A Design of Infrastructure Backup System for Private 5G Networks

Ha Duong Phung
Dept. of Information and Communication Convergence Soongsil University
Seoul, Korea
phunghaduong@dcn.ssu.ac.kr

Thang Tran Huy
Dept. of Information and Communication Convergence Soongsil University
Seoul, Korea
trhthang0401@dcn.ssu.ac.kr

Younghan Kim
School of Electronic Engineering Soongsil University
Seoul, Korea
younghak@ssu.ac.kr

***Abstract*— The private 5G network has matured rapidly in recent years based on the service-based architecture (SBA) of cloud-native. However, virtualization technology and layered separation design of 5G core networks still face many difficulties in the management of network operation, maintenance, and fault handling, especially in disaster contexts. Therefore, the requirement for the backup strategy of private 5G networks is needed for high reliability. In this paper, we propose a design of an infrastructure backup system to ensure the high reliability of private 5G networks. In our design, 5G network functions are deployed and managed automatically by the infrastructure orchestration, and system information is managed by the backup controller to meet the reliability guarantee in case of disasters or system failures.**

Keywords—Private 5G network, disaster backup, automation infrastructure

I. Introduction

With the conclusion of Release-17 within the 3GPP [1], the field of telecommunications progresses into the realm of 5G-Advanced, marking a new stage in global 5G commercial deployment. The primary objective of the designers is to furnish technological solutions encompassing high-speed, minimal latency, and extensive connectivity for three central scenarios: Ultra-Reliable Low Latency Communications (URLLC), Massive Machine-Type Communications (mMTC), and Enhanced Mobile Broadband (eMBB). This pursuit aims to enhance the dependability of the 5G network. However, in contrast to the conventional architecture of the 4G network, the architecture of the 5G core (5GC) network adopts a service-based approach rooted in cloud-native technology[2]. This entails a separation between the control plane and the user plane. The control plane consists of diverse Network Functions (NFs), each associated with its own set of microservices and responsibilities, interconnected through service-based interfaces. Therefore, mobile core networks have evolved towards the perspective on Network Function Virtualization (NFV), guaranteeing the adjustment of Virtual Network Functions (VNFs) operating within virtual machines (VMs). These have been replaced in favor of Cloud-native Network Functions (CNFs), alternatively known as Network Functions based on Kubernetes (KNFs).

Figure 1 describes Private 5G Network Architecture with NFV technology. At the Central Cloud, 5G Core Network Functions can be deployed together as VNF(s) in the same cluster. But in some scenarios, due to the constraints imposed by the 5G requirements in terms of low latency, high availability, some network functions can be deployed separately in other cluster at different locations to effectively handle and meet the requirements. Therefore, the management of network operation, maintenance, and fault handling becomes very complicated. Thus, disaster backup for a private 5G network needs to be considered to ensure reliability. For instance, UDM stores 5G subscriber data and session contexts for authorization, registration, mobility management, and session management. So UDMs are vulnerable to failure due to database synchronization or if other NFs(e.g., AMF, SMF) fail, users cannot access 5G networks.

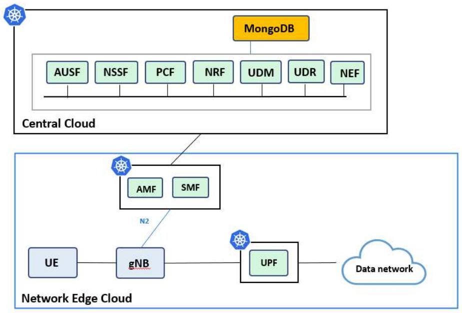


Fig. 1 Cloud-native-based private 5G network architecture.

In this paper, we propose a design architecture for an infrastructure backup system in a private 5G network to address the above challenges. Our private 5G network is deployed and managed by an automation infrastructure of network functions and we leverage the backup technology to provide the backup ability of private 5G Network Functions.

