Netcracker Active Resource Inventory on AWS

November 4, 2021

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.

Introduction

Communication Service Providers (CSPs) regard business agility and operational efficiency as critical success factors for securing the win in the platform economy race. By transforming into Digital Service Providers (DSPs), they aim to speed up time-to-market, reduce operational costs, and maximize the benefits of 5G, cloud and emerging technologies.

To support their cloud transformation, CSPs can take advantage of business-driven workload placement: deploying and managing their virtual and cloud-native network functions (xNFs) in regional data centers, on specialized hardware, on-premises cloud extension (using AWS Outposts) and on AWS infrastructure to improve agility and optimize efficiency.

This flexible workload placement extends further with the introduction of 5G Network Slicing (3GPP TS 23.501, 23.502, 28.530). This allows virtual and independent logical networks to run on the same underlying network infrastructure, enabling resources to be dedicated - or shared - and optimized for a specific service or customer segment.

The Operational Support Systems (OSS) used by CSPs must adapt to enable this highly dynamic network behavior by automating the design, provisioning and lifecycle management of digital services, network slices and xNF resources. A cornerstone of OSS evolution is an inventory management (IM) platform that continuously adapts to an evolving, malleable network. This is also known as an **active inventory system**.

An active inventory system serves as a single point of truth by storing both current and planned network states. Based on this knowledge, it enables resources to be reserved for service provisioning, supports the logic required to identify current and future faulty states and enables the optimization of resource utilization – proactively and in real-time and proactively. The inventory system evolves from a storage-like platform to become an active hub that is integral to all network operations processes.

A strategic partnership between Netcracker and AWS has allowed joint customers of the two companies to be among the first ones to enjoy the benefits of evolved OSS and active resource inventory. Netcracker's multi-vendor and cloud-native Active Inventory System on AWS enables CSPs to transition from an on-premises, centralized, monolith platform to a real-time distributed architecture.

It leverages AWS functionalities to address CSP inventory management challenges by providing a scalable, reliable and highly available infrastructure, while facilitating the flexible deployment and operation of dynamic networks.

This white paper explores how AWS Services and tools, coupled with Netcracker's Active Resource Inventory platform, **can help CSPs transform to a modern cloud-based inventory management solution to enable dynamic digital services**.

Why is Active Resource Inventory Management Critical?

In today's digital society, both consumers and businesses expect services that are on-demand and highly personalized. CSPs are responding to this demand with dynamic networks that are agile, automated, service-oriented, and easily scalable, leveraging 5G and cloud technologies.

To operate it efficiently, service providers need real-time visibility of the available resources with a modern active resource inventory system. They need a consistent, accurate source of data for the entire Business Support Systems (BSS)/ Operational Support Systems (OSS) and enterprise application ecosystem. It is a critical component for business success.

Conventional resource inventory systems are not fit to address this challenge. The manual data entry, daily updates and limited scalability of these conventional systems prevent CSPs from enjoying the benefits of cloud networks. Inventory management needs to become real-time, including network discovery and status updates, enabling federated visibility and optimized utilization. It needs to become an ACTIVE resource inventory, able to assign, reassign, monitor and manage hybrid resources on-demand.

The digital transformation of networks and services will take some time. While legacy and new networks coexist, service providers will face a challenge of optimal resource utilization. Active resource inventory, being a critical part of most operational processes, can help smooth the transition. For the hybrid network, it needs to present a multi-layer, end-to-end view and keep it up-to-date in line with technology or domain specifics. For example, it can skip real-time status updates for the physical equipment to avoid unnecessary event flow, but optimize the placement of xNF workloads across AWS cloud platforms in real time.

Another challenge an active resource inventory needs to address is the growing agility and variety of the xNF market: the list of vendors keeps growing, releases happen more frequently, standards mature step-by-step. The ease of change management thus becomes an essential success factor.

Microservices, containers, DevOPs, CI/CD, standardized interfaces and descriptor formats are generally considered the best practices. Besides networks, the digital agility impacts the supporting IT infrastructure.

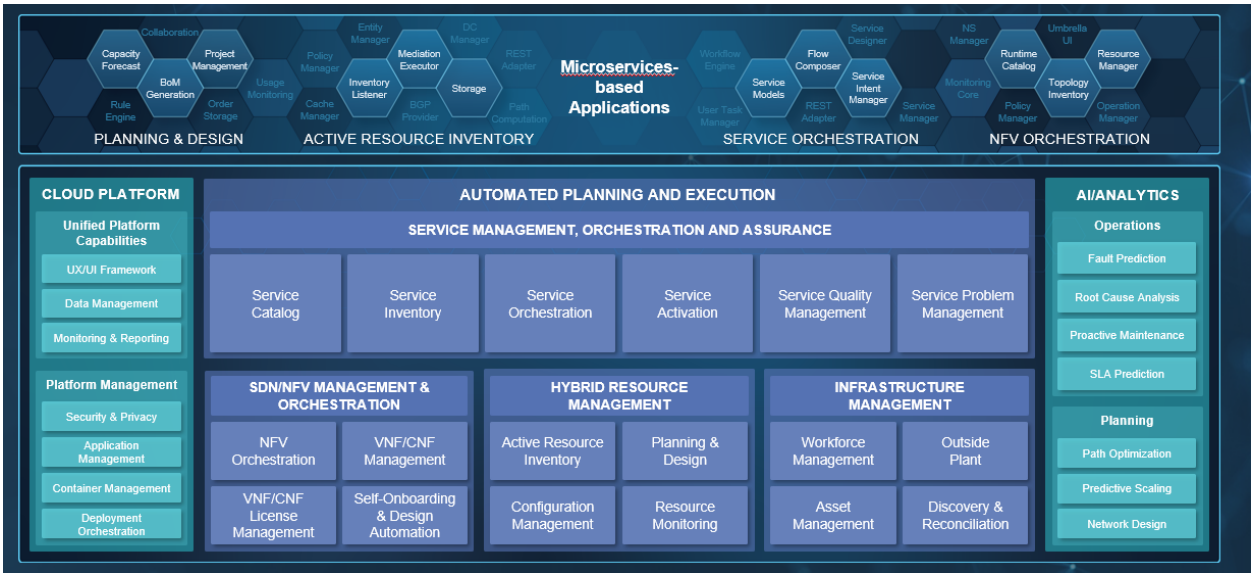
To make the best use of assets and guarantee service quality, providers try to make their edge more intelligent. The resulting multi-hybrid-cloud environment allows the placement of xNFs (network functions) where they are needed most.

To minimize latency and avoid bottlenecks, an active resource inventory system needs to work in a similar manner: extend compute and storage infrastructure as close to the resource as possible and it also needs to support distributed deployment in a variety of setups.

Transition to XaaS (anything as a service) adds to the list. It provides benefits of pay-as-you-grow, on-demand scaling and zero operational effort, but on the other hand it sets strict security requirements and asks for OSS vendor to have experience, resources and capabilities for cloud-native operation – on their own or through partnership with a cloud provider.

Introducing Netcracker Digital OSS Portfolio

Netcracker Active Resource Inventory is a key component of its Digital OSS suite, built on more than 25 years' experience of successfully delivering end-to-end service and resource management, enabling service agility and operational efficiency through automation. Netcracker Digital OSS solutions have been implemented and used by customers worldwide to help transition operations from physical to virtualized and cloud environments, embrace greater levels of service innovation and stay competitive in fast-changing markets.



Netcracker Digital OSS

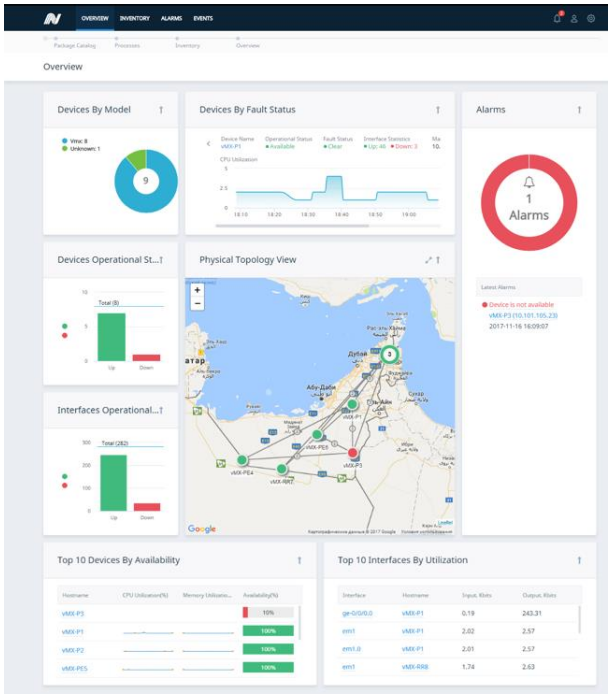
Netcracker's entire Digital OSS suite, including Active Resource Inventory, uses a microservices-based, cloud-native architecture to enable solution flexibility, elasticity and resilience. It provides end-to-end, zero-touch orchestration for traditional and digital services including SDN/NFV, cloud, IoT and 5G.

