

NETCONF Working Group  
Internet-Draft  
Updates: [4253](#) (if approved)  
Intended status: Standards Track  
Expires: June 8, 2015

K. Watsen  
Juniper Networks  
December 5, 2014

**NETCONF Call Home and RESTCONF Call Home  
draft-ietf-netconf-call-home-02**

Abstract

This document presents NETCONF Call Home and RESTCONF Call Home, which respectively enable a NETCONF/RESTCONF server to initiate a secure connection to a NETCONF/RESTCONF client.

Status of This Memo

This Internet-Draft is submitted in full conformance with the provisions of [BCP 78](#) and [BCP 79](#).

Internet-Drafts are working documents of the Internet Engineering Task Force (IETF). Note that other groups may also distribute working documents as Internet-Drafts. The list of current Internet-Drafts is at <http://datatracker.ietf.org/drafts/current/>.

Internet-Drafts are draft documents valid for a maximum of six months and may be updated, replaced, or obsoleted by other documents at any time. It is inappropriate to use Internet-Drafts as reference material or to cite them other than as "work in progress."

This Internet-Draft will expire on June 8, 2015.

Copyright Notice

Copyright (c) 2014 IETF Trust and the persons identified as the document authors. All rights reserved.

This document is subject to [BCP 78](#) and the IETF Trust's Legal Provisions Relating to IETF Documents (<http://trustee.ietf.org/license-info>) in effect on the date of publication of this document. Please review these documents carefully, as they describe your rights and restrictions with respect to this document. Code Components extracted from this document must include Simplified BSD License text as described in Section 4.e of the Trust Legal Provisions and are provided without warranty as described in the Simplified BSD License.

## Table of Contents

<a href="#">1.</a>	Introduction . . . . .	<a href="#">2</a>
<a href="#">1.1.</a>	Motivation . . . . .	<a href="#">3</a>
<a href="#">1.2.</a>	Requirements Terminology . . . . .	<a href="#">3</a>
<a href="#">1.3.</a>	Applicability Statement . . . . .	<a href="#">3</a>
<a href="#">1.4.</a>	Update to <a href="#">RFC 4253</a> . . . . .	<a href="#">4</a>
<a href="#">2.</a>	The NETCONF Server or RESTCONF Server . . . . .	<a href="#">4</a>
<a href="#">2.1.</a>	Protocol Operation . . . . .	<a href="#">4</a>
<a href="#">2.2.</a>	Configuration Data Model . . . . .	<a href="#">5</a>
<a href="#">3.</a>	The NETCONF Client or RESTCONF Client . . . . .	<a href="#">5</a>
<a href="#">3.1.</a>	Protocol Operation . . . . .	<a href="#">5</a>
<a href="#">3.2.</a>	Server Identification and Verification . . . . .	<a href="#">5</a>
<a href="#">4.</a>	Security Considerations . . . . .	<a href="#">7</a>
<a href="#">5.</a>	IANA Considerations . . . . .	<a href="#">8</a>
<a href="#">6.</a>	Acknowledgements . . . . .	<a href="#">8</a>
<a href="#">7.</a>	References . . . . .	<a href="#">8</a>
<a href="#">7.1.</a>	Normative References . . . . .	<a href="#">9</a>
<a href="#">7.2.</a>	Informative References . . . . .	<a href="#">10</a>
<a href="#">Appendix A.</a>	Change Log . . . . .	<a href="#">11</a>
<a href="#">A.1.</a>	00 to 01 . . . . .	<a href="#">11</a>
<a href="#">A.2.</a>	01 to 02 . . . . .	<a href="#">11</a>

**[1.](#) Introduction**

This document presents NETCONF Call Home and RESTCONF Call Home, which respectively enable a NETCONF/RESTCONF server to initiate a secure connection to a NETCONF/RESTCONF client. The NETCONF protocol is described in [[RFC6241](#)] and the RESTCONF is described in [[draft-ietf-netconf-restconf](#)].

Both NETCONF Call Home and RESTCONF Call Home preserve the SSH [[RFC4253](#)] and TLS [[RFC5246](#)] transport roles, as when compared to standard NETCONF and RESTCONF connections. Specifically, regardless if call home is used or not, the NETCONF server is always the SSH or TLS server, and the RESTCONF server is always the TLS server.

