

SoftLayer Solutioning – VLANs & Subnets

An introduction to using VLANs and Subnets effectively in SoftLayer, including routing and firewalling in a multi-tiered implementation.

29th Sept, 2015
Authors: EJK/RG

SOFTLAYER[®]
an IBM Company

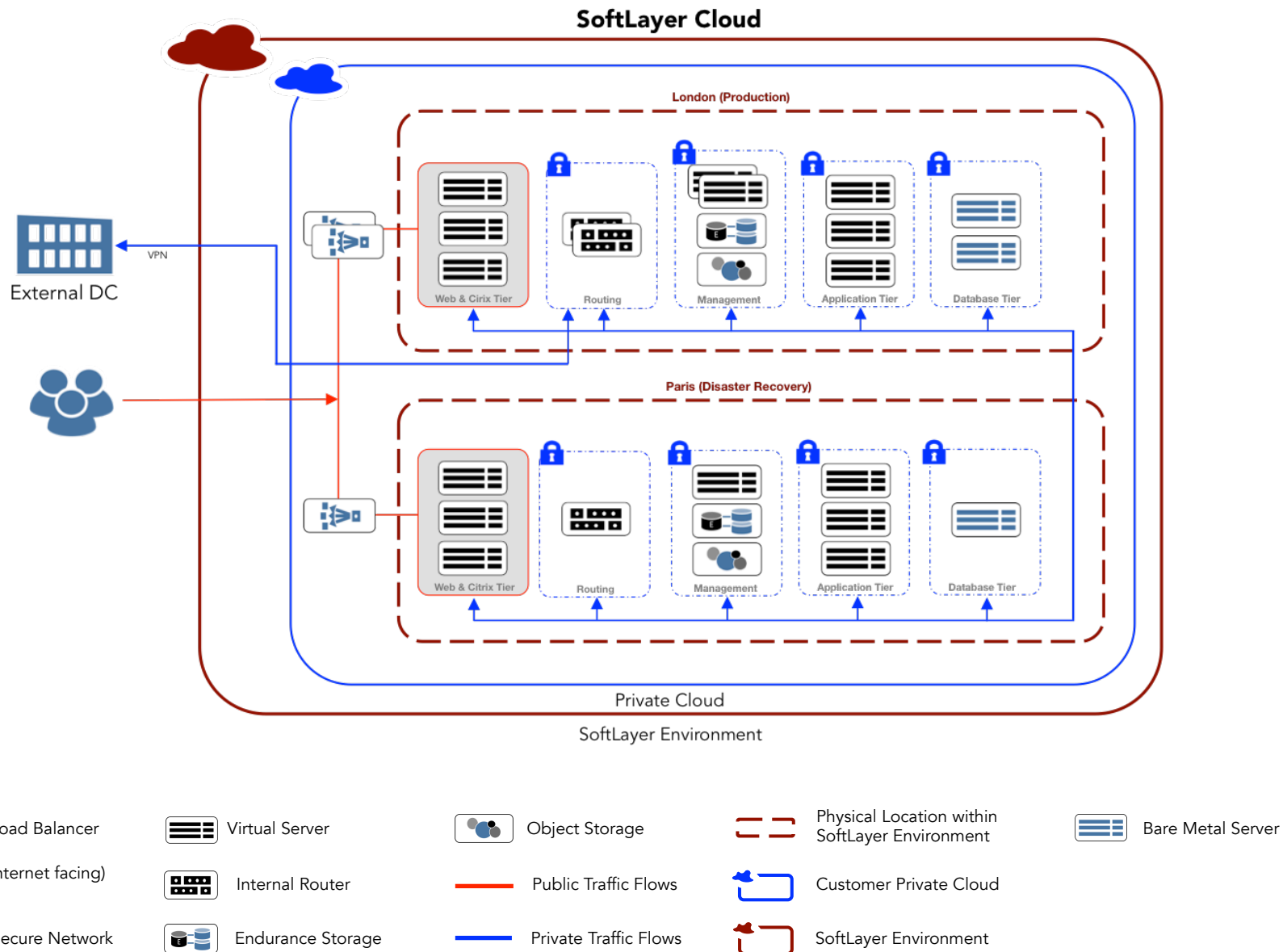


Contents

1. The Initial Design Phase
2. The Ordering Phase
3. The Detailed Design Phase
4. The Baseline implementation
5. The Bonding Implementation
6. The High Availability Implementation
7. The VLANs Implementation
8. Final Preparations for Firewalls
9. The Zone Firewall Implementation

 **The initial design phase**

Work out & agree the total vision first



Ask the more detailed questions

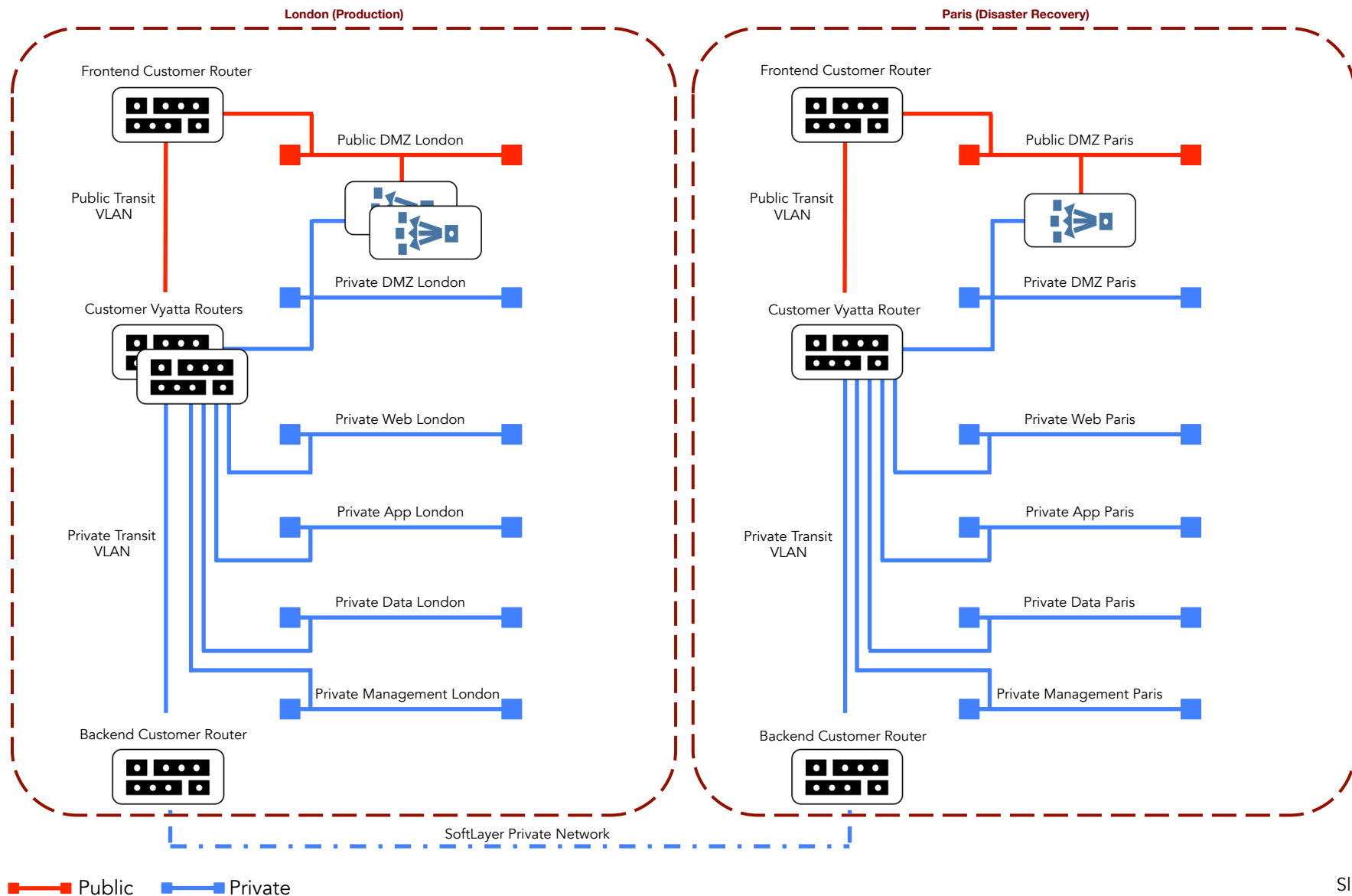
Questions to ask:

1. How many separate networks will you need?
2. Of these:
 - How many public facing?
 - How many private only?
3. In the total vision example its clear there is a need in both London and Paris for:
 - One public Internet facing network;
 - One private DMZ network to isolate incoming traffic;
 - One private web traffic network for the web & web application servers;
 - One application network for the application servers;
 - One data network to isolate out the database and file servers;
 - One management network for all the management traffic to all our nodes;
4. Will they be using highly available pairs of Vyatta's?
5. Work out the traffic flows between each network – what traffic are we going to allow?
 - What TCP ports will be allowed between these networks?
6. In SoftLayer the Vyatta Gateway Appliance will be the only entry & exit point for networks associated with it. To do this the Vyatta will come with two very specific VLANs – a public transit network and a private transit network that are 802.1q trunks assigned to the Vyatta's public and private interfaces.



Now work out the networks topology

We can now drilldown to a greater level of detail to understand how the networks (the VLANs) will hang together:



 **The ordering phase**

With this information we can begin orders

So before ordering anything! Log onto the SoftLayer Portal choose "Network" in the "Order" pane and then under Vyatta click "Order". Make sure in our design to order a HA Pair and name them rtr-lon-01 and rtr-lon-02. This could take up to two days but is usually 12 hours.

Order

Devices

Storage

Network

Security

Marketplace

Devices

Storage

Network

Security

Marketplace

work options below. All Network order links will direct you to other locations within Control unless marked otherwise.

[Vyatta](#)

L4 load balancing, L7 traffic management, hping and compression and web application acceleration for the largest websites in the world.

Vyatta protects your cloud infrastructure and optimizes its performance. Create virtual routers, virtual firewalls, and virtual VPN devices and manage devices through userdefined parameters.

[Order](#)

SoftLayer® Technologies - Customer Po

SoftLayer Technologies, Inc. [US] | <https://manage.softlayer.com/Sales/gatewayServer>

Gateway Appliances

Single Processor Multi-Core Servers

Manufacturer	CPU Type	Core Count	CPU Speed	Storage	Ram	Private Network Only	10 Gbps Uplink Support	Starting Price Per Month
Intel	Intel Xeon E3-1270	4 Cores	3.40GHz	Up to 4 drives	4 GB - 32 GB	-	-	\$418.00
Intel	Intel Xeon E3-1270 v3	4 Cores	3.50GHz	Up to 4 drives	4 GB - 32 GB	-	-	\$428.00

Dual Processor Multi-Core Servers

Manufacturer	CPU Type	Core Count	CPU Speed	Storage	Ram	Private Network Only	10 Gbps Uplink Support	Starting Price Per Month
Intel	Intel Xeon E5-2620	12 Cores	2.00GHz	Up to 12 drives	16 GB - 256 GB	-	Yes	\$748.00
Intel	Intel Xeon E5-2650	16 Cores	2.00GHz	Up to 12 drives	16 GB - 256 GB	-	Yes	\$928.00
Intel	Intel Xeon E5-2620 v3	12 Cores	2.40GHz	Up to 12 drives	16 GB - 256 GB	-	Yes	\$918.00
Intel	Intel Xeon E5-2690	16 Cores	2.90GHz	Up to 12 drives	16 GB - 256 GB	-	Yes	\$928.00
Intel	Intel Xeon E5-2650 v3	20 Cores	2.30GHz	Up to 12 drives	16 GB - 256 GB	-	Yes	\$998.00
Intel	Intel Xeon E5-2690 v3	24 Cores	2.60GHz	Up to 12 drives	16 GB - 256 GB	-	Yes	\$1,098.00

SOFTLAYER
an IBM Corp

Configure your Network Gateway Appliance (Monthly)

Quantity

☒ High Availability Pair

Location

DATA CENTER

☒ LON02 - London

Show Data Centers



Step 2: Check your Vyatta's networks

Once you get the email saying the Vyatta's are 'ready/available' log into the portal and go to DEVICES->DEVICE LIST->rtr-lon-01 and scroll down to the network section to see something like this:

Network								
Public		Order IPs	Private		Order IPs	Management		
eth1			eth0			mgmt0		
Status:	ACTIVE		Status:	ACTIVE		Status:	ACTIVE	
IP Address:	159.10.12.107		IP Address:	10.1.8.0		IP Address:	10.1.8.62	
Default Gateway:	159.10.12.97		Default Gateway:	10.1.8.1		Default Gateway:	10.1.8.1	
Subnet Mask:	255.255.255.240		Subnet Mask:	255.255.255.192		Subnet Mask:	255.255.255.192	
Speed:	<div>2000 Mbps (Dual) ▼</div>		Speed:	<div>2000 Mbps (Dual) ▼</div>		VLAN:	lon02.bcr01a.1232	
Max Speed:	2000 Mbps Modify Max Speed		Max Speed:	2000 Mbps Modify Max Speed		Network Hardware:	Show Details	
VLAN:	lon02.fcr01a.1231		VLAN:	lon02.bcr01a.1232				
Network Hardware:	Show Details		Network Hardware:	Show Details				
eth1			eth0					
Redundant for eth1			Redundant for eth0					

Make a note of the two VLAN numbers – in this case 1231 and 1232. These are your transit VLANs for London. Now go and do the same for the Paris Vyatta and make a note of the VLANs.



Step 3: Name your transit VLANs

Before ordering anything else from the Portal go to NETWORK->IP MANAGEMENT->VLANs. You will see an output like this:

Viewing 1 to 16 of 16 VLANs

Displaying per page

Name(Click cell to edit)	VLAN number	Primary Router	Notes	Gateway / Firewall
	1231	fcr01a.lon02		Add Firewall
	1232	bcr01a.lon02		
	901	fcr01a.par01		
	902	bcr01a.par01		

⋮

Note: Your VLAN numbers are highly unlikely to be sequential.

Click on the box beside each VLAN Number and name the VLANs to end up with this:

Viewing 1 to 16 of 16 VLANs

Displaying per page

Name(Click cell to edit)	VLAN number	Primary Router	Notes	Gateway / Firewall
Public Transit Londo	1231	fcr01a.lon02		Add Firewall
Private Transit Lond	1232	bcr01a.lon02		
Public Transit Paris	901	fcr01a.par01		
Private Transit Pari	902	bcr01a.par01		

⋮

Note: Your VLAN numbers are highly unlikely to be sequential.

Now we are in a position to raise a ticket for all our other VLANs. In the ticket answer the questions for SoftLayer:

1. How many public and private VLANs are you needing?
2. Which data center are you needing these VLANs in/Which specific router?
3. Which primary subnet size are you needing for each VLAN?
4. How soon will you be placing servers in these VLANs?
5. How many servers will you be placing in these VLANs?
6. What will these servers be used for?
7. Please approve the \$25 per month per VLAN pricing.

From our design we know in London we will need:

- 1 x Public VLAN;
- 5 x Private VLANs;

And we know in Paris we will need:

- 1 x Public VLAN;
- 5 x Private VLANs;



Step 4a: Name all the other VLANs

Once your order for the VLANs is approved you can log into the Portal and name all your VLANs in accordance with the design. When you log in you and go to NETWORK -> IP Management -> VLANs you will see something like this:

Viewing 1 to 16 of 16 VLANs

Displaying per page

Name(Click cell to edit)	VLAN number	Primary Router	Notes	Gateway / Firewall
Public Transit London	1231	fcr01a.lon02		Add Firewall
Private Transit London	1232	bcr01a.lon02		
	1233	fcr01a.lon02		Add Firewall
	1234	bcr01a.lon02		
	1235	bcr01a.lon02		
	1236	bcr01a.lon02		
	1237	bcr01a.lon02		
	1238	bcr01a.lon02		
Public Transit Paris	901	fcr01a.par01		Add Firewall
Private Transit Paris	902	bcr01a.par01		
	903	fcr01a.par01		Add Firewall
	904	bcr01a.par01		
	905	bcr01a.par01		
	906	bcr01a.par01		
	907	bcr01a.par01		
	908	bcr01a.par01		

Note: Your VLAN numbers are highly unlikely to be sequential.



Step 4b: Name all the other VLANs

Fill in all the VLAN names to end up with something like this:

Viewing 1 to 16 of 16 VLANs

Displaying per page

Name(Click cell to edit)	VLAN number	Primary Router	Notes	Gateway / Firewall
Public Transit London	1231	fcr01a.lon02		Add Firewall
Private Transit London	1232	bcr01a.lon02		
Public DMZ London	1233	fcr01a.lon02		Add Firewall
Private Web London	1234	bcr01a.lon02		
Private App London	1235	bcr01a.lon02		
Private Data London	1236	bcr01a.lon02		
Private Manage London	1237	bcr01a.lon02		
Private DMZ London	1238	bcr01a.lon02		
Public Transit Paris	901	fcr01a.par01		Add Firewall
Private Transit Paris	902	bcr01a.par01		
Public DMZ Paris	903	fcr01a.par01		Add Firewall
Private Web Paris	904	bcr01a.par01		
Private App Paris	905	bcr01a.par01		
Private Data Paris	906	bcr01a.par01		
Private Manage Paris	907	bcr01a.par01		
Private DMZ Paris	908	bcr01a.par01		

Note: Your VLAN numbers are highly unlikely to be sequential.

Step 4c: Associate all the VLANs

Now we have everything named correctly we can use the portal to associate our VLANs with a particular gateway. To do this navigate to NETWORK -> GATEWAY APPLICANCES. You should see output like this:

Viewing 1 to 2 of 2

Displaying 25 per page

Gateway	Public VLAN	Public IP address	Private VLAN	Private IP Address	Associated VLANs
london	lon02.fcr01a.1231	159.122.84.105	lon02.bcr01a.1232	10.1.9.99	
paris	par01.fcr01a.901	159.8.101.23	par01.bcr01a.902	10.2.9.99	

Clicking on a Gateway reveals its details and enables us to associated our VLANs:

London Details

Network: Public/Private **Status:** Public/Private **Group Number:** 1
Public Gateway IP: 159.10.12.2 **Private Gateway IP:** 10.1.9.2 **Public Gateway IPv6:** 2a03:8180:1101:20b::4
Configuration: High-Availability

Members

Member	Public IP address	Public IPv6	Private IP Address	Priority	Manage Gateway	Username / Password
rtr-lon-01	159.10.12.100	lon02.bcr01a.1232	10.1.9.99	254	10.1.9.2	root / Xthl3RQs
rtr-lon-02	159.10.12.101	lon02.bcr01a.1232	10.2.9.99	253	10.2.9.3	root / KL5HtyuP

Gateway VLANs

Gateway VLAN		Network
▶	lon02.fcr01a.1231	Public
▶	lon02.bcr01a.1232	Private

Associate a VLAN

Eligible VLANs are limited to those on this gateway's router(s) which aren't firewalled or used by another gateway.



Use the drop down to associate your VLANs



Step 4c: Associate all the VLANs

Once you have associated the VLANs, returning to NETWORK -> IP Management -> VLANs you will see something like this :

Viewing 1 to 16 of 16 VLANs

Displaying per page

Name(Click cell to edit)	VLAN number	Primary Router	Notes	Gateway / Firewall
Public Transit London	1231	fcr01a.lon02		Add Firewall
Private Transit London	1232	bcr01a.lon02		
Public DMZ London	1233	fcr01a.lon02		Add Firewall
Private Web London	1234	bcr01a.lon02		London
Private App London	1235	bcr01a.lon02		London
Private Data London	1236	bcr01a.lon02		London
Private Manage London	1237	bcr01a.lon02		London
Private DMZ London	1238	bcr01a.lon02		London
Public Transit Paris	901	fcr01a.par01		Add Firewall
Private Transit Paris	902	bcr01a.par01		
Public DMZ Paris	903	fcr01a.par01		Add Firewall
Private Web Paris	904	bcr01a.par01		Paris
Private App Paris	905	bcr01a.par01		Paris
Private Data Paris	906	bcr01a.par01		Paris
Private Manage Paris	907	bcr01a.par01		Paris
Private DMZ Paris	908	bcr01a.par01		Paris


Note: Your VLAN numbers are highly unlikely to be sequential.





Step 5: Order the Citrix VPX's & other devices


Now we have named all our VLANs we are in a position to make the remainder of our orders. On the confirmation page make sure to choose the correct VLAN placement for each device.


Order


Devices


Storage


Network


Security


Marketplace

Devices

Find information on ordering devices below. All device order links will redirect to another site.

Bare Metal Servers

Sometimes you need the raw horsepower of bare metal. SoftLayer dedicated servers give you options: hex-core, and even GPU-powered workhorses.

[Hourly](#) [Monthly](#)

Virtual Server (private node)

Single-tenant environment with SoftLayer managed hypervisor, ideal for applications with stringent resource requirements.

[Hourly](#) [Monthly](#)

Virtual Server (public node)

Multi-tenant environment with SoftLayer managed hypervisor scalability and higher-cost effectiveness.

[Hourly](#) [Monthly](#)

Network

Find information on ordering Network options below. All Network order links will

Citrix NetScaler VPX

Cost effectively deploy the same L4 load balancing, L7 traffic management, TCP and SSL offload, content caching and compression and web application firewall functionality used by the largest websites in the world.