The solution automates operations across complex cloud and CSP network environments leveraging advanced analytics and AI to optimize resource utilization and improve service quality. Service providers benefit from faster service creation and delivery, improved operational agility and lower TCO.

Netcracker Active Resource Inventory

Netcracker Active Resource Inventory delivers consolidated, real-time visibility of telco multi-level resources and infrastructure, including physical and logical networks, devices and technologies, adjacent numeric resources and asset data, virtual and cloud network functions and slices,as well as telco software workloads running on AWS. It serves as a consistent, accurate source of data for the entire BSS/OSS and enterprise application ecosystem.

The system offers a fully open solution, configurable and technology-agnostic, which allows for the quick introduction and support of new network technologies including 5G New Radio and Core, Open vRAN, ,and SD-WAN. It enables YANG-based data modelling with OpenConfig, IETF standard-based model. *Full explanations of all these terms are in the glossary at the end of this White Paper.*

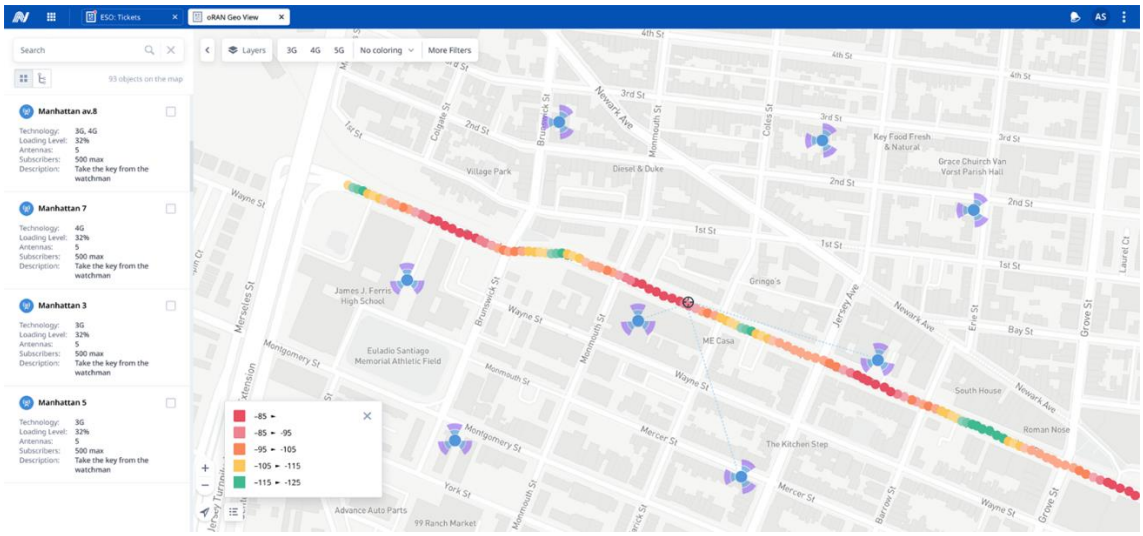


Netcracker Active Resource Inventory Graphical UI Screenshot

Active Resource Inventory provides an end-to-end multi-layer view of physical and logical resource inventory, location management, equipment management, number management, capacity management and IT infrastructure management.

It offers rich graphical UI, enabling friendly views of network topology, properly positioned on a geospatial (GIS) background map, of devices and racks properly positioned on a floor plan, and circuit visualization.It also offers real-time discovery and synchronization with network via support of various dynamic protocols, including BGP-LS, LLDP, NETCONF, CLI. Network workload running on AWS infrastructure are updated dynamically and reflected in the end-to-end topology. *Full explanations of all these terms are in the glossary at the end of this White Paper.*

It can act as a master inventory system across all a provider's assets as well as address the needs of a specific domain, technology or equipment. The module feeds planning, operations, support & readiness (OS&R), fulfillment, assurance, revenue management and ERP processes with precise and truthful data.

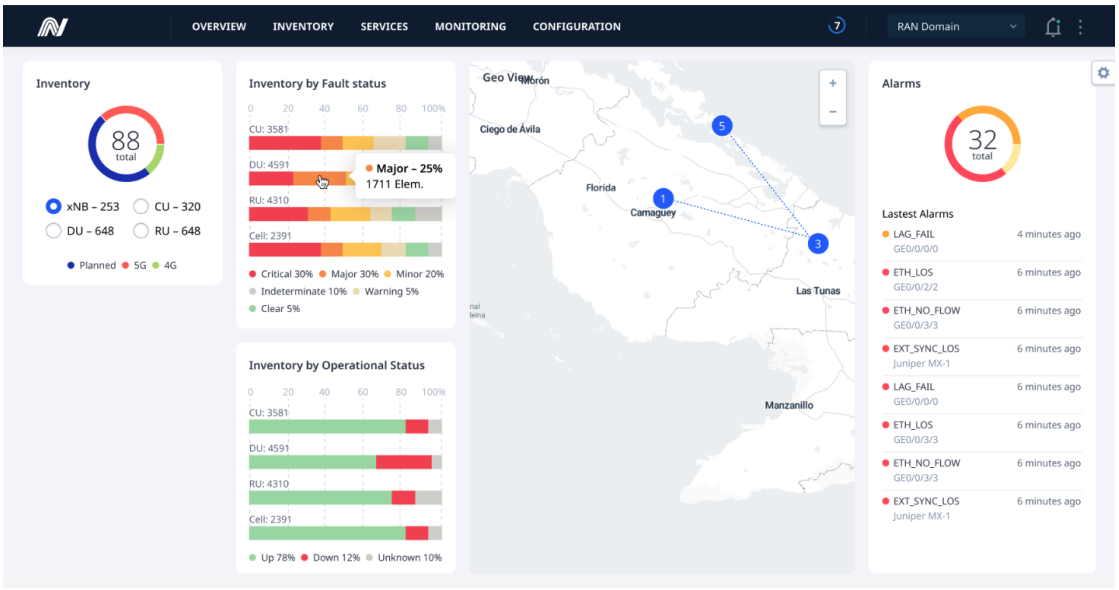


Netcracker Active Resource Inventory Network Topology Screenshot

Netcracker Active Resource Inventory enables each network domain to be self-sufficient and fully automated. Below are highlights of Active Resource Inventory as applied in select domains.

- **Open vRAN domain:** Netcracker Active Resource Inventory enables the smooth adoption of virtual and cloud 5G RAN. It offers up-to-date inventory of hardware, e.g., remote radio heads (RRH) and cell site gateways, and of software components, including RIC, CU, DU, RU, slices. To help rollout new access networks, it consolidates inputs from many sources including RF planning systems for physical radio parameters, vCU/DU EMS for configuration parameters and assurance systems for load performance, usage statistics and quality degradation. The resulting end-to-end view can be visualized in different ways, including real time geo views to highlight the most problematic or business critical RAN zones. *Full explanations of all these terms are in the glossary at the end of this White Paper.*

- **Edge domain:** Netcracker Active Resource Inventory also consolidates information about SD-WAN, SD-LAN, SD-WiFi and Edge Cloud. It serves as an umbrella system with multilayer cross-domain topology and performs cross-domain event correlation. It stores information about the available resources – virtual machines, routers, servers, HW CPEs; L3 overlay topology inventory - interfaces, IP Addresses, VLANs/VXLANs; correlation of events in routed L3 network with on-premises workloads, and xNFs running on AWS; VNF/CNF Inventory – uCPE, running on AWS Outposts and in AWS regions. It is designed to make self-serve service management easy enough for providers to successfully grow in the SMB market and strengthen their role in enterprises. *Full explanations of all these terms are in the glossary at the end of this White Paper.*



