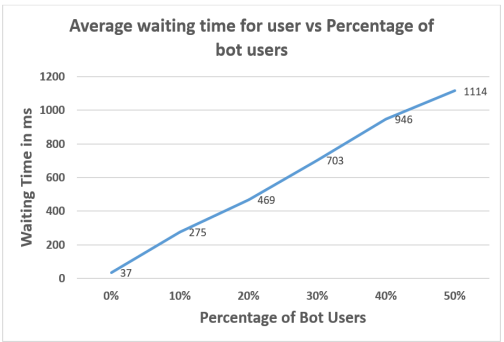

Fig. 4: Attack result on AWT

shown in Fig. 4, we see AWT of users in the environment against the percentage of bot users in the environment.

The waiting time increases linearly with an increase in the number of bot users, which signifies a denial of service to the legitimate users. This is because increasing the bot users in the environment leads to an increased number of inter-slice handover requests, which results in saturation of control plane for normal users. Since the attack is targeted

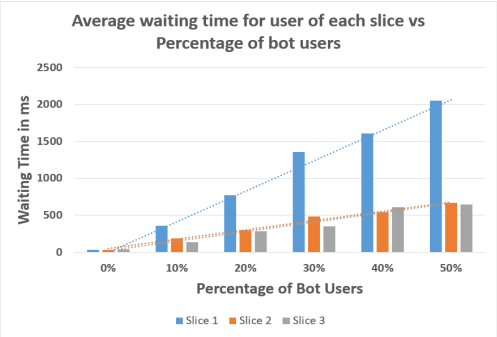

Fig. 5: Attack result on AWT for specific slices

toward slice 1 only, it is important to look at slice specific user data. As shown in Fig. 5, we see that the AWT of users of slice 1 is always significantly higher than slice 2 and slice 3. Although the attack is targeted towards slice 1 but due to the selection of an alternate slice for handover of bot users, the other two slice users are also impacted by the attack as seen in the Fig. 5. The impact intensity of slice 2 and slice 3 is less compared to slice 1. This is because the bot users will be initiating slice switch requests to and from slice 1 to the other two slices. Therefore, the AWT of slice 1 user is almost equal to combined total of slice 2 and slice 3.

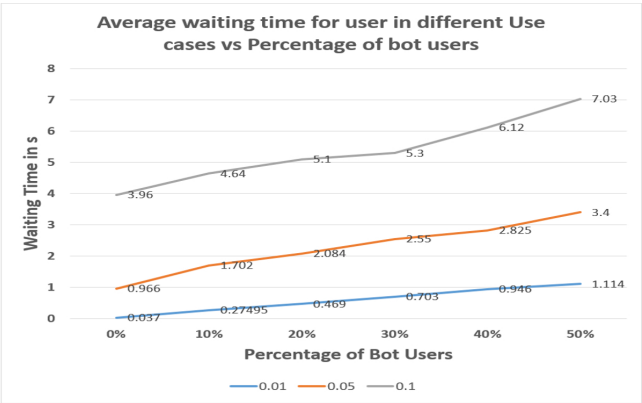

Fig. 6: Attack result on different types of use cases

The attack will impact the environment of different use cases with different intensities. Since the attack focuses on user's ability to initiate slice switch the attack will be more impactful in use cases where a normal user's slice mobility is higher. Fig. 6 shows the attack results in various kinds of use cases represented by various mobility rates of normal users (0.01, 0.05, 0.1). The result shows an increase in attack intensity as the mobility of normal users increases. This is because the increased mobility of normal users will only add slice handover requests to the already saturated control plane, which is under attack by bot users. As specified in [16], in 5G, Timer T300 starts at the transmission of RRC

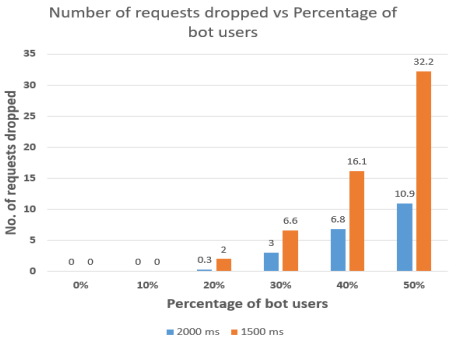

Fig. 7: Attack result on number of requests dropped

Connection Request at UE and stops at reception of RRC Connection Setup or RRC Connection Reject message, cell selection. A request is considered as dropped if the response is not received within given time. The value of this timer varies from UE to UE. We have compared the result with two values of T300 timer– 1500ms and 2000ms as given in [16]. Fig. 7 shows the impact of the attack on the requests dropped. The attack not only increases the waiting time of the user but if the waiting time exceeds the T300 value the request is altogether dropped and a new request is generated by the user. Initially, we do not see much impact with less percentage of bot users, but as the bot users increase, we see an exponential increase in requests dropped.

