MTU in RouterOS

Introduction

It is the sole responsibility of administrators to configure the Maximum Transmission Unit (MTU) such that intended services and applications can be successfully implemented in the network. In other words - administrators must make sure that MTUs are configured in a way that packet sizes do not exceed the capabilities of network equipment.

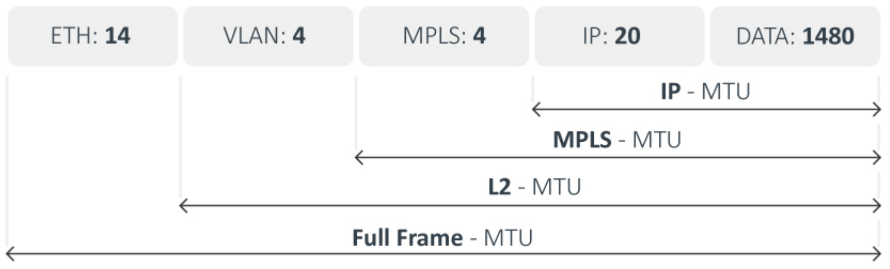
Originally MTU was introduced because of the high error rates and low speed of communications. Fragmentation of the data stream gives the ability to correct corruption errors only by resending corrupted fragments, not the whole stream. Also on low-speed connections such as modems, it can take too much time to send a big fragment, so in this case, communication is possible only with smaller fragments.

But in the present day we have much lower error rates and higher speed of communication, this opens a possibility to increase the value of MTU. By increasing the value of MTU we will result in less protocol overhead and reduce CPU utilization mostly due to interrupt reduction. This way some non-standard frames started to emerge:

- **Giant** or **Jumbo** frames - frames that are bigger than standard (IEEE) Ethernet MTU;
- **Baby Giant** or **Baby Jumbo** frames - frames that are just slightly bigger than standard (IEEE) Ethernet MTU;

It is common now for Ethernet interfaces to support physical MTU above standard, but this can not be taken for granted. Abilities of other network equipment must be taken into account as well - for example, if 2 routers with Ethernet interfaces supporting physical MTU 1526 are connected through an Ethernet switch, in order to successfully implement some application that will produce these big Ethernet frames, the switch must also support forwarding such frames.

Maximum Transmission Unit



Mikrotik RouterOS recognizes several types of MTU:

- IP/Layer-3/L3 MTU
- MPLS/Layer-2.5/L2.5 MTU
- MAC/Layer-2/L2 MTU
- Full frame MTU

Full frame MTU

Full frame MTU indicates the actual size of the frame that is sent by a particular interface. Frame Checksum is not included as it is removed by an ethernet driver as soon as it reaches its destination.

MAC/Layer-2/L2 MTU

L2MTU indicates the maximum size of the frame without the MAC header that can be sent by this interface.

In RouterOS L2MTU values can be seen in the "/interface" menu. L2MTU support is added for all Routerboard related Ethernet interfaces, VLANs, Bridge, VPLS, and wireless interfaces. Some of them support the configuration of the L2MTU value. All other Ethernet interfaces might indicate L2MTU only if the chipset is the same as Routerboard Ethernets.

This will allow users to check if the desired setup is possible. Users will be able to utilize additional bytes for VLAN and MPLS tags, or simply increase interface MTU to get rid of some unnecessary fragmentation.

