

IT Technology 2. sem

Link Aggregation



University College

Authors

Tihamer Biliboc

tiha0006@edu.eal.dk

Sebastian Thomle Mason

seba7286@edu.eal.dk

Anthony James Peak

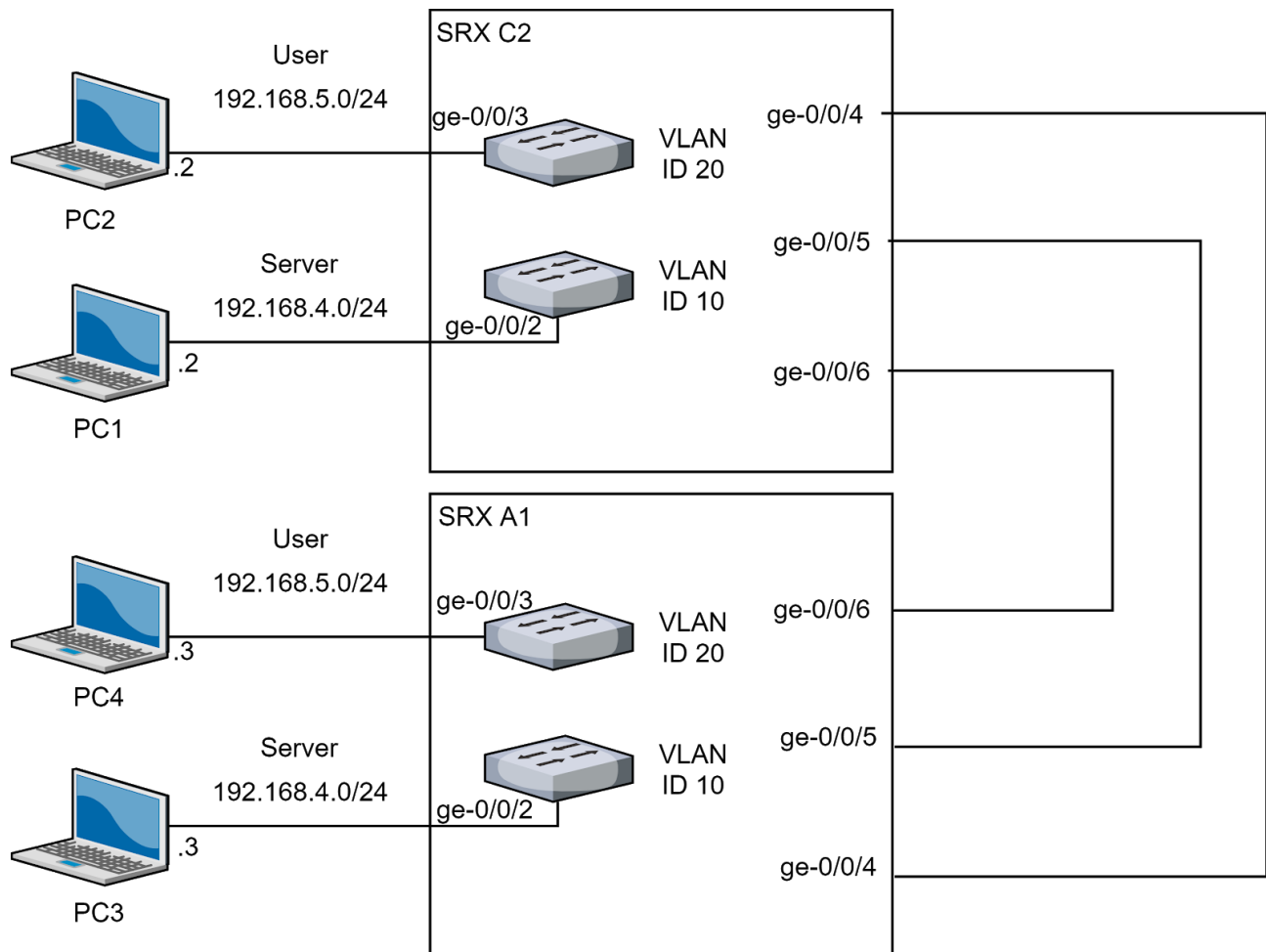
anth0662@edu.eal.dk

Sunday 17 Mar 2019

Table of Contents

A HLD with explanation	2
An inventory of used devices.	3
A Low Level Design	4
Show tagged and untagged traffic on the HLD diagram	5
Show how it is seen that the traffic is tagged and not tagged using wireshark	6
A filled out test plan	7
Put the router configuration and the topology diagram in GitHub	8

A HLD with explanation



By using an 802.1Q trunk, it is possible to have computers in separate buildings connected to the same subnet. In the above topology, a company has some devices in Building A (SRX A1) and some in Building B (SRX C2), but as indicated by the network addresses, they are on the same subnet. The servers in both buildings are on the .4.0/24 subnet, and the users are on the .5.0/24 subnet.

