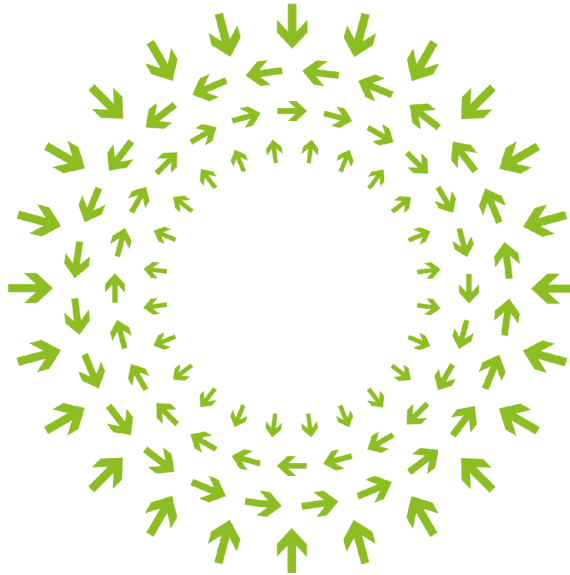




OPEN

Compute Project

WAN Router Secure Management

0.3.0

Base Specification

Effective September 1, 2025

Authors: Google, Microsoft, Meta.

Version History

Date	Version #	Author	Description
18 JUN 2025	0.1.0		Initial Release
20 AUG 2025	0.2.0		Incorporated feedback from SC review
4 SEP 2025	0.3.0		Additional updates from SC review; submitted for SC approval

Table of Contents

Version History.....	2
1. Executive Summary.....	4
2. License.....	4
2.1. Open Web Foundation (OWF) CLA.....	5
3. Introduction.....	7
3.1. Definitions.....	7
3.2. Threat Analysis.....	8
3.2.1. Counterfeit Devices.....	8
3.2.2. Counterfeit Software.....	8
3.2.3. Unauthorised API access from external systems.....	8
3.2.4. Expansion of privileges from external systems.....	9
4. Processes.....	9
4.1. Secure Device Turn-Up.....	9
4.2. Certificate Installation and Rotation.....	11
4.3. Service Authentication, and Authorization.....	11
4.4. Granular (Path-based) Authorization.....	12
4.5. Accounting.....	12
5. Hardware Requirements.....	13
5.1. Hardware RoT.....	13
6. Software Requirements.....	13
6.1. Summary of Required API Support.....	13
7. Compliance.....	13
8. Acknowledgements.....	14
9. Compliance with OCP Tenets.....	14
9.1. Openness.....	14
9.2. Efficiency.....	15
9.3. Impact.....	15
9.4. Scale.....	15
9.5. Sustainability.....	16
10. References.....	17

1. Executive Summary

This specification standardises secure turn-up and management of network elements (routers, switches) that are deployed in Wide Area Networks (WANs). With the continued growth of compute infrastructure, especially that driven by Cloud and AI/ML applications, demands on WANs continue to grow exponentially. This leads to a significant requirement for additional device deployments at a rapid rate, into a variety of different physical locations – which have a variety of security models.

Operators of these WANs must be able to protect the security of their network devices, ensuring that only authorised network equipment is introduced into the network, only authorised software is executed by the devices within their networks, and only authorised systems can read and write to those devices. Achieving these requirements today requires patching together definitions from multiple SDOs, and often requires operators to handle differences between network equipment vendor implementations of the same standard.

This document provides a specification that allows for:

- Secure turn-up of a network device – going from a factory-fresh device to one that is attested to be genuine, running the correct software, and provisioned with the relevant security artifacts.
- Ongoing secure management of a network device – providing means to allow client and server mutual authentication, and comply with the principle of least privilege.

The goal is to leverage the Open Compute Project to provide a common set of requirements for network equipment vendors to comply with – and network operators to implement against to reduce the overhead and improve the industry-wide security of WAN routers.

The specifications defined herein are designed to ensure reliable operation at scale – with a focus on automated interaction with network devices. With today's WAN scale, such system-driven interaction with network devices is critical to meet the ongoing growth demands.

2. License

THE UPDATED DEFAULT CONTRIBUTOR LICENSE AGREEMENT (CLA) IS [OWFa 0.9](#). PLEASE VERIFY THE CORRECT CLA/FSA IS USED AND EXECUTED FOR THIS CONTRIBUTION.

