

NFDiscovery Bypass Attack on 5G core Network

1 Introduction

The 5G system offers significant improvements in data speed, latency, and reliability compared to previous cellular networks. However, opening up the 5G core network to third parties presents new security and access control challenges. To address these challenges, the 5G system utilizes the OAuth 2.0 authorization framework as its access control mechanism for the first time. However, there has been no formal analysis of this access control mechanism to date.

The Service-Based Architecture (SBA) of 5G system enables third-party partners to set up additional/complementary core components, e.g., slices on top of network provider's. The introduction of third parties presents new security and access control challenges. To address these challenges, the 5G system utilizes the OAuth 2.0 authorization framework as its access control mechanism for the first time. However, there has been no formal security analysis of the design of this access control mechanism to date.

In our recent research endeavor, we attempted to formally verify the access control mechanism of 5G core. However, this poses several critical challenges, including the lack of commercially deployed 5G Core networks, incomplete open-source implementations, and scalability issues due to numerous configurations.

To address these challenges, we developed NGCoreVerifier, a model-based framework that leverages parameterized model checking to test various 5G core network configurations. We reduce the problem of verifying the access control mechanism of 5G core into a model checking problem. Our framework incorporates a modular design and several 5G system-specific abstraction mechanisms to ensure flexibility, customizability, and scalability of

our analysis. Our testing upon 73 safety properties on 27 different core network configurations uncovered six new weaknesses in the 5G access control mechanism which can be exploited by an compromised NF to gain unauthorized access to various sensitive resources. A research paper on our work is currently under review.

We responsibly disclose one of our findings in the following.

2 NFDiscovery Bypass Attack

2.1 Main idea

The main idea of the attack is that a malicious consumer (C_1) is able to discover a producer (P_1), even if the `allowedSnsais` attribute in P_1 's NF-Profile clearly forbids access of C_1 . In *NFDiscoveryRequest*, the consumer may set two attributes along with others i.e. `sNssais`, which denotes sNssais to be discovered by the consumer, and `requestersNssais`, which refers to sNssais served by the consumer. During verification process, NRF shall follow somewhat the following steps. First, it finds those target NFs that serves the sNssais as appeared in `requestersNssais`, and then, filters them based on `sNssais` attribute [1]. However, because both of the attributes in the message are set by the consumer, if it is malicious, it can set these attributes to any values to discover any NF in the 5G core. In this way, a compromised consumer NF can extract NFProfile of any NF which includes sensitive metadata of the victim NF that may be exploited further as explored in the our paper.

2.2 Example demonstrating the vulnerability

Consider a partial core network setup that contains a compromised consumer C_1 (e.g. AMF) in sNssai 1, and a benign producer P_1 (e.g. UDM) in sNssai 3 and only accepts *NFServiceRequests* from sNssai 3. A malicious consumer can exploit the vulnerability using the following flow.

Step ①: Consumer C_1 invokes *NFDiscoveryRequest* API call with `sNssais` and `requesterSnsais`, both, set to be sNssai 3 along with other parameters.

Step ②: After receiving the message, NRF, first, pull all the NFs of the target NF type. Here the target NF type is UDM. So, NRF will find P_1 . Further, NRF will check whether `sNssai 3` (as in `requesterSnsais`) matches the `allowedSnsais` of P_2 . As it is a match, NRF, finally, checks if C_1 wants to discover `sNssai 3` (as in `sNssais`). As it the case here, NRF will send P_1 's NFProfile with related NFServices to the consumer C_1 .

Note that even though we only consider an AMF instance as a consumer NF and a UDM as a producer instance, this attack is applicable for other type of NF instances as well.

2.3 Adversary Assumption

To successfully carry out the attack, the NF service consumer needs to be malicious so that the adversary can forge and make API calls on behalf of the consumer.

2.4 Root Cause

The root cause of the attack is the lack of cross-checking between the `requesterSnsais` attribute in the *NFDiscoveryRequest* message, and `sNssais` attribute in the NFProfile of the consumer NF. While the `allowedSnsais` of the target NFs is checked against `requesterSnsais` by NRF, the latter is not cross-checked against `sNssais` in the consumer NF's profile. Note 12 of Table 6.2.3.2.3.1-1 in 3GPP TS 29.510 [1] explains that the `requesterSnsais` should be checked against the target NF's `allowedSnsais` but we have not found any reference in the specification that clearly mandates the above cross-check. This check is important to ensure that the consumer is not exploiting `requesterSnsais`. Lack of this check allows a malicious consumer to misuse the `requesterSnsais` attribute by setting it to any arbitrary `sNssai` in the *NFDiscoveryRequest*, which, subsequently, exposes NFProfiles containing sensitive metadata e.g. endpoints of producer NFs, etc.

2.5 Possible Fixes

As discussed on the above. 5G specification has not mandated the cross-check between `requesterSnsais` in the *NFDiscoveryRequest* message and

`sNssais` of the NF service consumer during the verification of *NFDiscoveryRequest* by NRF. From our speculation, enforcing this check should fix the above weakness.

References

- [1] 3GPP. 5G System; Network function repository services; Stage 3. TS 29.510. 17.7.0.