

Project no. 015926

6DISS

IPv6 Dissemination and Exploitation

Instrument: SPECIFIC SUPPORT ACTION

Thematic Priority 2

D12: IPv6 Training Material

Due date of deliverable: 31st March 2006

Original submission date: 27th April 2006

Version V1.1 submission date: 16th November 2006

Start date of project: 1st April 2005

Duration: 30 months

Organization name of lead contractor for this deliverable:

Cisco

Revision: V1.1

Abstract

This deliverable describes the material that will be made available to participants of the IPv6 technical training courses, which are intended mainly for ISP operational staff.

The document contains a description of the test laboratories in Brussels and Paris, followed by details of the exercises that will be performed, in order to give the attendees a deep and practical understanding of how to manage in a practical way the deployment of the following important fundamental IPv6 features:

- Basic IPv6 commands
- Enabling IPv6 on client terminals
- IPv6 routing protocols (OSPF, IS-IS, BGP and Multicast) for the Brussels 6DISS Lab
- IPv6 Routing protocols for the Paris 6DISS Lab
- IPv6 addressing and routing protocols overview for the Brussels 6DISS Lab
- Transitioning with tunnels
- Multicast using the Paris 6DISS Lab
- Other exercises (DHCPv6, Basic IPv6, HTTPD, Traffic Filters, Stateless configuration for clients)

Project co-funded by the European Commission within the Sixth Framework Programme (2002-2006)		
Dissemination Level		
PU	Public	✓
PP	Restricted to other programme participants (including the Commission Services)	
RE	Restricted to a group specified by the consortium (including the Commission Services)	
CO	Confidential, only for members of the consortium (including the Commission Services)	

History

V1.1 includes information about the configuration commands for the Alcatel equipment (sub-section 3.1.2.2 added).

Table of Contents

1	INTRODUCTION	5
2	LAYOUT OF THE LABORATORIES IN BRUSSELS AND PARIS, AND THE PROCEDURES FOR RESERVING THEM	6
2.1	THE BRUSSELS LABORATORY	6
2.1.1	Reservation procedure.....	6
2.1.2	Accessing the Brussels laboratory	6
2.1.3	Configuring the equipment for the experiments.....	6
2.2	THE PARIS TESTBED.....	14
2.2.1	Reservation procedure.....	14
2.2.2	Accessing the Paris testbed.....	15
2.2.3	Configuring the equipment for the experiments.....	15
3	BASIC IPV6 COMMANDS.....	16
3.1.1	Objective of the Exercise	16
3.1.2	Basic configurations.....	16
4	ENABLING IPV6 ON CLIENT TERMINALS.....	23
4.1	IPV6 SUPPORT IN CLIENTS - SUMMARY	23
4.1.1	Objective of the Exercise	23
4.1.2	Lab Exercise	23
5	IPV6 ROUTING PROTOCOLS.....	30
5.1	OSPFV3 USING THE BRUSSELS LAB	30
5.1.1	Lab Topology	30
5.1.2	Objective of the Exercise	30
5.1.3	Lab Exercise	30
5.2	IS-ISV6 USING THE BRUSSELS LAB	35
5.2.1	Lab Topology	35
5.2.2	Objective of the Exercise	35
5.2.3	Lab Exercise	35
5.3	ROUTING (AGGREGATION)	42
5.3.1	Objective of the Exercise	42
5.3.2	Lab Exercise	42
5.4	BGP FOR IPV6 USING THE BRUSSELS LAB	43
5.4.1	Lab Topology	43
5.4.2	Objective of the Exercise	43
5.4.3	Lab Exercise	43

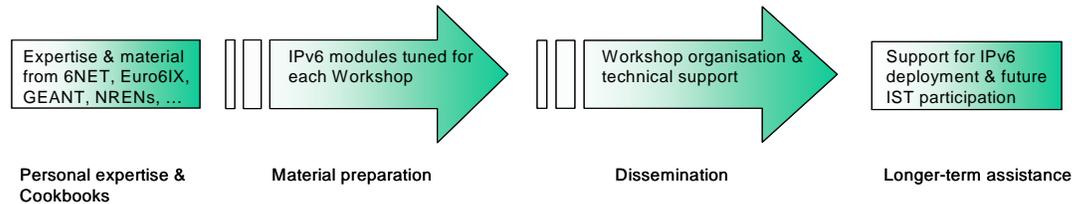
5.5	IPv6 PREFIX FILTERS	54
5.5.1	Lab Exercise	54
5.6	MULTICAST BGP FOR IPV6 USING THE BRUSSELS LAB	55
5.6.1	Lab Topology	55
5.6.2	Objective of the Exercise	55
5.6.3	Lab Exercise	56
5.7	ROUTING (TUNNELLING)	57
5.7.1	Objective of the Exercise	57
5.7.2	Lab Exercise	57
6	ENABLING IPV6 ADDRESSING AND ROUTING	58
6.1	PARIS 6DISS LAB	58
6.1.1	Lab Topology	58
6.1.2	Lab Exercise	58
6.2	BRUSSELS 6DISS LAB	61
6.2.1	Lab Topology	61
6.2.2	Lab Exercise	62
6.3	ADDRESSING CONFIGURATION	62
7	IPV6 TRANSITIONING	66
7.1	TRANSITIONING WITH TUNNELS	66
7.1.1	Lab Topology	66
7.1.2	Objective of the Exercise	66
7.1.3	Lab Exercise	66
8	MULTICAST (USING THE PARIS 6DISS LAB)	68
8.1	PART I - MULTICAST ON THE LINK	68
8.1.1	Lab Exercise	68
8.2	PART II - CONFIGURING PIM / MBGP	68
8.2.1	Lab Exercise	68
8.3	PART III - MONITORING AND TROUBLESHOOTING	69
8.3.1	Lab Exercise	69
8.4	RECEIVING VIDEO WITH IPV6 MULTICAST	70
8.4.1	Objective of the Exercise	70
8.4.2	Lab Exercise	70
9	OTHER EXERCISES	71
9.1	LAB TOPOLOGY	71
9.2	DHCPV6 EXPERIMENTS	71
9.2.1	Objective of the Exercise	71
9.2.2	Lab Exercise	71
9.3	BASIC STATIC IPV6 CONNECTIVITY EXPERIMENTS	72



9.3.1	Objective of the Exercise	72
9.3.2	Lab Exercise	72
9.4	HTTPD EXPERIMENTS	73
9.4.1	Objective of the Exercise	73
9.4.2	Lab Exercise	73
9.5	ACCESS CONTROL EXPERIMENTS	74
9.5.1	Objective of the Exercise	74
9.5.2	Lab Exercise	74
9.6	STATELESS AUTOCONFIGURATION EXPERIMENTS	74
9.6.1	Objective of the Exercise	74
9.6.2	Lab Exercise	74
10	OTHER TECHNICAL TRAINING MATERIAL AVAILABLE FROM 6DISS	75
10.1	CISCO IPV6 TECHNICAL E-LEARNING MATERIAL	75
11	CONCLUSION	76

1 Introduction

The methodology behind the 6DISS approach to giving support for IPv6 deployment through dissemination and other forms of assistance is shown below:



Dissemination through workshops is one of the major activities in the project, but dedicated “hands-on” courses are also given for engineers that will have to work with IPv6 networks. These courses will be given either in Brussels by Cisco staff, in Paris, by RENATER staff, or on site.

The attendees will be trained on how to configure and operate IPv6 clients and routers. Support will be made available from the appropriate engineers for answering specific technical questions.

The course comprises both slides and hands-on sessions. It is intended for engineers and network managers (especially from ISPs) who will work with equipment on a daily basis, and who want a deeper technical training on IPv6 configuration and management. The main objectives of this “complementary, non-workshop training” are:

- To develop an IPv6 training course for engineers (e.g. deployment engineers, maintenance engineers, NOC¹ personnel)
- To give IPv6 training to engineers (e.g. deployment engineers, maintenance engineers, NOC personnel) in a testbed environment

The training course will typically last 1 week and will cover the same items as in the workshops, but with more focus on hands-on practical examples. Equipment from Cisco, Alcatel and Juniper is available. Typically, the course is suitable for up to 20 people.

This deliverable begins with a description of the two testbed laboratories in Brussels and Paris (Section 2), and then explains each of the topics covered, through a detailed description of the exercises that will be performed to demonstrate the procedures necessary to establish an IPv6 network. i.e.:

Section 3: Basic IPv6 commands

Section 4: Enabling IPv6 on client terminals

Section 5: IPv6 routing protocols (OSPF, IS-IS, Aggregation, BGP, Prefix filters, Multicast and Tunnelling)

Section 6: Enabling IPv6 addressing and routing in the Paris and Brussels labs

Section 7: Transitioning with tunnels

Section 8: Multicast using the Paris 6DISS Lab

Section 9: Other exercises (DHCPv6, Basic IPv6, HTTPD, Traffic Filters, Stateless configuration for clients)

¹ Network Operations Centre

2 Layout of the laboratories in Brussels and Paris, and the procedures for reserving them

This section contains information concerning how to reserve, access and configure the laboratories in Brussels and Paris.

2.1 The Brussels laboratory

Cisco Systems Belgium
Pegasus Park
De Kleetlaan 7
1831 Diegem
Belgium

The Brussels 6DISS lab is available for anyone who wants to obtain some hands-on experience and can be used for IPv6 tests that may disturb traffic. It is therefore especially useful for testing transitioning mechanisms. It is possible to request the availability of support for certain IPv6 features by contacting the lab manager in advance. The lab can then be configured accordingly. The lab also contains traffic generators. If needed, network convergence and disruptive restoration can also be experimented and tested.

2.1.1 Reservation procedure

The Brussels laboratory can be reserved via the URL: <http://www.6diss.org/labs> or by sending an e-mail request to lab@6diss.org. The reservation will typically start 1 week before the training course, during the course and 1 week after the course. The request should be sent at least 2 weeks before the access is required.

The Brussels testbed will be reserved by the responsible lab manager (Gunter Van de Velde) on a “first-come, first-served” basis; priority will be given for the official 6DISS workshops and training courses.

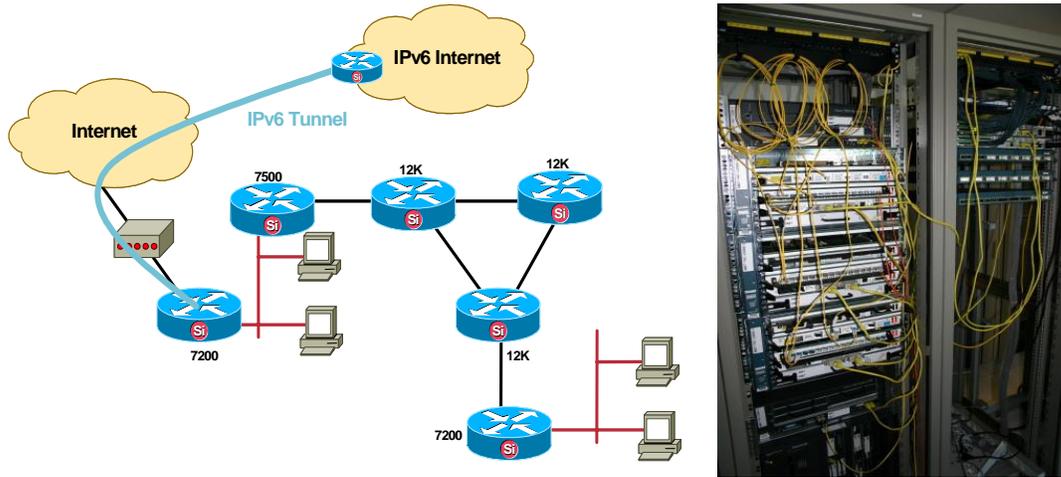
2.1.2 Accessing the Brussels laboratory

Access page of the testbed at Cisco is <http://www.6diss.org/labs>

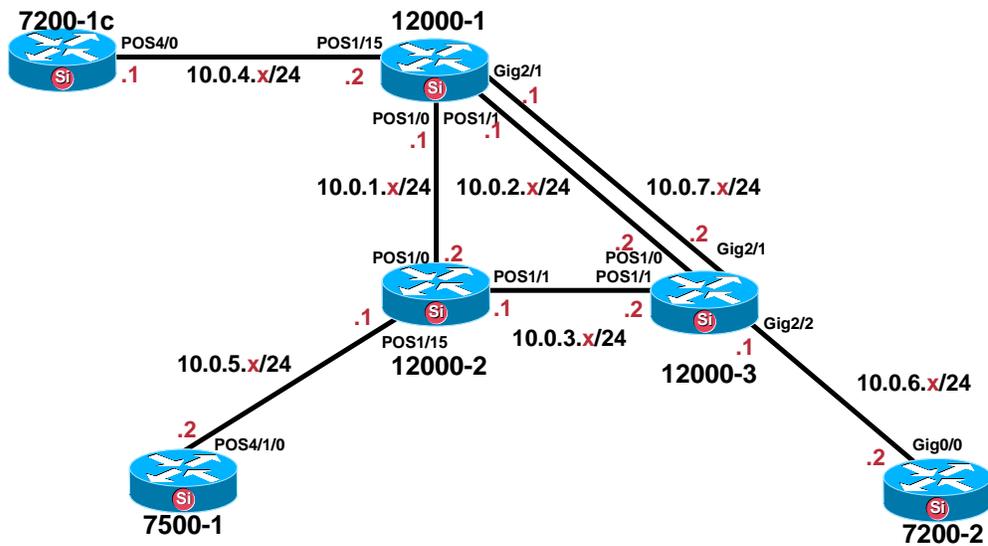
The external access is via a 1Mbps SDSL link, to a public address. The console IP address and password will be given to the trainers.

2.1.3 Configuring the equipment for the experiments

The lab contains a file-server and Cisco routing devices. The default configuration does not contain any IPv6 implementation. This is done deliberately to allow the trainee to start from the basic configuration to simulate real-life transitioning for his own managed network. The lab can be configured for either an IPv6 client environment or for a pure router environment. All IPv6 router components (OSPFv3, IS-ISv6, BGP, Security, ...) in an IPv6 network infrastructure can be tested on this network.



