Carrier-Grade Mobile Packet Core Network on AWS

Best Practices for Designing Carrier-Grade Mobile Packet Core Virtual Network Functions on AWS

November 2019

Notices

Customers are responsible for making their own independent assessment of the information in this document. This document: (a) is for informational purposes only, (b) represents current AWS product offerings and practices, which are subject to change without notice, and (c) does not create any commitments or assurances from AWS and its affiliates, suppliers or licensors. AWS products or services are provided “as is” without warranties, representations, or conditions of any kind, whether express or implied. The responsibilities and liabilities of AWS to its customers are controlled by AWS agreements, and this document is not part of, nor does it modify, any agreement between AWS and its customers.

Contents

Abstract

This whitepaper introduces the key considerations and best practices for designing carrier-grade 4G Long Term Evolution (LTE) Evolved Packet Core (EPC) workloads on AWS. These carrier-grade packet core network EPCs can support millions of mobile subscribers while meeting high availability and scalability requirements. AWS telecom partners have created carrier-grade EPC virtual network functions (VNF) tailored to the AWS Cloud that satisfy telecom industry requirements. This offers a new cost effective, pay-as-you-go model for large, mobile core networks that is available globally in minutes. With this solution, mobile network operator (MNO) or mobile virtual network operator (MVNO) can deploy new 4G core networks on AWS with the ability to easily evolve to the 5G core networks of the future.

Introduction

The market for Mobile Packet Core Network, also known as Evolved Packet Core (EPC), is still growing despite the maturity of the 4G LTE market. This is because of forthcoming opportunities such as MVNO, business plans for new network slices like IoT and 5G Non-standalone (NSA) deployments, and the increased amount of data traffic that subscribers are using[1]. However, because revenue for traditional voice and data service still remains either flat or slowly increasing compared to the demand of network scalability, Network Function Virtualization (NFV) for EPC is the essential choice for mobile service providers to expand or newly deploy the packet core network. NFV has already shown the telecom industry the cost efficiency benefits of using a common Commercial Off-the-Shelf (COTS) server and resource sharing among network entities with achieving scalability of network functions[2]. Furthermore, NFV brought faster instantiation of network functions (VNF) because it decouples the software and hardware, and is able to shorten the preparation schedule for new deployments, compared to legacy, hardware-based network equipment.

NFV solves many of the operational burdens and cost concerns of traditional, legacy, hardware-based network systems[2, 3]. There are still many challenges, however, such as difficulties in orchestration and OpenStack management; the server host OS, storage, and network infrastructure; and hardware dependencies (COTS server, switch or router). Though NFV tried to bring independence to the underlying hardware—compared to hardware-based systems—it still has the constraints of the COTS server and OpenStack installation, deployment, and management. Also, despite the advancements that NFV offers, most mobile service providers must continue to overprovision servers, storage, and networks to handle sudden traffic surges, whether they are temporary or expected seasonal increases.

The solution for this continuing telecom industry challenge is the same solution other IT industry professionals have found: hosting their enterprise applications and infrastructure on the AWS Cloud. Even though the solution is clear and straightforward, mobile service providers have been hesitant to adopt this solution because mobile packet core networks like VNFs have characteristics with very specific requirements, such as high-performance computing, network packet processing, and high-availability. More specifically, VNFs require techniques such as Single Root I/O Virtualization (SR-IOV) Data Plane Development Kit (DPDK), Hugepage, Non-uniform Memory Access (NUMA), and Anti-affinity group deployment.

Before mobile service providers can build their architectures in the cloud, they need clear guidelines for how their requirements can be mapped to existing AWS services, and how vEPC (virtual Evolved Packet Core) such as VNF can be reformed for the public cloud.

One example of how mobile service providers can correlate AWS services and best practices of architecting vEPC on AWS is Affirmed Networks. Affirmed Networks is a leader in virtualized mobile networks. Its Mobile Content Cloud (MCC) has provided 3GPP, fully-compliant Mobility Management Entity (MME) and Serving/PDN-Gateway (SPGW) functions as an industry-proven EPC solution. Many MNO and MVNOs are already using on-premises versions of this solution all around the world. This vEPC solution on AWS will enable mobile service providers—MNO, MVNO, and also enterprises who want private LTE—to easily launch versatile and 3GPP compliant vEPC, with the benefit of OPEX/CAPEX savings that is automatically available with the *pay-as-you-go* model of cloud economics. Because this solution offers one of the biggest benefits of the cloud—scalability—when mobile service providers start a new service, they can make sure their core network has the necessary capacity to avoid network resource overbooking.

This whitepaper describes the benefits of vEPC on AWS. It includes a vEPC reference architecture, an overview of EPC functions and requirements based on the 3GPP standard, use cases, and best practices for architecting vEPC on AWS, which includes high-availability, scalability, security, performance, and operational excellence. It also contains information about how you could use this vEPC solution on AWS to develop a next generation 5G Core solution on AWS.

Benefits & Use Cases of vEPC on AWS

Several telecom partners of AWS have designed and implemented vEPC on AWS. This solution can enable mobile operators to extend their mobile presence to almost anywhere in the globe, without the cost and complexity of deploying a network and data center. Key benefits of this solution include:

- **Less total cost of ownership and no upfront hardware cost**

  In an on-premises environment, not only for hardware installation cost but also operation cost including power and labor cost should be considered. However, using AWS, mobile service providers take full advantage of cloud economics.

- **No end-of-life for hardware or platform**

  All hardware platforms have end-of-life dates, including COTS servers. When the hardware is no longer supported, you must buy new hardware again. With the AWS Cloud, instead of buying new hardware, you can simply upgrade vEPC with new AWS instance types if necessary, in a single click at no cost for the upgrade.

- **Right size anytime**

  When mobile service providers initially build their environments, they typically oversize their core networks to cope with both expected and unexpected traffic surges. With AWS, vEPC core networks can scale up or down automatically with the number of subscribers and the amount of traffic those subscribers are using. The ability to right size the core network during the early phase of deployment allows mobile operators to accelerate their time-to-market for new services and offer those services at a competitive price point.

With these benefits, you can use vEPC on AWS for the following use cases:

- **New service slice enablement such as VoLTE, IoT, and 5G NSA Core**

  When mobile operators start new services, such as 5G NSA, it can be difficult to correctly estimate the effective size of the core network in terms of capacity and cost. The AWS Cloud can be the best choice to successfully launch this new service. Also, new services such as IoT/M2M applications are mostly low-revenue services, which require mobile operators to launch them at a lower cost in order to deliver IoT Services profitably. vEPC on AWS helps mobile operators to scale IoT service easily and in a cost-effective way.

- **Home-routed roaming with local breakout on AWS**

Because of the global presence of AWS, when mobile operators host vEPC user planes or vPGW on AWS in visiting countries, they can provide seamless local breakout roaming service (*Figure 1*). This efficiently breaks out traffic in the regional or visiting country AWS data centers instead of backhauling the user traffic to centralized data centers, which reduces latency and provides exceptional quality of experience to roaming subscribers.