Netcracker Active Resource Inventory Geo View Screenshot

- **Core domain:** Netcracker Active Resource Inventory manages topology across virtual, containerized, logical resources, licenses as well as slices at a domain level. It spans 4G, 5G and vIMS. It supports offload of core functions to the edge with multi-cloud deployment capabilities and location management, including NFx workload placement between AWS Outposts and AWS Regions. Active resource inventory solution becomes the key enabler of zero-touch provisioning, orchestration and assurance of Core xNFs and slice subnets. A range of pre-onboarded descriptors for major vendors speeds up the TTM and reduces cost. *Full explanations of all these terms are in the glossary at the end of this White Paper.*

- **Network transport domain:** Netcracker Active Resource Inventory stores (and keeps up-to-date) information on links, interfaces, NEs, numeric resource, DWDM and OTN equipment, L1-L3 topology. It distinguishes between those which are updated in real time for next-gen services like Bandwidth-On-Demand, and those which are static. The southbound integrations can be direct or via EMS/NMS – using REST APIs. The solution allows for automation of daily complex manual and error-prone operations. Customers use it to launch new B2B services and to cut costs with reducing overprovisioning of services. Full explanations of all these terms are in the glossary at the end of this White Paper.
- **Across Network and IT:** Each domain Active Resource Inventory consolidates information about network and IT resources. It acts as a single point of contact for northbound requests to achieve full-service automation. The solution breaks down those requests into segment-specific requests to a fully-integrated AWS solution, CISMs, SDN Controllers and EMS systems. Netcracker Active Resource Inventory supports resource discovery functionality via the southbound REST API/TMF 639 interface for both IT and Telco resources. As the result, service providers can optimize xNF placement and cost of IT infrastructure. Full explanations of all these terms are in the glossary at the end of this White Paper.

Why Netcracker Active Resource Inventory on AWS?

The Netcracker Active Resource Inventory application helps CSPs address the challenge of transitioning to virtualized, cloud and 5G networks, as well as responding to the business demands of a digital customer experience. To maximize value and guarantee agility, a matching, cutting-edge IT infrastructure agility is needed.

By hosting Netcracker's Active Resource Inventory on AWS, CSPs can reduce time to market by accelerating infrastructure procurement and setup processes. They can do so by reusing predefined configurations and instantly available resources.

Netcracker Active Resource Inventory uses AWS flexibility to speed up solution deployment with autoscaling and the capacity of the leading public cloud provider.

Joint global expertise by AWS and Netcracker brings with it best practices for deployment schemes - including deployment scripts, recommended settings and resource utilization.

The “smart tools” provided by AWS use scalability to adjust consumption automatically, based on traffic patterns, thus avoiding complex capacity planning processes.

The swift response to business demand made possible by AWS is critical in cost reduction. From a CAPEX perspective, it helps minimize upfront costs, significantly reduce overprovisioning for peak capacity and avoids far-in-advance pre-provisioning.

The dynamic workload accommodation is highly automated and available as a service, reducing CSP’s complexity and optimizing costs for their IT infrastructure operations: day-to-day operations, spare part management, space availability, connectivity and networking. For those CSPs where cloud transition is still ongoing, AWS and Netcracker support a hybrid model.

Moving the business-critical telecom network data to AWS cloud requires adhereance to strict reliability, security and legislated requirements. Netcracker and AWS have developed, tested and keep enhancing a set of reusable security and integration best practices.

The wide selection of locations ensures convenient access to the service, as well as compliance with limitations related to sensitive data. The end-to-end monitoring and automation environment provides for application availability. Netcracker’s inherent support of evergreen, blue-green and canary deployment schemas offers additional flexibility and simplifies change management.

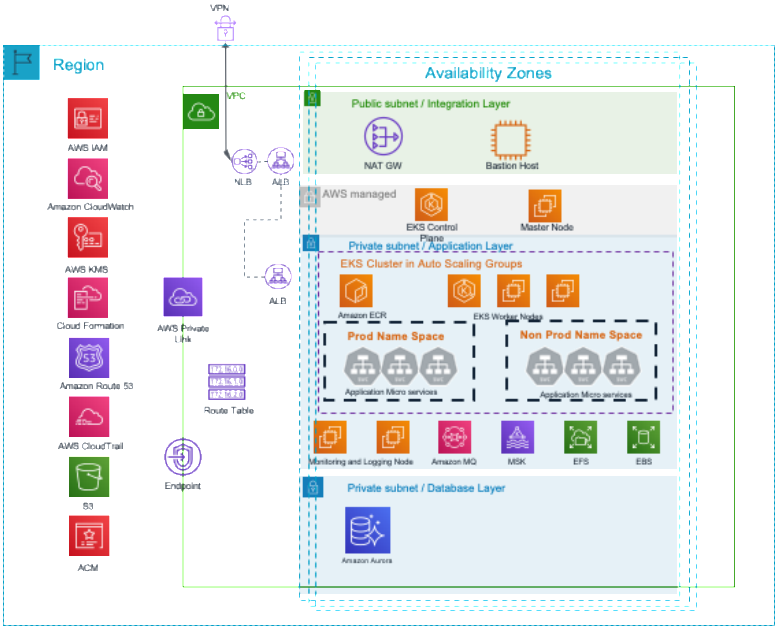
Architecture and Deployment Details

Architecture overview

The building blocks of the Netcracker Active Resource deployment architecture on AWS are shown in the diagram below. Netcracker active resource inventory is deployed within a single AWS Region, within a single Virtual Private Cloud (VPC).

The single VPC spans multiple Availability Zones, typically two or three Availability Zones are used for high availability and failover.

In a simplified view, in every Availability Zone, Netcracker active resource inventory Applications and Database infrastructures are deployed within two separate subnets. In addition, a third subnet is created to support Integration and remote management.



Building blocks of Netcracker active resource inventory deployment architecture on AWS

All Netcracker Active Resource Inventory applications are packaged as Docker images and deployed through Helm charts to Amazon Elastic Kubernetes Service (Amazon EKS), an AWS managed Kubernetes cluster. Helm charts are available in the Netcracker Installation Guide and provided to customers during deployment.

Amazon EKS provides a scalable and highly-available control plane that runs across multiple Availability Zones to eliminate a single point of failure. Kubernetes master nodes are managed by AWS.

Kubernetes worker nodes are deployed on Amazon Elastic Compute Cloud (Amazon EC2) instances, which are scaled and distributed across Availability Zone independently. Netcracker active resource inventory Docker images are pulled from an Amazon Elastic Container Registry (Amazon ECR) Docker registry.

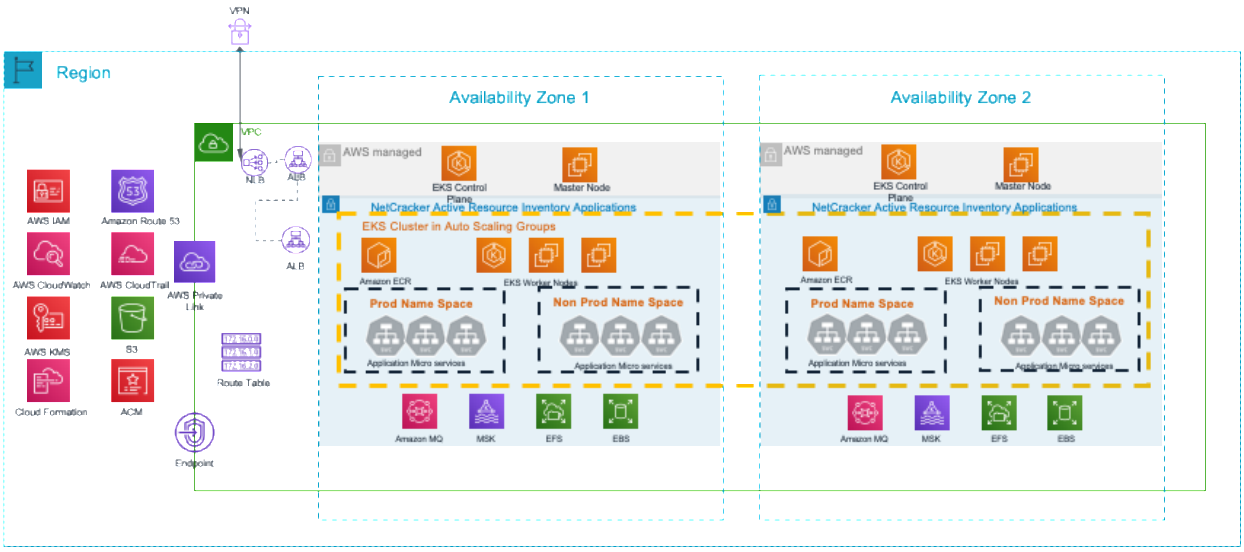
The Netcracker active resource inventory use the following AWS services for its database infrastructure:

- Amazon Aurora is the main relational data store for storing/retrieving Netcracker active resource inventory data components. Aurora supports both PostgreSQL and MySQL as option. Aurora is up to five time faster than standard MySQL databases and three times faster than PostgreSQL databases. It provides security, availability and reliability of commercial databases at 1/10th of the cost. Amazon Aurora is fully managed by Amazon Relational Database Service (RDS) which automates time-consuming administration tasks like hardware provisioning, database setup, patching and backups.

