NOKIA

Network Functions Manager - Mobile (NFM-M)

Release 23R1

System Architecture Guide

3HE-19440-AAAA-TQZZA

Issue 1

March 2023

Legal notice

Nokia is committed to diversity and inclusion. We are continuously reviewing our customer documentation and consulting with standards bodies to ensure that terminology is inclusive and aligned with the industry. Our future customer documentation will be updated accordingly.

Contents

About this document

Purpose

The *NFM-M System Architecture Guide* describes the NFM-M architecture and interoperation with other systems from a high-level perspective. The audience is a technology officer, network planner, or system administrator who requires a broad technical understanding of the NFM-M system structure and design methodology.

Scope

The guide scope is limited to a description of the integral elements that are common to NFM-M components. For information about the architecture of a specific NFM-M component, or a product or appliance that integrates with the NFM-M, see the component, product, or appliance documentation.

Document support

Customer documentation and product support URLs:

- Documentation Center
- Technical support

How to comment

Documentation feedback

1 NFM-M product overview

1.1 Introduction to the NFM-M

1.1.1 Product description

The Network Function Manager - Mobile, or NFM-M, is a network management system for packet core and core enterprise networks with a focus on mobile network technology. The NFM-M is an evolution of the NFM-P that simplifies network configuration, provisioning, assurance, and management using industry standard technology including Java, XML/SOAP, REST, and open-standard application interfaces.

1.1.2 Functional highlights

The NFM-M enables and automates network management using:

- an integrated suite of applications that are adaptable to your network management and rollout use cases
- resource-control functions for real-time network control and traffic optimization
- an open, programmable platform for the automation of network operations and integration with in-house or third-party support systems

1.2 Principal NFM-M application functions

1.2.1 Network performance and fault management

NFM-M applications such as Service Supervision, Fault Management, and Network Supervision applications provide service performance, usage, and fault information for timely fault isolation and correction.

1.2.2 KPI monitoring and reporting

The NFM-M monitors network KPIs for immediate trend-based application reporting. For example, reporting agents such as the Analytics application use flow statistics to report on long-term network trends.

1.2.3 Inventory management

The NFM-M dynamically maintains a network equipment data store for SNMP-managed devices.

1.2.4 Network administration

Applications such as the NFM-M Group Manager and User Manager, define the scope and flow of network management operations.

1.3 Related documentation

1.3.1 NFM-M platform guides

The following guides are fundamental to the planning, deployment, and administration of an NFM-M system, and are intended to be consumed in the order shown:

- *NFM-M System Architecture Guide*—acquaints the reader with the NFM-M product functions, main system elements, and platform programmability; describes structural and design aspects that include system security, networking, fault tolerance
- *NFM-M Planning Guide*—describes the required NFM-M environment and platform resources, provides network and security specifications, and includes scaling and performance guidelines
- *NFM-M Installation and Upgrade Guide*—describes NFM-M deployment options and platform configuration; includes software installation, upgrade, and integration procedures
- *NFM-M System Administrator Guide*—describes administrative responsibilities such as NFM-M security, system control and configuration, NFM-M application configuration, and data backup and restore
- *NFM-M Security Hardening Guide*—describes security considerations, support for security features, and best practices for NFM-M system security.

1.3.2 Other NFM-M guides

The NFM-M documentation suite also includes the following:

- *NFM-M Release Description*—describes the features and functions in an NFM-M release; includes information such as functional enhancements and resolved product issues
- *NFM-M Release Notice*—contains special or notable information about an NFM-M release, such as deployment limitations or product enhancements not covered in the core documentation
- *NFM-M User Guide*—describes NFM-M access and use for operators; introduces the NFM-M Launchpad and applications, and describes how to obtain NFM-M software and documentation
- *NFM-M application help*—multiple guides that each describe the function and operation of an NFM-M application; example use cases are provided
- *NFM-M utility help*—multiple guides that each describe the function and operation of an NFM-M utility; example use cases are provided
- *NFM-M Glossary*—defines acronyms, initialisms, and other terms used in the NFM-M product interfaces and documentation

2 NFM-M architecture fundamentals

2.1 System design

2.1.1 Development methodology

The NFM-M architecture incorporates design considerations that include:

- open standards that promote interworking and integration with in-house or third-party systems; see Appendix A, “NFM-M technology standards” for information
- flexible internal model that accommodates product evolution
- deployment agility for adaptation to changing network scope or complexity
- programmability for dynamic management of network operations
- SSO access to NFM-M applications
- IPv4 and IPv6 support on internal, external, and mediation interfaces
- fault-tolerance safeguards that include local and geographic redundancy
- stringent internal security among components
- highly secure local and remote client access

2.1.2 Core system elements

The various components of the modular NFM-M architecture work together as a customized management solution designed to meet your current and changing network or business requirements. Components and functions can be readily added, updated, or removed, as required.

Depending on the installation options that you choose, ancillary components in an NFM-M system may provide the following:

- graphical user interfaces
- public APIs
- management domain or business logic
- mediation services

The NFM-M is designed to interface or integrate with external systems such as the following:

- remote authentication agents for user access
- other EMS or network controllers

2.2 NFM-M graphical user interfaces

2.2.1 Operator interfaces

The NFM-M provides the following graphical user interfaces, or GUIs, for network management:

- browser-based—for NFM-M applications
- Java-based—for management functions using the NFM-M

Each GUI supports application cross-launch, and provides direct access to the application documentation.