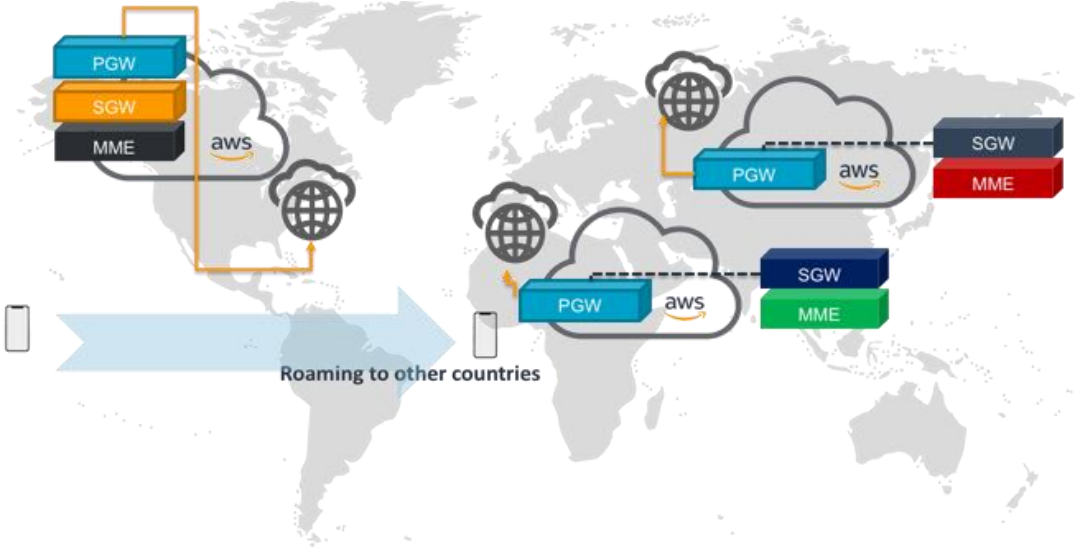


Figure 1 – Home-routed roaming model with local PGW on AWS in global regions

- **Geo-redundancy DR site**

  Having a disaster recovery (DR) site for your existing core network is also challenging in on-premises environments because it involves new investment in hardware installation and operation. vEPC on AWS can provide an easy way for mobile operators to build DR backup networks for their existing core network without purchasing new hardware.

- **Market expansion**

  Breaking into new regional markets is expensive and risky—not only does it require an investment in hardware and data centers, but also an investment in local personnel to manage the network. With vEPC on AWS, mobile operators can quickly and cost-effectively test new markets and services by starting small and scaling as demand grows.

- **3GPP fully-compliant and industry-proven EPC**

  Affirmed networks vEPC is a global leader in virtualized core networks. Its MCC has provided 3GPP, fully-compliant Mobility Management Entity (MME) and Serving/PDN-Gateway (SPGW) functions as an industry-proven solution, with more than 89 deployments in production in over 76 countries around the globe.

Mobile Packet Core Network

Evolved Packet Core (EPC)

The high-level architecture and interfaces of a 4G LTE network, which is composed of a Radio Access Network (RAN) and Core Network (CN), is shown in *Figure 2*. Core Network is typically referred to as an *Evolved Packet Core* (EPC). *Evolved* refers to the 4th generation in the evolution of the mobile network. Radio Access Network is often referred to as *Evolved Universal Terrestrial Access Network* (E-UTRAN) or *eNB* (evolved Node B). EPC is an IP-based, packet switched network that supports 4G data service as well as voice service with the Voice-over-LTE (VoLTE) feature. As a Core Network, EPC plays a role in authentication, authorization, mobility management, QoS enforcement, and charging functions in the 4G network[4].

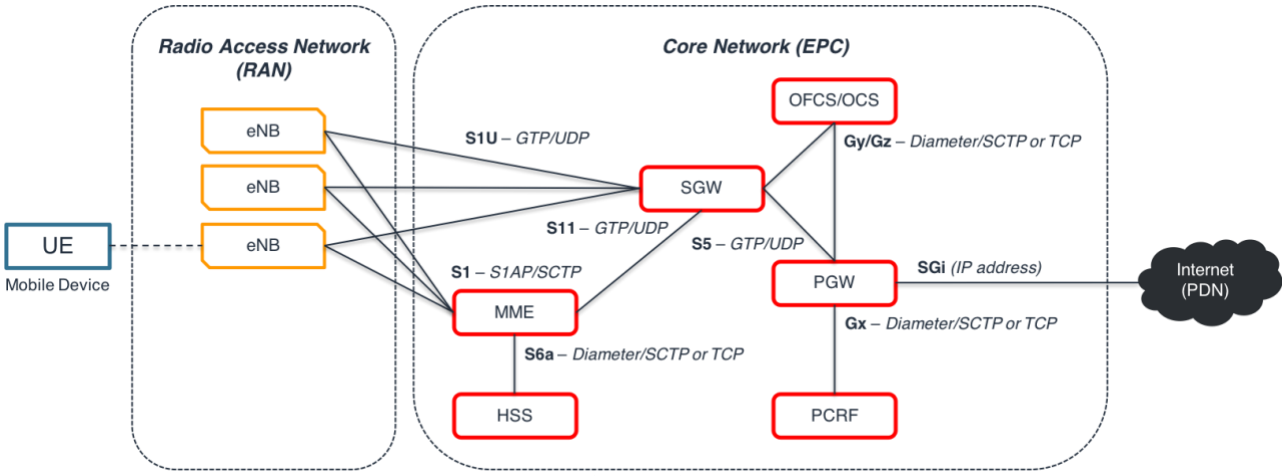


Figure 2 – 3GPP LTE architecture

Elements of EPC

Among EPC elements, major functions are provided by MME (Mobile Management Entity), SGW (Serving Gateway), and PGW (Packet Data Network Gateway), which are outlined in red in *Figure 2.*

- **MME** is a key control plane element and is also responsible for authentication and tracking subscriber mobility. The major functions of MME are authentication, subscriber-session-bearer management, paging, and mobility management, which refers handover and tracking area update (TAU).
- **SGW** forwards and routes subscriber traffic to and from eNB or PGW. It also provides an anchor IP address during intra-LTE handover and a roaming interface to the home PGW.
- **PGW** assigns IP addresses to mobile devices, manages QoS and charging enforcement, manages non-3GPP interworking (such as Wi-Fi networks), and interfaces with external networks (such as the internet or ISP service network). Because both SGW and PGW process user data traffic, they require high rate of data packet processing capability. In common practice, SGW and PGW can be collocated, which some network vendors refer to as *SPGW*.

vEPC on AWS Deployment Model

When you deploy vEPC on AWS, vEPC integrates with either Macro eNB or Smallcell eNB in your legacy network, as shown in *Figure 3*. These examples of network configuration for vEPC on AWS include:

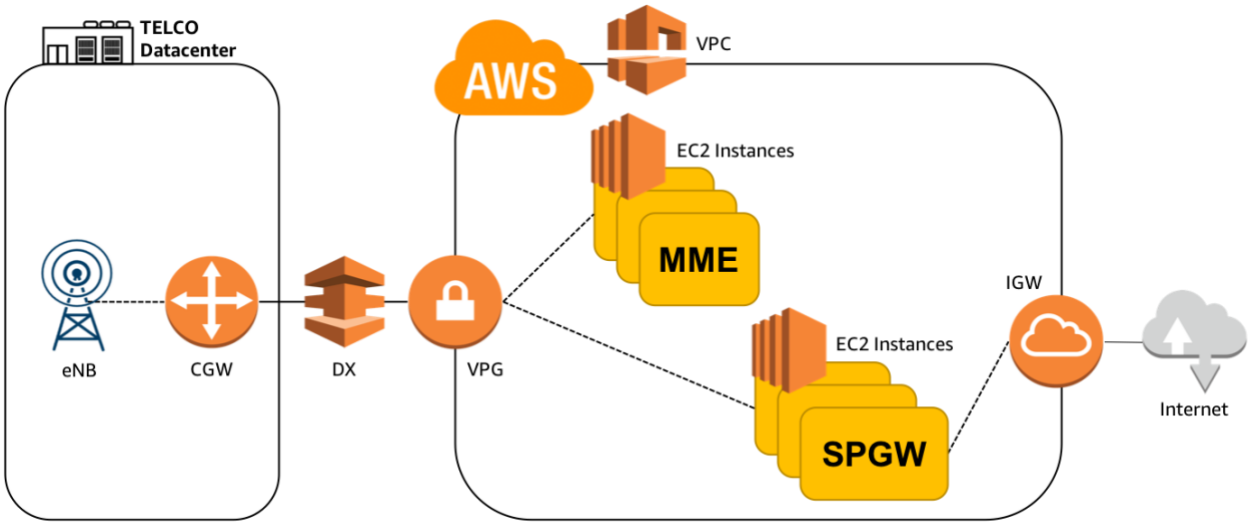
- Option A – All vMME and vSPGW functions are located in one VPC
- Option B – vMME and vSPGW are hosted separately on each VPC using VPC peering
- Option C – SGi through telecom data center
- Option D – Hybrid network with the control plane only on AWS—3GPP Control and User Plane Separation (CUPS) architecture (Control Plane on AWS and Data Plane on-premises)

