Network Working Group
Internet-Draft
Intended status: Experimental
Expires: August 3, 2016

T. Henderson
University of Washington
S. Venema
Polyverse Security
D. Mattes
Tempered Networks
January 31, 2016

HIP-based Virtual Private LAN Service (HIPLS)
draft-henderson-hip-vpls-10

Abstract

The Host Identity Protocol (HIP) and architecture adds a cryptographic name space to name Internet hosts. This draft describes a use case of the HIP architecture, which is to provide a HIP-enabled virtual private LAN service (VPLS) over an untrusted network. In this case, HIP is used to secure tunnels between the provider edge (PE) equipment.

Status of This Memo

This Internet-Draft is submitted in full conformance with the provisions of BCP 78 and BCP 79.

Internet-Drafts are working documents of the Internet Engineering Task Force (IETF). Note that other groups may also distribute working documents as Internet-Drafts. The list of current Internet-Drafts is at http://datatracker.ietf.org/drafts/current/.

Internet-Drafts are draft documents valid for a maximum of six months and may be updated, replaced, or obsoleted by other documents at any time. It is inappropriate to use Internet-Drafts as reference material or to cite them other than as "work in progress."

This Internet-Draft will expire on August 3, 2016.

Copyright Notice

Table of Contents

1. Introduction

Virtual private networks (VPNs) are popular in the wide-area Internet and also among enterprises that wish to separate multiple LAN broadcast domains across shared network infrastructure. Several techniques have been defined to provide VPNs at different layers in the stack, including layer-1 [RFC4847], layer-2 (virtual LAN, virtual private LAN service (VPLS), and pseudo-wire (PW)) [RFC4664], and layer-3 (virtual router and BGP/MPLS provider-provisioned VPNs) [RFC4176].

The Host Identity Protocol (HIP) [RFC7401] and architecture [RFC4423] adds a new public-key-based name space for use as host identifiers in Internet protocols. HIP specifies a means for hosts to use public keys to authenticate one another over Internet protocols and to set up secure data channels using Encapsulating Security Payload [RFC7402] and possibly other transports in the future.

This document describes how HIP can be used to create a customer Virtual Private LAN Service (VPLS) overlaid on top of a standard IPv4 and/or IPv6 provider network. Using the nomenclature in RFC 4664 [RFC4664], a VPLS connects several physically separate LAN segments

into a single logical LAN segment. The Provider Edge (PE) devices that connect the Customer Edge (CE) devices behave like a learning bridge, and the CE devices may be any layer-2 or layer-3 device, including hosts, routers, bridges, or switches.

In the specific use case described, the tunnels between PEs are realized by Encapsulating Security Payload (ESP) tunnels, whose management is controlled by the Host Identity Protocol (HIP) signaling protocol. Each PE device is assigned a cryptographic host identifier, which may be bound to other identifiers in the system via certificates or other means. The HIP signaling protocol is used to allow PE devices to authenticate one another and to build secure tunnels over untrusted provider network infrastructure. Extensions to HIP are described to allow the PE devices to integrate with a public-key infrastructure, in order to ease deployment.

Readers may note that this application of HIP differs from the traditional implementation of HIP within end hosts. The key differences are that HIP is here implemented within a middlebox (using the terminology of RFC 4301 [RFC4301], a "bump-in-the-wire" implementation) and that the payloads of the ESP-encrypted datagrams are not transport protocol data units (PDUs) but instead are layer-2 frames.

2. Terminology

Terminology is reused from [RFC4664] and and [RFC7401].

3. Reference model

Section 2.2 of RFC 4664 [RFC4664] specifies the VPLS reference model where PE devices that are VPLS-capable provide a logical interconnect such that CE devices belonging to a specific VPLS appear to be on a single bridged Ethernet. A VPLS can contain a single VLAN or multiple tagged VLANs.