Brussels lab network diagram



Loopback addresses:

12000-1	10.1.1.1	7200-1	10.1.1.4	7500-1	10.1.1.6
12000-2	10.1.1.2	7200-2	10.1.1.5		
12000-3	10.1.1.3				

There is no default configuration; IPv6 has to be configured per laboratory session, as follows:

12000-1#sh ip route

Codes:

C – connected S - static I - IGRP R - RIP M - mobile B - BGP D - EIGRP EX - EIGRP external	O - OSPF IA - OSPF inter area N1 - OSPF NSSA external type 1 N2 - OSPF NSSA external type 2 E1 - OSPF external type 1 E2 - OSPF external type 2 E – EGP	i - IS-IS su - IS-IS summary L1 - IS-IS level-1 L2 - IS-IS level-2 ia - IS-IS inter area * - candidate default U - per-user static route o - ODR
---	---	---

The Gateway of last resort is not set

```

10.0.0.0/8 is variably subnetted, 17 subnets, 2 masks
i L1 10.1.1.2/32 [115/20] via 10.0.1.2, POS1/0
C    10.0.2.0/24 is directly connected, POS1/1
i L1 10.1.1.3/32 [115/10] via 10.0.7.2, GigabitEthernet2/1
      [115/10] via 10.0.2.2, POS1/1
i L1 10.0.3.0/24 [115/20] via 10.0.7.2, GigabitEthernet2/1
      [115/20] via 10.0.1.2, POS1/0
      [115/20] via 10.0.2.2, POS1/1
C    10.0.1.2/32 is directly connected, POS1/0
C    10.0.2.2/32 is directly connected, POS1/1
C    10.1.1.1/32 is directly connected, Loopback0
C    10.0.1.0/24 is directly connected, POS1/0
i L1 10.1.1.6/32 [115/30] via 10.0.1.2, POS1/0
i L1 10.0.6.0/24 [115/20] via 10.0.7.2, GigabitEthernet2/1
      [115/20] via 10.0.2.2, POS1/1
C    10.0.7.0/24 is directly connected, GigabitEthernet2/1
i L1 10.1.1.4/32 [115/20] via 10.0.4.1, POS1/15
i L2 10.0.5.1/32 [115/30] via 10.0.1.2, POS1/0
C    10.0.4.0/24 is directly connected, POS1/15
i L1 10.1.1.5/32 [115/30] via 10.0.7.2, GigabitEthernet2/1
      [115/30] via 10.0.2.2, POS1/1
i L1 10.0.5.0/24 [115/20] via 10.0.1.2, POS1/0
C    10.0.4.1/32 is directly connected, POS1/15
12000-1#

```

2.1.3.1 Configuration of the 12000 Routers

12000-1	12000-2	12000-3
12000-1#sho run	12000-2#sho run	12000-3#sho run
Building configuration...	Building configuration...	Building configuration...
Current configuration: 2570 bytes	Current configuration: 2447 bytes	Current configuration: 2521 bytes
!	!	!
version 12.0	version 12.0	version 12.0
no service pad	no service pad	no service pad
service timestamps debug uptime	service timestamps debug uptime	service timestamps debug uptime
service timestamps log uptime	service timestamps log uptime	service timestamps log uptime
no service password-encryption	no service password-encryption	no service password-encryption
!	!	!
hostname 12000-1	hostname 12000-2	hostname 12000-3
!	!	!
boot-start-marker	boot-start-marker	boot-start-marker
boot-end-marker	boot-end-marker	boot-end-marker
!	!	!
redundancy	redundancy	redundancy
mode rpr	mode rpr	mode rpr
enable secret 5	enable secret 5	enable secret 5
\$1\$hUzK\$hWeq/MreR4QaI41Qb	\$1\$3ytu\$MlvmoS/v/g6PEQ2Gn7	\$1\$etw.\$Z9R41Nw4ciwavkS8Tc
wbyN1	vx	yLB0
!	!	!
username 6diss password 0 xxx	username 6diss password 0 xxx	username 6diss password 0 xxx
!	!	!
!	!	!
ip subnet-zero	ip subnet-zero	ip subnet-zero
no ip domain-lookup	no ip domain-lookup	no ip domain-lookup
!	!	!
controller SYSCLOCK 1	controller SYSCLOCK 1	controller SYSCLOCK 1
!	!	!
!	!	!
interface Loopback0	interface Loopback0	interface Loopback0
ip address 10.1.1.1	ip address 10.1.1.2	ip address 10.1.1.3
255.255.255.255	255.255.255.255	255.255.255.255
no ip directed-broadcast	no ip directed-broadcast	no ip directed-broadcast
!	ip router isis	!
!	!	!
interface POS1/0	interface POS1/0	interface POS1/0
ip address 10.0.1.1	ip address 10.0.1.2	ip address 10.0.2.2
255.255.255.0	255.255.255.0	255.255.255.0

<pre> no ip directed-broadcast ip router isis encapsulation ppp crc 16 ! interface POS1/1 ip address 10.0.2.1 255.255.255.0 no ip directed-broadcast ip router isis encapsulation ppp crc 16 ! interface POS1/2 thru POS1/14 no ip address no ip directed-broadcast shutdown crc 16 ! interface POS1/15 ip address 10.0.4.2 255.255.255.0 no ip directed-broadcast ip router isis encapsulation ppp crc 16 clock source internal pos scramble-atm ! interface GigabitEthernet2/0 no ip address no ip directed-broadcast shutdown negotiation auto ! interface GigabitEthernet2/1 ip address 10.0.7.1 255.255.255.0 no ip directed-broadcast ip router isis no negotiation auto ! </pre>	<pre> no ip directed-broadcast ip router isis encapsulation ppp crc 16 ! interface POS1/1 ip address 10.0.3.1 255.255.255.0 no ip directed-broadcast ip router isis encapsulation ppp crc 16 ! interface POS1/2 thru POS1/14 no ip address no ip directed-broadcast shutdown crc 16 ! interface POS1/15 ip address 10.0.5.1 255.255.255.0 no ip directed-broadcast ip router isis encapsulation ppp crc 16 clock source internal pos scramble-atm ! interface GigabitEthernet2/0 no ip address no ip directed-broadcast shutdown no negotiation auto ! interface GigabitEthernet2/1 no ip address no ip directed-broadcast shutdown no negotiation auto ! </pre>	<pre> no ip directed-broadcast ip router isis encapsulation ppp crc 16 ! interface POS1/1 ip address 10.0.3.2 255.255.255.0 no ip directed-broadcast ip router isis encapsulation ppp crc 16 ! interface POS1/2 thru POS1/14 no ip address no ip directed-broadcast shutdown crc 16 ! interface POS1/15 no ip address no ip directed-broadcast encapsulation ppp shutdown crc 16 ! ! ! ! interface GigabitEthernet2/0 no ip address no ip directed-broadcast shutdown negotiation auto ! interface GigabitEthernet2/1 ip address 10.0.7.2 255.255.255.0 no ip directed-broadcast ip router isis no negotiation auto ! </pre>
---	--	--

<pre> interface GigabitEthernet2/2 no ip address no ip directed-broadcast shutdown negotiation auto ! interface GigabitEthernet2/3 no ip address no ip directed-broadcast negotiation auto ! interface Ethernet0 no ip address no ip directed-broadcast shutdown ! router isis net 49.0001.0000.0001.00 passive-interface Loopback0 ! ip classless ! ! line con 0 exec-timeout 0 0 line aux 0 line vty 0 4 login local ! no cns aaa enable end 12000-1# </pre>	<pre> interface GigabitEthernet2/2 no ip address no ip directed-broadcast shutdown no negotiation auto ! interface Ethernet0 no ip address no ip directed-broadcast shutdown ! router isis net 49.0001.0000.0002.00 ! ip classless ! ! line con 0 exec-timeout 0 0 line aux 0 line vty 0 4 login local ! no cns aaa enable end 12000-2# </pre>	<pre> interface GigabitEthernet2/2 ip address 10.0.6.1 255.255.255.0 no ip directed-broadcast ip router isis negotiation auto ! interface GigabitEthernet2/3 no ip address no ip directed-broadcast shutdown negotiation auto ! interface Ethernet0 no ip address no ip directed-broadcast shutdown ! router isis net 49.0001.0000.0003.00 passive-interface Loopback0 ! ip classless ! ! line con 0 line aux 0 line vty 0 4 login local ! no cns aaa enable end 12000-3# </pre>
---	---	---



2.1.3.2 Configuration of the 7200 and 7500 Routers

7200-1	7200-2	7500-1
7200-1#sho run	7200-2#sho run	7500-1#sho run
Building configuration...	Building configuration...	Building configuration...
Current configuration: 1300 bytes	Current configuration: 1028 bytes	Current configuration: 1302 bytes
!	!	!
version 12.3	version 12.3	version 12.3
no service pad	service timestamps debug	service timestamps debug
service timestamps debug	datetime msec	datetime msec
datetime msec	service timestamps log datetime	service timestamps log datetime
service timestamps log datetime	msec	msec
msec	no service password-encryption	no service password-encryption
no service password-encryption	!	service multiple-config-sessions
!	!	no service single-slot-reload-
hostname 7200-1	hostname 7200-2	enable
!	!	!
boot-start-marker	boot-start-marker	hostname 7500-1
boot-end-marker	boot-end-marker	!
!	!	boot-start-marker
enable secret 5	enable secret 5	boot-end-marker
\$1\$UKJg\$J7QWkGLjxAd9yrhE	\$1\$.tYM\$yEpjSOCZ.99.qei0YU	!
OHnEn1	NP3	!
!	!	redundancy
username 6diss password 0 xxx	username 6diss password 0 xxx	mode hsa
no aaa new-model	no aaa new-model	enable secret 5
ip subnet-zero	ip subnet-zero	\$1\$T0Ju\$eXxIeC1cyAxNrmNj/B
!	!	W2r/
!	!	!
ip cef	ip cef	username c password 0 yyy
no ip domain lookup	no ip domain lookup	username 6diss password 0 xxx
!	!	no aaa new-model
!	!	ip subnet-zero
interface Loopback0	interface Loopback0	!
ip address 10.1.1.4	ip address 10.1.1.5	!
255.255.255.255	255.255.255.255	no ip domain lookup
ip router isis	ip router isis	!
!	!	!
interface FastEthernet0/0	interface Ethernet0/0	ip cef
no ip address	no ip address	ip audit po max-events 100

<pre> shutdown duplex auto speed auto no cdp enable ! interface FastEthernet0/1 no ip address shutdown duplex auto speed auto no cdp enable ! interface ATM2/0 no ip address shutdown no atm ilmi-keepalive ! interface POS4/0 ip address 10.0.4.1 255.255.255.0 ip router isis encapsulation ppp pos scramble-atm pos flag c2 22 no cdp enable ! router isis net 49.0001.0000.0004.00 ! ! no cdp run ! ! control-plane ! ! line con 0 exec-timeout 0 0 transport preferred all </pre>	<pre> shutdown duplex auto ! interface GigabitEthernet0/0 ip address 10.0.6.2 255.255.255.0 ip router isis duplex full speed 1000 media-type gbic negotiation auto ! router isis net 49.0001.0000.0005.00 ! ip classless no ip http server ! ! control-plane ! ! line con 0 exec-timeout 0 0 transport preferred all transport output all stopbits 1 line aux 0 line vty 0 4 login local transport preferred all transport input all transport output all ! ! end 7200-2# </pre>	<pre> no ftp-server write-enable ! ! no crypto isakmp enable ! ! interface Loopback0 ip address 10.1.1.6 255.255.255.255 ip router isis ! interface GigabitEthernet1/0/0 no ip address load-interval 30 shutdown negotiation auto ! interface FastEthernet4/0/0 no ip address shutdown duplex auto speed auto ! interface FastEthernet4/0/1 no ip address shutdown duplex auto speed auto ! interface POS4/1/0 ip address 10.0.5.2 255.255.255.0 ip router isis encapsulation ppp clock source internal pos scramble-atm pos flag c2 22 ! interface ATM5/0/0 no ip address shutdown no atm ilmi-keepalive </pre>
--	---	--

<pre> transport output all stopbits 1 line aux 0 transport preferred all transport output all stopbits 1 line vty 0 4 login local transport preferred all transport input all transport output all ! ! end 7200-1# </pre>		<pre> ! router isis net 49.0001.0000.0006.00 ! ip classless no ip http server no ip http secure-server ! ! control-plane ! ! line con 0 line aux 0 line vty 0 4 login local ! ! end 7500-1# </pre>
--	--	---

2.2 The Paris testbed

GIP RENATER
151 Bd de l'Hôpital
75013 Paris
France

2.2.1 Reservation procedure

The Paris testbed availability is shown on the following URL:

http://www.renater.fr/article.php3?id_article=190

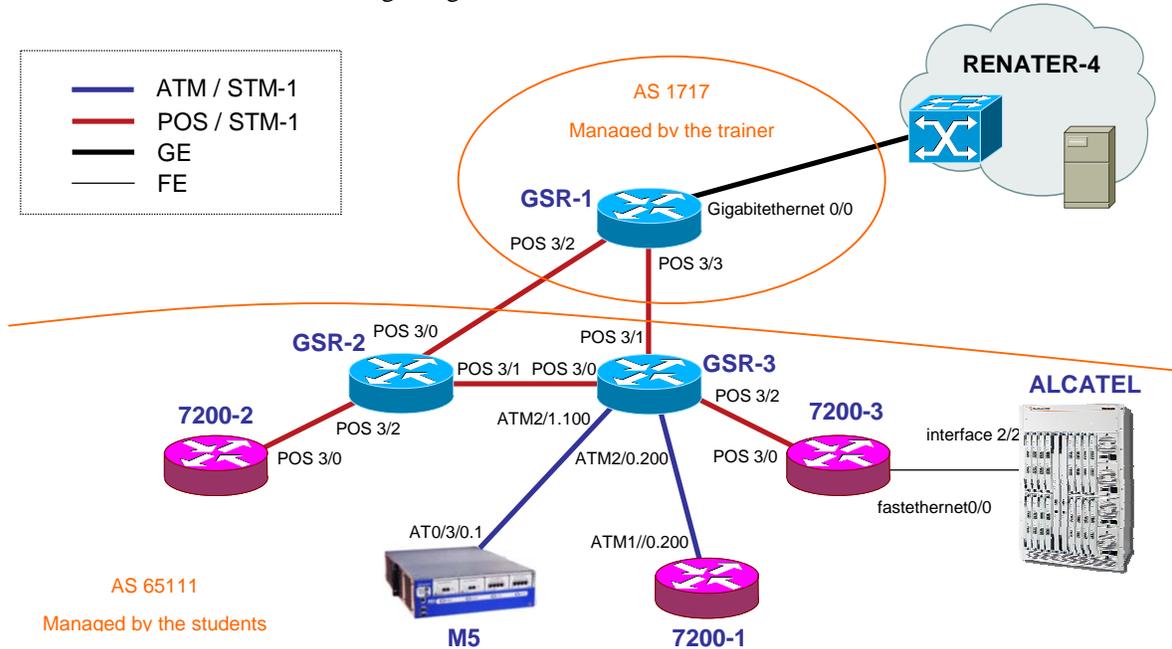
It can be reserved via the URL: <http://www.6diss.org/labs> or by sending an e-mail request to lab@6diss.org. The reservation will typically start 1 week before the training course, during the course and 1 week after the course. The request should be sent at least 2 weeks before the access is required.

The Paris testbed will be reserved by the responsible lab manager (Jérôme Durand) on a “first-come, first-served” basis.

People willing to access the testbed must send a summary of the operations that will be performed. There are no special restrictions on the usage. Nevertheless, according to the requests made, the lab owner reserves the right to decline a request or to ask for some modifications to the initial proposal.

2.2.2 Accessing the Paris testbed

The Paris testbed has the following design:



Paris lab network diagram

Use telnet protocol with the following addresses to login on the routers:

Name	How to connect
GSR-2	194.254.101.5
GSR-3	194.254.101.6
7200-1	194.254.101.12
7200-2	194.254.101.8
7200-3	194.254.101.9
ALCATEL	194.254.101.58
M5	194.254.101.2

Router connection information

Login: 6diss

Password: 6diss

2.2.3 Configuring the equipment for the experiments

The default configuration does not contain any IPv6 implementation. This is done deliberately to allow the trainee to start from the basic configuration to simulate real-life transitioning for his own managed network. The lab can be configured for either an IPv6 client environment or for a pure router environment. All IPv6 router components (OSPFv3, IS-ISv6, BGP, Security, ...) in an IPv6 network infrastructure can be tested on this network.

3 Basic IPv6 Commands

3.1.1 Objective of the Exercise

This exercise will educate the trainees about the basic IPv6 commands for the routing platforms seen most frequently by individuals implementing IPv6 services on a network infrastructure. It will help them when configuring IPv6.