Option A is the simplest configuration, which is also cost effective because all vEPCs are in the cloud. With Option B, vMME and vSPGW must be hosted in two Regions. They can then be connected using VPC peering on AWS. Options C shows how traffic through the SGi interface to the internet is routed back to the telecom data center,

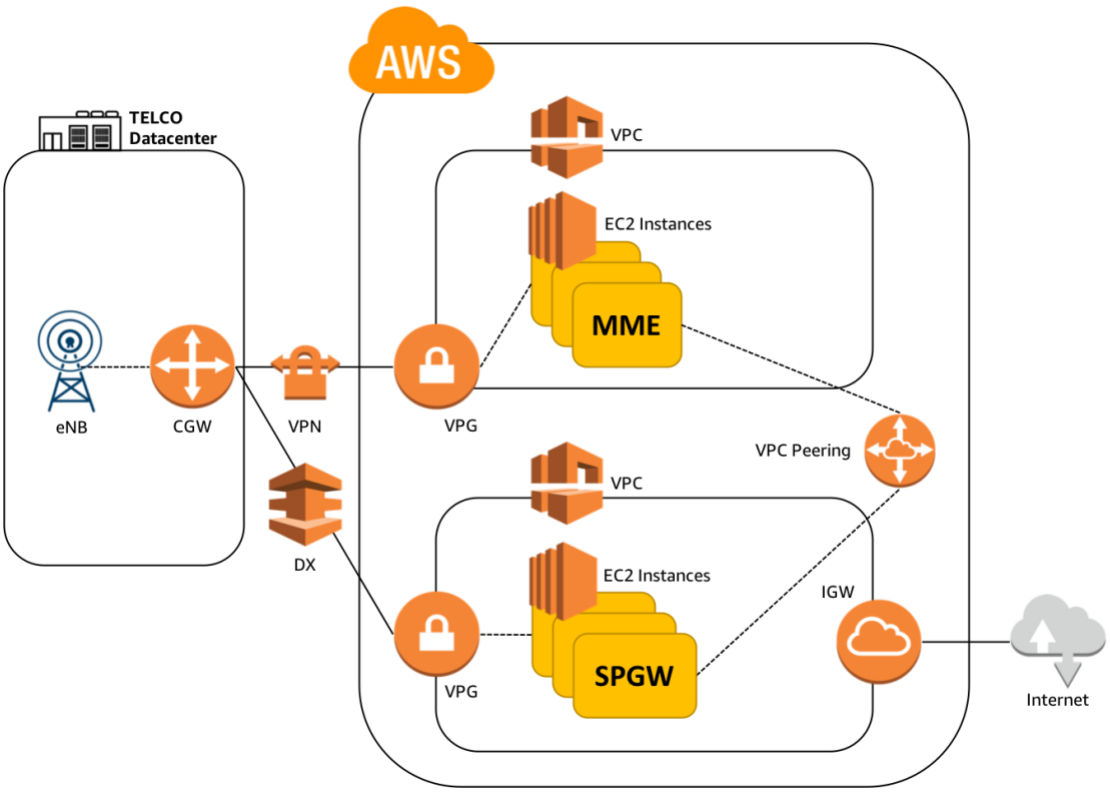
where carrier-grade NAT is already deployed. Option D is less cost effective, but it is useful when a mobile service provider wants to use its own internet point-of-presence (PoP) and local services in their SGi network at the edge data center with CUPS architecture. In other cases, such as options A and B, vSPGW should be able to connect to the internet through the AWS internet gateway (IGW).

The following are some possible deployment models of vEPC on AWS.

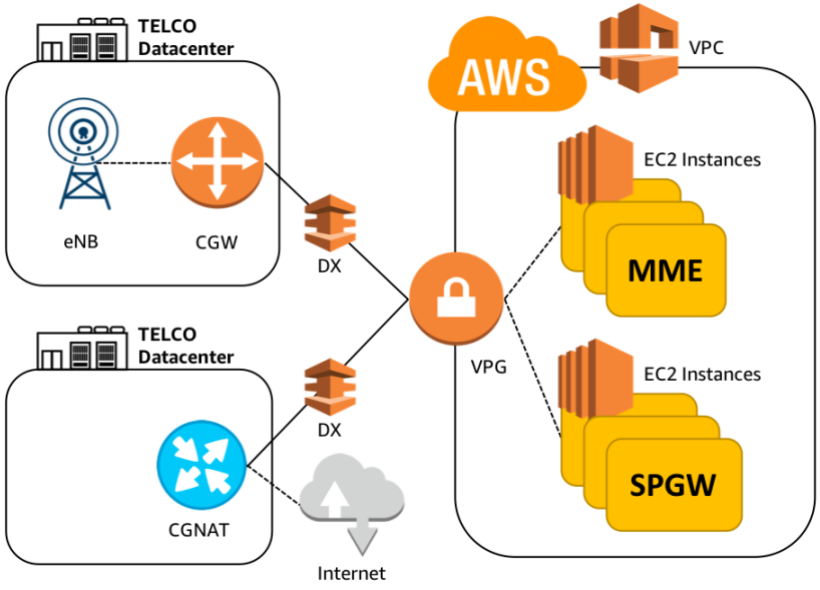
Figure 3 – vEPC on AWS Deployment Models