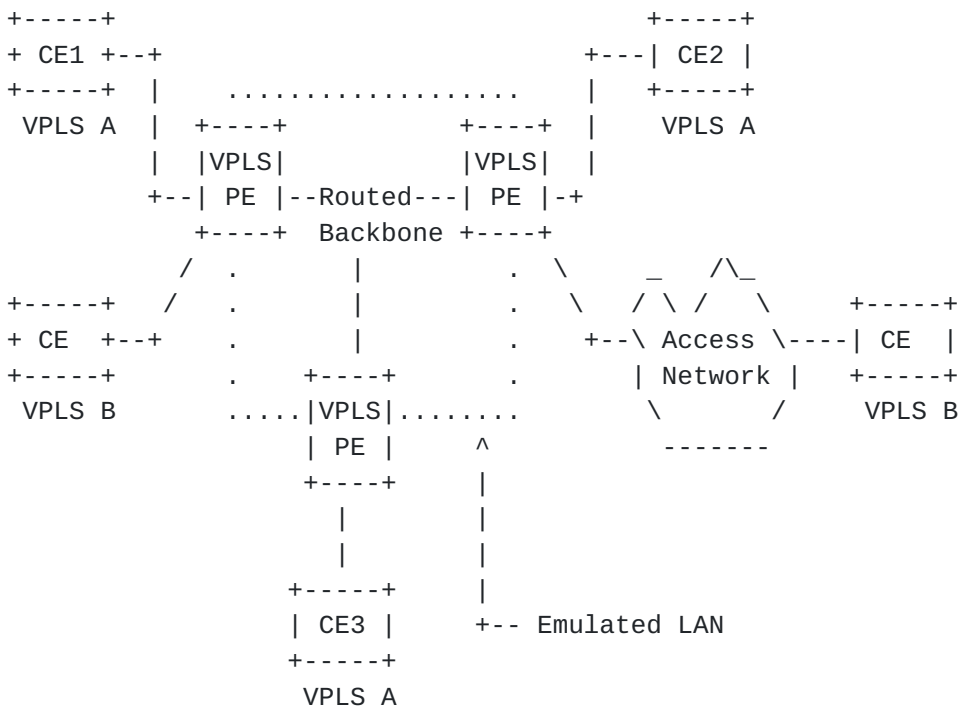

Figure 1: Reference model

Figure 1, copied from Figure 2 of [RFC4664], depicts the reference model for this use case. A number of CE devices are connected to PE devices over layer-2 networks. Although not shown in the figure, each CE device may be reachable by one or more PE device (for example, CE1 and CE3 may also be able to reach each other directly without using the VPLS). Moreover, the connectivity of the L2 networks (and correspondingly, between a given PE and CE) may change over time. No assumptions are made about the capabilities of the CE devices. From the perspective of the CE devices, each other CE device is reachable, using broadcast, multicast, or unicast, as if it were on the same LAN segment. Therefore, the service provided by the PE devices is that of a L2VPN. Since this is a L2VPN, CE devices are free to use higher layer protocols such as IPv4 and IPv6 and domain specific protocols such as those found in industrial control systems.