Ensuring consistency in the SSH and TLS roles is both necessary and desirable. Ensuring consistency is necessary, for the SSH protocol, as SSH channels and subsystems can only be opened on the SSH server, as is needed to support NETCONF over SSH [[RFC6242](#)]. Ensuring consistency is desirable, for both the SSH and TLS protocols, as it conveniently leverages infrastructure that may be deployed for host-key or certificate verification and user authentication.

Watsen

Expires June 8, 2015

[Page 2]

### **1.1. Motivation**

Call home is generally useful for both the initial deployment and on-going management of networking elements. Here are some scenarios enabled by call home:

- o The network element may proactively call home after being powered on for the first time in order to register itself with its management system.
- o The network element may access the network in a way that dynamically assigns it an IP address and it doesn't register its assigned IP address to a mapping service.
- o The network element may be configured in "stealth mode" and thus doesn't have any open ports for the management system to connect to.
- o The network element may be deployed behind a firewall that doesn't allow management access to the internal network.
- o The network element may be deployed behind a firewall that implements network address translation (NAT) for all internal network IP addresses, thus complicating the ability for a management system to connect to it.
- o The operator may prefer to have network elements initiate management connections, believing it is easier to secure one open-port in the data center than to have an open port on each network element in the network.

Having call home for the NETCONF protocol, and the RESTCONF protocol by extension, is particularly useful as NETCONF is the recommended protocol for configuration [[iesg-statement](#)], which is needed for provisioning workflows.

### **1.2. Requirements Terminology**

The key words "MUST", "MUST NOT", "REQUIRED", "SHALL", "SHALL NOT", "SHOULD", "SHOULD NOT", "RECOMMENDED", "MAY", and "OPTIONAL" in this document are to be interpreted as described in [RFC 2119](#) [[RFC2119](#)].

### **1.3. Applicability Statement**

The techniques described in this document are suitable for network management scenarios such as the ones described in [Section 1.1](#). However, these techniques SHOULD only be used for NETCONF Call Home and RESTCONF Call Home, as described in this document.



The reason for this restriction is that different protocols have different security assumptions. The NETCONF and RESTCONF protocols require clients and servers to verify the identity of the other party before starting the NETCONF/RESTCONF protocol ([section 2.2 of \[RFC6241\]](#) and section FIXME of [\[draft-ietf-netconf-restconf\]](#)).

This contrasts with the base SSH and TLS protocols, which do not require programmatic verification of the other party ([section 9.3.4 of \[RFC4251\]](#), [section 4 of \[RFC4252\]](#), and [section 7.3 of \[RFC5246\]](#)). In such circumstances, allowing the SSH/TLS server to contact the SSH/TLS client would open new vulnerabilities. Any use of call home with SSH/TLS for purposes other than NETCONF or RESTCONF will need a thorough, contextual security analysis.

#### **1.4. Update to [RFC 4253](#)**

This document updates the SSH Transport Layer Protocol [\[RFC4253\]](#) only by removing the "The client initiates the connection" statement made in [Section 4](#) (Connection Setup). This document assumes that the reference to "connection" refers to the underlying transport connection (e.g., TCP), which the server initiates in a call home connection. Security implications related to this change are discussed in Security Considerations ([Section 4](#)).

## **2. The NETCONF Server or RESTCONF Server**

### **2.1. Protocol Operation**

- o The NETCONF/RESTCONF server initiates a TCP connection to the NETCONF/RESTCONF client on one of the IANA-assigned ports for call home (PORT-X for netconf-ch-ssh, PORT-Y for netconf-ch-tls, or PORT-Z for restconf-ch-tls).
- o The TCP connection is accepted and a TCP session is established.
- o Using this TCP session, the NETCONF/RESTCONF server immediately starts either the SSH-server or the TLS-server protocol, depending on which port is connected. The server MUST start the SSH-server protocol when port PORT-X is connected or the TLS-server protocol when either port PORT-Y or PORT-Z is connected. The SSH-server and TLS-server protocols are described by [\[RFC4253\]](#) and [\[RFC5246\]](#) respectively.
- o When port PORT-X or PORT-Y is connected, the NETCONF protocol proceeds normally for SSH and TLS, as defined in [\[RFC6242\]](#) and [\[RFC5539\]](#) respectively. When port PORT-Z is connected, the RESTCONF protocol proceeds normally for TLS, as defined in [\[draft-ietf-netconf-restconf\]](#).



