Tunnel-Based versus Tunnel-Free SD-WAN

Ray Mota PhD

ACG RESEARCH

Table of Contents

EXECUTIVE SUMMARY

With the emerging need to connect users to clouds, traditional WAN connectivity using leased lines and MPLS is falling short. Today, the services are everywhere, not just in enterprises' data centers, but also in public clouds, private clouds and SAAS clouds. Users need the ability to reach the clouds directly, and SD-WAN efficiently provides this bridge.

Traditional SD-WAN solves the issues of WAN connectivity by creating virtual networks, overlays, on top of the current transport network. However, overlay-based SD-WAN comes with one caveat: dependence on tunnels. Although the use of tunnels can make the creation of overlays easier, there are issues. The network transport becomes heavyweight and less optimized; this results in poor usage of bandwidth. In some applications, such as VoIP, tunnel overhead consumes as much as 40% to 100% additional packet bandwidth, resulting in poor bandwidth efficiency, increased latency, packet drops and, hence. poor customer experience.

> **"In some application such as VoIP, tunnel overhead consumes as much as 40% to 100% additional packet bandwidth, resulting in poor bandwidth efficiency, increased latency, packet drop and, hence, poor customer experience."**

SD-WAN without virtual networks is both innovative and beneficial, and tunnel-free SD-WAN makes the network more scalable, bandwidth efficient, eliminating fragmentation and delivering better security when compared to the traditional tunnel-based SD-WAN. When tunnel-free SD-WAN is combined with session awareness, the network becomes dynamic and stateful. This results in an intelligent distributed fabric that goes beyond the stateless L2 and L3 connectivity provided by SD-WAN, today. By removing the overhead burden from transport and the need to process such overheads from CPE, the SD-WAN network becomes simple. SD-WAN CPE becomes more scalable yet less costly, resulting in potential capex savings.

This paper addresses the pitfalls of tunnels and explains how SD-WAN can be implemented to be completely tunnel free, natively on IP without the use of additional overhead and how businesses can benefit from it.

Introduction

Traditional WAN connectivity, such as leased lines and MPLS, cannot meet the emerging requirements of the cloud age. Clouds are redefining the way users connect to services, which no longer reside in just one data center. They are everywhere, most notably in private clouds, public clouds and SAAS clouds. Users need the ability to reach services hosted anywhere. SD-WAN enables this kind of connectivity and more by creating software-defined overlays, which are abstracted from the transport underlay. With these overlays, it is easier for a user located anywhere to reach services hosted anywhere.

However, overlay-based SD-WAN comes with one big caveat: dependence on tunnels. Although these tunnels can make the creation of overlays easier, there are issues: the network transport is heavyweight and less optimized; fragmentation is introduced; and scalability and security are negatively impacted. Implementing tunnels comes at certain cost. Some of these issues are known and even documented in RFCs, but some are still new, and the industry is looking for ways to address them. However, it is possible to leverage tunnel-free SD-WAN, which is more native, lightweight and scalable, to implement SD-WAN.

This paper discusses the benefits and the option of tunnel-free SD-WAN. It focuses on GRE, VXLAN and IPSec, which are the primary technologies used in tunnels in SD-WAN. Note that this is by no means an exhaustive list of the tunnels used today but is reflective of the major technologies.

Technology Background: What Is Tunnel?

A tunnel is defined according to the RFC 1853. It encapsulates original IP payload with a new IP header. This means the original IP header is maintained while a new one is added.

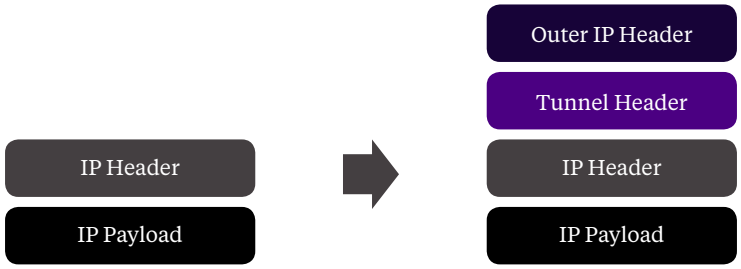

Figure 1. Tunnel versus Tunnel Free

The advantage of tunnel is to bridge portions of the network that have disjointed capabilities and policies. This is the same as encapsulating traffic at the sender end. When traffic reaches its destination, the outer header is decapsulated, leaving the original IP header with its payload. Together, the IP header and the IP payload is called IP packet.