2.1. Open Web Foundation (OWF) CLA

Contributions to this Specification are made under the terms and conditions set forth in **Modified Open Web Foundation Agreement 0.9 (OWFa 0.9)**. (As of October 16, 2024) (“Contribution License”) by:

Google
Microsoft
Meta

Usage of this Specification is governed by the terms and conditions set forth in **Modified OWFa 0.9 Final Specification Agreement (FSA)** (As of October 16, 2024) (“**Specification License**”).

You can review the applicable Specification License(s) referenced above by the contributors to this Specification on the OCP website at <https://www.opencompute.org/contributions/templates-agreements>.

For actual executed copies of either agreement, please contact OCP directly.

Notes:

The above license does not apply to the Appendix or Appendices. The information in the Appendix or Appendices is for reference only and non-normative in nature.

NOTWITHSTANDING THE FOREGOING LICENSES, THIS SPECIFICATION IS PROVIDED BY OCP "AS IS" AND OCP EXPRESSLY DISCLAIMS ANY WARRANTIES (EXPRESS, IMPLIED, OR OTHERWISE), INCLUDING IMPLIED WARRANTIES OF MERCHANTABILITY, NON-INFRINGEMENT, FITNESS FOR A PARTICULAR PURPOSE, OR TITLE, RELATED TO THE SPECIFICATION. NOTICE IS HEREBY GIVEN, THAT OTHER RIGHTS NOT GRANTED AS SET FORTH ABOVE, INCLUDING WITHOUT LIMITATION, RIGHTS OF THIRD PARTIES WHO DID NOT EXECUTE THE ABOVE LICENSES, MAY BE IMPLICATED BY THE IMPLEMENTATION OF OR COMPLIANCE WITH THIS SPECIFICATION. OCP IS NOT RESPONSIBLE FOR IDENTIFYING RIGHTS FOR WHICH A LICENSE MAY BE REQUIRED IN ORDER TO IMPLEMENT THIS SPECIFICATION. THE ENTIRE RISK AS TO IMPLEMENTING OR OTHERWISE USING THE SPECIFICATION IS ASSUMED BY YOU. IN NO EVENT WILL OCP BE LIABLE TO YOU FOR ANY MONETARY DAMAGES WITH RESPECT TO ANY CLAIMS RELATED TO, OR ARISING OUT OF YOUR USE OF THIS SPECIFICATION, INCLUDING BUT NOT LIMITED TO ANY LIABILITY FOR LOST PROFITS OR ANY CONSEQUENTIAL, INCIDENTAL, INDIRECT, SPECIAL OR PUNITIVE DAMAGES OF ANY

CHARACTER FROM ANY CAUSES OF ACTION OF ANY KIND WITH RESPECT TO THIS SPECIFICATION, WHETHER BASED ON BREACH OF CONTRACT, TORT (INCLUDING NEGLIGENCE), OR OTHERWISE, AND EVEN IF OCP HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

3. Introduction

This specification standardises a set of technologies for secure management of Wide Area Network (WAN) routers. At the surface level, the objective of secure management is to enable:

- Initial turn-up of network equipment – the process of taking a factory-fresh network device and inserting it securely into WAN infrastructure.
- Run-time authentication and authorisation of clients interacting with services that are exposed by network devices.

Achieving these objectives requires some consideration of underlying nuances, particularly, hardware specifications for cryptoprocessors such as Trusted Platform Modules (TPMs) that are available on these devices, and client/server certificate requirements. Today's ecosystem is somewhat fragmented, with different hardware being provided by network equipment vendors with no standardised mechanisms available to interact with the processes described above.

Thus, the goal of this specification is not to define new technologies or interfaces that are to be used to fulfill these functions, but rather to clarify a set of standards that are required to be adhered to for deployment by network operators. The primary authors of this specification are operators of large-scale, global WAN backbone networks with significant experience in their automated operation.

The specification does not intend to provide “backwards compatibility” to cover all devices that are deployed within WAN infrastructure, but rather to provide a minimum baseline that should be met by network equipment manufacturers. In some cases, standards that are referenced within this document provide means to have backward-compatibility, or can be deployed in a piecemeal fashion. Such usages are outside the scope of this specification.

3.1. Definitions

Root of Trust (RoT): A root of trust is a hardware device installed in a network device which is relied upon to provide credentials to assert the correctness of other components of the system. In this specification a RoT must always be a hardware device. (See [this section](#) for further discussions.)

Enrollment: Device enrollment is a process through which a factory-state device (which may be new, or have replaced control card hardware) is verified to be a genuine device, and its initial key material (initial attestation key pair, initial device ID key pair) are replaced with credentials issued by the CA of the network operator.

