Routing Area Working Group
Internet-Draft
Updates: draft-conduit-tunnel-fabric (if approved)
Intended status: Standards Track
Expires: 23 August 2026

J. E. W. V
Department of the Air Force

19 February 2026

Underlay for IPsec Transport
draft-willman-rtgwg-conduit-tunnels-01

Abstract

This document specifies CONDUIT, a tunnel fabric management protocol for tactical and enterprise networks operating over heterogeneous Wide Area Network (WAN) links. CONDUIT automates the lifecycle management of IPsec tunnels, monitors tunnel health through active probing, and publishes quality metrics to enable Segment Routing over IPv6 (SRv6) based traffic engineering.

CONDUIT follows a strict separation of concerns: it manages the underlay tunnel fabric while delegating all traffic engineering decisions to SRv6. This architecture simplifies network operations, enables rapid adaptation to changing WAN conditions, and leverages standard SRv6 capabilities including Flexible Algorithm (Flex-Algo) and Topology-Independent Loop-Free Alternate (TI-LFA).

The protocol is fully programmable via gRPC and mandates compliance with Commercial National Security Algorithm Suite 2.0 (CNSA 2.0) cryptographic requirements.

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 https://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 23 August 2026.

Table of Contents

1. Introduction 3
1.1. Problem Statement 3
1.2. Design Philosophy 4
1.3. Design Goals 5
1.4. Scope 5
2. Conventions and Terminology 6
2.1. Requirements Language 6
2.2. Definitions 6
2.3. Abbreviations 7
3. Architecture 8
3.1. System Overview 8
3.2. Tunnel Fabric Concept 8
3.3. Separation of Concerns 9
4. Cryptographic Requirements 10
4.1. CNSA 2.0 Compliance 10
4.2. Mandatory Algorithms 10
4.3. Prohibited Algorithms 11
4.4. IPsec Profile 12
4.5. Certificate Requirements 12
4.6. TLS Requirements 13
5. Tunnel Fabric Management 13
5.1. Tunnel Lifecycle States 13
5.2. Tunnel Establishment Policy 14
5.3. Tunnel Naming Convention 15
5.4. Full Mesh Fabric 15
6. Health Monitoring 15
6.1. Probe Mechanism 16
6.2. Probe Packet Format 16
6.3. Metrics Calculation 18
6.3.1. Round-Trip Time 18
6.3.2. Jitter 19
6.3.3. Packet Loss 19

6.4. Probe Scheduling 19
6.5. Tunnel State Determination 20
7. Metric Publishing 20
7.1. Publishing Methods 20
7.2. Interface Metric Calculation 21
7.3. IGP TE Extensions 21
7.4. Flex-Algo Integration 22
7.5. SRLG Assignment 23
8. Control Protocol 23
8.1. Transport 23
8.2. Message Header 24
8.3. Message Types 25
8.4. HELLO Message 25
8.5. WAN Descriptor 26
8.6. Message Authentication 28
8.7. Anti-Replay Protection 29
9. SRv6 Integration 29
9.1. Tunnel Interfaces as SRv6 Adjacencies 29
9.2. Flex-Algo Tunnel Assignment 30
9.3. Traffic Steering 31
10. High Availability 31
10.1. HA Architecture 31
10.2. Synchronized State 32
10.3. Failover Behavior 32
11. gRPC API 32
11.1. Service Overview 33
11.2. Authentication and Authorization 33
12. Security Considerations 33
12.1. Cryptographic Security 33
12.2. Threat Mitigations 34
12.3. Operational Security 34
13. IANA Considerations 35
13.1. UDP Port Allocation 35
13.2. WAN Type Registry 35
14. References 35
14.1. Normative References 35
14.2. Informative References 36
Appendix A. WAN Type Characteristics 37
Acknowledgements 37
Author’s Address 37

1. Introduction

1.1. Problem Statement

Modern tactical and enterprise networks increasingly operate over diverse Wide Area Network (WAN) links with varying characteristics:

* Satellite Communications (SATCOM): Both Geostationary (GEO) with approximately 600ms round-trip latency and Low Earth Orbit (LEO) with 20-40ms latency, limited bandwidth, and weather sensitivity.

* Line-of-Sight (LOS) Radio: Low latency, high mobility, but terrain-dependent coverage.

* Cellular Networks (LTE/5G): Variable quality, broad coverage, but may traverse untrusted infrastructure.

* High Frequency (HF) Radio: Very low bandwidth but extended range with atmospheric effects.

* Wired Connections: When available at fixed installations.

Managing IPsec tunnels across these heterogeneous links presents significant operational challenges:

1. Links appear and disappear as network elements move or environmental conditions change.

2. Link quality varies dramatically and unpredictably.

3. Multiple paths to each destination may exist simultaneously.

4. Failover must complete within sub-second timeframes to maintain voice and command-and-control (C2) applications.

5. All traffic must be encrypted using approved cryptographic algorithms.

Existing approaches typically couple tunnel management with traffic engineering decisions, creating unnecessary complexity and limiting interoperability with standard routing protocols.

1.2. Design Philosophy

CONDUIT adheres to a strict separation of concerns:

* SRv6 Traffic Engineering Layer: Responsible for all path selection decisions, including Flexible Algorithm computation, fast reroute via TI-LFA, traffic class steering, and load balancing across equal-cost paths.

* CONDUIT Tunnel Fabric Layer: Responsible for tunnel lifecycle management (creation, maintenance, deletion), health monitoring through active probing, metric publication to enable SRv6-based decisions, and IPsec security association management.

This separation ensures that CONDUIT does not duplicate functionality available in standard SRv6 implementations while providing essential tunnel management capabilities that SRv6 requires but does not natively provide.