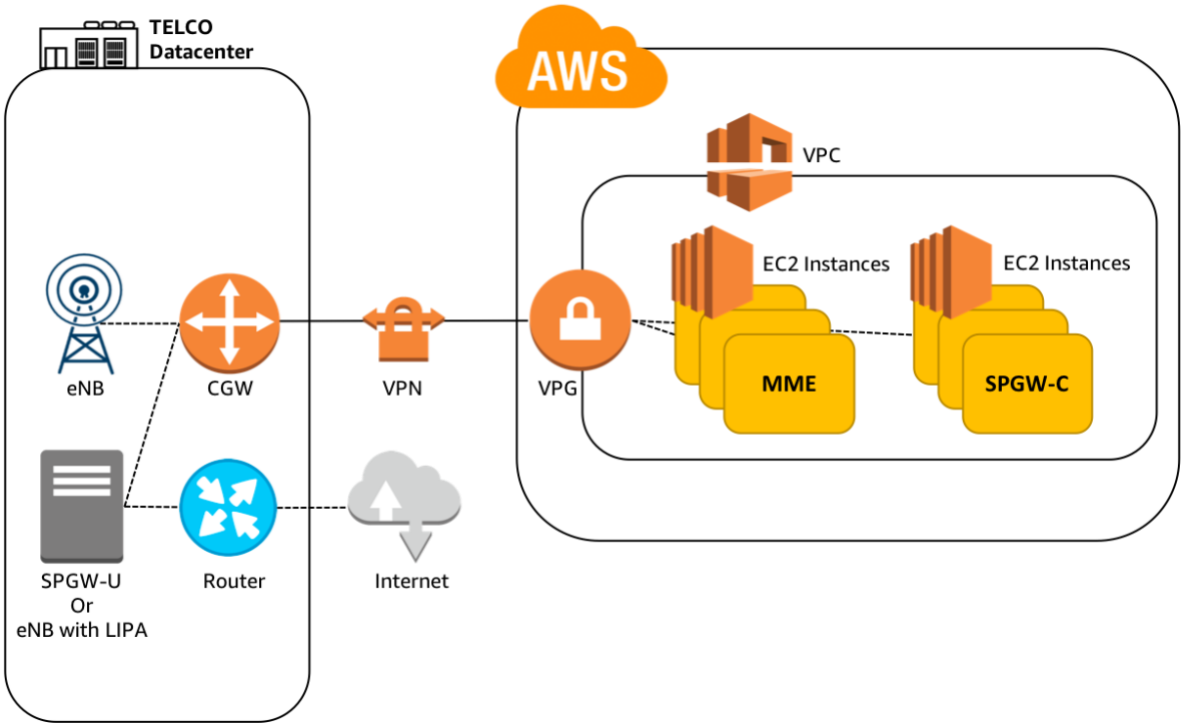
Option A – MME and SPGW in one VPC



Option B – MME and SPGW in separate VPCs with VPC peering



Option C – SGi through telecom data center



Option D – CUPS with Hybrid Network

Building Blocks for Carrier-Grade EPC on AWS

AWS has a broad set of building blocks that mobile service providers can use to implement vEPC on AWS. You can use those blocks to build OpenStack based NFV for vEPC with all the required functions for the telecom industry's demand on VNF.

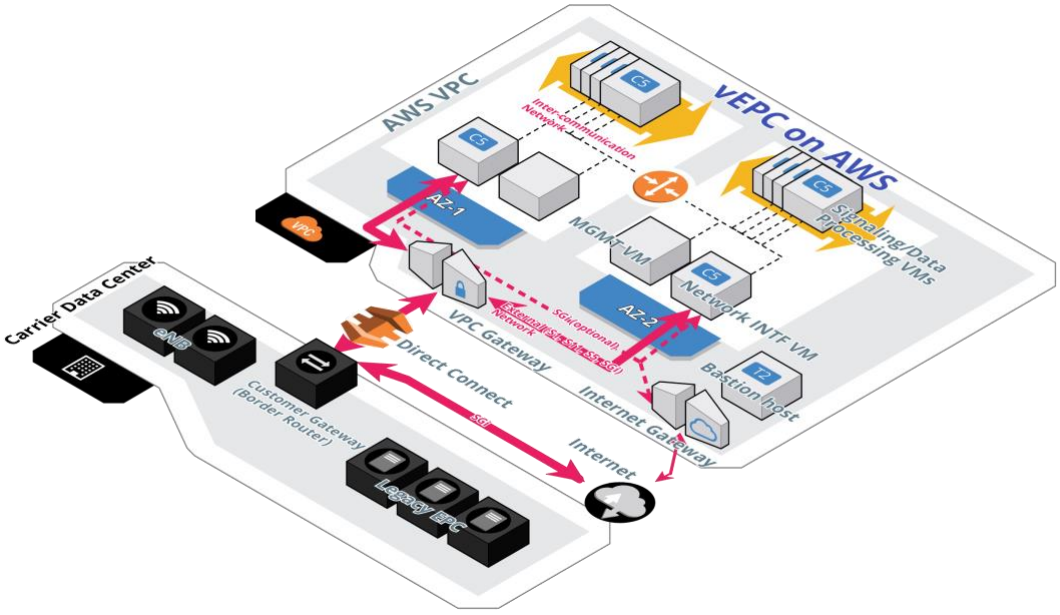


Figure 4 – Conceptual vEPC diagram on AWS

Availability Zone

The AWS Cloud infrastructure is built around Regions and Availability Zones[5]. Availability Zones consist of one or more discrete data centers—each with redundant power, networking, and connectivity—housed in separate facilities. These Availability Zones offer the ability to run production applications and databases that are more highly available than would be possible from a single data center. The distance between zones is designed to provide enough isolation from outage, while still facilitating a guaranteed level of latency between zones . It is essential to have a Multi-AZ deployment for one vEPC VNF to make sure there is high availability to meet the service level agreement (SLA) of the telecom industry.

Placement Groups

Similar to the Affinity and Anti-Affinity Groups in OpenStack, AWS has *placement groups* [6]. There are two placement group strategies that are relevant for this solution: cluster and spread. Cluster placement groups can provide low latency and maximum bandwidth, while spread placement groups can reduce the risk of simultaneous failures that might occur when instances share the same underlying hardware. A spread placement group can span multiple Availability Zones, and you can have a maximum of seven running instances per Availability Zone in each group.

VPC, Subnet, and Security Group

Virtual Private Cloud (VPC) is a virtual data center that AWS users can create to run VNFs in a logically isolated environment. This provides complete control over your virtual networking environment, including the ability to select your IP address range, create subnets, and configure route tables, network gateways, security groups, and the access control list (ACL). Because each VNFC in your vEPC VNF or instance must typically have multiple interfaces and multiple subnets for each different purpose, such as operation and management (OAM), internal communication, and external signaling or user plane data, when you design the internet-facing, subnet design of your VPC, make sure to consider each role of the subnet and the required number of IP addresses. Inside of the VPC, you can configure your security groups and network ACL with various levels and methods of security protection. A security group acts as a virtual firewall for instances with VNFCs that control inbound and outbound traffic. A security group is applied at the instance level, and network ACL is applied at the subnet level[7].

Amazon Elastic Compute Cloud & Elastic Network Adaptor

AWS Elastic Compute Cloud (Amazon EC2) instances offer a variety of implementation options. When you host VNF on AWS, you can create an Amazon EC2 instance to replace an on-premises Virtual Machine (VM) or VNFC with a guest OS installed. Among the various types of Amazon EC2 instances, it is important for mobile service operators or network equipment vendors to choose the type of Amazon EC2 instance that is cost effective and meets the following requirements of telecom VNF.

- **Support for single root I/O virtualization (SR-IOV) data plane development kit (DPDK)**

  The DPDK is a set of data plane libraries and network interface controller drivers that makes fast-path processing of data packets available. In AWS, this is provided by Elastic Network Adaptor (ENA)[8]. Various Amazon EC2 instance families, such as C3, C4, C5, D2, I2, R3, and M4, support the Intel 82599 VF driver or ENA driver that provides a maximum of 25Gbps to the instance, depending on its size.

- **Support for non-uniform memory access (NUMA)**

  With the NUMA design, a cluster of microprocessors in a multiprocessing system are configured to share memory locally. Because this design improves performance and enables expansion of the system, most VNF vendors apply NUMA to run their VNF in an on-premises environment. If VNF is designed to use Amazon EC2 instances that have access to more than one physical processor on the host system (typically an instance type larger than .8xlarge), you can have access to the underlying NUMA topology. This enables you to architect and operate your VNF with the best memory access performance.