2.2.2 NFM-M user interfaces

The NFM-M Launchpad is the browser-based entry point to the browser-based NFM-M applications. The *NFM-M User Guide* describes the NFM-M Launchpad, and provides general access and usage information for operators. Because the available applications depend on the specified NFM-M installation options, each application on the NFM-M Launchpad has a separate application help document that provides comprehensive usage information.

NFM-M Client GUI

The NFM-M Java-based client GUI enables operators to perform policy-based classic device management. GUI client deployment is supported on multiple platforms; see the *NFM-M Installation and Upgrade Guide* for information.

The *NFM-M User Guide* describes how to use the Java-based client GUI to perform classic management.

2.3 NFM-M APIs

2.3.1 External / public APIs

The NFM-M architecture publicizes the following APIs for Online Support System (OSS) clients:

- NFM-M REST—for HTTP CRUD (Create, Retrieve, Update, Delete) operations on NFM-M carrier SDN, NFM-M and NFM-M application objects
- NFM-M RESTCONF—for communication with the NFM-M system and state notifications

See the *Wireless OSS Interface Developer Guide* for more information.

Classic management APIs

The following NFM-M APIs are used for access to legacy applications:

- NFM-M REST—provides access to network management functions for use cases that the NFM-M APIs do not fulfill
- NFM-M XML SOAP/JMS—for SNMP network management; see the *NFM-M XML API Developer Guide* for information

2.4 NFM-M network mediation

2.4.1 Description

The NFM-M interacts with the network using SNMP management. The mediation provides:

- network data access
- object provisioning and modification mechanisms
- network change and event notifications

The NFM-M provides FCAPS functions for classically managed networks.

Note: The NFM-M supports NE mediation using IPv4 and IPv6 concurrently; see “IP version support” in the *NFM-M Installation and Upgrade Guide* for information.

2.5 Management architecture

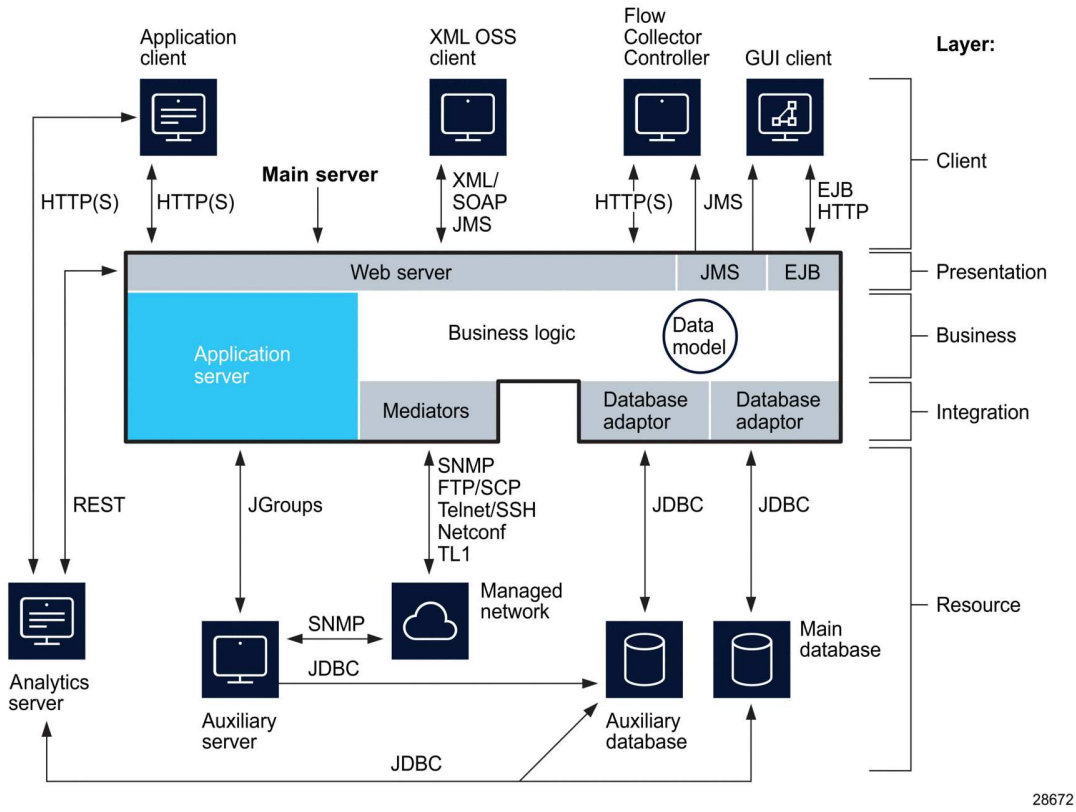
2.5.1 NFM-M system

The NFM-M, which provides the classic management functions, performs mediation for SNMP-managed devices and includes a local database for user authentication and authorization.