The largest IP packet that a Layer 2 Ethernet frame can carry is called maximum transmission unit (MTU). The default Ethernet MTU size is 1500 bytes, meaning the largest IP packet an Ethernet frame can carry is 1500 bytes. However, not every application uses 1500 bytes. Some applications use smaller packets as do some applications that may need a larger size (also called jumbo frames).

How are tunnels used in SD-WAN?

SD-WAN uses tunnel technology to provide overlays on top of the transport underlay. In Figure 2, a branch office is connected to the main office through two underlays, MPLS and internet. A tunneling technology is used by encapsulating IP traffic between the branch office and headquarters to create two SD-WAN tunnels, one through MPLS and the other through the internet.

This encapsulation (tunnel) helps keep customers' traffic isolated from each other. Therefore, tunnel technology has become a de-facto method of how SD-WAN is implemented today.

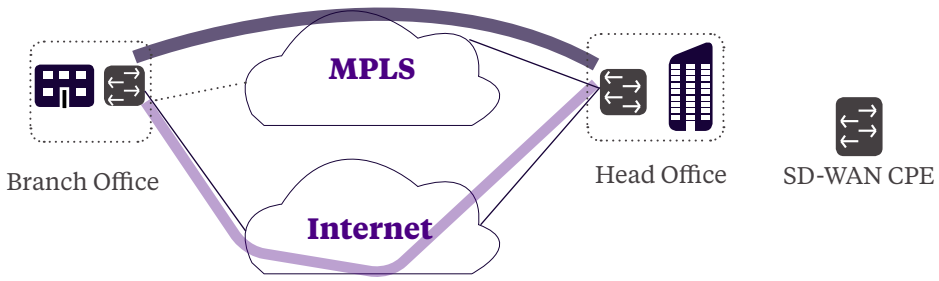

Figure 2. SD-WAN

There are different ways of establishing tunnels. Most common methods are GRE, VXLAN and IPSec. In the following discussion, we analyze them mainly from the perspective of the number of overhead bytes they need.

Overhead

Although there are many kinds of tunnels, all have one thing in common: they add additional bytes to existing IP packets. For simplicity, we consider an MTU size of 1500 bytes. Some examples follow.

Generic Routing Encapsulation (GRE)

GRE is described by the IETF in the RFC 2784[1] and is very popular in the SD-WAN industry. To create a GRE tunnel, a GRE overhead is added to an IP Packet of 1500 bytes (IP header, TCP header and data). This overhead is then removed by a receiving router.

GRE Overhead = 4 (GRE bytes) + 20 (IP GRE Header) = 24 Bytes. Therefore, 24 additional bytes are added to the packet size.