- **Support for huge pages**

  Huge pages are a mechanism that allow the Linux kernel to use the multiple page size capabilities of modern hardware architectures. Based on the guest OS on each VM, huge pages are also configurable in most Amazon EC2 instances.

VPC Networking: Virtual Private Gateway, Direct Connect, VPC Peering, AWS Route 53, and Transit Gateway

When your vEPC is hosted on AWS, it should be able to connect with external network elements, such as eNB, in a traditional data center. Amazon VPC provides the option to create an IPsec VPN connection between remote customer networks and Amazon VPC over the internet. A virtual private gateway (VPG) is the endpoint of the IPSec connection at the AWS end of the VPN tunnel that is managed by AWS. When you create a virtual private gateway, you can use either static routing or border gateway protocol (BGP) as the dynamic routing protocol to send the IP address of the VPC to the on-premises site. Another method to make a connection between an AWS VPC and a data center is AWS Direct Connect (DX), which enables you to establish a dedicated network connection between your network and an AWS DX location with a virtual MPLS network. If you use industry standard 802.1q VLANs, you can partition this dedicated connection into multiple virtual interfaces.

If you deploy EPC on multiple VPCs, you can introduce VPC Peering to provide a connection between two VPCs. Because traffic using inter-region VPC Peering always stays on the global AWS infrastructure and never traverses the public internet, threat vectors, such as DDoS attacks, are reduced.

Amazon Route 53 is an AWS managed Domain Name Service (DNS) that is highly available and scalable, and also provides various records types, such as Name Authority Pointer (NAPTR), SRV, and A/AAAA record types. Route 53 supports a Private Hosted Zone that can be exploited for FQDN resolution inside of the VPC on which the EPC is hosted.

The last optional component of VPC networking for vEPC implementation is a transit gateway, which can bring sophisticated routing capability for the traffic to the VPC. AWS Transit Gateway is designed to provide centralized routing policies across VPCs and the on-premises datacenter, while providing horizontally scalable capability and simplified management of a complex routing environment.

Elastic IP Address

An Elastic IP address is a static, public IPv4 address designed for dynamic cloud computing in AWS that is similar to the concept of floating IPs in OpenStack. It can be associated with any instance or network interface for any VPC in one account. This publicly accessible IP address is assigned to the VNF management interface and to the SGi interface for IP address of the user equipment (UE) session. EIP can be re-associated to another instance with a single API call step. At the time of writing, EIP only supports IP version 4 (IPv4).

Auto Scaling Group

An Auto Scaling group contains a collection of Amazon EC2 instances that share similar characteristics, and are treated as a logical grouping for the purposes of instance scaling and management. For example, if a single application operates across multiple Amazon EC2 instances, you might have to increase the number of instances in that group to improve the performance of the application, or decrease the number of instances to reduce costs when demand is low. An Auto Scaling group can be used to scale the number of instances automatically based on the criteria that you specify, or to maintain a fixed number of instances if an instance becomes unhealthy. This automatic scaling and maintaining of the number of instances in an Auto Scaling group is the core functionality of the Amazon EC2 Auto Scaling service. For vEPC to support horizontal scaling, the call processing VM and data processing VM should both be included in the Auto Scaling group.

AWS CloudFormation

AWS CloudFormation is an orchestration tool for instances. Similar to Heat in OpenStack, AWS CloudFormation enables mobile service providers to manage infrastructure as code using simple JSON or YAML formatted text-based templates to define resources, mappings, parameters, and output on AWS. With AWS CloudFormation, related resources are managed as a single unit, known as a *stack*. All the resources in a stack are defined by the stack's AWS CloudFormation template. These stacks can be controlled and managed with the AWS CloudFormation console, an API, or the AWS CLI. If the resources running in a stack need to be replaced, the stack must be updated. Before you make changes to your VNF resources, make sure to create a change set, which is a summary of the proposed changes. Change sets enable you to see how changes might impact the resources running in your stack , especially for critical resources, before you implement them.

Best Practices for Architecting vEPC on AWS

When you develop your architecture for vEPC on AWS, it is important to make sure you follow best practices, which include security concerns, Amazon EC2 optimization and performance, high availability, scalability, networking, and orchestration and automation.

The following diagram shows an example of a reference architecture that follows the best practices for architecting vEPC on AWS.

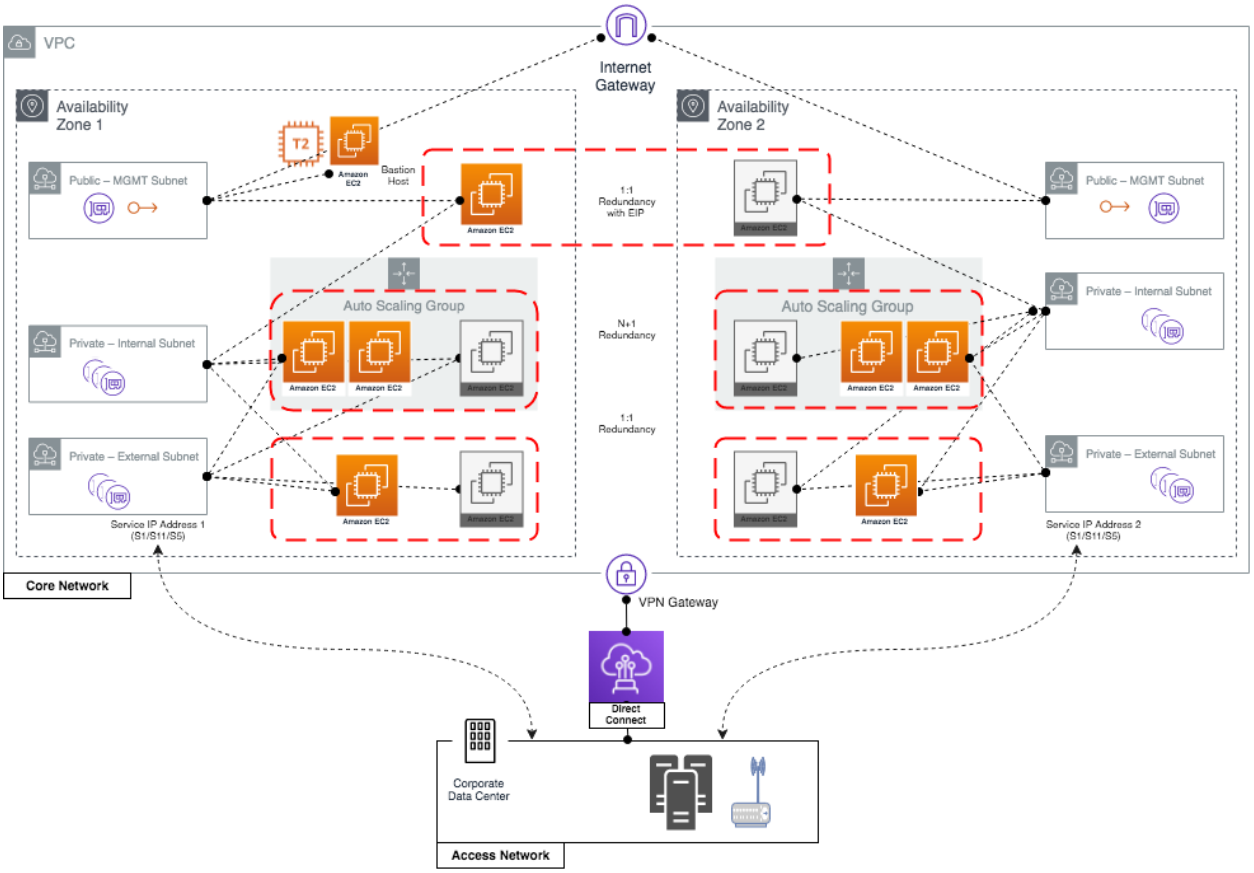


Figure 5 – Example reference architecture of best practices for vEPC on AWS

Security Considerations

Data Protection in Transit

Like other services and applications running on AWS, security is always the highest priority for the telecom VNF. With telecom VNF, when you design your vEPC on AWS, you should consider these two security aspects: Customer Proprietary Network Information (CPNI) protection, and securing access to VNFCs and VNF with a firewall.