The NFM-M is deployed on multiple server and database stations, and supports redundant deployment using a warm standby model. The NFM-M system elements use proprietary and third-party software, and are logically organized in a framework that has the following layers, as illustrated in Figure 2-1, “NFM-M multi-layer model” (p. 10):

- resource
- integration
- business
- presentation
- client

Figure 2-1 NFM-M multi-layer model



Resource layer

The resource layer includes the network of managed NEs, the main database, and optional components like auxiliary servers and an auxiliary database. The available resources include, for example, NE configuration backups and software images, network topology information, customer service configurations, and statistics.

Integration layer

The integration layer buffers resource-layer elements from the business layer. This layer contains the mediators, which communicate with equipment in the managed network, and the database adapter. The mediator components translate messages from the business layer into the commands that are sent to the managed network. Messages from the network are processed by the mediator components and passed to the business layer. The database adapter translates business logic requests into JDBC commands, and translates JDBC responses into Java business model objects.

Business layer

The business layer contains the logic and data model for NFM-M functions. The business logic processes client requests, SNMP traps from managed NEs, and internal server events, and

performs the appropriate actions on the managed network, clients, and data model. The data model maintains information about network objects and their relationships. To support the business layer, an application server provides Java EE services.

Presentation layer

The presentation layer buffers the application logic from the client layer, and includes the following:

- web server that receives messages from OSS clients and passes them to the business layer
- application server that processes EJB method invocations from GUI clients and returns the responses from the business layer; the application server also forwards JMS messages from the business layer to GUI and OSS clients

Client layer

The GUI, OSS, and browser-based application clients comprise the client layer.

- A Java VM on a GUI client sends EJB RMIs to a main server.
- The OSS clients send XML/SOAP, or REST messages to a main server.
- Web clients use JNLP for portal access.

3 Deployment fundamentals

3.1 Overview

3.1.1 Introduction

The NFM-M is a highly distributed system that requires a number of hosts, where a host is defined as a physical or virtual processing entity that has a discrete OS instance. The functions provided by the NFM-M software are installed on the NFM-M hosts, which collectively constitute an NFM-M deployment.

Many different NFM-M deployment scenarios that employ various levels of redundancy are supported; see Chapter 6, “System redundancy and fault tolerance”, the *NFM-M Planning Guide*, and the *NFM-M Installation and Upgrade Guide* information.

3.1.2 Deployable NFM-M platform elements

The following elements, which are described in subsequent topics, comprise an NFM-M system:

- NFM-M
- Analytics servers
- Auxiliary database
- Flow Collectors and Flow Collector Controllers

The following figure shows a high-level conceptual diagram of an NFM-M deployment. A component in the diagram may require one or multiple hosts.

Figure 3-1 NFM-M deployment, abstract view

3.2 Core system components

3.2.1 NFM-M system components

A basic NFM-M system consists of the components described in the following topics. Some deployment types may require additional components, as described in the *NFM-M Planning Guide*.

Internal subcomponents

Internal subcomponents, for example, Java modules, database software, and web server software, are represented by license files in the following directory on a main server:

/opt/nsp/nfmp/server/nms/distribution/licenses

3.2.2 Main server

A main server is the central Java-based processing engine in an NFM-M system. A main server can be deployed on a dedicated station, or collocated on a station with a main database. A main server hosts an application server, JMS server, web server, protocol stack, and database adapter. Functions like statistics collection are performed by a main server only in a deployment that does not include any auxiliary servers.

3.2.3 Auxiliary server

An auxiliary server, like a main server, is a Java-based processing engine, but is an optional, scalable component that extends the system capacity for collecting data such as statistics, PCMD records, or call-trace records. An auxiliary server collects data directly from NEs, and is controlled and directed by a main server.

3.2.4 Main database

An NFM-M main database is a relational database that provides persistent storage to hold the network data repository. A main database can be deployed on a dedicated station, or collocated on a station with a main server.

3.2.5 Single-user GUI client

A single-user GUI client is a Java-based graphical interface for network operators. Single-user GUI client deployment is supported on multiple platforms.

3.2.6 Client delegate server

A client delegate server supports simultaneous GUI sessions using one client software installation. A client delegate server can host local and remote user sessions, and supports the use of a third-party remote access tool such as a Citrix gateway. Client delegate server deployment is supported on multiple platforms.

A GUI session that is opened through a client delegate server is functionally identical to a single-user client GUI session. The client delegate server locally stores the files that are unique to each user, such as the client logs and GUI preference files.

3.2.7 NFM-M analytics servers

The software associated with the Analytics installation option is deployed on one or more NFM-M analytics servers, which run business intelligence software to analyze raw and aggregated NE telemetry data collected via legacy SNMP management.

Like other NFM-M components, NFM-M analytics servers support redundant deployment; see 6.2.1 “Description” (p. 29) for information about NFM-M analytics server redundancy and other fault-tolerance mechanisms.

3.2.8 Flow Collectors and Flow Collector Controllers

An NFM-M Flow Collector is a horizontally scalable component that collects AA Cflowd or System Cflowd statistics directly from NEs. The statistics records can be forwarded to remote servers or kept in persistent storage, and are made available for processing by third-party tools or by applications such as NFM-M Analytics.