## **2.2. Configuration Data Model**

How to configure a NETCONF or RESTCONF server to initiate a call home connection is outside the scope of this document, as implementations can support this protocol using proprietary configuration data models. That said, a YANG [[RFC6020](#)] model for configuring both NETCONF Call Home and RESTCONF Call Home is provided in [[draft-ietf-netconf-server-model](#)].

## **3. The NETCONF Client or RESTCONF Client**

### **3.1. Protocol Operation**

- o The NETCONF/RESTCONF client listens for TCP connections on one or all of the IANA-assigned ports for NETCONF Call Home (PORT-X for netconf-ch-ssh and PORT-Y for netconf-ch-tls) or RESTCONF Call Home (PORT-Z for restconf-ch-tls).
- o The NETCONF/RESTCONF client accepts an incoming TCP connection and a TCP session is established.
- o Using this TCP session, the NETCONF/RESTCONF client immediately starts either the SSH-client or the TLS-client protocol, depending on which port is connected. The client MUST start the SSH-client protocol when port PORT-X is connected and the TLS-client protocol when port PORT-Y or PORT-Z is connected. The SSH-client and TLS-client protocols are described by [[RFC4253](#)] and [[RFC5246](#)] respectively.
- o When port PORT-X or PORT-Y is connected, the NETCONF protocol proceeds normally for SSH and TLS, as defined in [[RFC6242](#)] and [[RFC5539](#)] respectively. When port PORT-Z is connected, the RESTCONF protocol proceeds normally for TLS, as defined in [[draft-ietf-netconf-restconf](#)].

### **3.2. Server Identification and Verification**

Under normal circumstances, a NETCONF/RESTCONF client initiates the connection to the NETCONF/RESTCONF server. This action provides essential input to verify the NETCONF/RESTCONF server's identity. For instance, when using TLS, the input can be compared to the domain names and IP addresses encoded in X.509 certificates. Similarly, when using SSH, the input can be compared to information persisted previously.

However, when receiving a call home connection, the NETCONF/RESTCONF client does not have any context leading it to know the connection is from a particular NETCONF/RESTCONF server. Thus the NETCONF/RESTCONF





client must derive the NETCONF/RESTCONF server's identity using information provided by the network and the NETCONF/RESTCONF server itself. This section describes strategies a NETCONF/RESTCONF client can use to identify a NETCONF/RESTCONF server.

In addition to identifying a NETCONF/RESTCONF server, a NETCONF/RESTCONF client must also be able to verify the NETCONF/RESTCONF server's credentials. Verifying a NETCONF/RESTCONF server's credentials is necessary under normal circumstances but, due to call home being commonly used for newly deployed NETCONF/RESTCONF servers, how to verify its credentials the very first time becomes a prominent concern. Therefore, this section also describes strategies a NETCONF/RESTCONF client can use to verify a NETCONF/RESTCONF server's credentials.

The first information a NETCONF/RESTCONF client learns from a call rhHome connection is the IP address of the NETCONF/RESTCONF server, as provided by the source address of the TCP connection. This IP address could be used as an identifier directly, but doing so would only work in networks that use known static addresses, in which case a standard NETCONF/RESTCONF connection would have worked just as well. Due to this limited use, it is not recommended to identify a NETCONF/RESTCONF server based on its source IP address.

