

Practical Attacks against MPLS or Carrier Ethernet Networks

Version: 0.9

Date: 24. November 2011

Classification: Public

Author(s): Daniel Mende, Enno Rey, Hendrik Schmidt

Table of Contents

1	INTRODUCTION	4
2	EXECUTIVE SUMMARY	4
3	INVOLVED TECHNOLOGIES AND PROTOCOLS	5
3.1	MPLS & MPLS VPNs	5
3.1.1	Trust Model	6
3.1.2	Security Controls Inherent to Technology	6
3.2	Carrier Ethernet	6
3.2.1	Metro Ethernet	7
3.2.2	EoMPLS/ATOM	9
3.2.3	VPLS	10
3.2.4	L2TPv3	11
3.2.5	Full vs. Partial Transparency	11
3.2.6	Trust Model	11
3.2.7	Security Controls Inherent to Technologies	12
3.3	LDP	12
3.3.1	Trust Model	12
3.3.2	Security Controls Inherent to Technology	12
3.4	BGP	12
3.4.1	Trust Model	12
3.4.2	Security Controls Inherent to Protocol	12
3.5	Additional Notes on the Security Features of Protocols Involved in Datacenter Interconnect Scenarios	13
4	PRACTICAL ATTACKS	14
4.1	Tools.....	14
4.1.1	Loki	14
4.2	Attacking LDP	14
4.2.1	An Example for signaling EoMPLS virtual circuits	15
4.3	Attacking BGP	16
4.3.1	An Example for Injecting IPv4 Routing Information	16
4.3.2	Injection of MP-BGP Route	18
4.3.3	Cracking BGP MD5 Secrets	20
4.4	Attacking MPLS VPNs	22
4.4.1	Example of Bi-Directional MPLS-VPN Traffic Redirection	22
4.5	Security Problems in Carrier Ethernet Networks.....	25
4.5.1	Attacks From Within the (Carrier) Cloud	25
4.5.2	Network Behaviour with Security Impact, Resulting from Unified Layer2 Network	26
4.5.3	Traditional Layer2 Attacks from One Site to Another	28
4.5.4	Misconfigurations on the Carrier Side, leading to Security Breaches of/within Customer Network	28
4.5.5	Misconfigurations on the Customer Side, leading to Breaches	28
4.5.6	Product or Technology Change on Carrier Side may lead to different Level of Transparency	28
4.5.7	Inconsistent Transparency Level amongst "Carrier Ethernet" Product(s) from one Vendor	29

5	CONCLUSIONS	30
5.1	(How) Can an Attacker Get into the Traffic Path?	30
5.1.1	Device Compromise	30
5.1.2	Device Injection	31
5.1.3	Wire Access	31
5.2	Mitigation Approaches	32
7	APPENDIX A: SOME NOTES FROM A PENTEST.....	33
8	REFERENCES	35

1 INTRODUCTION

This paper discusses practical attacks in MPLS and Carrier Ethernet networks. Given the “isolation property” these types of networks seem to dispose of, they are usually regarded as secure and subsequently security controls otherwise common for WAN/data center interlink connections (namely encryption) frequently are not implemented. We will provide an overview which types of attacks are feasible once the *isolation property* is violated and which tools can be used for such attacks, together with an evaluation of the associated risks.

The paper is organized as follows. First we provide an overview of the technologies and protocols involved, including some discussion of their trust models and their inherent security properties. A description of attacks follows, some conclusions will be drawn and finally approaches how to gain confidence in such networks’ security will be outlined.

2 EXECUTIVE SUMMARY

So-called MPLS Layer 3 (“VPN”) connections and their Layer 2 counterparts (“Carrier Ethernet” or “Metro Ethernet”) are often regarded as secure network links because they are assumed to be under the full (security) control of a carrier.

While this assumption of a “trusted core” may be correct in most cases it might happen that scenarios arise where a party that is regarded untrusted from an individual customer’s perspective gains control over a network element. This might be an attacker compromising a router or just managing to get into the traffic path but this might also be another customer (of the respective carrier) who is allowed to administer own network devices being part of the backbone network.

In this paper we discuss practical attacks which become possible once an untrusted party is able to modify the headers of transmitted packets or to take part in the signaling processes of such networks. Given that the involved technologies and protocols do not dispose of any security mechanisms on their own (but just rely on the isolated environment they are supposed to run in) such attacks might lead to severe business risks.

This in turn means that either the trustworthiness of a carrier providing such links must be carefully evaluated or that customers using these technologies must implement appropriate security controls.

3 INVOLVED TECHNOLOGIES AND PROTOCOLS

3.1 MPLS & MPLS VPNS

Multiprotocol Label Switching is a technology specified in RFC 3031 (amongst others), whose main purpose is to forward packets based on so-called labels. Initially it was developed to avoid inefficient traditional IP Routing (routing each packet by means of its destination address and the usage of routing tables). MPLS is forwarding packets by using *labels*. Therefore a 32 bit long item will be added to the packet header. These 32 bits mainly consist of the 20 bit long *tag* (the label, that is the basis for the forwarding decision) and three more fields (e.g a *time-to-live* field). The labels and their “meaning” are negotiated by two neighboring routers by the usage of a protocol (mostly *Label Distribution Protocol*, LDP). Due to the adjustment of neighboring devices a central mechanism for the label management can be avoided.

“MPLS VPNs” is an independent, label-based technology with its own terminology (mainly described in RFC 4364). This technology is often compared to frame relay and ATM, because on a shared network infrastructure separated paths are established, transporting some customers’ VPNs traffic. As for the terminology one must differentiate here between the provider network (P-Network) using MPLS and the customer network[s] (C-Network) accessing a corresponding service but not involved in any labeling. The transition points between customer and provider are called PE-Devices (Provider Edge) on the provider’s side and a CE-Devices (Customer Edge) on the customer side. Usually a PE is serving multiple customers and for this reason not only maintains a “normal” routing table (here also called *global* routing table) but also maintains VPN specific routing tables, so called *Virtual Routing and Forwarding Instances* (VRFs) for each VPN. The PEs connecting the customers are sharing information about which net (prefix) is served by which customer, using *Multiprotocol BGP* (MP-BGP), an extended variant of the *Border Gateway Protocol* (see above). The respective transmitted information is expressing sth like: “Thru me (PE) with some given label it is possible to reach this or that prefix (net) of this or that customer”.

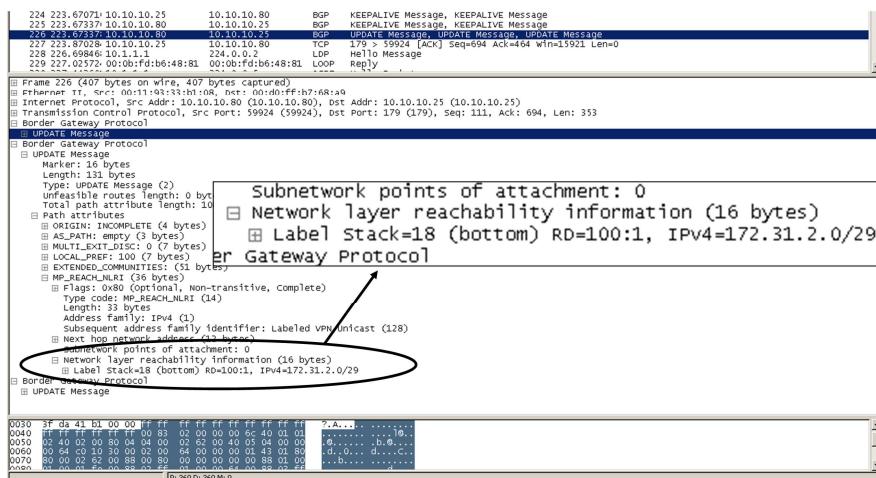


Figure 1: Routing exchange with MP-BGP

The VPN functionality can thus be summarized as follows:

Every prefix of a customer VPN is getting a label assigned by a PE router. An information triple containing *Route Distinguisher*, net prefix and label, is then propagated by the PE to peering PEs, by MP-BGP.

Assuming that no filtering of routing information (using so-called *route targets*) is taking place, every PE knows which prefix/subnet is reachable by a certain customer through individual PE devices and which labels have to be used.

Now, once a PE receives a packet of a customer device, this packet is labeled with at least two labels and forwarded. One label identifies to the path to another PE and the other one is specifying to which customer network the packet belongs to. So, in short, the whole VPN functionality is implemented by the use of labels.

3.1.1 Trust Model

The whole "core" (that is devices participating in label distribution and MP-BGP) is assumed to be trusted. MPLS does not dispose of any security properties on its own.

3.1.2 Security Controls Inherent to Technology

MPLS doesn't provide:

- Protection against mis-configurations in the core
- Protection against attacks from within the core
- Confidentiality, authentication, integrity, anti-replay
-> Use IPsec if required
- Customer network security

Figure 2: What MPLS doesn't provide, [12]

3.2 Carrier Ethernet

Carriers are increasingly offering services that provide end-to-end Ethernet connectivity across worldwide (mostly MPLS based) backbones. These services are often called something like "Carrier Ethernet Services" or "International Ethernet VPN". However enterprises know Ethernet predominantly as a LAN technology where all user data is multiplexed over the network with limited separation or isolation. Furthermore, the consequences of the subsequent possible merger of Layer2 and Layer3 networks might

impose a whole new class of security risks that seem not too well understood, neither in carrier space nor in enterprise environments.

It should be noted that several scenarios must be distinguished when talking about "Ethernet in Carrier Space".

First Ethernet may only be used as a *medium* (as opposed to a *service*) to access the carrier cloud (comparable to E1/T1, E3/T3, ATM, POS lines and the like) and "terminating" this line there's a carrier managed L3 device providing an Ethernet interface that supplies the site's uplink connection (either to the Internet, either to a VPN cloud). Cases of such mere *Ethernet access*¹ are not considered in this document, given there's a carrier supplied CPE that constitutes a L3 boundary between the local network and the carrier's RED (untrusted) network.