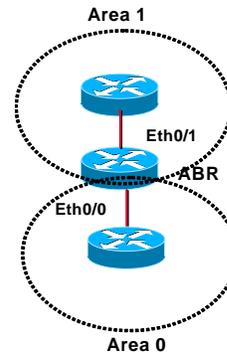
3.1.2 Basic configurations

3.1.2.1 Cisco commands

1. Enable IPv6 on an interface
 - interface xxxxx
 - ipv6 enable
2. Configure an address
 - interface xxxxx
 - ipv6 address X:X:X:X::X/<0-128> (general address)
 - ipv6 address X:X:X:X::X (link-local address)
 - ipv6 address autoconfig (auto-configuration)
- 2a. Example (LAN interface)
 - interface Ethernet0/0
 - ip address 192.168.1.254 255.255.255.0
 - ipv6 address 2001:db8:123:1::2/64
3. Configure an IPv6 in IPv4 tunnel
 - interface tunnel x
 - tunnel source *interface*
 - tunnel destination X.X.X.X
 - ipv6 address X:X:X:X::X/<0-128>
 - tunnel mode ipv6ip (for direct tunneling)
 - tunnel mode gre ip (for gre encapsulation)
4. Configure an IPv6 in IPv6 tunnel
 - interface tunnel x
 - tunnel source *interface*
 - tunnel destination X.X.X.X
 - ipv6 address X:X:X:X::X/<0-128>
 - tunnel mode ipv6 (for direct tunneling)
 - tunnel mode gre ipv6 (for gre encapsulation)
5. Enable IPv6 routing
 - ipv6 unicast-routing
 - interface tunnel x
6. Configure static routes
 - ipv6 route prefix/prefixlen next_hop
- 6a. Example
 - ipv6 route ::/0 2001:db8:10a:1001::1

7. Routing (OSPFv3)

- interface Ethernet0/0
 - ipv6 address 2001:db8:1:1::1/64
 - ipv6 ospf 1 area 0
 - !
 - interface Ethernet0/1
 - ipv6 address 2001:db8:1:2::2/64
 - ipv6 ospf 1 area 1
 - !
 - ipv6 router ospf 1
 - router-id 2.2.2.2



8. Routing (BGP)

- no bgp4 default unicast
- bgp router-id a.b.d.f
- router bgp xxxx
 - neighbor X:X:X:X::X remote-as ...
 - neighbor X:X:X:X::X ...
 - address-family ipv6
 - neighbor X:X:X:X::X activate
 - neighbor X:X:X:X::X ...
 - network 2001:db8::/32
 - no synchronization
 - exit address-family

9. Routing policy filtering

- ipv6 prefix-list bgp-in-6net seq 5 deny ::/0
 - Means filter ::/0 exactly
- ipv6 prefix-list bgp-in-6net seq 10 deny 3FFE:3000::/24 le 28
- ipv6 prefix-list bgp-in-6net seq 15 deny 2001:db8::/35 le 41
- ipv6 prefix-list bgp-in-6net seq 20 permit 2002::/16
- ipv6 prefix-list bgp-in-6net seq 25 permit 3FFE::/17 ge 24 le 24
- ipv6 prefix-list bgp-in-6net seq 30 permit 3FFE:8000::/17 ge 28 le 28
 - Means every prefix matching 3FFE:8000::/17 with length 28
- ipv6 prefix-list bgp-in-6net seq 35 permit 3FFE:4000::/18 ge 32 le 32
- ipv6 prefix-list bgp-in-6net seq 40 permit 2001::/16 ge 32 le 35
 - Means every 2001::/16 derived prefix, with length between 32 and 35

10. Access Control Lists (ACLs)

- ipv6 access-list vty-ipv6
- permit tcp 2001:db8:0:401::/64 any eq telnet
- deny ipv6 any any log-input

11. Applying an ACL to an interface

- ipv6 traffic-filter <acl_name> in | out

12. Restricting access to the router

- ipv6 access-class <acl_name> in | out

13. Applying an ACL to filter debug traffic

- `debug ipv6 packet [access-list <acl_name>] [detail]`

14. Show commands

- `show bgp`
- `show bgp ipv6 unicast/multicast/all summary`
- `show bgp ipv6 neigh <addr> routes`
- `show bgp ipv6 neigh <addr> advertised-routes`
- `show bgp ipv6 neigh <addr> received-routes`
- `show ipv6 route`
- `show ipv6 interface`
- `show ipv6 neighbors`

3.1.2.2 Alcatel commands

1. Enable IPv6 on an interface

The Alcatel OmniSwitch uses IP at the VLAN level through virtual routers. Therefore IP is started after a VLAN is created and after physical interfaces are added to this VLAN. VLANs are initially known by a number defined by the operator. Then a name can be associated to this number. This name is assigned to the virtual interface to which the IPv6 address is given.

- `vlan "number"`
- `show vlan`
- `vlan "number" port default a/b-c`
- `ipv6 interface "name" vlan number`
- `ipv6 address "2001:XXXX::3/prefix" "name"`

Where "a" is the board number and "b-c" the range of number of ports of board "a" associated to the VLAN "number".

2. Configure a tunnel

Like for interfaces, tunnels are associated to VLANs. Note that the IPv4 Tunnel End Point is also a VLAN. The configuration is performed as follows:

- `vlan "number"`
- `vlan "number" default a/b`
- `ip interface "name-v4" vlan "number"`
- `ip interface "name-v4" address D.E.F.G mask H.I.J.K`
- `ipv6 interface "name-v6" tunnel "number"`
- `ipv6 interface "name-v6" tunnel source "@v4" destination "@v4"`
- `ipv6 address "2001:XXXX::3/prefix" "name-v6"`

3. Router Advertisements and auto-configuration

It is possible to show the configuration of an IPv6 VLAN (including Route Advertisement configuration) and to modify any parameter. By default, Route Advertisements are sent when starting an IPv6 VLAN. The command to cancel it (and other commands to modify the behaviour) is shown below:

- `show ipv6 interface "name"`
- `ipv6 interface "name" parameter numerical-value/yes/no`
- `ipv6 interface name ra-send no`
- `ipv6 interface ns-interval value`
- `ipv6 interface ra-interval value`
- ...

3. Starting routing

- `ipv6 route IPv6_prefix/length IPv6_address`

4. Starting and configuring RIP routing

First load and enable RIP process on the switch then start RIP on some interfaces. By default RIPv2 is started.

- `ipv6 load rip`
- `ipv6 rip status enable`
- `ipv6 rip interface "name of the IPv6 VLAN"`
- `ipv6 rip host-route (enable route to hosts)`
- `show ipv6 routes`

Configure RIP

- `ipv6 rip interface "name of the IPv6 VLAN" send-version [v1|v2|v1compatible|none]`
- `ipv6 rip interface "name of the IPv6 VLAN" receive-version [v1|v2|v1compatible|none]`
- `ipv6 rip interface "name of the IPv6 VLAN" metric [1-15]`

3.1.2.3 Juniper commands

1. Interface configuration

- `interfaces {`
 - `name of interface {`
 - `unit x {`
 - `family inet {`
 - `address X.X.X.X/prefixlength;`
 - `}`
 - `family iso {`
 - `address Y.Y.Y.Y.Y.Y;`
 - `}`
 - `family inet6 {`
 - `address Z.Z.Z.Z::Z/prefixlength;`
 - `}`

2. Tunnels: Router advertisements (stateless autoconfiguration)

- `protocols {`
 - `router advertisement {`
 - `interface interface name {`
 - `prefix IPv6_prefix::/prefixlength;`
 - `}`

3. Configure tunnel (with Tunnel PIC)

- `interface {`
 - `ip-x/x/x {`
 - `tunnel {`
 - `source ipv4_source_address;`
 - `destination ipv4_destination_address;`
 - `}`
 - `family inet6 {`
 - `address ipv6_address_in_tunnel/prefixlength;`
 - `gr-x/x/y/z {`

- o unit 0 {...}}
- o }

4. Static routes

- Routing options {
 - o rib inet6.0 { -> Means IPv6 unicast routing table
 - o static {
 - o route *IPv6_prefix* next-hop *IPv6_address*;
 - o }
- Routing options {
 - o rib inet6.0 {
 - o static {
 - o route *IPv6_prefix* discard; -> Useful to originate a network
 - o }

5. Routing (OSPFv3)

- protocols {
 - o ospf3 {
 - o preference 20;
 - o area 0.0.0.0 {
 - o interface ge-0/3/0.808 {
 - o metric 100;
 - o }
 - o interface lo0.0 {
 - o passive;
 - o }}}

6. Routing (BGP)

- protocols {
 - o bgp {
 - o local-as *local_AS_number*;
 - o group *EBGP_peers* {
 - o type external;
 - o family inet6 {
 - o (any | multicast | unicast) }
 - o neighbor *neighbor_IPv6_address*;
 - o peer-as *distant_AS_number*;
 - o import *in-PS*;
 - o export *out-PS*; }

7. Policy routing

- policy statement in PS {
 - o term *from_outside_accept* {
 - o from {
 - o route-filter 2002::/16 exact;
 - o route-filter 3FFE::/17 prefix-length-range /24-/24;
 - o route-filter 3FFE:8000::/17 prefix-length-range /28-/28;
 - o route-filter 3FFE:4000::/18 prefix-length-range /32-/32;
 - o route-filter 2000::/3 prefix-length-range /16-/16;
 - o route-filter 2001::/16 prefix-length-range /29-/35; }
 - o then {
 - o accept; }
 - o then reject; }

8. Show commands

- `show bgp summary`
- `show route advert bgp <addr>`
- `show route rece bgp <addr>`
- `show route table inet6.0 (terse)`
- `show interfaces`
- `show ipv6 neighbors`

3.1.2.4 FreeBSD commands

1. Enable IPv6

- `ipv6_enable="YES"` in `/etc/rc.conf` file
Autoconfiguration is automatically done while the gateway function is off

2. Enable IPv6 forwarding

- `ipv6_gateway_enable="YES"` in `rc.conf` file

3. Add an IPv6 address on an interface

- `ifconfig interface inet6 X:X:X:X::X prefixlen 64`

4. Configure an IPv6 in IPv4 tunnel

- `ifconfig gif1 create`
- `ifconfig gif1 inet6 @IPv6_source @IPv6_dest prefixlen 128`
- `gifconfig gif1 inet @IPv4_source @IPv4_dest`
- `ifconfig gif1 up`

5. Configure an IPv6 in IPv6 tunnel

- `ifconfig gif1 create`
- `ifconfig gif1 inet6 @IPv6_source @IPv6_dest prefixlen 128`
- `gifconfig gif1 inet6 @IPv6_source @IPv6_dest`
- `ifconfig gif1 up`

6. Configure a static route

- Default route
 - `route add -inet6 default fe80::`
 - `route add -inet6 default X:X:X:X::X` (if global address)
- Default route
 - `route add -inet6 X:X:X:X:: -prefixlen YY X:X:X:X::X`
 - `route add -inet6 X:X:X:X:: -prefixlen YY fe80::`

`%interface` notation: If it is a link-local address, then you need to specify on which interface the address is available

7. RIPng: route6d daemon

- `route6d`
 - `-L IPv6_prefix,interface` (receives only prefixes derived from `IPv6_prefix` on interface `interface`)

8. BGP: bgpd daemon

- It is better to use the Zebra BGP daemon

3.1.2.5 *Debian commands*

Main URL: <http://people.debian.org/~csmall/ipv6/>

1. Enable IPv6

- Place "ipv6" in "/etc/modules"
- Edit "/etc/network/interfaces" :
 - iface eth0 inet6 static
 - address 2001:XXXX:YYYY:ZZZ::1
 - netmask 64

2. Tunnel configuration

- Edit "/etc/network/interfaces" :
 - iface tun0 inet6 v4tunnel
 - endpoint A.B.C.D
 - address 2001:XXXX:1:YYYY::2
 - gateway 2001:XXXX:1:YYYY::1
 - netmask 64

3. RA configuration

- Add in "/etc/radvd.conf" :
 - interface eth0
 - {
 - AdvSendAdvert on;
 - AdvLinkMTU 1472;
 - prefix 2001:XXXX:YYYY:ZZZ:/64
 - {
 - AdvOnLink on;
 - AdvPreferredLifetime 3600;
 - AdvValidLifetime 7200;
 - };
 - };
 - };

4 Enabling IPv6 on client terminals

4.1 IPv6 support in clients - Summary

The following table summarises the support for IPv6 in popular client devices

Vendor	IPv6 Support	Versions	More Info
Microsoft	YES	XP and .NET server 2003, CE .NET Pocket PC 2003	http://www.microsoft.com/ipv6
Sun	YES	Solaris 8, 9 and 10	http://www.sun.com/software/solaris/ipv6/
IBM	YES	z/OS Rel. 1.4, AIX 4.3 OS/390 V2R6 eNCS	http://www-3.ibm.com/software/os/zseries/ipv6/
BSD	YES	FreeBSD 4.0 OpenBSD 2.7, NetBSD 1.5 BSD/OS 4.2	http://www.kame.net/
Linux	YES	RH 6.2, Mandrake 8.0, SuSE 7.1, Debian 2.2	http://www.bieringer.de/linux/IPv6/status/IPv6+Linux-status-distributions.html
HP/Compaq	YES	HP-UX 11i Tru64 UNIX V5.1 OpenVMS V5.1	http://h18000.www1.hp.com/ipv6/next_gen.html
Novell	YES	Netware 6.1	http://www.novell.com/documentation/lg/nw65/index.html?page=/documentation/lg/nw65/readme/data/ajzlp6r.html
Apple	YES	MAC OS X 10.2	http://developer.apple.com/macosx/

4.1.1 Objective of the Exercise

This exercise will allow a participant to enable IPv6 on an end-system running frequently used operating systems.

4.1.2 Lab Exercise

4.1.2.1 IPv6 support in Windows Clients

There is full support for IPv6 in:

- Windows XP SP1 and later (Advanced Networking or SP2 is recommended)
- Windows Server 2003

Supported features are:

- autoconfiguration, IPv4 tunnel, 6to4 tunnel, 6to4 relay, ISATAP tunnel, IPSec (manual keying)
- SP1 additions:
 - vendor support
 - GUI installation
 - configuration via netsh command
- SP2 additions
 - Teredo client
 - host-specific relay support
 - IPv6 firewall

1. IPv6 installation on Windows XP:
 - If no service packs are installed:
 - type `ipv6 install` from the command prompt
 - If SP1 is installed:
 - install "Microsoft IPv6 Developer Edition" from the Connection Properties window
 - If SP2 is installed:
 - install "Microsoft TCP/IP version 6" from the Connection Properties window
2. Command for IPv6 configuration
 - `ipv6` (will be discontinued, not present in Windows Server 2003)
 - `netsh interface ipv6`
3. Commands for autoconfiguration
 - `netsh interface ipv6 4`
 - `interface 1 - loopback`
 - `interface 2 - ISATAP`
 - `interface 3 - 6to4 interface`
 - `interface 4 - real network interfaces`
 - `interface 5 - Teredo interface`
4. Commands for setting manual address:
 - `netsh ipv6 interface {add|set} address [interface=] <interface> [address=] <address>`
 - `<interface>` - interface name or index
 - `<address>` - address in IPv6 format
5. Commands for deleting manual address:
 - `netsh ipv6 interface delete address [interface=] <interface> [address=] <address>`
6. Commands for setting/removing static IPv6 routes:
 - `netsh ipv6 interface {add|set|delete} route [prefix=] <prefix>/<length> [interface=] <interface> [[nexthop=] <address>]`
7. Commands associated with applications:
 - `ipconfig`, `netstat`, `ping6`, `tracert6`, `pathping`
 - All Wininet.dll based applications
 - `ftp`, `telnet`, `IExplorer`,
8. Commands on Windows 2003 server:
 - `netsh interface ipv6 (only!)`
 - file/print sharing-et (site-local) supported over IPv6
 - IIS and media server
9. Neighbour cache:
 - `netsh interface ipv6 show neighbors (ipv6 nc)`
10. IPv6 routing table:
 - `netsh interface ipv6 show routes (ipv6 rt)`
11. Reconfiguration:
 - `netsh interface ipv6 renew (ipv6 renew)`
12. Address selection policy

- `netsh interface ipv6 show prefixpolicy`
- `netsh interface ipv6 set prefixpolicy [prefix=]<prefix>/<length> [precedence=]precedence [label=]label`

In Windows XP, you cannot:

- send DNS messages over IPv6 (but Windows Server 2003 has a built-in proxy for it)
- get IPv6 support for file and print sharing (but Windows 2003 can)
- get IPv6 support for the WinInet, IPHelper, and DCOM APIs

13. IPsec

- `ipsec6 sp/sa/s/l`
- No ESP support by default

14. .NET

- IPv6 support, but IPv6 literal address does not work
- IPv6 firewall support after SP2 or Advanced networking pack
- IPv6 Teredo support after SP2 or Advanced networking pack

15. Application

- www.threedegrees.com - instant messaging + p2p stream sharing
- Further information: <http://www.microsoft.com/ipv6/>
- Important! You should switch on IPv6 support if you have IPv6 connectivity

16. Windows XP ICF (same rules for IPv4 and IPv6)

- Show configuration:
 - `netsh firewall show globalport`
 - `netsh firewall show adapter`
- Set configuration
 - `set globalport [port#=enable|disable] [name=name] [protocol=tcp|udp]`
 - `set adapter [name] [icmp type#=enable|disable] [port port#=enable|disable [name=name] [protocol=tcp|udp]] [ignoreglobalport port#=enable|disable] [name=name] [protocol=tcp|udp]] [filtering=enable|disable]`
 - `set logging [filelocation=<location>] [filesize=integer] [droppedpackets=enable|disable] [successfulconnections=enable|disable]`
- After SP2
 - in the firewall you can configure Path MTU discovery support
 - per process configuration possible
- Further information:
 - <http://www.microsoft.com/technet/community/columns/cableguy/cg0104.msp>