C. Simulation results after solution is deployed

To demonstrate the effectiveness of our proposed detection and localization model, we conducted experiments to indicate that the model can identify and restore normalcy once an attack has been initiated. To compare the performance of the proposed parameters, we use a confusion matrix and metrics generated from it, such as Recall, Precision, and F1 Score. The four cases compared are:

1) AWT for Detection and Localization
2) ASR for Detection and Localization
3) ASR for Detection and AWT for Localization
4) AWT for Detection and ASR for Localization

Table I: Comparison of various metrics for different combination of parameters for detection and localization

	AWT		**ASR**		**ASR-AWT**		**AWT-ASR**	
	Det	***Loc***	***Det***	***Loc***	***Det***	***Loc***	***Det***	***Loc***
TPR	0.97	0.25	0.35	1	0.35	0.25	0.98	1
FNR	0.03	0.75	0.65	0	0.65	0.75	0.017	0
TNR	0.65	0.65	0.63	0.95	0.85	0.67	0.7	0.95
FPR	0.35	0.35	0.37	0.05	0.15	0.33	0.3	0.05
Precision	0.89	0.15	0.18	0.83	0.87	0.16	0.91	0.83
Recall	0.97	0.25	0.35	1	0.35	0.25	0.98	1
F1 Score	0.93	0.19	0.25	0.91	0.5	0.20	0.94	0.91
Accuracy	0.9	0.57	0.57	0.96	0.475	0.59	0.91	0.96

Table 1 shows the result for different combinations of parameters AWT and ASR for detection and localization. AWT gives the best performance for attack detection but alone is not sufficient to tackle this attack as we can see in the results of case 1, where we see a very poor performance in localization of user. This is because the waiting time of normal users can also be very high due to DDoS attacks and the differentiation between a normal user and a bot user is not concrete. ASR is also considered, as the attack nature is the rapid switching of bot users between

slices. But as the results show ASR is not appropriate for attack detection, especially in the case of less percentage of bot users, because in this case the overall switching rate of users will not have a significant change in order to concretely say that system is under attack. Although for localization of bot users, ASR gives good results. The worst overall performance is registered in the combination where ASR is taken for detection of attack and AWT is considered for localization of bot users. The best overall performance is obtained with a parameter combination of AWT for detection of an attack and ASR for localization of bot users. Based on the above findings we deployed our proposed detection and localization method with parameter combination of AWT for detection of an attack and ASR for localization of bot users during attack event and observed the system behaviour. We ran this simulation for 100 users divided into 80 normal users each with mobility rate of 0.01 and 20 bot users which are all activated at 11500 ms. As shown in Fig. 8 and Fig. 9, we can see the impact 20 bot user have on the system. As the bot users are activated in unison, the AWT and the ASR for users increases. Fig. 10

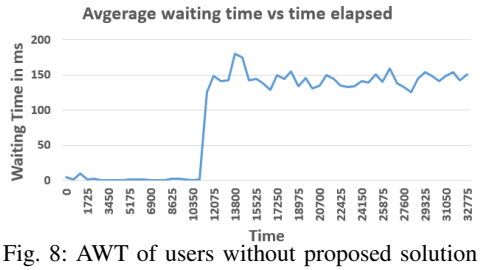

Fig. 8: AWT of users without proposed solution

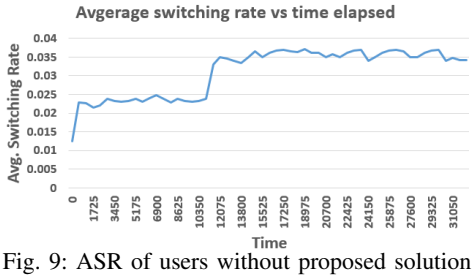

Fig. 9: ASR of users without proposed solution

and Fig. 11 show the efficiency of our model which is able to detect an attack and normalize the system within 4600 ms. The AWT and ASR for users gradually decreases as the time elapses.