If the carrier product is intended to offer *end-to-end Ethernet connectivity* (as a service) and marketed within the "VPN product" space, the security aspects depend highly on the type of CE connecting a site and on the type of device that's sitting next to that CE (i.e. between the CE and the site's local network).

In case the CE is a Layer3 device (a router) – which usually does not happen with "Metro Ethernet" – there's practically no difference to "MPLS Layer3 VPNs" as discussed above. In case the CE is a Layer2 device (a switch) – which should apply to most "Carrier Ethernet Products" in the sense of this document – the device "behind it" (looking from the cloud) plays an essential role for the scenario's security. If this device is a router, this breaks the *end-to-end Ethernet domain* and induces a Layer3 boundary. The resulting scenario can thus be regarded as the "own CE in Layer3 MPLS VPN" case which is discussed above.

If this next-to-CE-device is a Layer2 device (switch) and subsequently "real end-to-end Ethernet"² between the connected sites may be implemented, the security must be handled carefully. Unforeseen protocol behaviour may arise and the overall security may heavily depend on the concrete configuration of the (carrier managed) CE. This scenario is the main focus of the discussion that follows.

3.2.1 Metro Ethernet

"Metro Ethernet" is more of a collective term for several technologies providing Ethernet based access links in metropolitan areas than a well-defined technology in itself (e.g. there is no "Metro Ethernet RFC"). These technologies include MPLS based ones (described below) but historically the most widely implemented variant has been Ethernet over SONET/SDH3. Depending on the specific carrier product, "Metro Ethernet" can provide Point-to-Point connections or even Point-to-Multipoint or Any-to-Any connections. The main "standard body" for Metro Ethernet is the "Metro Ethernet Forum" [MEF, metroethernetforum.org], a global industry alliance comprising more than 120 organizations including telecommunications service providers, cable operators, MSOs, network equipment, test vendors, labs and software manufacturers, semiconductors vendors and testing organizations. The MEF's main purpose is to develop "technical specifications and implementation agreements to promote interoperability and deployment of Carrier Ethernet worldwide." (quoted from MEF website).

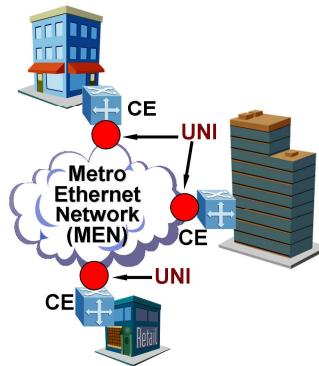
¹ This should apply to most carrier products of the FTTx or xDSL type offering cheap physical Ethernet lines as uplink connection, physically provided by means of some "plastic CPE".

² Please note discussion on "full vs. partial transparency" below.

³ Other technologies used for "Metro Ethernet" are Resilient Packet Rings (RPR, IEEE 802.17) or just "Ethernet Transport" from the access layer to the backbone.

Ethernet Service – Basic Model defined in MEF 1

- Customer Equipment (CE) attaches to UNI
- CE can be
 - router
 - IEEE 802.1Q bridge (switch)
- UNI (User Network Interface)
 - Standard IEEE 802.3 Ethernet PHY and MAC
 - 10Mbps, 100Mbps, 1Gbps or 10Gbps
- Metro Ethernet Network (MEN)
 - May use different transport and service delivery technologies
 - SONET/SDH, WDM, RPR, MAC-in-MAC, Q-in-Q, MPLS



METROEthernet Forum

Metro Ethernet Services Overview v1.5

11

Figure 3: Ethernet Service – Basic Model, from [1]

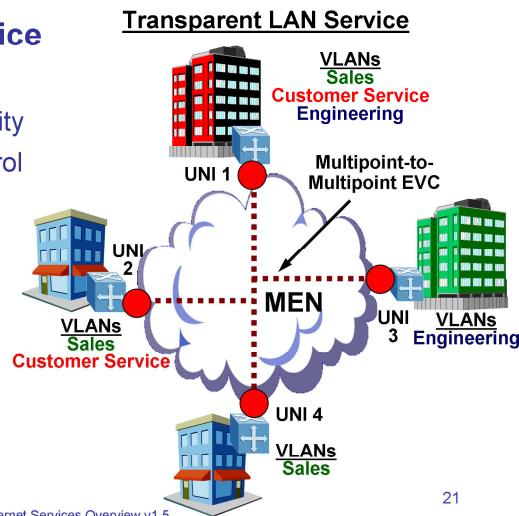
From an enterprise's security perspective "Metro Ethernet" links might always be treated as untrusted networks given the variety of potential technologies involved and the pure Layer2 environment that can be found in many cases. Additionally the following factors should be considered:

- The access link to the Carrier's network might be a Layer 2 device (a switch) that connects several customers (e.g. in business parks). Depending on this device's configuration there might even exist a "shared L2 infrastructure" between some (or all) of the Carrier's customers at this site, with subsequent security problems.
- Usually a "Metro Ethernet" connection provides a fully transparent Ethernet link [see picture below] between the connected sites (in contrast to several MPLS based "Ethernet Services" where this link might not behave fully transparently).
- The MEF has published several "certifiable" specifications defining the details of different Ethernet services. These specifications (in fact a Carrier's compliance with them) may be used to identify the details of an offered service.

Example Service using E-LAN Service Type

- **Transparent LAN Service (TLS) provides**
 - Intra-company Connectivity
 - Full transparency of control protocols (BPDUs)
- **New VLANs added**
 - without coordination with provider

TLS makes the MEN look like a LAN



METROEthernet Forum

Metro Ethernet Services Overview v1.5

21

Figure 4: Example Service using E-LAN Service Type, from [1]

3.2.2 EoMPLS/ATOM

Ethernet-over-MPLS (EoMPLS) is a technology where the MPLS backbone is used not to transport IP packets from one site to another (providing "Layer 3 service") but to transport whole Ethernet frames ("Layer 2 VPN"). The signalling and labeled transport are comparable to Layer3 MPLS VPNs. Only point-to-point connectivity is provided; therefore EoMPLS does not scale very well.

Both terms "EoMPLS" and "ATOM" are mostly used in the Cisco world.

It is described (but not "specified") in the (historic) RFC 4906 Transport of Layer 2 Frames Over MPLS⁴. The following diagram gives an idea of the functionality:

⁴ The most important "Standards Track" RFC for EoMPLS is RFC 4447- Pseudowire Setup and Maintenance Using the Label Distribution Protocol (LDP) which also includes some security discussion.

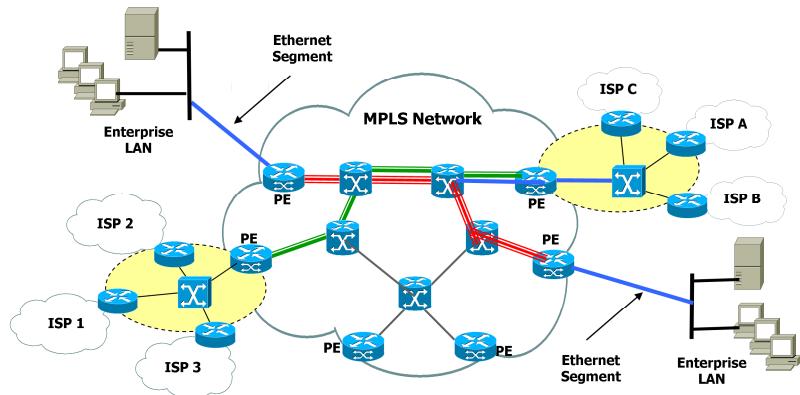


Figure 5: EoMPLS Example Network

3.2.3 VPLS

Virtual Private LAN Service (VPLS) is an MPLS-based service and extends the *pseudowire* concept of EoMPLS further effectively providing point-to-multipoint/any-to-any connectivity. Information which PEs are participating in one 'LAN' is exchanged by some signalling protocol (BGP, LDP, others) and the VPLS cloud is often regarded as a 'big switch' [albeit a 'big trunk' (in Cisco terms) would be more correct as the cloud does not interact with most L2 protocols (which a switch generally does)]. The CE devices are usually switches. It can be expected that most "Carrier Ethernet" services will be VPLS-based in the near future.

A more detailed description can be found in [5]. Furthermore there are two RFCs (4761 and 4762) specifying two different flavors of VPLS (differing mainly as for the signalling protocol). It should be noted that RFC 4762 explicitly mentions "a case, [where] STP Bridge PDUs (BPDUs) are simply tunneled through the provider cloud", thus expecting the PEs to behave "transparently" for (at least) some type(s) of BPDUs. The following diagram gives an idea of the working mode of VPLS:

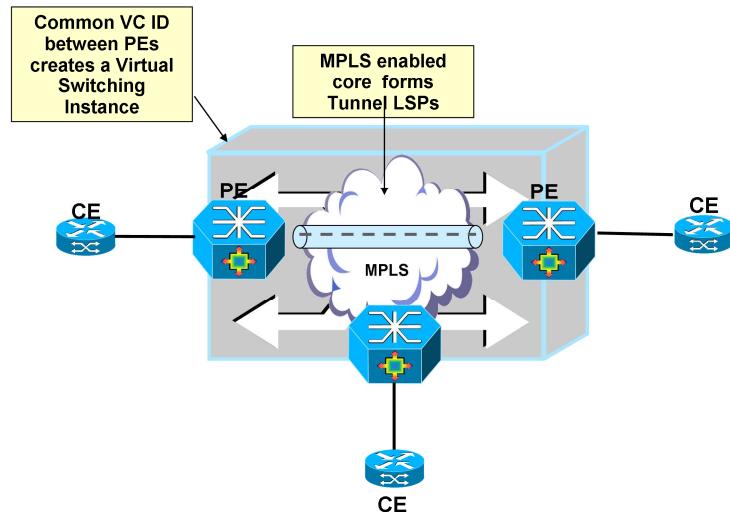


Figure 6: VPLS Working Mode Schema

3.2.4 L2TPv3

The Layer 2 Tunneling Protocol, Version 3 (L2TPv3) can be used as a control protocol and for data encapsulation to set up *Pseudowires* (PWs) for transporting layer 2 Packet Data Units across an IP network. It is specified in (Standards Track) *RFC 4719 Transport of Ethernet Frames over Layer 2 Tunneling Protocol Version 3 (L2TPv3)* and *RFC 3931*.

