Radiator VNF high level definition

High-level definition of Radiator VNF release 2017. Last revised on July 10, 2017

Table of Contents

1. Basic concepts of NFV and Radiator VNF

This section introduces the basic concepts of NFV (Network Functions Virtualisation), Radiator, and Radiator VNF.

1.1. Network function virtualisation

NFV is a network architecture concept, which uses virtualisation technologies to virtualise network node functions. Utilising NFV decreases costs because it runs on virtual machines in data centres using generic hardware, instead of having dedicated servers for the required functions. NFV also enables rapid service development and deployment and provides easy scalability.

To get the most out of NFV, it is important that the virtualisation process is automated. If the NFV concept is limited to only manually managing the virtual servers, many of its benefits are missed altogether.

Scaling

When talking about scaling NFV systems, it is important to separate the 2 different methods of scaling:

- Horizontal scaling

  Horizontal scaling (scale-out/scale-in) refers to adding or removing virtualised components instead of adding or removing the resources of the current components. Horizontal scaling is ideal for the processing nodes that do not communicate with external systems.

- Vertical scaling

  Vertical scaling (scale-up/scale-down) refers to adding or removing virtual processors and memory to a single virtualised component instead of increasing or decreasing their number. Vertical scaling is ideal for the components that communicate with external systems.

1.2. Radiator® AAA Server

Radiator® AAA Server (later Radiator) is a highly configurable and extensible RADIUS, TACACS+, and Diameter server that allows you to easily customise and control how to authenticate users and record accounting information.

1.3. Radiator VNF

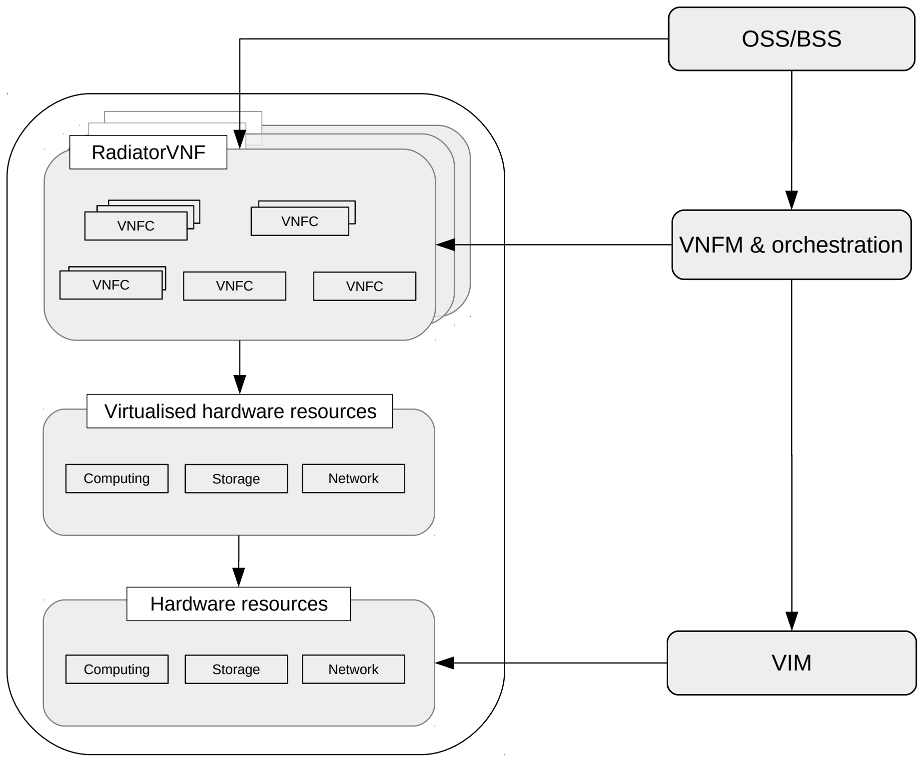
Radiator VNF is a true NFV solution because it handles the virtualisation process automatically. Radiator VNF Manager scales and configures the virtual servers without human interaction. It also automatically configures all new instances. The scaling is based on KPI (Key Performance Indicator)s, which the Radiator VNF system produces. Scaling does not compromise service reliability, there are no interruptions in service when scaling the system.

Radiator VNF is a hardware-independent solution. This provides easy integration for Radiator VNF and makes it truly portable for different environments. Radiator VNF also works with any industry-standard NFV infrastructure.