An NFM-M Flow Collector Controller is required in any deployment that includes NFM-M Flow Collectors. A Flow Collector Controller extracts the NFM-M network data model for use as a statistics-collection framework by each Flow Collector, and updates the model as the NFM-M sends JMS notifications about model updates.

For a small-scale deployment, you can collocate an NFM-M Flow Collector Controller and an NFM-M Flow Collector on one station. A small-scale deployment has a maximum of two stations, and supports the following:

- standalone—one station that hosts a Flow Collector Controller and Flow Collector, and a second station that hosts only a Flow Collector
- redundant—two stations that each host a Flow Collector Controller and Flow Collector

Redundancy

Like other NFM-M components, NFM-M Flow Collectors and NFM-M Flow Collector Controllers support redundant deployment; see 6.1.1 “Description” (p. 29) for information about NFM-M Flow Collector and Flow Collector Controller redundancy and other fault-tolerance mechanisms.

3.2.9 Auxiliary database

An auxiliary database is an optional, horizontally scalable database that expands the NFM-M storage capacity for demanding operations such as statistics collection, and performs the data aggregation required by applications such as NFM-M Analytics.

The database is deployed on one station, or in a cluster of three or more stations, depending on the scale requirement. In a multi-station auxiliary database, load balancing and data replication among the stations provide high performance and robust fault tolerance.

Auxiliary databases support DR deployment consists of two matching auxiliary database clusters that are deployed in separate data centers.

4 Security architecture

4.1 NFM-M system security

4.1.1 Security documentation

Use the *NFM-M Security Hardening Guide* as the primary reference for NFM-M security considerations.

4.1.2 TLS

The NFM-M supports the use of Transport Layer Security (TLS) throughout the NFM-M system. The NFM-M installation software includes a utility called a Public Key Infrastructure (PKI) server that you can use to automate the distribution of TLS artifacts for NFM-M components. A PKI server can generate, sign, and distribute a self-signed TLS certificate, or use a certificate from another source.

Note: The NFM-M supports only TLS v1.2; however, you can enable older TLS versions for compatibility with OSS or external systems that do not support TLS v1.2.

TLS ensures secure external communication between NFM-M clients and NFM-M applications, and among NFM-M components. The NFM-M supports the use of external TLS certificates signed by a trusted public Certificate Authority (CA), and self-signed certificates.

You determine the source and signing authority of the external TLS certificate in an NFM-M system. The internal certificate, however, is automatically created and signed by an internally generated private CA on the PKI server, so no certificate from any external CA is trusted for internal system access.

Each NFM-M serves as the central store of the following certificates for the other NFM-M components in the local datacenter:

- external, customer-provided or generated by PKI server
- internal, generated by PKI server

Other external security mechanisms

In addition, session credentials and messaging can be protected using mechanisms and protocols such as the following:

- NAT between system components
- HTTPS at the application layer for API clients
- SNMPv3 for communication with network devices

You can also enable HTTP Strict-Transport-Security, or HSTS, during system deployment, which enforces the use of HTTPS by any browser that connects to the NFM-M. See the *NFM-M Installation and Upgrade Guide* for information about enabling HSTS.

4.1.3 SELinux

The deployment of SELinux to log user operations is supported on NFM-M components.

Note: The support for SELinux enforcing mode may vary, depending on the system component. You must ensure that all components are set to the same mode.

4.2 User security and session management

4.2.1 Single-sign-on

The NFM-M single-sign-on (SSO) mechanism enables a common security framework for all supported NFM-M applications and services. NFM-M supports authentication against a local user database, and against external authentication agents such as an LDAP(S), RADIUS, or TACACS+ server.

4.3 Firewall support

4.3.1 Description

The NFM-M supports firewall deployment on all host interfaces. See the *NFM-M Planning Guide* and any specific component planning documentation, as required, for firewall port requirements and restrictions.

4.4 NFM-M system security

4.4.1 Platform security

The RHEL OS is common to all NFM-M components, including the NFM-M. The OS is protected by firewalls and stringent internal security measures that restrict access to files and functions. The NFM-M supports platform-wide mechanisms such as SELinux, which records RHEL user actions, and HTTPS Strict-Transport-Security (HSTS), which restricts client browser access. NE management communication can be secured using protocols such as SNMPv3, as well as the strong security associated with the Federal Information Processing Standards (FIPS).

See the *NFM-M Security Hardening Guide* for more information about NFM-M platform security.

4.4.2 Communication security

The NFM-M employs strict security at the session and other communication layers. Interfaces between a main server and other system components are secured using Transport Layer Security, or TLS.

Communication with a main database is secured using Oracle Network Data Encryption.