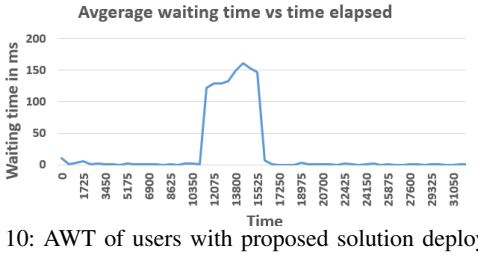

Fig. 10: AWT of users with proposed solution deployed

VII. Conclusions and Future Work

In this work, we have demonstrated the importance of addressing the threat of a DDoS attack during inter-slice handover in 5G. The impact of this DDoS attack is demonstrated using several metrics such as average waiting time for users and number of requests dropped. Further, we propose algorithms for detecting and localizing these DDoS attack. Results for various combinations of these parameters are compared, and based on that, it can be concluded that average waiting time as a metric for detection of an attack and average switching rate as a metric for localization of bot users give the best performance. In future, we plan to conduct a detailed analysis of user slice switching patterns based on which we can propose a prediction-based solution.

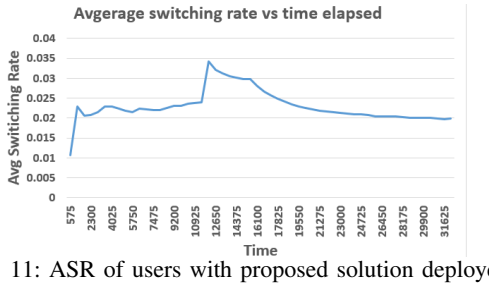

Fig. 11: ASR of users with proposed solution deployed

References

[1] L. U. Khan, I. Yaqoob, N. H. Tran, Z. Han, and C. S. Hong, "Network slicing: Recent advances, taxonomy, requirements, and open research challenges," *IEEE Access*, vol. 8, pp. 36 009–36 028, 2020.

[2] R. F. Olimid and G. Nencioni, "5G network slicing: A security overview," *IEEE Access*, vol. 8, pp. 99 999–100 009, 2020.

[3] M. Sajjad, C. Bernardos, D. Jayalath, and Y.-C. Tian, "Inter-slice mobility management in 5G: Motivations, standard principles, challenges and research directions," Mar 2020.

[4] S. Wijethilaka and M. Liyanage, "Survey on network slicing for internet of things realization in 5G networks," *IEEE Communications Surveys Tutorials*, vol. 23, no. 2, pp. 957–994, 2021.

[5] R. Piqueras Jover, "Security attacks against the availability of lte mobility networks: Overview and research directions," pp. 1–9, 2013.

[6] G. Escudero Andreu, K. Kyriakopoulos, J. Flint, and S. Lambotharan, "Detecting signalling dos attacks on lte networks," pp. 283–301, Aug 2019.

[7] D. Sattar and A. Matrawy, "Towards secure slicing: Using slice isolation to mitigate ddos attacks on 5G core network slices," pp. 82–90, 2019.

[8] V. N. Sathi and C. S. R. Murthy, "Distributed slice mobility attack: A novel targeted attack against network slices of 5G networks," *IEEE Networking Letters*, vol. 3, no. 1, pp. 5–9, 2021.

[9] J. Ni, X. Lin, and X. S. Shen, "Efficient and secure service-oriented authentication supporting network slicing for 5G-enabled iot," *IEEE Journal on Selected Areas in Communications*, vol. 36, no. 3, pp. 644–657, 2018.

[10] Z. Ren, X. Li, Q. Jiang, Q. Cheng, and J. Ma, "Fast and universal inter-slice handover authentication with privacy protection in 5G network," *Security and Communication Networks*, vol. 2021, pp. 1–19, Jan 2021.

[11] C.-I. Fan, Y.-T. Shih, J.-J. Huang, and W.-R. Chiu, "Cross-network-slice authentication scheme for the 5th generation mobile communication system," *IEEE Transactions on Network and Service Management*, vol. 18, no. 1, pp. 701–712, 2021.

[12] V. N. Sathi and C. S. R. Murthy, "Dsm attack resistant slice selection in 5G," *IEEE Wireless Communications Letters*, vol. 10, no. 7, pp. 1469–1473, 2021.

[13] 3GPP, "3rd generation partnership project; technical specification group services and system aspects; system architecture for the 5G system (5GS); stage 2 (release 16). ts 23.501, 3GPP."

[14] M. Iavich, S. Gnatyuk, R. Odarchenko, R. Bocu, and S. Simonov, *The Novel System of Attacks Detection in 5G*, Apr 2021, pp. 580–591.

[15] D. H. Wang, "5G security: Standard and technologies," p. 11, Oct 2017.

[16] 3GPP, "Nr; radio resource control (rrc); protocol specification; 3gpp technical specification, release 15 ts 38.331."