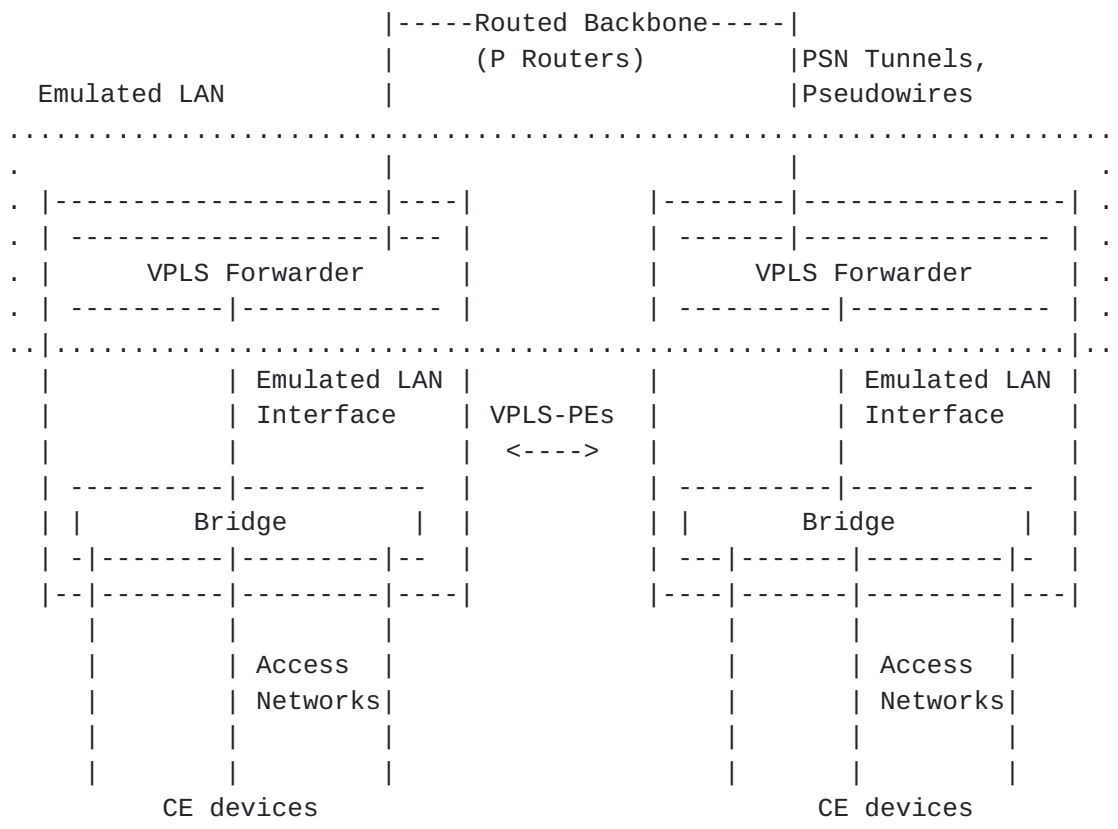

Figure 2: PE Reference model

Figure 2, copied from Figure 3 of RFC4664, depicts the design model for the PE. In this model, a CE device attaches, possibly through an access network, to a "bridge" module of a VPLS-PE. Within the VPLS-PE, the bridge module attaches, through an "Emulated LAN Interface", to an Emulated LAN. For each VPLS, there is an Emulated LAN instance. Figure 3 shows some internal structure to the Emulated LAN: it consists of "VPLS Forwarder" modules connected by pseudowires, where the pseudowires may be traveling through PSN tunnels over a routed backbone.

A "VPLS instance" consists of a set of VPLS Forwarders (no more than one per PE) connected by pseudowires. In our application, it is the HIP-enabled ESP tunnels that constitute the pseudowires.

The PE devices are interconnected by an IP-based network. This network may be IPv4-based or IPv6-based, or a hybrid. The PEs are responsible for providing a secure (encrypted, authenticated) tunnel over which Layer-2 frames may flow betweeen CEs that are interconnected by the VPN. The PE devices are also responsible for authenticating the peer PE devices as belonging to the same overlay

(L2VPN). Furthermore, PE devices may be responsible for maintaining access control lists (ACLs) that govern which CEs are permitted to talk to which other CEs. In addition to IP and MAC addresses found in ACLs, the ACLs may also use the cryptographic identities already bound to the PE devices for use by the HIP protocol.

To build tunnels, the PEs must use pre-provisioned configuration information or must consult, on-demand, a mapping database (such as DNS or an LDAP server) to find the bindings between PE and CE device. These bindings may be secured by a public key infrastructure (PKI). PEs may change their point of attachment (and also, their IP address) to the IP-based network, and may be multihomed to the IP-based network (see PE3 in the above figure), and the PE devices must accommodate such changes such that they are transparent to the L2VPN overlay and the CEs.

In this model, the PE devices use HIP as follows. Each PE device is assigned (provisioned with) a unique name, such as a serial number or asset tag, and with a public/private key pair. This unique name may be bound to the public key using an X.509 certificate. The L2VPN is also given a name. Each PE device knows which of its interfaces belong to a particular named overlay, and which of its interfaces belong to the underlay (the "routed backbone" in Figure 2). Each PE device knows or learns which CE devices it is fronting for, and how to obtain mapping information that maps a remote CE to a remote PE device.

The tunnels established between PE devices are HIP-enabled ESP tunnels. HIP signaling between PE devices is used to establish and maintain the tunnels. A certificate, signed by a trust anchor in the system, binds the PE name to the PE's public key; this public key is used as the host identity in the HIP exchanges. The HIP exchanges carry a PE's certificate, thereby allowing a remote PE to authenticate the PE as a member of the overlay. HIP signaling may also be used between the PE devices and the mapping database, or this communications channel may be secured by other means.

4. Service description

RFC 4665 [RFC4665] describes service requirements for L2VPNs, and outlines a number of options for variations on the L2VPN design. In this section, we describe the HIPLS service in terms of the RFC 4665 taxonomy.

With respect to Section 5 of RFC 4665, we are describing a full VPLS solution; any variations or caveats should be documented according to Section 5.1 of RFC 4665. For example, a VPLS must support unicast,

multicast, and broadcast traffic, even if realized with ESP unicast-based tunnels.

5. System description

In this section, we walk through how the HIP-enabled VPLS can be provisioned and how it operates in a few use case scenarios.

In the following, we refer to each L2VPN as an overlay network, and to the routed backbone as the underlay.

5.1. Provisioning the PEs

At a minimum, a network operator must define a unique overlay name, and must authorize (or list) the PEs that belong to that overlay. In particular, the interfaces (overlay and underlay) that belong to the system must be identified for each PE. Additionally, each PE must possess a public/private key pair, which must be accessible to a host via a smart card, Trusted Platform Module (TPM) hardware, or a local file.