3.2.5 Full vs. Partial Transparency

Depending on the (carrier's) service/product, potentially the devices used and the configuration of PE and CE the connection may or may not provide full transparency.

"Full transparency" means, that all BPDUs (including e.g. STP, DTP, VTP, GVRP, LACP, 802.1x packets and the like) and all Layer2 Headers (incl. VLAN tags, CoS) are transparently transported from one site to another/others across the cloud⁵.

In contrast "partial transparency" means that some of the BPDUs or header information is filtered/discard when entering the cloud.

From a customer perspective "full transparency" offers some advantages (for instance the ability to implement corporate wide VLANs or QoS policies without interaction with the carrier) but might also induce new security risks resulting mainly from a lack of understanding of the impact on network (management) communication (see below for a more detailed discussion). To implement business reasonable controls it might hence be helpful to figure out in advance if full or partial transparency is/will be in place.

3.2.6 Trust Model

Mostly the same as with MPLS ("Layer 3") VPNs discussed above.

⁵ A User Network Interface Type 1.1 UNI-N as of the Metro Ethernet Specification could for example provide such a fully (or at least mostly) "transparent" service.

3.2.7 Security Controls Inherent to Technologies

These are the same that traditional Ethernet disposes of, that means practically none.

3.3 LDP

The Label Distribution Protocol, initially specified in the RFC 3036, is a signaling protocol for distributing labels for a label switched path in an MPLS network. In 2007 RFC 5036 was released and replaces the old specification. LDP serves a set of procedures and messages by which *Label Switched Routers* (LSRs) establish *Label Switched Paths* through a network by mapping network routing information to data-link layer switched paths. The procedures consist of four kind of functions: discovery functions, session management, advertisement and notification.

3.3.1 Trust Model

LDP uses TCP to establish sessions between two LSRs. UDP is used for basic operations like discovery mechanisms which are periodically sent over the network to a well-known discovery port for all routers of a specific subnet. As these are sent to the "all routers on this subnet" group multicast address, with regard to the discovery process all routers on the local link are regarded trustworthy.

3.3.2 Security Controls Inherent to Technology

To protect the authenticity and integrity of LDP messages, LDP supports the TCP MD5 signature options described in RFC 2385. It has to be activated at the LSRs and may protect the messages by validating the segment by calculating and comparing the MD5 digest. To use the MD5 option a preconfigured password on each LSR is necessary.

3.4 BGP

The *Border Gateway Protocol* (BGP, most important RFC is number 1771 on BGP v4, dating from March 1995) takes care of interconnecting the Internet's participating networks and provides dynamic pathfinding mechanisms by means of exchanging topology information. Devices implementing BGP to route packets on the basis of this routing information and are called BGP routers. BGP speaking routers with a direct relationship are considered as BGP neighbors or peers.

3.4.1 Trust Model

As BGP uses a TCP based communication channel (which inherently does not work via multicast messages, in contrast to many other routing protocols) the BGP peer usually have to be kind-of preconfigured by human operators. This might provide additional trust and security in the first place, still it makes quite some parts of the BGP based Internet infrastructure susceptible to human error (AS 7007 incident in the late 90s or YouTube/Pakistan incident in 2008) or to attacks by operator personnel (see for example Kapela's/Pilosov's presentation at DefCon 2008).

3.4.2 Security Controls Inherent to Protocol

In order to protect the TCP based communication BGP relies on the TCP MD5 Signature Option which has been defined in RFC 2385. This option makes use of the Message Digest 5 (MD5) algorithm. The MD5 Signature Option extends TCP in a way which allows to carry digest messages within TCP segments. To calculate the digest messages, additional information is used which in this case can be regarded as a kind of passphrase.

3.5 Additional Notes on the Security Features of Protocols Involved in Datacenter Interconnect Scenarios

The most common protocols for SAN traffic or replication are NFS, FCoE and iSCSI. While a detailed discussion of their respective functionality and specifications is not relevant for our discussion, it should nonetheless be noted that, again, most of them do not dispose of inherent security properties on their own. A notable exception is NFSv4 which has some mature security mechanisms but the authors are not aware of many organizations using those.

4 PRACTICAL ATTACKS

4.1 Tools

4.1.1 Loki⁶

Initially the tool *Loki* was meant to combine some stand-alone command line tools, like the *bgp_cli*, the *ospf_cli* or the *ldp_cli* and to give them a user friendly, graphical interface. In the meantime *Loki* is more than just the combination of the single tools; it enables its modules to base upon each other (like combining ARP-spoofing from the ARP module with some man-in-the-middle actions, rewriting MPLS-labels for example) and even interoperate with each other.

GUI:

Loki is based upon the GTK library. The base program creates the main window with the general command-buttons and a few sub-windows, like the log-, the preference- or the about-window and a status bar. In the center of the main window, it creates a notebook, with one tab for each module. The tabs are filled with GTK-widgets from the module code. These widgets are fully under control of the module code, so the main program doesn't need to worry about.

Traffic capturing:

For capturing the network data, *libpcap* is used. The main program enumerates all network interfaces and gives the user a graphical interface to select the interface to use. Instead of capturing data live data from an interface, also a capture file can be opened. Once the interface, or input file, is selected, a new thread is created in the main program, which permanently captures the input data and demultiplexes it to the single modules.

Traffic injection:

Traffic injection is done via the *dnet* library. The *Loki* main program creates a *dnet* instance for the selected interface and passes it directly to the modules.

Firewalling:

Firewalling is also done via the *dnet* library. The main program creates a global *dnet* firewall object and passes it to the modules.

4.2 Attacking LDP

Loki contains a universal LDP module, written in python. It implements the most common used LDP packet and data types and can be used to participate in the LDP discovery process, as well as establish targeted LDP sessions for advanced signaling. If such a targeted session is established, the tool starts a background thread which sends keep-alive packages to hold the connection open and the signaled data valid. To create such signaling data e.g. EoMPLS virtual circuits signaling, the module provides build-in data types which can be merged to the appropriated signaling packet.

⁶ See <http://www.insinuator.net/2010/08/try-loki/>

4.2.1 An Example for signaling EoMPLS virtual circuits

The peer is a Cisco 3750ME with a configured, but not activated virtual circuit:

PE2_3750me#show mpls l2transport vc				
Local intf	Local circuit	Dest address	VC ID	Status
Fa1/0/2	Ethernet	192.168.94.128	200	DOWN
PE2_3750me#				

Figure 7: Cisco 3750 Output

Loki is used to establish an LDP session and to send the necessary signaling information:

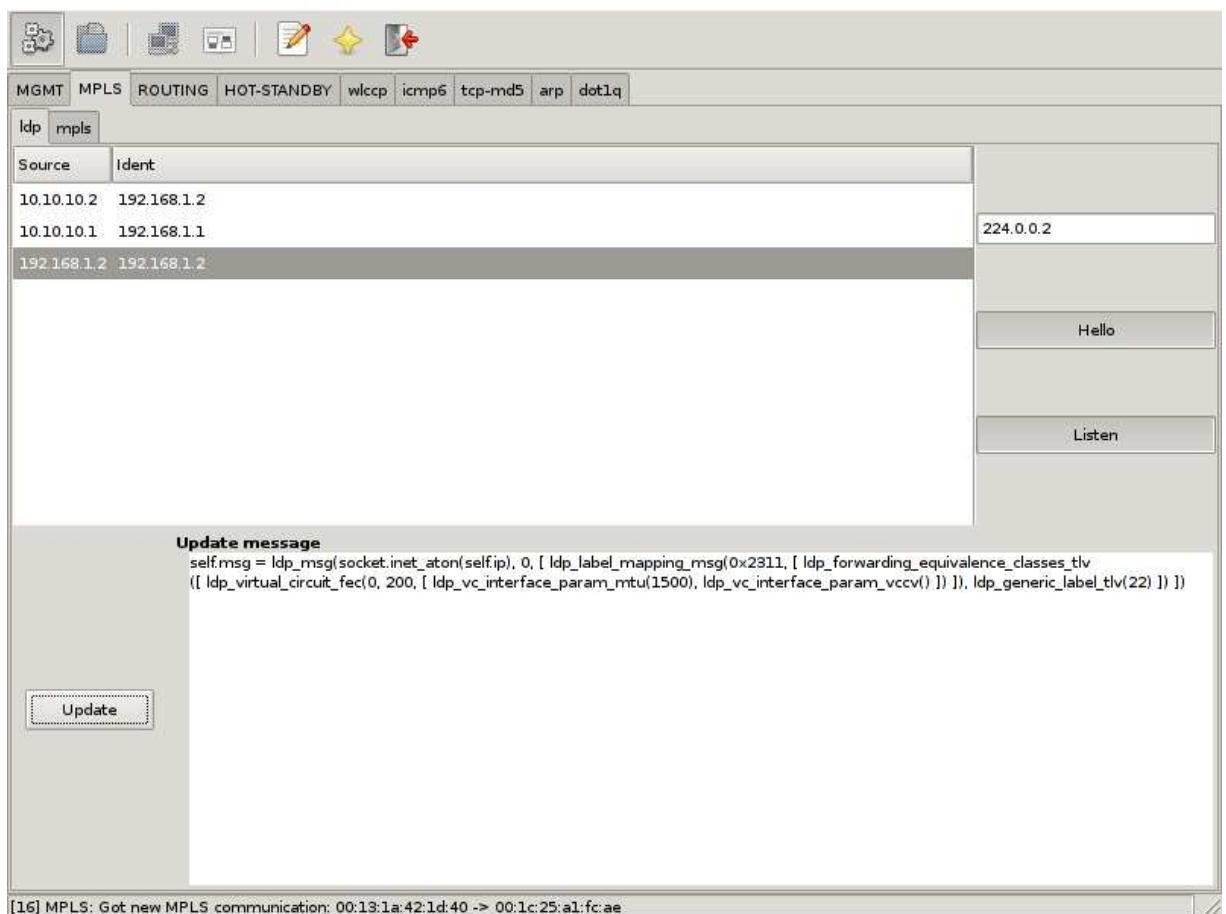


Figure 8: Establishment of LDP Session with *Loki*

First *Loki* needs to take part in the LDP discovery process; this is done by activating the Hello-Thread via clicking on the "Hello" button. Next the remote host tries to connect the attack host via TCP, so *Loki* needs to listen for that incoming connection; this is done by activating the Listen-Thread via clicking on the "Listen" button. Once the Connection is established, the remote Host will show up in the Host-List. The next step is to configure the LDP update message, which defines the signaling message to publish to the remote host. In this case we generate a LDP Label-Mapping-Message. In the end we send the

prepared update message by selecting the designated host from the host list and clicking on the "Update" button.