This table shows *max-l2mtu* supported by Mikrotik RouterBoards (available in the "/interface print" menu as the value of the read-only "max-l2mtu" option):

Model name	MTU description
RB SXT series, RB LHG, RB LDF, PL6411-2nD, PL7411-2nD, RB711 series, wAP R-2nD, RB912R-2nD-LTm (LtAP mini), RB Metal series, RB SXT Lite series, RB Groove series, Cube Lite60, LHG Lite60	ether1:2028
RB SXT G series, RB DynaDish, wAP ac, RB QRT series, RB711G series, RB911G, RB912UAG	ether1:4076
RB OmniTik series, RB750, RB750UP, RB751U-2HnD, RB951-2n	ether1:4076; ether2-ether5:2028
RB OmniTik ac series, RB750GL, RB750Gr2	ether1-ether5:4074
RB mAP, RB mAP lite, RB cAP, RB wAP	ether1-ether2:2028
RB750r2, RB750P-PBr2, RB750UPr2, RB941-2nD, RB951Ui/RB952Ui series	ether1-ether5:2028
RB750Gr3	ether1-ether5:2026
RB751G-2HnD, RB951G-2HnD	ether1-ether5:4074
RB962UiGS, RB960PGS	ether1-ether5:4074; sfp1:4076
RB LHGG series	ether1:9214
LHG XL 52 ac	ether1:9214; sfp1:9214
RB1100Hx2, RB1100AHx2	ether1-ether10:9498; ether11:9500; ether12-ether13:9116
RB4011iGS+ series	ether1-ether10:9578; sfp-sfpplus1:9982
CCR1009 series	ether1-ether4:10224; ether5-ether8:10226; sfp1:10226; sfp-sfpplus1:10226
CCR1016 series	ether1-ether12:10226; sfp1-sfp12:10226; sfp-sfpplus1:10226
CCR1036 series	ether1-ether12:10226; sfp1-sfp4:10226; sfp-sfpplus1-sfp-sfpplus2:10226
CCR1072 series	ether1:9116; sfp-sfpplus1-sfp-sfpplus8:10226
CCR2004-1G-12S+2XS	ether1:9586; sfp-sfpplus1-sfp-sfpplus12:9578; sfp28-1 - sfp28-2:9578
CCR2004-16G-2S+	ether1-ether16:9582; sfp-sfpplus1-sfp-sfpplus2:9586
CCR2116-12G-4S+	ether1-ether12:9570; ether13:9586; sfp-sfpplus1-sfp-sfpplus4:9570
CCR2216-1G-12XS-2XQ	ether1:9586; sfp28-1 - sfp28-12:9570; qsfp28-1-1 - qsfp28-2-4:9570
CRS109-8G-1S	ether1-ether8:4064; sfp1:4064
CRS125-24G-1S	ether1-ether24:4064; sfp1:4064