1.3. Design Goals

The following quantitative goals guide CONDUIT's design:

Parameter	Requirement
Tunnel Scale	Support for 1000+ tunnels per node
Failover Detection	Less than 500ms to detect tunnel failure
Metric Publish Rate	1 Hz default, 10 Hz burst capability
Cryptographic Suite	Full CNSA 2.0 compliance
API Coverage	100% of functionality accessible via gRPC
Probe Overhead	Less than 1% of available bandwidth

Table 1

1.4. Scope

This document specifies:

* Tunnel lifecycle management procedures
* Health monitoring through active probing
* Metric calculation and publishing mechanisms
* Control protocol message formats
* Integration requirements for SRv6-based traffic engineering
* Cryptographic requirements for CNSA 2.0 compliance

This document does not specify:

* SRv6 data plane operations (covered by [RFC8986])

* Flexible Algorithm computation (covered by draft-ietf-lsr-flex-algo)

* IPsec key exchange procedures (covered by [RFC7296])

* IGP extensions for traffic engineering (covered by [RFC8570])

2. Conventions and Terminology

2.1. Requirements Language

The key words "MUST", "MUST NOT", "REQUIRED", "SHALL", "SHALL NOT", "SHOULD", "SHOULD NOT", "RECOMMENDED", "NOT RECOMMENDED", "MAY", and "OPTIONAL" in this document are to be interpreted as described in BCP 14 [RFC2119] [RFC8174] when, and only when, they appear in all capitals, as shown here.

2.2. Definitions

Node: A device implementing the CONDUIT protocol.

Peer: A remote node with which tunnels are established.

WAN: A Wide Area Network interface, either physical or virtual, capable of carrying IP traffic to remote peers.

Tunnel: An IPsec-protected communication path between two WAN interfaces on different nodes.

Fabric: The complete set of tunnels managed by a CONDUIT node.

Probe: A measurement packet transmitted through a tunnel to assess its quality characteristics.

Metric: A quantitative measure of tunnel quality, such as latency, jitter, or packet loss.

Locator: The SRv6 locator block assigned to a node, typically a /48 IPv6 prefix.

2.3. Abbreviations

CNSA: Commercial National Security Algorithm Suite

ECMP: Equal-Cost Multi-Path

ESP: Encapsulating Security Payload

Flex-Algo: Flexible Algorithm

GEO: Geostationary Earth Orbit

gRPC: gRPC Remote Procedure Call

HMAC: Hash-based Message Authentication Code

IGP: Interior Gateway Protocol

IKE: Internet Key Exchange

LEO: Low Earth Orbit

LOS: Line-of-Sight

mTLS: Mutual Transport Layer Security

RTT: Round-Trip Time

SA: Security Association

SPI: Security Parameter Index

SRLG: Shared Risk Link Group

SRv6: Segment Routing over IPv6

TE: Traffic Engineering

TI-LFA: Topology-Independent Loop-Free Alternate

3. Architecture

3.1. System Overview

A CONDUIT deployment consists of the following components:

* SDN Controller (Optional): Provides centralized policy management, SRv6 policy configuration, and network-wide optimization. Communicates with CONDUIT nodes via gRPC.

* SRv6 Data Plane: Executes traffic engineering decisions using standard SRv6 mechanisms including IS-IS or OSPFv3 with SRv6 extensions, Flexible Algorithm for constraint-based path computation, and TI-LFA for sub-50ms fast reroute.

* CONDUIT Daemon: Manages the tunnel fabric, comprising the Tunnel Lifecycle Manager for creating and deleting tunnels, Health Monitor for probing tunnel quality, Metric Publisher for feeding metrics to SRv6/IGP, and the gRPC API Server for external control.

* IKEv2/IPsec Subsystem: Handles cryptographic operations and security association management. CONDUIT interfaces with this subsystem but does not implement cryptographic operations directly.

* WAN Interfaces: Physical or virtual interfaces connecting to various transport networks.

3.2. Tunnel Fabric Concept

CONDUIT creates tunnels between WAN interfaces according to the configured tunnel policy. In a full mesh configuration with N local WANs and M remote WANs to a given peer, CONDUIT establishes up to N x M tunnels.

For example, consider a node with three WAN interfaces (SATCOM, LOS Radio, and LTE) connecting to a peer also having three WAN interfaces. A full mesh fabric would comprise nine tunnels:

* SATCOM to SATCOM

* SATCOM to LOS
* SATCOM to LTE
* LOS to SATCOM
* LOS to LOS
* LOS to LTE
* LTE to SATCOM
* LTE to LOS
* LTE to LTE

Each tunnel becomes a distinct interface visible to the SRv6 data plane and IGP. The IGP sees multiple adjacencies to the same peer, each with potentially different metrics based on tunnel quality.

3.3. Separation of Concerns

CONDUIT explicitly delegates the following functions to SRv6:

Function	Mechanism
Path Selection	SRv6 Flex-Algo, IGP SPF computation
Traffic Classification	SRv6 Policy color matching
Load Balancing	IGP ECMP across equal-metric tunnels
Fast Reroute	TI-LFA pre-computed backup paths
QoS Marking	SRv6 Traffic Class field

Table 2

CONDUIT is responsible for:

Function	Mechanism
Tunnel Creation	Automatic based on WAN availability
Tunnel Deletion	Automatic when WAN becomes unavailable
Health Monitoring	Active probing with configurable rates
Metric Calculation	RTT, jitter, loss measurement
Metric Publishing	Interface metrics, IGP TE extensions
IPsec SA Management	Via IKEv2 with CNSA 2.0 parameters

Table 3

4. Cryptographic Requirements

4.1. CNSA 2.0 Compliance

CONDUIT implementations MUST comply with the Commercial National Security Algorithm Suite 2.0 (CNSA 2.0) as specified by the National Security Agency for protection of classified information.

Implementations intended for use in environments not requiring CNSA 2.0 compliance MAY support additional algorithms but MUST default to CNSA 2.0 algorithms and MUST provide configuration options to enforce CNSA 2.0-only operation.

4.2. Mandatory Algorithms

The following algorithms are REQUIRED:

Function	Algorithm	Parameters
Symmetric Encryption	AES-256-GCM	256-bit key, 96-bit IV, 128-bit authentication tag
Key Exchange	ECDH	P-384 curve (secp384r1)
Digital Signature	ECDSA	P-384 curve (secp384r1)
Hash Function	SHA-384	384-bit output
Message Authentication	HMAC-SHA-384 [RFC4868]	384-bit output
Pseudo-Random Function (IKEv2)	HMAC-SHA-384	Per [RFC4868]
Key Derivation	HKDF-SHA-384	Per [RFC5869]

Table 4

4.3. Prohibited Algorithms

CONDUIT implementations MUST reject configuration of the following algorithms and MUST NOT negotiate their use:

* AES with key sizes less than 256 bits (AES-128, AES-192)

* Triple DES (3DES) and DES

* SHA-1 and SHA-256 for authentication or integrity

* MD5

* RSA with key sizes less than 3072 bits

* Diffie-Hellman groups with less than 3072-bit modulus

* ECDH or ECDSA with curves smaller than P-384

* ChaCha20-Poly1305 (not approved for CNSA)

Implementations MUST generate a configuration error and refuse to operate if prohibited algorithms are specified.