Application architecture

The architecture diagram below shows the parts of the architecture that make up an Amazon EKS cluster. Within the EKS cluster, all Netcracker active resource inventory applications are running as Micro-Applications / Docker containers (subsequently referred to as “apps”). Kubernetes master nodes are managed by AWS and Kubernetes worker nodes are scaled in an Amazon EC2 Auto Scaling Group (ASG).

Netcracker Active Resource Inventory supports deployment in production and non-productions namespaces per best practices. This supports isolation while allowing for better resource utilization to support a testing environment.



Kubernetes cluster high-level architecture

Netcracker active resource inventory users the following AWS services for its application layer:

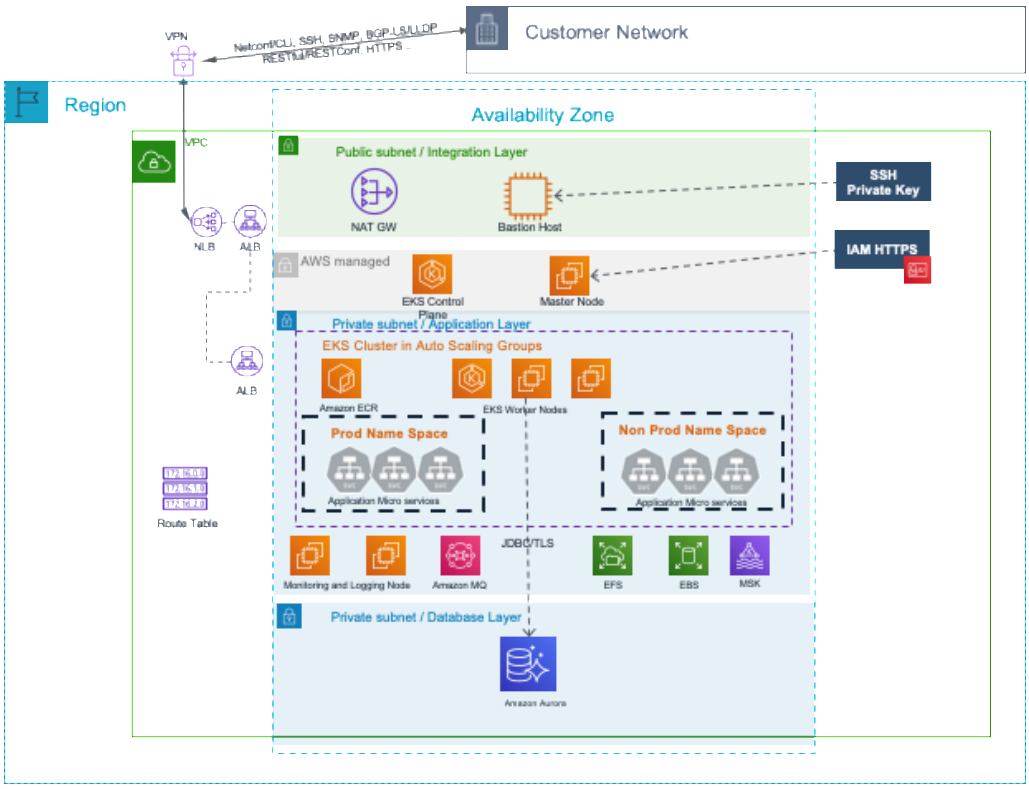
- Amazon Elastic Kubernetes Services (Amazon EKS) is a service that makes it easy to deploy, manage and scale containerized applications using Kubernetes on AWS. Amazon EKS makes it easy to provision and manage the compute capacity for Netcracker Active Resource Inventory cluster with a single command. EKS manages worker nodes for Netcracker active resource inventory apps using latest EKS-optimized Amazon Machine Images (AMI) while node updates and terminations gracefully drain nodes to ensure Netcracker active resource inventory apps stay stable.
- Amazon Elastic Container Registry (Amazon ECR) is a fully managed container registry that makes it easy to store, manage, and deploy Netcracker container images and artifacts. With Amazon ECR, Netcracker does not have the need to operate its own container repositories. Netcracker doesn't worry about the scaling of the underlying infrastructure. Moreover, it simplifies distribution of containers to customers leveraging the Netcracker Active Resource Inventory/ AWS-powered solution.
- Amazon Elastic Compute Cloud (Amazon EC2) provides secure, resizable compute capacity in the cloud. It provides the compute environment on which containers are managed. The required EC2 resources are created as needed by Amazon EKS, which manages the underlying resources required to run Netcracker active resource inventory.
- Amazon Elastic File System (Amazon EFS) provides a simple, scalable, fully managed elastic NFS file system for use with AWS cloud services. It is built to scale on-demand to petabytes, without disrupting applications - growing and shrinking automatically as files are added and removed, eliminating the need to provision and manage capacity to accommodate growth. As Netcracker Active Resource Inventory apps create, modify and remove files, EFS supports its growth.
- Amazon Elastic Block Store (Amazon EBS) provides persistent block-level storage volumes for use with EC2 instances in the AWS Cloud. Each Amazon EBS volume is automatically replicated within its Availability Zone to protect from component failure, offering high availability and durability. EBS volumes offer the consistent and low-latency performance needed to run workloads.

- Amazon Message Queue (Amazon MQ) provides a managed message broker service for open-source message-broker software RabbitMQ that makes it easy to set up and operate message brokers in the cloud for the Netcracker active resource inventory AWS solution. Amazon MQ enables apps to communicate with complex messaging patterns and with low-latency, high availability and message durability.
- Amazon Managed Streaming for Apache Kafka (Amazon MSK) provides a fully managed service that makes it easy to build and run applications that use Apache Kafka to process streaming data. MSK clusters are compatible with:
    - Apache Kafka partition reassignment tooling
    - Apache Kafka APIs
    - Apache Kafka Admin Client
    - 3rd Party tools

  Message and retention limits are determined by apps configuration.

Infrastructure setup

Compared to the high-level application architecture overview shown earlier, the figure below shows a more advanced infrastructure setup. For simplicity reasons, it is restricted to a single Availability Zone. For a high availability setup, see the High Availability (HA) Setup section.



Secure infrastructure via tiered configuration

The application and database subnets are private subnets. That is, the instances do not have public IP addresses and, so, are separated from the public internet.

Optionally, it is possible to deploy an access/integration layer subnet, which is a public subnet. This means that its Amazon EC2 bastion host, used for operational consideration, is visible by way of public IP address. A network address translation (NAT) gateway in the public subnet enables private EC2 instances and Kubernetes pods to communicate with the public internet.

Inbound communication runs through one of the following components:

- The bastion host, which can be used to establish SSH tunnels using private keys of the users
- The Kubernetes master nodes, which can be used to establish connection to the Kubernetes cluster through AWS Identity and Access Management (IAM)
- VPN connection allowing access to the Network Load Balancer / Application Load Balancers, which have separate security groups attached.

Thus, it is possible to do a fine-grained configuration of who has access to which part of the platform.

A VPN connection from the Customer Network support a secure encrypted connection between Netcracker active resource inventory on AWS and the network resources running on-premises such as Domain Controllers, Domain Managers, Optical Transport & Access, Microwave Transport and so on.

Depending on the size of the supported Customer Network and its associated data profiling, data can be ingested using AWS Direct Connect.

The Application Load Balancer is the single Access Point for user-related HTTPS traffic through the Netcracker active resource inventory, HTML5-based applications. Depending on the use case, the attached security group can be configured to be completely open or restricted to specific user groups.

