NETCONF Working Group
Internet-Draft
Updates: 4253 (if approved)
Intended status: Standards Track
Expires: February 20, 2015

K. Watsen
Juniper Networks

August 19, 2014

NETCONF Call Home
draft-ietf-netconf-call-home-00

Abstract

This document presents NETCONF Call Home, which enables a NETCONF server to initiate a secure connection to the NETCONF client. NETCONF Call Home supports both the SSH and TLS transports, and does so in a way that preserves the SSH and TLS roles when compared to standard NETCONF over SSH or TLS connections.

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 February 20, 2015.

Copyright Notice

Table of Contents

1. Motivation 2
2. Requirements Terminology 3
3. Applicability Statement 3
4. Update to RFC 4253 3
5. Overview 3
6. Operation 4
7. NETCONF Server Identification and Verification 5
8. Configuration Data Model 6
9. Security Considerations 6
10. IANA Considerations 7
11. Acknowledgements 7
12. References 7
12.1. Normative References 8
12.2. Informative References 8

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:

- The network element may proactively call home after being powered on for the first time to register itself with its management system.

- 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 addressed to a mapping service.

- The network element may be configured in "stealth mode" and thus doesn't have any open ports for the management system to connect to.

- The network element may be deployed behind a firewall that doesn't allow management access to the internal network.

- 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.

- 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 NETCONF is particularly useful as NETCONF is the recommended protocol for configuration [iesg-statement], which is needed for provisioning workflows.

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].

3. Applicability Statement

The techniques described in this document are suitable for network management scenarios such as the ones described in section 3. However, these techniques SHOULD only be used for a NETCONF server to initiate a connection to a NETCONF client, as described in this document.

The reason for this restriction is that different protocols have different security assumptions. The NETCONF transport specifications require NETCONF clients and servers to verify the identity of the other party before starting the NETCONF protocol (section 6 of [RFC6242]).

This contrasts with the base SSH and TLS protocols, which do not require programmatic verification of the other party (e.g., section 9.3.4 of [RFC4251] and section 4 of [RFC4252]). 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 will need a thorough, contextual security analysis.

4. Update to RFC 4253

This document updates the SSH Transport Layer Protocol [RFC4253] only by removing the restriction in Section 4 (Connection Setup) of [RFC4253] that the SSH client initiates the connection. Security implications related to this change are discussed in Security Considerations (Section 9).

5. Overview

The same technique is used to enabled call home for both the SSH and TLS transports. The technique is to have the network element initiate a TCP connection to its remote peer. The remote peer then

uses the established TCP connection to initiate either the SSH or TLS protocols. In this way, the network element is always the SSH or TLS server, regardless if call home is used or not.

Enabling the network element to maintain the role of SSH or TLS server is both necessary and desirable. It is necessary for the SSH protocol, as SSH channels and subsystems can only be opened on the SSH server. It 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.

6. Operation

The NETCONF server's perspective (e.g., the network element)

- The NETCONF server initiates a TCP connection to the NETCONF client on one of the IANA-assigned ports for NETCONF Call Home (YYYY or ZZZZ).
- The TCP connection is accepted and a TCP session is established.
- Using this TCP connection, the NETCONF server immediately starts either the SSH-server or TLS-server protocol. That is, the next message sent on the TCP stream is the initial message defined for these protocols, per [RFC4253] or [RFC5246].
- The NETCONF protocol proceeds normally for SSH and TLS, as defined in [RFC4253] and [RFC5539] respectively.

The NETCONF client's perspective (e.g., the management system)

- The NETCONF client listens for TCP connections on one or both of the IANA-assigned ports for NETCONF Call Home port (YYYY and/or ZZZZ).
- The NETCONF client accepts an incoming TCP connection and a TCP session is established.
- Using this TCP connection, the NETCONF client immediately starts either the SSH-server or TLS-server protocol. That is, the next message sent on the TCP stream is the initial message defined for these protocols, per xref target="RFC4253"/> or [RFC5246].
- The NETCONF protocol proceeds normally for SSH and TLS, as defined in [RFC4253] and [RFC5539] respectively.

7. NETCONF Server Identification and Verification

Under normal circumstances, a management system initiates the NETCONF connection to the network element. This action provides essential input to verify the network element'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 NETCONF Call Home connection, the management system does not have an expectation for the connection to be from a specific network element, and thus must derived the network element's identity using information provided by the network and the network element itself. This section describes strategies a management system can use to identify a network element.

In addition to identifying a network element, a management system must also be able to verify the network element's credentials. Verifying a network element's credentials is of course necessary under normal circumstances, but due to call home being commonly used for newly deployed network elements, how to verify its credentials the very first time becomes a critical concern. Therefore, this section also describes strategies a management system can use to verify a network element's credentials.

The first information a management system learns from a NETCONF Call Home connection is the IP address of the remote peer, as provided as the source address of the TCP connection. This IP address could be used as an identifier, but it can only work in networks that use known static addresses, in which case having the management system initiate the NETCONF connection to the network element would have worked just as well. Due to its limited use, it is not recommended to identify a network element based on its source IP address.

The next information a management system learns is provided by the network element 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 network element using it (e.g., a fingerprint), since each network element is assumed to have a statistically unique public key, even in virtualized environments. This strategy also provides a mechanism to verify the network element, in that a secure connection can only be established with the network element having the matching private key. This strategy is commonly implemented by SSH clients, but could be used equally well by TLS-based clients, such as may be required when the network elements have self-signed certificates. This strategy is viable and useful when the network elements call

home using either SSH with standard RSA/DSA host-keys, or using TLS with self-signed certificates.

Yet another option for identifying a network element 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 trust anchor known to the management application. This strategy enables a management application to transparently authenticate network elements, thus eliminating the need for manual authentication, as required by the previously discussed strategy. Elimination of manual steps is needed to achieve scalable solutions, however one can claim that this merely pushes equivalent work to provisioning the network elements 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 network elements are pre-provisioned this way, management applications can transparently authenticate network elements using just the manufacturer's trust anchor and a list of expected network element identifiers, which could be provided along with shipping information.

TLS uses X.509 certificates by default. To use X.509 certificates with SSH, implementations should reference [RFC6187].

8. Configuration Data Model

How to configure a network element to initiate a NETCONF 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 configuring NETCONF Call Home is provided in [draft-ietf-netconf-server-model].

9. Security Considerations

The security considerations described throughout [RFC6242] and [RFC5539], and by extension [RFC4253] and [RFC5246], 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.

For SSH, not having the SSH client initiate the TCP connection means that it does not have a preconceived notion of the SSH server's identity, and therefore must dynamically derive one from information

provided by the network or the SSH server itself. Security Considerations for strategies for this are described in Section 7.

An attacker could DoS the management application by having it perform computationally expensive operations, before deducing that the attacker doesn't posses 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).

10. IANA Considerations

This document requests that IANA assigns two TCP port numbers in the "Registered Port Numbers" range with the service names "netconf-ch-ssh" and "netconf-ch-tls". These ports will be the default ports for NETCONF Call Home protocol when using SSH and TLS respectively. 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:            YYYY

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:            ZZZZ
```

11. 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.

12. References

12.1. 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.

[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.

12.2. 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>.

Author's Address

Kent Watsen
Juniper Networks

EMail: kwatsen@juniper.net