4.1.2.2 IPv6 support in *BSD Clients

There is support for:

- autoconfiguration, IPv4 tunnel, 6to4, MLDv1, IPSec, Jumbogram, ICMP mode information query, TRT, privacy extension
... since FreeBSD 4.0, OpenBSD 2.7, NetBSD 1.5
1. KAME extension brings:
 - NAT-PT, DHCPv6, PIM-(S)SM, multicast DNS, EDNS resolver, ISATAP (not any more), anycast (integrated)
 2. FreeBSD
 - Installation: not necessary, the default kernel has it
 - The following must be added in /etc/rc.conf file:
 - o `ipv6_enable="yes"`
 - Autoconfiguration is supported
 - o `ifconfig -a`
 3. Manual address configuration
 - `ipv6_prefix_fxp0="2001:db8:1:2"`
 - `ipv6_ifconfig_fxp0="2001:db8:1:2::1 prefixlen 64"`
 - `then /etc/netstart`
 - `or ifconfig`
 4. Neighbor cache:
 - `ndp -a`
 5. Routing table:
 - `route/netstat`
 6. Configure an IPv6 in IPv4 tunnel
 - `ifconfig gif1 create`
 - `ifconfig gif1 tunnel @IPv4_source @IPv4_dest`
 - `ifconfig gif1 inet6 @IPv6_address up`
 7. Configure an IPv6 in IPv6 tunnel
 - `ifconfig gif1 create`
 - `ifconfig gif1 tunnel @IPv6_source @IPv6_dest`
 - `ifconfig gif1 inet6 @IPv6_address up`
 8. Configuration of further addresses
 - `ipv6_ifconfig_if0_alias0="fec0:0:0:5::2/64"`
 9. Configuration of a static route
 - **Default route**
 - o `route add -inet6 default fe80::X:X:X:X%interface`
 - o `route add -inet6 default X:X:X:X::X (if global address)`
 - **Others**
 - o `route add -inet6 X:X:X:X:: -prefixlen YY X:X:X:X::X`
 - o `route add -inet6 X:X:X:X:: -prefixlen YY`
`fe80::X:X:X:X%interface`

`%interface` notation: If it is a link-local address, then you need to specify on which interface the address is available

10. RIPng: route6d daemon

- `route6d-L IPv6_prefix, interface` (receives only prefixes derived from `IPv6_prefix` on the interface `interface`)
- `route6d-N interface` (do not receive and advertise routes on the interface `interface`)
- `route6d-O IPv6_prefix, interface` (advertise only on the interface `interface` the IPv6 prefix `IPv6_prefix`)

11. Router advertisement

- `router advertisement: /etc/rtadvd.conf`
- `default:\`
`:chlim#64:raflags#0:rltime#1800:rtime#0:retrans#0:\`
`:pinfocflags="la":vltime#2592000:pltime#604800:mtu#auto:`
- `ef0:\` `:addr="2001:db8:ffff:1000::":prefixlen#64:tc=default:`

12. Reconfiguration

- `rtsol fxp0`

13. Applications:

- ping6, traceroute6, ftp, telnet, r* commands, sendmail, apache, Mozilla, proftpd, OpenSSH, LPD, NFS/YP (FreeBSD 5.0 t6l), courier-imap, irc, openldap, tftp, tcpdump, inn, tin

14. Further information:

- <http://www.freebsd.org>
- <http://ipv6.niif.hu/faq>
- <http://www.hs247.com>
- <http://www.kame.net>

4.1.2.3 IPv6 support in Linux Clients

There is support for:

- autoconfiguration, IPv4 tunnel, 6to4
... since Kernel 2.2.x. Recommended is at least 2.4.8

1. USAGI patch (included in 2.6.x series) brings:

- Node information query, anycast, ISATAP, privacy extension, IPSec, applications, bugfix, mobile IP

2. Kernel compile options:

- `CONFIG_IPv6=m/y`
- If the IPv6 module is loaded, the file `/proc/net/ipv6` should be present
- The IPv6 module can be loaded by:
 - o `modprobe ipv6`
- Autoconfiguration is supported
 - o `ifconfig`

3. Address configuration

- `ifconfig <interface> inet6 add <ipv6address>/<prefixlength>`

4. Neighbor cache:
 - `ip -6 neigh show`
5. IPv6 routing table:
 - `route -A inet6/netstat`

4.1.2.4 RedHat configuration

1. Enabling Global IPv6 support
 - `/etc/sysconfig/network` file:
 - `NETWORKING_IPV6="yes"`
2. Enabling IPv6 support on a particular interface
 - `/etc/sysconfig/network-scripts/ifcfg-eth0` file:
 - `IPV6INIT="yes"`
3. Configuring IPv6 interface address
 - `/etc/sysconfig/network-scripts/ifcfg-eth0` file:
 - `IPV6ADDR="3FFE:2F00:20::291D:6A83/48"`
4. Default route configuration
 - `/etc/sysconfig/static-routes-ipv6` file:
 - `eth0 ::/0 3FFE:2F00:20::922:A678`
5. Applications:
 - ping6, traceroute6, tcpdump, tracepath6, apache, bind, imap (xinetd), sendmail, openssh, telnet, ftp, mozilla, lynx, wget, kde, xchat,
6. Further Information:
 - <http://www.bieringer.de/linux/IPv6/>, <http://www.hs247.com>, <http://www.linux-ipv6.org/>

4.1.2.5 Fedora configuration

1. (Fedora Core 2 only)
 - Append to `/etc/sysconfig/network`:
 - `NETWORKING_IPV6=yes`
 - `IPV6_DEFAULTDEV="your exit device e.g. tun6to4"`
2. (Fedora Core 1 only)
 - Append to `/etc/sysconfig/network`:
 - `NETWORKING_IPV6=yes`
 - `IPV6_GATEWAYDEV="your exit device e.g. tun6to4"`
3. 6to4 gateway
 - Append to `/etc/sysconfig/network-scripts/ifcfg-eth0`:
 - `IPV6INIT=yes`
 - `IPV6TO4INIT=yes`

4.1.2.6 Debian configuration

1. Enabling IPv6
 - Type: "ipv6" in "/etc/modules"
2. Address configuration:
 - Type in "/etc/network/interfaces":
 - o iface eth0 inet6 static
 - o address 2001:XXXX:YYYY:ZZZZ::1
 - o netmask 64
3. Firewalls
 - iptables -I INPUT -j ACCEPT --proto 41

4.1.2.7 IPv6 support in Solaris Clients

There has been support since Solaris 8 for:

- autoconfiguration, IPv4 tunnel, 6to4, IPsec, applications
1. Auto-configuration
 - This should already exist. Check by: "/etc/hostname6.<intf>"
 2. Static address configuration
 - Type in "/etc/hostname6.<intf>":
 - o addif 2001:db8:1:2::100 up
 3. Static name ↔ IPv6 address resolution
 - DNS resolution should be enabled in /etc/inet/ipnodes
 - o /etc/nsswitch.conf
 - o ipnodes: files dns

4.1.2.8 IPv6 support in Macintosh Clients

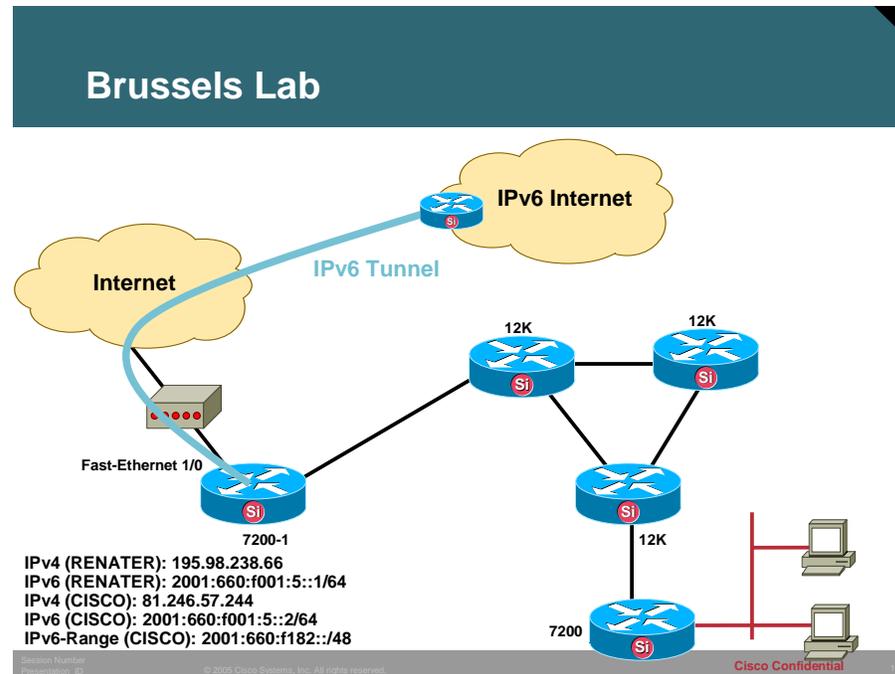
There has been support since MacOSX 10.2 (the Darwin kernel version 6) for:

- autoconfiguration, IPv4 tunnel, 6to4, IPsec, applications, Apple Filing Protocol (since AFP version 3.1)
 - Rendezvous point supports IPv6
1. ip6config command interface
 - Commands:
 - o start-v6 - enable IPv6 on given (all) interface
 - o stop-v6 - disable IPv6 on given (all) interface
 - o start-stf - enable IPv6 as defined in /etc/6to4.conf
 - o start-rtadvd - start router advertisement daemon and enable IPv6 packet forwarding between interfaces
 - ip6 - enable disable per interface
 2. Auto-configuration
 - Enabled by default

5 IPv6 routing protocols

5.1 OSPFv3 using the Brussels Lab

5.1.1 Lab Topology



5.1.2 Objective of the Exercise

The OSPFv3 exercise will show the trainee how to add the OSPFv3 Routing protocol to an existing IPv4 infrastructure. By following the guidelines the participant will enable IPv6 within the network and establish IPv6 connectivity to the IPv6 Unicast Internet.

5.1.3 Lab Exercise

5.1.3.1 Router Login

Use the lab connection instructions to login into the testbed network and open telnet sessions to the different devices.



5.1.3.2 Check the 6DISS Lab Border Router

This step has the goal to make sure that there is IPv6 connectivity to the global IPv6 Internet. On the 6DISS Border Router (7200-1) verify the IPv6 Unicast routing table and confirm Internet connectivity.

```
7200-1#sho ipv6 route summ
IPv6 Routing Table Summary - 630 entries
  4 local, 2 connected, 2 static, 0 RIP, 622 BGP 0 IS-IS, 0 OSPF
Number of prefixes:
  /8: 1, /10: 1, /16: 1, /19: 1, /20: 3, /21: 1, /24: 29, /27: 1
  /28: 28, /29: 1, /30: 2, /32: 516, /35: 23, /48: 17, /64: 3, /128: 2
7200-1#

7200-1#ping 2001:660:3007:419:1::1

Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 2001:660:3007:419:1::1, timeout is 2
seconds:
!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 16/19/24 ms
7200-1#
```

The last check is to verify if there are no IPv6 IGP routes from OSPFv3 at this router

```
7200-1#sho ipv6 route ospf
IPv6 Routing Table - 630 entries
Codes: C - Connected, L - Local, S - Static, R - RIP, B - BGP
       U - Per-user Static route
       I1 - ISIS L1, I2 - ISIS L2, IA - ISIS interarea, IS - ISIS
summary
       O - OSPF intra, OI - OSPF inter, OE1 - OSPF ext 1, OE2 - OSPF
ext 2
       ON1 - OSPF NSSA ext 1, ON2 - OSPF NSSA ext 2
7200-1#

7200-1#sho ipv6 prot summ
Index Process Name
0      connected
1      static
```



```
2    bgp 65001
3    bgp multicast
7200-1#
```

5.1.3.3 Enable IPv6 between 7200-1 and 12000-1

The IPv6 address used for this link is 2001:660:f182:1::/64

1. Enable IPv6 routing on 7200-1 and 12000-1
2. Enable CEF switching on the routers
3. Add the global IPv6 addresses on the
 - a. 7200-1 – POS 4/0 - **2001:660:f182:1::1/64**
 - b. 12000-1 – POS 1/15 - **2001:660:f182:1::2/64**
 - c. Verify connectivity between both routers with ping command
4. Add OSPFv3 routing protocol
 - a. on both routers use: `ipv6 router ospf 1`
 - b. add IPv6 address 2001:660:f182:2::1/128 on loopback 0 of router 12000-1
 - c. enable IPv6 OSPFv3 on the interfaces: `IPv6 ospf 1 area 0`
 - d. Check if the neighborship is established

```
14w0d: %OSPFv3-5-ADJCHG: Process 1, Nbr 10.1.1.4 on POS1/15 from
LOADING to FULL, Loading Done
```

5. Check on 7200-1 if the loopback address is learned correctly?

```
7200-1#sho ipv6 route ospf
IPv6 Routing Table - 634 entries
Codes: C - Connected, L - Local, S - Static, R - RIP, B - BGP
       U - Per-user Static route
       I1 - ISIS L1, I2 - ISIS L2, IA - ISIS interarea, IS - ISIS
summary
       O - OSPF intra, OI - OSPF inter, OE1 - OSPF ext 1, OE2 -
OSPF ext 2
       ON1 - OSPF NSSA ext 1, ON2 - OSPF NSSA ext 2
OE2 2001:660:F182:2::1/128 [110/20]
       via FE80::208:E2FF:FE3C:200, POS4/0
7200-1#
```

On 12000-1 the routing table should look like:

```
12000-1#sho ipv6 route
IPv6 Routing Table - 5 entries
Codes: C - Connected, L - Local, S - Static, R - RIP, B - BGP
I1 - ISIS L1, I2 - ISIS L2, IA - ISIS interarea
      - OSPF intra, OI - OSPF inter, OE1 - OSPF
      ext 1, OE2 - OSPF ext 2
C   2001:660:F182:1::/64 [0/0]
via ::, POS1/15, 00:11:23
L   2001:660:F182:1::2/128 [0/0]
via ::, POS1/15, 00:11:22
LC  2001:660:F182:2::1/128 [0/0]
via ::, Loopback0, 00:05:40
L   FE80::/10 [0/0]
via ::, Null0, 00:11:23
L   FF00::/8 [0/0]
via ::, Null0, 00:11:23
12000-1#
```

6. Enable 7200-1 as default gateway for OSPFv3

- a. Initially check if routing to the Internet is really not working on 12000-1

```
12000-1#ping 2001:660:3007:419:1::1

Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 2001:660:3007:419:1::1, timeout
is 2 seconds:
.....
Success rate is 0 percent (0/5)
```

- b. For this to happen correctly it is needed to allow the OSPFv3 process on router 7200-1 to originate the default information. This needs to be enabled with a specific command

```
7200-1(config)#ipv6 router ospf 1
7200-1(config-rtr)#default-information originate always
7200-1(config-rtr)#
```