The Network Load Balancer is the single Access Point for network-related traffic such as network discovery through various protocols for discovery such as RESTful/RESTConf, NETCONF/CLI, SNMP and BGP-LS/LLDP. Depending on the use case, the attached security group can be configured to be completely open or restricted to specific portion of the network.

Both the bastion host - and access to the Kubernetes cluster - are restricted to the DevOps team operating the platform.

Netcracker active resource inventory uses the following AWS services for its infrastructure setup:

- AWS Identity and Access Management (IAM) allows Netcracker to have fine-grained access control of AWS resources. IAM allows Netcracker active resource inventory resources to interact with each other with the minimal required privileges for a given task. IAM provides multi-factor authentication for highly-privileged users to protect the Netcracker active resource inventory/AWS environment even further.
- Amazon Route 53 provides Netcracker with a highly available and scalable cloud Domain Name System (DNS) web service. The Private DNS feature of Amazon Route 53 allows Netcracker to have an authoritative DNS within a Netcracker active resource inventory VPC without exposing DNS records to the internet . In addition, it provides an extremely reliable and cost-effective way to implement a DNS solution.

- Amazon CloudWatch provides Netcracker active resource inventory with the monitoring and observability required for DevOps engineers, developers, site reliability engineers (SREs) and IT managers. It provides insight on Netcracker active resource inventory resources utilization and provide a unified view of operational health. Amazon CloudWatch provides Netcracker active resource inventory customers with end-to-end operational visibility of metrics, logs, and distributed traces summarizing the performance and health of their Amazon Elastic Container Service for Kubernetes (EKS) cluster by pods/tasks, containers and services.
- Amazon CloudTrail simplifies Netcracker active resource inventory compliance audits by automatically recording and storing event logs for action made within the AWS account. AWS CloudTrail provides visibility into Netcracker active resource inventory resource activity by recording AWS Management Console actions and API calls. For example, it provides Netcracker active resource inventory customers with an easy way to identify if an authorized user attempts to modify a Kubernetes production namespace.
- Amazon Simple Storage Service (Amazon S3) is an object storage service built to store and retrieve any amount of data. It offers industry-leading durability, availability, performance, security and virtually unlimited scalability at very low costs. It provides Netcracker active resource inventory with a storage solution for disaster recoveries by hosting database snapshots and applications backups.
- AWS CloudFormation enables developers and businesses to create collections of related AWS and third-party resources, and provision and manage them in an orderly and predictable fashion. AWS CloudFormation simplifies the codification of infrastructure by supporting JSON or YAML declarative code files that describe the intended state of all the resources needed to deploy applications. Netcracker uses AWS CloudFormation templates as a proxy to Terraform, an open-source infrastructure as code software tool created by HashiCorp. A Terraform server running on Amazon EC2 can be used to support customers through VPC peering.
- Amazon Key Management Service (Amazon KMS) is used to create and manage cryptographic keys and control their use across a wide range of AWS services and in Netcracker active resource inventory applications. AWS KMS is a secure and resilient service that uses hardware security modules that have been validated under FIPS 140-2, or are in the process of being validated, to protect Netcracker active resource inventory keys.

- AWS Certificate Manager (ACM) is a service that allows Netcracker to easily provision, manage, and deploy Secure Sockets Layer/Transport Layer Security (SSL/TSL) certificates for use with AWS services and secure Netcracker active resource inventory internally-connected resources. AWS Certificate Manager allows Netcracker to quickly request a certificate, deploy it, and let AWS Certificate Manager handle certificate renewals. It manages the certificate lifecycle centrally.

Application scaling

In order to maintain Netcracker active resource inventory availability when application traffic increases - such as during a large number of network inventory updates - the application need to be scaled up. When the application traffic decreases, the application needs to be scaled down in order to prevent unnecessary costs. In short, Netcracker active resource inventory on AWS uses the resources that it needs - and not more - limiting costly overprovisioning.

As outlined earlier, Netcracker active resource inventory Kubernetes worker nodes are started and scaled within a separate AWS Auto Scaling Group (ASG). Netcracker active resource inventory leverage ASG which monitor the applications, and automatically adjusts capacity to maintain steady, predictable performance at the lowest possible cost.

Using AWS Auto Scaling, Netcracker active resource inventory on AWS uses application scaling for multiple resources across multiple services in minutes, based on predefined rules.

Kubernetes worker nodes scaling is actually triggered by Kubernetes Horizontal Pod Autoscaler (HPA). Depending on the CPU and memory consumption metrics of Kubernetes deployments, HPA decides to increase or decrease the number of pods within this deployment. If the new pod cannot be scheduled, it switches to the pending state. This triggers the Cluster Auto Scaler, which scales up the number of worker nodes via the assigned AWS Auto Scaling group.

Infrastructure Availability

First, AWS has the concept of Regions, which are physical locations around the world where AWS clusters data centers. Each group of logical data centers is called an Availability Zone (AZ).

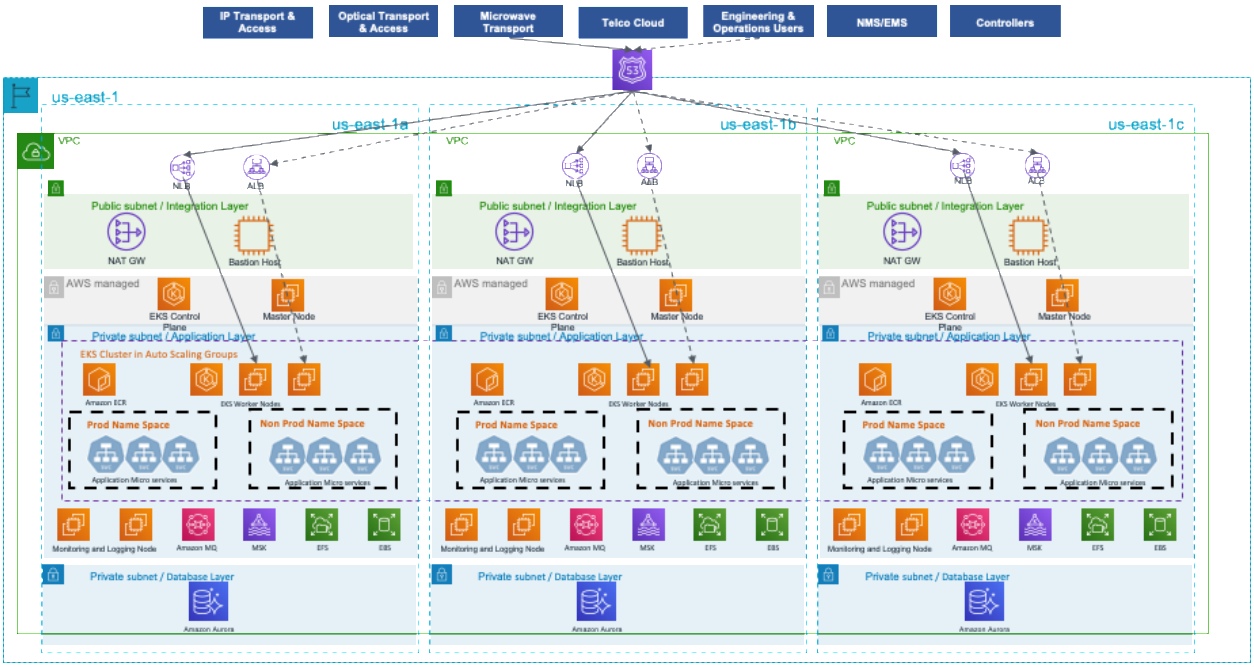
Each AWS Region consists of multiple, isolated, and physically separate AZs within a geographic area. Each AZ has independent power, cooling, and physical security and is connected via redundant, ultra-low-latency network. All traffic between AZs is encrypted. AZ's make partitioning applications for high availability easy.

If an application such as Netcracker active resource inventory on AWS is portioned across AZs, companies are better isolated and protected from issues such as power outages, lightning strikes, tornadoes, earthquakes, and mode. Each AZ is physically separated by a meaningful distance, many kilometers from any other AZ, although all are within 100 km (60 miles) of each other.

High availability setup

A typical high availability (HA) setup spans multiple Availability Zones. If one of the Availability Zones becomes unavailable, the application continues to stay available, because the architecture is highly available in all the layers as described below. Netcracker active resource inventory on AWS is designed to be highly available by spreading applications components and their underlying resources across physically isolated availability zones.