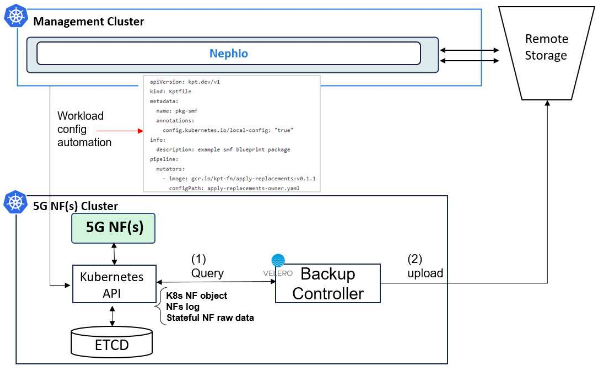
II. Proposed Architecture

To address these above challenges, our proposed architecture has two main components: Management Cluster and 5G Network Functions Cluster. The Management Cluster is the automation infrastructure that simplifies the deployment and management of multi-vendor cloud and network functions across large-scale edge deployments. In this proposal, we utilize the Nephio[3] for that responsibility. Nephio is Kubernetes-based cloud-native intent automation. It enables faster onboarding of network functions including provisioning of underlying cloud infrastructure. Therefore, we use Nephio to automatically configure and deploy workload as well as 5G Network Function(s) on distributed 5G Clusters. In our proposed system, firstly depending on the User-defined Kubernetes manifest file, Nephio can create a corresponding K8S cluster. In the created K8S cluster, Nephio can automatically deploy and manage the 5G Network Function(s) as the workload. As Figure 2 describes, Nephio reads all user-defined config from the kpt file on the remote repository and then generates corresponding NF packages to deploy them into the K8S cluster.

In each 5G NF(s) Cluster, as Figure 2 describes, each cluster can deploy one or more 5G Network Functions depending on the operators. We assume that the 5G Network Functions are deploying and running based on Kubernetes environments.

979-8-3503-1327-7/23/$31.00 ©2023 IEEE

Each cluster is deployed with one Backup Controller which communicates with Kubernetes API for scraping backup data. Since manual cluster backup and restoration need human participation, well-defined automated provisioning methods are necessary. We choose Velero[4] as one of the most automated backup tools for K8S. Velero is an open-source tool for safely backing up and restoring resources in a Kubernetes cluster, performing disaster recovery, and migrating resources and persistent volumes to another Kubernetes cluster.



```
apiVersion: kpt.dev/v1
kind: Kptfile
metadata:
  name: pkg-smf
  annotations:
    config.kubernetes.io/local-config: "true"
info:
  description: example smf blueprint package
pipeline:
  mutators:
    - image: gcr.io/kpt-fn/apply-replacements:v0.1.1
      configPath: apply-replacements-owner.yaml
```

Fig. 2 Proposed Infrastructure backup system architecture.

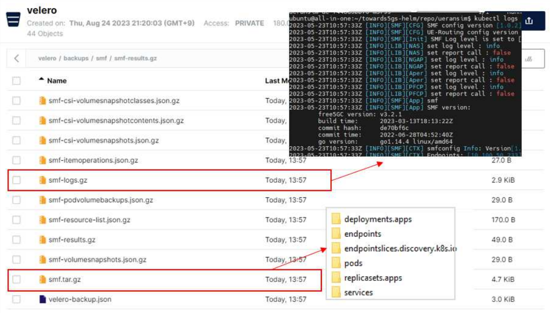
When the 5G NF(s) Cluster is initiated, Nephio allows user to define their own 5G NF workload configuration which is abstracted to be easy to configure as Figure 2 describes. Then extracting abstract workload configuration, Nephio generates corresponding 5G NF packages, deploy them into K8S cluster and creates Backup object(s). Each 5G Network Function has one Backup object and Backup object(s) are managed by one Backup controller. It collects the data to back up by querying the K8S API server for Network Function resources.

There are two types of 5G network functions: stateless 5G Network Function or stateful 5G network Function. Backup controller should identify the type of 5G network function to scrape the data for backup. If it is stateless network function such as AMF, PCF,.., backup controller communicate with K8S API to get Network Function object metadata: deployment yaml file, service yaml file,.. and its current log file. And if it is stateful network function like UDM, backup controller does not only scrape its object metadata, but also query the data location where that stateful NF stores and saves that raw data. After that, The Backup Controller makes a call to the remote storage service – for example, AWS S3 – to upload the backup file if the disaster happens.

III. Preliminary Experimental results

In our experiment, we deploy 5G private network on multi clusters as illustration in Figure 1. Each cluster we utilize Nephio for deploying corresponding 5G NFs working with our *BackupController*. Our backup controller enhances Velero tool to backup important data for 5G private network.

Figure 3 shows what backup data is necessary for a 5G stateless NF(s). There are two important types of data we are focusing on: the current logs of running 5G NF(s) and the current K8S objects which was created to run that 5G NF. The current logs can be used for failure detection and K8S NF object information can be useful for the restore phase.



velero
Created on: Thu, Aug 24 2023 21:20:03 (GMT+9) Access: PRIVATE 180.
44 Objects