4.4. IPsec Profile

CONDUIT implementations MUST configure IPsec with the following parameters:

IKEv2 Parameters:

* Version: IKEv2 only (version 1 MUST NOT be used)
* Encryption: AES-256-GCM with 16-octet ICV (aes256gcm16)
* PRF: HMAC-SHA-384 (prf-hmac-sha384)
* Diffie-Hellman Group: 20 (384-bit random ECP, P-384)
* Authentication: ECDSA with SHA-384 using P-384 certificates

ESP (Child SA) Parameters:

* Encryption: AES-256-GCM with 16-octet ICV
* Extended Sequence Numbers: Enabled
* Traffic Selectors: As required for overlay traffic

Lifetime Parameters:

* IKE SA Lifetime: 86400 seconds (24 hours) maximum
* Child SA Lifetime: 28800 seconds (8 hours) maximum
* Child SA Volume Limit: 1 TiB maximum

4.5. Certificate Requirements

Node certificates MUST meet the following requirements:

* Format: X.509 version 3
* Key Algorithm: ECDSA with P-384 curve
* Signature Algorithm: ecdsa-with-SHA384
* Key Usage: digitalSignature, keyAgreement
* Extended Key Usage: serverAuth, clientAuth, IPsecIKE

Certificate Authority certificates MUST use ECDSA with P-384 and SHA-384 signatures.

Certificate validation MUST include revocation checking via CRL or OCSP when network connectivity permits.

4.6. TLS Requirements

The gRPC API MUST be protected using TLS with the following configuration:

* Minimum Version: TLS 1.3 (TLS 1.2 and earlier MUST NOT be used)
* Cipher Suite: TLS_AES_256_GCM_SHA384 only
* Certificate Type: ECDSA with P-384 curve
* Client Authentication: Required (mutual TLS)

Implementations MUST reject TLS connections that do not meet these requirements.

5. Tunnel Fabric Management

5.1. Tunnel Lifecycle States

Each tunnel exists in one of the following states:

DISCOVERED: A peer has been discovered but tunnel establishment has not begun. Entry: peer discovery. Exit: initiate IKEv2.

INITIATING: IKEv2 negotiation is in progress. Entry: IKEv2 initiation. Exit: IKE_SA and CHILD_SA established, or timeout/ failure.

ESTABLISHED: The tunnel is operational and probes are succeeding within acceptable thresholds. Entry: successful IKEv2 completion or recovery from DEGRADED. Exit: metrics degrade, probe failures accumulate, or rekey initiated.

DEGRADED: The tunnel is operational but metrics exceed preferred thresholds. Entry: metrics exceed degradation thresholds. Exit: metrics recover or probe failures accumulate.

REKEYING: Security association rekeying is in progress. Entry:

rekey initiated (time or volume based). Exit: rekey success or failure.

DOWN: The tunnel has failed and is not currently usable. Entry:

consecutive probe failures exceed threshold or IKE failure. Exit: probes succeed or WAN becomes unavailable.

DELETED: The tunnel is marked for removal. Entry: WAN unavailable

or administrative deletion. Exit: cleanup complete.

State transitions MUST be reported via the gRPC API event stream and SHOULD be logged for operational visibility.

5.2. Tunnel Establishment Policy

CONDUIT supports three fabric modes:

Full Mesh: Create tunnels for all combinations of local and remote

WAN interfaces. This mode maximizes path diversity but increases resource consumption.

Primary Only: Create tunnels only between designated primary WAN

interfaces on each node. This mode minimizes resource consumption but limits path diversity.

Custom: Apply a rule-based policy to determine which tunnel

combinations to establish. Rules may specify:

* Local WAN type (e.g., SATCOM, LOS, cellular)
* Remote WAN type
* Action (create or skip)
* Priority (affects establishment order)

The tunnel policy SHOULD be configurable via the gRPC API and SHOULD support runtime modification without service interruption.

5.3. Tunnel Naming Convention

Tunnel interfaces SHOULD be named according to the following convention to facilitate operational identification:

tun-<peer>-<local_wan>-<remote_wan>

Where:

* <peer> is a short identifier for the remote node
* <local_wan> is a short identifier for the local WAN interface
* <remote_wan> is a short identifier for the remote WAN interface

Example names:

* tun-hq-sat-sat (to headquarters, SATCOM to SATCOM)
* tun-hq-los-lte (to headquarters, LOS radio to LTE)
* tun-fwd1-hf-hf (to forward unit 1, HF to HF)

5.4. Full Mesh Fabric

When operating in full mesh mode, the number of tunnels T to a single peer is:

T = L x R

Where L is the number of local WAN interfaces and R is the number of remote WAN interfaces on the peer.

The total tunnels across all peers is:

Total = sum(L x R_i) for each peer i

Implementations MUST support configurable limits on:

* Maximum tunnels per peer
* Maximum total tunnels

When limits would be exceeded, implementations SHOULD prioritize tunnel creation based on configured policy and WAN type preferences.