[+ Order](#)

Slide 15

 **The detailed design phase**

Examining our VLANs

We are now ready to build much more detailed diagrams of our environment. Log in to the Portal and go to the VLAN page. From this page you can see hyperlinks for each of the VLANs. Click on these links to reveal more details about your VLAN – specifically we are needing for detailed design to get to the subnets:

Viewing 1 to 16 of 16 VLANs

Displaying per page

Name(Click cell to edit)	VLAN number	Primary Router	Notes	Gateway / Firewall
Public Transit London	1231	fcr01a.lon02		Add Firewall
Private Transit London	1232	bcr01a.lon02		
Public DMZ London	1233	fcr01a.lon02		Add Firewall
Private Web London	1234	bcr01a.lon02		London
Private App London	1235	bcr01a.lon02		London
Private Data London	1236	bcr01a.lon02		London
Private Manage London	1237	bcr01a.lon02		London
Private DMZ London	1238	bcr01a.lon02		London
Public Transit Paris	901	fcr01a.par01		Add Firewall
Private Transit Paris	902	bcr01a.par01		
Public DMZ Paris	903	fcr01a.par01		Add Firewall
Private Web Paris	904	bcr01a.par01		Paris
Private App Paris	905	bcr01a.par01		Paris
Private Data Paris	906	bcr01a.par01		Paris
Private Manage Paris	907	bcr01a.par01		Paris
Private DMZ Paris	908	bcr01a.par01		Paris

Note: Your VLAN numbers are highly unlikely to be sequential.



Examining our Subnets by VLAN

When we click on VLAN 1231 we will see output like this:

Public VLAN 1231 on fcr01a.lon02 Details

Name: Public Transit London	VLAN Number: 1231	Primary Router: fcr01a.lon02	Location: London
Gateway / Firewall: None			

Devices

Name	Type	Public IP
rtr-lon-01.mydomain.com	Server	159.10.12.107
rtr-lon-02.mydomain.com	Server	159.10.12.110

Subnets

Subnet	Type	Target	Type
159.10.12.96/28	Primary		16

Note: Depending on whether you have already ordered all the devices you will get more listed in the “Devices” box.

From here we can review our subnet in detail. Click on the subnet hyperlink “159.10.12.96/28” to review all of the available addresses.



Examining your Subnets IP addresses

All the IPs for the subnet are revealed:

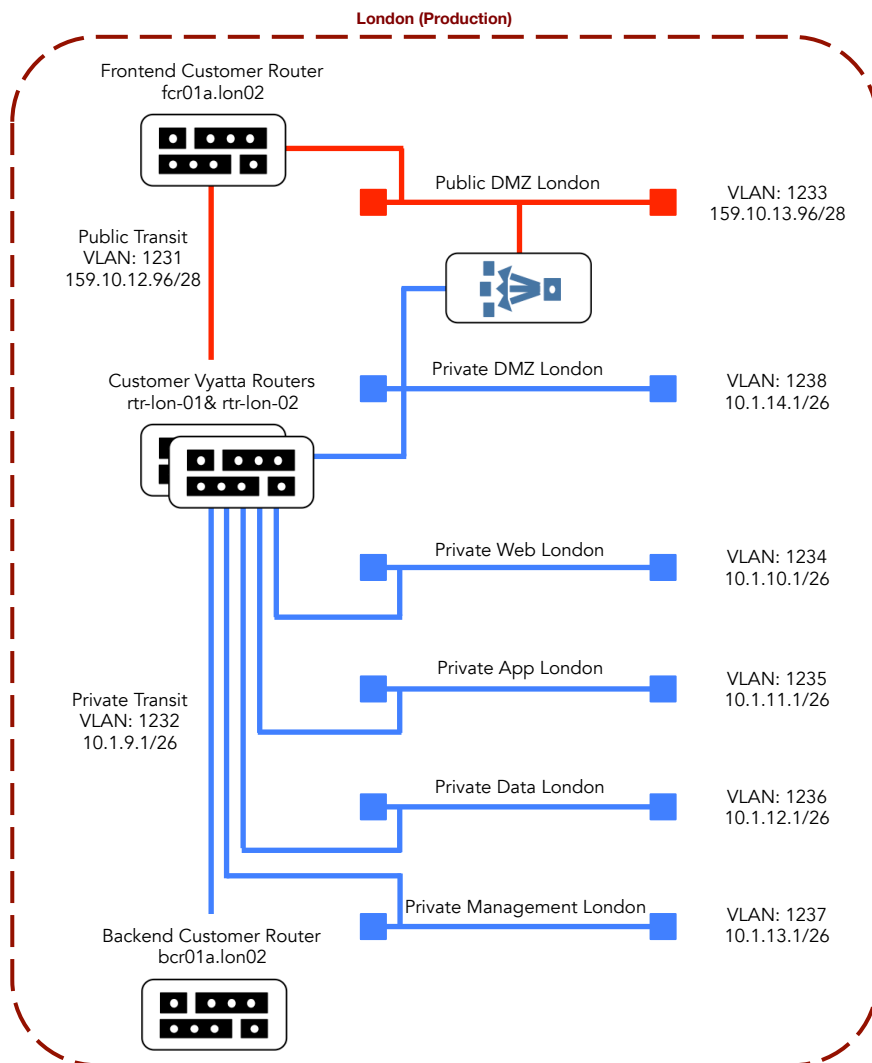
Subnet 159.10.12.96/28 Details

IP	Status	Description	Notes
159.10.12.96	Reserved	Network	
159.10.12.97	Reserved	Gateway	
159.10.12.98	Reserved		Reserved for HSRP
159.10.12.99	Reserved		Reserved for HSRP
159.10.12.100	Reserved	Primary IP for future server only	
159.10.12.101	Reserved	Primary IP for future server only	
159.10.12.102	Reserved	Primary IP for future server only	
159.10.12.103	Reserved	Primary IP for future server only	
159.10.12.104	Reserved	Primary IP for future server only	
159.10.12.105	Reserved	Primary IP for future server only	
159.10.12.106	Reserved	Primary IP for future server only	
159.10.12.107	Admin IP	rtr-lon-01.mydomain.com	
159.10.12.108	Reserved	Primary IP for future server only	
159.10.12.109	Reserved	Primary IP for future server only	
159.10.12.110	Admin IP	rtr-lon-02.mydomain.com	
159.10.12.111	Reserved	Broadcast	

Make a note of the Network address, the Gateway address, the HSRP addresses and the broadcast. Unfortunately you will need to do this for every VLAN. Once you have all this information you can proceed to Traffic Flows.

Traffic Flows – Vyatta London

Draw out a more detailed design of each network - like this:



VLAN key details:

VLAN 1231: 159.10.12.96/26
Gateway: 159.10.12.97/26
HSRP1: 159.10.12.98/26
HSRP2: 159.10.12.99/26
rtr-lon-01 assigned on creation: 159.10.12.100/26
rtr-lon-02 assigned on creation: 159.10.12.101/26
rtr-lon-01 assigned on creation: 159.10.12.102/26
rtr-lon-02 assigned on creation: 159.10.12.103/26

VLAN 1232: 10.1.9.1/26
Gateway: 10.1.9.2/26
HSRP1: 10.1.9.3/26
HSRP2: 10.1.9.4/26
rtr-lon-01 assigned on creation: 10.1.9.5/26
rtr-lon-02 assigned on creation: 10.1.9.6/26
rtr-lon-01 assigned on creation: 10.1.9.7/26
rtr-lon-02 assigned on creation: 10.1.9.8/26

VLAN 1233: 159.10.13.96/28
Gateway: 159.10.13.97/28
HSRP1: 159.10.13.98/28
HSRP2: 159.10.13.99/28

VLAN 1234: 10.1.10.1/26
Gateway: 10.1.10.2/26
HSRP1: 10.1.10.3/26
HSRP2: 10.1.10.4/26

VLAN 1235: 10.1.11.1/26
Gateway: 10.1.11.2/26
HSRP1: 10.1.11.3/26
HSRP2: 10.1.11.4/26

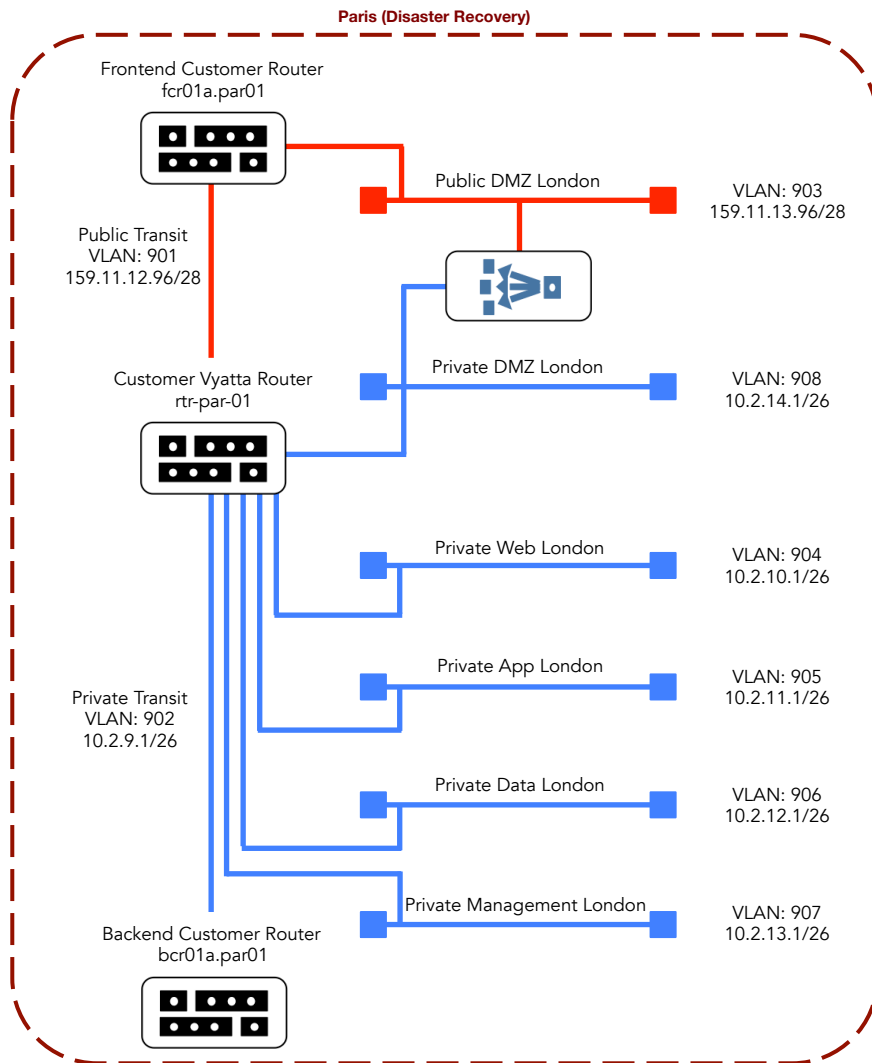
VLAN 1236: 10.1.12.1/26
Gateway: 10.1.12.2/26
HSRP1: 10.1.12.3/26
HSRP2: 10.1.12.4/26

VLAN 1237: 10.1.13.1/26
Gateway: 10.1.13.2/26
HSRP1: 10.1.13.3/26
HSRP2: 10.1.13.4/26

VLAN 1238: 10.1.14.1/26
Gateway: 10.1.14.2/26
HSRP1: 10.1.14.3/26
HSRP2: 10.1.14.4/26

Traffic Flows – Vyatta Paris

Draw out a more detailed design of each network - like this:



VLAN key details:

VLAN 901: 159.11.12.96/26
Gateway: 159.11.12.97/26
HSRP1: 159.11.12.98/26
HSRP2: 159.11.12.99/26
rtr-par-01 assigned on creation: 159.11.12.100/26
rtr-par-01 assigned on creation: 159.11.12.101/26

VLAN 902: 10.2.9.1/26
Gateway: 10.2.9.2/26
HSRP1: 10.2.9.3/26
HSRP2: 10.2.9.4/26
rtr-par-01 assigned on creation: 10.2.9.5/26
rtr-par-01 assigned on creation: 10.2.9.6/26

VLAN 903: 159.11.13.96/28
Gateway: 159.11.13.97/28
HSRP1: 159.11.13.98/28
HSRP2: 159.11.13.99/28

VLAN 904: 10.2.10.1/26
Gateway: 10.2.10.2/26
HSRP1: 10.2.10.3/26
HSRP2: 10.2.10.4/26

VLAN 905: 10.2.11.1/26
Gateway: 10.2.11.2/26
HSRP1: 10.2.11.3/26
HSRP2: 10.2.11.4/26

VLAN 906: 10.2.12.1/26
Gateway: 10.2.12.2/26
HSRP1: 10.2.12.3/26
HSRP2: 10.2.12.4/26

VLAN 907: 10.2.13.1/26
Gateway: 10.2.13.2/26
HSRP1: 10.2.13.3/26
HSRP2: 10.2.13.4/26

VLAN 908: 10.2.14.1/26
Gateway: 10.2.14.2/26
HSRP1: 10.2.14.3/26
HSRP2: 10.2.14.4/26

Note: It is acknowledged that it is entirely unlikely that the numbers would be so closely aligned but this is for the purposes of illustration.

Traffic Flows - design

This type of VLAN design makes it possible to have zone to zone firewall configurations to block/permit traffic. The following pages outline a mechanism for building that design. Begin with a blank canvas of all your VLANs:

LONDON	To Zone From Zone	DMZ	Web	App	Database	Management
	DMZ					
	Web					
	App					
	Database					
	Management					

PARIS	To Zone From Zone	DMZ	Web	App	Database	Management
	DMZ					
	Web					
	App					
	Database					
	Management					



Traffic Flows – Intra-Site Firewall Zones

Then we can methodically fill in the appropriate permissions between the VLANs as follows:

LONDON	To Zone From Zone	DMZ	Web	App	Database	Management
	DMZ	---	YES	NO	NO	NO
	Web	NO	---	YES	YES (445/1433)	YES
	App	NO	YES (445)	---	YES	YES
	Database	NO	YES (445)	YES (445)	---	YES
	Management	NO	YES	YES	YES	---

PARIS	To Zone From Zone	DMZ	Web	App	Database	Management
	DMZ	---	YES	NO	NO	NO
	Web	NO	---	YES	YES (445/1433)	YES
	App	NO	YES (445)	---	YES	YES
	Database	NO	YES (445)	YES (445)	---	YES
	Management	NO	YES	YES	YES	---



Traffic Flows – Site-to-Site Firewall Zones

Finally we can fill in the Zone to Zone between the sites - as follows:

LONDON TO PARIS	To Zone From Zone	DMZ	Web	App	Database	Management
	DMZ	NO	NO	NO	NO	NO
	Web	NO	NO	NO	NO	NO
	App	NO	NO	NO	NO	NO
	Database	NO	NO	NO	YES	NO
	Management	NO	NO	NO	NO	YES
PARIS TO LONDON	To Zone From Zone	DMZ	Web	App	Database	Management
	DMZ	NO	NO	NO	NO	NO
	Web	NO	NO	NO	NO	NO
	App	NO	NO	NO	NO	NO
	Database	NO	NO	NO	YES	NO
	Management	NO	NO	NO	NO	YES

 **The baseline implementation**

Understanding the Vyatta

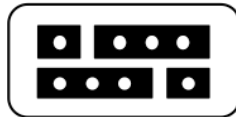
It's a ROUTER



The Vyatta is a Debian based Linux distribution that provides for Layer 3 routing:

- Create multi-tiered infrastructures;
- Connect diverse networks;
- Enable QoS for traffic management;
- Utilize:
 - BGPv4/v6;
 - OSPFv2/v3;
 - BGP Multipath;
 - RIPv2;
 - Static Routes;
 - Policy-Based Routing (PBR)
- Deliver high availability with VRRP

It's a FIREWALL



The Vyatta firewall capability analyzes and filters IP packets between its network interfaces. Vyatta delivers:

- Packet filtering for traffic that traverses the router using the "in" and "out" keywords;
- Definable packet matching criteria such as the source IP address, the destination IP address, the IP protocol, etc.;
- General detection on IP options for instance source routing, broadcast packets, etc.;
- Global stateless or stateful configuration

It's a VPN GATEWAY



The Vyatta firewall capability analyzes and filters IP packets between its network interfaces. Vyatta delivers:

- SSL-based OpenVPN;
- Site to site VPN (IPSec);
- Remote VPN (PPTP, L2TP, IPSec);
- OpenVPN Client Auto-Configuration;
- Layer 2 Bridging over GRE;
- Layer 2 Bridging over OpenVPN;
- OpenVPN Dynamic Client;
- Dynamic Multipoint VPN

In this presentation we are concerned primarily with Firewalling with rudimentary Routing.



First time login on the Vyatta or VyOS

On your local VirtualBox (Vbox) VyOS machine:

```
Welcome to VyOS – vyos tty1

Vyos login: vyos
Password:
Last login: Sun Sep 20 19:30:37 UTC 2015 on tty1
Linux vyos 3.13.11-1-amd64-vyos #1 SMP Wed Aug 12 02:08:05 UTC 2015 x86_64
Welcome to VyOS.
This system is open-source software. The exact distribution terms for each
module comprising the full system are described in the individual files
inn /usr/share/doc/*/copyright.
vyos@vyos:~$
```

On your SoftLayer Vyatta:

```
Eamonns-iMac:~ saasify$ ssh vyatta@159.10.12.100
Welcome to Vyatta
Version:          VSE6.7R9
Description:      Brocade Vyatta 5415 vRouter 6.7 R9
Copyright:       2006-2015 Vyatta, Inc.
Last login: Sun Sep 20 19:30:37 2015 from host82-159-53-166.host82-159-53-166
vyatta@hostname:~$
```

From this point forwards this presentation will utilize the open source version VyOS to outline commands in order that the user can follow using the accompanying tutorial videos.

Initial “hardening” tasks

Your first job on logging in is to harden the device. Hardening varies depending on the security requirements of your organization but in general this means:

1. Adding a new admin user specific for your organization;
2. Deleting the ability of “vyos” (or Vyatta in the case of a SoftLayer machine) to login;
3. Set the systems hostname;
4. Set the systems time zone;
5. Set the systems domain name;
6. In the case of our Vbox machine set up the Ethernet interfaces (this will already be done on your SoftLayer Vyatta);
7. Enable SSH access on a random port of your choice;

```
vyatta@hostname:~$ configure
[edit]
vyatta@hostname:~# set system login user zeus authentication plaintext-password "Gr33kGods"
[edit]
vyatta@hostname:~# set system login user zeus level admin
[edit]
vyatta@hostname:~# commit
[edit]
vyatta@hostname:~# save
Saving configuration to '/config/config.boot' ...
Done
[edit]
vyatta@hostname:~# exit
exit
vyatta@hostname:~$ exit
```

Now log back in as “zeus”, make sure you can get to configuration mode and we can safely remove “vyos” (or “vyatta”) access:

```
zeus@hostname:~$ configure
[edit]
zeus@hostname:~# delete system login user vyos
[edit]
zeus@hostname:~# commit
[edit]
zeus@hostname:~# save
Saving configuration to '/config/config.boot' ...
Done
```



Initial “hardening” tasks continued ...

Now we can set the hostname, time zone and domain name:

```
zeus@hostname:~# set system host-name rtr-lon-01
[edit]
zeus@hostname:~# commit
[ system host-name rtr-01-vyos ]
Stopping enhanced syslogd: rsyslogd.
Starting enhanced syslogd: rsyslogd.

[edit]
zeus@hostname:~# save
Saving configuration to '/config/config.boot' ...
Done
```

Log out using “exit” then “exit” and your login screen should be:

```
Welcome to VyOS – rtr-lon-01 tty1

rtr-lon-01 login:
```

To set the local time zone and domain name do:

```
zeus@rtr-lon-01:~$ configure
[edit]
zeus@rtr-lon-01:~# set system time-zone Europe/London
[edit]
zeus@rtr-lon-01:~# commit
[ system time-zone Europe/London ]
Stopping enhanced syslogd: rsyslogd.
Starting enhanced syslogd: rsyslogd.

[edit]
zeus@rtr-lon-01:~# set system domain-name saasify.com
[edit]
zeus@rtr-lon-01:~# save
Saving configuration to '/config/config.boot' ...
Done
```



Initial “hardening” tasks continued ...

Finally the Ethernet interfaces and the random SSH port:

```
zeus@rtr-lon-01:~# set interfaces ethernet eth0 address 159.10.12.97/26
[edit]
zeus@rtr-lon-01:~# set interfaces ethernet eth1 address 10.1.9.2/26
[edit]
zeus@rtr-lon-01:~# set service ssh
[edit]
zeus@rtr-lon-01:~# set service ssh port 22272
[edit]
zeus@rtr-lon-01:~# commit
[ service ssh ]
Restarting OpenBSD Secure Shell server: sshd

[edit]
zeus@rtr-lon-01:~# save
Saving configuration to '/config/config.boot' ...
Done
```

If you are building your Vbox machine as part of a broader GNS3 design then it can be difficult to check this ssh change directly. However, at this stage (given we’ve only juts begun) you could just power the machine up from Vbox itself with two private Adapters and you should be able to check ssh is working as expected on port 22272. Here’s a quick picture from my own implementation showing refusal on port 22 and connectivity on 22272.

```
Eamonns-iMac:~ saasify$ ssh -p 22272 zeus@159.10.12.97
The authenticity of host '[159.10.12.97]:22272 ([159.10.12.97]:22272)' can't be established.
RSA key fingerprint is 95:75:29:1e:ce:59:1c:1a:4f:c8:15:b5:64:5e:8f:80.
Are you sure you want to continue connecting (yes/no)? yes
Warning: Permanently added '[159.10.12.97]:22272' (RSA) to the list of known hosts.
Welcome to VyOS
zeus@159.10.12.97's password:
Linux rtr-01-vyos 3.13.11-1-amd64-vyos #1 SMP Wed Aug 12 02:08:05 UTC 2015 x86_64
Welcome to VyOS.
This system is open-source software. The exact distribution terms for
each module comprising the full system are described in the individual
files in /usr/share/doc/*/copyright.
Last login: Mon Sep 21 10:21:42 2015
zeus@rtr-01-vyos:~$ exit
logout
Connection to 159.10.12.97 closed.
Eamonns-iMac:~ saasify$ ssh zeus@159.10.12.97
ssh: connect to host 159.10.12.97 port 22: Connection refused
Eamonns-iMac:~ saasify$ █
```



Baseline or default settings

A number of default settings are recommended for Vyatta/VyOS in particular:

- set firewall all-ping 'enable'
- set firewall broadcast-ping 'disable'
- set firewall ipv6-receive-redirects 'disable'
- set firewall receive-redirects 'disable'
- set firewall ipv6-src-route 'disable'
- set firewall ip-src-route 'disable'
- set firewall log-martians 'enable'
- set firewall send-redirects 'enable'
- set firewall source-validation 'disable'
- set firewall syn-cookies 'enable'
- Set firewall config-trap 'disable'

So what do these lines do? Why are they set this way?