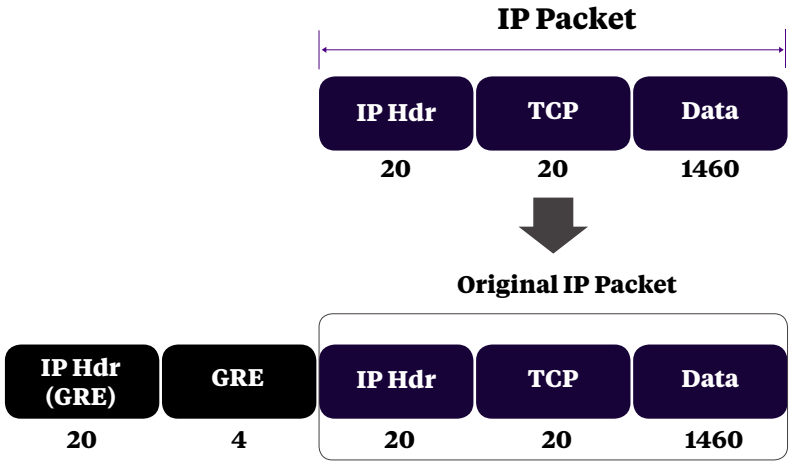

Figure 3. GRE Encapsulation

IPSec with GRE: IPSec (IP security) is described by IETF in the RFC 6071[2]. IPSec is a very popular way of creating encrypted tunnels in SD-WAN. IPSec enables authentication and encryption of IP packets. IPSec uses two main protocols, Authentication Header (AH) and Encapsulating Security Payload (ESP). These protocols authenticate (AH) and encrypt plus authenticate (ESP), respectively. IPSec can be used in conjunction with GRE or VXLAN tunneling protocols. In this discussion we consider IPSec used with GRE utilizing tunnel mode. This means GRE header is added first and then followed by the IPsec header.

IPSec Overhead: 20 (IPSec Header) + 8 (ESP Header) + 8 (Init. Vector) + 2 (ESP Trailer) + 12 (ESP Auth.) = 50 Bytes

Therefore, a total of 50 bytes are added additionally to MTU. This is in addition to the 24 bytes added by GRE.

[1] *https://tools.ietf.org/html/rfc2784*

[2] *https://tools.ietf.org/html/rfc6071*

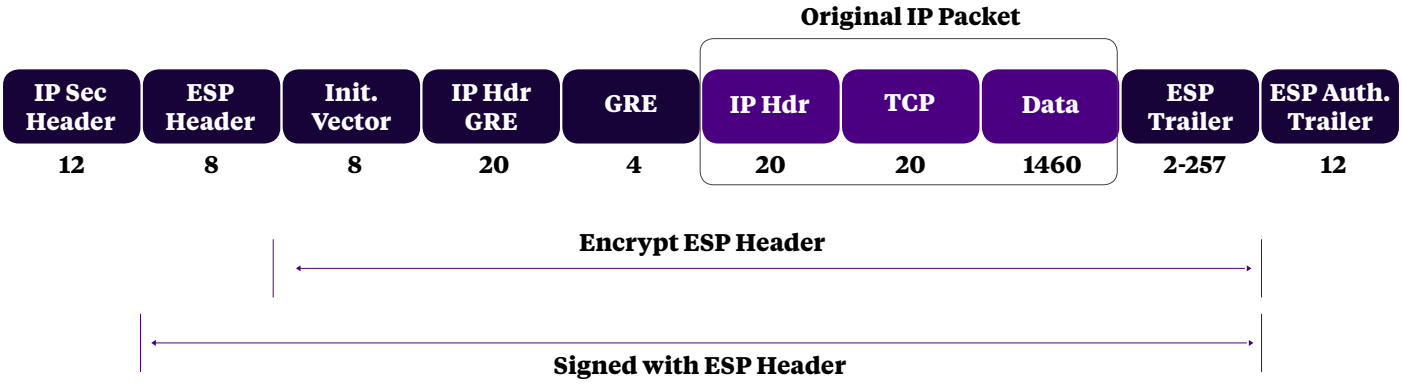

Figure 4. IPSec Encapsulation

Virtual Extensible LAN (VXLAN)[3] : VXLAN is defined by IETF in RFC 7348[4] . Although VXLAN is very popular in the data center world, it has found its way into the SD-WAN industry. Because VXLAN is a Layer 2 encapsulation, it encapsulates the entire Ethernet frame (Figure 5).

VXLAN Overhead = 20 (Outer IP Header) + 8 (Outer UDP) + 8 (VXLAN Header) +14 (Inner Ethernet Header) = 50 Bytes, which is 50 bytes extra as compared to the native IP packet.

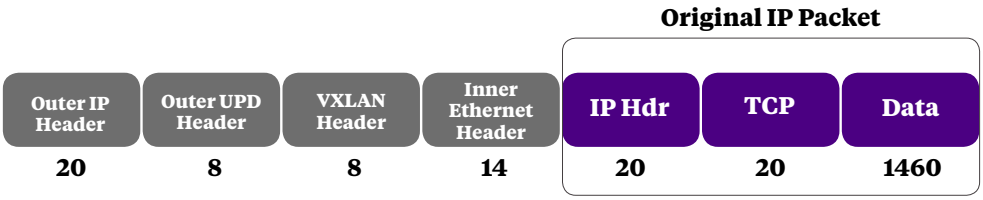

Figure 5. VXLAN Encapsulation

[3] *VXLAN can also be used with IPSec, which adds overhead on top of the VXLAN overhead.*