2. Radiator VNF architecture

Radiator VNF follows ETSI principles in the general architecture. For more information about ETSI, see ETSI GS NFV-SWA 001 [http://www.etsi.org/deliver/etsi_gs/NFV-SWA/001_099/001/01.01.01_60/gs_nfv-swa001v010101p.pdf]. Radiator VNF is built on multiple VNFC (Virtualised Network Function Component)s, which all provide interfaces for external or internal communication. These features provide the possibility to have access-type-specific VNF (Virtualised Network Function)s running on the same hardware. This makes the configurations easy to manage. The configuration change does not affect the whole system but only the specific VNF. Also the testing process is easy when there is no need to test the complete system.

Figure 1. Unified vAAA architecture with Radiator VNF



2.1. VNF components

Radiator VNF has a modular configuration that provides flexibility and a customisable solution for NFV. It has 5 basic components, which handle certain parts of the infrastructure:

- Loadbalancer

  The loadbalancer component communicates with NAS (Network Access Server) clients and similar from the external systems and distributes the messages to the worker components.

- Worker

  The worker component handles the messages depending on business logic.

- Control

  The control component provides the redundant message queue that other components use for internal communicating.

- BEDBINT (Backend Database Interface)

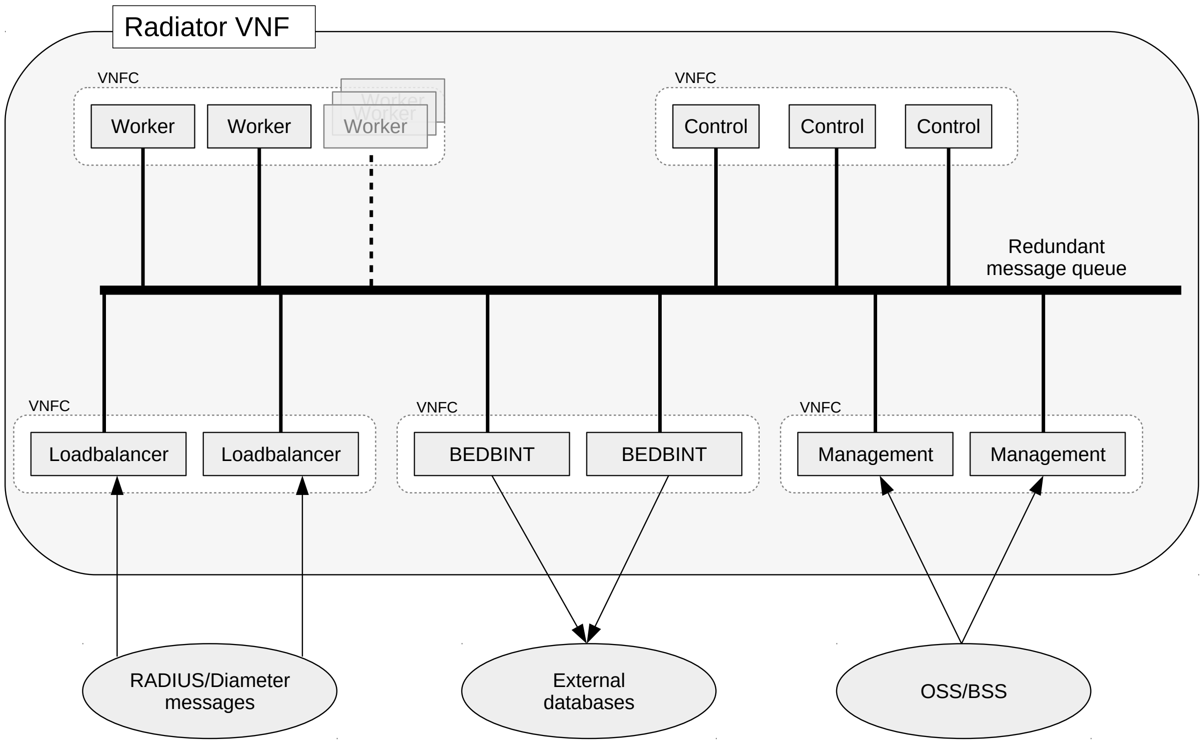
  BEDBINT provides an interface for communicating with subscriber databases.

- Management

The management component provides management options for the system and different API functions for external systems.

The components and their relations are shown in figure 2.