Attestation: Device attestation is an ongoing process that is used to assert that the state of the network device is that which is expected by the operator. The attestation state of the device is affected by changes to the software running on the system - including, but not limited to, the boot loader, the secure boot policy, and the operating system image to be loaded by the device. Attestation uses Platform Configuration Registers (PCRs) to store the particular state of these components.

3.2. Threat Analysis

The following sections provide brief overviews of the threat models that this specification seeks to provide mitigations for.

3.2.1. Counterfeit Devices

A potential threat model for a wide-area network is that a device that is within a remote location is counterfeit – that is to say, it is not a device that is from the intended network equipment vendor. In such cases, the hardware design of the device itself may no longer meet the requirements of the network that it is installed in. This vulnerability may allow for exfiltration of both network traffic and/or sensitive material such as device keys that can be used to mount an attack on other areas of the network system.

This specification provides means to ensure that such devices can be detected, and are not integrated into the wider WAN system.

3.2.2. Counterfeit Software

A network device may execute code that is not from the intended source – for example, an operating system image that is not built by the original network equipment vendor. Such software execution may allow for network traffic, or other sensitive material to be exfiltrated from the network device. This vulnerability may allow for attacks to be mounted on the wider network system.

This specification provides means to ensure that such software images are detected, and devices running them are not integrated into the wider WAN system.

3.2.3. Unauthorised API access from external systems

A system able to interact with a network device may not be authorised to do so. For example, through obtaining authentication and authorisation keys that allows the system to make changes to the configuration of the network device.

This specification provides means to ensure that mutual authentication between client and server is performed for network operations, and that the credentials that are used for doing so can be managed (provisioned, revoked) securely.

3.2.4. Expansion of privileges from external systems

A system able to interact with a network device may overstep its intended scope – for example, a system that is intended to manage routing functionality may change security credentials. In such cases, external systems that are authorised to communicate with a WAN router, may be used to make wider changes – allowing for further unauthorised access, or other attacks to be mounted on the WAN system.

This specification provides means to define the allowed scopes of particular systems, and authorize them against these scopes – both at an API call level, as well as more granular levels within data that is owned by a WAN device.

4. Processes

4.1. Secure Device Turn-Up

Secure device turn-up encompasses the following components:

- Device: a network device provided by a network equipment manufacturer.
- DHCP server: a network-operator run server able to respond to DHCP requests issued by the network device.
- Bootstrap Server: a network-operator run server that implements the OpenConfig bootz specification.
- Enrollment/Attestation server: a network-operator run server that implements the OpenConfig enrollz and attestz services.
- Ownership Voucher server: A network equipment manufacturer run server that implements the OpenConfig OVGS service.

The secure device turn-up process is described in detail by the specifications of each of the following services.

- OpenConfig [bootz](#)
- OpenConfig [attestz and enrollz](#)
- OpenConfig [OVGS](#)

The following overview is provided to give an overview of the process used to turn up devices using the above services.

Prior to device install:

1. Prior to device turnup, a network operator obtains the *Ownership Voucher* for a particular network device from the third-party vendor OVGS server. Through calling the `GetOwnershipVoucher` RPC exposed by the OVGS server, the device owner is able to obtain the device's ownership voucher along with the TPM public key. This is an artifact – as defined in [RFC8366](#) – that provides core details of the network devices' security artifacts.
2. The equipment manufacturer must provision certificates in the base device configuration that relies on the IDevID key pair and IDevID certificate (signed by the equipment manufacturer's CA) for any SSL services.

During device install:

3. Upon device boot, the device issues a DHCP request to its attached LAN segment(s). The DHCP request includes the MAC address of the active control card. The network operator DHCP server responds to allocate an IP address, and gateway to the device – and includes a DHCP response option (`OPTION_V4_SZTP_REDIRECT` or `OPTION_V6_SZTP_REDIRECT`) containing a `bootz://` URL.
4. Following receiving such a response, the device initiates a gRPC connection to the Bootstrap Server whose address was specified. For this connection, the device uses the `IDevID` certificate of the active control card. The Bootstrap Server provides a response which contains a signed (using the device's ownership certificate) set of bootstrap data. This response allows for the device to boot (as it includes a required software image, bootloader password, configuration, and initial set of security profiles) and ensures that the device is the expected device (since data is signed using the certificate provided by the device manufacturer).
5. Following this initial bootstrapping, the device moves into an enrollment and attestation phase for the TPM module. The intention of this phase is to move from the Initial Attestation Key (IAK) and Initial Device ID (IDevID) certificates that are installed by the original equipment manufacturer, to “owner” versions of the same certificates – signed by the network operator's CA. These new certificates are referred to as Owner Initial Attestation Key (oIAK) and Owner Initial Device ID (oIDevID) respectively. This process is initiated by the network operator's systems initiating calls to the device (via `GetIakCert`) in the OpenConfig `enrollz` service – which verifies the IAK and IDevID certificates received for each control card against the expected set (received in the ownership voucher).