[4] *https://tools.ietf.org/html/rfc7348*

Issues with the Tunnels

Inefficient Bandwidth Utilization

Although the calculation of the bytes was based on the MTU size of 1500 bytes, in practice every application needs a specific IP packet size, which may be less (or for some applications even bigger). When an application uses a smaller packet size, adding tunnel overhead results in a very inefficient utilization of bandwidth. This is true about common applications, such as VOIP that utilizes a packet size of 60 bytes when utilizing G.729 codec.

For the impact of tunnel on VOIP, consider the case of GRE tunnel:

- IP Packet = 60 bytes (as required for G.729 codec)
- Additional bytes for GRE = 24 Bytes
- Percentage of additional bytes needed = 24/60 = 40%

These means 40% additional bytes are utilized to carry IP packets that are otherwise not needed if the packet is sent natively. This kind of calculation can be repeated for IPSec (with GRE) and VXLAN. The impact of the additional bytes can be clearly seen. The worse-case scenario is IPSec with GRE that needs additional bytes close to 123%. Tunnels require more bandwidth to carry them.

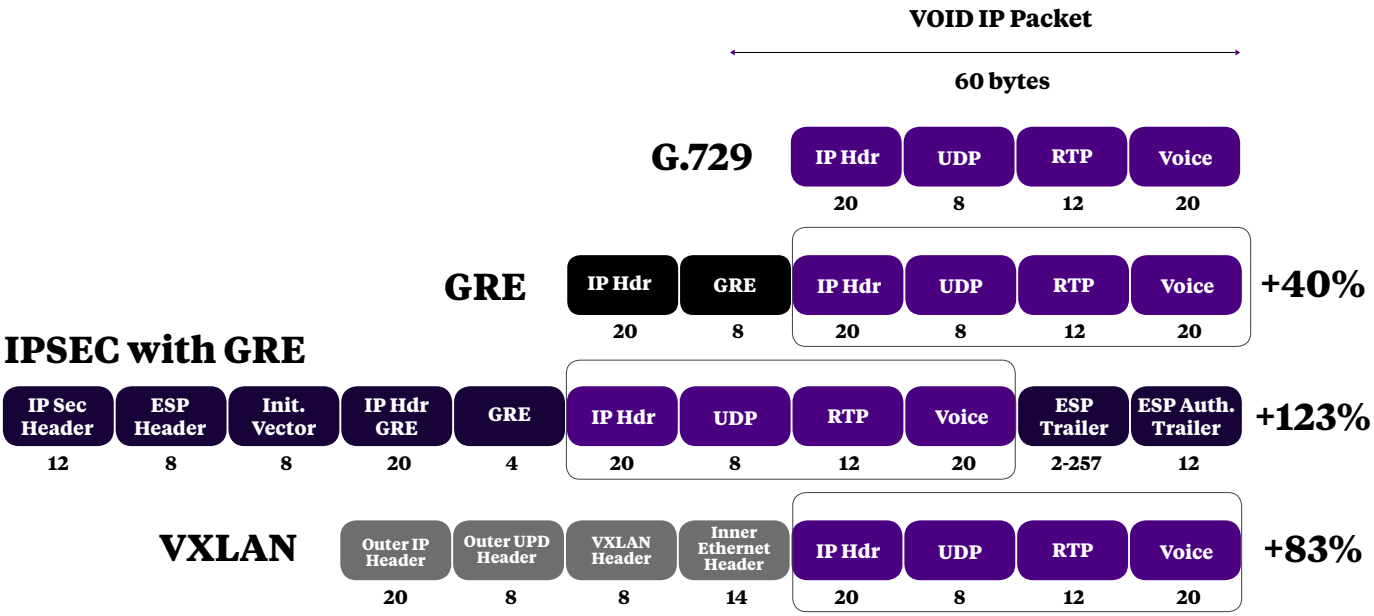

Figure 6. Additional Tunnels Overhead

If there is enough transport bandwidth available, the use of tunnels may not have considerable impact. However, in the case of SD-WAN, it is common to use internet links, and if the link is low bandwidth or if the link suffers from congestion, which if often the case, this leads to performance degradation of the application.