6. Health Monitoring

6.1. Probe Mechanism

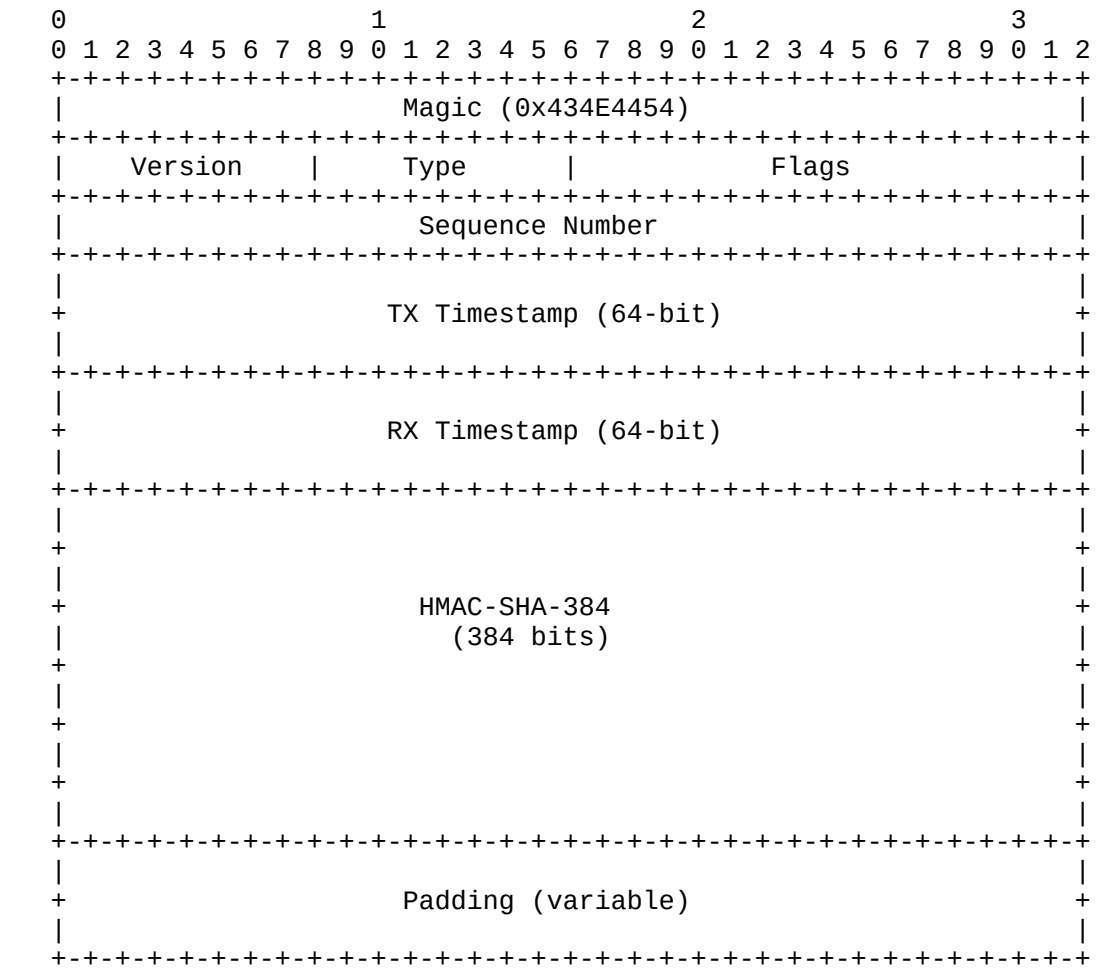
CONDUIT monitors tunnel health through active probing. Each tunnel is independently probed to measure:

Metric	Unit	Collection Method
Round-Trip Time	Microseconds	Timestamp echo
Jitter	Microseconds	[RFC3550] algorithm
Packet Loss	Percentage	Sequence tracking
Availability	Percentage	Probe success rate

Table 5

6.2. Probe Packet Format

CONDUIT probe packets use the following format:



Field descriptions:

Magic (32 bits): Fixed value 0x434E4454 ("CNDT" in ASCII). Used to identify CONDUIT probe packets.

Version (8 bits): Protocol version. This specification defines version 1.

Type (8 bits): Probe type:

* 0x01: Echo Request

* 0x02: Echo Reply
* 0x03: Timestamp Request
* 0x04: Timestamp Reply

Flags (16 bits): Reserved for future use. MUST be set to zero on transmission and ignored on receipt.

Sequence Number (32 bits): Monotonically increasing sequence number used for loss detection and RTT matching.

TX Timestamp (64 bits): Transmission timestamp in microseconds since the Unix epoch. Set by the sender.

RX Timestamp (64 bits): Reception timestamp in microseconds since the Unix epoch. Set by the responder in reply packets; zero in request packets.

HMAC-SHA-384 (384 bits): Message authentication code computed over all preceding fields. Provides probe authenticity and integrity.

Padding (variable): Optional padding to achieve desired probe size for bandwidth estimation or MTU testing.

6.3. Metrics Calculation

6.3.1. Round-Trip Time

RTT is calculated as:

```
RTT = T_local_rx - T_local_tx
```

Where T_local_rx is the local time when the reply was received and T_local_tx is the local time when the request was sent.

For more accurate one-way delay measurement when nodes have synchronized clocks:

```
OWD_forward = T_remote_rx - T_local_tx
OWD_reverse = T_local_rx - T_remote_tx
```

6.3.2. Jitter

Jitter is calculated using the algorithm specified in [RFC3550]:

```
D(i) = (R(i) - R(i-1)) - (S(i) - S(i-1))
J(i) = J(i-1) + (|D(i)| - J(i-1)) / 16
```

Where:

* R(i) is the receive timestamp of probe i

* S(i) is the send timestamp of probe i

* J(i) is the jitter estimate after probe i

6.3.3. Packet Loss

Packet loss is calculated over a sliding window:

```
Loss% = ((Probes_sent - Probes_received) / Probes_sent) x 100
```

The default window size is 100 probes. Implementations SHOULD make this configurable.

6.4. Probe Scheduling

The base probe interval is configurable, with a default of 100ms.

The effective probe interval SHOULD be adjusted based on tunnel state:

State	Multiplier	Rationale
Established	1.0	Normal monitoring
Degraded	0.5	Increased monitoring of troubled tunnel
Recovering	0.25	Rapid assessment during recovery
Down	2.0	Reduced load on failed path

Table 6

To prevent probe synchronization across tunnels, implementations SHOULD add random jitter of +/- 10% to each probe interval.

6.5. Tunnel State Determination

Tunnel state transitions are triggered by threshold crossings:

Degraded Thresholds:

* RTT increases by more than 50% above baseline
* Packet loss exceeds 1%
* Jitter increases by more than 100% above baseline

Down Thresholds:

* Five or more consecutive probe failures
* Packet loss exceeds 25%

Thresholds SHOULD be configurable via the gRPC API.

7. Metric Publishing

7.1. Publishing Methods

CONDUIT publishes tunnel metrics through multiple mechanisms to enable SRv6-based traffic engineering:

Method	Target	Use Case
Interface Metrics	Operating system	Simple deployments
IGP TE Extensions	IS-IS / OSPF	Distributed TE
gRPC Streaming	SDN Controller	Centralized TE
Flex-Algo Affinity	IGP Flex-Algo	Constraint-based TE

Table 7

Implementations MUST support interface metric publishing. Implementations SHOULD support IGP TE extensions and gRPC streaming.

