HYPERSEGMENTATION UNDER THE HOOD

CONTENTS

INTRODUCTION

Network segmentation is the logical partitioning of a single physical network into multiple, independently managed networks. Used primarily in the interests of performance and security, data travels over common physical infrastructure but is segregated using a discriminator. The most common type of network segmentation deployed today is the virtual local area network, or VLAN. VLAN segmentation has been a mainstay of network design for nearly two decades; ratified by the IEEE in 2003 as 802.1q (but deployed long before this), packets are segregated by use of a four byte shim known as a VLAN tag, that was retrofitted into the Layer 2 (Ethernet) header. Ethernet switches are responsible for adding tags, removing tags, and ensuring that packets on one virtual LAN do not comingle with packets on another virtual LAN.

VLANs have been a fine tool in the network architect's toolbox, but have several limitations. First, by virtue of the fact that VLAN tags are shimmed into the L2 header of Ethernet frames, they are generally limited to a physical site, creating its own broadcast domain. Second, there are a relatively small number of them (4,094 in all); although this probably seemed like a king's ransom in 2003, modern network design – particularly in data center and cloud networks, and particularly in environments where there is heavy use of virtualization – can speed past that number in a heartbeat.

Overlay networking, comprised of a diverse set of protocols like IPsec, GRE, VXLAN, and GENEVE (and even 802.1ad, which cascades VLAN tags into more VLAN tags), were created to extend segments between sites, to create more segments than the four byte VLAN tag affords, and to scale segmentation to large networks (particularly important in the era of compute virtualization). These architectures typically create L3 groups that tie together sets of resources into overlay segments, and can be used in conjunction (by mapping L3 segments onto L2 segments at other locations) to "extend" segments to alternate locations. Provisioning the intricate dance of interplay between these different networking designs grew so complex, a whole orchestration industry was born.

At 128 Technology, we decided to approach segmentation from a different tack. Instead of building a new [over]layer on top of existing legacy technology, we rethought how sources and sinks of traffic communicate. And we designed a segmentation model we call *hypersegmentation*: one in which each bi-directional communication between a source and sink of traffic – what we call a *session* – is treated unto itself, from an encryption, authentication, and routing point of view. Best of all, it does this over your existing network infrastructure, irrespective of public/private network boundaries, broadcast domains, or any other infrastructure.

This is a vital concept for network and security architects looking to segment their networks to provide scalability, availability, security, and manageability for different cross sections of their traffic profiles.

ENTER HYPERSEGMENTATION

128 Technology has developed a way of segmenting networks down to single endpoints and services on those endpoints, while providing a named based hierarchy, enabling easy and effective administration and enforcement of security policies.

Traditional network segmentation is zone-based, defining users into trusted and untrusted zones and providing many security layers within that network or subnetwork. All the users, computers, and servers within a given zone can freely talk with each other. In a LAN environment, this would equate to sharing an Ethernet broadcast domain. To go between zones requires going through a firewall, which requires an explicit policy to allow the IP traffic through. The firewalls control the so-called "north/south" movement of network traffic into and out of the zone, and allow "any-to-any" communication within a segment.

Rather than limit yourself to L2 broadcast domains, or create overlay networks that encapsulate data over and over, what if you could create an entirely segmented network without overhead? One in which sessions from users on the same broadcast domain can be authenticated, encrypted, and routed uniquely from each other? Network segmentation should span end-to-end across boundaries and borders between mobile users, the cloud applications they consume, and back end services.

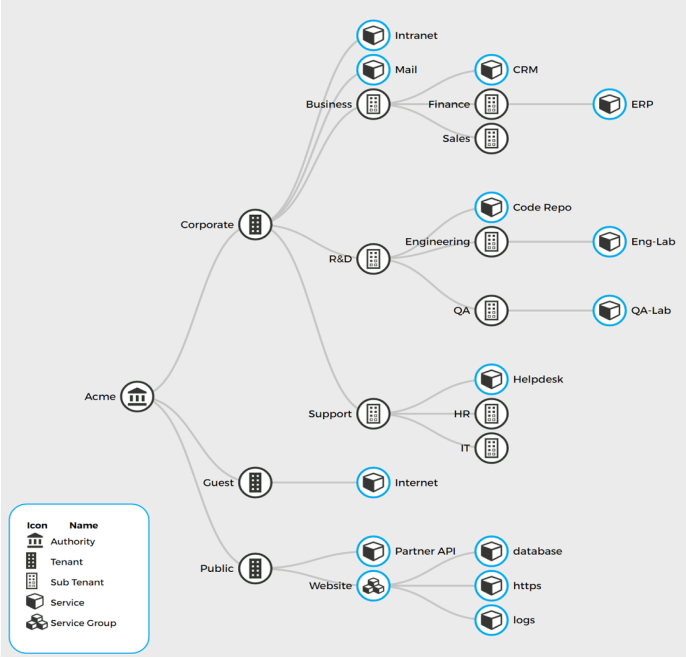

Figure 1: Organizing tenants and their services

The *hypersegmentation* found in the 128T Networking Platform (128T) has four distinct attributes that make it perfectly suited for modern network design:

1. Each 128T can apply intelligent, dynamic encryption for each session sent to another 128T.

2. Each session is individually authenticated by each 128T, on a hop-by-hop basis.
3. Route policies apply universally, including across borders of networks (public/private, through firewalls, IPv6, IPv4).
4. The 128T enforces per-session controls such as rate of establishment and bandwidth constraints.

We'll describe each of these in detail, but first, some context on the 128T *data model*.

TOP-DOWN DESIGN

128T is a software-based router that uses an innovative data model that lets network architects describe how their network will be used in a whole new way. It starts with the services that form the raison d'être of the network: things such as your CRM system, ERP system, mail, voice, and web resources. Access to these services is granted based on what we call *tenancy*: each tenant in the 128T data model represents a collection of users and their devices that share common policies – things such as access policies and security policies. Unlike zone-based schemes, tenancy is applied and enforced at every 128T network-wide: your tenant's policies "stretch" across your network.

Administrators define the tenants (user populations) that use the network and the services that the network offers. Using an intuitive, text-based, associative language, administrators grant or deny access to those services for members of the various tenants on the network. These tenants and services are shared among all the 128Ts within an administrative domain (what we call an *Authority*), along with security properties such as authentication and encryption keys. This ensures network resources are offered only to those permitted to use them.

Under the hood, these tenant and service definitions govern the construction of each 128T routing information base (RIB) and forwarding information base (FIB).

NETWORK TENANCY

As new sessions arrive at a 128T, the router will attempt to classify the source of that session request into one of its configured tenants. This classification is done in one of three ways:

1. The session request arrives on an interface that has been designated as exclusively belonging to a single tenant.
2. The session request arrives on an interface from a prefix that has been specified as belonging to a tenant (i.e., a single interface can be partitioned into separate tenants based upon subnet mask).
3. The session request arrives containing 128T metadata, supplied by an adjacent instance of the 128T, which has already classified this packet as belonging to a specific tenant.

Should none of these result in a definitive determination on the tenant of the source of this session request, the session is associated with a *global* tenant.

Once the tenant has been identified – either as a specific tenant, or as the global tenant – this acts as a filter into the 128T FIB. Only the routes associated with that tenant are available to that user group. While this somewhat resembles the way a legacy router uses virtual routing and forwarding (VRF) to create separate RIBs and FIBs, the segment by *tenant* is pervasive among all routers within an Authority by design, and is applied ubiquitously among all varieties of networks: public IP space, private, cloud, IPv4, IPv6, etc.

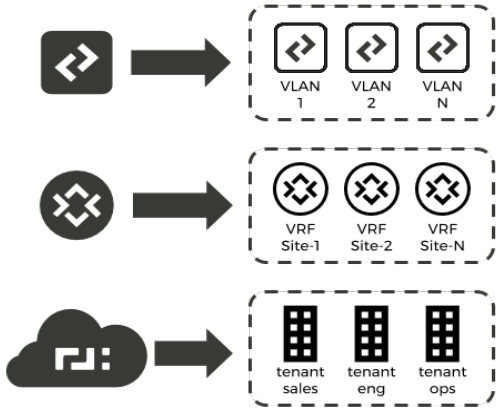

Figure 2: L2, L3, and session-oriented segmentation

THE SERVICE-ORIENTED FIB

The 128T FIB – in addition to being segmented by tenant – is L4 aware: it includes matching on transport protocol (e.g., TCP and UDP) and L4 port. FIB entries are applied based on the services that were configured and made available based on tenancy and access policy. A service-oriented FIB combines firewalling and routing in a unique way: it's possible for your network's *routing layer* to have a route to a specific machine's HTTPS port, but not for that same machine's SSH port. This is one route policy, constructed within an Authority, that has subtle but powerful ramifications for network design. Rather than build an "anywhere-to-anywhere" routed network and cordon off undesirable access using firewalls, describe a network's services and create an intelligent routing plane.