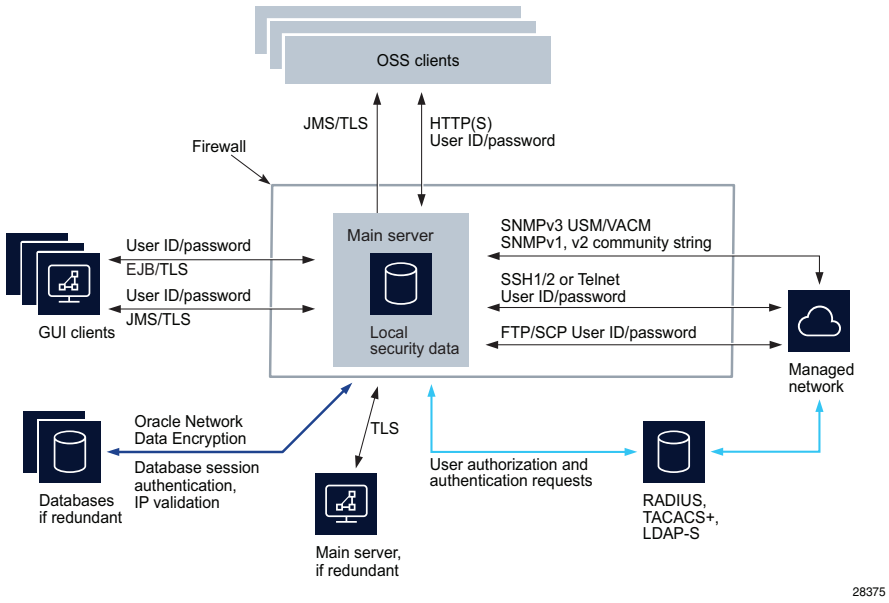
A GUI, application, or OSS client must provide user credentials for access to the NFM-M. Session credentials and messages are protected using mechanisms and protocols that include the following:

- HTTPS, as the application-layer transport for clients

- SSH, SCP, and SNMPv3 with USM or VACM, at the application layer for communication between a main server and the managed network
- NAT, at the network layer, between the following:
  - main server and single-user GUI client or client delegate server
  - main or auxiliary server and OSS client
  - main or auxiliary server and managed network
- IP validation, at the network layer, between the main or other server components and each main database

The following figure shows the NFM-M components and the security mechanisms.

Figure 4-1 NFM-M security mechanisms



4.4.3 NFM-M session management

An NFM-M operator can configure authentication, accounting, and administrative (AAA) functions using the local NFM-M security mechanisms, a third-party server, or both.

- Local NFM-M authentication is performed using a local database of users and a local security scheme.
- Supported remote authentication servers are RADIUS, TACACS+, LDAP and LDAP-S, and CSA, which have separate user lists and administration processes.

NFM-M user accounts consist of a user name, password, and an associated user group, scope of command, and span of control over network objects. User groups define user authorization levels, and control the level of access to objects such as equipment, customers, services, and alarms. An NFM-M administrator can limit the type of user access per managed NE; for example, allowing FTP access but denying console or SNMP access.

Client sessions

Client sessions use the following authentication mechanisms.

- A GUI client EJB session is authenticated using the client username and password.
- An OSS client session is authenticated using cached information from an authorization server.
- A JMS session is authenticated using the client username and password.

Database sessions

A main database is accessible through a connection that is secured by a user name and password. After each database update in response to a GUI or OSS client request, the client activity log records the request information, which includes the user name.

Secure communication between a main server and main database is available using IP validation, which is typically configured on a main database station during installation or upgrade.

Managed NE sessions

A main or auxiliary server opens CLI, FTP, SFTP and SCP sessions on managed NEs. A managed NE uses a local security database, or a third-party service such as RADIUS or TACACS+, to perform AAA functions.

SNMPv3 message authentication and authorization are handled by the USM and VACM mechanisms, which define the user authorization permissions. Older SNMP versions are authenticated using community strings. Each SNMP message is individually authenticated.

4.4.4 Network transport security

Transport-layer security is available to the network protocols that carry messages between NFM-M components.

Main server and clients

Communication between a main server and clients is performed using messaging such as the following.

- XML API clients use HTTP or HTTPS to send XML/SOAP messages, and receive notifications using JMS, which can be secured using TLS.
- REST API clients use HTTPS.
- GUI clients use the EJB interface, which can be secured using TLS.

Servers and managed NEs

When SNMPv3 is used, an authentication key using current algorithms and ciphers is included in each message and checked against the shared encryption key. SSH provides the security for a CLI session between a GUI client and a managed NE.

RSA encryption is available for communication between auxiliary servers and managed NEs; contact customer support for information.

Firewall support

The NFM-M supports firewall deployment on all server interfaces; for example, between a main server and the auxiliary servers, GUI, and OSS clients, and between a main or auxiliary server and the managed network. See the *NFM-M Planning Guide* for firewall and reserved TCP port information.

4.5 NFM-M software security summary

4.5.1 DFSEC requirement implementation

The NFM-M follows Nokia's Design For Security (DFSEC) process to ensure the security of the NFM-M product software. The DFSEC defines a framework for delivering secure products based on providing defence-in-depth, while utilizing a continuous secure delivery process based on industry-recognized standards and best practices.

The DFSEC requirements have been assembled by Nokia over several years and are based on Nokia involvement in global standards development, engagement with customers and regulators, participation in industry forums, and the collective experiences of many product development teams. The DFSEC requirements cover general software security, network security, operating system security, information security, database security, application/web server security, virtualization security, as well as authentication, authorization, and accounting. Requirements compliance is verified at various points throughout the release lifecycle from initial design to delivery.