7.2. Interface Metric Calculation

CONDUIT calculates an interface metric from tunnel quality measurements using the following formula:

Metric = Latency_component + Loss_component + Jitter_component

Where:

* Latency_component = RTT_ms (1 ms = 1 metric unit)

* Loss_component = Loss% x 100 (1% loss = 100 metric units)

* Jitter_component = Jitter_ms x 10 (1 ms jitter = 10 metric units)

The resulting metric MUST be clamped to the range 1-65535.

Example calculations:

Tunnel Type	RTT	Loss	Jitter	Metric
GEO SATCOM	600 ms	0.1%	10 ms	710
LOS Radio	5 ms	0%	0.5 ms	10
LTE Cellular	50 ms	0.5%	5 ms	150

Table 8

The SRv6 data plane, seeing these metrics, will naturally prefer the LOS tunnel (metric 10) over LTE (metric 150) over SATCOM (metric 710) when computing shortest paths.

7.3. IGP TE Extensions

When IGP TE metric publishing is enabled, CONDUIT advertises the following per-tunnel attributes:

* TE Metric: Calculated as described in Section 7.2

* Maximum Link Bandwidth: Configured WAN bandwidth

* Available Bandwidth: Estimated available capacity

* Unidirectional Link Delay: Measured one-way delay per [RFC8570]

* Unidirectional Delay Variation: Measured jitter per [RFC8570]

* Unidirectional Packet Loss: Measured loss per [RFC8570]

* Shared Risk Link Groups: Assigned SRLG values

* Administrative Group: For Flex-Algo affinity matching

For IS-IS, these attributes are advertised using the extensions defined in [RFC8570]. For OSPF, the equivalent extensions from [RFC7471] apply.

7.4. Flex-Algo Integration

CONDUIT assigns tunnels to Flexible Algorithm groups based on measured characteristics and configured constraints.

A Flex-Algo definition specifies:

* Algorithm ID: An integer in the range 128-255

* Metric Type: IGP metric, minimum delay, or TE metric

* Constraints: Maximum delay, minimum bandwidth, SRLG exclusions

CONDUIT evaluates each tunnel against each Flex-Algo definition and sets the appropriate affinity bits in IGP advertisements.

Example Flex-Algo definitions for tactical networks:

Flex-Algo 128 (Low Latency):

* Metric Type: Minimum Delay

* Constraint: Maximum delay 100 ms

* Use: Voice, real-time C2

Flex-Algo 129 (High Bandwidth):

* Metric Type: IGP

* Constraint: Minimum bandwidth 1 Mbps

* Use: Video, bulk transfer

Flex-Algo 130 (Resilient):

* Metric Type: IGP

* Constraint: Exclude SRLG "satcom"

* Use: Critical C2 requiring terrestrial diversity

With these definitions, the SRv6 data plane automatically steers voice traffic over low-latency tunnels (LOS radio), video over high-bandwidth tunnels (possibly ECMP across multiple), and critical C2 over tunnels not sharing SATCOM failure modes.

7.5. SRLG Assignment

Shared Risk Link Groups identify tunnels that share common failure modes. CONDUIT assigns SRLG values based on:

* WAN Type: All tunnels using SATCOM share a "satcom" SRLG

* Physical Path: Tunnels using the same physical link share an SRLG

* Provider: Tunnels using the same carrier share an SRLG

* Geographic: Tunnels transiting the same geographic area may share an SRLG

SRLG assignment enables SRv6 TI-LFA to compute backup paths that avoid shared failure modes and Flex-Algo to exclude entire failure domains.

8. Control Protocol

8.1. Transport

CONDUIT control messages are transported via UDP. Two ports are used:

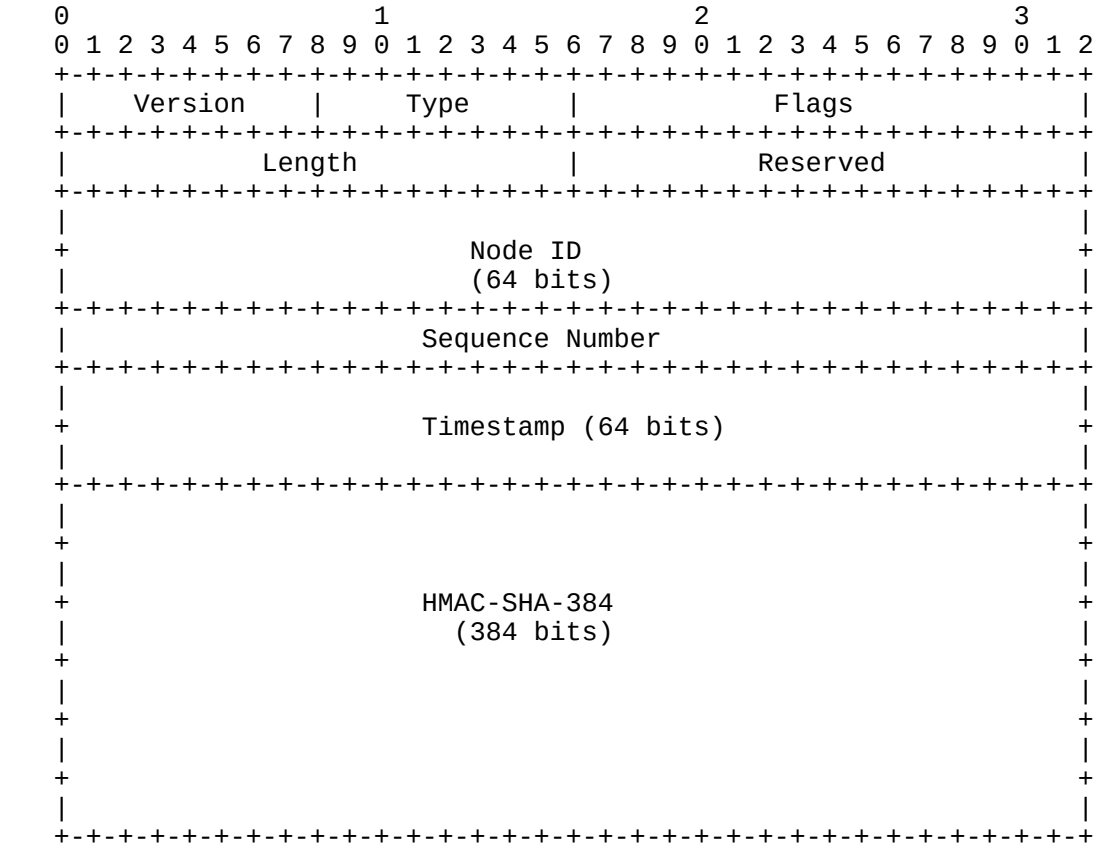
* Port 4794: Control messages (HELLO, WAN_UPDATE, etc.)