- c. The next step is to verify internet connectivity from 12000-1

```
12000-1#ping 2001:660:3007:419:1::1

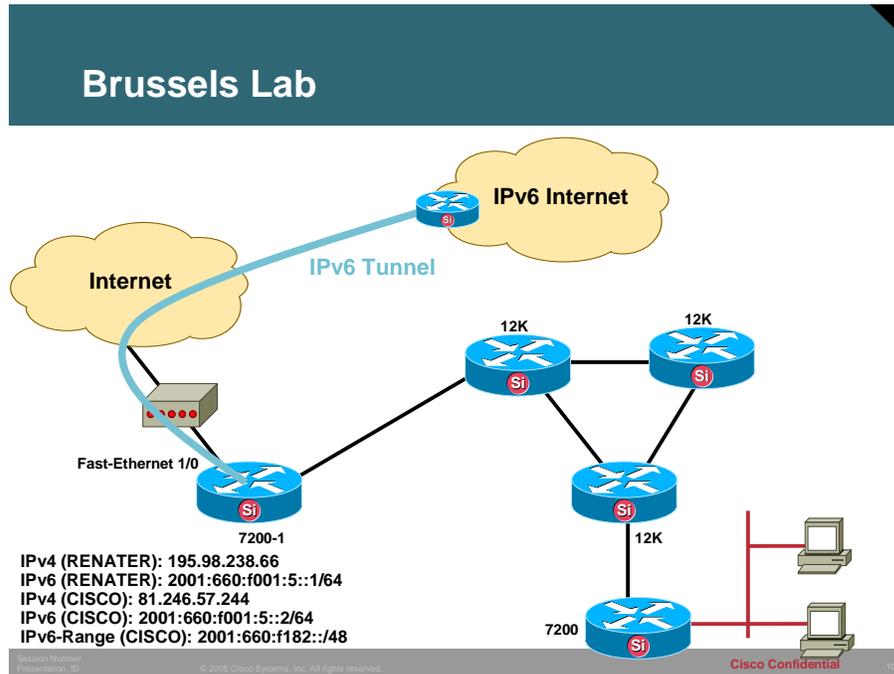
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 2001:660:3007:419:1::1,
timeout is 2 seconds:

!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max =
16/20/24 ms
12000-1#
12000-1#
12000-1#sho ipv6 route
IPv6 Routing Table - 6 entries
Codes: C - Connected, L - Local, S - Static, R - RIP, B - BGP
        I1 - ISIS L1, I2 - ISIS L2, IA - ISIS interarea
        O - OSPF intra, OI - OSPF inter, OE1 - OSPF ext 1, OE2 -
OSPF ext 2
OE2  ::/0 [110/1], tag 1
        via FE80::208:A4FF:FEA7:A408, POS1/15, 00:05:04
C   2001:660:F182:1::/64 [0/0]
        via ::, POS1/15, 00:18:04
L   2001:660:F182:1::2/128 [0/0]
        via ::, POS1/15, 00:18:03
LC  2001:660:F182:2::1/128 [0/0]
        via ::, Loopback0, 00:12:21
L   FE80::/10 [0/0]
        via ::, Null0, 00:18:04
L   FF00::/8 [0/0]
        via ::, Null0, 00:18:04
12000-1#
```

7. Repeat the same steps to enable OSPFv3 in the complete infrastructure

5.2 IS-ISv6 using the Brussels Lab

5.2.1 Lab Topology



5.2.2 Objective of the Exercise

The IS-ISv6 exercise will show the trainee how to add the IPv6 address family when IS-IS Routing protocol is enabled on an existing IPv4 infrastructure. By following the guidelines the participant will enable IPv6 within the network and establish IPv6 connectivity to the IPv6 Unicast Internet.

5.2.3 Lab Exercise

5.2.3.1 Router Login

Use the lab connection instructions to login into the testbed network and open telnet sessions to the different devices.

5.2.3.2 Check the 6DISS Lab Border Router

This step has the goal to make sure that there is IPv6 connectivity to the global IPv6 Internet. On the 6DISS Border Router (7200-1) verify the IPv6 Unicast routing table and confirm Internet connectivity.

```
7200-1#sho ipv6 route summ
```

```
IPv6 Routing Table Summary - 630 entries
```

```
4 local, 2 connected, 2 static, 0 RIP, 622 BGP 0 IS-IS, 0 OSPF
```

```
Number of prefixes:
  /8: 1, /10: 1, /16: 1, /19: 1, /20: 3, /21: 1, /24: 29, /27: 1
  /28: 28, /29: 1, /30: 2, /32: 516, /35: 23, /48: 17, /64: 3, /128: 2
7200-1#

7200-1#ping 2001:660:3007:419:1::1

Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 2001:660:3007:419:1::1, timeout is 2
seconds:
!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 16/19/24 ms
7200-1#
```

The last check is to verify if there are no IPv6 IGP routes from IS-ISv6 at this router

```
7200-1#sho ipv6 route isis
IPv6 Routing Table - 634 entries
Codes: C - Connected, L - Local, S - Static, R - RIP, B - BGP
        U - Per-user Static route
        I1 - ISIS L1, I2 - ISIS L2, IA - ISIS interarea, IS - ISIS
summary
        O - OSPF intra, OI - OSPF inter, OE1 - OSPF ext 1, OE2 -
OSPF ext 2
        ON1 - OSPF NSSA ext 1, ON2 - OSPF NSSA ext 2
7200-1#

7200-1#sho ipv6 prot summ
Index Process Name
0      connected
1      static
2      bgp 65001
3      bgp multicast
7200-1#
```

IS-IS is already configured on both routers for IPv4

```
7200-1
!
router isis
net 49.0001.0000.0004.00
default-information originate
```

```
12000-1
!
router isis
net 49.0001.0000.0001.00
spf-interval 1 120 240
passive-interface Loopback0
!
```

5.2.3.3 Enable IPv6 between 7200-1 and 12000-1

The IPv6 address used for this link is 2001:660:f182:1::/64

1. Enable IPv6 routing on 7200-1 and 12000-1
2. Enable CEF switching on the routers
3. Add the global IPv6 addresses on the
 - a. 7200-1 – POS 4/0 - **2001:660:f182:1::1/64**
 - b. 12000-1 – POS 1/15 - **2001:660:f182:1::2/64**
 - c. Verify connectivity between both routers with ping command
4. Add IPv6 address family to IS-IS routing protocol
 - a. on both routers use:

```
router isis
address-family ipv6
redistribute connected
```
 - b. add IPv6 address 2001:660:f182:2::1/128 on loopback 0 of router 12000-1
 - c. enable IPv6 IS-IS on the interfaces: `ipv6 router isis`
 - d. Check if the neighborship is established

```
7200-1#sho ipv6 route isis
IPv6 Routing Table - 637 entries
```



Codes: C - Connected, L - Local, S - Static, R - RIP, B - BGP
 U - Per-user Static route
 I1 - ISIS L1, I2 - ISIS L2, IA - ISIS interarea, IS -
 ISIS summary
 O - OSPF intra, OI - OSPF inter, OE1 - OSPF ext 1, OE2 -
 OSPF ext 2
 ON1 - OSPF NSSA ext 1, ON2 - OSPF NSSA ext 2
 I1 2001:660:F182:2::1/128 [115/10]
 via FE80::208:E2FF:FE3C:200, POS4/0
 7200-1#**sho isis nei**

System Id Circuit Id	Type	Interface	IP Address	State	Holdtime
12000-1	L1L2	PO4/0	10.0.4.2	UP	28 02

7200-1#

7200-1#**sho isis neighbors det**

System Id Circuit Id	Type	Interface	IP Address	State	Holdtime
12000-1	L1L2	PO4/0	10.0.4.2	UP	28 02

Area Address(es): 49
 SNPA: *PPP*
 IPv6 Address(es): **FE80::208:E2FF:FE3C:200**
 State Changed: 00:05:37
 Format: Phase V
 7200-1#

5. Check on 7200-1 if the loopback address is learned correctly

7200-1#**sho ipv6 route isis**
 IPv6 Routing Table - 637 entries
 Codes: C - Connected, L - Local, S - Static, R - RIP, B - BGP
 U - Per-user Static route
 I1 - ISIS L1, I2 - ISIS L2, IA - ISIS interarea, IS - ISIS
 summary
 O - OSPF intra, OI - OSPF inter, OE1 - OSPF ext 1, OE2 -
 OSPF ext 2
 ON1 - OSPF NSSA ext 1, ON2 - OSPF NSSA ext 2
I1 2001:660:F182:2::1/128 [115/10]

```
via FE80::208:E2FF:FE3C:200, POS4/0
7200-1#
```

On 12000-1 the routing table should look like:

```
12000-1#sho ipv6 route
IPv6 Routing Table - 5 entries
Codes: C - Connected, L - Local, S - Static, R - RIP, B - BGP
       I1 - ISIS L1, I2 - ISIS L2, IA - ISIS interarea
       O - OSPF intra, OI - OSPF inter, OE1 - OSPF ext 1, OE2 -
OSPF ext 2
C   2001:660:F182:1::/64 [0/0]
    via ::, POS1/15, 00:11:01
L   2001:660:F182:1::2/128 [0/0]
    via ::, POS1/15, 00:11:00
LC  2001:660:F182:2::1/128 [0/0]
    via ::, Loopback0, 1w0d
L   FE80::/10 [0/0]
    via ::, Null0, 1w0d
L   FF00::/8 [0/0]
    via ::, Null0, 1w0d
12000-1#
```

6. Enable 7200-1 as default gateway for IS-ISv6

- a. Initially check if routing to the Internet is really not working on 12000-1

```
12000-1#ping 2001:660:3007:419:1::1

Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 2001:660:3007:419:1::1, timeout
is 2 seconds:
.....
Success rate is 0 percent (0/5)
```

- b. For this to happen correctly it is needed to allow the IS-IS process on router 7200-1 to originate the default information. This needs to be enabled with a specific command

```
7200-1(config)#router isis
7200-1(config-router)#default-information originate ?
```

```
route-map Route-map reference
<cr>
```

```
7200-1(config-router)#default-information originate
7200-1(config-router)#
```

- c. The next step is to verify internet connectivity from 12000-1

```
12000-1#
12000-1#ping 2001:660:3007:419:1::1

Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 2001:660:3007:419:1::1,
timeout is 2 seconds:
.....
Success rate is 0 percent (0/5)
12000-1#sho ipv6 route
IPv6 Routing Table - 5 entries
Codes: C - Connected, L - Local, S - Static, R - RIP, B - BGP
        I1 - ISIS L1, I2 - ISIS L2, IA - ISIS interarea
        O - OSPF intra, OI - OSPF inter, OE1 - OSPF ext 1, OE2 -
        OSPF ext 2
C   2001:660:F182:1::/64 [0/0]
    via ::, POS1/15, 00:15:22
L   2001:660:F182:1::2/128 [0/0]
    via ::, POS1/15, 00:15:21
LC  2001:660:F182:2::1/128 [0/0]
    via ::, Loopback0, 1w0d
L   FE80::/10 [0/0]
    via ::, Null0, 1w0d
L   FF00::/8 [0/0]
    via ::, Null0, 1w0d
12000-1#sho ipv6 route 2001:660:3007:419:1::1
% Route not found
12000-1#
```

There is an additional step that needs to be done when creating a default router in IS-IS. This router should in addition to the 'default-information originate' command also have the default IPv6 prefix (= ::/0) in its database. It is a requirement for IS-IS.



```
7200-1(config)#ipv6 route ::/0 null 0
7200-1(config)#router isis
7200-1(config-router)#address-family ipv6
7200-1(config-router-af)#redistribute static
7200-1(config-router-af)#
```

Now on the 12000-1 the default prefix can be seen

```
12000-1#sho ipv6 route
IPv6 Routing Table - 8 entries
Codes: C - Connected, L - Local, S - Static, R - RIP, B - BGP
       I1 - ISIS L1, I2 - ISIS L2, IA - ISIS interarea
       O - OSPF intra, OI - OSPF inter, OE1 - OSPF ext 1, OE2 -
OSPF ext 2
I2 ::/0 [115/10]
   via FE80::208:A4FF:FEA7:A408, POS1/15, 00:00:04
I2 2001:660:3007:400::/64 [115/10]
   via FE80::208:A4FF:FEA7:A408, POS1/15, 00:00:04
I2 2001:660:F182::/48 [115/10]
   via FE80::208:A4FF:FEA7:A408, POS1/15, 00:00:04
C 2001:660:F182:1::/64 [0/0]
   via ::, POS1/15, 00:20:20
L 2001:660:F182:1::2/128 [0/0]
   via ::, POS1/15, 00:20:20
LC 2001:660:F182:2::1/128 [0/0]
   via ::, Loopback0, 1w0d
L FE80::/10 [0/0]
   via ::, Null0, 1w0d
L FF00::/8 [0/0]
   via ::, Null0, 1w0d
12000-1#ping 2001:660:3007:419:1::1
```

Type escape sequence to abort.

```
Sending 5, 100-byte ICMP Echos to 2001:660:3007:419:1::1, timeout
is 2 seconds:
```

```
!!!!!
```

```
Success rate is 100 percent (5/5), round-trip min/avg/max =
20/20/24 ms
```

```
12000-1#
```

7. Repeat the same steps to enable IS-IS in the complete infrastructure
8. For advanced configuration the IS-ISv6 Multi-topology technology can be enabled
9. Use show commands to understand the operation of IS-IS for IPv6

5.3 Routing (Aggregation)

5.3.1 Objective of the Exercise

The objective of aggregation is to show the trainees how to aggregate prefixes when using IS-ISv6.

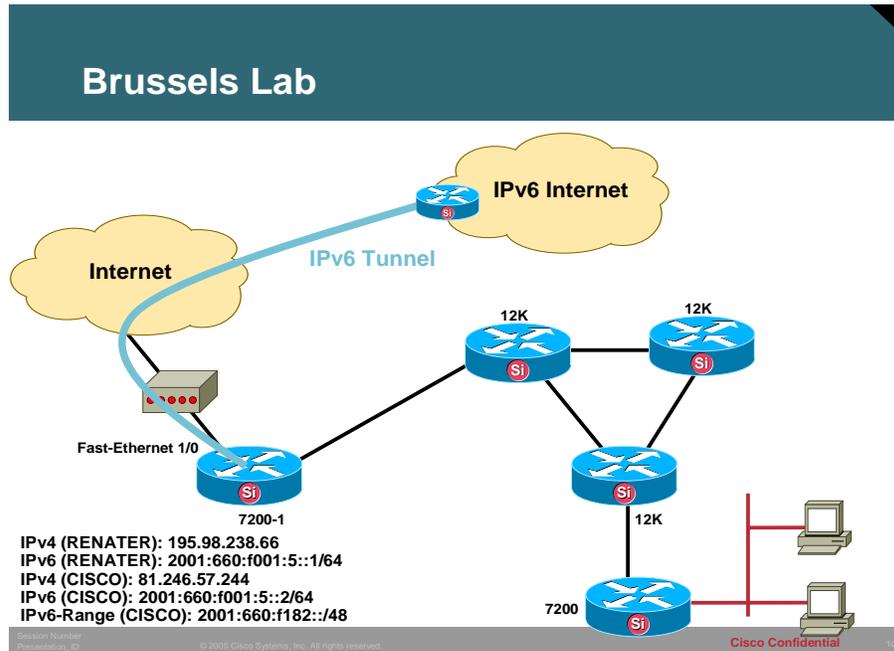
5.3.2 Lab Exercise

5.3.2.1 Perform Aggregation

1. On each Cisco 12000 router perform aggregation so that each group only announces their /56 prefix
 - Create on each 12000 a static route 2001:660:F182:G002::1/128 to the null interface
 - `ipv6 route ::/0 null 0`
 - Enter `router isis` and `address-family ipv6`. Inside there type:
 - `summary-prefix 2001:660:F182:G002::/56` where “G” is the 12000 number
 - repeat `show isis database detail` You should see that you now announce this /56 instead of your downstream /64s
 - Repeat this until you also see that the /64s from other groups are replaced with /56s
 - Static routes etc. can also be aggregated this way

5.4 BGP for IPv6 using the Brussels Lab

5.4.1 Lab Topology



5.4.2 Objective of the Exercise

The BGP for IPv6 exercise will show the trainees how to distribute IPv6 prefixes when using BGP for IPv6. It is assumed that one router has IPv6 Internet connectivity and that IPv4 connectivity is enabled within the initial network. It is also assumed that IPv6 is enabled within the IGP of the 6DISS lab infrastructure. The example below shows how to add BGP to an additional single router, but the trainees are encouraged to add BGP into the complete testbed infrastructure.

5.4.3 Lab Exercise

5.4.3.1 Router Login

Use the lab connection instructions to login into the testbed network and open telnet sessions to the different devices.