All Ping:

While reconnaissance is feasible with ping enabled it is not generally considered best or leading practice to disable ping entirely. A more advanced approach would be to set firewall rules specifically allowing certain machines or source subnets to send pings and illicit responses. By default however the recommendation is to enable pinging. If you 'disable' then no ping responses at all.

Broadcast Pings:

Broadcast pings provide an ability for people to ping the entire subnet and based on the responses they can gain a picture of what exists on that subnet. This can be used to develop a list of what exists on a subnet with a view to planning an attack against those hosts. To stop this by default we disable broadcast pinging.

Receive Redirects & IPv6 Receive Redirects:

By default these options are disabled. But what does it do? Well the purpose of an ICMP Type 5 message (which this is) is for network gateways or routers to notify hosts of a shorter path to a destination. When a host receives an ICMP redirect message it then updates its own routing table based on the redirect message input. This means ICMP redirect messages could be used for denial-of-service attacks. Therefore it is recommended to leave these disabled.

Ip-src-route & ipv6-src-route:

By default both options are disabled. This means the vRouter will drop IPv4 and IPv6 packets that contain source route information in their IP headers. If the vRouter is allowed to process IP headers source route information this can override the standard destination based route processing. This in turn presents an opportunity to 'spoof' the system with an attacker able to redirect responses to a spoofed destination addresses back to itself.

[This Font](#) = next page



Baseline or default settings ... continued

A number of default settings are recommended for Vyatta/VyOS in particular:

- `set firewall all-ping 'enable'`
- `set firewall broadcast-ping 'disable'`
- `set firewall ipv6-receive-redirects 'disable'`
- `set firewall receive-redirects 'disable'`
- `set firewall ipv6-src-route 'disable'`
- `set firewall ip-src-route 'disable'`
- `set firewall log-martians 'enable'`
- `set firewall send-redirects 'enable'`
- `set firewall source-validation 'disable'`
- `set firewall syn-cookies 'enable'`
- `set firewall config-trap 'enable'`

So what do these lines do? Why are they set this way?

Log Martians:

By default this is enabled. When it is the vRouter will generate a log message entry when it receives a packet that has come with an invalid source or destination address. These types of packets are often generated by denial-of-service or spoofing attackers. You can also get martians with Layer 2 misconfigurations where two interfaces are connected to the same VLAN.

Send Redirects:

By default this is enabled. It means that the vRouter will redirect ICMP Type 5 messages to directly connected hosts under certain conditions, namely:

- Packet received and routed out the same interface;
- The source address of a packet is in the same subnet as the next hop for the packets destination

Source Validation:

By default this option is disabled. But what does it do? Well the purpose of source-validation relates to reverse path filtering which in turn is the process of validating whether or not the source address of a packet is expected on the interface it arrived in on. For some customers i.e. ISPs this would likely be enabled. This can be used to stop spoofing attacks.

Syn-Cookies:

By default this is enabled. This means the vRouter will use a TCP SYN cookie to mitigate (potentially stop) SYN flood attacks by decreasing the likelihood of overloading a TCP connection queue.

Config-trap:

By default this is disabled. This means the vRouter will not log configuration changes. For better accounting it is best practice to enable this option.

Setting the baselines

To set the baselines we issue the following commands:

```
zeus@rtr-lon-01:~$ configure
[edit]
zeus@rtr-lon-01:~# set firewall all-ping enable
[edit]
zeus@rtr-lon-01:~# set firewall broadcast-ping disable
[edit]
zeus@rtr-lon-01:~# set firewall ipv6-receive-redirects disable
[edit]
zeus@rtr-lon-01:~# set firewall receive-redirects disable
[edit]
zeus@rtr-lon-01:~# set firewall ipv6-src-route disable
[edit]
zeus@rtr-lon-01:~# set firewall ip-src-route disable
[edit]
zeus@rtr-lon-01:~# set firewall log-martians enable
[edit]
zeus@rtr-lon-01:~# set firewall send-redirects enable
[edit]
zeus@rtr-lon-01:~# set firewall source-validation disable
[edit]
zeus@rtr-lon-01:~# set firewall syn-cookies enable
[edit]
zeus@rtr-lon-01:~# set firewall config-trap enable
[edit]
zeus@rtr-lon-01:~# commit
[edit]
zeus@rtr-lon-01:~# save
Saving configuration to '/config/config.boot' ...
Done
```



Global state policies & ARP

Statefulness:

By default your Vyatta/VyOS vRouter firewall is stateless. Stateless firewalls filter packets in isolation based on source and destination rules. Another possibility is to configure the vrouter as “stateful” which means the firewall will track whether a connection state is known and therefore authorized. For example host FRED initiates a conversation with host WILMA then WILMA's response traffic to FRED is allowed automatically in a stateful situation.

On your vRouter you can:

1. Leave it stateless and set up stateful rules per rule set;
2. Enable global stateful behavior through Global State Policies

Global State Policies:

A Vyatta can have globally configured stateful behaviors. To do this we set global state policies which can obviate the need for you to set up specific return traffic rules. Global state policies apply to IPv4 and IPv6 traffic:

- Destined for;
- Originating from; or
- Traversing

the vRouter.

ARP Table Size:

One final consideration is the ARP table size. If you are implementing your Vyatta/VyOS within a large environment then you may want to consider a larger ARP table space.

```
zeus@rtr-lon-01:~# set firewall state-policy established action 'accept'
[edit]
zeus@rtr-lon-01:~# set firewall state-policy related action 'accept'
[edit]
zeus@rtr-lon-01:~# set system ip arp table-size 32768
[edit]
zeus@rtr-lon-01:~# commit
[edit]
zeus@rtr-lon-01:~# save
Saving configuration to '/config/config.boot' ...
Done
```

Remember: Any global state policies can be over-ridden within a specific interfaces firewall rules! So to set up the global state policies you can issue the following commands:

Netfilter & Conntrack

At this stage its probably worth mentioning what Netfilter & Conntrack are.

Netfilter:

Netfilter is the part of Linux that delivers lots of different network related functions and capability. To really understand Netfilter is beyond the scope of this quick introduction so for further reading please see:

- <http://www.netfilter.org/documentation/> (very deep)
- <https://en.wikipedia.org/wiki/Netfilter>
- <http://www.netfilter.org/documentation/HOWTO/netfilter-hacking-HOWTO-1.html>

Conntrack:

In order to enable the capabilities required to deliver stateful firewalling there needs to be some means to track each connection to the vRouter. Simply stated this is what the "conntrack" module does. Basically, conntrack stores information about the state of a connection in a memory structure that contains the source and destination IP addresses, port number pairs, protocol types, state, and timeout. With this extra information available our Vyatta/VyOS can be configured to deliver much richer & intelligent filtering policies.

Once you have enabled the global state policies AND created a firewall with a rule you can examine the conntrack status:

```
zeus@rtr-lon-01:~# set firewall name OUTSIDE-2-LOCAL default-action drop
[edit]
zeus@rtr-lon-01:~# commit
[edit]
zeus@rtr-lon-01:~# save
Saving configuration to '/config/config.boot' ...
Done
zeus@rtr-lon-01:~# exit
zeus@rtr-lon-01:~$ show conntrack table ipv4
TCP state codes:  SS - SYN SENT, SR - SYN RECEIVED, ES - ESTABLISHED,
                  FW - FIN WAIT, CW - CLOSE WAIT, LA - LAST ACK,
                  TW - TIME WAIT, CL - CLOSE, LI - LISTEN

CONN  ID      Source                Destination            Protocol  TIMEOUT
```

Conntrack baselines

Depending on the amount of traffic you are expecting you may need to adjust the conntrack default values or you may want to ensure the defaults are set. The defaults are:

- Conntrack expect-table-size: the expect table is used to store expected or related connections. In standard operating mode entries should graduate quickly to the main table but this can fill up. The default value is 2048 but 8192 is not an unusual setting for heavy use;
- Conntrack table-size: this is the actual conntrack table itself. The default value of 262144 is considered low and best practice from Vyatta is 1048567 unless the vRouter is memory constrained;
- Conntrack hash-size: this is the hash table used to manage actions such as lookups performed against the conntrack table. The default is 4096 but (at time of writing) best practice is to increase to 131070;
- Vyatta also recommend disabling all the conntrack helper modules unless absolutely required;
- Finally make sure you have conntrack logging disabled and never use it on production vRouters.

To do this you can issue the following commands:

```
zeus@rtr-lon-01:~# set system conntrack expect-table-size 8192
[edit]
zeus@rtr-lon-01:~# set system conntrack table-size 1048567
[edit]
zeus@rtr-lon-01:~# set system conntrack hash-size 131070
[edit]
zeus@rtr-lon-01:~# set system conntrack modules ftp disable
[edit]
zeus@rtr-lon-01:~# set system conntrack modules gre disable
[edit]
zeus@rtr-lon-01:~# set system conntrack modules h323 disable
[edit]
zeus@rtr-lon-01:~# set system conntrack modules nfs disable
[edit]
zeus@rtr-lon-01:~# set system conntrack modules pptp disable
[edit]
zeus@rtr-lon-01:~# set system conntrack modules sip disable
[edit]
zeus@rtr-lon-01:~# set system conntrack modules sqlnet disable
[edit]
zeus@rtr-lon-01:~# set system conntrack modules tftpd disable
[edit]
zeus@rtr-lon-01:~# commit
[edit]
zeus@rtr-lon-01:~# save
Saving configuration to '/config/config.boot' ...
Done
```



NTP

Given we will be closely knitting these vRouters together – in London with high availability pair and in Paris for disaster recovery we need to ensure each vRouter is time synchronized. To do this within SoftLayer we issue the following command:

```
zeus@rtr-lon-01:~# set system ntp server time.service.networklayer.com
[edit]
zeus@rtr-lon-01:~# commit
[edit]
zeus@rtr-lon-01:~# save
Saving configuration to '/config/config.boot' ...
Done
```



Gateway

We can now also set the gateway address for this vRouter:

```
zeus@rtr-lon-01:~# set system gateway-address 159.10.12.97
[edit]
zeus@rtr-lon-01:~# commit
[edit]
zeus@rtr-lon-01:~# save
Saving configuration to '/config/config.boot' ...
Done
```



DNS & Name Servers

We can now also set the DNS resolver addresses for this vRouter:

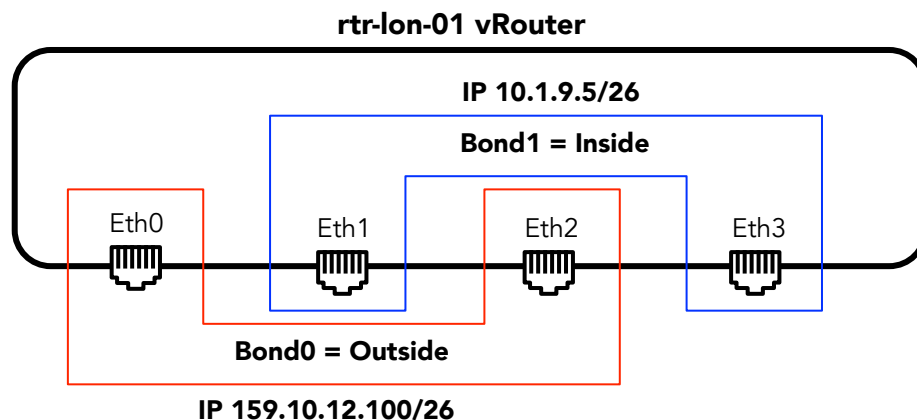
```
zeus@rtr-lon-01:~# set system name-server 10.0.80.11
[edit]
zeus@rtr-lon-01:~# set system name-server 10.0.80.12
[edit]
zeus@rtr-lon-01:~# commit
[edit]
zeus@rtr-lon-01:~# save
Saving configuration to '/config/config.boot' ...
Done
```

 **The bonding implementation**



Interface high availability with bonding

If you wish it is possible to combine one or more physical NIC interfaces into single logical interface. This is generally called bonding but other names exist like LAG, etherchannel, or portchannel. It is always easier to try to visualize what we mean by this and what advantages it provides.



- Four Physical Ethernet Interfaces
- Two Logical Bonded Interfaces

We will create a bonding between the physical interfaces Eth0 and Eth2. The bond will be named Bond0 and it will have an associated IP address on the Public network that is shared between the interfaces. SoftLayer supports 802.3ad which is Dynamic Link Aggregation but we cannot use this on Vbox so we will utilize "active-backup" in our lab.

We will create a bonding between the physical interfaces Eth1 and Eth3. The bond will be named Bond1 and it will have an associated IP address on the Private network that is shared between the interfaces.

The bonding works such that if there is a physical interface failure on the Vyatta the other interface takes the strain yielding highly available physical interfaces. In this picture our firewall and routing services will be delivered on the shared virtual IP addresses assigned to the Public and Private networks as Bond0 and Bond1 respectively.



Implementing the Bonds (Public on rtr-lon-01)

Earlier (Slide #28) we implemented IP addresses on Eth0 and Eth1 in order to configure SSH. To properly implement our requirement for bonding we will delete the previous Ethernet configuration and implement bonding. To do this for the public interfaces we issue the following commands:

```
zeus@rtr-lon-01:~$ configure
[edit]
zeus@rtr-lon-01:~# delete interfaces ethernet eth0 address 159.10.12.97/26
[edit]
zeus@rtr-lon-01:~# delete interfaces ethernet eth0 address 10.1.9.2/26
[edit]
zeus@rtr-lon-01:~# set interfaces bonding bond0 address 159.10.12.100/26
[edit]
zeus@rtr-lon-01:~# set interfaces bonding bond0 description "Public Internet"
[edit]
zeus@rtr-lon-01:~# set interfaces bonding bond0 hash-policy layer3+4
[edit]
zeus@rtr-lon-01:~# set interfaces bonding bond0 mode active-backup
[edit]
zeus@rtr-lon-01:~# set interfaces ethernet eth0 bond-group bond0
[edit]
zeus@rtr-lon-01:~# set interfaces ethernet eth0 smp_affinity auto
[edit]
zeus@rtr-lon-01:~# set interfaces ethernet eth2 bond-group bond0
[edit]
zeus@rtr-lon-01:~# set interfaces ethernet eth2 smp_affinity auto
[edit]
zeus@rtr-lon-01:~# commit
[edit]
zeus@rtr-lon-01:~# save
Saving configuration to '/config/config.boot' ...
Done
```

Note: If you are using a SoftLayer machine this set of commands will likely kill your SSH session so you will need to log in to the Vyatta device again after the commit and then issue the save.



Implementing the Bonds (Public on rtr-lon-02)

Earlier (Slide #28) we implemented IP addresses on Eth0 and Eth1 in order to configure SSH. To properly implement our requirement for bonding we will delete the previous Ethernet configuration and implement bonding. To do this for the public interfaces we issue the following commands:

```
zeus@rtr-lon-02:~$ configure
[edit]
zeus@rtr-lon-02:~# set interfaces bonding bond0 address 159.10.12.101/26
[edit]
zeus@rtr-lon-02:~# set interfaces bonding bond0 description "Public Internet"
[edit]
zeus@rtr-lon-02:~# set interfaces bonding bond0 hash-policy layer3+4
[edit]
zeus@rtr-lon-02:~# set interfaces bonding bond0 mode active-backup
[edit]
zeus@rtr-lon-02:~# set interfaces ethernet eth0 bond-group bond0
[edit]
zeus@rtr-lon-02:~# set interfaces ethernet eth0 smp_affinity auto
[edit]
zeus@rtr-lon-02:~# set interfaces ethernet eth2 bond-group bond0
[edit]
zeus@rtr-lon-02:~# set interfaces ethernet eth2 smp_affinity auto
[edit]
zeus@rtr-lon-02:~# commit
[edit]
zeus@rtr-lon-02:~# save
Saving configuration to '/config/config.boot' ...
Done
```

Note: If you are using a SoftLayer machine this set of commands will likely kill your SSH session so you will need to log in to the Vyatta device again after the commit and then issue the save.



Implementing the Bonds - Private (rtr-lon-01)

Earlier (Slide #28) we implemented IP addresses on Eth0 and Eth1 in order to configure SSH. To properly implement our requirement for bonding we have deleted the previous Ethernet configuration and implemented bonding for the Public. To do this for the private interfaces on rtr-lon-01 we issue the following commands:

```
zeus@rtr-lon-01:~$ configure
[edit]
zeus@rtr-lon-01:~# set interfaces bonding bond1 address 10.1.9.5/26
[edit]
zeus@rtr-lon-01:~# set interfaces bonding bond1 description "Private Internet"
[edit]
zeus@rtr-lon-01:~# set interfaces bonding bond1 hash-policy layer3+4
[edit]
zeus@rtr-lon-01:~# set interfaces bonding bond1 mode active-backup
[edit]
zeus@rtr-lon-01:~# set interfaces ethernet eth1 bond-group bond1
[edit]
zeus@rtr-lon-01:~# set interfaces ethernet eth1 smp_affinity auto
[edit]
zeus@rtr-lon-01:~# set interfaces ethernet eth3 bond-group bond1
[edit]
zeus@rtr-lon-01:~# set interfaces ethernet eth3 smp_affinity auto
[edit]
zeus@rtr-lon-01:~# commit
[edit]
zeus@rtr-lon-01:~# save
Saving configuration to '/config/config.boot' ...
Done
```



Implementing the Bonds - Private (rtr-lon-02)

To do this for the private interfaces on rtr-lon-02 we issue the following commands:

```
zeus@rtr-lon-02:~$ configure
[edit]
zeus@rtr-lon-02:~# set interfaces bonding bond1 address 10.1.9.6/26
[edit]
zeus@rtr-lon-02:~# set interfaces bonding bond1 description "Private Internet"
[edit]
zeus@rtr-lon-02:~# set interfaces bonding bond1 hash-policy layer3+4
[edit]
zeus@rtr-lon-02:~# set interfaces bonding bond1 mode active-backup
[edit]
zeus@rtr-lon-02:~# set interfaces ethernet eth1 bond-group bond1
[edit]
zeus@rtr-lon-02:~# set interfaces ethernet eth1 smp_affinity auto
[edit]
zeus@rtr-lon-02:~# set interfaces ethernet eth3 bond-group bond1
[edit]
zeus@rtr-lon-02:~# set interfaces ethernet eth3 smp_affinity auto
[edit]
zeus@rtr-lon-02:~# commit
[edit]
zeus@rtr-lon-02:~# save
Saving configuration to '/config/config.boot' ...
Done
```



Reviewing the Bonds on rtr-lon-01

We can see our interfaces by doing:

```
zeus@rtr-lon-01:~# commit
[edit]
zeus@rtr-lon-01:~# save
Saving configuration to '/config/config.boot' ...
Done
zeus@rtr-lon-01:~# run show interfaces
Codes:      S - State, L - Link, u - Up, d - Down, A - Admin Down
Interface    IP Address      S/L  Description
-----
bond0        159.10.12.100/26 u/u   Public Internet
bond1        10.1.9.5/26     u/u   Private Intranet
eth0         -               u/u
eth1         -               u/u
eth2         -               u/u
eth3         -               u/u
lo           127.0.0.1/8     u/u
```

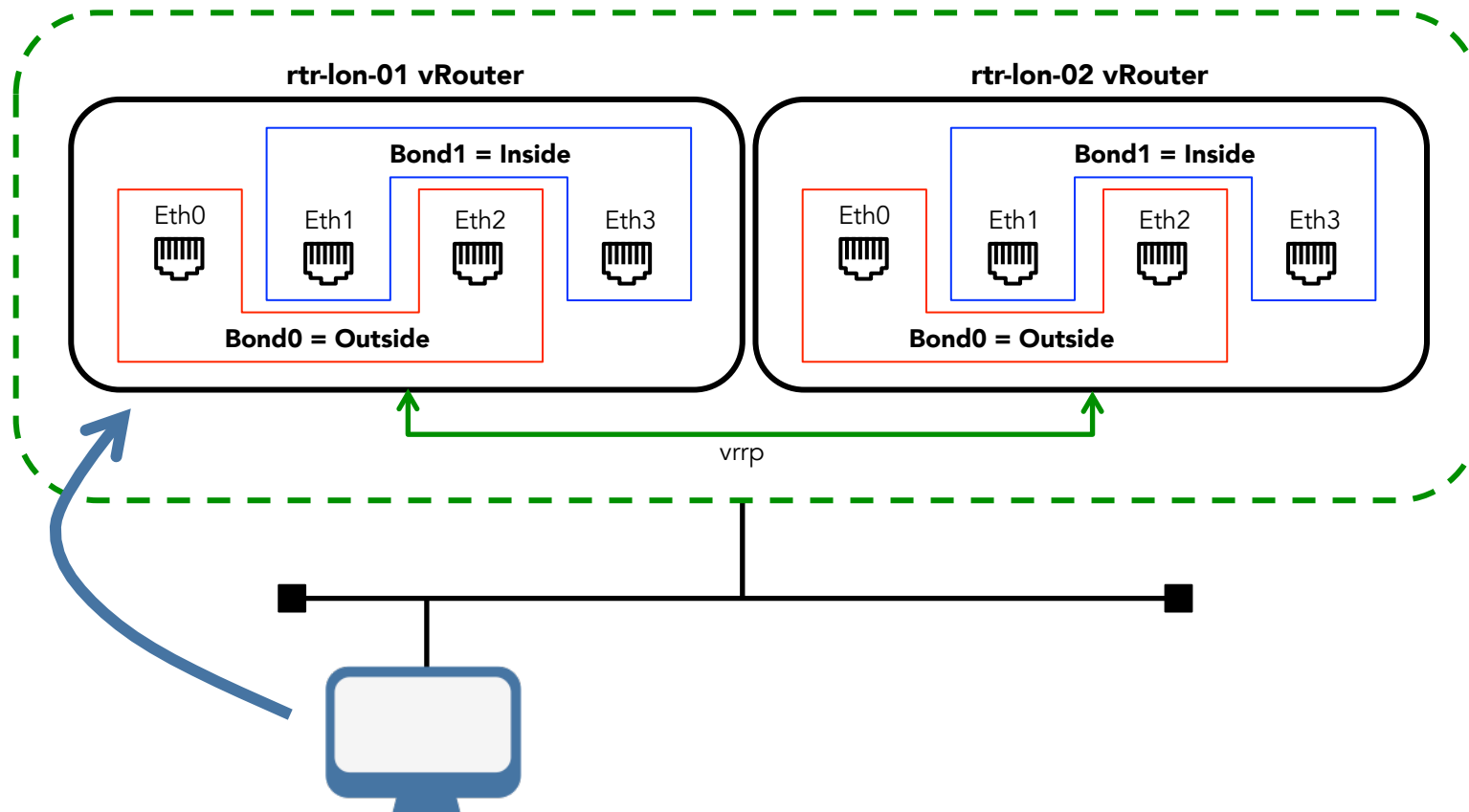
 **The high availability implementation**