The PEs must be able to authenticate the other PEs in the underlay as belonging to a given overlay. One way to do this is to pre-provision a list of PEs (and their HITs) that belong to the overlay, and deploy this list on each PE in a static configuration file. A drawback to this approach is that whenever the set of PEs on the overlay changes, each PE's master list must be edited. An alternative is to deploy an authorization system in which a PE's key is authorized by a server as belonging to that overlay.

In addition, there are a number of other configuration items that may either be pre-provisioned or dynamically learned. These include access control lists, associations between PE devices and local CEs, and associations between remote PE devices and remote CEs. All of this type of information may either be pre-provisioned in static configuration files, or stored in a database accessible on the underlay.

5.2. Walkthrough of unicast protocol operation

Referring again to Figure 1, consider the case in which CE1 wishes to send an IPv4 unicast datagram to CE3, and no corresponding session state exists between the respective PEs. We assume that CE1 and CE3 both share a network prefix, and that CE1 first sends an ARP request or Neighbor Discovery on its local LAN segment. This request is picked up by PE1 which listens promiscuously on its LAN segment. No other devices respond to this request.

PE1 learns that it is the responsible PE device for the source MAC address of the ARP request, and stores this forwarding entry in its forwarding database (address learning). Note that some implementations may populate the forwarding database manually. Manual configuration is required for CE devices that never send an L2 frame ("listen only" devices) or that only send L2 frames when they have received instructions to do so. Since the ARP message is a broadcast layer-2 frame, the PE device must either perform a proxy-ARP function or must send the ARP request to all other PEs on the overlay. Therefore, a means whereby each PE knows all of the other PEs in the overlay is required, either by static configuration or by dynamic discovery.

Next, the PE device must forward the ARP request to all peer PEs servicing a particular overlay, or to a specific peer PE if the MAC-to-PE mapping is already known (either by static configuration or earlier dynamic discovery). Since the PEs communicate with each other via HIP, the PE forwarding the ARP must build a HIP tunnel to each target PE if it does not already exist. The source PE wraps the L2 frame within the ESP payload, fragments it if necessary, and sends to the remote PEs where it is detunneled and placed on the remote access network segment again as a L2 broadcast frame. From this point, the intended host will ARP reply with a unicast frame. This frame should be mapped to the ESP association back to the originating PE.

Note that flooding of broadcast datagrams in an L2 network is prone to loops. There may be other transparent bridges present in the access network. Therefore, the PE devices must implement and participate in an 802.1d spanning tree algorithm. Note that the nature of 802.1d and the number of broadcast frames typical in most networks will require the setup and maintenance of a full mesh of ESP associations between PEs on an overlay, in general.

5.3. Names and access control lists

The name by which the PE devices know one another, at the protocol level, is the HIT, which is a hash of the host identity public key. This key can be used to authenticate messages from PE devices purporting to be a named PE device.

However, from a management perspective, the names that operators will want to use in configuration files and in access control lists should be more operationally relevant, such as human- friendly strings and asset tags. Certificates are used to bind a PE device's operational name to its HIT. The HIT is obtained as usual, as a hash of the PE device's public key. All PE devices in the overlay must share a common set of CAs.

Certificates should be presented as parameters in the base exchange, to allow peer PE devices to validate them.

5.4. Walkthrough of multicast operation

Multicast operation is similer to that described in the section on handling of broadcast ARP requests.

5.5. Mobility, multihoming, and address families

The PE devices may be mobile or multihomed on the underlay. The HIP mobility mechanisms [RFC5206] may be used in this case to preserve existing security associations and to update database records upon such changes to the underlying IP addresses.

The underlay may itself be a combination of IPv4 and IPv6 network segments. A given overlay may be supported by either or both IPv4 and IPv6-based ESP security associations.

The CE devices may be multihomed to PE devices. In this case, the PEs must coordinate to ensure that only one PE sources ingress frames destined from CE4 to another CE. The PE devices may have "backdoor" connections with one another. The 802.1d spanning tree protocol should alleviate problems of this sort.

6. Proposed extensions to HIP

The system described above relies on the ability of the PE devices to exchange certificates in the R1, I2, and UPDATE messages, based on local policy. Note that passing of certificates in the HIP exchanges is not strictly necessary, but it will reduce latency if the host proactively provides its certificate as part of the signaling exchange. Work is already underway in the HIP working group to define such a certificate (CERT) parameter [RFC6253].

The system described above can be thought of as a "bump-in-the-wire" type of HIP deployment. Conceptually, what is being encapsulated is not a transport PDU but instead a layer-2 frame. Therefore, HIP implementations in the PE devices need to be able to successfully encapsulate and decapsulate such frames; i.e., this system alters the protocol processing in the stack compared to a host-based HIP implementation.