For CPNI protection, all signaling interfaces that can carry an international mobile subscriber identity (IMSI) and a Mobile Station International Subscriber Directory Number (MSISDN), such as S1, S11, and Gx interfaces, should use an IPsec VPN for connections to the VPC and the on-premises data center. To implement IPsec for those interfaces, you can use a custom IPsec VPN at the AWS EC2 instance and AWS managed tunnel mode IPSec from VPG.

Secure Access and Intrusion Protection

When you configure secure access to the VNFC and Amazon EC2 instance, make sure to configure the AWS Identity and Access Management (IAM) service as described in AWS Security Best Practices. With the configuration specified in these instructions, access to the Amazon EC2 instance or VNFC is through AWS provided asymmetric key pairs (Amazon EC2 key pairs), which are industry-standard RSA key pairs. In addition to this secure access to each VNFC guest OS, make sure to configure access to the GUI (user interface) with a secure protocol, such as TLS or HTTPS. Make sure to configure security for the VPC and each EC2 instance that will protect them from security threats, such as a firewall.

For example, to enable S1 packets to be transmitted from the eNB to the vMME VNF, create an inbound rule that allows access to SCTP port 36412 with protocol 132 for the vMME VNF in the VPC. Make sure to also open ports 2123 and 2152 for the GTP protocol over UDP, for the MME and SPGW in the VPC. For stronger security for the vEPC, configure ACL to allow SCTP port 36412 (for S1AP S1-MME), UDP port 2123/2152 (for GTP S1U, S11, and S5) and port 8805 (for PFCP Sx), and SCTP or TCP port 3386 (for Diameter S6a, Gx, Gy, and Rf).

Amazon EC2 Optimization and Performance

The type of Amazon EC2 instance you select is important for both cost-effectiveness and performance, especially for telecom VNF deployments, because VNF has multiple VNFC groups and VNFCs in one VNFC group. Usually, telecom VNF deployments such as vEPC include three types of VNFC groups:

- Type-1 – OAM management VNFC group
- Type-2 – Network interfacing and load balancing VNFC group
- Type-3 – Call or signaling processing VNFC group

Type-1 usually requires less computing power than the others, but it also usually requires more memory and disk space. Type-2 and Type-3 require high computing performance and high bandwidth network interfaces to provide enough network per unit capacity and total capacity of VNF. Because of this, it often requires SR-IOV DPDK, huge page, and NUMA control which are already available building blocks in AWS. If you choose the most recent generation of the C and M families, you must install the ENA pull mode driver (PMD) on the guest OS of the instance. AWS recently announced a new compute-optimized and network-optimized C5n instance that can outperform other instance types, with available peak bandwidth up to 100Gbps. The various types and sizes of Amazon EC2 instances offer more opportunities for mobile service providers and network equipment vendors to create vEPCs with an optimal cost point.

High Availability

Because telecom providers require five-nines of availability (99.999%) against a single point of failure (SPOF), most telecom VNFs, such as vEPC, typically follow the architecture shown in *Figure 6*, which is inherited from their legacy, dedicated hardware-based generation environment. In that type of hardware-based environment, to ensure high-availability and fault tolerance, each layer and cluster with OAM, Call/Packet Processing, and Load balancer must have a 1:1, all-active, M:N and N:1 redundancy configuration and structure. vEPC on AWS should follow the same direction and design principles to have enough high availability to meet the telecom Core Network requirements.

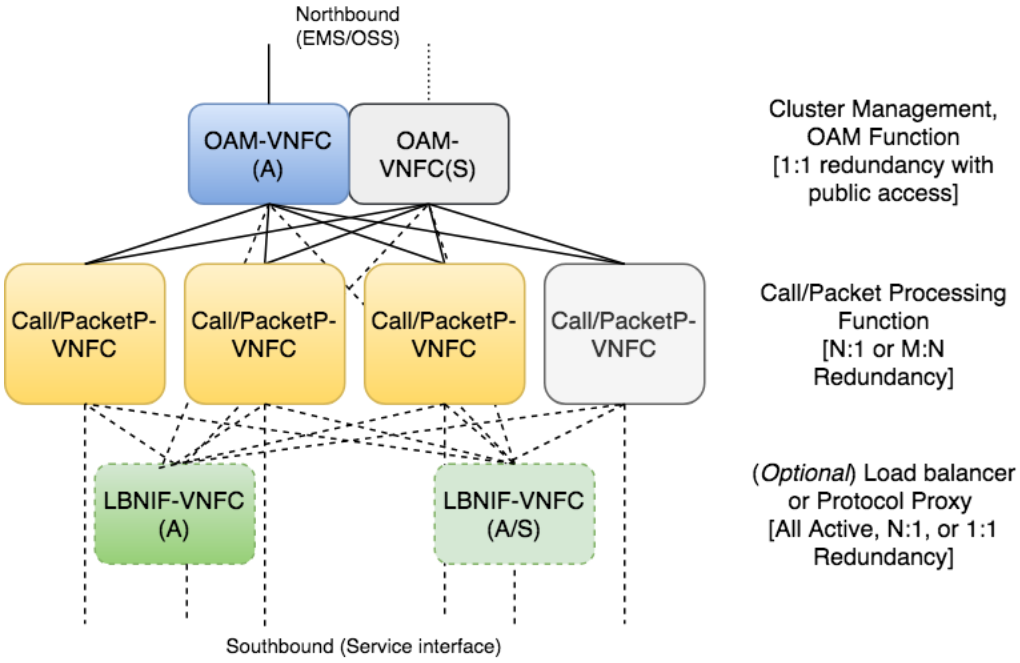


Figure 6 – Telecom VNF architecture for high availability

Floating IP Address Implementation (IP Address Failover)

For VNFC-level high availability, we recommend that you implement a floating IP address design (EIP for public IP addresses and secondary IP addresses for private IP addresses in AWS), as described in Real-Time Communication on AWS. Specifically, you assign an EIP or a static secondary private IP address to the active instance. You then continuously monitor the instance either with CloudWatch or with other script tools, so the IP address can be switched over to another instance if a sudden instance or network failure occurs, as shown in *Figure 7*. When a failure is detected, the IP address is automatically moved to another instance through Amazon EC2 API call.

