NETCONF Overview

The NETCONF protocol defines a simple mechanism through which a network device can be managed, configuration data information can be retrieved, and new configuration data can be uploaded and manipulated.

A protocol that defines configuration datastores and a set of Create, Retrieve, Update, Delete (CRUD) operations that can be used to access these datastores. There are three NETCONF datastores – candidate, running & startup.

NETCONF uses a simple RPC-based mechanism to facilitate communication between a client and a server. The client can be a script or application typically running as part of a network manager. The server is typically a network device.

Configuration data: The set of writable data that is required to transform a system from its initial default state into its current state.

Datastore: A conceptual place to store and access information. A datastore might be implemented, for example, using files, a database, flash memory locations, or combinations thereof.

Configuration datastore: The datastore holding the complete set of configuration data that is required to get a device from its initial default state into a desired operational state.

State data: The additional data on a system that is not configuration data such as read-only status information and collected statistics.

Candidate configuration datastore: A configuration datastore that can be manipulated without impacting the device’s current configuration and that can be committed to the running configuration datastore. Not all devices support a candidate configuration datastore.

Running configuration datastore: A configuration datastore holding the complete configuration currently active on the device. The running configuration datastore always exists.

Startup configuration datastore: The configuration datastore holding the configuration loaded by the device when it boots. Only present on devices that separate the startup configuration datastore from the running configuration datastore.

The <rpc> element is used to enclose a NETCONF request sent from the client to the server.

The <rpc-reply> message is sent in response to an <rpc> message.

The NETCONF protocol uses a remote procedure call (RPC) paradigm. A client encodes an RPC in XML and sends it to a server using a secure, connection-oriented session. The server responds with a reply encoded in XML.

NETCONF allows a client to discover the set of protocol extensions supported by a server. These “capabilities” permit the client to adjust its behavior to take advantage of the features exposed by the device.

NETCONF can be used in concert with XML-based transformation technologies, such as XSLT, to provide a system for automated generation of full and partial configurations. The system can query one or more databases for data about networking topologies, links, policies, customers, and services. This data can be transformed using one or more XSLT scripts from a task-oriented, vendor-independent data schema into a form that is specific to the vendor, product, operating system, and software release. The resulting data can be passed to the device using the NETCONF protocol.