velero / backups / smf / smf-results.gz

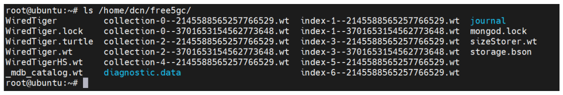
Name	Last M	
smf-csi-volumesnapshotclasses.json.gz	Today,	
smf-csi-volumesnapshotcontents.json.gz	Today,	
smf-csi-volumesnapshots.json.gz	Today,	
smf-itemoperations.json.gz	Today, 13:57	27.0 B
smf-logs.gz	Today, 13:57	2.9 KiB
smf-podvolumebackups.json.gz	Today, 13:57	29.0 B
smf-resource-list.json.gz	Today, 13:57	170.0 B
smf-results.gz	Today, 13:57	49.0 B
smf-volumesnapshots.json.gz	Today, 13:57	29.0 B
smf.tar.gz	Today, 13:57	4.7 KiB
velero-backup.json	Today, 13:57	3.0 KiB

```
ubuntu@all-in-one:~/towards5gs-helm/repo/ueransim$ kubectl logs
2023-05-23T10:57:33Z [INFO][SMF][CFG] SMF config version [1.0.2]
2023-05-23T10:57:33Z [INFO][SMF][CFG] UE-Routing config version
2023-05-23T10:57:33Z [INFO][SMF][Init] SMF Log level is set to [
2023-05-23T10:57:33Z [INFO][LIB][NAS] set log level : info
2023-05-23T10:57:33Z [INFO][LIB][NAS] set report call : false
2023-05-23T10:57:33Z [INFO][LIB][NGAP] set log level : info
2023-05-23T10:57:33Z [INFO][LIB][NGAP] set report call : false
2023-05-23T10:57:33Z [INFO][LIB][Aper] set log level : info
2023-05-23T10:57:33Z [INFO][LIB][Aper] set report call : false
2023-05-23T10:57:33Z [INFO][LIB][PFCP] set log level : info
2023-05-23T10:57:33Z [INFO][LIB][PFCP] set report call : false
2023-05-23T10:57:33Z [INFO][SMF][App] smf
2023-05-23T10:57:33Z [INFO][SMF][App] SMF version:
        free5GC version: v3.2.1
        build time:      2023-03-13T18:13:22Z
        commit hash:     de70bf6c
        commit time:     2022-06-28T04:52:40Z
        go version:      go1.14.4 linux/amd64
2023-05-23T10:57:33Z [INFO][SMF][CTX] smfconfig Info: Version[1.
```

deployments.apps
endpoints
endpointslices.discovery.k8s.io
pods
replicasets.apps
services

Fig. 3 Backup data for 5G SMF.

Figure 4 shows how we backup stateful data for the stateful NF(s). To backup stateful NF(s), we also backup the current logs and the K8S NF objects same as stateless NF backup. But one more requirement is the state data of that 5G stateful NF(s). The current Velero just backup persistent volumes object but does not care to backup that raw local data. Therefore, our enhanced backup controller communicates with K8S API to get the location where persistent volume saves the raw data and backup that folder.



```
root@ubuntu:~# ls /home/dcn/free5gc/
WiredTiger          collection-0--2145588565257766529.wt  index-1--2145588565257766529.wt  journal
WiredTiger.lock     collection-0--3701653154562773648.wt  index-1--3701653154562773648.wt  mongod.lock
WiredTiger.turtle   collection-2--2145588565257766529.wt  index-3--2145588565257766529.wt  sizeStorer.wt
WiredTiger.wt       collection-2--3701653154562773648.wt  index-3--3701653154562773648.wt  storage.bson
WiredTigerHS.wt     collection-4--2145588565257766529.wt  index-5--2145588565257766529.wt
_mdb_catalog.wt     diagnostic.data                       index-6--2145588565257766529.wt
root@ubuntu:~# 
```

Fig. 4 Backup raw data for 5G UDM.

IV. Conclusion and Future Works

Our system provides end-to-end 5G network functions orchestration that deploy and manage automatically 5G Network Function(s) on distributed edge environments. With the support of automated infrastructure management, our proposed system improves the reliability of the 5G private network by using a backup controller for managing system information and status. With this ability, it also supports recovery capability in case of disaster or failure. For future work, we will consider enhancing our design by using other approaches (i.e. Machine Leaning) to enable proactive backup ability in further research.

Acknowledgment

This work was supported by Institute of Information & Communications Technology Planning & Evaluation (IITP) grant funded by the Korea government(MSIT) (No.2022-0-01015, Development of Candidate Element Technology for Intelligent 6G Mobile Core Network).

References

[1] 3GPP, "Release 17 Description; Summary of Rel-17 Work Items,"3rd Generation Partnership Project (3GPP), Technical report (TR 21.916), Jul. 2022, version 0.7.0.. [Online]. Available: https://portal.3gpp.org/desktopmodules/Specifications/SpecificationDetails.aspx?specificationI d=3937.

[2] Y. Wu and X. Wang, 'Research on Network Element Management Model Based on Cloud Native Technology', in 2022 IEEE 2nd International Conference on Computer Communication and Artificial Intelligence (CCAI), 2022,

[3] 'Nephio - Cloud-native network automation'. https://nephio.org/ (accessed August 2023).

[4] Arundel, J.; Domingus, J. Cloud Native DevOps with Kubernetes: Building, Deploying, and Scaling Modern Applications in the Cloud; O'Reilly Media: Newton, MA, USA, 2019; ISBN 978-1492040767.