The diagram below shows an example of Netcracker active resource inventory high availability setup deployed in us-east-1 Region spawning three Availability Zones: namely us-east-1a, us-east-1b and us-east-1c.



High Availability Setup Across Multiple Availability Zones

Application Availability

For high availability, the following application layer components are relevant:

- Amazon Route 53
- Application Load Balancer and Network Load Balancer
- Kubernetes worker nodes
- Kubernetes deployments
- Amazon Managed Streaming for Apache Kafka (Amazon MSK)

The relevant database layer component is Amazon Aurora, discussed in the next Database Availability section.

As shown above, Amazon Route 53 simplifies private DNS management by support Fully Qualified Domain Names (FQDNs)for communicating with Netcracker active resource inventory applications. Dynamic notifications for network inventory are routed toward each Availability Zone using the configured DNS Routing Policies to a Network Load Balancer (NLB). The NLB then balances toward the relevant worker node.

Similarly, HTTPS traffic is routed toward each Availability Zone using the configured DNS Routing Policies to an Application Load Balancer (ALB). The ALB then balances toward the relevant worker node.

Elastic Load Balancing and Amazon Route 53

AWS Elastic Load Balancing (ELB) , comprised of an Application Load Balancer (ALB) and Network Load Balancer (NLB), creates a load balancer node in every Availability Zone where Kubernetes worker nodes are deployed. Every load balancing node is registered with Amazon DNS service. If a request is sent via the DNS name of the load balancer, it receives one or all of the IP addresses of the load balancers. Therefore, if one Availability Zone is unavailable or has no healthy nodes, the load balancer can continue to route traffic to the healthy targets in another Availability Zone.

Kubernetes worker nodes

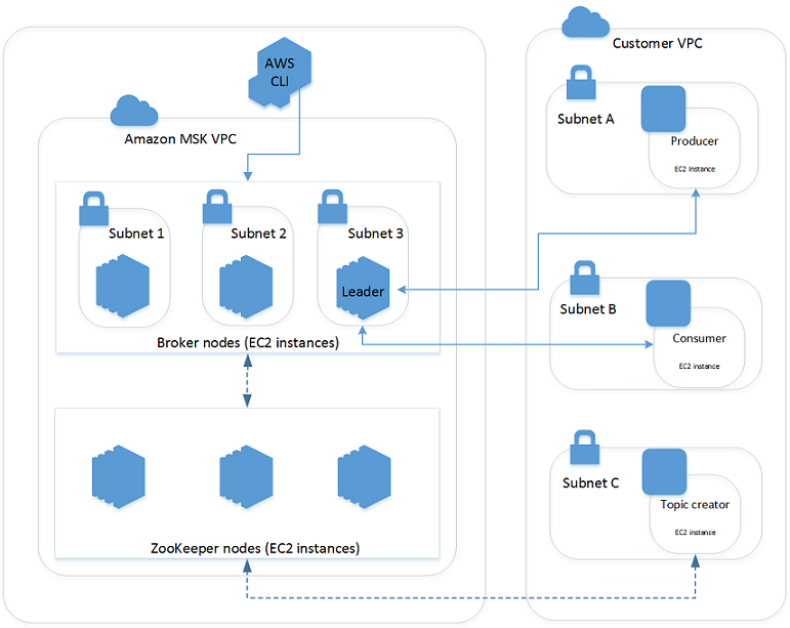
Kubernetes worker nodes are distributed across all Availability Zone by the respective Auto Scaling Group. In case of a failing Availability zone, the Auto Scaling Group detects the failure and balances requests to worker nodes in other Availability Zones.

Kubernetes Deployments

Since Kubernetes automatically spreads the pods in a deployment across nodes and Availability Zone, the impact of an Availability Zone failures is mitigated. Because pod placement is best-effort, pods might not evenly spread, especially if the Availability Zones in the cluster are heterogeneous (that is, different numbers of nodes, different types of nodes, or different pod resource requirements). Using homogeneous Availability Zones (same number and types of nodes) reduces the probability of unequal spreading. For more information, see Running in multiple zones in the Kubernetes best practices documentation.

Amazon Managed Streaming for Apache Kafka (Amazon MSK)

Amazon MSK continuously monitors the health of your clusters and replaces unhealthy brokers without downtime to your applications. Amazon MSK manages the availability of your Apache ZooKeeper nodes so you will not need to start, stop, or directly access the nodes yourself. Amazon MSK uses multi-AZ replication for high-availability.



Amazon MSK Cluster

The diagram above demonstrates the interaction between the following components:

- **Broker nodes** – When creating an Amazon MSK cluster, you specify how many broker nodes you want Amazon MSK to create in each Availability Zone. Each Availability Zone has its own Amazon MSK virtual private cloud (VPC) subnet.
- **ZooKeeper nodes** – Amazon MSK creates the Apache ZooKeeper nodes for you. Apache ZooKeeper is an open-source server that enables highly reliable distributed coordination
- **Producer, consumers, and topic creators** – Amazon MSK lets you use Apache Kafka data-plane operations to create topics and to produce and consume data
- **AWS CLI** – Allows usage of the AWS Command Line Interface (AWS CLI) or the APIs in the SDK to perform control-plane operations.

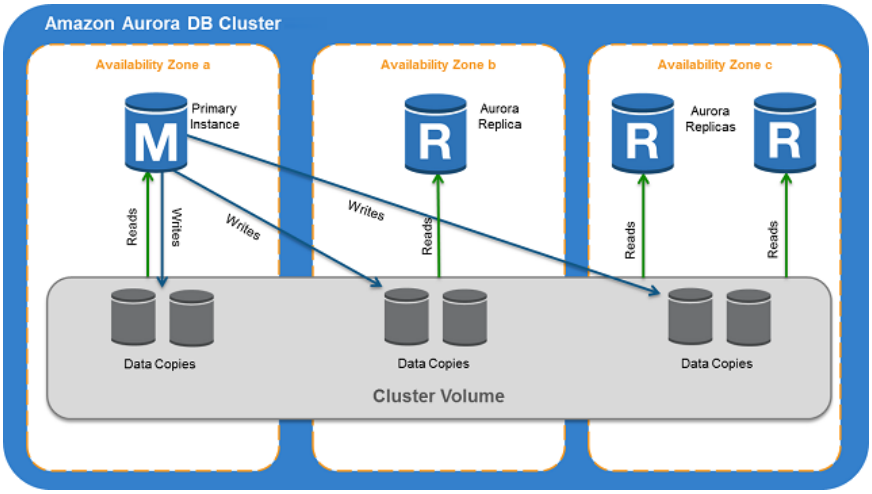
Amazon MSK detects and automatically recovers from the most common failure scenarios for the cluster so that Netcracker active resource inventory on AWS producer and consumer applications can continue their write and read operations with minimal impact. When Amazon MSK detects a broker failure, it mitigates the failures or replaces the unhealthy or unreachable broker with a new one.

Database Availability

Netcracker active resource inventory on AWS uses Amazon Aurora which further extends the benefits of multi-AZ employing an SSD-backed virtualized storage layer purpose-built for database workloads. It automatically replicates your storage six ways, across three Availability Zones.

Amazon Aurora is fault-tolerant, transparently handling the loss of up to two copies of data without affecting database write availability and up to three copies without affecting read availability. Aurora always replicates your data across three Availability Zones, regardless of whether your database uses read replicas.

Failover is automatically handled by Amazon Aurora, so that Netcracker active resource inventory on AWS always resume database operations as quickly as possible without manual administrative intervention.



Amazon Aurora DB Cluster

An Amazon Aurora DB clusters consists of one or more DB instances and a cluster volume that manage the data for those DB instances. An Aurora cluster volume is a virtual database storage volume that spans multiple Availability Zones, with each Availability Zone having a copy of the DB cluster data.

The Primary DB instances support read and write operations, while Aurora replicas only support read operations. Failover is achieved by promoting one of the replicas to a Primary DB instance and usually takes less than 30 seconds.

Architecture Pillars

The AWS Well-Architected Framework helps cloud architects build secure, high-performing, resilient, and efficient infrastructure for their applications and workloads. Based on five pillars — operational excellence, security, reliability, performance efficiency, and cost optimization — AWS Well-Architected provides a consistent approach for customers and partners to evaluate architectures, and implement designs that can scale over time.