* Port 4795: Probe packets

Control messages are transmitted through established IPsec tunnels when available. For initial peer discovery before tunnel establishment, messages may be sent via the underlying WAN with HMAC authentication.

8.2. Message Header

All CONDUIT control messages share a common header:



Field descriptions:

Version (8 bits): Protocol version. This specification defines version 1.

Type (8 bits): Message type as defined in Section 8.3.

Flags (16 bits): Message-specific flags.

Length (16 bits): Total message length in octets, including header.

Reserved (16 bits): Reserved for future use. MUST be zero.

Node ID (64 bits): Unique identifier of the sending node.

Sequence Number (32 bits): Per-peer monotonically increasing counter for anti-replay protection.

Timestamp (64 bits): Message creation time in microseconds since Unix epoch.

HMAC-SHA-384 (384 bits): Message authentication code computed over the entire message with the HMAC field set to zero.

Total header size: 72 octets.

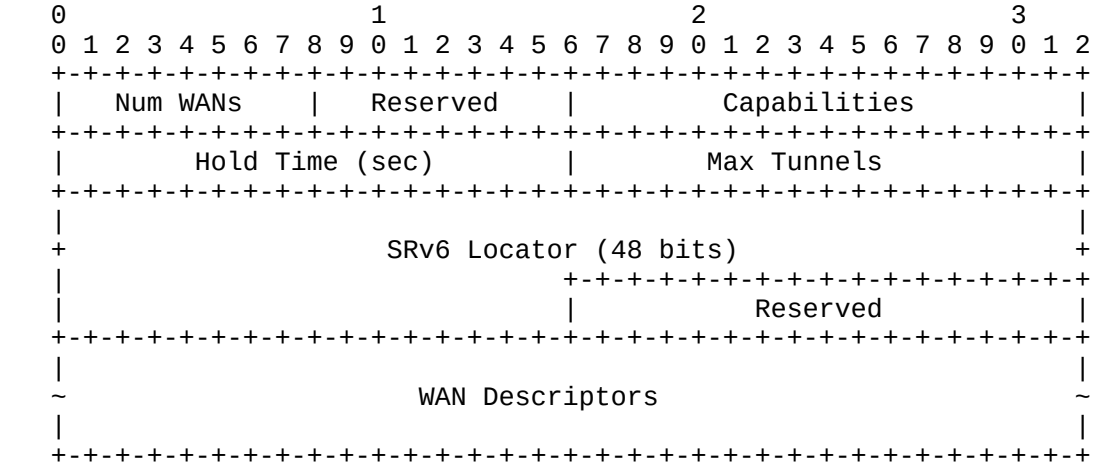
8.3. Message Types

Value	Name	Description
0x01	HELLO	Peer discovery and WAN advertisement
0x02	HELLO_ACK	Response to HELLO
0x03	WAN_UPDATE	WAN availability change notification
0x04	TUNNEL_STATE	Tunnel state change notification
0x05	KEEPALIVE	Peer liveness check
0x06	KEEPALIVE_ACK	Response to KEEPALIVE
0x0F	ERROR	Error notification

Table 9

8.4. HELLO Message

The HELLO message is used for peer discovery and WAN capability advertisement:



Field descriptions:

Num WANs (8 bits): Number of WAN Descriptors following.

Capabilities (16 bits): Bitmask of supported capabilities.

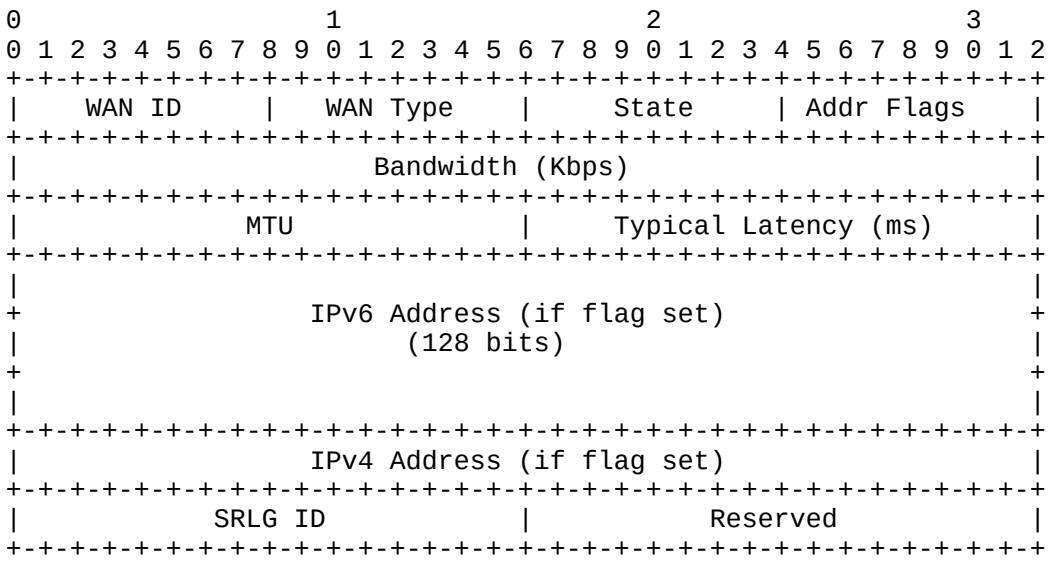
Hold Time (16 bits): Time in seconds the receiver should consider the sender reachable without further messages.

Max Tunnels (16 bits): Maximum tunnels this node can establish to a single peer.

SRv6 Locator (48 bits): The node’s SRv6 locator prefix.