Implementing VRRP for the London Vyatta's

Not only can we use bonding for the physical interfaces on the vRouter machines – we can also group the vRouters into high availability pairs using VRRP (Virtual Router Redundancy Protocol). This means the pair of machines will act as one machine for all the users of its services. Pictorially we can visualize this as:

Shared Virtual Addresses for the HA Pair



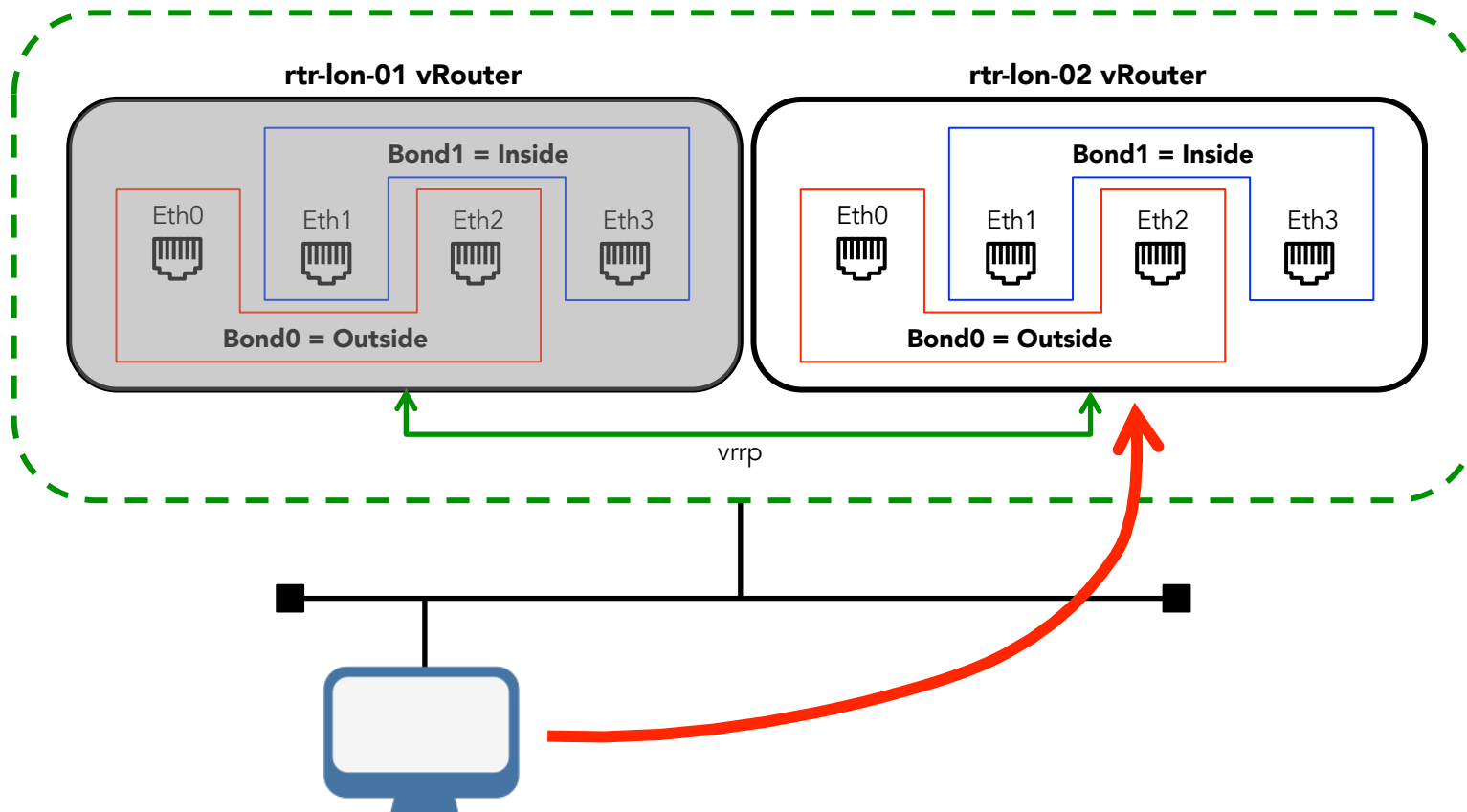
→ Logical flow if rtr-lon-01 is up



If we lose one of London Vyatta's?

With VRRP configured and by using a single shared virtual IP address for Bond0 and Bond1 on the vRouter if we lose one of London's Vyatta's our machines will still have firewall and routing from the secondary (or backup) vRouter.

Shared Virtual Addresses for the HA Pair



→ Logical flow if rtr-lon-01 is down



Setting up VRRP for Bond0 on rtr-lon-01

First we need to ascertain what address to use for the shared virtual IP address that will failover between the machines. These are the HSRP addresses in your IP subnet ranges that are marked as “Reserved”. So from (Slide #20) above we can see:

```
VLAN 1231: 159.10.12.96/26
Gateway: 159.10.12.97/26
HSRP1: 159.10.12.98/26
HSRP2: 159.10.12.99/26
rtr-lon-01 assigned on creation: 159.10.12.100/26
rtr-lon-02 assigned on creation: 159.10.12.101/26
rtr-lon-01 assigned on creation: 159.10.12.102/26
rtr-lon-02 assigned on creation: 159.10.12.103/26
```

We know we have used 159.10.12.100 as the primary IP for Bond0 on rtr-lon-01 and 159.10.12.101 for Bond0 on rtr-lon-02. We can see a reserved HSRP address of 159.10.12.98/26 and this can be used for the VRRP virtual service address shared across both vRouters. The following commands set up high availability for the main bonded interface Bond0 across the two devices rtr-lon-01 and rtr-lon-02:

```
zeus@rtr-lon-01:~# set interfaces bonding bond0 vrrp vrrp-group 1 advertise-interval 1
[edit]
zeus@rtr-lon-01:~# set interfaces bonding bond0 vrrp vrrp-group 1 preempt false
[edit]
zeus@rtr-lon-01:~# set interfaces bonding bond0 vrrp vrrp-group 1 priority 254
[edit]
zeus@rtr-lon-01:~# set interfaces bonding bond0 vrrp vrrp-group 1 rfc3768-compatibility
[edit]
zeus@rtr-lon-01:~# set interfaces bonding bond0 vrrp vrrp-group 1 sync-group vgroup1
[edit]
zeus@rtr-lon-01:~# set interfaces bonding bond0 vrrp vrrp-group 1 virtual-address 159.10.12.98/26
[edit]
zeus@rtr-lon-01:~# commit
[edit]
zeus@rtr-lon-01:~# save
Saving configuration to '/config/config.boot' ...
Done
```



Setting up VRRP for Bond1 on rtr-lon-01

First we need to ascertain what address to use for the shared virtual IP address that will failover between the machines. These are the HSRP addresses in your IP subnet ranges that are marked as “Reserved”. So from (Slide #20) above we can see:

```
VLAN 1232: 10.1.9.1/26
Gateway: 10.1.9.2/26
HSRP1: 10.1.9.3/26
HSRP2: 10.1.9.4/26
rtr-lon-01 assigned on creation: 10.1.9.5/26
rtr-lon-02 assigned on creation: 10.1.9.6/26
rtr-lon-01 assigned on creation: 10.1.9.7/26
rtr-lon-02 assigned on creation: 10.1.9.8/26
```

We know we have used 10.1.9.5 as the primary IP for Bond1 on rtr-lon-01 and 10.1.9.6 for Bond1 on rtr-lon-02. We can see a reserved HSRP address of 159.10.12.98/26 and this can be used for the VRRP virtual service address shared across both vRouters. The following commands set up high availability for the main bonded interface Bond1 across the two devices rtr-lon-01 and rtr-lon-02:

```
zeus@rtr-lon-01:~# set interfaces bonding bond1 vrrp vrrp-group 1 advertise-interval 1
[edit]
zeus@rtr-lon-01:~# set interfaces bonding bond1 vrrp vrrp-group 1 preempt false
[edit]
zeus@rtr-lon-01:~# set interfaces bonding bond1 vrrp vrrp-group 1 priority 254
[edit]
zeus@rtr-lon-01:~# set interfaces bonding bond1 vrrp vrrp-group 1 rfc3768-compatibility
[edit]
zeus@rtr-lon-01:~# set interfaces bonding bond1 vrrp vrrp-group 1 sync-group vgroup1
[edit]
zeus@rtr-lon-01:~# set interfaces bonding bond1 vrrp vrrp-group 1 virtual-address 10.9.1.3/26
[edit]
zeus@rtr-lon-01:~# commit
[edit]
zeus@rtr-lon-01:~# save
Saving configuration to '/config/config.boot' ...
Done
```



Setting up VRRP for Bond0 & 1 on rtr-lon-02

Continuing the VRRP set up across rtr-lon-01 and rtr-lon-02 by implementing the VRRP group on rtr-lon-02:

```
zeus@rtr-lon-02:~# set interfaces bonding bond0 vrrp vrrp-group 1 advertise-interval 1
[edit]
zeus@rtr-lon-02:~# set interfaces bonding bond0 vrrp vrrp-group 1 preempt false
[edit]
zeus@rtr-lon-02:~# set interfaces bonding bond0 vrrp vrrp-group 1 priority 253
[edit]
zeus@rtr-lon-02:~# set interfaces bonding bond0 vrrp vrrp-group 1 rfc3768-compatibility
[edit]
zeus@rtr-lon-02:~# set interfaces bonding bond0 vrrp vrrp-group 1 sync-group vgroup1
[edit]
zeus@rtr-lon-02:~# set interfaces bonding bond0 vrrp vrrp-group 1 virtual-address 159.10.12.98/26
[edit]
zeus@rtr-lon-02:~# set interfaces bonding bond1 vrrp vrrp-group 1 advertise-interval 1
[edit]
zeus@rtr-lon-02:~# set interfaces bonding bond1 vrrp vrrp-group 1 preempt false
[edit]
zeus@rtr-lon-02:~# set interfaces bonding bond1 vrrp vrrp-group 1 priority 253
[edit]
zeus@rtr-lon-02:~# set interfaces bonding bond1 vrrp vrrp-group 1 rfc3768-compatibility
[edit]
zeus@rtr-lon-02:~# set interfaces bonding bond1 vrrp vrrp-group 1 sync-group vgroup1
[edit]
zeus@rtr-lon-02:~# set interfaces bonding bond1 vrrp vrrp-group 1 virtual-address 10.9.1.3/26
[edit]
zeus@rtr-lon-02:~# commit
[edit]
zeus@rtr-lon-02:~# save
Saving configuration to '/config/config.boot' ...
Done
```



Reviewing the VRRP configuration on rtr-lon-01

We can now review our VRRP set up using the following commands on rtr-lon-01:

```
zeus@rtr-lon-01:~$ show interfaces
```

Codes: S – State, L – Link, u – Up, d – Down, A – Admin Down

Interface	IP Address	S/L	Description
-----	-----	---	-----
bond0	159.10.12.100/26	u/u	Public Internet
bond0v1	159.10.12.98/26	u/u	
bond1	10.1.9.5/26	u/u	Private Intranet
bond1v1	10.1.9.3/26	u/u	
eth0	-	u/u	
eth1	-	u/u	
eth2	-	u/u	
eth3	-	u/u	
lo	127.0.0.1/8	u/u	

```
zeus@rtr-lon-01:~$ show vrrp
```

Interface	Group	State	RFC Complaint	Addr Owner	Last Transition	Sync Group
-----	-----	-----	-----	-----	-----	-----
bond0	1	MASTER	yes	no	40m5s	vgroup1
bond1	1	MASTER	yes	no	40m5s	vgroup1



Reviewing the VRRP configuration on rtr-lon-02

We can now review our VRRP set up using the following commands on rtr-lon-02:

```
zeus@rtr-lon-02:~$ show interfaces
```

Codes: S – State, L – Link, u – Up, d – Down, A – Admin Down

Interface	IP Address	S/L	Description
-----	-----	---	-----
bond0	159.10.12.101/26	u/u	Public Internet
bond0v1	-	A/D	
bond1	10.1.9.5/26	u/u	Private Intranet
bond1v1	-	A/D	
eth0	-	u/u	
eth1	-	u/u	
eth2	-	u/u	
eth3	-	u/u	
lo	127.0.0.1/8	u/u	

```
zeus@rtr-lon-02:~$ show vrrp
```

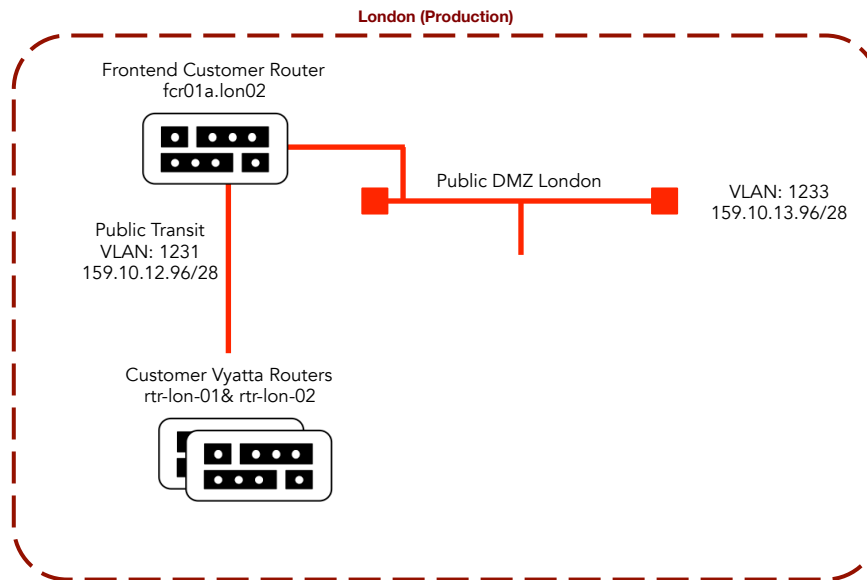
Interface	Group	State	RFC Complaint	Addr Owner	Last Transition	Sync Group
-----	-----	-----	-----	-----	-----	-----
bond0	1	BACKUP	yes	no	40m5s	vgroup1
bond1	1	BACKUP	yes	no	40m5s	vgroup1

The VLANs implementation



Public DMZ - VLAN 1233

Reviewing our network design – we can see that traffic flows on the Public DMZ VLAN 1233 is not traversing our vRouter so there is no need to add any entries for this network.





Private DMZ - VLAN 1238 VRRP rtr-lon-01

We can now add all of our additional VLANs as a series of virtual interfaces attached to our bonds. First up is VLAN 1238 with subnet range 10.1.14.1/26 on our private bond i.e. Bond1. The key information we need is:

Gateway: 10.1.14.2/26

HSRP1: 10.1.14.3/26

HSRP2: 10.1.14.4/26

To set up this VLAN we issue these commands on rtr-lon-01:

```
zeus@rtr-lon-01:~# set interfaces bonding bond1 vif 1238 address 10.1.14.3/26
[edit]
zeus@rtr-lon-01:~# set interfaces bonding bond1 vif 1238 description "Private DMZ"
[edit]
zeus@rtr-lon-01:~# set interfaces bonding bond1 vif 1238 vrrp vrrp-group 1 advertise-interval 1
[edit]
zeus@rtr-lon-01:~# set interfaces bonding bond1 vif 1238 vrrp vrrp-group 1 preempt false
[edit]
zeus@rtr-lon-01:~# set interfaces bonding bond1 vif 1238 vrrp vrrp-group 1 priority 254
[edit]
zeus@rtr-lon-01:~# set interfaces bonding bond1 vif 1238 vrrp vrrp-group 1 rfc3768-compatibility
[edit]
zeus@rtr-lon-01:~# set interfaces bonding bond1 vif 1238 vrrp vrrp-group 1 sync-group vgroup1
[edit]
zeus@rtr-lon-01:~# set interfaces bonding bond1 vif 1238 vrrp vrrp-group 1 virtual-address 10.1.14.2/26
[edit]
zeus@rtr-lon-01:~# commit
[edit]
zeus@rtr-lon-01:~# save
Saving configuration to '/config/config.boot' ...
Done
```



Private DMZ - VLAN 1238 VRRP rtr-lon-02

To set up this VLAN we issue these commands on rtr-lon-02:

```
zeus@rtr-lon-02:~# set interfaces bonding bond1 vif 1238 address 10.1.14.4/26
[edit]
zeus@rtr-lon-02:~# set interfaces bonding bond1 vif 1238 description "Private DMZ"
[edit]
zeus@rtr-lon-02:~# set interfaces bonding bond1 vif 1238 vrrp vrrp-group 1 advertise-interval 1
[edit]
zeus@rtr-lon-02:~# set interfaces bonding bond1 vif 1238 vrrp vrrp-group 1 preempt false
[edit]
zeus@rtr-lon-02:~# set interfaces bonding bond1 vif 1238 vrrp vrrp-group 1 priority 253
[edit]
zeus@rtr-lon-02:~# set interfaces bonding bond1 vif 1238 vrrp vrrp-group 1 rfc3768-compatibility
[edit]
zeus@rtr-lon-02:~# set interfaces bonding bond1 vif 1238 vrrp vrrp-group 1 sync-group vgroup1
[edit]
zeus@rtr-lon-02:~# set interfaces bonding bond1 vif 1238 vrrp vrrp-group 1 virtual-address 10.1.14.2/26
[edit]
zeus@rtr-lon-02:~# commit
[edit]
zeus@rtr-lon-02:~# save
Saving configuration to '/config/config.boot' ...
Done
```



Private Web - VLAN 1234 VRRP rtr-lon-01

Second up is VLAN 1234 with subnet range 10.1.10.1/26. Remember:

Gateway: 10.1.10.2/26

HSRP1: 10.1.10.3/26

HSRP2: 10.1.10.4/26

To set up this VLAN we issue these commands on rtr-lon-01:

```
zeus@rtr-lon-01:~# set interfaces bonding bond1 vif 1234 address 10.1.10.3/26
[edit]
zeus@rtr-lon-01:~# set interfaces bonding bond1 vif 1234 description "Private Web"
[edit]
zeus@rtr-lon-01:~# set interfaces bonding bond1 vif 1234 vrrp vrrp-group 1 advertise-interval 1
[edit]
zeus@rtr-lon-01:~# set interfaces bonding bond1 vif 1234 vrrp vrrp-group 1 preempt false
[edit]
zeus@rtr-lon-01:~# set interfaces bonding bond1 vif 1234 vrrp vrrp-group 1 priority 254
[edit]
zeus@rtr-lon-01:~# set interfaces bonding bond1 vif 1234 vrrp vrrp-group 1 rfc3768-compatibility
[edit]
zeus@rtr-lon-01:~# set interfaces bonding bond1 vif 1234 vrrp vrrp-group 1 sync-group vgroup1
[edit]
zeus@rtr-lon-01:~# set interfaces bonding bond1 vif 1234 vrrp vrrp-group 1 virtual-address 10.1.10.2/26
[edit]
zeus@rtr-lon-01:~# commit
[edit]
zeus@rtr-lon-01:~# save
Saving configuration to '/config/config.boot' ...
Done
```



Private Web - VLAN 1234 VRRP rtr-lon-02

To set up this VLAN we issue these commands on rtr-lon-02:

```
zeus@rtr-lon-02:~# set interfaces bonding bond1 vif 1234 address 10.1.10.4/26
[edit]
zeus@rtr-lon-02:~# set interfaces bonding bond1 vif 1234 description "Private Web"
[edit]
zeus@rtr-lon-02:~# set interfaces bonding bond1 vif 1234 vrrp vrrp-group 1 advertise-interval 1
[edit]
zeus@rtr-lon-02:~# set interfaces bonding bond1 vif 1234 vrrp vrrp-group 1 preempt false
[edit]
zeus@rtr-lon-02:~# set interfaces bonding bond1 vif 1234 vrrp vrrp-group 1 priority 253
[edit]
zeus@rtr-lon-02:~# set interfaces bonding bond1 vif 1234 vrrp vrrp-group 1 rfc3768-compatibility
[edit]
zeus@rtr-lon-02:~# set interfaces bonding bond1 vif 1234 vrrp vrrp-group 1 sync-group vgroup1
[edit]
zeus@rtr-lon-02:~# set interfaces bonding bond1 vif 1234 vrrp vrrp-group 1 virtual-address 10.1.10.2/26
[edit]
zeus@rtr-lon-02:~# commit
[edit]
zeus@rtr-lon-02:~# save
Saving configuration to '/config/config.boot' ...
Done
```



Private App - VLAN 1235 VRRP rtr-lon-01

Third up is our VLAN 1235 with a subnet range of 10.1.11.1/26. Again remember our key information:

Gateway: 10.1.11.2/26

HSRP1: 10.1.11.3/26

HSRP2: 10.1.11.4/26

To set up this VLAN we issue these commands on rtr-lon-01:

```
zeus@rtr-lon-01:~# set interfaces bonding bond1 vif 1235 address 10.1.11.3/26
[edit]
zeus@rtr-lon-01:~# set interfaces bonding bond1 vif 1235 description "Private App"
[edit]
zeus@rtr-lon-01:~# set interfaces bonding bond1 vif 1235 vrrp vrrp-group 1 advertise-interval 1
[edit]
zeus@rtr-lon-01:~# set interfaces bonding bond1 vif 1235 vrrp vrrp-group 1 preempt false
[edit]
zeus@rtr-lon-01:~# set interfaces bonding bond1 vif 1235 vrrp vrrp-group 1 priority 254
[edit]
zeus@rtr-lon-01:~# set interfaces bonding bond1 vif 1235 vrrp vrrp-group 1 rfc3768-compatibility
[edit]
zeus@rtr-lon-01:~# set interfaces bonding bond1 vif 1235 vrrp vrrp-group 1 sync-group vgroup1
[edit]
zeus@rtr-lon-01:~# set interfaces bonding bond1 vif 1235 vrrp vrrp-group 1 virtual-address 10.1.11.2/26
[edit]
zeus@rtr-lon-01:~# commit
[edit]
zeus@rtr-lon-01:~# save
Saving configuration to '/config/config.boot' ...
Done
```



Private App - VLAN 1235 VRRP rtr-lon-02

To set up this VLAN we issue these commands on rtr-lon-02:

```
zeus@rtr-lon-02:~# set interfaces bonding bond1 vif 1235 address 10.1.11.4/26
[edit]
zeus@rtr-lon-02:~# set interfaces bonding bond1 vif 1235 description "Private App"
[edit]
zeus@rtr-lon-02:~# set interfaces bonding bond1 vif 1235 vrrp vrrp-group 1 advertise-interval 1
[edit]
zeus@rtr-lon-02:~# set interfaces bonding bond1 vif 1235 vrrp vrrp-group 1 preempt false
[edit]
zeus@rtr-lon-02:~# set interfaces bonding bond1 vif 1235 vrrp vrrp-group 1 priority 253
[edit]
zeus@rtr-lon-02:~# set interfaces bonding bond1 vif 1235 vrrp vrrp-group 1 rfc3768-compatibility
[edit]
zeus@rtr-lon-02:~# set interfaces bonding bond1 vif 1235 vrrp vrrp-group 1 sync-group vgroup1
[edit]
zeus@rtr-lon-02:~# set interfaces bonding bond1 vif 1235 vrrp vrrp-group 1 virtual-address 10.1.11.2/26
[edit]
zeus@rtr-lon-02:~# commit
[edit]
zeus@rtr-lon-02:~# save
Saving configuration to '/config/config.boot' ...
Done
```



Private Data - VLAN 1236 VRRP rtr-lon-01

Fourth up is our VLAN 1236 with subnet range 10.1.12.1/26. As with above remember our design and information:

Gateway: 10.1.12.2/26

HSRP1: 10.1.12.3/26

HSRP2: 10.1.12.4/26

To set up this VLAN we issue these commands on rtr-lon-01:

```
zeus@rtr-lon-01:~# set interfaces bonding bond1 vif 1236 address 10.1.12.3/26
[edit]
zeus@rtr-lon-01:~# set interfaces bonding bond1 vif 1236 description "Private Data"
[edit]
zeus@rtr-lon-01:~# set interfaces bonding bond1 vif 1236 vrrp vrrp-group 1 advertise-interval 1
[edit]
zeus@rtr-lon-01:~# set interfaces bonding bond1 vif 1236 vrrp vrrp-group 1 preempt false
[edit]
zeus@rtr-lon-01:~# set interfaces bonding bond1 vif 1236 vrrp vrrp-group 1 priority 254
[edit]
zeus@rtr-lon-01:~# set interfaces bonding bond1 vif 1236 vrrp vrrp-group 1 rfc3768-compatibility
[edit]
zeus@rtr-lon-01:~# set interfaces bonding bond1 vif 1236 vrrp vrrp-group 1 sync-group vgroup1
[edit]
zeus@rtr-lon-01:~# set interfaces bonding bond1 vif 1236 vrrp vrrp-group 1 virtual-address 10.1.12.2/26
[edit]
zeus@rtr-lon-01:~# commit
[edit]
zeus@rtr-lon-01:~# save
Saving configuration to '/config/config.boot' ...
Done
```



Private Data - VLAN 1236 VRRP rtr-lon-02

To set up this VLAN we issue these commands on rtr-lon-02:

```
zeus@rtr-lon-02:~# set interfaces bonding bond1 vif 1236 address 10.1.12.4/26
[edit]
zeus@rtr-lon-02:~# set interfaces bonding bond1 vif 1236 description "Private Data"
[edit]
zeus@rtr-lon-02:~# set interfaces bonding bond1 vif 1236 vrrp vrrp-group 1 advertise-interval 1
[edit]
zeus@rtr-lon-02:~# set interfaces bonding bond1 vif 1236 vrrp vrrp-group 1 preempt false
[edit]
zeus@rtr-lon-02:~# set interfaces bonding bond1 vif 1236 vrrp vrrp-group 1 priority 253
[edit]
zeus@rtr-lon-02:~# set interfaces bonding bond1 vif 1236 vrrp vrrp-group 1 rfc3768-compatibility
[edit]
zeus@rtr-lon-02:~# set interfaces bonding bond1 vif 1236 vrrp vrrp-group 1 sync-group vgroup1
[edit]
zeus@rtr-lon-02:~# set interfaces bonding bond1 vif 1236 vrrp vrrp-group 1 virtual-address 10.1.12.2/26
[edit]
zeus@rtr-lon-02:~# commit
[edit]
zeus@rtr-lon-02:~# save
Saving configuration to '/config/config.boot' ...
Done
```




Private Management - VLAN 1237 VRRP rtr-lon-01

And finally our VLAN 1237 with subnet range 10.1.13.1/26 and key information:

Gateway: 10.1.13.2/26

HSRP1: 10.1.13.3/26

HSRP2: 10.1.13.4/26

To set up this VLAN we issue these commands on rtr-lon-01:

```
zeus@rtr-lon-01:~# set interfaces bonding bond1 vif 1237 address 10.1.13.3/26
[edit]
zeus@rtr-lon-01:~# set interfaces bonding bond1 vif 1237 description "Private Management"
[edit]
zeus@rtr-lon-01:~# set interfaces bonding bond1 vif 1237 vrrp vrrp-group 1 advertise-interval 1
[edit]
zeus@rtr-lon-01:~# set interfaces bonding bond1 vif 1237 vrrp vrrp-group 1 preempt false
[edit]
zeus@rtr-lon-01:~# set interfaces bonding bond1 vif 1237 vrrp vrrp-group 1 priority 254
[edit]
zeus@rtr-lon-01:~# set interfaces bonding bond1 vif 1237 vrrp vrrp-group 1 rfc3768-compatibility
[edit]
zeus@rtr-lon-01:~# set interfaces bonding bond1 vif 1237 vrrp vrrp-group 1 sync-group vgroup1
[edit]
zeus@rtr-lon-01:~# set interfaces bonding bond1 vif 1237 vrrp vrrp-group 1 virtual-address 10.1.13.2/26
[edit]
zeus@rtr-lon-01:~# commit
[edit]
zeus@rtr-lon-01:~# save
Saving configuration to '/config/config.boot' ...
Done
```



Private Management - VLAN 1237 VRRP rtr-lon-02

To set up this VLAN we issue these commands on rtr-lon-02:

```
zeus@rtr-lon-02:~# set interfaces bonding bond1 vif 1237 address 10.1.13.4/26
[edit]
zeus@rtr-lon-02:~# set interfaces bonding bond1 vif 1237 description "Private Management"
[edit]
zeus@rtr-lon-02:~# set interfaces bonding bond1 vif 1237 vrrp vrrp-group 1 advertise-interval 1
[edit]
zeus@rtr-lon-02:~# set interfaces bonding bond1 vif 1237 vrrp vrrp-group 1 preempt false
[edit]
zeus@rtr-lon-02:~# set interfaces bonding bond1 vif 1237 vrrp vrrp-group 1 priority 253
[edit]
zeus@rtr-lon-02:~# set interfaces bonding bond1 vif 1237 vrrp vrrp-group 1 rfc3768-compatibility
[edit]
zeus@rtr-lon-02:~# set interfaces bonding bond1 vif 1237 vrrp vrrp-group 1 sync-group vgroup1
[edit]
zeus@rtr-lon-02:~# set interfaces bonding bond1 vif 1237 vrrp vrrp-group 1 virtual-address 10.1.13.2/26
[edit]
zeus@rtr-lon-02:~# commit
[edit]
zeus@rtr-lon-02:~# save
Saving configuration to '/config/config.boot' ...
Done
```



Reviewing all our VLANs

To review all of our VLANs we can issue the show interfaces command and you should see output like this:

```
zeus@rtr-lon-01:~$ show interfaces
Codes:      S - State, L - Link, u - Up, d - Down, A - Admin Down
Interface    IP Address      S/L  Description
-----
bond0        159.10.12.100/26 u/u   Public Internet
bond0v1      159.10.12.98/26  u/u
bond1        10.1.9.5/26      u/u   Private Intranet
bond1.1234   10.1.14.3/26     u/u   Private Web
bond1.1234v1 10.1.14.2/26     u/u
bond1.1235   10.1.14.3/26     u/u   Private App
bond1.1235v1 10.1.14.2/26     u/u
bond1.1236   10.1.14.3/26     u/u   Private Data
bond1.1236v1 10.1.14.2/26     u/u
bond1.1237   10.1.14.3/26     u/u   Private Management
bond1.1237v1 10.1.14.2/26     u/u
bond1.1238   10.1.14.3/26     u/u   Private DMZ
bond1.1238v1 10.1.14.2/26     u/u
bond1v1      10.1.9.3/26      u/u
eth0         -                u/u
eth1         -                u/u
eth2         -                u/u
eth3         -                u/u
lo           127.0.0.1/8      u/u
zeus@rtr-lon-01:~$
```

 **Final preparation for firewalls**

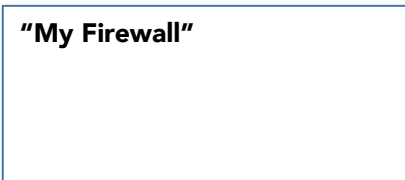


Background

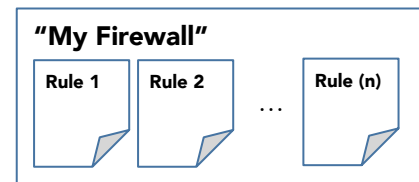
We have touched very lightly on Slide [26](#) above as to the firewall features of a Vyatta/VyOS vRouter. At a lower level of detail the Vyatta/VyOS vRouter:

- Utilizes Netfilter for packet filtering;
 - What does this mean? Well it means that a portion of the Linux operating system called Netfilter (originating with developer Paul 'Rusty' Russell in 1999) to analyze and make decisions about what to do with an IP packet arriving and leaving through the servers network interfaces. This type of capability is most often used to protect traffic between an internal LAN and the Internet. Netfilter enables truly complex rules to be defined to **'filter'** (i.e. make decisions relating to a packet) specifically:
 - Filter traffic traversing the vRouter using the `in` and `out` keywords;
 - Filter traffic destined for the vRouter itself with the `local` keyword;
 - Filter based on matching the contents or characteristics of IP packets like the source address, the destination address, the port number, the protocol being used etc.
- Supports the creation of groups of:
 - Ports;
 - Addresses; and
 - Networks
- Works with 'firewall instances' which are named sets of firewall rules. As you will see when we define these in subsequent pages the Firewall (used to denote the vRouter itself) comprises many (sometimes hundreds) of firewall instances or firewall rule sets. The point being pressed here is do not think monolithically – see the Firewall as made up of lots of named sets of rules;
- Once you define the firewall instance and its rules you apply the firewall instance to an interface or a zone;
- Applies rules in a firewall instance sequentially according to its rule number;
- Does three things or takes three actions:
 - Accepts the packet (or traffic);
 - Drops the packet (or traffic);
 - Rejects the packet (or traffic) – same as above but sends back a message saying "Port Unreachable"

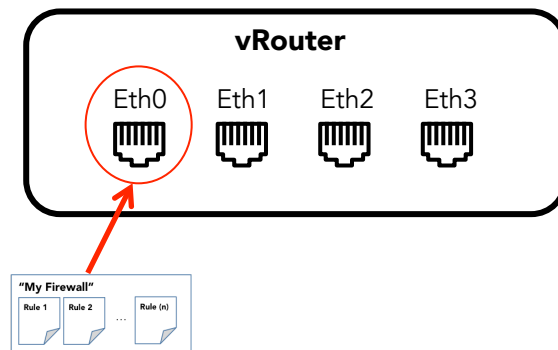
Step 1) Define a Firewall Instance



Step 2) Define rules in the Firewall Instance



Step 3) Apply the Firewall Instance to an Interface





A little more on **in**, **out** and **local** keywords

Given their importance its worth spending another slide on these keywords. So from the previous slide we define our Firewall Instance and its rules. These rules are applied in three ways depending on what we specify when we apply the Firewall Instance. So what can we specify? That would be:

- **in** – if we apply the Firewall Instance as **in**, then our rules are applied to packets (or traffic) entering the interface and traversing (coming in one interface and out another) the vRouter;
- **out** – guesses? Yep if we apply the Firewall Instance as **out**, then our rules are applied to packets (or traffic) leaving the interface. This can be packets traversing the vRouter or they could be packets originating on the vRouter;
- **local** – Well if we've covered **in** and **out** – what's left? Packets for the vRouter itself. If we apply the Firewall Instance as **local** our rules are applied to packets (or traffic) destined for the vRouter itself.

Lets make this more real with a pseudo-code example ... emphasis on the pseudo-code!

Step 1) Define a Firewall Instance – How?

```
zeus@rtr-lon-01:~# set firewall name FROM-PRIVATE-TO-PUBLIC default-action drop
```

This creates a Firewall Instance with a good name describing its function. What does it do? It drops everything. But this Firewall Instance is not working yet because we need to apply it to an interface or a zone (more on zones coming).

Step 2) Apply the Firewall Instance – How?

```
zeus@rtr-lon-01:~# set interfaces ethernet eth0 firewall out name FROM-PRIVATE-TO-PUBLIC
```

Now our firewall is active and it simply drops everything going **OUT** the eth0 interface!



A couple more helpful things!

Groups:

To help simplify the process of designing and defining a firewall the vRouter comes with the ability to define Firewall Groups. Groups can be configured to be groups of ports, groups of IP addresses, or even groups of networks. The VyOS/Brocade documentation provides examples of these, for instance defining a group of networks:

```
zeus@rtr-lon-01:~# set firewall group network-group SOFTLAYER_PRIVATE_NETWORKS network 10.0.64.0/19
zeus@rtr-lon-01:~# set firewall group network-group SOFTLAYER_PRIVATE_NETWORKS network 10.0.128.0/19
```

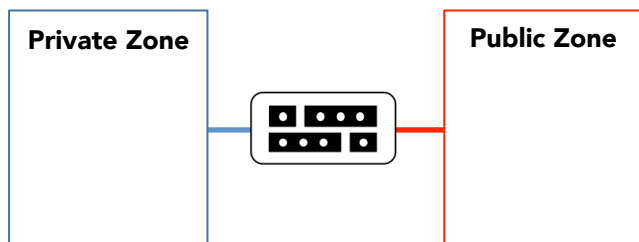
This creates a group of networks whose members include the 10.0.64.0 and 10.0.128.0 networks. This makes it simpler for us to define whole groups of networks that we want to deny traffic to/from or indeed allow to/from. We can also do this with port groups:

```
zeus@rtr-lon-01:~# set firewall group port-group PORTS_ALLOWED port 22
zeus@rtr-lon-01:~# set firewall group port-group PORTS_ALLOWED port 80
zeus@rtr-lon-01:~# set firewall group port-group PORTS_ALLOWED port 443
```

In this example we are defining a group of ports called PORTS_ALLOWED and including ports 22, 80 and 443.

Zones:

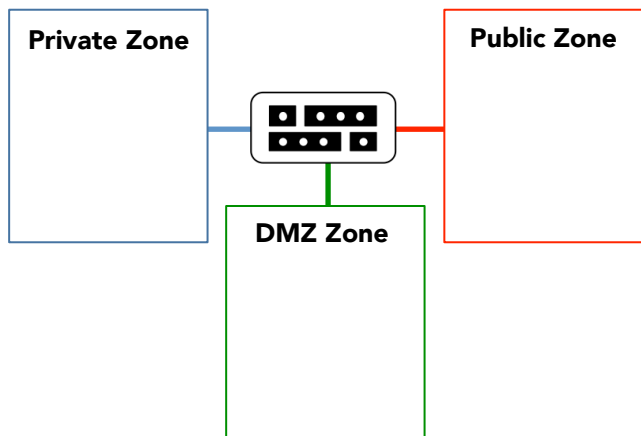
Most of the firewalls I've built on SoftLayer are zone based. Why? It is (mostly) easier to use zones as they present a clearer more readable picture of what's going on within our firewall design. The examples above showed interface based firewalling. In a zone based firewall we group interfaces into security "Zones" where each interface assigned in the "Zone" will have the same security policies. Then packets (or traffic) are filtered between the Zones. This also has the advantage that traffic between networks within a specific Zone does not require a firewall and can be unfiltered.



When setting up Zones we do this in pairs – that means a **from_zone** and a **to_zone** pairing. In the diagram we can see three Zones, a public, a private and what's the third? The local zone i.e. the vRouter itself. This means our Zone Policy pair will be:

- From Private to Public
- From Public to Private

Another Zone pair example



Remember when setting up Zones we do this in pairs – that means a **from_zone** and a **to_zone** pairing. In this diagram we can see four Zones, a public, a private, a DMZ and what's the fourth? The local zone i.e. the vRouter itself. This means our Zone Policy pairs will be:

- From Private to Public
- From Public to Private
- From Public to DMZ
- From DMZ to Public
- From Private to DMZ
- From DMZ to Private

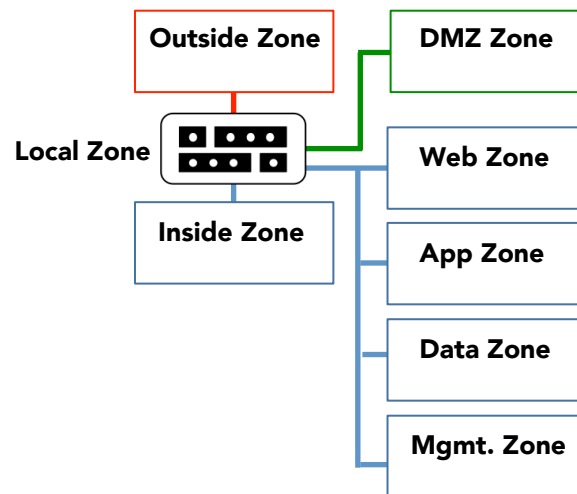
Building our Zone map

Reviewing our traffic flows from slide [20](#) we can design our Zone based firewall based on eight Zones:

- DMZ Zone
- Web Zone
- App Zone
- Database Zone
- Management Zone
- Outside Zone (this is the Internet)
- Inside Zone (this is the Private SoftLayer networks)
- Local Zone (the vRouter itself)

From this (and Slide #[20](#)) we can derive our sets of zone pairs:

APP-TO-DATA	DATA-TO-APP
APP-TO-INSIDE	INSIDE-TO-APP
APP-TO-MANAGEMENT	MANAGEMENT-TO-APP
APP-TO-WEB	WEB-TO-APP
DATA-TO-INSIDE	INSIDE-TO-DATA
DATA-TO-MANAGEMENT	MANAGEMENT-TO-DATA
DATA-TO-WEB	WEB-TO-DATA
DMZ-TO-INSIDE	INSIDE-TO-DMZ
DMZ-TO-WEB	WEB-TO-DMZ
MANAGEMENT-TO-INSIDE	INSIDE-TO-MANAGEMENT
MANAGEMENT-TO-WEB	WEB-TO-MANAGEMENT
WEB-TO-INSIDE	INSIDE-TO-WEB
WEB-TO-OUTSIDE	OUTSIDE-TO-WEB
OUTSIDE-TO-LOCAL	LOCAL-TO-OUTSIDE





One final thing! A group for SoftLayers networks

The final preparation step to getting our firewalls defined is to build a group of all the internal 10.x.x.x networks and the external 159.x.x.x (or other) required SoftLayer networks. These networks are required for SoftLayer to be able to build and manage machines on your VLANs. Given this statement you can see how tight you can make security on your VLANs – yes you can even lock out SoftLayer! A list is available on <http://knowledge.softlayer.com/faqs/196#154> but there are others we've collated over the months. These networks are there to enable things like vulnerability scans, SSL VPN, PPTP, OS updates, OS loads, etc. To build the group for the private SoftLayer networks we do:

```
zeus@rtr-lon-01:~$ configure
[edit]
zeus@rtr-lon-01:~# set firewall group network-group SL_Private description "Allowed SL Private IP range"
[edit]
zeus@rtr-lon-01:~# set firewall group network-group SL_Private network 10.0.64.0/19
[edit]
zeus@rtr-lon-01:~# set firewall group network-group SL_Private network 10.1.128.0/19
[edit]
zeus@rtr-lon-01:~# set firewall group network-group SL_Private network 10.1.208.0/20
[edit]
zeus@rtr-lon-01:~# set firewall group network-group SL_Private network 10.2.144.0/20
[edit]
zeus@rtr-lon-01:~# set firewall group network-group SL_Private network 10.2.220.0/24
[edit]
zeus@rtr-lon-01:~# set firewall group network-group SL_Private network 10.3.236.0/24
[edit]
zeus@rtr-lon-01:~# set firewall group network-group SL_Private network 10.2.221.0/24
[edit]
zeus@rtr-lon-01:~# set firewall group network-group SL_Private network 10.3.237.0/24
[edit]
zeus@rtr-lon-01:~# set firewall group network-group SL_Private network 10.2.65.0/24
[edit]
zeus@rtr-lon-01:~# set firewall group network-group SL_Private network 10.0.64.0/20
[edit]
zeus@rtr-lon-01:~#
```



And the SoftLayer public networks group

To build the group for the public SoftLayer networks we add:

```
zeus@rtr-lon-01:~# set firewall group network-group SL_Public description "Allowed SL Public IP range"
[edit]
zeus@rtr-lon-01:~# set firewall group network-group SL_Public network 5.10.116.0/24
[edit]
zeus@rtr-lon-01:~# set firewall group network-group SL_Public network 159.8.116.0/24
[edit]
zeus@rtr-lon-01:~# set firewall group network-group SL_Public network 173.192.255.230/32
[edit]
zeus@rtr-lon-01:~# commit
[edit]
zeus@rtr-lon-01:~# save
Saving configuration to '/config/config.boot' ...
Done
```

We will use these group definitions within our firewall rules.