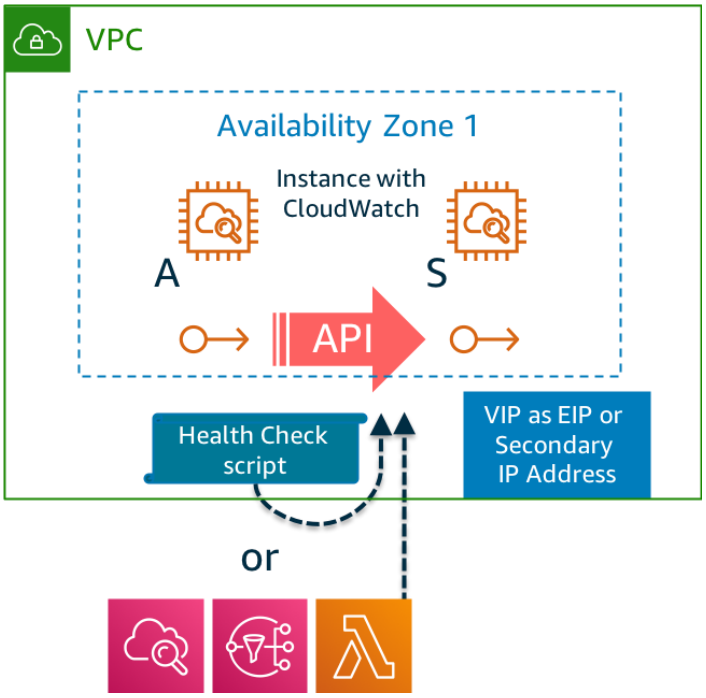


Figure 7 – Floating IP address implementation in AWS

In addition to this floating IP address implementation in a cloud environment, AWS Transit Gateway is another option for a floating IP address implementation. Because AWS Transit Gateway enables users to create static routing tables for virtual IP addresses, the *delete-route* and *create-route* Amazon EC2 API call can make route updates for the virtual IP addresses and reroute them to the new instance. One example of this implementation is shown in *Figure 8*. You can use this floating IP address implementation method when the VNF management IP address must remain in your private network to work with the carrier's existing datacenter OSS network through the VPN or DX.

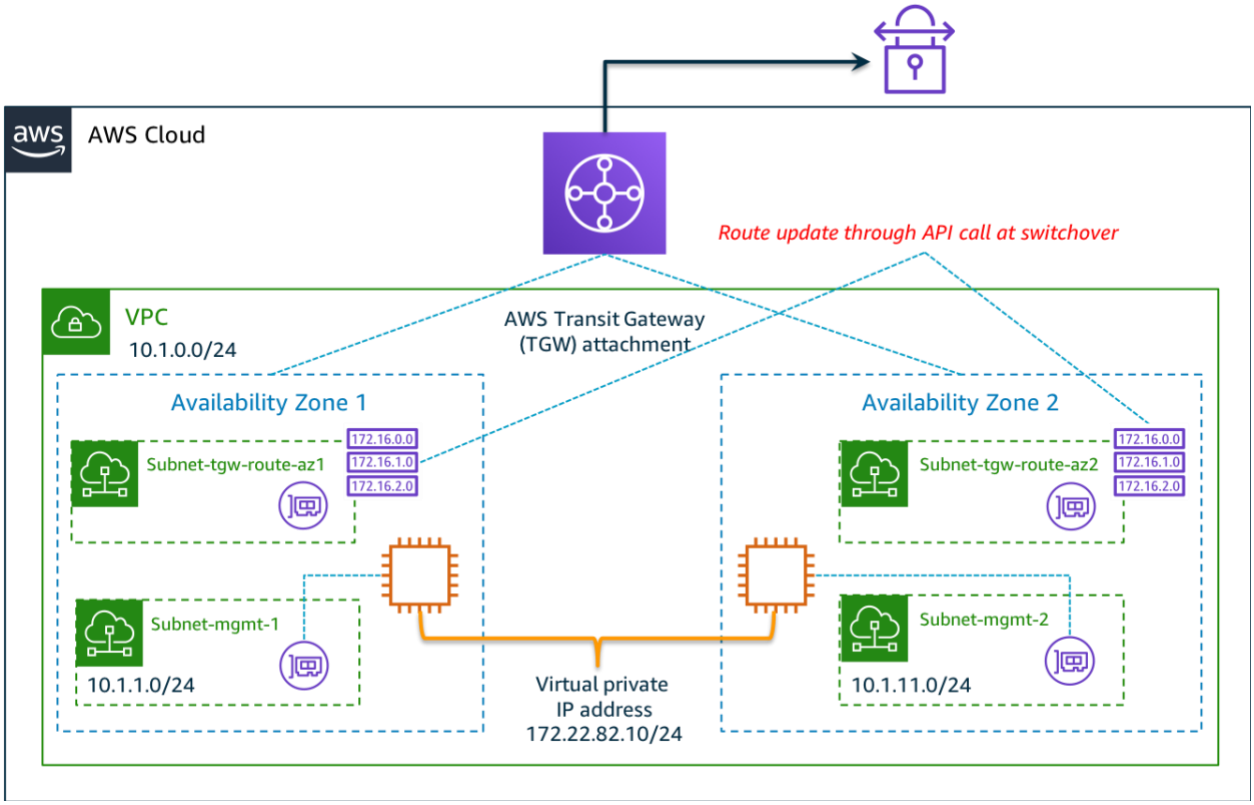


Figure 8 – Floating IP address implementation with AWS Transit Gateway (TGW)

Placement Group and Multi-AZ Deployment

To increase service availability while reducing the possibility of multiple instance failures caused by a single hardware component in one Availability Zone, you can use *spread* placement groups in the VNFC group design. For example, you can bind both the call processing VNFC group and the load balancing VNFC group using a spread placement group. Because spread placement groups currently have a limit of a maximum of seven instances, if the number of instances in one VNFC group exceed seven, then you must create another VNFC group for scalability with a placement group. To ensure high-availability of the service, a Multi-AZ deployment (as show in *Figure 5*) is the recommended best practice of all the AWS workload design types. In this example architecture, vEPC (either MME or SPGW) would have at least two service IP addresses (such as S1 or S11/S5 IP addresses) in the total VNF (each per Availability Zone). Additionally, the static configuration at each interworking peer (for example, eNB) or DNS for the Core Network fully qualified domain name (FQDN) resolution should be pre-provisioned with these two IP addresses. For more information about the details of DNS resolution in the LTE network, see the 3GPP standards[4, 9].

Geo-Redundancy

For geo-redundancy in your mobile service vEPC, you can deploy vEPC on AWS in multiple Regions and enable disaster recovery mode in the vEPC application. For vSPGW, whether failure occurs in one Region or the entire VNF, vSPGW service can failover to another VNF in another Region, using DNS update for the tracking area code (TAC) and access point name (APN) FQDN with the newly activated VNF IP address. You can also automate this process with VNF monitoring and the relevant API calls in AWS. For vMME geo-redundancy, set up your MME pooling configuration with both active and disaster recovery mode vMMEs located in one MME pool so they can be used for eNB failover for the entire vMME VNF.

Scalability

To take advantage of the cloud environment, vEPC on AWS must have horizontal scalability for the core call/data processing VNFC group. In the example shown in *Figure 5,* an all-active VNFC group (such as the call processing VNFC) is in an Auto Scaling group. With this architecture, the VNFC can increase and decrease automatically based on the given KPI criteria, such as CPU load and the network in/out metric. In addition to the inherent scalability of the instance layer, VNFC scaling can be triggered from the application layer using additional triggers, such as the number of subscribers, sessions, and bearers, or the measured signaling packet transaction rate (TPS) or measured user traffic throughput using an Amazon EC2 API call. In addition, if that instance has more than one interface, which is typical for telecom VNF, you can leverage the Amazon EC2 instance-launch lifecycle action and Lambda implementation to automatically bind additional interfaces to the instance[10].