In this case, if a user in Building A wants to communicate with a user in Building B, the traffic will happen through the three aggregated links. Aggregation is more useful than a single trunk not only because it's faster, but also more durable. If one of the connection fails or disconnects for some reason, there is still connection on the other links. Our diagram illustrates 3 aggregated links, but of course it can be more.

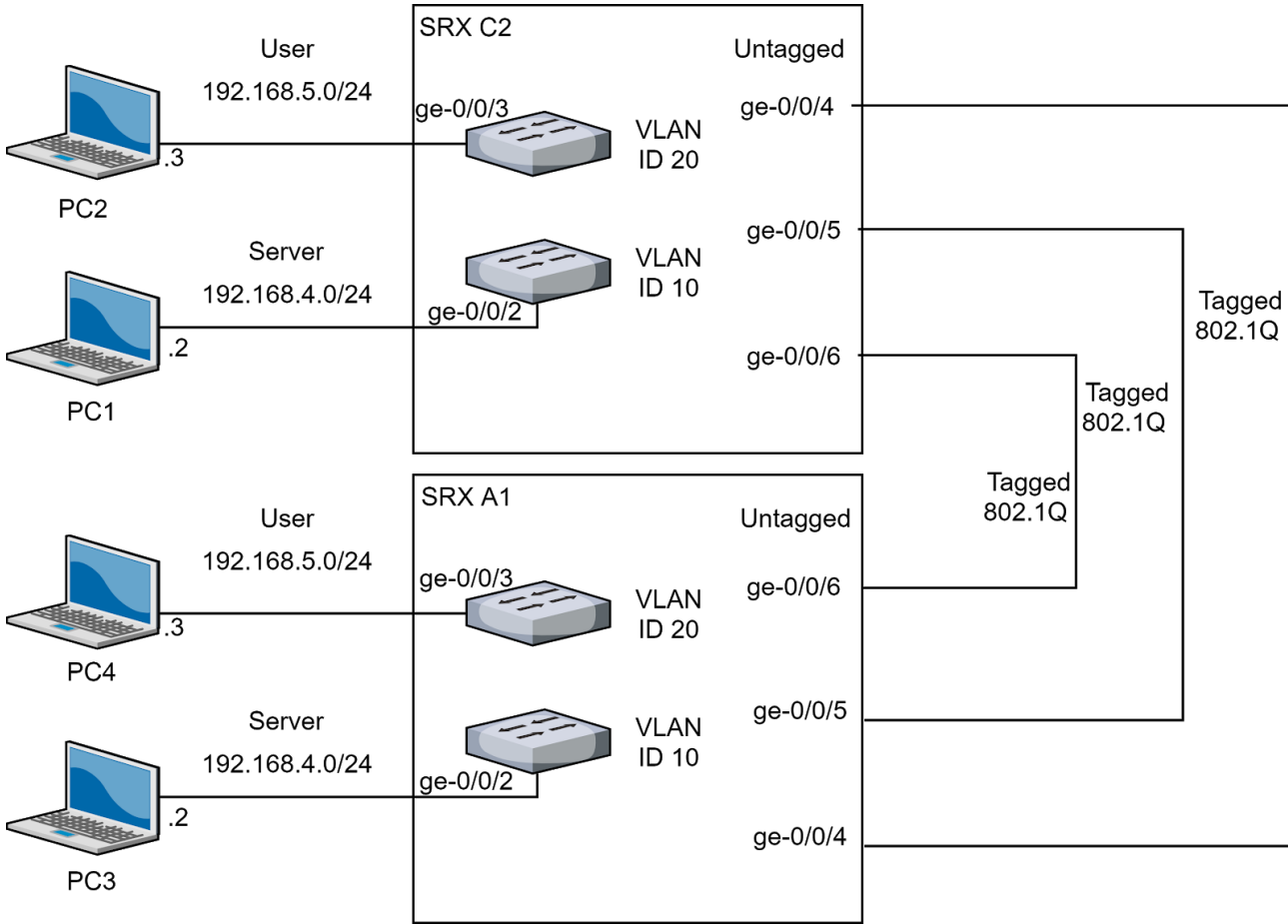
An inventory of used devices.