The AWS Well-Architected Framework was used by Netcracker to design the Netcracker active resource inventory on AWS.

Below is an overview of the five pillars of the AWS Well-Architected Framework:

- **The Operational Excellence** pillar focuses on running and monitoring systems to deliver business value, and continually improving processes and procedures. Key topics include automating changes, responding to events, and defining standards to manage daily operations.
- **The Security** pillar focuses on protecting information and systems. Key topics include confidentiality and integrity of data, identifying and managing who can do what with privilege management, protecting systems, and establishing controls to detect security events.
- **The Reliability** pillar focuses on ensuring a workload performs its intended function correctly and consistently when it's expected to. A resilient workload quickly recovers from failures to meet business and customer demand. Key topics include distributed system design, recovery planning, and how to handle change.
- **The Performance Efficiency** pillar focuses on using IT and computing resources efficiently. Key topics include selecting the right resource types and sizes based on workload requirements, monitoring performance, and making informed decisions to maintain efficiency as business needs evolve.
- **The Cost Optimization** pillar focuses on avoiding unnecessary costs. Key topics include understanding and controlling where money is being spent, selecting the most appropriate and right number of resource types, analyzing spend over time, and scaling to meet business needs without overspending.

Operational Excellence for Netcracker active resource inventory on AWS

The Netcracker active resource inventory on AWS takes advantage of a wide range of managed services offered by AWS. Amazon CloudWatch enables the monitoring of Netcracker active resource inventory and provide insights on application health. CloudWatch dashboards provides a single view of Netcracker active resource inventory resources and support collection of metrics and alarms. CloudWatch enables easy monitoring of Container Insight on Amazon EKS and Kubernetes. This ability to collect useful metrics allows on-going optimization of Netcracker apps.

The Netcracker active resource inventory on AWS uses Amazon Aurora for its database layer. Combined with Amazon CloudWatch, Netcracker is able to provide enhanced monitoring and performance insight for the database layer that includes:

- Process/Thread list with host metrics
- 1-60 second granularity
    - Recommended to start with 15s and dial down to 1s when troubleshooting
- Measurement of database load
- Measurement of database average active sessions
- Ability to view in the AWS Management Console or push to monitoring systems

Because AWS offers a wide range of tools tightly integrated across the platforms and the applications, Netcracker active resource inventory on AWS is built to automatically change and respond to events and support the optimization of daily operations in support of both the network responses and user responses.

You can find prescriptive guidance on implementation in the Operational Excellence Pillar whitepaper.

Security for Netcracker active resource inventory on AWS

Netcracker active resource inventory on AWS takes advantage of AWS inherent security capabilities and features to protect data, implement detection and control of security events and encrypt communication between Netcracker active resource inventory microservices and toward the customer network, both users and data ingestion.

As discussed in section Architecture Overview, Netcracker leverage the simplicity of subnet management to improve on its legacy Netcracker active resource inventory solution.

With AWS, it is simple to manage a three-tier application allow to control the communication flow from within the application sub-networks and outside of the application network. It is optimized for network segmentation and application firewalling.

Certificate Management

Netcracker active resource inventory on AWS makes use of the Amazon Certificate Manager (ACM) to reduce complexity of managing Certificate Authority (CA). Amazon Certificate Manager simplifies security management by automating the renewal of expiring certificates.

Netcracker active resource inventory on AWS leverage the Amazon Certificate Manager Private CA service, which allows enterprise customers to build public key infrastructure (PKI) inside the AWS cloud and intended for private use within an organization.

Refer to the Amazon Certificate Manager User Guide for additional details.

Shared Responsibility Model

A shared responsibility model of security between AWS and the customers is applied to Netcracker implementation on AWS. The shared responsibility model describes this as security ***of*** the cloud, and security ***in*** the cloud.

- **Security *of* the cloud** – AWS is responsible for protecting the infrastructure that runs AWS services in the AWS Cloud. For Amazon Simple Storage Service (Amazon S3), where data resides, AWS is responsible for the underlying infrastructure and its security. For Amazon Elastic Kubernetes Service (Amazon EKS), AWS is responsible for the Kubernetes control plane, which includes the control plane nodes and etcd database. Third-party auditors regularly test and verify the effectiveness of our security as part of the AWS compliance programs.
- **Security *in* the cloud** – Your responsibility includes the following:
    - The sensitivity of your data, your company's requirements and applicable laws and regulations. This includes configuring Amazon S3 and Amazon Elastic File System (Amazon EFS) with proper access policies and encryption.
    - Security configuration of the data plane, including the configuration of security groups that allow traffic to pass from Amazon EKS control plane into the customer virtual private cloud (VPC).
    - Configuration of the worker nodes and the containers themselves.

- The worker node guest operating system (including updates and security patches).
- Setting up and managing network controls, such as firewall rules managing platform-level identity and access management, either with or in addition to AWS Identity and Access Management (IAM).

You can find prescriptive guidance on implementation in the Security Pillar whitepaper.

AWS compliance

AWS compliance enables you to understand the robust controls in place at AWS to maintain security and data protection in the cloud. AWS engages with external certifying bodies and independent auditors to provide you with extensive information regarding the policies, processes, and controls established and operated by AWS. To learn more, see AWS Compliance.

Reliability for Netcracker active resource inventory on AWS

Netcracker active resource inventory on AWS is built with the ability to perform its intended function correctly and consistently when it's expected to. Below a small overview of how Netcracker active resource inventory on AWS uses the five design principles for reliability in the cloud:

1. **Automatically recover from failure** – As illustrated in above, Netcracker leverage Elastic Load Balancing to distribute load to an Amazon EKS Cluster in an Auto Scaling Group spawning multiple Availability Zones. Netcracker leverage Amazon CloudWatch and EKS inherent capabilities to automatically recover from pods failures.
2. **Test Recovery procedures** – With AWS, Netcracker has the ability to use automation to simulate different failures and validate the recovery procedures. Netcracker uses automation and the AWS cloud capabilities to test and fix before a real failure scenario occurs, thus reducing risk.
3. **Scale horizontally to increase aggregate workload availability** – Netcracker active resource inventory on AWS allows for using multiple small resources to reduce the impact of a single failure on the overall workload. As shown above, distributing requests across multiple, smaller resources to ensure that they don't share a common point of failure.

4. **Stop Guessing Capacity** – Netcracker active resource inventory on AWS allows for a solution that grows with the network demand and growth. AWS allows Netcracker to grow with the Network, where additional resources are spinned-off as the amount of inventory data flow into Netcracker active resource inventory. Similarly, it allows Netcracker to adapt its CPU and memory to the amount of inventory required at a given time. This is a complete opposite to on-premises workloads that often reach resource saturation.
5. **Manage change in automation** – Changes to Netcracker active resource inventory is made through automation using AWS CloudFormation.

You can find prescriptive guidance on Implementation in the Reliability Pillar whitepaper.

Performance Efficiency for Netcracker active resource inventory on AWS

Netcracker active resource inventory on AWS is built to use computing resources efficiently to meet network requirements, and to maintain that efficiency as new technologies (such as 5G) are introduced.

It is designed to globally deploy a customer solution in minutes. This can reduce lengthy on-premises procurement processes..

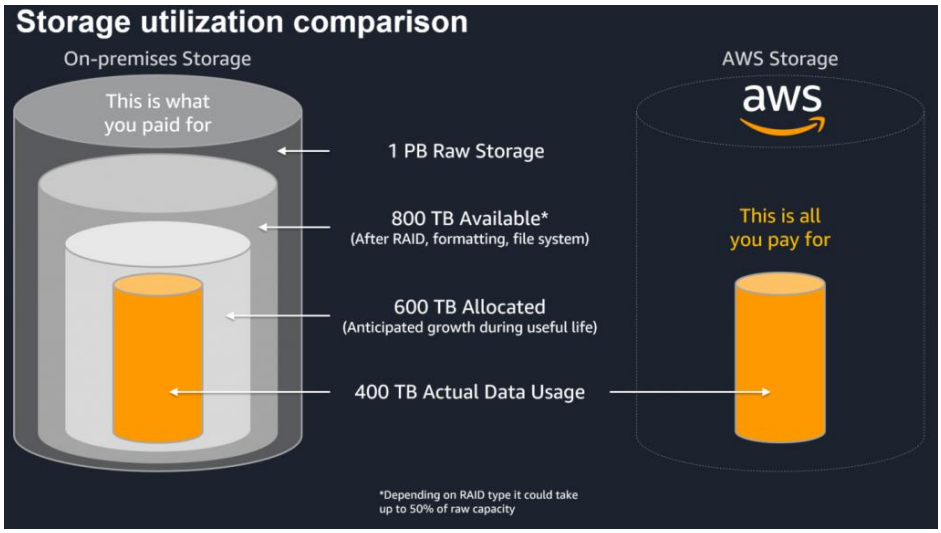
Netcracker Active Resource Inventory on AWS allows Netcracker to innovate by using AI/ML.