Networking

VPN Connection with a Legacy Datacenter

vEPC on AWS works with eNBs or other network elements in an on-premises environment, such as a legacy telecom regional data center (RDC), as shown in *Figure 3*. Because telecom VNF is sensitive to packet loss and latency, particularly for signaling and control packets, and also always requires an extremely high level of security for the network because of subscriber credential information (such as IMSI), it is important to build a reliable and secure VPN network between the on-premises site and the AWS VPC where the vEPC is hosted. For example, even though a VPN through the VPG over the internet provides a secure path between the telecom data center and the AWS VPC, it can be challenging to use S1-MME and S1U interfaces

because they use the internet connection without any guarantee of quality of service (QoS). As a best practice, we highly recommended using AWS Direct Connect (DX) for connectivity between the telecom DC and the AWS VPC. Using DX, you can build a private connection between the AWS Cloud and eNB data center. This private connection can reduce network costs, increase bandwidth throughput, and provide a more consistent network experience than internet-based connections. To meet high-availability for VPN connections, either through simple VPG or DX, we recommend that you implement two redundant VPN links between your on-premises environment and the AWS VPC. In a practical implementation of DX, because the telecom data center requires support for a Virtual Routing and Forwarding (VRF) domain for each logically segregated network, DX must have the same number of Virtual Interfaces (VIFs) with the same VLAN tag configuration.

Inside of the AWS VPC, if MME and SPGW need to be hosted on two different VPCs, as shown in *Figure 3*, Option B, make sure to connect the two VPCs. A VPC peering connection is a networking connection between two VPCs that enables routing using the private IP addresses of each VPC, as though they were in the same network. You can create VPC peering connections between your VPCs or with a VPC in another AWS account. Inter-region VPC peering is also supported.

SGi Routing

Another networking concern for mobile networks is the security of the SGi interface. Make sure to configure the vPGW SGi interface to route downlink user traffic to vPGW. In a 3GPP architecture, PGW allocates IP addresses to mobile subscribers for each PDN session. It is also the entry point for downlink traffic from the internet to mobile subscribers. To make sure that mobile subscribers receive this traffic, PGW must be able to receive packets from the internet through the SGi interface and route them to the IP address of the mobile user. If vPGW on AWS has NAT built in, and one of the VNFC network interfaces is configured with NAT, then EIP for UEIP can be assigned to this instance with the outbound NAT IP address. You can then connect this instance to IGW with a subnet and routing table association. If NAT isn't enabled for vPGW, then the NAT instance or NAT Gateway with EIP can be located with configuring similar network connection to IGW so as to make SGi routing to be working properly for incoming traffic (downlink user traffic) to the mobile subscriber.

If internet traffic through the SGi interface must always go through the telecom carrier's local datacenter, you can specify that traffic through the SGi interface must use a VPN or DX connection, as described for the S1 interface. This can be a requirement for UEIP governance or regulatory issues. In this specific case, because the downlink data UEIP

passes through the VPG as virtual private IP addresses, you must configure your network for virtual IP address routing, with either an overlay network or using AWS Transit Gateway (TGW).

Orchestration & Automation

For operational excellence and ease of maintenance, we recommend you use AWS CloudFormation templates with your VNF design because they provide the ability to easily reproduce your VNF for your DevOps and DR environments. AWS CloudFormation provisions resources in a safe, repeatable, and automated manner, which enables operators to build and rebuild vEPCs, without having to manually perform actions or write custom scripts.

In most *lift-and-shift* cases of existing vEPC in NFV environments to AWS, because most telecom VNFs already have an OpenStack Heat template or their own VNFD (VNF descriptor), the AWS CloudFormation template can be designed and developed so it is similar to the Heat or VNFD development. In the AWS CloudFormation template for your vEPC, make sure to correctly define and map all components, including EC2 instance, EBS volumes, VPC, subnets, route tables, and network interfaces. For the VNFC, you must bootstrap the guest OS and the application. This can be configured with config-drive or using metadata, if you use Heat in the NFV environment. In AWS, you can do this either with user data provided as a property of the AWS::EC2::Instance or with an AWS CloudFormation Helper Script, such as a metadata key in cfn-init[11]. When you use an AWS CloudFormation template, you can execute the template through the AWS CloudFormation console, or through a vendor-specific VNF manager (S-VNFM). If you choose a S-VNFM, you must host it on AWS and connect to it with an AWS CloudFormation API call.

Conclusion

This paper outlines the best practices and benefits of architecting vEPC on AWS. With vEPC on AWS, mobile operators can easily deploy a new core network or can improve their services continuously with the cost and operational benefit of the AWS Cloud. Because VNF on AWS can maximize the benefit of NFV of the telecom industry, telecom VNF on AWS is likely to become more important in future mobile networks. As the telecom industry develops 5G technology, this architecture is likely to become even more relevant, because 5G mobile networks are evolving to include CUPS, stateless architecture, microservices, DevOps principles, and service-based architecture, which all work well with existing AWS services. Hosting vEPC on AWS will enable you to quickly and easily upgrade your 4G network to a 5G network in the near future.

Contributors

Contributors to this document include:

- Young Jung, Ph.D., Partner Solutions Architect, Global Telecom Strategic Alliance, Amazon Web Services
- Tipu Qureshi, Principal Engineer, AWS Premium Support, Amazon Web Services
- Srinivas Kappla, Senior Product Architect, Affirmed Networks
- Kent Nickell, Senior Product Architect, Affirmed Networks

About Affirmed Networks

In 2010, Affirmed Networks introduced a visionary cloud-native network that is redefining the future of the mobile industry. Our virtualized Evolved Packet Core (vEPC) enables communication service providers to scale their networks to scale to the insatiable demands for mobile services, while dramatically reducing CAPEX and OPEX. Using Affirmed technology, many of the world's most forward-thinking service providers are transforming their networks and businesses every day, creating and delivering new generations of differentiated web scale mobile services at unprecedented speed.

Document Revisions

Date	Description
November 2019	Minor edits for clarity.
May 2019	First publication

Notes

1 Dell'Oro Group Report, http://www.delloro.com/news/evolved-packet-core-market-revenue-grew-four-percent-quarter-quarter-2q-2018

2 H. Hawilo, A. Shami, M. Mirahmadi and R. Asal, "NFV: state of the art, challenges, and implementation in next generation mobile networks (vEPC)," in IEEE Network, vol. 28, no. 6, pp. 18-26, Nov.-Dec. 2014.

3 B. Han, V. Gopalakrishnan, L. Ji and S. Lee, "Network function virtualization: Challenges and opportunities for innovations," in IEEE Communications Magazine, vol. 53, nfo. 2, pp. 90-97, Feb. 2015.

4 3GPP. Technical Specification Group Services and System Aspects; General Packet Radio Service (GPRS) enhancements for Evolved Universal Terrestrial Radio Access Network (E-UTRAN) access; TS 23.401, 3rd Generation Partnership Project (3GPP).

5 AWS Documentation, https://docs.aws.amazon.com/AWSEC2/latest/UserGuide/using-regions-availability-zones.html

6 AWS Documentation, https://docs.aws.amazon.com/AWSEC2/latest/UserGuide/placement-groups.html

7 AWS Documentation, https://docs.aws.amazon.com/vpc/latest/userguide/VPC_Security.html

8 AWS Documentation, https://docs.aws.amazon.com/AWSEC2/latest/UserGuide/enhanced-networking-ena.html

9 3GPP. 3rd Generation Partnership Project; Technical Specification Group Core Network and Terminals; Domain Name System Procedures; TS 29.303, 3rd Generation Partnership Project (3GPP).

10 How do I automatically attach a second ENI to an instance launched through Auto Scaling, AWS Knowledge-center, https://aws.amazon.com/premiumsupport/knowledge-center/attach-second-eni-auto-scaling/

11 AWS Documentation, https://s3.amazonaws.com/cloudformation-examples/BoostrappingApplicationsWithAWSCloudFormation.pdf