- SRX-C2
 - SRX240 Router
- SRX-A1
 - SRX240 Router
- Bunch of straight through ethernet cables with RJ45 connectors
- 1 Cisco Hub
- 3 PCs
- USB to RJ45 converter cable

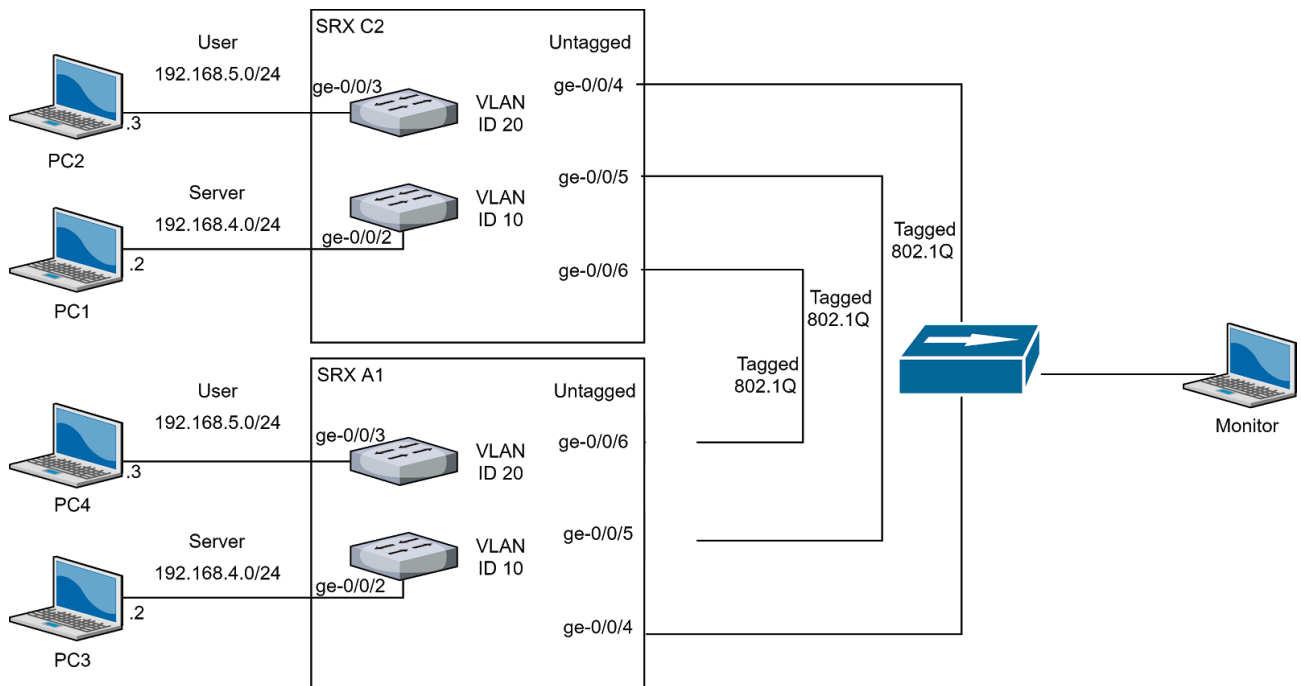
A Low Level Design

LOW LEVEL DESIGN ass10						
INSTANCE TYPE	INTERFACE	IP ADDRESS	VLAN ID	MASK	CONNECTS TO	Comments
SRX240-A1	ge-0/0/2	192.168.4.1	VLAN-10	/24	PC1	SERVER vlan
	ge-0/0/3	192.168.5.1	VLAN-20	/24	PC2	USER vlan
	ge-0/0/4				SRX240-C2	ae0
	ge-0/0/5				SRX240-C2	ae0
	ge-0/0/6				SRX240-C2	ae0
SRX240-C2	ge-0/0/2	192.168.4.1	VLAN-10	/24	PC3	SERVER vlan
	ge-0/0/3	192.168.5.1	VLAN-20	/24	PC4	USER vlan
	ge-0/0/4				SRX240-A1	ae0
	ge-0/0/5				SRX240-A1	ae0
	ge-0/0/6				SRX240-A1	ae0
PC1	eth0	192.168.4.2		/24	SRX240-A1	
PC2	eth0	192.168.5.2		/24	SRX240-A1	
PC3	eth0	192.168.4.3		/24	SRX240-C2	
PC4	eth0	192.168.5.3		/24	SRX240-C2	