There are studies on how additional overhead because of IPSec tunnels can result in performance degradation on VoIP calls.[5] Packet size increase has negative impact not only on bandwidth usage but also affects the transmission and queueing delay, thus affecting jitter and overall packet delay and the customer's experience.

Fragmentation

In a network, one MTU size is usually set on all routers. Although the standard MTU size configured is 1500 bytes, there are certain applications that require an MTU size closer to 1500 bytes or more. Although it is possible to adjust the MTU size on routers to be larger than 1500 bytes, it requires that every router be configured or there will be compatibility issues with the existing network that may result in packet drops if the intermediate router is not configured or does not support MTU greater than 1500 bytes.

The most commonly used solution is to fragment a packet by enabling fragmentation in a router itself. Fragmentation allows a sending router to fragment packets greater than 1500 bytes in fragments of 1500 (or lower for the last fragment) and this facilitates carrying packets in chunks or fragments. The destination router reassembles those fragments and facilitates the transport of IP packets that are larger than 1500 bytes.

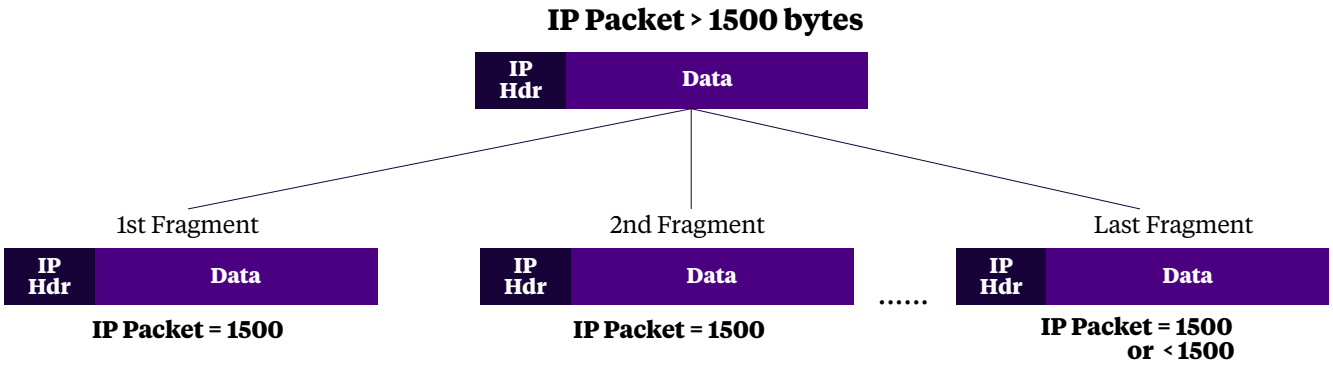

Figure 7. IP Fragmentation

Consider an IP packet size close to but less than 1500 bytes; there is a high likelihood of the packets being fragmented by the sending router. This is because adding tunnel bytes to the IP packets increases their size to more than 1500 bytes thus necessitating fragmentation. It is not uncommon to have IP packets to be close to 1500. Benson et al. studied about characteristics of practical data centers traffics and their results showed that about 40% of packets were over 1400 bytes.

Fragmentation has undesirable effects and results in more work for the receiver as it has to reassemble fragments back into the same order. If one fragment is dropped, the entire IP packet, which is now fragmented, needs to be sent again.

Firewalls that filter packets based on Layer 4 to 7 might have trouble processing IP fragments correctly. If the IP fragments are not in the correct sequence, a firewall may block the noninitial fragments because they do not carry the information that matches the packet filter. This means the receiver will have problems reassembling the packet correctly.

[5] *https://www.researchgate.net/publication/4001426_Voice_over_IPsec_Analysis_and_solutions*

RFC 4459[7] further highlights the security issues related to fragmentation. Fragment reassembly can lead to buffer memory exhaustion if the attacker sends continuous fragments without sending all of them; this could make the reassembly stall until time out. These kind of fragment attacks have happened against firewalls and host stacks and must be taken into consideration during implementation. RFC 4459 highly recommends that fragments should be avoided as much as possible.