8.5. WAN Descriptor

Each WAN interface is described by a WAN Descriptor:



WAN Type values:

Value	Name	Typical Characteristics
0x01	SATCOM_GEO	600ms RTT, 2-10 Mbps
0x02	SATCOM_LEO	30ms RTT, 50-200 Mbps
0x03	LOS_RADIO	5ms RTT, 10-100 Mbps
0x04	TROPOSCATTER	10ms RTT, 5-20 Mbps
0x05	HF_RADIO	50ms RTT, 9.6-64 Kbps
0x06	CELLULAR_LTE	30ms RTT, 10-100 Mbps
0x07	CELLULAR_5G	10ms RTT, 100-1000 Mbps
0x08	WIRE_ETHERNET	1ms RTT, 1-100 Gbps
0x09	WIRE_FIBER	1ms RTT, 1-100 Gbps
0x0A	WIFI	5ms RTT, 50-500 Mbps

Table 10

Address Flags:

* Bit 0: IPv4 address present

* Bit 1: IPv6 address present

* Bit 2: Behind NAT (IPv4)

* Bit 3: Behind NAT (IPv6)

8.6. Message Authentication

All CONDUIT control messages MUST be authenticated using HMAC- SHA-384.

The authentication key is derived from the IKEv2 SA when available:

```
AUTH_KEY = HKDF-SHA-384(
   IKM = SK_d,
   salt = "CONDUIT-v1-auth",
   info = Node_ID_local || Node_ID_remote,
   length = 48
)
```

For initial HELLO messages before IKEv2 SA establishment, a pre-shared key MAY be used. Unauthenticated HELLO messages MUST NOT be accepted in production deployments.

8.7. Anti-Replay Protection

Each peer maintains anti-replay state consisting of:

* Expected sequence number (next expected value)
* Window bitmap (received sequences within window)
* Window size (default 64)
* Maximum timestamp skew (default 60 seconds)

Message acceptance rules:

1. If sequence >= expected: Accept and update expected
2. If sequence < expected - window_size: Reject (too old)
3. If sequence in window and bit set: Reject (replay)
4. If sequence in window and bit clear: Accept and set bit
5. If timestamp differs from local time by more than maximum skew: Reject

9. SRv6 Integration

9.1. Tunnel Interfaces as SRv6 Adjacencies

Each CONDUIT tunnel is presented to the SRv6 data plane as a distinct interface. From the perspective of IS-IS or OSPFv3, each tunnel represents a separate adjacency to the peer node.

This design enables:

* Independent metrics per tunnel based on measured quality

* ECMP load balancing across multiple tunnels to the same peer

* Flex-Algo differentiation based on tunnel characteristics

* TI-LFA backup path computation considering tunnel diversity

For example, a node with three tunnels to a peer (via SATCOM, LOS, and LTE) would advertise three adjacencies in IS-IS, each with its own metric. The IGP SPF computation naturally prefers lower-metric paths.

9.2. Flex-Algo Tunnel Assignment

CONDUIT evaluates each tunnel against configured Flex-Algo definitions and assigns appropriate affinities.

A tunnel qualifies for a Flex-Algo if:

* Measured delay is less than the maximum delay constraint (if specified)

* Configured bandwidth meets the minimum bandwidth constraint (if specified)

* Tunnel SRLG membership does not intersect with excluded SRLGs (if specified)

Assignment is dynamic: as tunnel metrics change, Flex-Algo membership is re-evaluated and IGP advertisements are updated accordingly.

Example assignment for a deployment with three Flex-Algos:

Tunnel	Algo 128 (Low Lat)	Algo 129 (High BW)	Algo 130 (Resilient)
SATCOM (600ms)	Excluded	Included	Excluded
LOS Radio (5ms)	Included	Included	Included
LTE (50ms)	Included	Included	Included

Table 11

The SATCOM tunnel is excluded from Flex-Algo 128 (delay exceeds 100ms threshold) and Flex-Algo 130 (SRLG "satcom" is excluded).

9.3. Traffic Steering

With CONDUIT providing tunnel metrics and Flex-Algo assignments, traffic steering is accomplished through standard SRv6 mechanisms:

* SRv6 Policy Color: Traffic matching a policy with a specific color is steered to paths computed using the associated Flex-Algo

* DSCP Mapping: DSCP values can be mapped to SRv6 policy colors

* BGP Communities: BGP communities can signal desired Flex-Algo

Example mapping for tactical traffic:

Traffic Type	DSCP	SR Color	Flex-Algo
Voice	EF (46)	100	128 (Low Latency)
Video	AF41 (34)	200	129 (High BW)
Critical C2	CS6 (48)	300	130 (Resilient)
Best Effort	BE (0)	0	Default (SPF)

Table 12

10. High Availability

10.1. HA Architecture

CONDUIT supports high availability through active-standby or active-active node pairs sharing a virtual IP address.

In active-standby mode:

* Primary node handles all tunnel management and peer communication

* Standby node maintains synchronized state and pre-established tunnels

* Failover occurs when primary failure is detected

In active-active mode:

* Both nodes actively manage tunnels

* Anycast addressing distributes peer connections

* State synchronization ensures consistency

10.2. Synchronized State

The following state is synchronized between HA peers:

Data	Sync Method	Purpose
Peer List	Periodic + Event	Both nodes know all peers
Tunnel State	Event-driven	Enable fast failover
Metrics	Periodic (1 Hz)	Consistent path costs
Configuration	Event-driven	Policy consistency

Table 13

IKEv2 Security Associations are NOT synchronized. Each node maintains independent SAs with peers. This simplifies implementation and avoids complexities of SA state transfer.

10.3. Failover Behavior

Upon detecting primary node failure:

1. Standby assumes the virtual IP address

2. Standby’s pre-established tunnels become active

3. SRv6 reconverges via TI-LFA (sub-50ms for pre-computed backups)

4. Traffic flows through standby’s tunnels

Target failover time is less than 3 seconds for complete recovery, with TI-LFA providing sub-50ms forwarding continuity for traffic with pre-computed backup paths.

11. gRPC API

11.1. Service Overview

CONDUIT exposes all functionality through a gRPC API comprising three services:

TunnelFabric Service: Primary interface for tunnel management,

including node configuration, WAN interface management, peer discovery and management, tunnel lifecycle operations, tunnel policy configuration, and metric publishing configuration.

Telemetry Service: Provides access to operational metrics, including

current tunnel metrics, real-time metric streaming, historical metric queries, and fabric statistics.

HighAvailability Service: Manages HA operations, including HA status

and configuration, synchronization status, and manual failover triggers.

The complete Protocol Buffer definitions for these services are available in a companion document.

11.2. Authentication and Authorization

All gRPC connections MUST use mutual TLS (mTLS) with certificates meeting the requirements of Section 4.5.