Figure 2. Internal architecture of Radiator VNF



2.1.1. Worker

The worker is the VNFC that handles the actual business logic. It processes the incoming RADIUS and Diameter messages. The workers provide the required functions, such as:

- Radiator AAA Core
- Radiator VoWiFi
- Radiator Service Provider Wi-Fi
- Radiator PCRF
- Radiator OCS
- Radiator Custom/OEM

Radiator AAA Core module is the mandatory main module, which is used in combination with other modules to provide the business logic.

One worker node can have multiple Radiator processes for handling the incoming messages. Radiator VNF can utilise resources better if there are several Radiator processes ongoing. When workers are created, they send their availability information to the loadbalancer through the control component using the redundant message queue. Workers do not communicate directly with external clients.

2.1.2. Loadbalancer

The loadbalancer is the primary entry point for incoming RADIUS and Diameter messages. It forwards the messages to the workers and takes care of balancing the workers' load. The loadbalancer also gathers the needed client information of the incoming messages and, based on this, sends the messages back to the correct clients.

The loadbalancer is connected to the redundant message queue. The workers sign up to the control component when they are initialised and the control component sends worker information to the loadbalancer through the redundant message queue. Based on worker information, the loadbalancer creates a list of the available workers. To ensure easy scalability, the loadbalancer is stateless from the RADIUS and Diameter perspective.

2.1.3. Control

The control component provides the redundant message queue for inter-component communication. The control component sends and receives data from all other components through different protocols and acts as a control-plane. Its most important roles are the following:

- Redundant message queue
  - Component registration
  - Statistics publishing
- Non-persistent data storage
- Persistent data storage
- Service discovery

The redundant message queue uses the publish/subscriber method for communication, as all other components need to communicate with each other through the redundant message queue. When created, every worker communicates its availability to the loadbalancer. Also every worker shutdown is informed through the redundant message queue.

The control component acts as a data storage, it stores non-persistent session data and persistent data, such as IP addresses, prefixes, and pools. The control component is not stateless. The control component also maintains real-time user session data and provides that data to the worker and management components. The persistent data is stored in ACID-compliant storage.

The control component uses a service discovery mechanism to handle the initial configuration and any possible changes in connectivity. Radiator VNF uses Consul for service discovery, simple key/value storage, and leader election.

2.1.4. BEDBINT

The BEDBINT connects Radiator VNF to any external backend systems, such as customer LDAP (Lightweight Directory Access Protocol), SQL, or HSS (Home Subscriber Server), depending on the configuration.

The BEDBINT uses proprietary protocol to communicate with the LDAP system through a secure connection. Most subscriber databases use synchronous systems, such as LDAP, for storing the subscriber information. The BEDBINT enables the internal components of Radiator VNF to use the asynchronous approach.

The BEDBINT includes a caching function that stores configurable semi-static or dynamic customer and configuration information. The cache is periodically refreshed and it can be turned off altogether.

The BEDBINT supports the overload protection mechanisms, scaling, and self-healing.

The BEDBINT can act as a Diameter routing agent, which provides real-time routing capabilities. It also manages session binding to maintain an efficient and scalable network roll-out.

2.1.5. Management

The management component holds the statistics and logging data. It also provides certain services which handle session cleaning and other management tasks related to the VNF data, such as logs and statistics. The management component hosts several different systems that support the rest of the solution:

- Statistical applications, which collect system information
- Application-level information logging from the control component, such as user level trace and application debugging information
- CDR (Call Detail Record) generation and session cleanup

2.2. Communication

Radiator VNF supports several different interfaces for communication with external components.

Data plane communication

Radiator VNF support the following interfaces for external clients:

- RADIUS (UDP (User Datagram Protocol))
- Diameter (TCP or SCTP (Stream Control Transmission Protocol))

Radiator supports the following interfaces for external databases:

- SQL
- LDAP (including Ud)
- MAP (Mobile Application Part)
- SOAP (Simple Object Access Protocol)
- Diameter

Control plane communication

The control plane includes proprietary notification or request-reply protocols over a publish/subscribe redundant message queue.

Radiator supports the following interfaces for OSS (Operations Support Systems)/BSS (Business Support Systems):

- SNMPv2c (Community-Based Simple Network Management Protocol version 2)
- SNMPv3 (Simple Network Management Protocol version 3)
- Syslog

3. Radiator VNF features

This section discusses features that Radiator VNF provides for users.