It is clear that the tunnels can result in fragmentation, which can create issues such as packet drops and also makes the router work extra in terms of processing power, memory and extra CPU. This could necessitate procuring routers with high-end hardware specifications to cope with fragmentation, resulting in higher capex. Without tunnels the network is simpler and more secure; the routers can scale much better.

Scalability

Another major issue of tunnels is router and network scalability related to the number of tunnels that a router can support. There is always a maximum limit to the number of terminals that a CPE router can support, which could be 10s to 100s (or more for high end CPE). For example, there are 1000 sites for a company. For full mesh configuration between sites, there must be n*(n-1) unidirectional tunnels or n*(n-1)/2 bi-directional tunnels.

Considering bidirectional tunnels, for a company with 1000 sites this would mean at least 499,500 tunnels. Each CPE needs to support at least 999 tunnels to other sites (if unprotected) or double number of them if they are protected. A double number of tunnels is also needed because the traditional SD-WAN requires the IPSec tunnels to be pre-established, otherwise there is a considerable delay in the case of service disruption.

This kind of scale puts a load on the processor of the CPE and challenges CPE scalability. Therefore, SD-WAN vendors usually put a bar on the number of tunnels that should not be exceeded. To cope with this issue and reduce the number of tunnels, vendors often may recommend switching to the hub and spoke model instead of a mesh model.

Full Mesh **Hub and Spoke**

Figure 8. Mesh versus Hub & Spoke

[7] *Fragmentation Issues with In-the-Network Tunneling.*

Hub and spoke allows communication between branches but through the hub only. Although the hub and spoke is a workable solution for data connectivity, real-time applications, such as VoIP and video conferencing, can face latency issues because all the traffic between branches is routed through the central hub, increasing the link distance between the branches. Hub and spoke is a compromise but not an ideal solution for SD-WAN.

If the end-user's business application must have mesh configuration in the network with tunnel-based SD-WAN, it necessitates buying high-end processing SD-WAN routers (as small or medium branch routers may not scale well), resulting in high capex.

Security

Tunnels raise network security concerns. One security issue, discussed under fragmentation, happens when an attacker sends continuous fragments without stopping and makes the receiver run into memory exhaustion that results in time-outs.

Other security concerns raised by RFC 6169 related to tunnels:

- The network cannot apply filters to the tunnel traffic as in case of native IP traffic. This can lead to security gaps. It is called evasion by tunneling and is a problem for network-based security devices, such as network firewalls, IDS and IPS. Although the network filtering may do something in the case of encapsulations based on RFC standards, there is little or no mitigation if IPSec tunnels are used.

- Tunneled traffic also presents challenges to tools such as DPI-Deep Packet Inspection (DPI). This makes it difficult to apply the same controls as those applied to native IP. Some tunnels are easy to identify if they use well-known UDP or TCP port. Other protocols either use dynamic ports or share ports with other protocols (for example, tunnel over HTTP). Network-based devices that want to passively inspect the encapsulated traffic must inspect all TCP and UDP traffic. This is inefficient and too slow, especially if it is needed to take an urgent action on the packets.

If a customer has options, no tunnel traffic is recommended over tunnel traffic.

Is Tunnel-Free SD-WAN Possible?

With so many disadvantages related to tunnels, is it possible to have a tunnel-free SD-WAN? Without tunnels, native IP delivers the following benefits:

- No bandwidth tax
- No fragmentation because of tunnel overheads
- No scalability issues

However, out-of-the-box native IP cannot provide customers' isolation, for example, IP VPN like behavior, which would be the least needed to call a technology SD-WAN. What are the options then? We reviewed the market and found that tunnel-free SD-WAN is not only possible but being delivered today. It is already deployed in some big production networks. This innovative SD-WAN solution uses Secure Vector Routing (SVR) technology that enables enterprises and service providers to build service-centric fabrics without the debts of tunnels.