Implementations SHOULD support role-based access control (RBAC) for API authorization. Recommended roles include:

* Administrator: Full access to all operations

* Operator: Read access plus tunnel state changes

* Monitor: Read-only access to status and metrics

12. Security Considerations

12.1. Cryptographic Security

CONDUIT’s security relies on the strength of its cryptographic foundations. The mandatory use of CNSA 2.0 algorithms provides protection appropriate for classified information.

Key security properties:

* All tunnel traffic is protected by AES-256-GCM, providing confidentiality and integrity

* All control messages are authenticated using HMAC-SHA-384

* Key exchange uses ECDH with P-384, providing forward secrecy

* Node authentication uses ECDSA with P-384 certificates

12.2. Threat Mitigations

CONDUIT addresses the following threats:

Threat	Mitigation
Probe Spoofing	HMAC-SHA-384 authentication on all probes
Replay Attacks	Sequence numbers and timestamp validation
Man-in-the-Middle	IPsec with certificate-based auth
Unauthorized API	Mutual TLS with certificate validation
Tunnel Manipulation	Authenticated control protocol messages
Denial of Service	Rate limiting on control plane

Table 14

12.3. Operational Security

Deployments SHOULD implement the following operational security measures:

* Certificates should be rotated at least annually

* Private keys should be stored in hardware security modules (HSM) where available

* All tunnel state changes should be logged for audit purposes

* Rate limiting should be applied to control plane message processing

* Network segmentation should isolate CONDUIT management traffic

13. IANA Considerations

13.1. UDP Port Allocation

This document requests allocation of two UDP ports:

Port	Name	Description
4794	conduit-control	CONDUIT control messages
4795	conduit-probe	CONDUIT probe packets

Table 15

13.2. WAN Type Registry

This document requests creation of a "CONDUIT WAN Types" registry with the following initial values:

Value	Name	Reference
0x00	Reserved	This document
0x01	SATCOM_GEO	This document
0x02	SATCOM_LEO	This document
0x03	LOS_RADIO	This document
0x04	TROPOSCATTER	This document
0x05	HF_RADIO	This document
0x06	CELLULAR_LTE	This document
0x07	CELLULAR_5G	This document
0x08	WIRE_ETHERNET	This document
0x09	WIRE_FIBER	This document
0x0A	WIFI	This document
0x0B-0xEF	Unassigned	
0xF0-0xFF	Private Use	This document

New values in the range 0x0B-0xEF require Standards Action.

14. References

14.1. Normative References

[RFC7296] Kaufman, C., Hoffman, P., Nir, Y., Eronen, P., and T. Kivinen, "Internet Key Exchange Protocol Version 2 (IKEv2)", STD 79, RFC 7296, DOI 10.17487/RFC7296, October 2014, <https://www.rfc-editor.org/info/rfc7296>.

[RFC8174] Leiba, B., "Ambiguity of Uppercase vs Lowercase in RFC 2119 Key Words", BCP 14, RFC 8174, DOI 10.17487/RFC8174, May 2017, <https://www.rfc-editor.org/info/rfc8174>.

[RFC8570] Ginsberg, L., Ed., Previdi, S., Ed., Giacalone, S., Ward, D., Drake, J., and Q. Wu, "IS-IS Traffic Engineering (TE) Metric Extensions", RFC 8570, DOI 10.17487/RFC8570, March 2019, <https://www.rfc-editor.org/info/rfc8570>.

[RFC8986] Filsfils, C., Ed., Camarillo, P., Ed., Leddy, J., Voyer, D., Matsushima, S., and Z. Li, "Segment Routing over IPv6 (SRv6) Network Programming", RFC 8986, DOI 10.17487/RFC8986, February 2021, <https://www.rfc-editor.org/info/rfc8986>.

14.2. Informative References

[RFC2119] Bradner, S., "Key words for use in RFCs to Indicate Requirement Levels", BCP 14, RFC 2119, DOI 10.17487/RFC2119, March 1997, <https://www.rfc-editor.org/info/rfc2119>.

[RFC3550] Schulzrinne, H., Casner, S., Frederick, R., and V. Jacobson, "RTP: A Transport Protocol for Real-Time Applications", STD 64, RFC 3550, DOI 10.17487/RFC3550, July 2003, <https://www.rfc-editor.org/info/rfc3550>.

[RFC4868] Kelly, S. and S. Frankel, "Using HMAC-SHA-256, HMAC-SHA-384, and HMAC-SHA-512 with IPsec", RFC 4868, DOI 10.17487/RFC4868, May 2007, <https://www.rfc-editor.org/info/rfc4868>.

[RFC5869] Krawczyk, H. and P. Eronen, "HMAC-based Extract-and-Expand Key Derivation Function (HKDF)", RFC 5869, DOI 10.17487/RFC5869, May 2010, <https://www.rfc-editor.org/info/rfc5869>.

[RFC7471] Giacalone, S., Ward, D., Drake, J., Atlas, A., and S. Previdi, "OSPF Traffic Engineering (TE) Metric Extensions", RFC 7471, DOI 10.17487/RFC7471, March 2015, <https://www.rfc-editor.org/info/rfc7471>.

Appendix A. WAN Type Characteristics

The following table provides typical characteristics for each WAN type. Actual values vary based on specific equipment, configuration, and environmental conditions.

WAN Type	Latency	Bandwidth	Notes
SATCOM_GEO	600ms	2-10 Mbps	Weather sensitive
SATCOM_LEO	20-40ms	50-200 Mbps	Handover during orbit
LOS_RADIO	5ms	10-100 Mbps	Terrain dependent
TROPOSCATTER	10ms	5-20 Mbps	Over-horizon capability
HF_RADIO	50ms	9.6-64 Kbps	Atmospheric effects
CELLULAR_LTE	30ms	10-100 Mbps	Coverage dependent
CELLULAR_5G	10ms	100-1000 Mbps	Limited coverage
WIRE_ETHERNET	<1ms	1-100 Gbps	Fixed installations
WIRE_FIBER	<1ms	1-100 Gbps	Fixed installations
WIFI	5ms	50-500 Mbps	Short range

Table 16

Acknowledgements

The authors thank the members of the tactical networking community for their input on operational requirements and deployment considerations.

Author’s Address

John Edward Willman V
Department of the Air Force
1800 Air Force Pentagon
Washington, DC 20330
United States of America
Phone: +1 786 994 3023
Email: john.willman.1@us.af.mil

URI: https://www.linkedin.com/in/johnewillmanv