After send the LDP Label-Mapping-message the configured virtual circuit is activated on the remote side:

Local intf	Local circuit	Dest address	VC ID	Status
Fal/0/2	Ethernet	10.10.10.10	200	UP
PE2_3750me#				

Figure 9: Cisco 3750 Console Output after using Loki

So we activated the virtual circuit and mapped it to a label defined in the update message. A tool like *mplstun* could be used to set up a valid endpoint on the attacker's side.

4.3 Attacking BGP

Loki contains a universal BGP module, written in python. It implements the most common used BGP packet and data types and can be used to establish a connection to a BGP speaking peer. Once a connection is established, the tool starts a background thread which sends keep-alive packages to hold the connection established and the published routes valid. To publish BGP routing information the module provides built-in data types which can be merged to the appropriated update statement. Once an update statement is set up it can be send once or multiple times to the connected peer. It is possible to use kernel based MD5 authentication, as described in RFC2385. Another module makes it possible to brute force the used MD5 authentication key.

4.3.1 An Example for Injecting IPv4 Routing Information

The peer is a Cisco 3750ME with a (pre-attack) routing table looking like this:

PE1_3750me#sh ip route
Codes: C - connected, S - static, R - RIP, M - mobile, B - BGP
D - EIGRP, EX - EIGRP external, O - OSPF, IA - OSPF inter area
N1 - OSPF NSSA external type 1, N2 - OSPF NSSA external type 2
E1 - OSPF external type 1, E2 - OSPF external type 2
i - IS-IS, su - IS-IS summary, L1 - IS-IS level-1, L2 - IS-IS level-2
ia - IS-IS inter area, * - candidate default, U - per-user static route
o - ODR, P - periodic downloaded static route
Gateway of last resort is not set
10.0.0.0/29 is subnetted, 1 subnets
C 10.0.0.0 is directly connected, FastEthernet1/0/11
192.168.1.0/32 is subnetted, 1 subnets
C 192.168.1.1 is directly connected, Loopback0
PE1_3750me#

Figure 10: Cisco 3750 Routing Table

Loki is then used to inject IPv4 routing information:

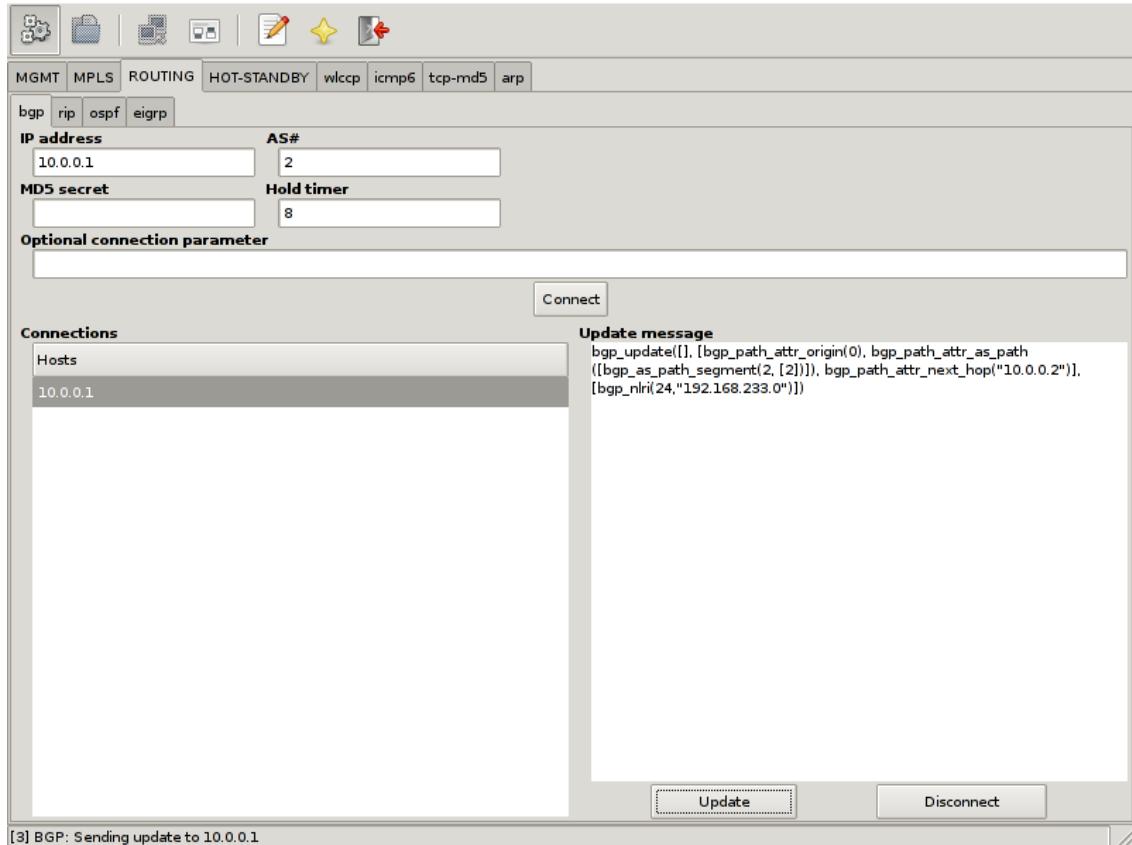


Figure 11: Injecting IPv4 Routing Information with *Loki*

The first step is to configure the target IP address, the autonomous system number 2 and a hold timer of 8 seconds. Afterwards the session can be established by clicking on the "Connect" button. If *Loki* is able to establish the connection, a background keep alive thread is started, which sends an BGP keep alive packet every hold time / 4 seconds. The next step is to configure the BGP update message, which defines, the routing information to publish to the connected host. In the example case we build up a RFC1771 IPv4 routing BGP update packet which says we are announcing the network 192.168.233.0/24 and traffic for this network should be forwarded to the IP address 10.0.0.2 which is our attack host. In the end we send the prepared update packet out by selecting the designated host from the connection list and clicking the "Update" button.

After publishing the routing information, the router's routing table looks like this:

```
00:07:17: %BGP-5-ADJCHANGE: neighbor 10.0.0.2 Up
PE1_3750me#
PE1_3750me#
PE1_3750me#sh ip route
Codes: C - connected, S - static, R - RIP, M - mobile, B - BGP
      D - EIGRP, EX - EIGRP external, O - OSPF, IA - OSPF inter area
      N1 - OSPF NSSA external type 1, N2 - OSPF NSSA external type 2
      E1 - OSPF external type 1, E2 - OSPF external type 2
      i - IS-IS, su - IS-IS summary, L1 - IS-IS level-1, L2 - IS-IS level-2
      ia - IS-IS inter area, * - candidate default, U - per-user static route
      o - ODR, P - periodic downloaded static route

Gateway of last resort is not set

          10.0.0.0/29 is subnetted, 1 subnets
C        10.0.0.0 is directly connected, FastEthernet1/0/11
B        192.168.233.0/24 [20/0] via 10.0.0.2, 00:00:07
          192.168.1.0/32 is subnetted, 1 subnets
C        192.168.1.1 is directly connected, Loopback0
PE1_3750me#
```

Figure 12: Cisco 3750 Routing Table after using Loki

So we injected a route to the network 192.168.233.0/24 which, in this case, directs all matching traffic to our (attack) host.

4.3.2 Injection of MP-BGP Route

The second example shows how to inject MPLS-VPN routing information (as described in RFC4364) into a MPLS Provider Edge router.

The peer again is a Cisco 3750ME with a MPLS-VPN virtual routing and forwarding table associated with the customer 'RED':

```
PE1_3750me#sh ip route vrf RED

Routing Table: RED
Codes: C - connected, S - static, R - RIP, M - mobile, B - BGP
      D - EIGRP, EX - EIGRP external, O - OSPF, IA - OSPF inter area
      N1 - OSPF NSSA external type 1, N2 - OSPF NSSA external type 2
      E1 - OSPF external type 1, E2 - OSPF external type 2
      i - IS-IS, su - IS-IS summary, L1 - IS-IS level-1, L2 - IS-IS level-2
      ia - IS-IS inter area, * - candidate default, U - per-user static route
      o - ODR, P - periodic downloaded static route

Gateway of last resort is not set

B        192.168.113.0/24 [200/0] via 192.168.1.2, 00:46:42
C        192.168.112.0/24 is directly connected, Vlan120
```

Figure 13: Cisco 3750 MP-BGP Routing Information

Loki is then used to inject the MPLS-VPN routing information:

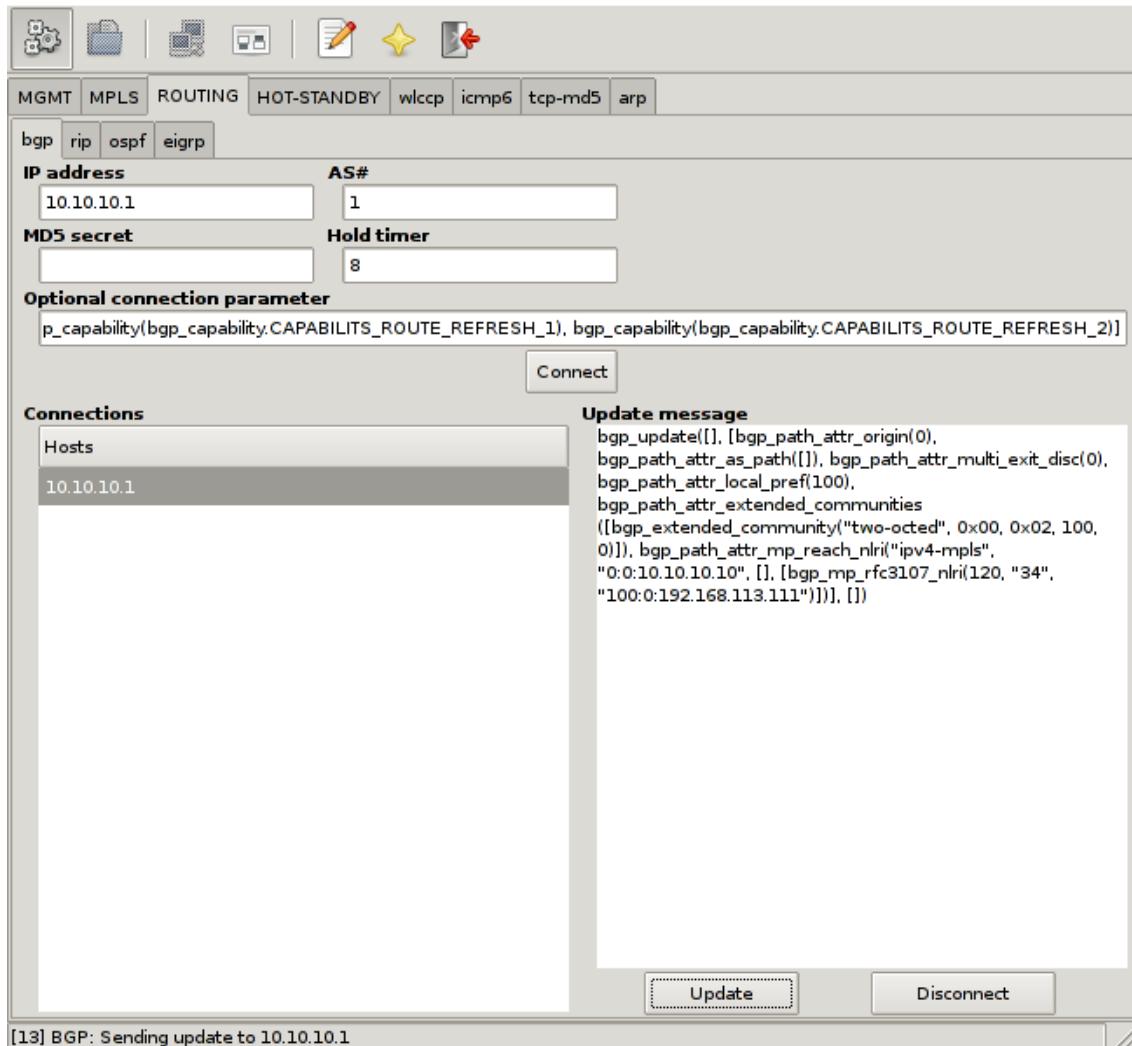


Figure 14: Injecting MPLS-VPN Routing Information with *Loki*

Before setting up the session we need to overwrite the default session parameters with our custom BGP capabilities. This is done by filling in the optional connection parameters. Next the AS number and the hold timer needs to be set. At last the target host is missing, which in this example is the host with the IP address 10.10.10.1. After clicking on "Connect" a session setup is performed. If *loki* is able to establish the connection, a background keep alive thread is started, which sends an BGP keep alive packet every hold time / 4 seconds. The next step is to assigns the BGP update message. This message defines, which routing information to publish to the connected host. In the example case we build up a RFC4364 Multi-Protocol-BGP update packet, which says we are announcing the network 192.168.113.111/32 with the route distinguisher 100:0, which should be forwarded to the next hop 10.10.10.10. In the end we send the prepared update message by clicking on "Update".

After publishing the routing information, the routers virtual routing and forwarding table for the customer 'RED' looks like this:

```
PE1_3750me#sh ip route vrf RED

Routing Table: RED
Codes: C - connected, S - static, R - RIP, M - mobile, B - BGP
      D - EIGRP, EX - EIGRP external, O - OSPF, IA - OSPF inter area
      N1 - OSPF NSSA external type 1, N2 - OSPF NSSA external type 2
      E1 - OSPF external type 1, E2 - OSPF external type 2
      i - IS-IS, su - IS-IS summary, L1 - IS-IS level-1, L2 - IS-IS level-2
      ia - IS-IS inter area, * - candidate default, U - per-user static route
      o - ODR, P - periodic downloaded static route

Gateway of last resort is not set

      192.168.113.0/24 is variably subnetted, 2 subnets, 2 masks
B        192.168.113.0/24 [200/0] via 192.168.1.2, 00:01:30
B        192.168.113.111/32 [200/0] via 10.10.10.10, 00:00:00
C        192.168.112.0/24 is directly connected, Vlan120
```

Figure 15: Cisco 3750 MP-BGP Routing Information after using Loki

One can see the new route for the host 192.168.113.111 pointing to our attack host (10.10.10.10).

4.3.3 Cracking BGP MD5 Secrets

Loki's tcp-md5 module is used for cracking a secret used for RFC2385 based packet signing and authentication. It is designed for offline cracking, means to work on a snuffed, correct signed packet. This packet can either be directly sniffed off the wire or be provided in a pcap file. The cracking can be done in two modes first with a dictionary attack, in this case an additional wordlist is needed, or second without a dictionary in real brute force mode. If the real brute force mode is chosen the tool can enumerate either alphanumeric characters, or the whole printable ASCII space.

4.3.3.1 An example Secret Crack

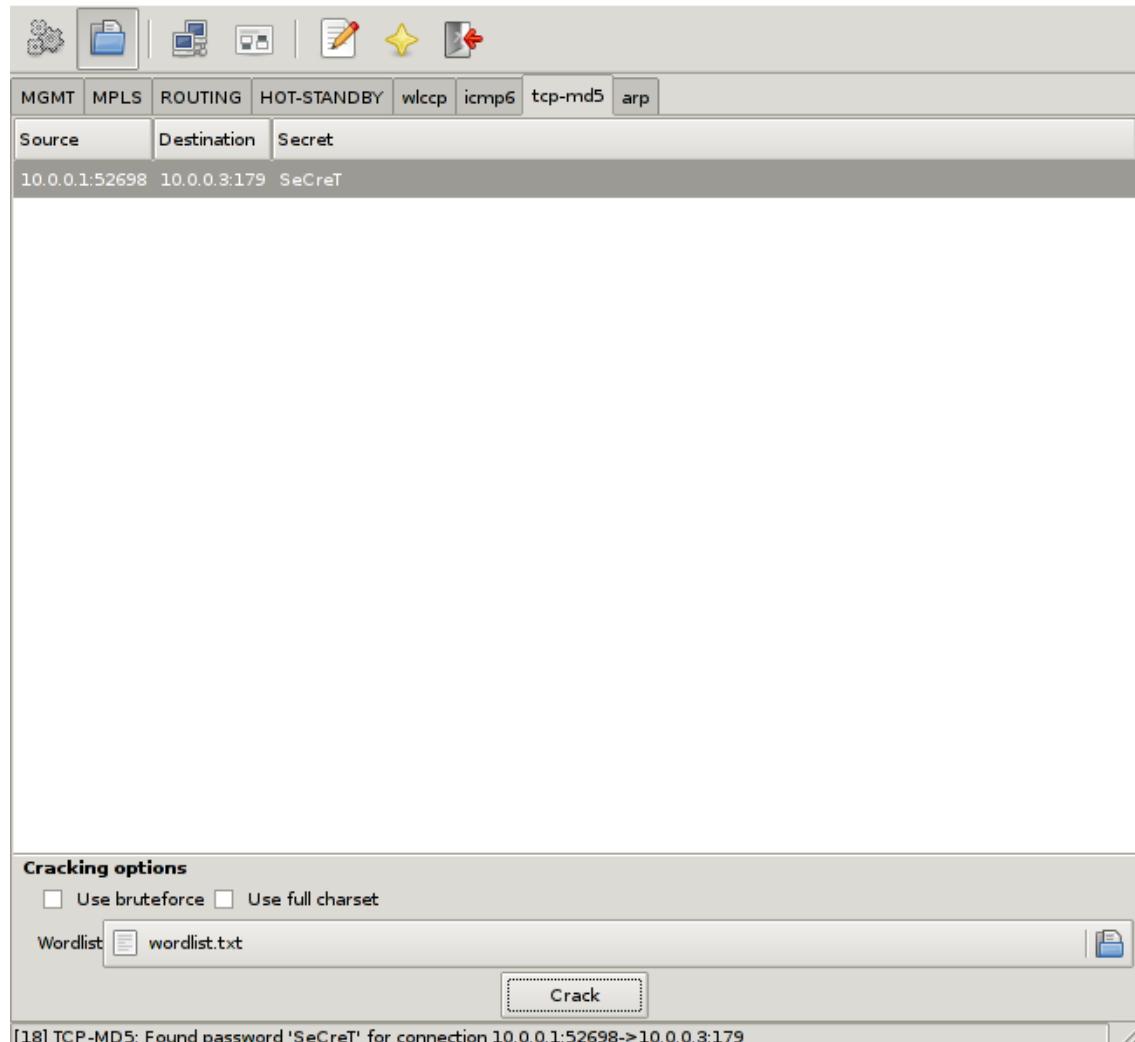


Figure 16: BGP MD5 Cracking Example with Loki

4.4 Attacking MPLS VPNs

Loki's MPLS module is designed to relabel specified MPLS traffic with a given label. It can be used to manipulate the transport label and change the destination of the packet, or to redirect traffic into another MPLS-VPN. The module automatically detects all MPLS labeled traffic on the wire and let the user easily set up relabeling rules. It is possible to add a tcpdump filter to the relabeling rule, if the module should only redirect some special kind of traffic. Last but not least one can define which label in the label stack should be modified.