5.4.3.2 Check the 6DISS Lab Border Router

This step has the goal to make sure that there is IPv6 connectivity to the global IPv6 Internet. On the 6DISS Border Router (7200-1) verify the IPv6 Unicast routing table and confirm Internet connectivity.

```
7200-1#sho ipv6 route summ
IPv6 Routing Table Summary - 630 entries
  4 local, 2 connected, 2 static, 0 RIP, 622 BGP 0 IS-IS, 0 OSPF
Number of prefixes:
```



```
/8: 1, /10: 1, /16: 1, /19: 1, /20: 3, /21: 1, /24: 29, /27: 1
/28: 28, /29: 1, /30: 2, /32: 516, /35: 23, /48: 17, /64: 3, /128: 2
7200-1#
```

```
7200-1#ping 2001:660:3007:419:1::1
```

Type escape sequence to abort.

Sending 5, 100-byte ICMP Echos to 2001:660:3007:419:1::1, timeout is 2 seconds:

!!!!!

Success rate is 100 percent (5/5), round-trip min/avg/max = 16/19/24 ms

```
7200-1#
```

```
7200-1#sho bgp ipv6 unicast neighbors
```

BGP neighbor is 2001:660:3007:400:1::, remote AS 1717, external link

BGP version 4, remote router ID 194.254.101.4

BGP state = Established, up for 01:39:15

Last read 00:00:27, hold time is 180, keepalive interval is 60 seconds

Neighbor capabilities:

Route refresh: advertised and received(old & new)

Address family IPv4 Unicast: advertised

Address family IPv6 Unicast: advertised and received

Message statistics:

InQ depth is 0

OutQ depth is 0

	Sent	Rcvd
Opens:	1	1
Notifications:	0	0
Updates:	1	590
Keepalives:	102	102
Route Refresh:	0	0
Total:	104	693

Default minimum time between advertisement runs is 30 seconds

For address family: IPv4 Unicast

BGP table version 1, neighbor version 0/0

Output queue sizes : 0 self, 0 replicated

Index 1, Offset 0, Mask 0x2

1 update-group member



	Sent	Rcvd
Prefix activity:	----	----
Prefixes Current:	0	0
Prefixes Total:	0	0
Implicit Withdraw:	0	0
Explicit Withdraw:	0	0
Used as bestpath:	n/a	0
Used as multipath:	n/a	0

	Outbound	Inbound
Local Policy Denied Prefixes:	-----	-----
Total:	0	0

Number of NLRIs in the update sent: max 0, min 0

For address family: IPv6 Unicast

BGP table version 648, neighbor version 648/0

Output queue sizes : 0 self, 1 replicated

Index 1, Offset 0, Mask 0x2

1 update-group member

	Sent	Rcvd
Prefix activity:	----	----
Prefixes Current:	1	626 (Consumes 45072 bytes)
Prefixes Total:	1	650
Implicit Withdraw:	0	22
Explicit Withdraw:	0	2
Used as bestpath:	n/a	626
Used as multipath:	n/a	0

	Outbound	Inbound
Local Policy Denied Prefixes:	-----	-----
Suppressed duplicate:	0	6
Bestpath from this peer:	644	n/a
Total:	644	6

Number of NLRIs in the update sent: max 0, min 0

Connections established 1; dropped 0

Last reset never

External BGP neighbor may be up to 255 hops away.

Connection state is ESTAB, I/O status: 1, unread input bytes: 0

Connection is ECN Disabled

Local host: 2001:660:F001:5::2, Local port: 11000

Foreign host: 2001:660:3007:400:1::, Foreign port: 179

Enqueued packets for retransmit: 0, input: 0 mis-ordered: 0 (0 bytes)

Event Timers (current time is 0x5BC6FC):

Timer	Starts	Wakeups	Next
Retrans	104	0	0x0
TimeWait	0	0	0x0
AckHold	157	113	0x0
SendWnd	0	0	0x0
KeepAlive	0	0	0x0
GiveUp	0	0	0x0
PmtuAger	0	0	0x0
DeadWait	0	0	0x0

iss: 2136562055 snduna: 2136564119 sndnxt: 2136564119 sndwnd: 15909

irs: 363553207 rcvnxt: 363599619 rcvwnd: 15961 delrcvwnd: 423

SRTT: 300 ms, RTTO: 303 ms, RTV: 3 ms, KRTT: 0 ms

minRTT: 16 ms, maxRTT: 300 ms, ACK hold: 200 ms

Flags: active open, nagle

IP Precedence value : 0

Datagrams (max data segment is 516 bytes):

Rcvd: 339 (out of order: 69), with data: 237, total data bytes: 46411

Sent: 374 (retransmit: 0, fastretransmit: 0, partialack: 0, Second Congestion: 0), with data: 374, total data bytes: 17031

7200-1#

7200-1#sho bgp ipv6 unicast summ

BGP router identifier 10.1.1.4, local AS number 65001

BGP table version is 666, main routing table version 666

626 network entries using 90770 bytes of memory

626 path entries using 45072 bytes of memory

613/564 BGP path/bestpath attribute entries using 71108 bytes of memory

601 BGP AS-PATH entries using 20108 bytes of memory

15 BGP community entries using 392 bytes of memory

```

0 BGP route-map cache entries using 0 bytes of memory
0 BGP filter-list cache entries using 0 bytes of memory
BGP using 227450 total bytes of memory
BGP activity 723/26 prefixes, 739/42 paths, scan interval 60 secs

```

```

Neighbor          V    AS MsgRcvd MsgSent   TblVer  InQ  OutQ  Up/Down
State/PfxRcd
2001:660:3007:400:1::
                4  1717   744   138    666    0   0 02:13:47   625
7200-1#

```

The configuration in place for this to happen on the router 7200-1 is a bit special as the 7200-1 is connected to the IPv4 Internet and then connected through an IPv6-in-IPv4 tunnel to the Paris 6DISS lab:

```

!
!
interface Tunnel0
  no ip address
  ipv6 address 2001:660:F001:5::2/64
  tunnel source FastEthernet1/0
  tunnel destination 195.98.238.66
  tunnel mode ipv6ip
!
interface Tunnell
  no ip address
  ipv6 address 2001:660:3007:419:2::2/64
  tunnel source Tunnel0
  tunnel destination 2001:660:3007:400:1::
  tunnel mode gre ipv6
!
interface Loopback0
  ip address 10.1.1.4 255.255.255.255
  ip router isis
!
!
interface FastEthernet1/0
  ip address 81.246.57.244 255.255.255.240
  duplex half

```

```
no cdp enable
!
!
router bgp 65001
no synchronization
bgp log-neighbor-changes
neighbor 2001:660:3007:400:1:: remote-as 1717
neighbor 2001:660:3007:400:1:: ebgp-multihop 255
neighbor 2001:660:3007:400:1:: update-source Tunnel0
neighbor 2001:660:3007:419:1::1 remote-as 1717
neighbor 2001:660:3007:419:1::1 ebgp-multihop 255
neighbor 2001:660:3007:419:1::1 update-source Tunnel1
no auto-summary
!
address-family ipv6
neighbor 2001:660:3007:400:1:: activate
network 2001:660:F182::/48
exit-address-family
!
address-family ipv6 multicast
neighbor 2001:660:3007:419:1::1 activate
exit-address-family
!
ip route 0.0.0.0 0.0.0.0 81.246.57.241
!
!
!
no cdp run
ipv6 route 2001:660:3007:400::/64 Tunnel0
ipv6 route 2001:660:3007:419::/64 Tunnel0
ipv6 route 2001:660:F182::/48 Null0
```

The BGP AS number used for the Brussels testlab is 65001. The first check is to verify if there is correct IPv6 BGP unicast connectivity to the Internet.

```
7200-1#sho ip prot summ
Index Process Name
0      connected
1      static
```

```

2    isis
3    bgp 65001
7200-1(config)#

```

The lab will now implement the BGP setup to all other routers within the lab environment. The command line example shows how to transport IPv6 prefixes over either an IPv4 or an IPv6 TCP session.

The first iBGP example is the distribution of IPv6 prefixes over an IPv4 TCP session. On router 12000-1 the BGP process 65001 is created (assuming the setup will make usage of iBGP). Even if the IPv6 BGP prefixes are exchanged it **is required** that there is full IPv6 connectivity in the network due to the BGP next-hop check as that must result in a valid IPv6 address.

```

12000-1(config)#router bgp 65001
12000-1(config-router)#neigh 10.0.4.1 remote-as 65001
12000-1(config-router)#neigh 10.0.4.1 update-source pos 1/15
12000-1(config-router)#

```

On Router 7200-1 there is need to be configure 12000-1 a neighbor as BGP does not have the capability to dynamically discover its neighboring peers.

```

7200-1(config)#router bgp 65001
7200-1(config-router)#neighbor 10.0.4.2 remote-as 65001
7200-1(config-router)#neighbor 10.0.4.2 update-source pos 4/0
7200-1(config-router)#

```

Once all is configured correctly the BGP session should be established.

On 12000-1: 15w2d: %BGP-5-ADJCHANGE: neighbor 10.0.4.1 Up

On 7200-1: *Feb 9 10:26:13.850: %BGP-5-ADJCHANGE: neighbor 10.0.4.2 Up

Now the session is up and running. Lets now check if the IPv6 prefixes have been exchanged between 7200-1 and 12000-1:

```

12000-1#sho bgp ipv6 uni
12000-1#

```

There are NO IPv6 prefixes communicated from 7200-1 towards 12000-1.

This is because the IPv6 address-family has not been activated yet. This needs to be activated to have both BGP-speaking routers exchange IPv6 prefixes:

```

12000-1#sho bgp ipv6 uni
12000-1#sho ip bgp nei

```

```

BGP neighbor is 10.0.4.1, remote AS 65001, internal link
  BGP version 4, remote router ID 10.1.1.4
  BGP state = Established, up for 00:06:34

```

Last read 00:00:34, last write 00:00:34, hold time is 180, keepalive interval is 60 seconds

Neighbor capabilities:

Route refresh: advertised and received(new)

Address family IPv4 Unicast: advertised and received

Message statistics:

InQ depth is 0

OutQ depth is 0

	Sent	Rcvd
Opens:	1	1
Notifications:	0	0
Updates:	0	0
Keepalives:	9	9
Route Refresh:	0	0
Total:	10	10

Default minimum time between advertisement runs is 5 seconds

For address family: IPv4 Unicast

BGP table version 1, neighbor version 1/0

Output queue sizes : 0 self, 0 replicated

--More--

From the above output is seen that only IPv6 addresses are allowed over the connection.

Let us now activate the IPv6 prefix exchange over the IPv4 session:

In a real-life network environment as the activation of the IPv6 address-family will result in a BGP session flap.

```
12000-1(config)#router bgp 65001
12000-1(config-router)#address-family ipv6 unicast
12000-1(config-router-af)#neigh 10.0.4.1 activate
12000-1(config-router-af)#
15w2d: %BGP-5-ADJCHANGE: neighbor 10.0.4.1 Down Address family activated
15w2d: %BGP-5-ADJCHANGE: neighbor 10.0.4.1 Up

7200-1(config)#router bgp 65001
7200-1(config-router)#add ipv6 unicast
7200-1(config-router-af)#neigh 10.0.4.2 activate
```

Now we can check if the IPv6 addresses are exchanged:

```
12000-1#sho ip bgp nei

BGP neighbor is 10.0.4.1, remote AS 65001, internal link
  BGP version 4, remote router ID 10.1.1.4
  BGP state = Established, up for 00:02:37
  Last read 00:00:37, last write 00:00:37, hold time is 180, keepalive interval is 60 seconds
```

**Neighbor capabilities:****Route refresh: advertised and received (new)****Address family IPv4 Unicast: advertised and received****Address family IPv6 Unicast: advertised and received**

Message statistics:

InQ depth is 0

OutQ depth is 0

	Sent	Rcvd
Opens:	2	2
Notifications:	0	0
Updates:	0	566
Keepalives:	19	19
Route Refresh:	0	0
Total:	21	587

Default minimum time between advertisement runs is 5 seconds

For address family: IPv4 Unicast

BGP table version 1, neighbor version 1/0

--More--

And we see that on 12000-1 we now receive the IPv6 BGP prefixes:

12000-1#**sho bgp ipv6 uni**

BGP table version is 627, local router ID is 10.1.1.1

Status codes: s suppressed, d damped, h history, * valid, > best, i -
internal,

r RIB-failure, S Stale

Origin codes: i - IGP, e - EGP, ? - incomplete

Network	Next Hop	Metric	LocPrf	Weight	Path
*>i2001:200::/32	2001:660:3007:400:1::		0	100	0 1717 2200
20965 3356 6175 2497 2500 i					
*>i2001:208::/32	2001:660:3007:400:1::		0	100	0 1717 2200
20965 11537 7610 i					
*>i2001:218::/32	2001:660:3007:400:1::		0	100	0 1717 2200
5511 2914 i					
*>i2001:220::/32	2001:660:3007:400:1::		0	100	0 1717 2200
20965 24490 9270 i					
*>i2001:228::/35	2001:660:3007:400:1::		0	100	0 1717 2200
20965 1299 1752 2915 i					
*>i2001:238::/32	2001:660:3007:400:1::		0	100	0 1717 2200
20965 3356 6175 2497 9264 17419 i					

--More--

The next exercise will establish a TCP session between the same two routers, but it will use an IPv4 TCP session instead of a IPv6 TCP session. First, the original BGP session configuration should be removed.

```
On 12000-1
12000-1#conf t
Enter configuration commands, one per line. End with CNTL/Z.
12000-1(config)#no router bgp 65001
12000-1(config)#
15w2d: %BGP-5-ADJCHANGE: neighbor 10.0.4.1 Down BGP protocol
initialization
```

```
On 7200-1
7200-1(config)#router bgp 65001
7200-1(config-router)#no neigh 10.0.4.2 remote 65001
7200-1(config-router)#exit
7200-1(config)#exit
7200-1#
```

Next, the IPv6 TCP session for BGP between 7200-1 and 12000-1 needs to be configured. For simplicity reasons, the IPv6 addresses are allocated on the POS interfaces (7200-1: ipv6 address 2001:660:F182:1::1/64 and on 12000-1: 2001:660:F182:1::2/64).

```
On 12000-1:
12000-1(config)#router bgp 65001
12000-1(config-router)#neigh 2001:660:F182:1::1 remote-as 65001
12000-1(config-router)#neigh 2001:660:F182:1::1 update-source pos 1/15
12000-1(config-router)#address-family ipv6 unicast
12000-1(config-router)# neigh 2001:660:F182:1::1 activate
```

```
On 7200-1:
7200-1(config)#router bgp 65001
7200-1(config-router)#neighb 2001:660:F182:1::2 remote 65001
7200-1(config-router)#neighb 2001:660:F182:1::2 update-sou pos 4/0
7200-1(config-router)#address-family IPv6 unicast
7200-1(config-router)#neighb 2001:660:F182:1::2 activate
```

Next the BGP session should come up and the IPv6 BGP prefixes exchanged

```
On 12000-1: 15w2d: %BGP-5-ADJCHANGE: neighbor 2001:660:F182:1::1 Up
On 7200-1: *Feb 9 11:46:32.258: %BGP-5-ADJCHANGE: neighbor
2001:660:F182:1::2 Up
```

To see that the IPv6 prefixes are exchanged, we can check the BGP table on 12000-1:

```
12000-1#sho bgp ipv6 nei

BGP neighbor is 2001:660:F182:1::1, remote AS 65001, internal link
  BGP version 4, remote router ID 10.1.1.4
  BGP state = Established, up for 00:03:59
```



Last read 00:00:26, last write 00:00:58, hold time is 180, keepalive interval is 60 seconds

Neighbor capabilities:

Route refresh: advertised and received(new)

Address family IPv4 Unicast: received

Address family IPv6 Unicast: advertised and received

Message statistics:

InQ depth is 0

OutQ depth is 0

	Sent	Rcvd
Opens:	2	1
Notifications:	0	1
Updates:	0	566
Keepalives:	6	1
Route Refresh:	0	0
Total:	8	569

Default minimum time between advertisement runs is 5 seconds

For address family: IPv6 Unicast

BGP table version 628, neighbor version 628/0

--More--

And on the 12000-1 the BGP table can be seen:

12000-1#**sho bgp ipv6**

BGP table version is 628, local router ID is 10.1.1.1

Status codes: s suppressed, d damped, h history, * valid, > best, i - internal,

r RIB-failure, S Stale

Origin codes: i - IGP, e - EGP, ? - incomplete

Network	Next Hop	Metric	LocPrf	Weight	Path
*>i2001:200::/32	2001:660:3007:400:1::	0	100	0	1717 2200
20965 3356 6175 2497 2500 i					
*>i2001:208::/32	2001:660:3007:400:1::	0	100	0	1717 2200
20965 11537 7610 i					
*>i2001:218::/32	2001:660:3007:400:1::	0	100	0	1717 2200
5511 2914 i					
*>i2001:220::/32	2001:660:3007:400:1::	0	100	0	1717 2200
20965 24490 9270 i					
*>i2001:228::/35	2001:660:3007:400:1::	0	100	0	1717 2200
20965 1299 1752 2915 i					
*>i2001:238::/32	2001:660:3007:400:1::	0	100	0	1717 2200
20965 3356 6175 2497 9264 17419 i					

--More--

Repeat the enabling of BGP over an IPv6 session for the other routers in the lab infrastructure.