Now we can enhance our Traffic Flows now

So London's traffic flows enhanced with our new Zones are as follows:

LONDON	To Zone From Zone	DMZ	Web	App	Database	Management	Local	Inside	Outside
	DMZ	---	YES	NO	NO	NO	NO	YES (SL Nets)	NO
	Web	NO	---	YES	YES (445/1433)	YES	NO	YES (SL Nets)	YES
	App	NO	YES (445)	---	YES	YES	NO	YES (SL Nets)	NO
	Database	NO	YES (445)	YES (445)	---	YES	NO	YES (SL Nets)	NO
	Management	NO	YES	YES	YES	---	NO	YES (SL Nets)	NO
	Local	NO	NO	NO	NO	NO	---	NO	YES (22272,ping)
	Inside	YES (SL Nets)	YES (SL Nets)	YES (SL Nets)	YES (SL Nets)	YES (SL Nets)	NO	---	NO
	Outside	NO	YES (25,80,443)	NO	NO	NO	YES (22272,ping)	NO	---

 **Now we can configure our firewalls!**



DMZ-TO-WEB

We've done all the hard work now so we can work from top to bottom on our Zone map to develop the specific firewall rules. Lets start with the DMZ. We need to allow the DMZ layer access to the WEB Zone machines to allow Internet traffic to flow to our web services:

We can see from our Zone map (remember read left to right!) that we have YES with no specific ports on the intersect of DMZ & WEB. This means we will allow all traffic to flow from DMZ-TO-WEB!

LONDON	To Zone From Zone	DMZ	Web	App	Database	Management	Local	Inside	Outside
	DMZ	---	YES	NO	NO	NO	NO	YES (SL Nets)	NO
	Web	NO	---	YES (445/1433)	YES (445/1433)	YES	NO	YES (SL Nets)	YES
	App	NO	YES (445)	---	YES	YES	NO	YES (SL Nets)	NO
	Database	NO	YES (445)	YES (445)	---	YES	NO	YES (SL Nets)	NO
	Management	NO	YES	YES	YES	---	NO	YES (SL Nets)	NO
	Local	NO	NO	NO	NO	NO	---	NO	YES (22272,ping)
	Inside	YES (SL Nets)	YES (SL Nets)	YES (SL Nets)	YES (SL Nets)	YES (SL Nets)	NO	---	NO
	Outside	NO	YES (25,80,443)	NO	NO	NO	YES (22272,ping)	NO	---

So now we can issue commands:

```
zeus@rtr-lon-01:~# set firewall name DMZ-TO-WEB default-action drop
[edit]
zeus@rtr-lon-01:~# set firewall name DMZ-TO-WEB description "DMZ traffic to the WEB Zone"
[edit]
zeus@rtr-lon-01:~# set firewall name DMZ-TO-WEB rule 100 action accept
[edit]
zeus@rtr-lon-01:~# set firewall name DMZ-TO-WEB rule 100 destination group
[edit]
zeus@rtr-lon-01:~# commit
[edit]
zeus@rtr-lon-01:~# save
Saving configuration to '/config/config.boot' ...
Done
```



DMZ-TO-INSIDE

We need to allow the DMZ layer access to specified private SoftLayer networks to access OS updates and centralized services i.e. object store, anti-virus updates and OS patches:

We can see from our Zone map (remember read left to right!) that we have YES with no specific ports on the intersect of DMZ & INSIDE. That said we know we have created a specific group of networks called SL_PRIVATE. This means we will allow all traffic to this specific group to flow from DMZ-TO-INSIDE!

LONDON	To Zone From Zone		DMZ	Web	App	Database	Management	Local	Inside	Outside
	DMZ	---	---	YES	NO	NO	NO	NO	YES (SL Nets)	NO
	Web	NO	---	YES	YES (445/1433)	YES	YES	NO	YES (SL Nets)	YES
	App	NO	YES (445)	---	YES	YES	YES	NO	YES (SL Nets)	NO
	Database	NO	YES (445)	YES (445)	---	YES	YES	NO	YES (SL Nets)	NO
	Management	NO	YES	YES	YES	---	---	NO	YES (SL Nets)	NO
	Local	NO	NO	NO	NO	NO	NO	---	NO	YES (22272,ping)
	Inside	YES (SL Nets)	YES (SL Nets)	YES (SL Nets)	YES (SL Nets)	YES (SL Nets)	YES (SL Nets)	NO	---	NO
	Outside	NO	YES (25,80,443)	NO	NO	NO	NO	YES (22272,ping)	NO	---

So now we can issue commands:

```
zeus@rtr-lon-01:~# set firewall name DMZ-TO-INSIDE default-action drop
[edit]
zeus@rtr-lon-01:~# set firewall name DMZ-TO-INSIDE description "DMZ traffic to the SL Private Networks Group"
[edit]
zeus@rtr-lon-01:~# set firewall name DMZ-TO-INSIDE rule 100 action accept
[edit]
zeus@rtr-lon-01:~# set firewall name DMZ-TO-INSIDE rule 100 destination group network-group SL_Private
[edit]
zeus@rtr-lon-01:~# commit
[edit]
zeus@rtr-lon-01:~# save
Saving configuration to '/config/config.boot' ...
Done
```



WEB-TO-APP

Moving down a line on our mapping we need to cover WEB to APP:

We can see from our Zone map (remember read left to right!) that we have YES with no specific ports on the intersect of WEB & APP. This means we will allow all traffic to flow from WEB-TO-APP!

LONDON	To Zone From Zone	DMZ	Web	App	Database	Management	Local	Inside	Outside
		DMZ	Web	App	Database	Management	Local	Inside	Outside
	DMZ	---	YES	NO	NO	NO	NO	YES (SL Nets)	NO
	Web	NO	---	YES (445/1433)	YES (445/1433)	YES	NO	YES (SL Nets)	YES
	App	NO	YES (445)	---	YES	YES	NO	YES (SL Nets)	NO
	Database	NO	YES (445)	YES (445)	---	YES	NO	YES (SL Nets)	NO
	Management	NO	YES	YES	YES	---	NO	YES (SL Nets)	NO
	Local	NO	NO	NO	NO	NO	---	NO	YES (22272,ping)
	Inside	YES (SL Nets)	YES (SL Nets)	YES (SL Nets)	YES (SL Nets)	YES (SL Nets)	NO	---	NO
	Outside	NO	YES (25,80,443)	NO	NO	NO	YES (22272,ping)	NO	---

So now we can issue commands:

```
zeus@rtr-lon-01:~# set firewall name WEB-TO-APP default-action drop
[edit]
zeus@rtr-lon-01:~# set firewall name WEB-TO-APP description "WEB traffic to the APP Zone"
[edit]
zeus@rtr-lon-01:~# set firewall name WEB-TO-APP rule 100 action accept
[edit]
zeus@rtr-lon-01:~# set firewall name WEB-TO-APP rule 100 destination group
[edit]
zeus@rtr-lon-01:~# commit
[edit]
zeus@rtr-lon-01:~# save
Saving configuration to '/config/config.boot' ...
Done
```


WEB-TO-DATABASE

Next on our mapping we need to cover WEB to DATABASE:

We can see from our Zone map (remember read left to right!) that we have YES with specific ports on the intersect of WEB & DATABASE. This means we will allow the ports 445 & 1433 traffic to flow from WEB-TO-DATA!

LONDON	From Zone \ To Zone	DMZ	Web	App	Database	Management	Local	Inside	Outside
		DMZ	Web	App	Database	Management	Local	Inside	Outside
	DMZ	---	YES	NO	NO	NO	NO	YES (SL Nets)	NO
	Web	NO	---	YES	YES (445/1433)	YES	NO	YES (SL Nets)	YES
	App	NO	YES (445)	---	YES	YES	NO	YES (SL Nets)	NO
	Database	NO	YES (445)	YES (445)	---	YES	NO	YES (SL Nets)	NO
	Management	NO	YES	YES	YES	---	NO	YES (SL Nets)	NO
	Local	NO	NO	NO	NO	NO	---	NO	YES (22272,ping)
	Inside	YES (SL Nets)	YES (SL Nets)	YES (SL Nets)	YES (SL Nets)	YES (SL Nets)	NO	---	NO
	Outside	NO	YES (25,80,443)	NO	NO	NO	YES (22272,ping)	NO	---

So now we can issue commands:

```
zeus@rtr-lon-01:~# set firewall name WEB-TO-DATA default-action drop
[edit]
zeus@rtr-lon-01:~# set firewall name WEB-TO-DATA description "WEB traffic to the Database Zone"
[edit]
zeus@rtr-lon-01:~# set firewall name WEB-TO-DATA rule 110 action accept
[edit]
zeus@rtr-lon-01:~# set firewall name WEB-TO-DATA rule 110 destination port 445
[edit]
zeus@rtr-lon-01:~# set firewall name WEB-TO-DATA rule 110 protocol tcp_udp
[edit]
zeus@rtr-lon-01:~# set firewall name WEB-TO-DATA rule 120 action accept
[edit]
zeus@rtr-lon-01:~# set firewall name WEB-TO-DATA rule 120 destination port 1433
[edit]
zeus@rtr-lon-01:~# set firewall name WEB-TO-DATA rule 120 protocol tcp_udp
[edit]
zeus@rtr-lon-01:~# commit
[edit]
zeus@rtr-lon-01:~# save
Saving configuration to '/config/config.boot' ...
Done
```



WEB-TO-MANAGEMENT

Next on our mapping we need to cover WEB to MANAGEMENT:

We can see from our Zone map (remember read left to right!) that we have YES with no specific ports on the intersect of WEB & MANAGEMENT. This means we will allow all traffic to flow from WEB-TO-MANAGEMENT!

LONDON	To Zone From Zone	DMZ	Web	App	Database	Management	Local	Inside	Outside
	DMZ	---	YES	NO	NO	NO	NO	YES (SL Nets)	NO
	Web	NO	---	YES	YES (445/1433)	YES	NO	YES (SL Nets)	YES
	App	NO	YES (445)	---	YES	YES	NO	YES (SL Nets)	NO
	Database	NO	YES (445)	YES (445)	---	YES	NO	YES (SL Nets)	NO
	Management	NO	YES	YES	YES	---	NO	YES (SL Nets)	NO
	Local	NO	NO	NO	NO	NO	---	NO	YES (2272,ping)
	Inside	YES (SL Nets)	YES (SL Nets)	YES (SL Nets)	YES (SL Nets)	YES (SL Nets)	NO	---	NO
	Outside	NO	YES (25,80,443)	NO	NO	NO	YES (2272,ping)	NO	---

So now we can issue commands:

```
zeus@rtr-lon-01:~# set firewall name WEB-TO-MANAGEMENT default-action drop
[edit]
zeus@rtr-lon-01:~# set firewall name WEB-TO-MANAGEMENT description "WEB traffic to the Management Zone"
[edit]
zeus@rtr-lon-01:~# set firewall name WEB-TO-MANAGEMENT rule 100 action accept
[edit]
zeus@rtr-lon-01:~# commit
[edit]
zeus@rtr-lon-01:~# save
Saving configuration to '/config/config.boot' ...
Done
```



WEB-TO-INSIDE

Finally on this line we need to cover WEB to INSIDE to allow the specific :

We can see from our Zone map (remember read left to right!) that we have YES with no specific ports on the intersect of WEB & MANAGEMENT. That said we know we have created a specific group of networks called SL_PRIVATE. This means we will allow all traffic to this specific group to flow from WEB-TO-INSIDE!

LONDON	To Zone From Zone	DMZ	Web	App	Database	Management	Local	Inside	Outside
	DMZ	---	YES	NO	NO	NO	NO	YES (SL Nets)	NO
	Web	NO	---	YES	YES (445/1433)	YES	NO	YES (SL Nets)	YES
	App	NO	YES (445)	---	YES	YES	NO	YES (SL Nets)	NO
	Database	NO	YES (445)	YES (445)	---	YES	NO	YES (SL Nets)	NO
	Management	NO	YES	YES	YES	---	NO	YES (SL Nets)	NO
	Local	NO	NO	NO	NO	NO	---	NO	YES (22272,ping)
	Inside	YES (SL Nets)	YES (SL Nets)	YES (SL Nets)	YES (SL Nets)	YES (SL Nets)	NO	---	NO
	Outside	NO	YES (25,80,443)	NO	NO	NO	YES (22272,ping)	NO	---

So now we can issue commands:

```
zeus@rtr-lon-01:~# set firewall name WEB-TO-INSIDE default-action drop
[edit]
zeus@rtr-lon-01:~# set firewall name WEB-TO-INSIDE description "WEB traffic to the SL Private Networks Group"
[edit]
zeus@rtr-lon-01:~# set firewall name WEB-TO-INSIDE rule 100 action accept
[edit]
zeus@rtr-lon-01:~# set firewall name WEB-TO-INSIDE rule 100 destination group network-group SL_Private
[edit]
zeus@rtr-lon-01:~# commit
[edit]
zeus@rtr-lon-01:~# save
Saving configuration to '/config/config.boot' ...
Done
```



APP-TO-WEB

Moving down another line - now we can do APP-TO-WEB. We can allow one port to have access between Zones. So to allow MSDTC traffic on port 445 between APP and WEB Zones we can:

We can see from our Zone map (remember read left to right!) that we have YES with a specific port on the intersect of APP & WEB. This means we will allow port 445 traffic to flow from APP-TO-WEB!

LONDON	To Zone From Zone	DMZ	Web	App	Database	Management	Local	Inside	Outside
	DMZ	---	YES	NO	NO	NO	NO	YES (SL Nets)	NO
	Web	NO	---	YES (445/1433)	YES (445/1433)	YES	NO	YES (SL Nets)	YES
	App	NO	YES (445)	---	YES	YES	NO	YES (SL Nets)	NO
	Database	NO	YES (445)	YES (445)	---	YES	NO	YES (SL Nets)	NO
	Management	NO	YES	YES	YES	---	NO	YES (SL Nets)	NO
	Local	NO	NO	NO	NO	NO	---	NO	YES (22272,ping)
	Inside	YES (SL Nets)	YES (SL Nets)	YES (SL Nets)	YES (SL Nets)	YES (SL Nets)	NO	---	NO
	Outside	NO	YES (25,80,443)	NO	NO	NO	YES (22272,ping)	NO	---

So now we can issue commands:

```
zeus@rtr-lon-01:~# set firewall name APP-TO-WEB default-action drop
[edit]
zeus@rtr-lon-01:~# set firewall name APP-TO-WEB description "Applications MSDTC traffic to the WEB Zone"
[edit]
zeus@rtr-lon-01:~# set firewall name APP-TO-WEB rule 110 action accept
[edit]
zeus@rtr-lon-01:~# set firewall name APP-TO-WEB rule 110 destination port 445
[edit]
zeus@rtr-lon-01:~# set firewall name APP-TO-WEB rule 110 protocol tcp_udp
[edit]
zeus@rtr-lon-01:~# commit
[edit]
zeus@rtr-lon-01:~# save
Saving configuration to '/config/config.boot' ...
Done
```



APP-TO-DATA

Now we can do APP-TO-DATA:

We can see from our Zone map (remember read left to right!) that we have YES with no specific ports on the intersect of APP & DATA. This means we will allow all traffic to flow from APP-TO-DATA!

LONDON	To Zone From Zone	DMZ	Web	App	Database	Management	Local	Inside	Outside
	DMZ	---	YES	NO	NO	NO	NO	YES (SL Nets)	NO
	Web	NO	---	YES (445/1433)	YES (445/1433)	YES	NO	YES (SL Nets)	YES
	App	NO	YES (445)	---	YES	YES	NO	YES (SL Nets)	NO
	Database	NO	YES (445)	YES (445)	---	YES	NO	YES (SL Nets)	NO
	Management	NO	YES	YES	YES	---	NO	YES (SL Nets)	NO
	Local	NO	NO	NO	NO	NO	---	NO	YES (22272,ping)
	Inside	YES (SL Nets)	YES (SL Nets)	YES (SL Nets)	YES (SL Nets)	YES (SL Nets)	NO	---	NO
	Outside	NO	YES (25,80,443)	NO	NO	NO	YES (22272,ping)	NO	---

So now we can issue commands:

```
zeus@rtr-lon-01:~# set firewall name APP-TO-DATA default-action drop
[edit]
zeus@rtr-lon-01:~# set firewall name APP-TO-DATA description "Applications traffic to the Databases
Zone"
[edit]
zeus@rtr-lon-01:~# set firewall name APP-TO-DATA rule 100 action accept
[edit]
zeus@rtr-lon-01:~# commit
[edit]
zeus@rtr-lon-01:~# save
Saving configuration to '/config/config.boot' ...
```



APP-TO-MANAGEMENT

For the Application layer to the Management layer we are allowing all traffic:

So we can see from our Zone map (remember read left to right!) that we have YES with no specific ports on the intersect of APP & MANAGEMENT. This means we will allow all traffic to flow from APP-TO-MANAGEMENT!

LONDON	To Zone From Zone	DMZ	Web	App	Database	Management	Local	Inside	Outside
	DMZ	---	YES	NO	NO	NO	NO	YES (SL Nets)	NO
	Web	NO	---	YES	YES (445/1433)	YES	NO	YES (SL Nets)	YES
	App	NO	YES (445)	---	YES	YES	NO	YES (SL Nets)	NO
	Database	NO	YES (445)	YES (445)	---	YES	NO	YES (SL Nets)	NO
	Management	NO	YES	YES	YES	---	NO	YES (SL Nets)	NO
	Local	NO	NO	NO	NO	NO	---	NO	YES (22272,ping)
	Inside	YES (SL Nets)	YES (SL Nets)	YES (SL Nets)	YES (SL Nets)	YES (SL Nets)	NO	---	NO
	Outside	NO	YES (25,80,443)	NO	NO	NO	YES (22272,ping)	NO	---

So now we can issue commands:

```
zeus@rtr-lon-01:~# set firewall name APP-TO-MANAGEMENT default-action drop
[edit]
zeus@rtr-lon-01:~# set firewall name APP-TO-MANAGEMENT description "Applications traffic to the Management
Zone"
[edit]
zeus@rtr-lon-01:~# set firewall name APP-TO-MANAGEMENT rule 100 action accept
[edit]
zeus@rtr-lon-01:~# commit
[edit]
zeus@rtr-lon-01:~# save
Saving configuration to '/config/config.boot' ...
Done
```



APP-TO-INSIDE

We need to allow the Applications layer access to specified private SoftLayer networks to access OS updates and centralized services i.e. object store, anti-virus updates and OS patches:

We can see from our Zone map (remember read left to right!) that we have YES with no specific ports on the intersect of APP & INSIDE. That said we know we have created a specific group of networks called SL_PRIVATE. This means we will allow all traffic to this specific group to flow from APP-TO-INSIDE!

LONDON	To Zone From Zone	DMZ	Web	App	Database	Management	Local	Inside	Outside
	DMZ	---	YES	NO	NO	NO	NO	YES (SL Nets)	NO
	Web	NO	---	YES	YES (445/1433)	YES	NO	YES (SL Nets)	YES
	App	NO	YES (445)	---	YES	YES	NO	YES (SL Nets)	NO
	Database	NO	YES (445)	YES (445)	---	YES	NO	YES (SL Nets)	NO
	Management	NO	YES	YES	YES	---	NO	YES (SL Nets)	NO
	Local	NO	NO	NO	NO	NO	---	NO	YES (22272,ping)
	Inside	YES (SL Nets)	YES (SL Nets)	YES (SL Nets)	YES (SL Nets)	YES (SL Nets)	NO	---	NO
	Outside	NO	YES (25,80,443)	NO	NO	NO	YES (22272,ping)	NO	---

So now we can issue commands:

```
zeus@rtr-lon-01:~# set firewall name APP-TO-INSIDE default-action drop
[edit]
zeus@rtr-lon-01:~# set firewall name APP-TO-INSIDE description "Applications traffic to the SL Private Networks Group"
[edit]
zeus@rtr-lon-01:~# set firewall name APP-TO-INSIDE rule 100 action accept
[edit]
zeus@rtr-lon-01:~# set firewall name APP-TO-INSIDE rule 100 destination group network-group SL_Private
[edit]
zeus@rtr-lon-01:~# commit
[edit]
zeus@rtr-lon-01:~# save
Saving configuration to '/config/config.boot' ...
Done
```



DATA-TO-WEB

Lets move on to the DATA Zone, moving left to right, and cover DATA to WEB:

We can see from our Zone map (remember read left to right!) that we have YES with a specific port on the intersect of DATA & WEB. This means we will allow port 445 traffic to flow from DATA to WEB!