3.1. Basic usage features

The basic usage features of Radiator VNF include:

- Fault management

  If some kind of an error situation occurs, VNFCs try to recover automatically.

- Data analysing and visualisation tools

  Radiator VNF provides interfaces for log graphical tools, which analyse the log data and create diagrams showing system usage.

- Testing tools

  Radiator VNF package contains ready templates for testing tools.

- Manual scaling

  Radiator VNF also provides a possibility to scale the system manually.

3.2. Security features

Radiator VNF has several security-enhancing features:

- Radiator VNF components can be run in an SELinux-enforced environment.
- It serves only authorised RADIUS and Diameter clients.
- VNF workers support TLS (Transport Layer Security) protocol.
- VNF workers support several encryption protocols, such as SHA-256, RSA-2048, and ECC.
- It supports SNMPv3 interface.
- It supports SYSLOG over TLS.
- Communication between VNFCs can be encrypted.
- Communication between BEDBINT and subscriber database can be encrypted.

3.3. IPv6 and IPv4 support

Radiator VNF components are able to communicate both with each other and with outside systems via both IPv4 and IPv6. If the environment supports both protocols, IPv6 is preferred.

3.4. Performance and scalability

Implementing the AAA system in a virtualised environment provides good scaling possibilities and higher performance with fewer resources.

Virtualisation allows better utilisation of the hardware layer and improves the application availability. Scaling and better performance is achieved without adding more hardware resources. This means that there is no need for certain time-consuming phases, such as installing the environment, installing the applications, and testing the system.

The scaling process in Radiator VNF is modular, each VNFC is scaled separately according to the traffic needs. This approach provides possibilities to optimise the scaling and resource usage for each component. Scaling the workers is very easy because the worker nodes are stateless and they do not communicate with external systems. Scaling other VNFCs is not so straightforward but, for example, OpenStack platform provides so-called flavors to define the size of each VNFC. The flavors act as a method for dimensioning the system and have default values but it is also possible to define them according to the system needs.

The following table lists the preferred scaling scenarios for the automatic scaling of different components.

Table 1. Preferred scaling scenarios for Radiator VNF components

Component	Preferred scaling scenarios	Comments
Loadbalancer	Vertical	Horizontal scaling needs reconfiguration of connecting systems. If the system supports SDN (Software-defined Network), manual reconfiguration is not necessarily needed.
BEDBINT	Vertical	Horizontal scaling needs reconfiguration of connecting systems. If the system supports SDN, manual reconfiguration is not necessarily needed.
Worker	Horizontal or vertical	Workers are stateless and they attach automatically to the control plane so horizontal scaling is a feasible option.
Control	Horizontal or vertical	–
Management	Horizontal or vertical	Horizontal scaling needs reconfiguration of connecting systems. If the system supports SDN, manual reconfiguration is not necessarily needed.

3.5. High availability concepts

The main high availability (HA) principle of Radiator VNF is to provide redundant VNFCs which provide continuous service even if the hardware environment is not redundant. Radiator VNF is configured to provide full-time availability and it works in a cloud environment. Single points-of-failure are therefore avoided.

When a failure occurs, the self-healing process takes care of changing the failed node to a back-up node.

High availability concept is primarily covered by using redundancy in each node. Radiator VNF also supports geo-redundancy, it is possible to distribute the VNFCs onto different physical hardware. This improves reliability.

Loadbalancer

The loadbalancer can be configured to act in both active-active mode and active-passive mode.

All loadbalancer nodes are stateless, this allows fast and easy switching to another loadbalancer if problems occur. If one of the active nodes fails, a stand-by takes over its function. During the transition time of a switchover or a single node failure, a single loadbalancer can be dimensioned to be capable of taking over the traffic.

Worker

The workers are independent and stateless. All workers use active mode, this allows easy replacement. These features make the workers easy to be replaced by other workers in case of a failure. The workers are constantly observed by using KPIs. If a worker fails, the failure is noticed immediately and the worker is replaced with a new one. The replacement process does not affect the external clients in any way.

Control

The control component is not stateless. It uses redundancy together with clustering and replication to provide a reliable service. The minimal setup has 3 separate nodes, it is the minimum number of nodes clustering needs. One of the nodes is master and the others are slaves.

Management

High availability is the primary concern for the management component because it takes care of gathering log data and KPIs. High performance is a lesser issue for the management component. Using software replication provides the mechanism to ensure high availability.