CRS112-8G-4S, CRS112-8P-4S	ether1-ether8:9204; sfp9-sfp12:9204
CRS106-1C-5S	sfp1-sfp5:9204; combo1:9204
CRS210-8G-2S+	ether1-ether8:9204; sfp-sfpplus1:9204; sfpplus2:9204
CRS212-1G-10S-1S+	ether1:9204; sfp1-sfp10:9204; sfpplus1:9204
CRS226-24G-2S+	ether1-ether24:9204; sfp-sfpplus1:9204; sfpplus2:9204
CRS326-24G-2S+, CSS326-24G-2S+	ether1-ether24:10218; sfp-sfpplus1:10218; sfpplus2:10218
CRS317-1G-16S+	ether1:10218; sfp-sfpplus1-sfp-sfpplus16:10218
CRS328-24P-4S+	ether1-ether24:10218; sfp-sfpplus1-sfp-sfpplus4:10218
CRS328-4C-20S-4S+	combo1-combo4:10218; sfp1-sfp20:10218; sfp-sfpplus1-sfp-sfpplus4:10218
CRS305-1G-4S+	ether1:10218; sfp-sfpplus1-sfp-sfpplus4:10218
CRS309-1G-8S+	ether1:10218; sfp-sfpplus1-sfp-sfpplus8:10218
netFiber 9/IN (CRS310-1G-5S-4S+)	sfp1-sfp5:10218; sfp-sfpplus1-sfp-sfpplus4:10218
CRS310-8G+2S+IN	ether1-ether8:10218; sfp-sfpplus1-sfp-sfpplus2:10218
CRS312-4C+8XG	combo1-combo4:10218; ether1-ether8:10218; ether9:2028
netPower 15FR (CRS318-1Fi-15Fr-2S)	ether1-ether16:10218; sfp1-sfp2:10218
netPower 16P (CRS318-16P-2S+)	ether1-ether16:10218; sfp-sfpplus1-sfp-sfpplus2:10218
CRS326-4C+20G+2Q+	combo1-combo4:10218; ether1-ether20:10218; qsfpplus1-1-qsfpplus2-4:10218; ether21:2028
CRS326-24S+2Q+	sfp-sfpplus1-sfp-sfpplus24:10218; qsfpplus1-1-qsfpplus2-4:10218; ether1:2028
CRS354-48G-4S+2Q+, CRS354-48P-4S+2Q+	sfp-sfpplus1-sfp-sfpplus4:10218; qsfpplus1-1-qsfpplus2-4:10218; ether1-ether48:10218; ether49:2028
CRS504-4XQ-IN	ether1:2028; qsfp28-1-1 - qsfp28-4-4:10218
CRS510-8XS-2XQ-IN	ether1:2028; sfp28-1 - sfp28-8:10218; qsfp28-1-1 - qsfp28-2-4:10218
CRS518-16XS-2XQ	ether1:2028; sfp28-1 - sfp28-16:10218; qsfp28-1-1 - qsfp28-2-4:10218
CSS610-8G-2S+, CSS610-8P-2S+	ether1-ether8:10218; sfp-sfpplus1-sfp-sfpplus2:10218
D52G-5HacD2HnD (hAP ac^{2})	ether1-ether5:9124
C52iG-5HaxD2HaxD (hAP ax^{2})	ether1-ether5:9214
C53UiG+5HPaxD2HPaxD (hAP ax^{3})	ether1-ether5:9214
L41G-2axD (hAP ax lite)	ether1-ether4:2026
cAP ac	ether1-ether2:9124

GPEN21	ether1-ether2:10222; sfp1: 10222
wAP60G, LHG60G	ether1:9124
RB260GS series, CSS106-5G-1S, CSS106-1G-4P-1S	ether1-ether5:9198; sfp1:9198
RBFTC11	ether1:4046; sfp1:4046
RBM33G	ether1-ether3:2026
RBM11G	ether1:2026
RB760iGS	ether1-ether5:2026; sfp1:2026
RB411 series	ether1:1526
RB433 series, RB450, RB493 series	ether1:1526; ether2-ether3:1522
RB450Gx4	ether1-ether5:9214
RB411GL	ether1:1520
RB433GL, RB435G , RB450G, RB493G	ether1-ether3:1520
RB800	ether1-ether2:9500; ether3:9116
RB850Gx2	ether1-ether5:1580
RB921UAGS, RB922UAGS	ether1:4076; sfp1:4076
D23UGS-5HPacD2HnD (NetMetal ac²)	ether1:9214 ; sfp1:9214
RB953GS	ether1-ether2:4074; sfp1:4074; sfp2:4076
RB2011 series	ether1-ether5:4074; ether6-ether10:2028; sfp1: 4074
RB3011 series	ether1-ether5:8156; ether6-ether10:8156; sfp1: 8158
RB5009 series	ether1-ether8: 9796; sfp-sfpplus1: 9796
L009 series	ether1: 8158; ether2-ether8: 8154; sfp1: 8154
RB44Ge	ether1-ether4:9116

All wireless interfaces in RouterOS (including Nstreme2) support 2290 byte L2MTU.

L2MTU configuration changes evoke all interface reloads (link down/link up) due to necessary internal processes.
It is recommended to configure L2MTU with caution by keeping in mind that it can cause short interruption with connected devices.

MPLS/Layer-2.5/L2.5 MTU

Configured in the "/mpls interface" menu, specifies the maximal size of the packet, including MPLS labels, that is allowed to send out by the particular interface.