5.5 IPv6 Prefix filters

5.5.1 Lab Exercise

This exercise will teach the trainees to enable and verify BGP route filtering. First, an overview is given of the different filtering commands and then the lab will allow the participants to use these filters to filter out prefixes received from the 7200-1 BGP Border Router.

5.5.1.1 BGP Prefix filtering-1

Prefix list filtering (using prefix lists to filter out all but the 2001:DB8:D0::/48 prefix. This can be done in several ways:

- Blocking any prefixes longer than /48:
 - `ipv6 prefix-list denygt48 permit ::/0 le 48`
 - `ipv6 prefix-list denygt48 deny ::/0 le 128`
- Or allowing only the specific /48 prefix:
 - `ipv6 prefix-list Filter48 permit 2001:DB8:D0::/48`
 - `ipv6 prefix-list Filter48 deny ::/0 le 128`
- To check what these would allow, try:
 - `show bgp ipv6 unicast prefix-list denygt48`
 - `show bgp ipv6 unicast prefix-list Filter48`

To apply one of the prefix-lists:

- `type:neighbor <BGP Neighbor> prefix-list denygt48 in under address-family ipv6` in the BGP configuration
- Then type: `clear bgp ipv6 unicast <BGP Neighbor>`
 - This will reset the peering and the prefix-list will be used when it is back
- To check that it is used, try again the commands:
 - `show bgp ipv6 uni neigh <neighbor> routes`
 - `show ipv6 routes`

5.5.1.2 BGP Prefix filtering-2

A less brutal way to do this is by using soft reconfiguration. By using soft reconfiguration, one can add prefix-lists, etc. without resetting the peering.

- Add the following under `address-family ipv6 unicast`:
 - `neigh <IPv4/6 address neighbor> soft-reconfiguration inbound`
- Then type:
 - `clear bgp ipv6 unicast * soft` to reconfigure without resetting the sessions and temporarily losing the routes
- After adding soft reconfiguration, compare:
 - `sh bgp ipv6 uni neigh <neighbor> received-routes`
 - `sh bgp ipv6 uni neigh <neighbor> routes`

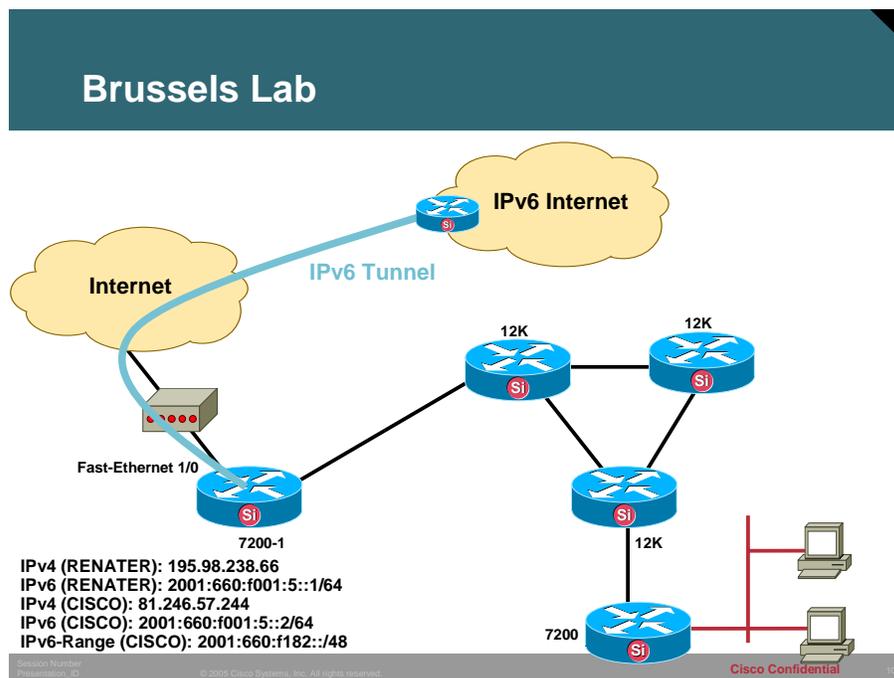
The former shows what we receive, while the latter shows what we actually use.

5.5.1.3 BGP Prefix filtering-3

Each group will look into the received BGP Routing table and search for BGP prefixes that they would like to filter out. The correct result can be verified if after applying the BGP prefix-list filter the BGP prefixes have been removed from the BGP table

5.6 Multicast BGP for IPv6 using the Brussels Lab

5.6.1 Lab Topology



5.6.2 Objective of the Exercise

The IPv6 Multicast BGP for IPv6 exercise will show the participant how to distribute Multicast IPv6 prefixes when using BGP for IPv6. It is assumed that one Router has IPv6 Internet connectivity and that IPv4 connectivity is enabled within the start network. It is also assumed that IPv6 and PIM is enabled within the IGP of the 6DISS testlab infrastructure if Multicast flows are to be tested. The IPv6 Multicast lab will not test Multicast flows. The example below is guided for adding Multicast BGP to an additional single router, but the lab participants are encouraged to add BGP in the complete testbed infrastructure.

The Useful commands and configuration tips contain helpful commands for enabling Multicast in the testbed infrastructure. In addition it will also show participants if there is a local router available with BGP multicast capability to enable multicast on their Linux hosts and to implement a multicast application.

5.6.3 Lab Exercise

5.6.3.1 Router Login

Use the lab connection instructions to login into the lab network and open telnet sessions to the different devices.

5.6.3.2 Check the 6DISS Lab Border Router

This lab exercise is the same as for the IPv6 Unicast lab session with the difference that, within the BGP process, the 'address-family ipv6 multicast' needs to be activated.

5.6.3.3 Useful commands and configuration tips

2. Look at your current configuration, and repeat everything you got under address-family ipv6 unicast in a new section address-family ipv6 multicast.
3. Check the status with the commands:
 - `show bgp ipv6 multicast summary`
 - `show bgp ipv6 multi neighbors <neighbor> advertised-routes`
 - `show bgp ipv6 multicast neighbors <neighbor> routes`
 - `show bgp ipv6 multi neighbors <neighbor> received-routes`
4. We have simply duplicated the unicast config and should see the same as for unicast, except that for multicast we receive a large number of prefixes (as can be seen with received-routes).
 - a. All the routes will then be used, as you can see with:
 - `show bgp ipv6 multicast`
 - b. The multicast routes are used for RPF checks. Try:
 - `show ipv6 rpf <IPv6 prefix>`
5. There is one more step to perform on the routers for multicast to work:
 - a. Enable IPv6 multicast with: ipv6 multicast routing in the global configuration
 - This enables IPv6 PIM-SM and you should do `show ipv6 pim neighbor detail` to verify that you have PIM neighbours
6. Testing SSM and embedded-RP
 - a. We will use a tool called *ssmping* to check multicast connectivity. This checks for SSM, and acts a bit like normal *ping*. A client joins the SSM channel, sends unicast requests and gets unicast and multicast replies
 - b. *ssmping* is available from <http://www.venaas.no/multicast/ssmping/>
 - c. Unfortunately Windows XP does not support IPv6 SSM, so we will only use Linux for this exercise
 - download the latest version of *ssmping*. Unpack the tar file with `tar xvzf ssmping-0.7.tar.gz`. Then you must go into the `ssmping-0.7` directory and type `make`. Next you can run *ssmping*, by typing: `/ssmping`
 - Try to run `./ssmping xiang.ecs.soton.ac.uk` and `./ssmping ssmping.uninett.no`

5.7 Routing (Tunnelling)

5.7.1 Objective of the Exercise

This lab will teach the creation of a tunnel interface on a Cisco Router. It will instruct participants in the different steps to go through and will show some techniques to verify proper tunnel operation. The configuring of tunnels is an important tool when involved in an IPv6 transitioning scenario.

5.7.2 Lab Exercise

1. Create an IPv6-in-IPv4 tunnel from your router to a remote router (called mode `ipv6ip` on IOS)
 - For local end-point (tunnel source) you must for these exercises use the IPv4 address of your upstream interface (Loopback0)
 - The remote tunnel end-point is loopback0 of the remote router
2. Specify the IPv6 address to use on the tunnel interface. You must here use `2001:DB8:D0:8000::G:1/112`, where G is the group digit
 - 112 is the prefix length for the tunnels. In general you can use any value in the range 64-126.
 - The name of the tunnel interface must be `tunnel` followed by a number. If you may be doing multicast, we suggest not starting from 0 (this is because IOS automatically creates tunnel interfaces for IPv6 RPs starting from 0).

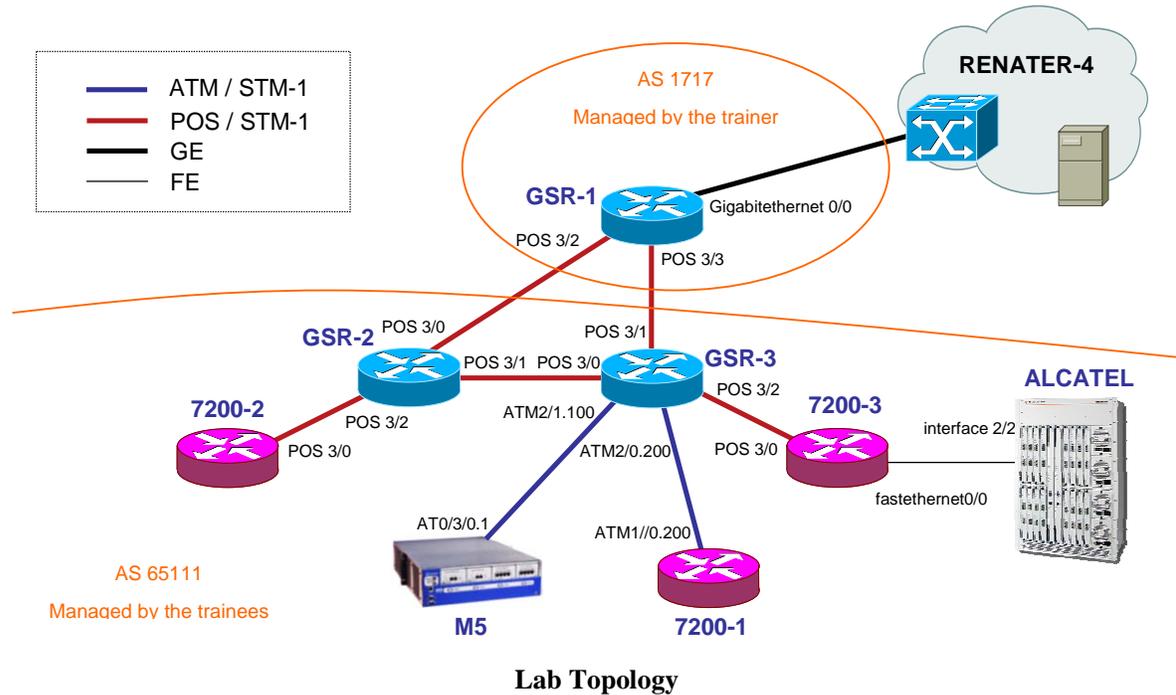
Instructions and expected results

1. For group 7 (for example), the configuration would be:
 - `interface Tunnel100`
 - `ipv6 address 2001:DB8:D0:8000::7:2/112`
 - `tunnel source <my loopback0>`
 - `tunnel destination <remote router loopback0>`
 - `tunnel mode ipv6ip`
2. Verify that it is working:
 - `type: interface Tunnel100`
 - `type: ipv6 address 2001:DB8:D0:8000::7:2/112`
 - `type: show ipv6 interfaces` will show see that it is created
 - ping the remote side of the tunnel. That should be the same address as you have on your interface, but with a "1" at the end. ie: `2001:DB8:D0:8000::7:1`
 - ping the all nodes multicast address `ff02::1` on the tunnel. ie: `type ping` and press enter. Then specify `ipv6` and target address `ff02::1`. Next use the proposed values, but the output interface should be `tunnel100` or whatever you called your interface
3. We can now reach the other side of the tunnel, but we cannot go beyond without adding some routing. One can use BGP for this.

6 Enabling IPv6 addressing and routing

6.1 Paris 6DISS Lab

6.1.1 Lab Topology



6.1.2 Lab Exercise

6.1.2.1 Router Login

Use telnet protocol with the following addresses:

Name	How to connect
GSR-2	194.254.101.5
GSR-3	194.254.101.6
7200-1	194.254.101.12
7200-2	194.254.101.8
7200-3	194.254.101.9
ALCATEL	194.254.101.58
M5	194.254.101.2

Routers connection information

Login: 6diss

Password: 6diss

Recommendations: participants must not change any IPv4 parameter and interfaces specific parameters. They should focus on IPv6 configuration statements.

6.1.2.2 Addressing configuration

1. Configure loopback addresses:

Name	Loopback address
GSR-2	2001:660:3007:8005::1/64
GSR-3	2001:660:3007:8006::1/64
7200-1	2001:660:3007:8012::1/64
7200-2	2001:660:3007:8008::1/64
7200-3	2001:660:3007:8009::1/64
M5	2001:660:3007:8002::1/64

2. Configure interconnection addresses:

Interconexions (R1 - R2)	Prefix
GSR-1 - GSR-2	2001:660:3007:8101::/64
GSR-1 - GSR-3	2001:660:3007:8102::/64
7200-2 - GSR-2	2001:660:3007:8103::/64
GSR-2 - GSR-3	2001:660:3007:8104::/64
GSR-3 - 7200-3	2001:660:3007:8105::/64
7200-3 - ALCATEL	2001:660:3007:8106::/64
GSR-3 - M5	2001:660:3007:8107::/64
GSR-3 - 7200-1	2001:660:3007:8108::/64

R1 has address = prefix::1

R2 has address = prefix::2

- Check you can ping the addresses of the routers connected to the router you manage.
- Take a look at the IPv6 details of an interface of your router. Write down the different IPv6 addresses and prefixes you observe and give their types and usage.