NFM-M security testing occurs every release and utilizes tools recommended in the DFSEC process. The security testing includes the execution of vulnerability scans, web application scans, port scans, targeted robustness (aka "fuzzing") and DoS testing. Security testing is carried out using a mix of commercial and internal tools. The DFSEC also specifies the use of the Nokia Software Vulnerability Management tool, which is used to manage any vulnerabilities that have been found in the third-party libraries used within the NFM-M product; these include:

- the identification and notification of any new vulnerabilities declared against the third-party libraries used,
- tracking the vulnerability assessment and severity assignment, and
- tracking the mitigation activity necessary to eliminate the vulnerability in cases where the vulnerability is not a false positive.

Mitigation of third-party library vulnerabilities is managed through the regular release planning process with priority given to the vulnerabilities that have a critical CVSS v3 score after the vulnerability assessment.

5 NFM-M communication

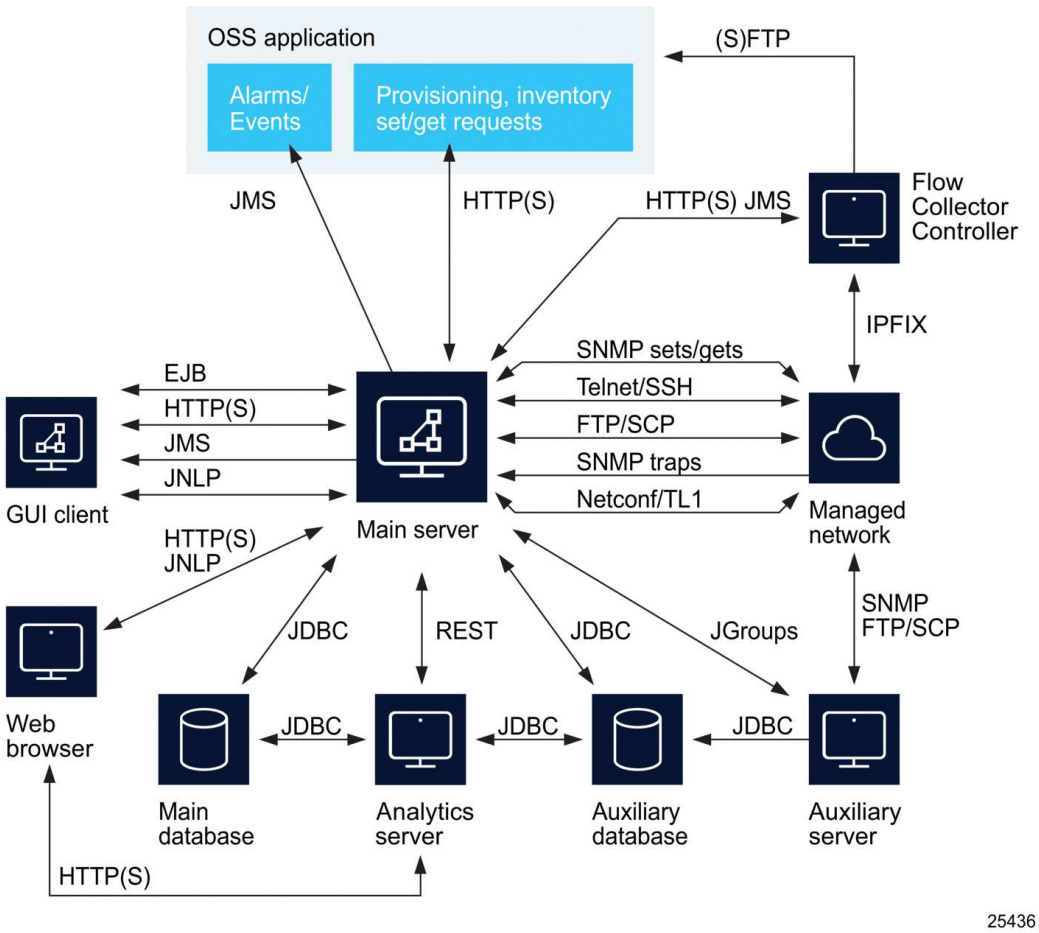
5.1 Overview

5.1.1 Introduction

Communication among NFM-M components and external systems uses IPv4 or IPv6 exclusively, with the following exceptions:

- The NFM-M can communicate with and manage a network using IPv4 and IPv6 concurrently.
- An NFM-M GUI client or browser-based application client can connect to the NFM-M using IPv4 or IPv6, regardless of the protocol version in use between the NFM-M server and database components.

Figure 5-1 NFM-M component communication



5.1.2 Servers and managed NEs

NFM-M main and auxiliary servers communicate directly with the managed network.

- A main server uses SNMP to monitor and manage network performance, and to deploy configuration changes to NEs.
- An auxiliary server polls NE MIBs for performance statistics, or collects PCMD or call-trace data. The NEs use asynchronous SNMP messages called traps to notify the NFM-M of events. UDP streaming is used by NEs for operations such as forwarding PCMD records to the NFM-M.

The CLI of a managed NE is accessible from the client GUI using Telnet or SSH.

The NFM-M uses protocols such as FTP, SFTP, and SCP to back up NE configuration data, collect NE accounting statistics, and download software to NEs.

5.1.3 Main server and clients

Client interfaces provide access to an NFM-M system and the managed network through a main server.