An additional change is that layer-2 (and, by extension, layer-3) multicast and broadcast frames, as well as layer-2 control frames such as bridge PDUs, must be passed as needed. This requires a capability for the PEs to send a copy of each such frame to all other PEs in the overlay. One technique to do this is to replicate each

frame and send to each other PE in the system. To support such a transmission framework, N*(N-1) tunnels must be maintained collectively between the PE devices. Alternatively, a constrained system may be deployed that does not support multicast or broadcast, nor bridge PDUs; this would be more like a unicast-only IPLS VPN.

If temporary certificates are used, it has not yet been defined in HIP how a host identity may change for active security associations.

7. Security Considerations

The model considered above assumes that PE devices that hold trusted credentials (certificates and private keys) are trustworthy; a malicious or misconfigured PE device could subvert packet delivery across the overlay.

The model also assumes that the information that PE devices need to obtain to bind the PE name to the overlay and to its respective public key is not compromised, and that the keys of the PE devices are themselves not compromised. A PKI revocation system may aid in dealing with compromised keys.

Otherwise, the system described above inherits the security properties found in HIP, including strong authentication of the binding between host identity and (underlay) IP address, and some level of robustness from denial-of-service attacks on the underlay network, based on the properties of the HIP base exchange.

Section 5.5 of RFC 4665 describes security features from the perspective of the L2VPN solution, while Section 6.5 of RFC 4665 describes the security from a user perspective. The HIPLS solution must protect against the attacks listed in Section 5.5 of RFC 4665.

8. IANA Considerations

There are no IANA considerations.

9. Acknowledgments

Jeff Ahrenholz, Orlie Brewer, Eric Byres, Jin Fang, Darren Lissimore and Jeff Meegan have provided invaluable support in the design and prototype implementation of this HIPLS functionality. Richard Paine and Craig Dupler were instrumental in guiding early work along these lines. Members of other Standards organizations such as The Open Group, the Trusted Computing Group (TCG), and the International Society of Automation (ISA) have been involved in standards development activities that leverage HIP and this HIPLS functionality.

10. References

10.1. Normative References

[RFC4664] Andersson, L. and E. Rosen, "Framework for Layer 2 Virtual Private Networks (L2VPNs)", RFC 4664, September 2006.

[RFC5206] Nikander, P., Henderson, T., Vogt, C., and J. Arkko, "End-Host Mobility and Multihoming with the Host Identity Protocol", RFC 5206, April 2008.

[RFC6253] Heer, T. and S. Varjonen, "Host Identity Protocol Certificates", RFC 6253, May 2011.

[RFC7401] Moskowitz, R., Ed., Heer, T., Jokela, P., and T. Henderson, "Host Identity Protocol Version 2 (HIPv2)", RFC 7401, DOI 10.17487/RFC7401, April 2015, <http://www.rfc-editor.org/info/rfc7401>.

[RFC7402] Jokela, P., Moskowitz, R., and J. Melen, "Using the Encapsulating Security Payload (ESP) Transport Format with the Host Identity Protocol (HIP)", RFC 7402, DOI 10.17487/RFC7402, April 2015, <http://www.rfc-editor.org/info/rfc7402>.

10.2. Informative References

[RFC4176] El Mghazli, Y., Nadeau, T., Boucadair, M., Chan, K., and A. Gonguet, "Framework for Layer 3 Virtual Private Networks (L3VPN) Operations and Management", RFC 4176, October 2005.

[RFC4301] Kent, S. and K. Seo, "Security Architecture for the Internet Protocol", RFC 4301, December 2005.

[RFC4423] Moskowitz, R. and P. Nikander, "Host Identity Protocol (HIP) Architecture", RFC 4423, May 2006.

[RFC4665] Augustyn, W. and Y. Serbest, "Service Requirements for Layer 2 Provider-Provisioned Virtual Private Networks", RFC 4665, September 2006.

[RFC4847] Takeda, T., "Framework and Requirements for Layer 1 Virtual Private Networks", RFC 4847, April 2007.

Authors' Addresses

Thomas R. Henderson
University of Washington
Campus Box 352500
Seattle, WA
USA

Email: tomhend@u.washington.edu

Steven C. Venema
Polyverse Security
550 NW Kirkland Way
Suite 210
Kirkland, WA
USA

Email: steve@polyverse.io

David Mattes
Tempered Networks
3101 Western Avenue
Suite 550
Seattle, WA
USA

Email: d.mattes@temperednetworks.com