Post initial install, and on an ongoing basis:

6. The OpenConfig `attestz` service is used to attest the state of the network device. Generally, this allows the operator to determine that the owner IAK (oIAK) is that which is

expected (and has not been revoked, for example) along with determining whether the reported PCR values match those that are expected.

7. Following the initial attestation, the network device can be provisioned with operator provided mTLS certificates, since the device has been verified to be genuine, running the expected software, and is in the expected booted state.

4.2. Certificate Installation and Rotation

Devices exposing services via gRPC use Transport Layer Security (TLS) to encrypt communications between themselves and clients. All gRPC services exposed by a network device must use TLS, and switches must support TLS 1.3.

In order to provide authentication and encryption of connections, devices are provisioned with TLS certificates that include a SPIFFE ID. This identifier allows clients and servers to identify one another's identity. Structured forms of the SPIFFE ID allow for security policies to be applied based on details such as the region of a particular client (if encoded into the SPIFFE ID).

Certificate installation onto a network device is expected to be via the OpenConfig `gNSI.Certz` service, which allows for generation of new key pairs, and the installation of corresponding certificates. Devices support at least one (default) profile – with the standard name `system_default_profile` which utilises the manufacturer-installed IDevID certificate of the device. Devices must adding additional policies and their installation and rotation through the certz service.

Along with the base certificate – a device must support installation and rotation of the TLS trust bundle, and certificate revocation lists.

4.3. Service Authentication, and Authorization

For gRPC services that are running on a network device, authentication is achieved via mutual TLS authentication – that is to say, both client and server use the provided certificates and root certificate sets that they are provisioned with (via `certz`) to authenticate that the certificate is valid, issued by an approved CA, and is not revoked. All gRPC APIs that are exposed by a network device must use mTLS authentication.

Granular authorization is achieved by means of an authorization policy provisioned to a network device. Authorization policies use the principal information encoded as a SPIFFE ID to identify clients. An authorization policy determines:

- The target service that is running on the network device.
- The RPC(s) that each principal is authorised to access – encoded as either permit or deny rules.

Authorization policies should be applied before processing any part of a gRPC request received from a client.

Network devices must support installing and rotating policies via the OpenConfig [gNSI.Authz](#) service.

4.4. Granular (Path-based) Authorization

Where data is exposed via a particular gRPC API, there are often requirements to constrain access to a particular client to a particular subset. For example, a network device may expose configuration for both routing protocols and local user accounts – a network operator may wish for only the routing protocol information to be configurable via a config management system. Such constraining of accessible information can be done within client systems (limiting their requests), but must also be enforced at the target network device in order to mitigate potential management system compromises or bad actors.

Where there is unstructured data on the system (e.g., CLI text documents) the ability to constrain such access is limited. Protocols that use a structured data format – a tree – such as gNMI provide mechanisms by which a particular can be constrained according to granular access rules. Such a mechanism works in conjunction with the authorization policies described above – which operate at the per-RPC level. For example, an [authz](#) policy may limit a particular principal to calling [gNMI.Subscribe](#) – whereas an additional policy may limit the principal to calling [gNMI.Subscribe](#) only for [/interfaces/interface/state/oper-status](#).

Granular, path-based policies are provisioned and rotated via [gNSI.Pathz](#). Network devices must support [Pathz](#) and evaluate [pathz](#)-provided policies as part of each gNMI request.

.

4.5. Accounting

In order to allow for an audit trail for any operations that are performed against a network device, accounting records must be produced and exported by a network device. Accounting records contain information about the caller of a particular operation in terms of their network addressing, local authentication, and the operation performed.

The `gNSI.Acctz` service must be supported by a device to allow these accounting records to be streamed to a remote collector.

5. Hardware Requirements

5.1. Hardware RoT

[H1] All devices that comply with this specification must be provisioned with TCG-certified Trusted Platform Module 2.0 hardware RoT, or an lowRisc OpenTitan-certified hardware RoT.

- The [TPM 2.0 specification](#) is defined by the Trusted Computing Group.
 - [TCG-certified devices](#) are listed within their resources.
- OpenTitan is an open definition of a hardware RoT that is defined by lowRisc.
 - Additional details of OpenTitan are defined on [opentitan.org](#).

TPM2.0 or OpenTitan hardware RoTs provide the basis for a number of the operations described in this document.