It should be noted that this attack requires that the attacker has access to the traffic path of the respective packets (see also discussion below).

4.4.1 Example of Bi-Directional MPLS-VPN Traffic Redirection

The setup for this example looks like this:

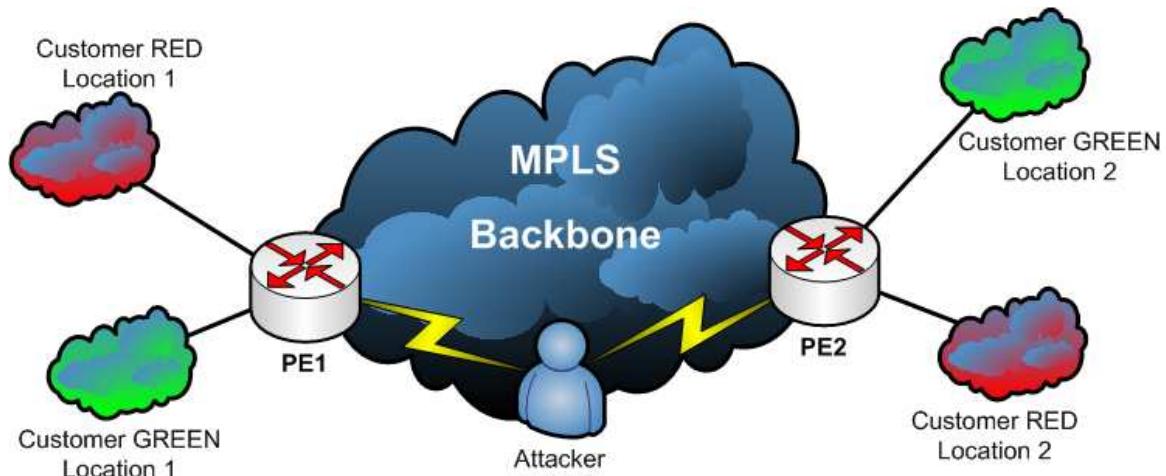


Figure 17: Example Network for a Bi-Directional MPLS-VPN

The attacker is in a Man-in-the-Middle situation inside the data path between Provider Edge 1 and Provider Edge 2 in the MPLS backbone.

On PE1 the label association for the both MPLS-VPNs looks like this:

```
PE1_3750me#sh ip bgp vpnv4 all labels
  Network          Next Hop      In label/Out label
Route Distinguisher: 100:0 (RED)
  192.168.112.0    0.0.0.0      20/nolabel (RED)
  192.168.113.0    192.168.1.2    nolabel/18
Route Distinguisher: 200:0 (GREEN)
  192.168.112.0    0.0.0.0      21/nolabel (GREEN)
  192.168.113.0    192.168.1.2    nolabel/19
```

Figure 18: Cisco 3750 Label Overview

Which means outgoing traffic for customer RED's location 2 is tagged with the MPLS label 18. In the other direction, traffic tagged with MPLS label 20 is sent out to customers RED's location 1. The same for customer GREEN, outgoing traffic for location 2 is tagged with label 19, incoming traffic with label 21 is sent out to location 1. Both customers use the same IP address space for the two locations, which is possible, as we got a logical separation in the routing of each customer.

Let's further assume we got a client with the IP address 192.168.113.100 connected to customer GREEN's location 2. So it's possible to ping this client from PE1 in the context of customer GREEN. We need to specify the virtual routing and forwarding context of customer GREEN to use the customer's specific routing table. If we run the same command in the context of customer RED, no response will be visible:

```
PE1_3750me#ping vrf GREEN 192.168.113.100

Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 192.168.113.100, timeout is 2 seconds:
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 1/2/9 ms
PE1_3750me#ping vrf RED 192.168.113.100

Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 192.168.113.100, timeout is 2 seconds:
.....
Success rate is 0 percent (0/5)
PE1_3750me#
```

Figure 19: Cisco 3750 Test of MPLS-VPN Connection

Next the attacker starts to redirect traffic from PE1 to PE2 in the backbone from customer RED's MPLS-VPN to customer GREEN's MPLS-VPN and redirect traffic from PE2 to PE1 in the backbone from customer GREEN's MPLS-VPN to customer RED's MPLS-VPN by *loki* like this:

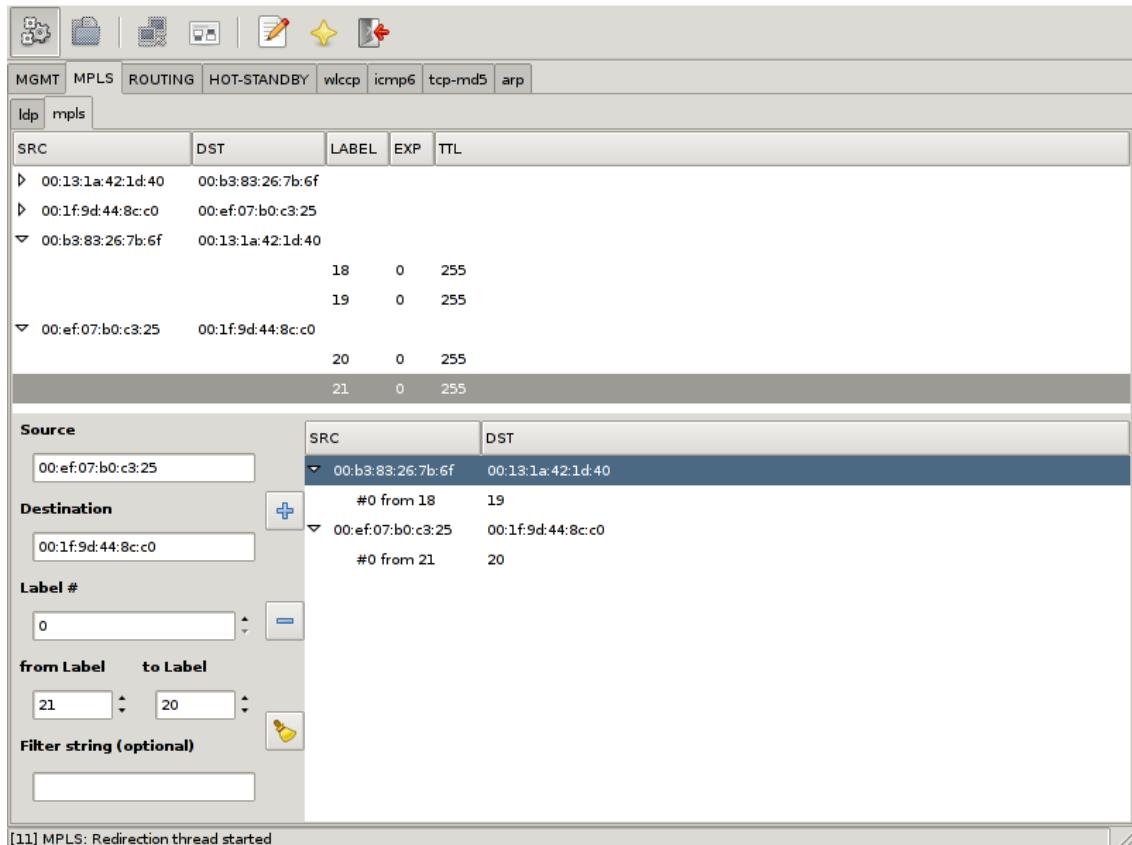


Figure 20: Redirecting MPLS-VPN Traffic with Loki

Once the redirection is in place it is possible to ping our assumed host from both, customer RED's and customer GREEN's context:

```
PE1_3750me#ping vrf GREEN 192.168.113.100
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 192.168.113.100, timeout is 2 seconds:
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 1/2/8 ms
PE1_3750me#ping vrf RED 192.168.113.100
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 192.168.113.100, timeout is 2 seconds:
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 1/4/9 ms
```

Figure 21: Cisco 3750 Test of MPLS-VPN Connection after using Loki

So this actually means that with right position in the traffic path and the right tool (e.g. *Loki*) an attacker can easily redirect a given site's traffic of a given customer to a different destination (provided the IP addresses are the same which presumably is a valid assumption when it comes to addresses like 10.1.1.1 or 192.168.10.1).

4.5 Security Problems in Carrier Ethernet Networks

4.5.1 Attacks From Within the (Carrier) Cloud

Here the same potential security problems as with all MPLS carrier networks (no encryption, PE might be shared with other customers and the like) apply.

4.5.1.1 Attack 1: Relabeling

Loki can be used to relabel 802.1Q tagged packets on the fly. Once an attacker is in the traffic path all seen 802.1Q communications are listed in the dot1q module. To rewrite a label in transmission between two hosts, it simply needs to be selected to fill in most of the fields for the rewrite rule. Only the target label and an optional tcpdump filter to match specific data streams need to be added. Once the rule is added a background thread takes care of the relabeling.