A main server and clients communicate in the following ways:

- GUI clients send requests to the server EJB session beans using Java RMI.
- The GUI client update function uses HTTP or HTTPS for client software updates and file downloads.
- NFM-M application clients use HTTP or HTTPS to communicate with the web service on a main server.
- A web-based GUI client communicates through a browser using JNLP.
- XML API clients send requests for processing by a main server, and subscribe to JMS topics to receive real-time event notifications. The messages between a main server and an XML API client are in XML/SOAP format, and are sent over HTTP or HTTPS. The JMS and the XML publisher service on a main server run in separate JVMs to support multiple concurrent client connections. See the *NFM-M XML API Developer Guide* for more information about the messaging between XML API clients and main servers.
- REST API clients perform network management functions and receive notifications using the NFM-M REST API. See the online REST API documentation for information.

5.1.4 Main server and main database

A main server communicates with a main database instance using JDBC, a Java API for interworking with SQL relational databases.

5.1.5 Main server and auxiliary servers

A main server sends requests to auxiliary servers. An auxiliary server notifies the main server after it finishes processing a request. If the main server fails to send a request, or all auxiliary servers are unresponsive to a request, the main server raises an alarm.

5.1.6 NFM-M integration with external systems

The NFM-M can be integrated with external network management systems for purposes such as alarm forwarding. Depending on the external system type, you can use a client GUI contextual menu option to open a session on the external system. See the *NFM-M User Guide* for information.

6 System redundancy and fault tolerance

6.1 NFM-M Flow Collector and Flow Collector Controller fault tolerance

6.1.1 Description

You can deploy redundant NFM-M Flow Collector Controllers in separate data centers to manage local Flow Collectors, or a common set of Flow Collectors that are deployed in logical proximity to the managed NEs, and not assigned to a data center. The latter configuration essentially eliminates switchover latency and any resulting data loss in the event of a Flow Collector Controller failure. The configuration also has no dependency on which Flow Collector Controller currently has the primary role.

The NFM-M Flow Collector Controllers continuously monitor the active nspOS; if the connection is lost or re-established after a connection loss, the Flow Collector Controller primary and standby roles are re-evaluated. In a DR deployment, redundancy is ensured only when the two Flow Collector Controllers are deployed in separate data centers, in order that the failure of one data center does not cause both Flow Collector Controllers to be unreachable.

In an HA or HA+DR deployment that includes the NFM-M, the primary Flow Collector Controller connects only to the primary NFM-M. An internal mechanism ensures that the primary Flow Collector Controller in a DR deployment is always in the same data center as the primary NFM-M.

Redundant NFM-M Flow Collectors can collect statistics from the same set of NEs to provide data-collection fault tolerance. The statistics that are stored in a database are de-duplicated beforehand; only one set is stored. If statistics are stored in files, duplicate files are created.

Remote statistics transfer

An NFM-M Flow Collector can transfer collected statistics files to redundant remote servers. For greater fault tolerance, redundant Flow Collectors can collect statistics from the same set of NEs and transfer the files to redundant destination servers. Such a configuration ensures that the statistics collection and transfer continue uninterrupted in the event that an NFM-M Flow Collector and a transfer destination are each unreachable.

6.2 Analytics server fault tolerance

6.2.1 Description

NFM-M analytics servers use the following fault-tolerance mechanisms to ensure that there is no single point of failure or server unavailability:

- multiple analytics servers in an active/active configuration, and a load-balancing algorithm that specifies which server responds to a client request
- access to redundantly deployed data sources

A NFM-M technology standards

A.1 NFM-M technology standards

A.1.1 Industry standards and open-standard interfaces

The NFM-M incorporates industry standards and open-standard interfaces that allow interoperation with other network monitoring and management systems. The following table lists and describes the technology standards and interfaces that are represented in the NFM-M system design.

Table A-1 Industry standards consulted in NFM-M design

Standard or interface	Description
draft-alvarez-pce-path-profiles-04	PCE path profiles
draft-dbwb-opsawg-sap-02	A network YANG model for Service Access Points (SAPs)
draft-ietf-i2rs-yang-network-topo-20	A data model for network topologies
draft-ietf-idr-bgp-ls-segment-routing-ext-16	BGP link-state extensions for segment routing
draft-ietf-idr-bgp-ls-segment-routing-msd-09	Signaling MSD using BGP-LS
draft-ietf-isis-mi-02	IS-IS Multi-Instance
draft-ietf-isis-segment-routing-extensions-04	IS-IS extensions for segment routing
draft-ietf-opsawg-l2nm	A YANG network data model for layer 2 VPN network
draft-ietf-ospf-segment-routing-extensions-04	OSPF extensions for segment routing
draft-ietf-pce-segment-routing-08	PCEP extensions for segment routing
draft-ietf-pce-stateful-pce-14	PCEP extensions for stateful PCE
draft-ietf-teas-yang-te-29	A YANG data model for traffic engineering tunnels and interfaces
ONF TR-547-TAPI v2.1.3	Reference Implementation Agreement
OpenFlow	OpenFlow Switch Specification version 1.3.1
REST	Representational State Transfer
RFC 3986	URI Generic Syntax
RFC 4655	Path Computation Element (PCE) based architecture