6. Software Requirements

6.1. Summary of Required API Support

[S1] Devices must support the OpenConfig `bootz` service, triggered via DHCP.

[S2] Network equipment vendors must support the OpenConfig `ovgs` service.

[S3] Network devices must support the OpenConfig `enrollz` service.

[S4] Network devices must support the OpenConfig `attestz` service.

[S5] Network devices must support the OpenConfig `gNSI.Certz` service, and expose a `system_default_profile` based on the device's IDevID key pair.

[S6] Network devices must support the OpenConfig `gNSI.Authz` service, and evaluate `authz`-provided policies prior to each gRPC request.

[S7] Network devices must support the OpenConfig `gNSI.Pathz` service, and evaluate `pathz`-provided policies on each gNMI RPC call.

7. Compliance

Compliance tests for the above APIs are published as part of the OpenConfig `featureprofiles` project. This project provides open source tests that measure the compliance of a particular implementation against these APIs.

Standalone feature tests for the above APIs are defined within:

- [feature/system/bootz](#)
- [feature/system/acctz](#)
- [feature/system/authz](#)
- [feature/system/certz](#)

These tests provide an open source, repeatable means to determine compliance of a particular operation with a number of the APIs that are required in this document.

8. Acknowledgements

The Contributors of this Specification would like to acknowledge the following for their input and feedback:

BT: Alex Bogdanov

Google: Rob Shakir, Marcus Hines, Anees Shaikh, Andrés Lagar-Cavila

Microsoft: Paul-Andre Bissonnette

9. Compliance with OCP Tenets

9.1. Openness

This document serves as a base specification for security capabilities and APIs required for turn-up of WAN routers in an automated, scalable and most importantly secure manner. The work reflects both open discussion amongst three large infrastructure operators (Meta, Microsoft, Google) as well as collaboration with the wider industry, including network equipment manufacturers. Base specifications for underlying components build on, clarify, and revise (where necessary) other SDO output – such as the TPM specification from TCG, and standards such as Secure ZTP defined in the IETF – as well as that of open source projects, particularly OpenConfig.

The intention of this standard is to provide an open specification that clarifies how these disparate standards come together into a deployable profile which can be directly used by network operators.

The goal of the workstream is – through further open discussion – to reflect the requirements of operators of modern WANs. It is intended that this document is revised to reflect those requirements, and build further industry standardisation.

9.2. Efficiency

Disparate solutions for managing secure turn up of network devices exist in the industry today – some of which are proprietary, incomplete, or have multiple different potential implementations. Where there are standardised processes defined these often require vendor-specific scripting or procedures. This specification optimises those existing solutions by:

- providing a standard, single, set of implementation criteria that can be built upon by providers of network equipment.
- removing the need for network operators to have per-vendor implementations of secure turnup workflows, and maintain version-specific scripts for turn up.
- automating interactions between network equipment vendors and their customers (network operators) for artifacts related to device security.

9.3. Impact

This specification is the first (to our knowledge) end-to-end common definition of required security characteristics, hardware features, and APIs that ensure that a device can be:

- attested to be a genuine device provided by an equipment manufacturer,
- attested to be running genuine software provided by the equipment manufacturer,
- subsequently managed using secure transport protocols that rely on industry-standard mutual client/server authentication and authorization.

Security is a critical characteristic of operating modern WANs, providing standard APIs across vendors to achieve these approaches vastly improves the base security of all WANs that utilise devices that comply with this standard.

9.4. Scale

A fundamental characteristic of the technologies described in this specification is their ability to be utilised at both low- and high-scale. Automated procedures defined within this document provide a means whereby network equipment vendors can offer a common set of capabilities that work across their customer base. Network operators can equally leverage common capabilities across their entire network fleet – with common procedures even where a vendor makes up a small population of that deployment.

The scaling of this standard is enabled by open specifications, and reference implementation code.

9.5. Sustainability

This standard promotes use of existing technologies which themselves consider their sustainability characteristics.

10. References

- [1]: OpenConfig bootz specification, github.com/openconfig/bootz
 - [2]: OpenConfig enrollment and attestation specification, github.com/openconfig/attestz
 - [3]: gRPC Network Security Interface (gNSI), github.com/openconfig/gnsi
 - [4]: Ownership Voucher gRPC Service (OVGS), github.com/openconfig/ovgs
 - [5]: Feature Profiles, github.com/openconfig/featureprofiles
 - [6]: TPM 2.0 Library specification, Trusted Computing Group, trustedcomputinggroup.org/resource/tpm-library-specification
 - [7]: OpenTitan, opentitan.org
-