You can find prescriptive guidance on implementation in the Performance Efficiency Pillar whitepaper.

Cost Optimization for Netcracker Active Resource Inventory on AWS

Netcracker Active Resource Inventory on AWS is designed with the goal of delivering business value at the lowest price point. It uses Amazon Aurora PostgreSQL. Meanwhile, Amazon Aurora is up to five times faster than standard MySQL databases and three times faster than standard PostgreSQL databases. It provides the security, availability, and reliability of commercial databases at 1/10th the cost.

Netcracker Active Resource Inventory on AWS uses Amazon S3 to reduce its storage cost compared to on-premises solution. The below figure illustrates how AWS Storage is more efficient, due to only paying for the resources and services used, rather than paying for what is needed 3-5 years down the road.



Storage utilization comparison

In addition, Amazon Linux 2 and CentOS images provide a considerable cost reduction opportunity in comparison to licensed and supported heavy operating systems.

AWS allows Netcracker Active Resource Inventory on AWS to optimize its resource footprint to match the network inventory update patterns. It is common practice for CSPs and ISPs to update their network during maintenance windows resulting in a surge of inventory data in the early morning. As such, the elasticity provided by the AWS through Auto Scaling Group, allows for considerable cost reduction.

You can find prescriptive guidance on implementation in the Cost Optimization Pillar whitepaper.

Conclusion

AWS and Netcracker provide capabilities for customers to deploy an Inventory Management architecture to attain scalability, elasticity, and high availability. Customers are using AWS, AWS Partner Network (APN) Partners, and open-source solutions to host mobile workloads on AWS. This has resulted in reduced cost, greater agility, and a reduced global footprint. For partner solutions, AWS has the broadest and strongest partners in the ecosystem, available through AWS Marketplace and APN Partner Central for each part of the stack presented in this paper.

This paper serves as a deep dive into how Netcracker addresses inventory managmenet in the Telecom industry by leveraging AWS Services.

It explains how this is achieved by running a scalable and resilient IM platform that can handles CSPs network evolution and growth. Running Netcracker Active Resource Inventory on AWS is a proven platform with experience in massive scale workloads.

Contributors

Contributors to this document include:

- Yaroslav Krasnov, Product Manager – Cloud OSS, Netcracker
- Aliaksandr Khaletski, Product Marketing Manager, Netcracker
- Visu Sontam, Senior Partner Solutions Architect, WW Telco Partner, Amazon Web Services
- Antonello Arpino, Principal OSS/BSS Transformation Lead, WW Telco Partner, Amazon Web Services
- Aymen Saidi, Principal Partner Solutions Architect, WW Telco Partner, Amazon Web Services

Further Reading

For additional information, see:

- 5G Network Evolution with AWS
- Continuous Integration and Continuous Delivery for 5G Networks on AWS
- Next-Generation Mobile Private Network Powered by AWS
- AWS Well-Architected Framework
- Next-Generation OSS with AWS

Document Revisions

Date	Description
November 2021	First publication

Glossary

- **3GPP**: 3rd Generation Partnership Project
- **ACM**: AWS Certificate Manager
- **AI / ML**: Artificial Intelligence / Machine Learning
- **ALB**: Application Load Balancer
- **Amazon EBS**: Amazon Elastic Block Store
- **Amazon EC2**: Amazon Elastic Compute Cloud
- **Amazon ECR**: Amazon Elastic Container Registry
- **Amazon EFS**: Amazon Elastic File System
- **Amazon EKS**: Amazon Elastic Kubernetes Service
- **Amazon KMS**: AWS Key Management Service
- **Amazon MQ**: Amazon MQ is a managed message broker service for Apache ActiveMQ and RabbitMQ
- **Amazon MSK**: Amazon Managed Streaming for Apache Kafka
- **Amazon RDS**: Amazon Relational Database Service
- **Amazon S3**; Amazon Simple Storage Service
- **AMI**: Amazon Machine Image
- **API**: Application Programming Interface
- **APN**: AWS Partner Network
- **ASG**: Auto Scaling Group
- **AWS CLI**: AWS Command Line Interface
- **AWS IAM**: Amazon Web Services Identity and Access Management
- **AZ**: Availability Zone
- **B2B**: Business-to-Business
- **BGP-LS**: Border Gateway Protocol - Link State
- **BoD**: Bandwidth On Demand

- **BSS**: Business Support Systems
- **CA**: Certificate Authority
- **CAPEX**: Capital Expenditure
- **CI / CD**: Continuous Integration and Continuous Deployment
- **CISM**: Container Infrastructure Service Management
- **CLI**: Command Line Interface
- **CPE**: Customer Premise(s) Equipment
- **CPU**: Central Processing Unit
- **CSP**: Communication Service Providers
- **CU**: Centralized Unit
- **DB**: Database
- **DNS**: Domain Name Service
- **DU**: Distributed Unit
- **DWDM**: Dense Wavelength Division Multiplexing
- **ELB**: Elastic Load Balancing
- **EMS**: Element Management System
- **ERP**: Enterprise Resource Planning
- **FIPS**: Federal Information Processing Standards
- **GIS**: Geographic information system
- **HA**: High Availability
- **HPA**: Horizontal Pod Autoscaler
- **HTML5**: Hypertext Markup Language 5
- **HTTPS**: Hypertext Transfer Protocol Secure
- **HW**: Hardware
- **IETF**: Internet Engineering Task Force
- **IM**: Inventory Management

- **IoT**: Internet of Things
- **IP**: Internet Protocol
- **IT**: Information Technology
- **JSON**: JavaScript Object Notation
- **LLDP**: Link Layer Discovery Protocol
- **NAT**: Network Address Translation
- **NEs**: Network Element
- **NETCONF**: Network Configuration Protocol
- **NFS**: Network File System
- **NLB**: Network Load Balancer
- **NMS**: Network Management System
- **Open vRAN**: Open and virtualized Radio Access Networks
- **OS&R**: Operations. Support &. Readiness
- **OSS**: Operational Support Systems
- **OTN**: Optical Transport Networking
- **PKI**: Public Key Infrastructure
- **RAN**: Radio Access Networks
- **REST API**: REpresentational State Transfer Application Programming Interface
- **RF**: Radio Frequency
- **RIC**: RAN Intelligent Controller
- **RRH**: Remote Radio Heads
- **RU**: Radio Unit
- **SD-LAN**: Software-Defined Local Access Network
- **SD-WAN**: Software-Defined Wide Area Network
- **SD-WiFi**: Software-Defined Wireless Fidelity
- **SDK**: Software Development Kit

- **SDN**: Software-Defined Networking
- **SDN/NFV**: Software-Defined Networking (SDN) and Network Function Virtualization (NFV)
- **SMB**: Small and Medium-sized Business
- **SNMP**: Simple Network Management Protocol
- **SREs**: Site Reliability Engineers
- **SSD**: Solid-State Drive
- **SSH**: Secure Shell Protocol
- **SSL / TLS**: Secure Sockets Layer/Transport Layer Security
- **TCO**: Total Cost of Ownership
- **TTM**: Time To Market
- **uCPE**: Universal Customer Premise Equipment
- **vCU/DU**: virtualized Central Unit (vCU), a virtualized Distributed Unit (vDU)
- **VIM**: Virtualized Infrastructure Manager
- **VLAN**: Virtual Local Area Network
- **VNF / CNF**: Virtual Network Functions / Cloud-native Network Functions
- **VPC**: Virtual Private Cloud
- **VPN**: Virtual Private Network
- **VXLAN**: Virtual Extensive Local Area Network
- **WW**: Worldwide
- **XaaS**: (Anything) as a Servic
- **xNFs**: Network Functions
- **YAML**: Yet Another Markup Language