SESSION-ORIENTED PROCESSING

Assuming a session request passes muster by way of a matching FIB entry, it's handled by the 128T service logic to determine its disposition. The 128T chooses a path for this session based not only on the typical routing fare (least cost route), but also influenced by the attributes required for that service – things such as tolerance to packet loss, latency, and the like. Load balancing across multiple paths based on real-time bandwidth data, with sensitivity to the bandwidth consumption that this session request will further impose, operate in unison within the routing plane to ensure a network's resources are utilized most efficiently and effectively on a session-by-session basis.

The 128T uses information it harvests from the network to affect its forwarding decision process. This data comes in one of two forms: Bidirectional Forwarding Detection (BFD) and 128T metadata. Furthermore, 128Ts that glean network data can share that information with their 128T peers, allowing your network to organize itself optimally.

Computing Path Metrics

Each 128T, when supplied with information about adjacent 128Ts, will proactively measure link quality between themselves using BFD packets. The use of BFD packets to test link liveliness is nothing new – in fact, the protocol was invented for this purpose! However, the 128T implementation of BFD augments the protocol to derive empirical data about three common impairments: loss, latency, and jitter. By periodically sending a small stream of BFD

packets at regular intervals, the receiving 128T can measure on link quality (and because they're sequentially numbered, it knows whether packets have been reordered). It reports this information back to the sender, which feeds into the route selection algorithm in the event that a service has any applied service level agreements (what we call *service policies*).

Concurrent with the BFD exchange, each 128T also reports occupancy data within metadata. When a new session is established, the terminating 128T reports session count information to the originating 128T, informing that device's decision process for future inbound sessions. This avoids the issue when two sources of traffic are sending to one sink of traffic and each source has a blind spot to the traffic supplied by the other.

NETWORK SECURITY RETHOUGHT

128T uses its knowledge of your network's topology (including other 128Ts and legacy routing equipment) to chart an optimal course through the network. It supplies metadata (a small, one-time inclusion by the first 128T in the path, containing information discerned about the inbound session) to the next 128T in this path, which does the same in turn for every subsequent 128T. Importantly, this metadata includes an HMAC signature, authenticating it using SHA-256 and based on a common security policy known to the two routers. Each router receiving this metadata will authenticate the payload to ensure that it has not been tampered with. This metadata is also encrypted using public key cryptography. This is performed at all subsequent hops through 128T nodes, validating that the sender is who it claims to be, implementing what the industry has termed Zero Trust Security (ZTS).

In addition to authenticating the sender, each 128T can also encrypt the payload of data transmitted between clients and servers. Also done using public key cryptography with keys exchanged a priori, the 128T can encrypt payload prior to egressing the initial 128T and decrypt it prior to egressing the last 128T, using strong AES-CBC ciphers.

There are three types of keys that are used by 128Ts when exchanging information between themselves:

1. Network keys are used to encrypt metadata when information is sent between routers.
2. Tenant keys are (optionally) used to encrypt all traffic flowing within a given tenant segment.
3. Service keys – when configured on a service – all traffic sent to that service will be encrypted using the keys specified here.

128T uses service keys rather than tenant keys in case both may apply to a given session. Each of the tenant and service security policies may set adaptive encryption.

Because more and more traffic traversing networks today is encrypted by default, 128T can also *selectively* encrypt traffic that it determines to be in the clear. We call this "adaptive encryption," and it works by skimming the first handful of packets in a session, looking for a telltale pattern (such as an IPsec or TLS header). When it finds this, a 128T can be told to avoid re-encryption – saving bandwidth overhead and compute cycles on all network equipment involved in the security association.

Authenticated pathways, consistently encrypted data across your entire network, done in a security-conscious, performance-friendly way: that's rethinking ZTS.

SUMMARY

For decades, routed and overlay networks have been designed around an increasing amount of prefixes, tags, labels, identifiers, and encapsulations that are only significant to the networks and their topology. "The tail wagging the dog," is a fitting expression. The 128 Technology approach to segmenting traffic is entirely distinct, introducing a whole new set of tools for network design intended to allow operators to build the network around the services it is meant to deliver, rather than around the network itself. It enables the routing and policies of the network to be expressed in terms, allowing for semantic meaning to be applied to businesses and applications, not network topology.

Furthermore, tenancy is not just some useful abstraction of the same archaic techniques behind the scenes. No, that would be perpetuating the same incrementalism that has gotten networks into the overly complex state they find themselves in. Rather, tenants are present in the very forwarding tables and packet pipelines of the 128T, making them a foundational component of the routed network itself. The ability to partition service availability – afforded by the network using tenants – is one of the many ways 128 Technology is fixing the network, by fixing the router.

To learn more about 128 Technology, visit www.128technology.com or call 781.203.8417.

www.128technology.com | info@128technology.com