Make sure that MPLS MTU is smaller or equal to L2MTU. MPLS MTU affects packets depending on what action the MPLS router is performing. It is strongly recommended that MPLS MTU is configured to the same value on all routers forming the MPLS cloud because of the effects MPLS MTU has on MPLS switched packets. This requirement means that all interfaces participating in the MPLS cloud must be configured to the smallest MPLS MTU values among participating interfaces, therefore care must be taken to properly select the hardware to be used.

You can read more about MPLS MTU here.

MPLS Switching

If the packet with labels included is bigger than MPLS MTU, MPLS tries to guess the protocol that is carried inside the MPLS frame:

- If this is an IP packet, MPLS produces an ICMP Need Fragment error. This behavior mimics IP protocol behavior. Note that this ICMP error is not routed back to the originator of a packet but is switched towards the end of LSP so that the egress router can route it back.
- If this is not an IP packet, MPLS simply drops it, because it does not know how to interpret the contents of the packet. This feature is very important in situations where MPLS applications such as VPLS are used (where frames that are MPLS tagged are not IP packets, but e.g. encapsulated Ethernet frames as in the case of VPLS) - if somewhere along the LSP MPLS MTU will be less than packet size prepared by ingress router, frames will simply get dropped.

IP ingress

When a router first introduces a label (or labels) on an IP packet, and the resulting packet size including MPLS labels exceeds MPLS MTU, the router behaves as if interface MTU was exceeded - either fragment packet in fragments that do not exceed MPLS MTU when labels are attached (if IP Don't Fragment is not set) or generate ICMP Need Fragmentation error that is sent back to the originator.

VPLS ingress

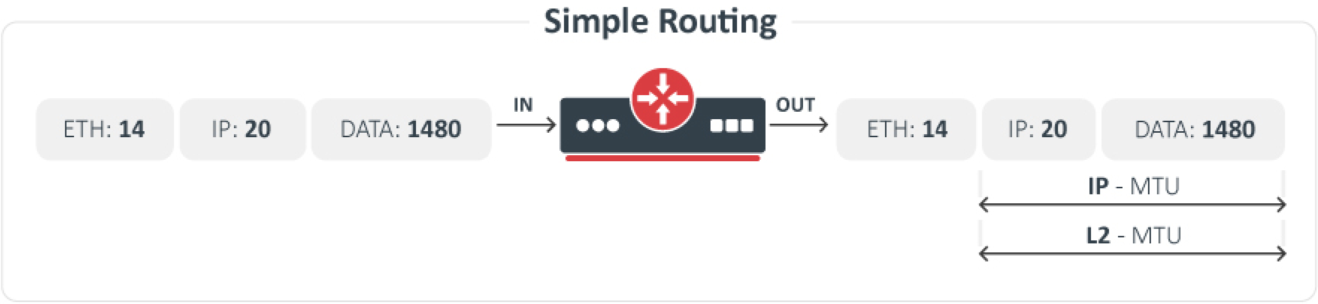
When the router encapsulates the Ethernet frame for forwarding over VPLS pseudowire, it checks if packet size with VPLS Control Word (4 bytes) and any necessary labels (usually 2 labels - 8 bytes), exceeds MPLS MTU of the outgoing interface. If it does, VPLS fragments packets so that it honors the MPLS MTU of the outgoing interface. A packet is defragmented at the egress point of the VPLS pseudowire.

Setup Examples

In these examples, we will take a look at frames entering and leaving the router via Ethernet interfaces.