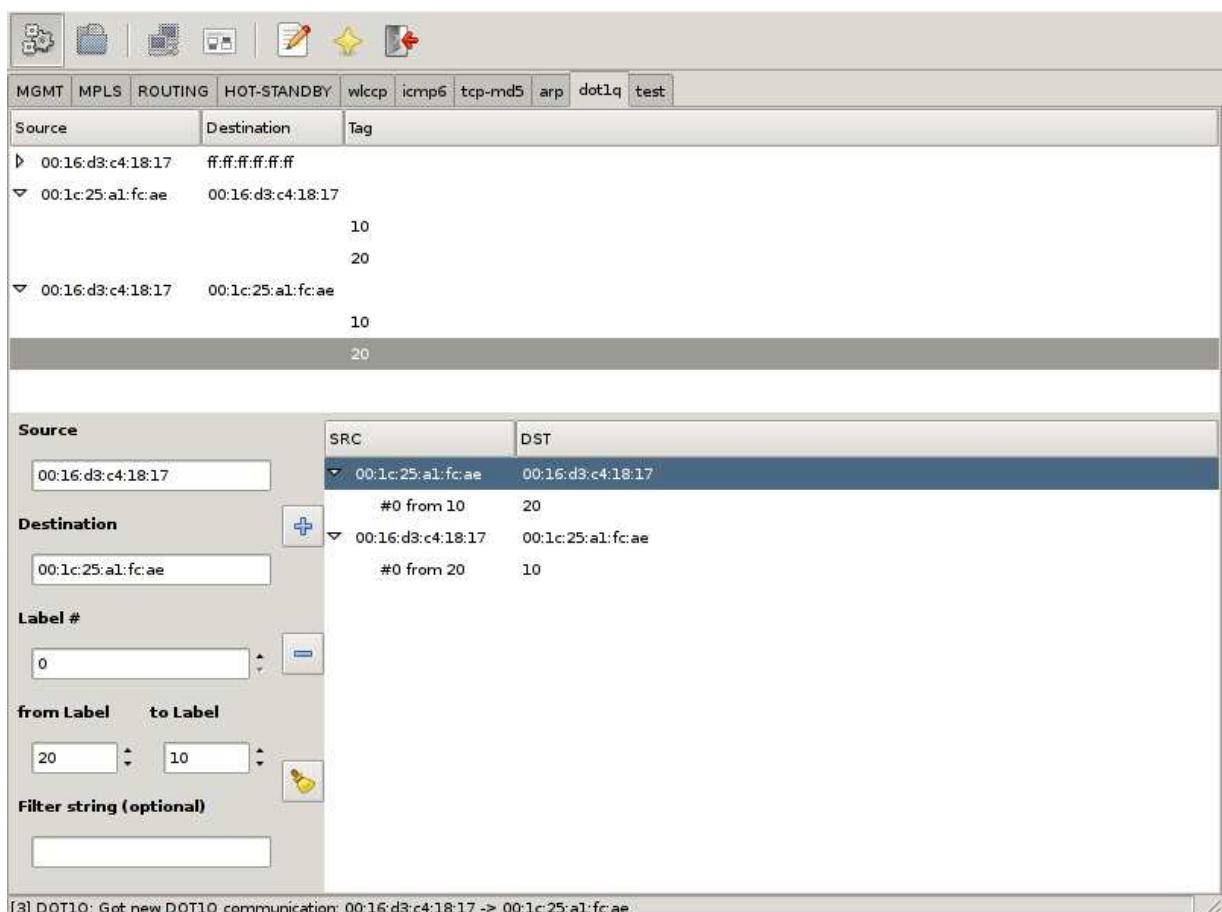


Figure 22: Relabeling with *Loki*

4.5.1.2 Attack 2: Modifying Q-in-Q

The dot1q module in *Loki* can also be used to rewrite the inner 802.1Q label used in Q-in-Q scenarios in the same way as when rewriting the outer 802.1Q label.

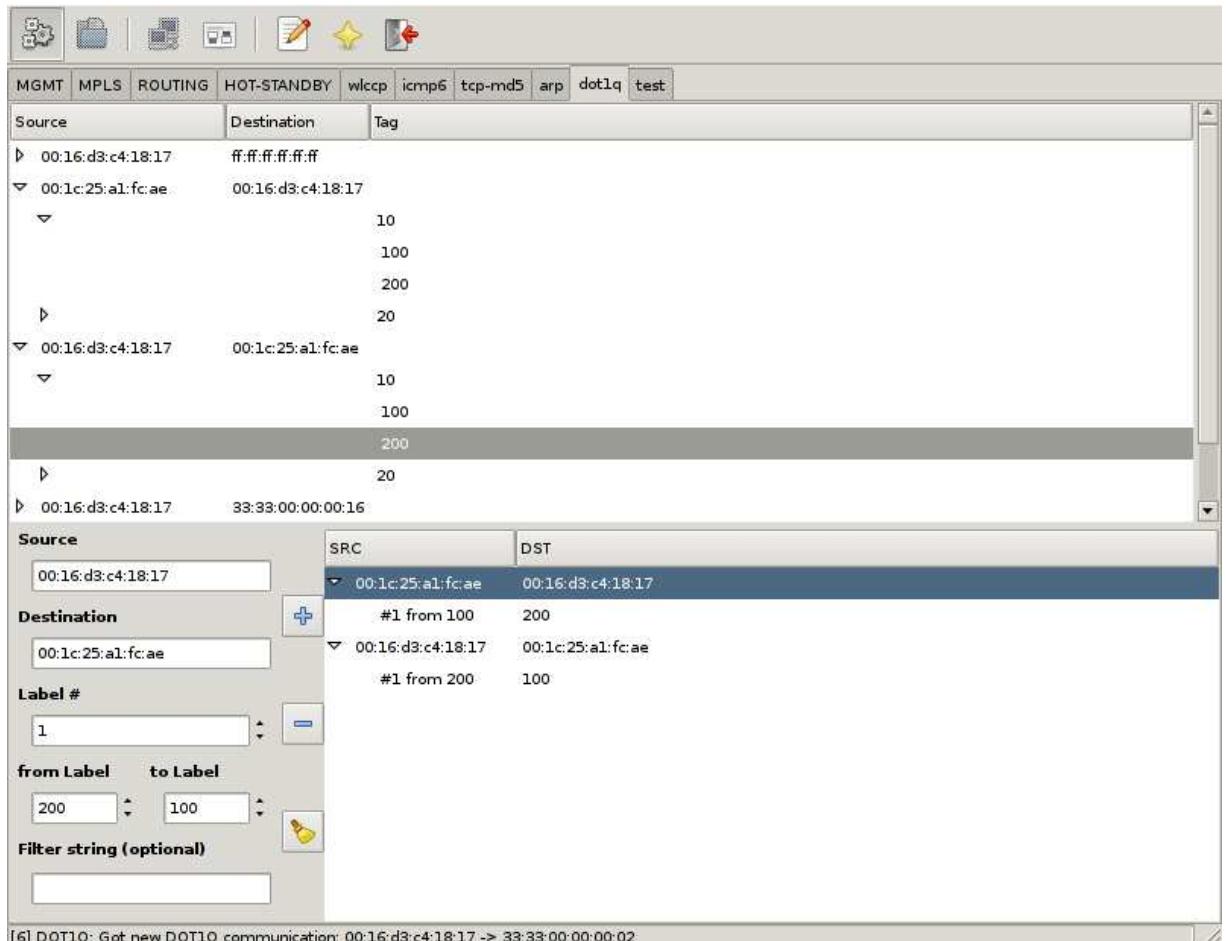


Figure 23: Modifying Q-in-Q with *Loki*

4.5.2 Network Behaviour with Security Impact, Resulting from Unified Layer2 Network

If several sites form a common Layer2 domain after connecting them (mainly in “full transparency” cases), some interesting settings – with potentially huge security impact – can emerge. For example there might only be one *Spanning Tree Root* in the whole (then world wide) L2 network (or one per VLAN). Combined with the fact that some sites may even implement redundant links to the carrier network the following scenario might follow:

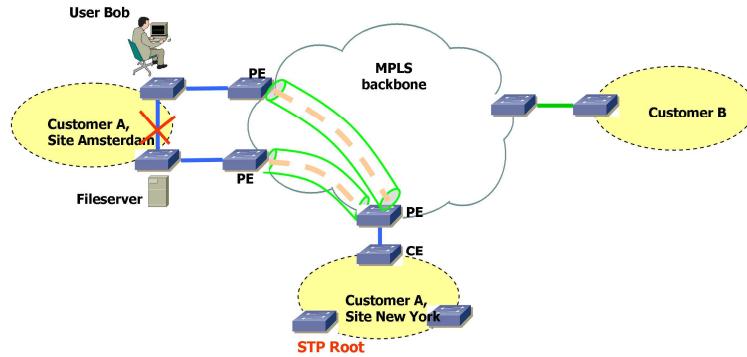


Figure 24: Example Scenario of a Carrier Network

Here the network traffic resulting from Bob's access to the fileserver will actually be forwarded to New York and back to Amsterdam (as the link between the switches in Amsterdam is in *blocking* state), effectively passing the MPLS backbone (possibly unencrypted). Moreover Bob (or the site's or the company's security officer) might be completely unaware of this situation.

Another example of (at the first glance) "unexpected" network behaviour is shown in the following diagram:

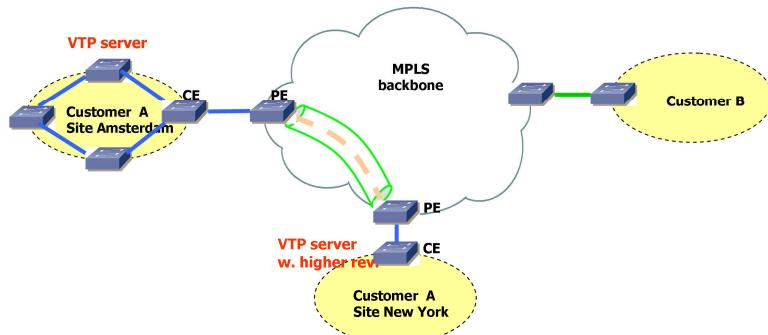


Figure 25: Example Scenario of a Carrier Network 2

With a fully transparent Intra-Site Ethernet connection the switch in New York will propagate its VLAN table to the switches in Amsterdam effectively melting down the complete network over there⁷.

Full transparency with regard to VLANs might impose another risk, shown in the following diagram: "VLAN visibility across the cloud":

⁷ Sure, some conditions must be met for this scenario (e.g. use of the same VTP password in both sites [maybe "cisco"], but again the involved parties might be unaware of such kind of effects when L2 connecting the sites).