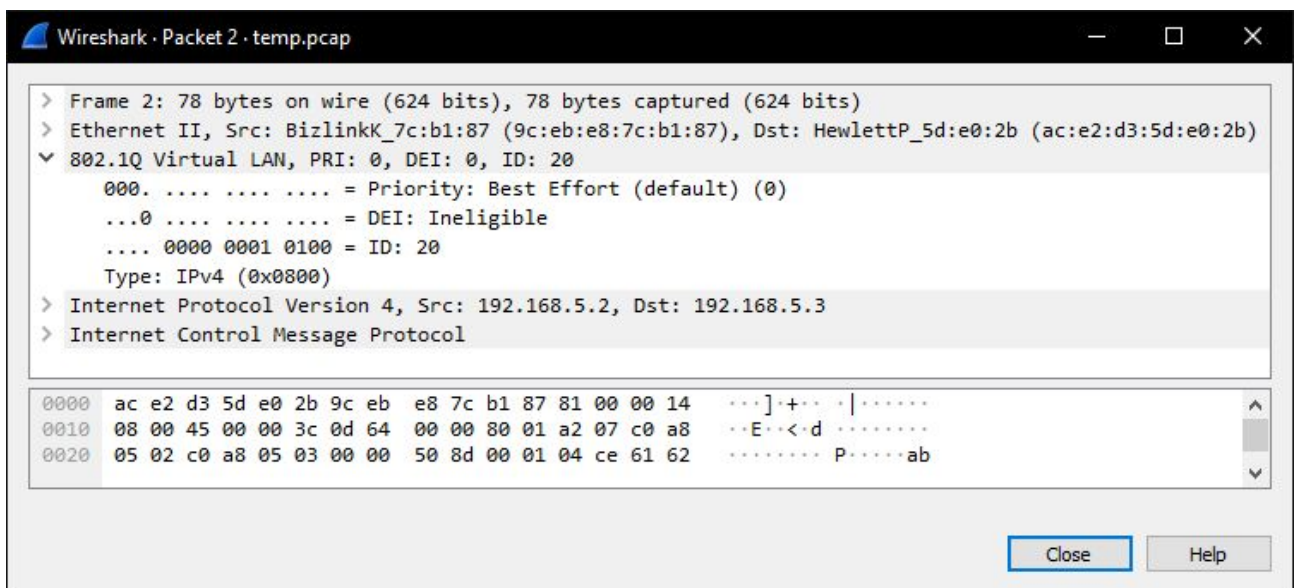
Show tagged and untagged traffic on the HLD diagram



Show how it is seen that the traffic is tagged and not tagged using wireshark



We unplugged one end of two of the cables to ensure that all traffic was sent through the last cable. We then attached a hub to the last cable and connected a monitor-PC to the hub so we could monitor the traffic.



If the “802.1Q” tab doesn't show up in your wireshark, see [here](https://osqa-ask.wireshark.org/questions/5996/how-to-configure-realtek-pcie-gbe-family-controller-to-capture-vlan-tag-packet/53113)¹

¹<https://osqa-ask.wireshark.org/questions/5996/how-to-configure-realtek-pcie-gbe-family-controller-to-capture-vlan-tag-packet/53113>

A filled out test plan

TEST PLAN			
ASSERTION	METHOD	EXPECTED RESULT	SUCCESS
VLAN10 (SRX-A1) can ping VLAN10 (SRX-C2) with 3 cables connecting the routers	PING from PC2 to PC4	Ping replies received	✓
VLAN10 (SRX-A1) can ping VLAN10 (SRX-C2) with 2 cables connecting the routers	PING from PC2 to PC4	Ping replies received	✓
VLAN10 (SRX-A1) can ping VLAN10 (SRX-C2) with 1 cables connecting the routers	PING from PC2 to PC4	Ping replies received	✓
VLAN10 (SRX-A1) can't ping VLAN10 (SRX-C2) with 0 cables connecting the routers	PING from PC2 to PC4	Ping replies not received	✓
The order of the cables connecting the routers doesn't matter	Swap the cables and PING from PC2 to PC4	Ping replies received	✓
Capturing Tagged packets with Wireshark	PING from PC2 to PC4 through a HUB	802.1Q appears in captured packets	✓

Put the router configuration and the topology diagram in GitHub

[Diagram](#)²

[Config](#)³

²<https://github.com/Sebski123/Network/blob/master/ITT2/ass10LinkAggregation/Diagrams/agregated.pdf>

³https://github.com/Sebski123/Network/blob/master/ITT2/ass10LinkAggregation/Router_configs/ass10.json