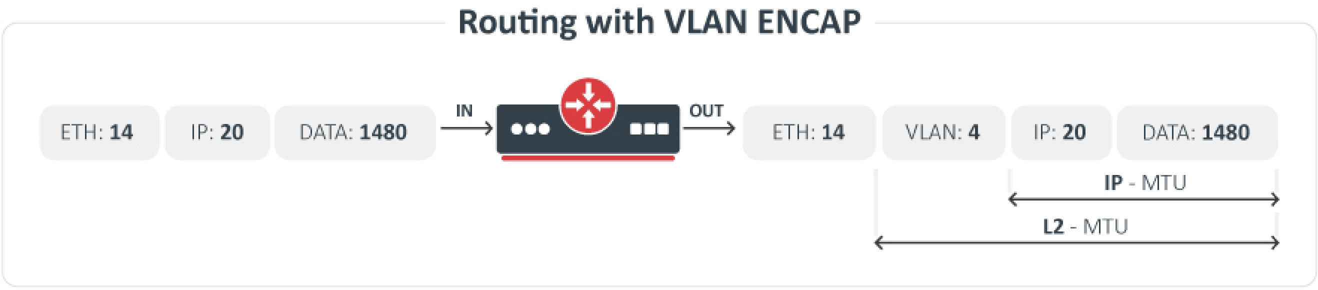
Simple Routing

The image shows the packet MTU size for simple routing, packets size is not modified.



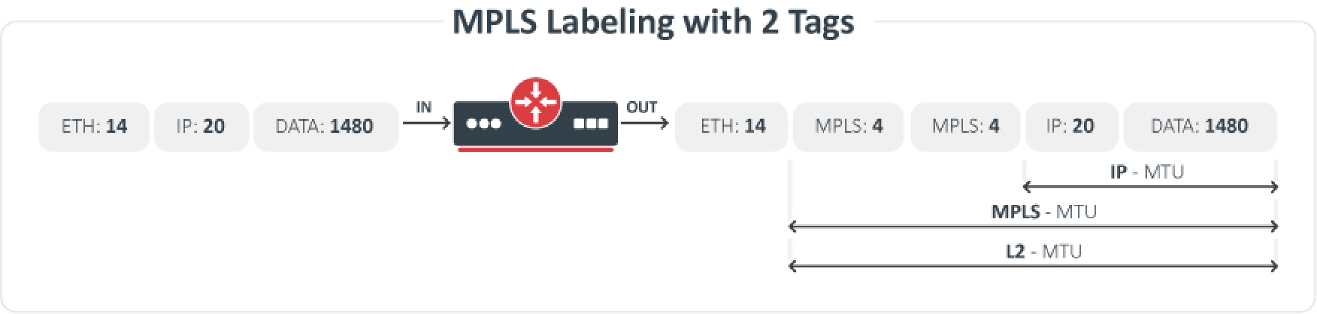
Routing with VLAN Encap

Each VLAN tag is 4 bytes long, the VLAN tag is added by a router. L2-MTU is increased by 4 bytes.



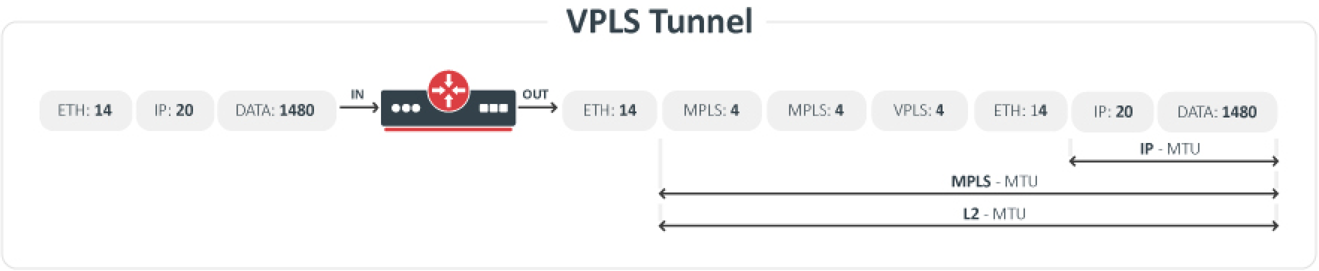
Simple MPLS with Tags

When MPLS is used as a plain replacement for IP routing, only one label is attached to every packet, therefore packet size increases by 4 bytes, we have the situation with two MPLS labels. In order to be able to forward standard size (1500 bytes) IP packets without fragmentation, MPLS MTU must be set to at least 1508 for two MPLS labels.