3.6. Self-healing

Each Radiator VNF node has an inner mechanism for self-healing. All running processes are monitored and restarted in the case of failure. Each Chef run also ensures that each relevant configuration file is in place and correct, all needed services are running, and the state of the system is as required. For more information on Chef, see Section 4.2. Chef on page 8.

On VNF system level, the self-healing mechanism means that if a node fails, it is replaced by a new node.

4. Configuration model

This section discusses DevOps, which is an agile-style method and culture of software development and operation. Chef is also introduced. It is a DevOps tool, which is used with Radiator VNF.

4.1. DevOps in general

DevOps (development and operations) refers to a set of practices of agile-style software development. It emphasises communication between different parts of the organisation and improves the processes with automation and agility. Changes to the softwares and configurations are planned, implemented, tested, deployed, and monitored continuously and automatically. The DevOps model encourages developers to make small changes, test them, and deploy them regularly. Strong collaboration between developers and operators improve reliability, and so does a fast, automated development cycle.

Radiator VNF works in a flexible and distributed cloud environment. This kind of an environment especially requires the DevOps-style approach, otherwise it would be very difficult for a single operator to manually handle thousands of nodes. The DevOps tools give the possibility to easily create identical environments with slightly different parameters. With automated processes there is no possibility to forget to run a command or to change a parameter.

Radiator VNF supports DevOps, which means that Radiator VNF supports DevOps tooling and it is easy to deploy Radiator to configuration management systems. The whole infrastructure is also managed automatically.

4.2. Chef

Chef is a set of tools for managing configurations and it is supported by Radiator VNF.

With Chef, all infrastructure and configurations are written as Ruby code. This makes them testable like any other piece of code.

In Chef terms, the user writes cookbooks that consist of recipes. The recipes describe how the server applications are configured. Each recipe contains resources and templates, which define the state of the system. With templates, you create your business logic. The local Chef instances execute the recipes on each VNFC. VNFM (Virtualised Network Function Manager) triggers the execution as a part of lifecycle operation.

All VNFCs have their own cookbook, which includes recipes for each internal function.

For more information about Chef, see Chef documentation [https://docs.chef.io/] and Chef tutorials [https://learn.chef.io/tutorials/].

5. Abbreviations

Backend Database Interface

BEDBINT (Backend Database Interface)

Acronym: **BEDBINT**

Business Support Systems

BSS (Business Support Systems)

Acronym: **BSS**

Call Detail Record

CDR (Call Detail Record)

Acronym: **CDR**

Home Subscriber Server

HSS (Home Subscriber Server)

Acronym: **HSS**

Key Performance Indicator

KPI (Key Performance Indicator)

Acronym: **KPI**

Lightweight Directory Access Protocol

LDAP (Lightweight Directory Access Protocol)

Acronym: **LDAP**

Mobile Application Part

MAP (Mobile Application Part)

Acronym: **MAP**

Network Access Server

NAS (Network Access Server)

Acronym: **NAS**

Network Functions Virtualisation

NFV (Network Functions Virtualisation)

Acronym: **NFV**

Operations Support Systems

OSS (Operations Support Systems)

Acronym: **OSS**

Stream Control Transmission Protocol

SCTP (Stream Control Transmission Protocol)

Acronym: **SCTP**

Software-defined Network

SDN (Software-defined Network)

Acronym: **SDN**

Community-Based Simple Network Management Protocol version 2

SNMPv2c (Community-Based Simple Network Management Protocol version 2)

Acronym: **SNMPv2c**

Simple Network Management Protocol version 3

SNMPv3 (Simple Network Management Protocol version 3)

Acronym: **SNMPv3**

Simple Object Access Protocol

SOAP (Simple Object Access Protocol)

Acronym: **SOAP**

Transport Layer Security

TLS (Transport Layer Security)

Acronym: **TLS**

User Datagram Protocol

UDP (User Datagram Protocol)

Acronym: **UDP**

Virtualised AAA

vAAA (Virtualised AAA)

Acronym: **vAAA**

Virtualised Network Function

VNF (Virtualised Network Function)

Acronym: **VNF**

Virtualised Network Function Component

VNFC (Virtualised Network Function Component)

Acronym: **VNFC**

Virtualised Network Function Manager

VNFM (Virtualised Network Function Manager)

Acronym: **VNFM**