The next information a NETCONF/RESTCONF client learns is provided by the NETCONF/RESTCONF server in the form of a host-key or a certificate, for the SSH and TLS protocols respectively. Without examining the contents of the host-key or certificate, it is possible to form an identity for the NETCONF/RESTCONF server using it directly (e.g., a fingerprint), since each NETCONF/RESTCONF server is assumed to have a statistically unique public key, even in virtualized environments. This strategy also provides a mechanism to verify the NETCONF/RESTCONF server, in that a secure connection can only be established with the NETCONF/RESTCONF server having the matching private key. This strategy is commonly implemented by SSH clients, and could be used equally well by TLS-based clients, such as may be required when the NETCONF/RESTCONF servers have self-signed certificates. This strategy is viable and useful when the NETCONF/RESTCONF servers call home using either SSH with standard RSA/DSA host-keys, or using TLS with self-signed certificates.

Yet another option for identifying a NETCONF/RESTCONF server is for its host key or certificate to encode its identity directly (e.g., within the "Subject" field). However, in order to trust the content encoded within a host-key or certificate, it must be signed by a certificate authority trusted by the NETCONF/RESTCONF client. This strategy's use of PKI enables a NETCONF/RESTCONF client to transparently authenticate NETCONF/RESTCONF servers, thus eliminating

Watsen

Expires June 8, 2015

[Page 6]

the need for manual authentication, as required by the previously discussed strategies. Elimination of manual steps is needed to achieve scalable solutions, however one can claim that this merely pushes equivalent work to provisioning the NETCONF/RESTCONF servers with signed credentials. This assessment is accurate in general, but not in the case where the manufacturer itself provisions the credentials, such as is described by [\[Std-802.1AR-2009\]](#). When NETCONF/RESTCONF servers are pre-provisioned this way, NETCONF/RESTCONF clients can transparently authenticate NETCONF/RESTCONF servers using just the manufacturer's trust anchor and a list of expected NETCONF/RESTCONF server identifiers, which could be provided along with shipping information. This strategy is recommended for all deployment scenarios.

In discussing the use of certificates, it is worth noting that TLS uses X.509 certificates by default. However, to use X.509 certificates with SSH, both the NETCONF client and server must support [\[RFC6187\]](#).

#### **4. Security Considerations**

The security considerations described throughout [\[RFC6242\]](#) and [\[RFC5539\]](#), and by extension [\[RFC4253\]](#), [\[RFC5246\]](#), and [\[draft-ietf-netconf-restconf\]](#) apply here as well.

This RFC deviates from standard SSH and TLS usage by having the SSH/TLS server initiate the underlying TCP connection. For SSH, [\[RFC4253\]](#) says "the client initiates the connection", whereas for TLS, [\[RFC5246\]](#) says it is layered on top of "some reliable transport protocol" without further attribution.

Not having the SSH/TLS client initiate the TCP connection means that it does not have a preconceived notion of the SSH/TLS server's identity, and therefore must dynamically derive one from information provided by the network or the SSH/TLS server itself. Security Considerations for strategies for this are described in [Section 3.2](#).

An attacker could DoS the NETCONF/RESTCONF client by having it perform computationally expensive operations, before deducing that the attacker doesn't possess a valid key. This is no different than any secured service and all common precautions apply (e.g., blacklisting the source address after a set number of unsuccessful login attempts).



## **5. IANA Considerations**

This document requests that IANA assigns three TCP port numbers in the "Registered Port Numbers" range with the service names "netconf-ch-ssh", "netconf-ch-tls", and "restconf-ch-tls". These ports will be the default ports for NETCONF Call Home and RESTCONF Call Home protocols. Below is the registration template following the rules in [\[RFC6335\]](#).

Service Name:            netconf-ch-ssh  
Transport Protocol(s):   TCP  
Assignee:                IESG <iesg@ietf.org>  
Contact:                 IETF Chair <chair@ietf.org>  
Description:             NETCONF Call Home (SSH)  
Reference:                RFC XXXX  
Port Number:             PORT-X

Service Name:            netconf-ch-tls  
Transport Protocol(s):   TCP  
Assignee:                IESG <iesg@ietf.org>  
Contact:                 IETF Chair <chair@ietf.org>  
Description:             NETCONF Call Home (TLS)  
Reference:                RFC XXXX  
Port Number:             PORT-Y