Tunnel-Free SD-WAN Implementation using Secure Vector Routing

We analyzed the emerging model for SD-WAN technology called Secure Vector Routing that uses a session-based approach to eliminate tunnels. SVR implements SD-WAN without tunnels using native IP behavior. This is done without encapsulating traffic but by rewriting source and destination IPs through Network Address Translation (NAT) combined with using metadata only on the first packet for signaling the sessions.

These kind of sessions are set for every tenant, and every tenant is assigned its own NAT address. This isolates customers' networks, keeping native IP behavior and bringing SD-WAN functionality without the use of tunnels, forwarding sessions, not just packets. A session-oriented perspective enables end-to-end, fine-grained control, and visibility. The term secure vector routing describes how it routes packets, which provides distributed control and simple intelligent service-based routing. SVR ensures that a bidirectional session follows the same path. The symmetric flow allows intelligent routes and control over sessions rather than control over packets. SVR based routers transform a stateless L2 or L3 network into one that is fully session aware.

The added benefit of SVR is the segmentation capabilities that tunnel-based SD-WAN is not able to provide. Traditionally, network segmentation has been done using VLANs. More recently using overlays, such as VXLAN, GRE, IPSec, have been used to extend segments between sites in different locations. Thanks to session-based awareness SVR instead uses hyper-segmentation in which a session is treated from encryption, authentication and routing point of view. All this is done without the use of overlays on the existing private or public networks. This is much more granular, simpler to implement and scalable beyond network boundaries. The granularity with which such segmentation can be implemented session by session results in much better utilization of transport links. Some customers have also reported better utilization of MPLS links resulting in reducing the costs of MPLS links.

Secure vector routing provides tremendous value from a security point of view because it provides zero-trust security by default. It does not require IPSec for encryption; it can enable encryption on payload using AES-256 or AES-128. Additionally, it has a feature of adaptive encryption, which avoids double encryption. It detects if payload is encrypted and does not re-encrypt the traffic in that case.

Conclusion and Summary

The following summarizes the benefits of tunnel-free SD-WAN compared to tunnel-free-based SD-WAN:

Tunnel-Based SD-WAN	Tunnel-Free SD-WAN
Tunnels forward packets instead of sessions, which leads to a static nature of connectivity. For session awareness, additional applications need to be added such as DPI. Stateless L2 and L3 network fabric.	Sessions are forwarded, which leads to stateful and dynamic routing, resulting in intelligent and distributed fabric. Stateless L2 and L3 network is transformed to session- aware data plane.
Additional bandwidth tax that can be as high as 123%.	No overhead means no bandwidth tax.
Risk of fragmentation if IP packet size reaches close to 1500. Fragmentation can result in packet drops during reassembly.	No risk of fragmentation (because of SD-WAN) as additional compensation for overhead bytes is not needed.
Scalability issues because of tunnels that necessitates hub and spoke configuration instead of mesh; suboptimal design for real-time traffic such as VoIP and video.	As there are no overheads, there are no risks of scalability. Thousands of session can be created. As scalable as IP.
For large-scale networks that demand more granular segmentation, static and complex to implement/maintain segmentation.	Hyper-segmentation based on sessions. Much more granular, easier to implement and results in better utilization of MPLS links. This can potentially reduce MPLS link costs.
Security risks can happen because of fragmentation. Evasion by tunneling can be a problem for network- based security devices such as network firewalls, IDS and IPS. Inefficiency because of potential double encryption in traditional SD-WAN and of re-encrypting customer's traffic even if it is encrypted.	Zero-trust security is default. End-to-end stateful session management and encryption. No need for IPSec as encryption can be done on the payload using AES-128/256. Adaptive encryption by detecting encryption on the customer's traffic; no need to re-encrypt the traffic.

SD-WAN has shifted the paradigm of networking from static connectivity to fully flexible connectivity where users need to reach services efficiently that are hosted anywhere. Tunnels do not scale well for the requirements of SD-WAN where any-to-any connectivity is required on the fly. A service-centric network that does not depend on tunnels is an ideal way to implement SD-WAN. Potential SD-WAN users need to think out of the box, question the tradeoffs associated with tunnels, and explore tunnel-free options available to them for implementing their SD-WAN.