VPLS Tunnel

Two MPLS labels are present when a remote endpoint is not directly attached. One MPLS label is used to get to a remote endpoint, the second label is used to identify the VPLS tunnel.

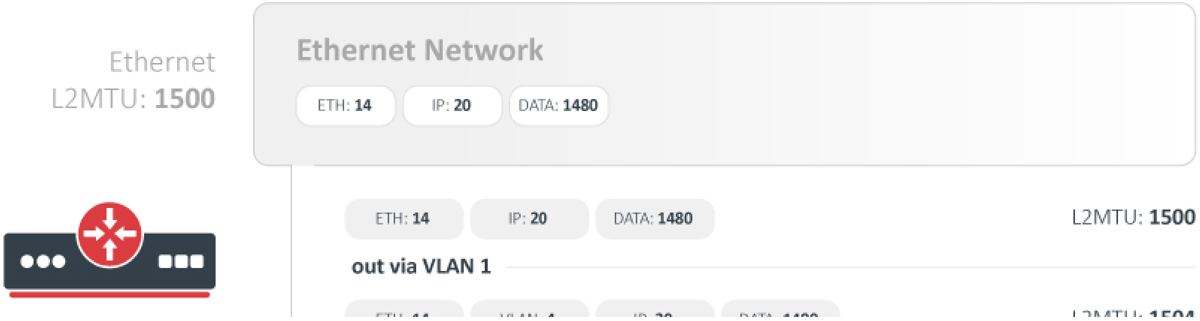


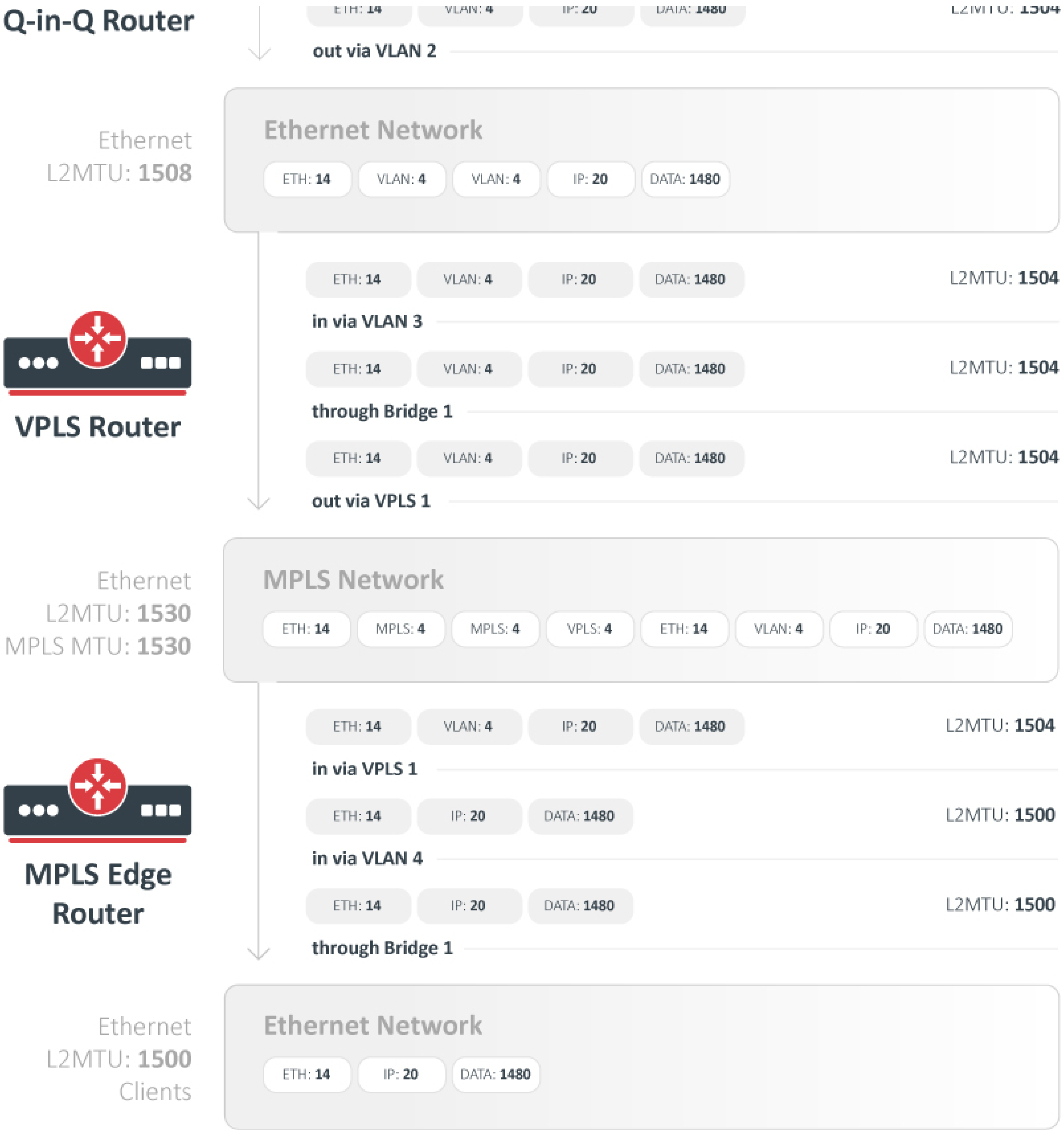
Advanced Setup Examples

In this example, we will take a closer look at the required L2MTU of all Ethernet-like interfaces including Bridge, VLAN, and VPLS interfaces.

In this setup we will have 3 routers:

- Q-in-Q router - this router will receive a standard 1500 byte Ethernet frame and will add two VLAN tags to the packet. Then packet will be sent out via an Ethernet network to the second router
- VPLS router - this router will remove the outer VLAN tag and will bridge the packet with the remaining VLAN tag with the VPLS tunnel. VPLS tunnel will take a packet through the MPLS network to the third router.
- MPLS Edge router - will remove VPLS and VLAN tags and bridge packet to the client Ethernet network.