Table A-1 Industry standards consulted in NFM-M design (continued)

Standard or interface	Description
RFC 5101	Specification of the IP Flow Information Export (IPFIX) Protocol for the exchange of IP traffic flow information
RFC 5102	Information model for IP flow information export
RFC 5440	Path Computation Element Communication Protocol (PCEP)
RFC 5575	Dissemination of flow specification rules
RFC 6020	YANG data modelling language for NETCONF
RFC 6021	Common YANG data types
RFC 6087	Guidelines for YANG Documents
RFC 6241	Network configuration protocol (NETCONF)
RFC 6242	NETCONF over SSH
RFC 6991	Common YANG data types
RFC 7223	A YANG data model for interface management
RFC 7224	IANA interface type YANG model
RFC 7420	PCEP Management Information Base (MIB) model
RFC 7231	HTTP/1.1 Semantics and Content
RFC 7752	North-bound distribution of link-state and Traffic Engineering (TE) information using BGP
RFC 7950	YANG 1.1
RFC 7951	JSON encoding of data modelled with YANG
RFC 7952	Defining and Using Meta Data with YANG
RFC 8040	RESTCONF
RFC 8072	YANG patch media type
RFC 8281	PCEP extensions for PCE-initiated LSP setup in a stateful PCE model
RFC 8321	PCEP extensions for stateful PCE
RFC 8345	A YANG data model for network topologies
REF 8346	A YANG data model for layer 3 topologies
RFC 8476	Signaling MSD using OSPF (node MSD)
RFC 8491	Signaling MSD using IS-IS (node MSD)

Table A-1 Industry standards consulted in NFM-M design (continued)

Standard or interface	Description
RFC 8525	YANG Library
RFC 8528	YANG schema mount
RFC 8665	OSPF extensions for Segment Routing
RFC 8667	IS-IS extensions for Segment Routing
RFC 8776	Common YANG data types for traffic engineering
RFC 8944	A YANG data model for layer 2 network topologies
RFC 9181	A common YANG data model for layer 2 and layer 3 VPNs
RFC 9182	A YANG network data model for layer 3 VPNs

A.2 NFM-M technology standards

A.2.1 Industry standards

The NFM-M incorporates industry standards and open-standard interfaces. The following table lists and describes the technology standards and interfaces that are represented in the NFM-M design.

Table A-2 Industry standards consulted in NFM-M design

Standard	Description
3GPP	3rd Generation Partnership Project IRPs for CORBA R8 and SOAP/XML R8 Solution Sets
draft-grant-tacacs-02.txt	TACACS+ client
draft-ylonen-ssh-protocol-00.txt	SSH
EJB	Java EE Enterprise Java Session Bean version 2.1
HTML5	HyperText Markup Language 5, for NFM-M applications
HTTP(S)	HyperText Transfer Protocol (Secure) version 1.1
ITU-T X.721	SMI
ITU-T X.734	Event report management function
Java SE	Java Standard Edition version 8
JBOSS EAP	Java Bean Open Source Software Enterprise Application Platform version 7
JMS	Java Message Service version 1.1
JSON	ECMA-404 JavaScript Object Notation Data Interchange Format
JS/ECMAScript 5	ECMA-262 ECMA Script Language Specification

Table A-2 Industry standards consulted in NFM-M design (continued)

Standard	Description
M.3100/3120	Equipment and connection models
MTOSI	Compliance of generic network objects, inventory retrieval, and JMS over XML
RFC 0959	FTP
RFC 1213	SNMPv1
RFC 1738	Uniform Resource Locators (URL)
RFC 2138	RADIUS client 2618
RFC 3411-3415	SNMPv3
RFC 3416	SNMPv2c
RFC 5246	The Transport Layer Security (TLS) Protocol
RFC 6241	Network Configuration Protocol (NETCONF)
SOAP	W3C SOAP 1.2
TMF 509/613	Network connectivity model
TR-069	TR-069 (Amendment 1) by way of the Home Device Manager
XML	W3C XML 1.0
	W3C Namespaces in XML
	W3C XML schemas

The following standards are considered in the NFM-M GUI design:

- Sun Microsystems, *Java Look and Feel Design Guidelines*, Addison-Wesley Publishing Company, Reading, Massachusetts 1999
- ANSI T1.232-1996, *Operations, Administration, and Provisioning (OAM&P)- G Interface Specifications for Use with the Telecommunications Management Network (TMN)*
- Telcordia (Bell Core) GR-2914-CORE Sept. 98, *Human Factors Requirements for Equipment to Improve Network Integrity*
- Telcordia (Bell Core) GR-826-CORE, June 1994, Issue 1, Section 10.2 of OTGR, *User Interface Generic Requirements for Supporting Network Element Operations*
- ITU-T Recommendation Z.361 (02/99), *Design guidelines for Human- Computer Interfaces (HCI) for the management of telecommunications networks*
- ETSI EG 201 204 v1.1.1 (1997-05), *Human Factors (HF); User Interface design principles for the Telecommunications Management Network (TMN) applicable to the “G” Interface*
- 3GPP 32-series R8 specification, published December, 2009.