Service Name:            restconf-ch-tls  
Transport Protocol(s):   TCP  
Assignee:                IESG <iesg@ietf.org>  
Contact:                 IETF Chair <chair@ietf.org>  
Description:             RESTCONF Call Home (TLS)  
Reference:                RFC XXXX  
Port Number:             PORT-Z

## **6. Acknowledgements**

The author would like to thank for following for lively discussions on list and in the halls (ordered by last name): Andy Bierman, Martin Bjorklund, Mehmet Ersue, Wes Hardaker, Stephen Hanna, David Harrington, Jeffrey Hutzelman, Radek Krejci, Alan Luchuk, Mouse, Russ Mundy, Tom Petch, Peter Saint-Andre, Joe Touch, Sean Turner, Bert Wijnen.

## **7. References**



## **7.1. Normative References**

- [RFC2119] Bradner, S., "Key words for use in RFCs to Indicate Requirement Levels", [BCP 14](#), [RFC 2119](#), March 1997.
- [RFC4251] Ylonen, T. and C. Lonvick, "The Secure Shell (SSH) Protocol Architecture", [RFC 4251](#), January 2006.
- [RFC4252] Ylonen, T. and C. Lonvick, "The Secure Shell (SSH) Authentication Protocol", [RFC 4252](#), January 2006.
- [RFC4253] Ylonen, T. and C. Lonvick, "The Secure Shell (SSH) Transport Layer Protocol", [RFC 4253](#), January 2006.
- [RFC5246] Dierks, T. and E. Rescorla, "The Transport Layer Security (TLS) Protocol Version 1.2", [RFC 5246](#), August 2008.
- [RFC5539] Badra, M., "NETCONF over Transport Layer Security (TLS)", [RFC 5539](#), May 2009.
- [RFC6020] Bjorklund, M., "YANG - A Data Modeling Language for the Network Configuration Protocol (NETCONF)", [RFC 6020](#), October 2010.
- [RFC6187] Igoe, K. and D. Stebila, "X.509v3 Certificates for Secure Shell Authentication", [RFC 6187](#), March 2011.
- [RFC6241] Enns, R., Bjorklund, M., Schoenwaelder, J., and A. Bierman, "Network Configuration Protocol (NETCONF)", [RFC 6241](#), June 2011.
- [RFC6242] Wasserman, M., "Using the NETCONF Protocol over Secure Shell (SSH)", [RFC 6242](#), June 2011.
- [RFC6335] Cotton, M., Eggert, L., Touch, J., Westerlund, M., and S. Cheshire, "Internet Assigned Numbers Authority (IANA) Procedures for the Management of the Service Name and Transport Protocol Port Number Registry", [BCP 165](#), [RFC 6335](#), August 2011.
- [[draft-ietf-netconf-restconf](#)] Bierman, A., Bjorklund, M., and K. Watsen, "RESTCONF Protocol", [draft-ietf-netconf-restconf-04](#) (work in progress), 2014.





## **7.2.   Informative References**

[Std-802.1AR-2009]

IEEE SA-Standards Board, "IEEE Standard for Local and metropolitan area networks - Secure Device Identity", December 2009, <<http://standards.ieee.org/findstds/standard/802.1AR-2009.html>>.

[[draft-ietf-netconf-server-model](#)]

Watsen, K. and J. Schoenwaelder, "NETCONF Server Configuration Model", 2014, <<http://tools.ietf.org/html/draft-ietf-netconf-server-model>>.

[iesg-statement]

"Writable MIB Module IESG Statement", March 2014, <<https://www.ietf.org/iesg/statement/writable-mib-module.html>>.



## [Appendix A.](#) Change Log

### [A.1.](#) 00 to 01

- o The term "TCP connection" is now used throughout.
- o The terms "network element" and "management system" are now only used in the Motivation section.
- o Restructured doc a little to create an Introduction section.
- o Fixed reference in Applicability Statement so it would work equally well for SSH and TLS.
- o Fixed reported odd wording and three references.

### [A.2.](#) 01 to 02

- o Added call home support for the RESTCONF protocol.
- o Fixed paragraph 3 of Security Considerations to equally apply to the TLS protocol.

#### Author's Address

Kent Watsen  
Juniper Networks

EMail: kwatsen@juniper.net