Q-in-Q Router

ETH: 14 VLAN: 4 IP: 20 DATA: 1480 L2MTU: 1504

out via VLAN 2

Ethernet
L2MTU: **1508**

Ethernet Network

ETH: **14** VLAN: **4** VLAN: **4** IP: **20** DATA: **1480**

ETH: **14** VLAN: **4** IP: **20** DATA: **1480** L2MTU: **1504**

in via VLAN 3

ETH: **14** VLAN: **4** IP: **20** DATA: **1480** L2MTU: **1504**

through Bridge 1

ETH: **14** VLAN: **4** IP: **20** DATA: **1480** L2MTU: **1504**

out via VPLS 1

VPLS Router

Ethernet
L2MTU: **1530**
MPLS MTU: **1530**

MPLS Network

ETH: **14** MPLS: **4** MPLS: **4** VPLS: **4** ETH: **14** VLAN: **4** IP: **20** DATA: **1480**

ETH: **14** VLAN: **4** IP: **20** DATA: **1480** L2MTU: **1504**

in via VPLS 1

ETH: **14** IP: **20** DATA: **1480** L2MTU: **1500**

in via VLAN 4

ETH: **14** IP: **20** DATA: **1480** L2MTU: **1500**

through Bridge 1

MPLS Edge
Router

Ethernet
L2MTU: **1500**
Clients

Ethernet Network

ETH: **14** IP: **20** DATA: **1480**