LONDON	To Zone From Zone	DMZ	Web	App	Database	Management	Local	Inside	Outside
		DMZ	Web	App	Database	Management	Local	Inside	Outside
	DMZ	---	YES	NO	NO	NO	NO	YES (SL Nets)	NO
	Web	NO	---	YES	YES (445/1433)	YES	NO	YES (SL Nets)	YES
	App	NO	YES (445)	---	YES	YES	NO	YES (SL Nets)	NO
	Database	NO	YES (445)	YES (445)	---	YES	NO	YES (SL Nets)	NO
	Management	NO	YES	YES	YES	---	NO	YES (SL Nets)	NO
	Local	NO	NO	NO	NO	NO	---	NO	YES (22272,ping)
	Inside	YES (SL Nets)	YES (SL Nets)	YES (SL Nets)	YES (SL Nets)	YES (SL Nets)	NO	---	NO
	Outside	NO	YES (25,80,443)	NO	NO	NO	YES (22272,ping)	NO	---

So now we can issue commands:

```
zeus@rtr-lon-01:~# set firewall name DATA-TO-WEB default-action drop
[edit]
zeus@rtr-lon-01:~# set firewall name DATA-TO-WEB description "Database Zone MSDTC traffic to the WEB Zone"
[edit]
zeus@rtr-lon-01:~# set firewall name DATA-TO-WEB rule 110 action accept
[edit]
zeus@rtr-lon-01:~# set firewall name DATA-TO-WEB rule 110 destination port 445
[edit]
zeus@rtr-lon-01:~# set firewall name DATA-TO-WEB rule 110 protocol tcp_udp
[edit]
zeus@rtr-lon-01:~# commit
[edit]
zeus@rtr-lon-01:~# save
Saving configuration to '/config/config.boot' ...
Done
```




DATA-TO-APP

Lets move on to the DATA Zone and cover DATA to APP:

We can see from our Zone map (remember read left to right!) that we have YES with a specific port on the intersect of DATA & APP. This means we will allow port 445 traffic to flow from DATA to APP!

LONDON	To Zone From Zone	DMZ	Web	App	Database	Management	Local	Inside	Outside
	DMZ	---	YES	NO	NO	NO	NO	YES (SL Nets)	NO
	Web	NO	---	YES	YES (445/1433)	YES	NO	YES (SL Nets)	YES
	App	NO	YES (445)	---	YES	YES	NO	YES (SL Nets)	NO
	Database	NO	YES (445)	YES (445)	---	YES	NO	YES (SL Nets)	NO
	Management	NO	YES	YES	YES	---	NO	YES (SL Nets)	NO
	Local	NO	NO	NO	NO	NO	---	NO	YES (22272,ping)
	Inside	YES (SL Nets)	YES (SL Nets)	YES (SL Nets)	YES (SL Nets)	YES (SL Nets)	NO	---	NO
	Outside	NO	YES (25,80,443)	NO	NO	NO	YES (22272,ping)	NO	---

So now we can issue commands:

```
zeus@rtr-lon-01:~# set firewall name DATA-TO-APP default-action drop
[edit]
zeus@rtr-lon-01:~# set firewall name DATA-TO-APP description "Database Zone MSDTC traffic to the APP Zone"
[edit]
zeus@rtr-lon-01:~# set firewall name DATA-TO-APP rule 110 action accept
[edit]
zeus@rtr-lon-01:~# set firewall name DATA-TO-APP rule 110 destination port 445
[edit]
zeus@rtr-lon-01:~# set firewall name DATA-TO-APP rule 110 protocol tcp_udp
[edit]
zeus@rtr-lon-01:~# commit
[edit]
zeus@rtr-lon-01:~# save
Saving configuration to '/config/config.boot' ...
Done
```



DATA-TO-MANAGEMENT

Lets move on to the DATA Zone and cover DATA to MANAGEMENT:

We can see from our Zone map (remember read left to right!) that we have YES for all traffic on the intersect of DATA & MANAGEMENT. This means we will allow all traffic to flow from DATA to MANAGEMENT!

LONDON	To Zone From Zone		DMZ	Web	App	Database	Management	Local	Inside	Outside
	DMZ	---	---	YES	NO	NO	NO	NO	YES (SL Nets)	NO
	Web	NO	---	YES	YES (445/1433)	YES	YES	NO	YES (SL Nets)	YES
	App	NO	YES (445)	---	YES	YES	YES	NO	YES (SL Nets)	NO
	Database	NO	YES (445)	YES (445)	---	YES	YES	NO	YES (SL Nets)	NO
	Management	NO	YES	YES	YES	---	---	NO	YES (SL Nets)	NO
	Local	NO	NO	NO	NO	NO	NO	---	NO	YES (22272,ping)
	Inside	YES (SL Nets)	YES (SL Nets)	YES (SL Nets)	YES (SL Nets)	YES (SL Nets)	YES (SL Nets)	NO	---	NO
	Outside	NO	YES (25,80,443)	NO	NO	NO	NO	YES (22272,ping)	NO	---

So now we can issue commands:

```
zeus@rtr-lon-01:~# set firewall name DATA-TO-MANAGEMENT default-action drop
[edit]
zeus@rtr-lon-01:~# set firewall name DATA-TO-MANAGEMENT description "Database Zone all traffic to MANAGEMENT
Zone"
[edit]
zeus@rtr-lon-01:~# set firewall name DATA-TO-MANAGEMENT rule 100 action accept
[edit]
zeus@rtr-lon-01:~# set firewall name DATA-TO-MANAGEMENT rule 100 destination
[edit]
zeus@rtr-lon-01:~# commit
[edit]
zeus@rtr-lon-01:~# save
Saving configuration to '/config/config.boot' ...
Done
```



DATA-TO-INSIDE

We need to allow the Database layer access to specified private SoftLayer networks to access OS updates and centralized services i.e. object store, anti-virus updates and OS patches:

We can see from our Zone map (remember read left to right!) that we have YES with no specific ports on the intersect of DATA & INSIDE. That said we know we have created a specific group of networks called SL_PRIVATE. This means we will allow all traffic to this specific group to flow from DATA-TO-INSIDE!

LONDON	To Zone From Zone	DMZ	Web	App	Database	Management	Local	Inside	Outside
	DMZ	---	YES	NO	NO	NO	NO	YES (SL Nets)	NO
	Web	NO	---	YES (445/1433)	YES (445/1433)	YES	NO	YES (SL Nets)	YES
	App	NO	YES (445)	---	YES	YES	NO	YES (SL Nets)	NO
	Database	NO	YES (445)	YES (445)	---	YES	NO	YES (SL Nets)	NO
	Management	NO	YES	YES	YES	---	NO	YES (SL Nets)	NO
	Local	NO	NO	NO	NO	NO	---	NO	YES (22272,ping)
	Inside	YES (SL Nets)	YES (SL Nets)	YES (SL Nets)	YES (SL Nets)	YES (SL Nets)	NO	---	NO
	Outside	NO	YES (25,80,443)	NO	NO	NO	YES (22272,ping)	NO	---

So now we can issue commands:

```
zeus@rtr-lon-01:~# set firewall name DATA-TO-INSIDE default-action drop
[edit]
zeus@rtr-lon-01:~# set firewall name DATA-TO-INSIDE description "Database traffic to the SL Private Networks
Group"
[edit]
zeus@rtr-lon-01:~# set firewall name DATA-TO-INSIDE rule 100 action accept
[edit]
zeus@rtr-lon-01:~# set firewall name DATA-TO-INSIDE rule 100 destination group network-group SL_Private
[edit]
zeus@rtr-lon-01:~# commit
[edit]
zeus@rtr-lon-01:~# save
Saving configuration to '/config/config.boot' ...
Done
```



MANAGEMENT-TO-WEB

We need to allow the Management layer access to the Web zone:

We can see from our Zone map (remember read left to right!) that we have YES with no specific ports on the intersect of MANAGEMENT & WEB. This means we will allow all traffic to flow from MANAGEMENT-TO-WEB!

LONDON	To Zone From Zone		DMZ	Web	App	Database	Management	Local	Inside	Outside
	DMZ	---	---	YES	NO	NO	NO	NO	YES (SL Nets)	NO
	Web	NO	---	YES	YES (445/1433)	YES	YES	NO	YES (SL Nets)	YES
	App	NO	YES (445)	---	YES	YES	YES	NO	YES (SL Nets)	NO
	Database	NO	YES (445)	YES (445)	---	YES	YES	NO	YES (SL Nets)	NO
	Management	NO	YES	YES	YES	---	---	NO	YES (SL Nets)	NO
	Local	NO	NO	NO	NO	NO	NO	---	NO	YES (2272,ping)
	Inside	YES (SL Nets)	YES (SL Nets)	YES (SL Nets)	YES (SL Nets)	YES (SL Nets)	YES (SL Nets)	NO	---	NO
	Outside	NO	YES (25,80,443)	NO	NO	NO	NO	YES (2272,ping)	NO	---

So now we can issue commands:

```
zeus@rtr-lon-01:~# set firewall name MANAGEMENT-TO-WEB default-action drop
[edit]
zeus@rtr-lon-01:~# set firewall name MANAGEMENT-TO-WEB description "Management traffic to Web zone"
[edit]
zeus@rtr-lon-01:~# set firewall name MANAGEMENT-TO-WEB rule 100 action accept
[edit]
zeus@rtr-lon-01:~# commit
[edit]
zeus@rtr-lon-01:~# save
Saving configuration to '/config/config.boot' ...
Done
```



MANAGEMENT-TO-APP

We need to allow the Management layer access to the App zone:

We can see from our Zone map (remember read left to right!) that we have YES with no specific ports on the intersect of MANAGEMENT & APP. This means we will allow all traffic to flow from MANAGEMENT-TO-APP!

LONDON	To Zone From Zone	DMZ	Web	App	Database	Management	Local	Inside	Outside
	DMZ	---	YES	NO	NO	NO	NO	YES (SL Nets)	NO
	Web	NO	---	YES (445/1433)	YES (445/1433)	YES	NO	YES (SL Nets)	YES
	App	NO	YES (445)	---	YES	YES	NO	YES (SL Nets)	NO
	Database	NO	YES (445)	YES (445)	---	YES	NO	YES (SL Nets)	NO
	Management	NO	YES	YES	YES	---	NO	YES (SL Nets)	NO
	Local	NO	NO	NO	NO	NO	---	NO	YES (22272,ping)
	Inside	YES (SL Nets)	YES (SL Nets)	YES (SL Nets)	YES (SL Nets)	YES (SL Nets)	NO	---	NO
	Outside	NO	YES (25,80,443)	NO	NO	NO	YES (22272,ping)	NO	---

So now we can issue commands:

```
zeus@rtr-lon-01:~# set firewall name MANAGEMENT-TO-APP default-action drop
[edit]
zeus@rtr-lon-01:~# set firewall name MANAGEMENT-TO-APP description "Management traffic to APP zone"
[edit]
zeus@rtr-lon-01:~# set firewall name MANAGEMENT-TO-APP rule 100 action accept
[edit]
zeus@rtr-lon-01:~# commit
[edit]
zeus@rtr-lon-01:~# save
Saving configuration to '/config/config.boot' ...
Done
```



MANAGEMENT-TO-DATA

We need to allow the Management layer access to the DATA zone:

We can see from our Zone map (remember read left to right!) that we have YES with no specific ports on the intersect of MANAGEMENT & DATA. This means we will allow all traffic to flow from MANAGEMENT-TO-DATA!

LONDON	To Zone From Zone	DMZ	Web	App	Database	Management	Local	Inside	Outside
	DMZ	---	YES	NO	NO	NO	NO	YES (SL Nets)	NO
	Web	NO	---	YES (445/1433)	YES (445/1433)	YES	NO	YES (SL Nets)	YES
	App	NO	YES (445)	---	YES	YES	NO	YES (SL Nets)	NO
	Database	NO	YES (445)	YES (445)	---	YES	NO	YES (SL Nets)	NO
	Management	NO	YES	YES	YES	---	NO	YES (SL Nets)	NO
	Local	NO	NO	NO	NO	NO	---	NO	YES (22272,ping)
	Inside	YES (SL Nets)	YES (SL Nets)	YES (SL Nets)	YES (SL Nets)	YES (SL Nets)	NO	---	NO
	Outside	NO	YES (25,80,443)	NO	NO	NO	YES (22272,ping)	NO	---

So now we can issue commands:

```
zeus@rtr-lon-01:~# set firewall name MANAGEMENT-TO-DATA default-action drop
[edit]
zeus@rtr-lon-01:~# set firewall name MANAGEMENT-TO-DATA description "Management traffic to APP zone"
[edit]
zeus@rtr-lon-01:~# set firewall name MANAGEMENT-TO-DATA rule 100 action accept
[edit]
zeus@rtr-lon-01:~# commit
[edit]
zeus@rtr-lon-01:~# save
Saving configuration to '/config/config.boot' ...
Done
```



MANAGEMENT-TO-INSIDE

We need to allow the Management layer access to specified private SoftLayer networks to access OS updates and centralized services i.e. object store, anti-virus updates and OS patches:

We can see from our Zone map (remember read left to right!) that we have YES with no specific ports on the intersect of MANAGEMENT & INSIDE. That said we know we have created a specific group of networks called SL_PRIVATE. This means we will allow all traffic to this specific group to flow from MANAGEMENT-TO-INSIDE!

LONDON	To Zone From Zone	DMZ	Web	App	Database	Management	Local	Inside	Outside
	DMZ	---	YES	NO	NO	NO	NO	YES (SL Nets)	NO
	Web	NO	---	YES	YES (445/1433)	YES	NO	YES (SL Nets)	YES
	App	NO	YES (445)	---	YES	YES	NO	YES (SL Nets)	NO
	Database	NO	YES (445)	YES (445)	---	YES	NO	YES (SL Nets)	NO
	Management	NO	YES	YES	YES	---	NO	YES (SL Nets)	NO
	Local	NO	NO	NO	NO	NO	---	NO	YES (22272,ping)
	Inside	YES (SL Nets)	YES (SL Nets)	YES (SL Nets)	YES (SL Nets)	YES (SL Nets)	NO	---	NO
	Outside	NO	YES (25,80,443)	NO	NO	NO	YES (22272,ping)	NO	---

So now we can issue commands:

```
zeus@rtr-lon-01:~# set firewall name MANAGEMENT-TO-INSIDE default-action drop
[edit]
zeus@rtr-lon-01:~# set firewall name MANAGEMENT-TO-INSIDE description "Management traffic to the SL Private
Networks Group"
[edit]
zeus@rtr-lon-01:~# set firewall name MANAGEMENT-TO-INSIDE rule 100 action accept
[edit]
zeus@rtr-lon-01:~# set firewall name MANAGEMENT-TO-INSIDE rule 100 destination group network-group SL_Private
[edit]
zeus@rtr-lon-01:~# commit
[edit]
zeus@rtr-lon-01:~# save
Saving configuration to '/config/config.boot' ...
Done
```



INSIDE-TO-DMZ

We have already covered DMZ, WEB, APP, DATA and MANAGEMENT to the INSIDE – now we can do the converse.

We can see from our Zone map (remember read left to right!) that we have YES with our specific groups of networks – denoted SL_PRIVATE. This means we will allow all traffic to this specific group to flow from INSIDE-TO-DMZ!

LONDON	To Zone From Zone	DMZ	Web	App	Database	Management	Local	Inside	Outside
	DMZ	---	YES	NO	NO	NO	NO	YES (SL Nets)	NO
	Web	NO	---	YES	YES (445/1433)	YES	NO	YES (SL Nets)	YES
	App	NO	YES (445)	---	YES	YES	NO	YES (SL Nets)	NO
	Database	NO	YES (445)	YES (445)	---	YES	NO	YES (SL Nets)	NO
	Management	NO	YES	YES	YES	---	NO	YES (SL Nets)	NO
	Local	NO	NO	NO	NO	NO	---	NO	YES (22272,ping)
	Inside	YES (SL Nets)	YES (SL Nets)	YES (SL Nets)	YES (SL Nets)	YES (SL Nets)	NO	---	NO
	Outside	NO	YES (25,80,443)	NO	NO	NO	YES (22272,ping)	NO	---

So now we can issue commands:

```
zeus@rtr-lon-01:~# set firewall name INSIDE-TO-DMZ default-action drop
[edit]
zeus@rtr-lon-01:~# set firewall name INSIDE-TO-DMZ description "SL Private Networks traffic to the DMZ Zone"
[edit]
zeus@rtr-lon-01:~# set firewall name INSIDE-TO-DMZ rule 100 action accept
[edit]
zeus@rtr-lon-01:~# set firewall name INSIDE-TO-DMZ rule 100 source group network-group SL_Private
[edit]
zeus@rtr-lon-01:~# commit
[edit]
zeus@rtr-lon-01:~# save
Saving configuration to '/config/config.boot' ...
Done
```




INSIDE-TO-WEB

We have already covered DMZ, WEB, APP, DATA and MANAGEMENT to the INSIDE – now we can do the converse.

We can see from our Zone map (remember read left to right!) that we have YES with our specific groups of networks – denoted SL_PRIVATE. This means we will allow all traffic to this specific group to flow from INSIDE-TO-WEB!

LONDON	To Zone From Zone	DMZ	Web	App	Database	Management	Local	Inside	Outside
	DMZ	---	YES	NO	NO	NO	NO	YES (SL Nets)	NO
	Web	NO	---	YES (445/1433)	YES (445/1433)	YES	NO	YES (SL Nets)	YES
	App	NO	YES (445)	---	YES	YES	NO	YES (SL Nets)	NO
	Database	NO	YES (445)	YES (445)	---	YES	NO	YES (SL Nets)	NO
	Management	NO	YES	YES	YES	---	NO	YES (SL Nets)	NO
	Local	NO	NO	NO	NO	NO	---	NO	YES (22272,ping)
	Inside	YES (SL Nets)	YES (SL Nets)	YES (SL Nets)	YES (SL Nets)	YES (SL Nets)	NO	---	NO
	Outside	NO	YES (25,80,443)	NO	NO	NO	YES (22272,ping)	NO	---

So now we can issue commands:

```
zeus@rtr-lon-01:~# set firewall name INSIDE-TO-WEB default-action drop
[edit]
zeus@rtr-lon-01:~# set firewall name INSIDE-TO-WEB description "SL Private Networks traffic to the WEB Zone"
[edit]
zeus@rtr-lon-01:~# set firewall name INSIDE-TO-WEB rule 100 action accept
[edit]
zeus@rtr-lon-01:~# set firewall name INSIDE-TO-WEB rule 100 source group network-group SL_Private
[edit]
zeus@rtr-lon-01:~# commit
[edit]
zeus@rtr-lon-01:~# save
Saving configuration to '/config/config.boot' ...
Done
```



INSIDE-TO-APP

We have already covered DMZ, WEB, APP, DATA and MANAGEMENT to the INSIDE – now we can do the converse.

We can see from our Zone map (remember read left to right!) that we have YES with our specific groups of networks – denoted SL_PRIVATE. This means we will allow all traffic to this specific group to flow from INSIDE-TO-APP!

	To Zone From Zone	DMZ	Web	App	Database	Management	Local	Inside	Outside
		DMZ	Web	App	Database	Management	Local	Inside	Outside
LONDON	DMZ	---	YES	NO	NO	NO	NO	YES (SL Nets)	NO
	Web	NO	---	YES (445/1433)	YES	YES	NO	YES (SL Nets)	YES
	App	NO	YES (445)	---	YES	YES	NO	YES (SL Nets)	NO
	Database	NO	YES (445)	YES (445)	---	YES	NO	YES (SL Nets)	NO
	Management	NO	YES	YES	YES	---	NO	YES (SL Nets)	NO
	Local	NO	NO	NO	NO	NO	---	NO	YES (22272,ping)
	Inside	YES (SL Nets)	YES (SL Nets)	YES (SL Nets)	YES (SL Nets)	YES (SL Nets)	NO	---	NO
	Outside	NO	YES (25,80,443)	NO	NO	NO	YES (22272,ping)	NO	---

So now we can issue commands:

```
zeus@rtr-lon-01:~# set firewall name INSIDE-TO-APP default-action drop
[edit]
zeus@rtr-lon-01:~# set firewall name INSIDE-TO-APP description "SL Private Networks traffic to the APP Zone"
[edit]
zeus@rtr-lon-01:~# set firewall name INSIDE-TO-APP rule 100 action accept
[edit]
zeus@rtr-lon-01:~# set firewall name INSIDE-TO-APP rule 100 source group network-group SL_Private
[edit]
zeus@rtr-lon-01:~# commit
[edit]
zeus@rtr-lon-01:~# save
Saving configuration to '/config/config.boot' ...
Done
```



INSIDE-TO-DATA

We have already covered DMZ, WEB, APP, DATA and MANAGEMENT to the INSIDE – now we can do the converse.

We can see from our Zone map (remember read left to right!) that we have YES with our specific groups of networks – denoted SL_PRIVATE. This means we will allow all traffic to this specific group to flow from INSIDE-TO-DATA!

LONDON	To Zone From Zone	DMZ	Web	App	Database	Management	Local	Inside	Outside
	DMZ	---	YES	NO	NO	NO	NO	YES (SL Nets)	NO
	Web	NO	---	YES (445/1433)	YES (445/1433)	YES	NO	YES (SL Nets)	YES
	App	NO	YES (445)	---	YES	YES	NO	YES (SL Nets)	NO
	Database	NO	YES (445)	YES (445)	---	YES	NO	YES (SL Nets)	NO
	Management	NO	YES	YES	YES	---	NO	YES (SL Nets)	NO
	Local	NO	NO	NO	NO	NO	---	NO	YES (22272,ping)
	Inside	YES (SL Nets)	YES (SL Nets)	YES (SL Nets)	YES (SL Nets)	YES (SL Nets)	NO	---	NO
	Outside	NO	YES (25,80,443)	NO	NO	NO	YES (22272,ping)	NO	---

So now we can issue commands:

```
zeus@rtr-lon-01:~# set firewall name INSIDE-TO-DATA default-action drop
[edit]
zeus@rtr-lon-01:~# set firewall name INSIDE-TO-DATA description "SL Private Networks traffic to the DATA Zone"
[edit]
zeus@rtr-lon-01:~# set firewall name INSIDE-TO-DATA rule 100 action accept
[edit]
zeus@rtr-lon-01:~# set firewall name INSIDE-TO-DATA rule 100 source group network-group SL_Private
[edit]
zeus@rtr-lon-01:~# commit
[edit]
zeus@rtr-lon-01:~# save
Saving configuration to '/config/config.boot' ...
Done
```



INSIDE-TO-MANAGEMENT

We have already covered DMZ, WEB, APP, DATA and MANAGEMENT to the INSIDE – now we can do the converse.

We can see from our Zone map (remember read left to right!) that we have YES with our specific groups of networks – denoted SL_PRIVATE. This means we will allow all traffic to this specific group to flow from INSIDE-TO-MANAGEMENT!