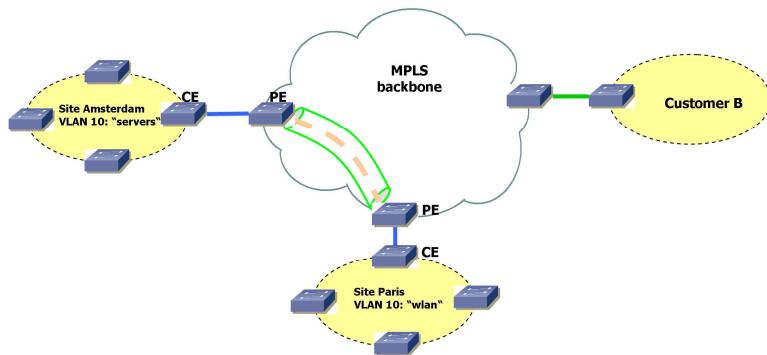


Figure 26: Example Scenario of a Carrier Network 3

Members of VLAN 10 in Paris ("wlan") might be able to communicate with members of VLAN 10 in Amsterdam ("servers")⁸, without notice or awareness of the sysadmins in Amsterdam. This is another example of the effects a fully transparent connection may have.

4.5.3 Traditional Layer2 Attacks from One Site to Another

It should be explicitly noted that – in a such a “unified Layer2 network” – the impact of a system compromise in one site may lead to Layer2 attacks against other sites (e.g. attacks against DTP with subsequent sniffing of remote VLANs with *yersinia* [6]). Previously such attacks mostly probably were not possible.

4.5.4 Misconfigurations on the Carrier Side, leading to Security Breaches of/within Customer Network

If, for instance, the carrier is expected to provide “partial transparency” but actually “full transparency” is implemented (due to operational deficiencies and/or human error), security problems (like those depicted above) may arise.

Another example (which in fact happens) is the accidental connection of sites belonging to different customers or leakage of routing information due to typos in the VRF/VFI identifiers.

4.5.5 Misconfigurations on the Customer Side, leading to Breaches

In “full transparency” scenarios diligent configuration of the customer’s network devices might be necessary to avoid security problems as discussed above. Bad operational practice or human errors may easily lead to severe problems here.

4.5.6 Product or Technology Change on Carrier Side may lead to different Level of Transparency

If the customer is unaware of the exact behaviour of the carrier’s Ethernet service at one point and “just doesn’t notice any problems”, a technology change (be a change of device firmware to a newer version, be a change of an infrastructure protocol’s configuration) may lead to security exposure. A well known historical example was the (mostly unannounced) introduction of a proprietary OSPF enhancement called

⁸ Even if the IP address ranges are different all (Windows-) broadcasts will be transported across the cloud inducing visibility of system names and IP address ranges.

Link Local Signaling in Cisco's IOS which effectively broke OSPF sessions with (customer) *Nokia* devices after (carrier) IOS upgrades some years ago.

4.5.7 Inconsistent Transparency Level amongst "Carrier Ethernet" Product(s) from one Vendor

Carriers offering a nation- or even world wide Ethernet service may technologically implement the product in different ways, depending on the distance between sites ("Metro Ethernet" in case of regional offices, VPLS if far distance between sites). The different technologies may behave differently then as for the level of transparency.

5 CONCLUSIONS

In this section conclusions will be discussed and some additional discussion on the feasibility of the attacks described above will be provided.

5.1 (How) Can an Attacker Get into the Traffic Path?

There are three main possibilities how an attacker (or "untrusted party") can get into a position enabling the performance of the attacks described above.

5.1.1 Device Compromise

Obviously this is the first (and probably most likely) possibility that comes to mind. The *North American Network Operators' Group* (NANOG) periodically collects data on network security incidents amongst its members. The following slide from [13] shows that devices from carrier environments actually get compromised in the real world:

ISP Security BoF – NANOG 28 Statistics as of 01 June 2003

- Hacked hosts – 423262
- Abused proxies – 192608
- Compromised routers – 5410

- Q: How hard is it to obtain a compromised device?
- A: Can you type any of the following?
 - !cisco
 - !cayman
 - !proxy

Figure 27: ISP Security BoF – NANOG 28 Statistics as of 01 June 2003, [13]

5.1.2 Device Injection

The term "device injection" designates all scenarios where an untrusted party is enabled to place a device under its own control in the MPLS network of a carrier. While this may seem highly unlikely for an attacker (to "insert" an own device in a datacentre with strong physical access controls) it should be noted that some carriers allow very large customers to run their own PE routers (thereby potentially violating the assumption of a "trusted core which is solely managed by the carrier")⁹. Similar scenarios might arise when PEs are located on customer premises which is why this practice is commonly advised against, see for example the following slide from a *Cisco Live* conference in 2010:

Key: PE Security

- What happens if a single PE in the core gets compromised?

Intruder has access to all VPNs; GRE tunnel to "his" CE in the Internet, bring that CE into any VPN

That VPN might not even notice...

Worst Case!!!!

- Therefore: PE Security is Paramount!!!!!!

- Therefore: No PE on customer premises!!!!!!

(Think about console access, password recovery...)

Figure 28: PE Security, Cisco Live 2010

5.1.3 Wire Access

By this term all those scenarios are designated where an attacker gains access to the traffic path of certain packets without necessarily having compromised a device. This includes physical access to the wire as well as traffic redirection attacks in shared network segments.

⁹ The authors of the present paper have been involved in projects where this was possible in the past. The names of the carriers will not be disclosed, for NDA reasons.

5.2 Mitigation Approaches

To reach a certain level of confidence¹⁰ when it comes to using these types of links, two main approaches can be undertaken by an organization:

- *Trust* the carrier, potentially after evaluating the overall trustworthiness and operational maturity of the carrier¹¹.
- *Control* the security properties of (potentially only certain) links by implementing appropriate measures, namely encryption.

¹⁰ See <http://www.insinuator.net/2011/06/broken-trust-part-1-definitions-fundamentals-some-more-reflections-on-rsa/> for a detailed discussion of this term.

¹¹ See http://www.ernw.de/content/e7/e181/e392/download775/ernw_are_they_secure_ger.pdf for an example how such an evaluation could look like.

7

APPENDIX A: SOME NOTES FROM A PENTEST

Given the authors have been performing various projects in carrier space we tend to be a bit skeptical as for the assumption of a "trusted core" that is inherent to some of the technologies discussed in this paper. Suffice to say that security of these networks highly depends on operational practice (even more than in typical corporate network environments) and that, well, it *may* happen human errors occur and lead to security breaches. Please note that we do not state that carrier networks are per se insecure. The reader should just not totally exclude the possibility of security incidents in this space...

To give the reader an idea about what can go wrong some notes from a private communication on a pentest in a Tier-1 carrier network in some part of the world follow:

```
> I got LAN access via a wireless access point that was only doing MAC  
> filtering... and it's easy to tailgate through physical access I tested  
that...once you plug in it's DHCP all the way...  
>  
> I took the Solaris NMS box with an old sadmind vulnerability...so a  
> quick MetaSploit later and I had a root shell...unfortunately this  
> box was only for monitoring not for configuration...didn't get a  
> whole lot out of this...and the shadow file hashes are still  
> cracking :(  
>  
> I took the admin jump box via a combination of issues - through a  
> web app running on the box I got limited command execution as  
> nobody...but I managed to see the /etc/password and /tftpboot and  
> /tmp..which led me to the RANCID box...  
>  
> I also managed to get a shell on this box through a weak password  
> for one of the users from the passwd file - this is the only host  
> which can access the core devices through the ACL's so this was  
> important...  
>  
> I found a file in /tftpboot that an admin had written with SNMP  
> communities in it...so that was another option...  
>  
> I took the RANCID box with password reuse for the account I had on  
> the jump box and pulled router configs off there - in the configs I  
> found and decrypted the Cisco 7 vty password...then trying this same  
> password as the enable password gave me luck...and I had enable on  
> one of the PEs. From then on it was clean going for the rest...  
>
```

> What is interesting though is that generally things were tight...MD5
> for protocol exchanges...even protected LDP exchanges...I could get
> nothing from an Internet perspective or a CE perspective.
>
> Funny also, the admin jump host mostly enforces SSH login via authorized
keys...but the account I took was one that had been created
> and not yet used / configured...and they allow password-based SSH
> for emergency maintenance...

8

REFERENCES

- [1] Metro Ethernet Services Definitions Phase 2
http://metroethernetforum.org/PDF_Documents/technical-specifications/MEF6-1.pdf
- [2] http://metroethernetforum.org/PDF_Documents/technical-specifications/MEF13.pdf
- [3] http://www.cisco.com/warp/public/cc/pd/si/casi/ca6000/prodlit/vlnwp_wp.pdf
- [4] <http://tools.ietf.org/html/draft-eastlake-trill-802-protocols-00>
- [5] Cisco VPLS Whitepaper:
http://www.cisco.com/en/US/tech/tk436/tk891/technologies_white_paper09186a00801f6084.shtml
- [6] Tool *Yersinia*: <http://www.yersinia.net/>
- [7] Cisco SAFE Blueprint Layer 2 Security in Depth:
http://www.cisco.com/warp/public/cc/so/cuso/epso/sqfr/sfblu_wp.pdf
- [8] NSA Guide Switch Security:
http://www.nsa.gov/snac/os/switch-guide-version1_01.pdf
- [9] Cisco Packet Magazine, Artikel „Layer 2 -- The Weakest Link“:
http://www.cisco.com/en/US/about/ac123/ac114/ac173/ac222/about_cisco_packet_feature09186a0080142deb.html
- [10] Catalyst Secure Template (for legacy access switches like 29xx/35xx-XL series):
<http://www.cymru.com/gillsr/documents/catalyst-secure-template.htm>
- [11] Cisco Security Advisories:
http://www.cisco.com/en/US/products/products_security_advisories_listing.html.
- [12] Apricot 2006, MPLS Tutorial Day, 27th Feb 2006, MPLS Security
http://www.apricot.net/apricot2006/slides/tutorial/monday/MPLS_Tutorial.zip
- [13] <http://www.nanog.org/meetings/nanog28/presentations/thomas.pdf>