LONDON	To Zone From Zone	DMZ	Web	App	Database	Management	Local	Inside	Outside
	DMZ	---	YES	NO	NO	NO	NO	YES (SL Nets)	NO
	Web	NO	---	YES	YES (445/1433)	YES	NO	YES (SL Nets)	YES
	App	NO	YES (445)	---	YES	YES	NO	YES (SL Nets)	NO
	Database	NO	YES (445)	YES (445)	---	YES	NO	YES (SL Nets)	NO
	Management	NO	YES	YES	YES	---	NO	YES (SL Nets)	NO
	Local	NO	NO	NO	NO	NO	---	NO	YES (22272,ping)
	Inside	YES (SL Nets)	YES (SL Nets)	YES (SL Nets)	YES (SL Nets)	YES (SL Nets)	NO	---	NO
	Outside	NO	YES (25,80,443)	NO	NO	NO	YES (22272,ping)	NO	---

So now we can issue commands:

```
zeus@rtr-lon-01:~# set firewall name INSIDE-TO-MANAGEMENT default-action drop
[edit]
zeus@rtr-lon-01:~# set firewall name INSIDE-TO-MANAGEMENT description "SL Private Networks traffic to the
MANAGEMENT Zone"
[edit]
zeus@rtr-lon-01:~# set firewall name INSIDE-TO-MANAGEMENT rule 100 action accept
[edit]
zeus@rtr-lon-01:~# set firewall name INSIDE-TO-MANAGEMENT rule 100 source group network-group SL_Private
[edit]
zeus@rtr-lon-01:~# commit
[edit]
zeus@rtr-lon-01:~# save
Saving configuration to '/config/config.boot' ...
Done
```



LOCAL – Traffic from the Router Pair

We can now do the traffic from the router to Outside – you may not have this requirement at all.

We can see from our Zone map (remember read left to right!) that we have YES with a specific ports or protocols namely 22272 for ssh and ping. This means we will allow traffic from port 22 and ping traffic to flow from LOCAL-TO-OUTSIDE!

LONDON	To Zone From Zone	DMZ	Web	App	Database	Management	Local	Inside	Outside
	DMZ	---	YES	NO	NO	NO	NO	YES (SL Nets)	NO
	Web	NO	---	YES	YES (445/1433)	YES	NO	YES (SL Nets)	YES
	App	NO	YES (445)	---	YES	YES	NO	YES (SL Nets)	NO
	Database	NO	YES (445)	YES (445)	---	YES	NO	YES (SL Nets)	NO
	Management	NO	YES	YES	YES	---	NO	YES (SL Nets)	NO
	Local	NO	NO	NO	NO	NO	---	NO	YES (22272,ping)
	Inside	YES (SL Nets)	YES (SL Nets)	YES (SL Nets)	YES (SL Nets)	YES (SL Nets)	NO	---	NO
	Outside	NO	YES (25,80,443)	NO	NO	NO	YES (22272,ping)	NO	---

So now we can issue commands:

```
zeus@rtr-lon-01:~# set firewall name LOCAL-TO-OUTSIDE default-action drop
[edit]
zeus@rtr-lon-01:~# set firewall name LOCAL-TO-OUTSIDE description "Local traffic to the Outside Zone"
[edit]
zeus@rtr-lon-01:~# set firewall name LOCAL-TO-OUTSIDE rule 100 action accept
[edit]
zeus@rtr-lon-01:~# set firewall name LOCAL-TO-OUTSIDE rule 100 destination port 22272
[edit]
zeus@rtr-lon-01:~# set firewall name LOCAL-TO-OUTSIDE rule 200 action accept
[edit]
zeus@rtr-lon-01:~# set firewall name LOCAL-TO-OUTSIDE rule 200 protocol icmp
[edit]
zeus@rtr-lon-01:~# commit
[edit]
zeus@rtr-lon-01:~# save
Saving configuration to '/config/config.boot' ...
Done
```

OUTSIDE-TO-WEB

We can now do the traffic from outside to the web.

We can see from our Zone map (remember read left to right!) that we have YES with specific ports namely 25, 80 & 443. This means we will allow traffic from these ports to flow from OUTSIDE-TO-WEB!

LONDON	To Zone From Zone	DMZ	Web	App	Database	Management	Local	Inside	Outside
		DMZ	Web	App	Database	Management	Local	Inside	Outside
	DMZ	---	YES	NO	NO	NO	NO	YES (SL Nets)	NO
	Web	NO	---	YES	YES (445/1433)	YES	NO	YES (SL Nets)	YES
	App	NO	YES (445)	---	YES	YES	NO	YES (SL Nets)	NO
	Database	NO	YES (445)	YES (445)	---	YES	NO	YES (SL Nets)	NO
	Management	NO	YES	YES	YES	---	NO	YES (SL Nets)	NO
	Local	NO	NO	NO	NO	NO	---	NO	YES (2272,ping)
	Inside	YES (SL Nets)	YES (SL Nets)	YES (SL Nets)	YES (SL Nets)	YES (SL Nets)	NO	---	NO
	Outside	NO	YES (25,80,443)	NO	NO	NO	YES (2272,ping)	NO	---

So now we can issue commands:

```
zeus@rtr-lon-01:~# set firewall name OUTSIDE-TO-WEB default-action drop
[edit]
zeus@rtr-lon-01:~# set firewall name OUTSIDE-TO-WEB description "Outside traffic to the Web Zone"
[edit]
zeus@rtr-lon-01:~# set firewall name OUTSIDE-TO-WEB rule 100 action accept
[edit]
zeus@rtr-lon-01:~# set firewall name OUTSIDE-TO-WEB rule 100 destination port 25
[edit]
zeus@rtr-lon-01:~# set firewall name OUTSIDE-TO-WEB rule 200 action accept
[edit]
zeus@rtr-lon-01:~# set firewall name OUTSIDE-TO-WEB rule 200 destination port 80
[edit]
zeus@rtr-lon-01:~# set firewall name OUTSIDE-TO-WEB rule 300 action accept
[edit]
zeus@rtr-lon-01:~# set firewall name OUTSIDE-TO-WEB rule 300 destination port 443
[edit]
zeus@rtr-lon-01:~# commit
[edit]
zeus@rtr-lon-01:~# save
Saving configuration to '/config/config.boot' ...
Done
```

OUTSIDE-TO-LOCAL

We can now do the traffic from outside to our router.

We can see from our Zone map (remember read left to right!) that we have YES with specific ports namely 22272 and protocol ping. This means we will allow traffic from these ports & protocols to flow from OUTSIDE-TO-LOCAL! We will also instigate a rule to stop SSH phishing/flooding.

So now we can issue commands:

LONDON	To Zone From Zone	DMZ	Web	App	Database	Management	Local	Inside	Outside
	DMZ	---	YES	NO	NO	NO	NO	YES (SL Nets)	NO
	Web	NO	---	YES	YES (445/1433)	YES	NO	YES (SL Nets)	YES
	App	NO	YES (445)	---	YES	YES	NO	YES (SL Nets)	NO
	Database	NO	YES (445)	YES (445)	---	YES	NO	YES (SL Nets)	NO
	Management	NO	YES	YES	YES	---	NO	YES (SL Nets)	NO
	Local	NO	NO	NO	NO	NO	---	NO	YES (22272,ping)
	Inside	YES (SL Nets)	YES (SL Nets)	YES (SL Nets)	YES (SL Nets)	YES (SL Nets)	NO	---	NO
	Outside	NO	YES (25,80,443)	NO	NO	NO	YES (22272,ping)	NO	---

```
zeus@rtr-lon-01:~# set firewall name OUTSIDE-TO-LOCAL default-action drop
[edit]
zeus@rtr-lon-01:~# set firewall name OUTSIDE-TO-LOCAL description "Outside traffic to the Local Zone"
[edit]
zeus@rtr-lon-01:~# set firewall name OUTSIDE-TO-LOCAL rule 100 action accept
[edit]
zeus@rtr-lon-01:~# set firewall name OUTSIDE-TO-LOCAL rule 100 destination port 22272
[edit]
zeus@rtr-lon-01:~# set firewall name OUTSIDE-TO-LOCAL rule 100 log enable
[edit]
zeus@rtr-lon-01:~# set firewall name OUTSIDE-TO-LOCAL rule 100 protocol tcp
[edit]
zeus@rtr-lon-01:~# set firewall name OUTSIDE-TO-LOCAL rule 100 recent count 3
[edit]
zeus@rtr-lon-01:~# set firewall name OUTSIDE-TO-LOCAL rule 100 recent time 30
[edit]
zeus@rtr-lon-01:~# set firewall name OUTSIDE-TO-LOCAL rule 100 state new enable
[edit]
zeus@rtr-lon-01:~# set firewall name OUTSIDE-TO-LOCAL rule 200 action accept
[edit]
zeus@rtr-lon-01:~# set firewall name OUTSIDE-TO-LOCAL rule 200 protocol icmp
[edit]
zeus@rtr-lon-01:~# commit
[edit]
zeus@rtr-lon-01:~# save
Saving configuration to '/config/config.boot' ...
Done
```

≡ And finally we can apply our firewalls!

Create our Zones first!

We'll once again use our Zone Map to drive through our eight zones. So starting from the top (this time reading top down on the "To Zone" line) we walk through what zones need to exist:

```
zeus@rtr-lon-01:~# set zone-policy zone DMZ description "Our DMZ zone"
[edit]
zeus@rtr-lon-01:~# set zone-policy zone DMZ interface bond1.1238
[edit]
zeus@rtr-lon-01:~# set zone-policy zone INSIDE description "Our Inside zone"
[edit]
zeus@rtr-lon-01:~# set zone-policy zone INSIDE interface bond1
[edit]
zeus@rtr-lon-01:~# set zone-policy zone WEB description "Our Web zone"
[edit]
zeus@rtr-lon-01:~# set zone-policy zone WEB interface bond1.1234
[edit]
zeus@rtr-lon-01:~# set zone-policy zone APP description "Our App zone"
[edit]
zeus@rtr-lon-01:~# set zone-policy zone APP interface bond1.1235
[edit]
zeus@rtr-lon-01:~# set zone-policy zone DATA description "Our Database zone"
[edit]
zeus@rtr-lon-01:~# set zone-policy zone DATA interface bond1.1236
[edit]
zeus@rtr-lon-01:~# set zone-policy zone MANAGEMENT description "Our Management zone"
[edit]
zeus@rtr-lon-01:~# set zone-policy zone MANAGEMENT interface bond1.1237
[edit]
zeus@rtr-lon-01:~# set zone-policy zone LOCAL description "Our Local zone"
[edit]
zeus@rtr-lon-01:~# set zone-policy zone MANAGEMENT interface local
[edit]
zeus@rtr-lon-01:~# set zone-policy zone OUTSIDE description "Our Outside zone"
[edit]
zeus@rtr-lon-01:~# set zone-policy zone OUTSIDE interface bond0
[edit]
zeus@rtr-lon-01:~# commit
[edit]
zeus@rtr-lon-01:~# save
```



Applying the DMZ Zone firewall

So we have our firewall rules defined but nothing is yet secured because we now need to apply these rules to the interfaces. We'll once again use our Zone Map to drive through our eight zones. So starting from the top (this time reading top down on the "To Zone" line) with the DMZ zone we ask ourselves "what traffic is this zone accepting? What did we call the firewall rules for this? So we read our line vertically down the Zone Map we made which shows that we receive traffic from the INSIDE Zone and nothing else. So we will drop everything and have one specific rule that says allow the traffic from the INSIDE to this DMZ Zone in accordance with the firewall rules :

```
zeus@rtr-lon-01:~# set zone-policy zone DMZ default-action drop
[edit]
zeus@rtr-lon-01:~# set zone-policy zone DMZ from INSIDE firewall name INSIDE-TO-DMZ
[edit]
zeus@rtr-lon-01:~# commit
[edit]
zeus@rtr-lon-01:~# save
Saving configuration to '/config/config.boot' ...
Done
```



Applying the INSIDE Zone firewall

For INSIDE that will receive inbound traffic from DMZ, WEB, APP, MANAGEMENT, and INSIDE. To do this we:

```
zeus@rtr-lon-01:~# set zone-policy zone INSIDE default-action drop
[edit]
zeus@rtr-lon-01:~# set zone-policy zone INSIDE from DMZ firewall name DMZ-TO-INSIDE
[edit]
zeus@rtr-lon-01:~# set zone-policy zone INSIDE from WEB firewall name WEB-TO-INSIDE
[edit]
zeus@rtr-lon-01:~# set zone-policy zone INSIDE from APP firewall name APP-TO-INSIDE
[edit]
zeus@rtr-lon-01:~# set zone-policy zone INSIDE from DATA firewall name DATA-TO-INSIDE
[edit]
zeus@rtr-lon-01:~# set zone-policy zone INSIDE from MANAGEMENT firewall name MANAGEMENT-TO-INSIDE
[edit]
zeus@rtr-lon-01:~# commit
[edit]
zeus@rtr-lon-01:~# save
Saving configuration to '/config/config.boot' ...
Done
```



Applying the WEB Zone firewall

So WEB is a similar thought process we can see on our Zone Map that the WEB Zone will receive inbound traffic from DMZ, APP, DATA, MANAGEMENT, INSIDE and OUTSIDE. To do this we:

```
zeus@rtr-lon-01:~# set zone-policy zone WEB default-action drop
[edit]
zeus@rtr-lon-01:~# set zone-policy zone WEB from DMZ firewall name DMZ-TO-WEB
[edit]
zeus@rtr-lon-01:~# set zone-policy zone WEB from APP firewall name APP-TO-WEB
[edit]
zeus@rtr-lon-01:~# set zone-policy zone WEB from DATA firewall name DATA-TO-WEB
[edit]
zeus@rtr-lon-01:~# set zone-policy zone WEB from MANAGEMENT firewall name MANAGEMENT-TO-WEB
[edit]
zeus@rtr-lon-01:~# set zone-policy zone WEB from INSIDE firewall name INSIDE-TO-WEB
[edit]
zeus@rtr-lon-01:~# set zone-policy zone WEB from OUTSIDE firewall name OUTSIDE-TO-WEB
[edit]
zeus@rtr-lon-01:~# commit
[edit]
zeus@rtr-lon-01:~# save
Saving configuration to '/config/config.boot' ...
Done
```



Applying the APP Zone firewall

So on to APP that will receive inbound traffic from WEB, DATA, MANAGEMENT, and INSIDE. To do this we:

```
zeus@rtr-lon-01:~# set zone-policy zone APP default-action drop
[edit]
zeus@rtr-lon-01:~# set zone-policy zone APP from WEB firewall name WEB-TO-APP
[edit]
zeus@rtr-lon-01:~# set zone-policy zone APP from DATA firewall name DATA-TO-APP
[edit]
zeus@rtr-lon-01:~# set zone-policy zone APP from MANAGEMENT firewall name MANAGEMENT-TO-APP
[edit]
zeus@rtr-lon-01:~# set zone-policy zone APP from INSIDE firewall name INSIDE-TO-APP
[edit]
zeus@rtr-lon-01:~# commit
[edit]
zeus@rtr-lon-01:~# save
Saving configuration to '/config/config.boot' ...
Done
```



Applying the DATA Zone firewall

For DATA that will receive inbound traffic from WEB, APP, MANAGEMENT, and INSIDE. To do this we:

```
zeus@rtr-lon-01:~# set zone-policy zone DATA default-action drop
[edit]
zeus@rtr-lon-01:~# set zone-policy zone DATA from WEB firewall name WEB-TO-DATA
[edit]
zeus@rtr-lon-01:~# set zone-policy zone DATA from APP firewall name APP-TO-DATA
[edit]
zeus@rtr-lon-01:~# set zone-policy zone DATA from MANAGEMENT firewall name MANAGEMENT-TO-DATA
[edit]
zeus@rtr-lon-01:~# set zone-policy zone DATA from INSIDE firewall name INSIDE-TO-DATA
[edit]
zeus@rtr-lon-01:~# commit
[edit]
zeus@rtr-lon-01:~# save
Saving configuration to '/config/config.boot' ...
Done
```



Applying the MANAGEMENT Zone firewall

For MANAGEMENT that will receive inbound traffic from WEB, APP, MANAGEMENT, and INSIDE. To do this we:

```
zeus@rtr-lon-01:~# set zone-policy zone MANAGEMENT default-action drop
[edit]
zeus@rtr-lon-01:~# set zone-policy zone MANAGEMENT from WEB firewall name WEB-TO-MANAGEMENT
[edit]
zeus@rtr-lon-01:~# set zone-policy zone MANAGEMENT from APP firewall name APP-TO-MANAGEMENT
[edit]
zeus@rtr-lon-01:~# set zone-policy zone MANAGEMENT from DATA firewall name DATA-TO-MANAGEMENT
[edit]
zeus@rtr-lon-01:~# set zone-policy zone MANAGEMENT from INSIDE firewall name INSIDE-TO-MANAGEMENT
[edit]
zeus@rtr-lon-01:~# commit
[edit]
zeus@rtr-lon-01:~# save
Saving configuration to '/config/config.boot' ...
Done
```



Applying the LOCAL Zone firewall

For LOCAL that will receive inbound traffic from OUTSIDE only. To do this we:

```
zeus@rtr-lon-01:~# set zone-policy zone LOCAL default-action drop
[edit]
zeus@rtr-lon-01:~# set zone-policy zone LOCAL from OUTSIDE firewall name OUTSIDE-TO-LOCAL
[edit]
zeus@rtr-lon-01:~# commit
[edit]
zeus@rtr-lon-01:~# save
Saving configuration to '/config/config.boot' ...
Done
```




Applying the OUTSIDE Zone firewall

Finally for OUTSIDE that will receive inbound traffic from LOCAL and WEB. To do this we:

```
zeus@rtr-lon-01:~# set zone-policy zone OUTSIDE default-action drop
[edit]
zeus@rtr-lon-01:~# set zone-policy zone OUTSIDE from WEB firewall name WEB-TO-OUTSIDE
[edit]
zeus@rtr-lon-01:~# set zone-policy zone OUTSIDE from LOCAL firewall name LOCAL-TO-OUTSIDE
[edit]
zeus@rtr-lon-01:~# commit
[edit]
zeus@rtr-lon-01:~# save
Saving configuration to '/config/config.boot' ...
Done
```

And that's it. Our Zones are in place and we can now test all of the traffic flows to ensure they behave as expected. Of course its also worth noting that all of these commands were issued from rtr-lon-01. This begs the question of "what about rtr-lon-02?" The answer to that can be covered in two ways:

- 1) Issue all the firewall and zone commands from this and the previous section on rtr-lon-02; or
- 2) Use the config-sync capability of Brocade's Vyatta version (recommended for SoftLayer)

One more possibility exists which is to utilize a configuration manager such as Chef to implement your policies on all your routers and firewalls. The next section covers Option 2 config-sync.

≡ One last thing – machine synchronization



Config-sync

As mentioned previously entering all of the commands that we have done on each and every router, in a cluster or (in our case) for London and Paris, would be fairly onerous. To alleviate this overhead the Brocade Vyatta comes with a command to help with machine synchronization.

```
zeus@rtr-lon-01:~# set system config-sync remote-router 10.1.9.7 username 'zeus'
[edit]
zeus@rtr-lon-01:~# set system config-sync remote-router 10.1.9.7 password 'Gr33kGods'
[edit]
zeus@rtr-lon-01:~# set system config-sync remote-router 10.1.9.7 sync-map 'SYNC'
[edit]
zeus@rtr-lon-01:~# set system config-sync remote-router 10.2.9.5 username 'zeus'
[edit]
zeus@rtr-lon-01:~# set system config-sync remote-router 10.2.9.5 password 'Gr33kGods'
[edit]
zeus@rtr-lon-01:~# set system config-sync remote-router 10.2.9.5 sync-map 'SYNC'
[edit]
zeus@rtr-lon-01:~# set system config-sync sync-map SYNC rule 1 action include
[edit]
zeus@rtr-lon-01:~# set system config-sync sync-map SYNC rule 1 location firewall
[edit]
zeus@rtr-lon-01:~# commit
[edit]
zeus@rtr-lon-01:~# save
Saving configuration to '/config/config.boot' ...
Done
```

 **Further training & literature**



Where to get more information

Documentation:

- <http://www.brocade.com/en/products-services/software-networking/network-functions-virtualization/5400-vrouter.html>
- http://vyos.net/wiki/User_Guide
- http://vyos.net/wiki/Category:User_documentation
- <http://www.brocade.com/content/dam/common/documents/content-types/configuration-guide/vyatta-firewall-3.5r3-v01.pdf>
- ftp://ftp.umiacs.umd.edu/.snapshot/hourly.5/pub/mvanopst/reader-sync/vyatta/Vyatta-BasicRouting_R6.4_v01.pdf
- ftp://ftp.ciens.ucv.ve/vyatta/Vyatta_BasicSystemRef_R6.0_v01.pdf
- file:///Users/saasify/Downloads/Vyatta_CommandRef_VC4.1_v03.pdf
- http://www1.brocade.com/downloads/documents/html_product_manuals/vyatta/vyatta_5400_manual/Firewall/wwhelp/wwhimpl/common/html/wwhelp.htm#context=Firewall&file=Configuration%20Examples.03.20.html
- http://vyos.net/wiki/Zone-policy_example

For a broad introduction to Netfilter:

- <http://people.netfilter.org/pablo/docs/login.pdf>

Web Training:

- <https://my.brocade.com/wps/myportal/>

Other examples:

- <https://rbgeek.wordpress.com/2013/05/06/vyatta-basic-configuration-after-installation/>
- https://community.brocade.com/dtscp75322/attachments/dtscp75322/SoftwareNetworking/14/1/Vyatta_Firewall_Best_Practices.pdf
- <http://soucy.org/vyos/UsingVyOSasaFirewall.pdf>
- http://www.rackspace.com/knowledge_center/article/configuring-interface-based-firewall-on-the-vyatta-network-appliance

And of course www.google.com and search for VyOS or Vyatta configurations