6.1.2.3 Routing configuration

1. IS-IS Routing:

- Use the following ISO addresses for IS-IS configuration:

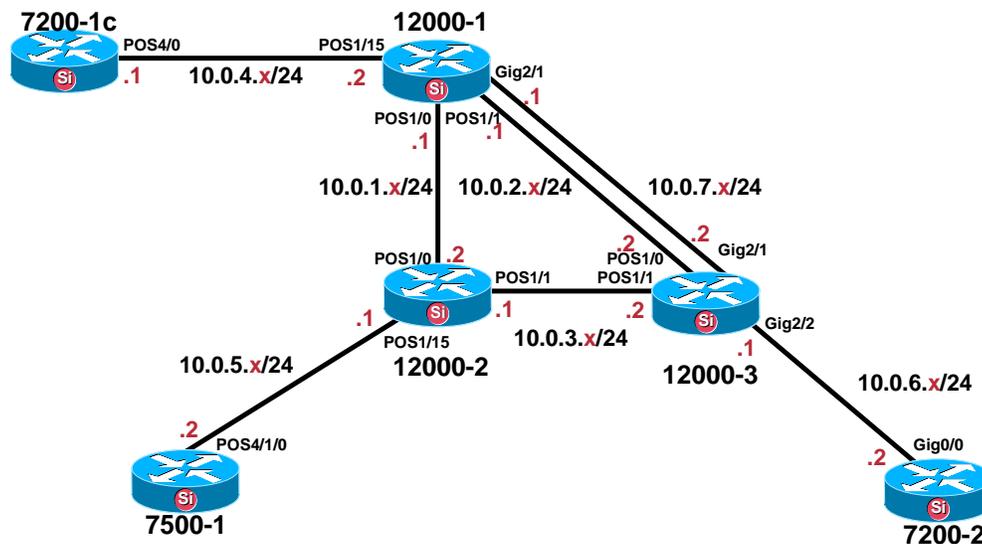
Name	Loopback address
GSR-2	49.0000.0000.0000.0005.00
GSR-3	49.0000.0000.0000.0006.00
7200-1	49.0000.0000.0000.0012.00
7200-2	49.0000.0000.0000.0008.00
7200-3	49.0000.0000.0000.0009.00
M5	49.0000.0000.0000.0002.00

- Enable CEF switching for IPv6 on the Cisco routers.
 - Enable IS-IS on all interfaces connecting 2 routers of the labs.
 - Don't configure IS-IS on interfaces toward GSR-1 router. Configure IS-IS level 1 only.
 - Check the IS-IS neighborship.
 - Make sure you advertise the loopback prefixes via IS-IS.
 - Check all routers receive all interconnection and loopback prefixes via IS-IS.
 - Check the reachability of all routers' loopback addresses from your router using the ping command.
 - Configure a default route on GSR-2 toward GSR-1.
 - Configure GSR-2 to advertise the IPv6 default route with IS-IS.
 - Note that you still won't have connectivity to the IPv6 internet as there is no static route on GSR-1 but a BGP peering. We will get IPv6 internet connectivity when we finish the BGP exercise.
2. BGP configuration for IPv6:
- Configure an eMBGP peering between GSR-2 and GSR-1 and another peering between GSR-3 and GSR-1.
 - For this purpose, interconnection addresses must be used to setup the peerings. Also note that the AS number of GSR-1 is 1717 and that the AS number to be configured in the lab is the private AS number 65111.
 - Configure an iMBGP full mesh between all routers of the labs.
 - Note that the iMBGP full mesh is configured between loopback addresses of the routers.
 - Check the status of the eMBGP and iMBGP peerings.
 - They must be in an *established* state before going to the next step.
 - Check that you receive prefixes via the eMBGP peerings.
 - Check they are properly propagated to all the routers of the lab through iMBGP peerings.
 - Advertise the prefix 2001:660:3007:8000::/49 to GSR-1 via the 2 eMBGP peerings.
 - Make sure it is advertised and received on GSR-1. Ask the trainer for that purpose.

- Check the connectivity to the IPv6 internet.
 - Use the ping / traceroute commands from the routers to some well known IPv6 web servers, e.g.:
 - www.6diss.org
 - www.renater.fr
 - www.kame.net
- Enforce policies on the eMBGP peerings to accept only legacy IPv6 prefixes.
 - Some more details about these legacy prefixes and the way to configure policies can be found at <http://www.space.net/~gert/RIPE/ipv6-filters.html>
- Apply a policy to prefer the path between GSR-1 and GSR-2
 - For this purpose, configure on GSR-2 the local-preference 200 on prefixes received from GSR-1. Configure on GSR-3 the local-preference of 150 on prefixes received from GSR-1.
- Check the BGP details to make sure the policy is properly configured.
 - Using traceroute command, make sure that the path between GSR-2 and GSR-1 is preferred.

6.2 Brussels 6DISS Lab

6.2.1 Lab Topology



Lab Topology

6.2.2 Lab Exercise

6.2.2.1 Router Login

Use telnet protocol with the following addresses and ports:

Name	How to connect
12000-1	81.246.57.242 port 2034
12000-2	81.246.57.242 port 2035
12000-3	81.246.57.242 port 2036
7200-1	81.246.57.242 port 2033
7200-2	81.246.57.242 port 2037
7500-1	81.246.57.242 port 2038

Routers connection information

Login: 6diss

Password: 6diss

6.3 Addressing configuration

6.3.1.1 Addressing configuration

1. Configure loopback addresses:

Name	Loopback address
12000-1	2001:660:f182:1001::1/64
7200-1	2001:660:f182:1002::1/64
12000-2	2001:660:f182:2001::1/64
12000-3	2001:660:f182:2002::1/64
7200-2	2001:660:f182:2003::1/64
7500-1	2001:660:f182:2004::1/64

2. Configure interconnection addresses:

Interconnections (R1 - R2)	Prefix
12000-1 - 12000-2	2001:660:f182:1011::/64
12000-1 - 12000-3 (POS)	2001:660:f182:1012::/64
12000-1 - 12000-3 (GE)	2001:660:f182:1013::/64
12000-1 - 7200-1	2001:660:f182:1014::/64
12000-2 - 12000-3	2001:660:f182:2011::/64
12000-3 - 7200-2	2001:660:f182:2012::/64
12000-2 - 7500-1	2001:660:f182:2013::/64

R1 has address = prefix::1

R2 has address = prefix::2

- Check you can ping the addresses of the routers connected to the router you manage.
 - Note you can only ping interconnection addresses at this stage.
- Take a look at the IPv6 details of an interface of your router. Write down the different IPv6 addresses and prefixes you observe and give their types and usage.
- On 7200-1, a tunnel should be configured for connecting the lab to the IPv6 internet. The tunnel end point offering the Internet connectivity is a tunnel broker located in RENATER backbone.
 - If this tunnel is not configured, use the following parameters for the setup:
 - Tunnel interface number: Tunnel0
 - Tunnel end point destination address: 193.51.182.109
 - Tunnel source interface: FastEthernet1/0
 - IPv6 address to configure on the tunnel interface: 2001:660:F001:5::2/64
 - Tunnel mode: IPv6 in IPv4 (protocol 41 encapsulation)

6.3.1.2 Routing configuration

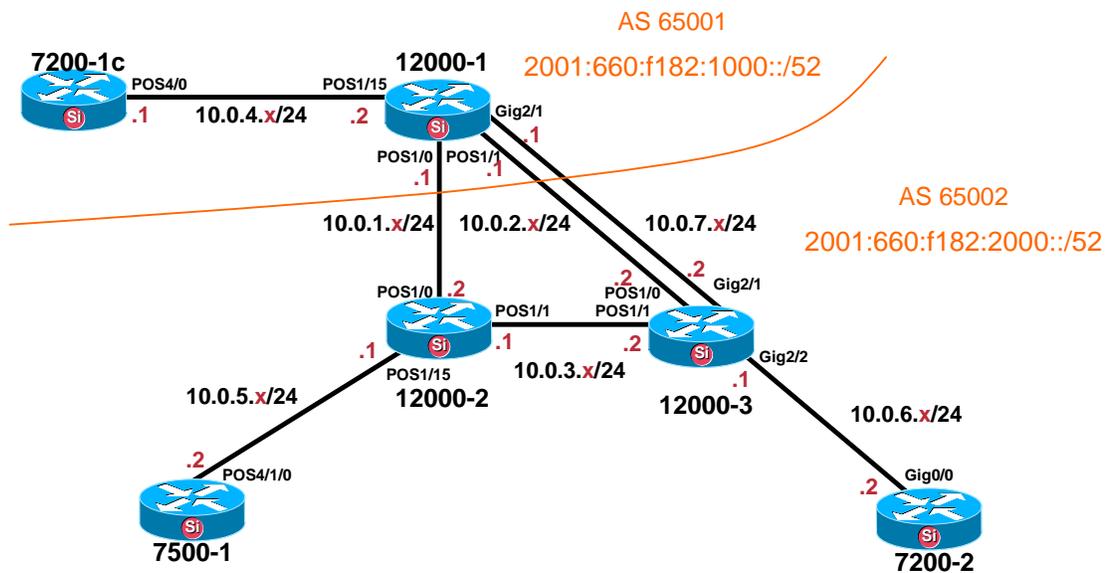
1. OSPF configuration for IPv6:

- Enable OSPFv3 routing protocol for IPv6 on all routers.
- Enable CEF switching for IPv6 on Cisco routers.
- Enable the OSPFv3 process you have configured in bullet point 1 on all interfaces of the lab (except loopback interfaces). Use area 0 for OSPFv3.
- Check OSPFv3 connections are established between routers.
- Redistribute the loopback addresses in OSPFv3.
- Check all routers in the labs receive all interconnection and loopback prefixes via OSPFv3.
- Check the reachability of all routers' loopback addresses from your router using the ping command.

- On 7200-1, configure an IPv6 default route toward the tunnel interface you have configured in the previous addressing exercise. Originate the IPv6 default route on 7200-1, and advertise it using OSPFv3.
- Check that the route is received on the other routers of the lab.
 - Note that you still won't have connectivity to the IPv6 Internet as there is no static route at the other side of the tunnel but a BGP peering. We will get IPv6 internet connectivity when we finish the BGP exercise.

2. BGP configuration for IPv6:

- On 7200-1, remove the IPv6 default route toward the tunnel and stop advertising that route with OSPFv3.
- We will now split the lab in 2 distinct IPv6 Autonomous Systems (ASs) as depicted in the figure below. The AS 65001 will contain the 7200-1 and 12000-1 routers and the AS 65002 will contain the other routers. Remove OSPFv3 configuration on interfaces between the 2 ASs, so that there is no more route information exchanged via an IGP between domains.



Lab Topology with 2 ASs

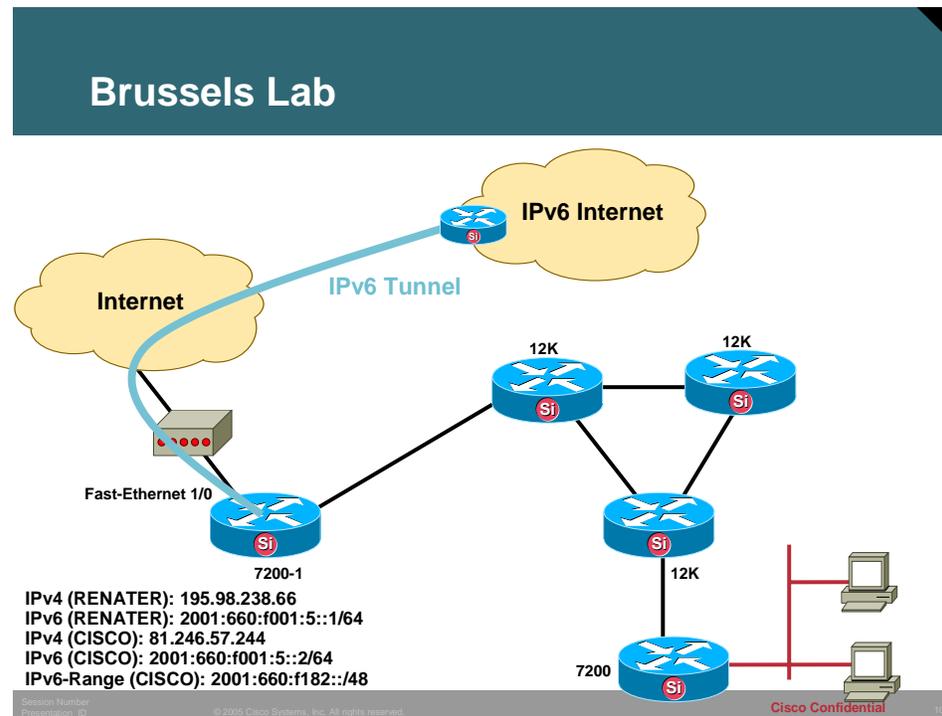
- On 7200-1, configure a multi-hop eMBGP peering toward a router located in RENATER having the entire IPv6 BGP table. Use the following configuration parameters:
 - Distant AS number: 1717
 - Local AS number: 65001
 - Neighbor address: 2001:660:3007:400:1::
 - Source interface: Tunnel0
- Configure also an eMBGP peering between 12000-1 and 12000-2, and another eMBGP peering between 12000-1 and 12000-3.
 - We remind you at this stage that the AS number of 12000-1 is 650001 and that the AS number of 12000-2 and 12000-3 is 65002.
 - We remind also that eBGP peerings are configured using the addresses of the links interconnecting the routers peering together.

- Configure an iMBGP full mesh in each AS.
 - Note that the iMBGP full mesh is configured between the loopback addresses of the routers.
- Check the status of the eMBGP and iMBGP peerings.
 - They must be in an *established* state before going to the next step.
- Check that you receive prefixes via the eMBGP toward RENATER. Check they are properly propagated to 12000-1 through the iMBGP peering.
- Check that 12000-2 and 12000-3 receive properly the prefixes through the eMBGP peering these routers have toward 12000-1.
- Check that 7200-2 and 7500-1 receive properly IPv6 prefixes from the iMBGP peerings they have to 12000-2 and 12000-3.
- Configure 12000-1 and 7200-1 to originate via BGP the prefix 2001:660:f182:1000::/52. Configure 12000-2, 12000-3, 7200-2 and 7500-1 to originate via BGP the prefix 2001:660:f182:2000::/52. Make sure these prefixes are exchanged on the eMBGP peerings.
- Configure 7200-1 to aggregate both /52 prefixes in the prefix 2001:660:f182::/48 that is the only prefix that is accepted by its BGP peer.
- Check the connectivity to the IPv6 internet.
 - Use the ping / traceroute commands from the routers to some well known IPv6 web servers, e.g.:
 - www.6diss.org
 - www.renater.fr
 - www.kame.net
- Enforce policies on the eMBGP peerings to accept only legacy IPv6 prefixes.
 - Some more details about this legacy prefixes and the way you can configure the policy can be found at <http://www.space.net/~gert/RIPE/ipv6-filters.html>
- Configure 7200-1 to aggregate both /52 prefixes in the prefix 2001:660:f182::/48 that is the only prefix that is accepted by its BGP peer.
- Apply a policy to prefer the path between 12000-1 and 12000-2.
 - For this purpose, configure on 12000-2 the local-preference 200 on prefixes received from 12000-1. Configure on 12000-3 the local-preference of 150 on prefixes received from 12000-1.
 - Do the symmetric operation on 12000-1.
- Check the BGP details to make sure the policy is properly configured.
 - Using the traceroute command, make sure that the path between 12000-2 and 12000-1 is always preferred.

7 IPv6 Transitioning

7.1 Transitioning with tunnels

7.1.1 Lab Topology



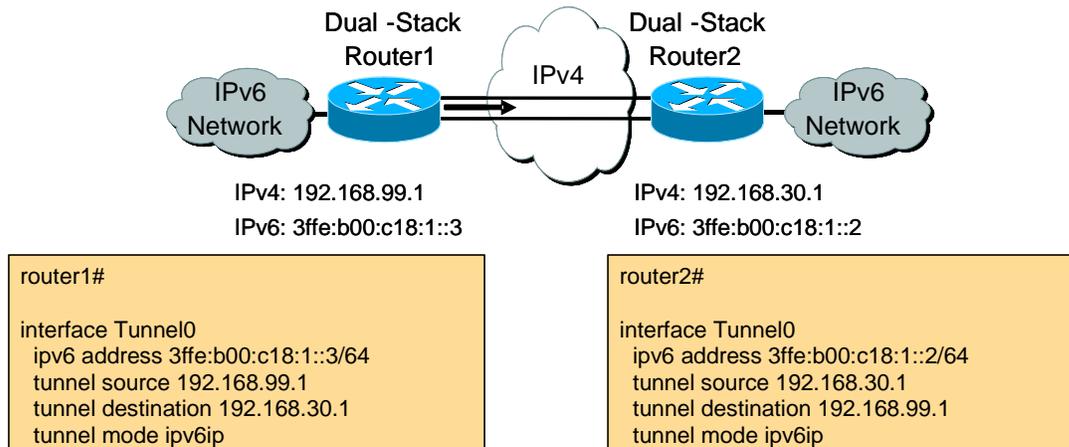
7.1.2 Objective of the Exercise

This exercise will show the trainees how to enable various tunnel mechanisms on Cisco routers. The tunnels should be executed between the Cisco7200-1 and Cisco7200-2 routers, due to their support for tunnel features.

7.1.3 Lab Exercise

7.1.3.1 Manual tunnels according RFC2893

1. General tunnel configuration overview:

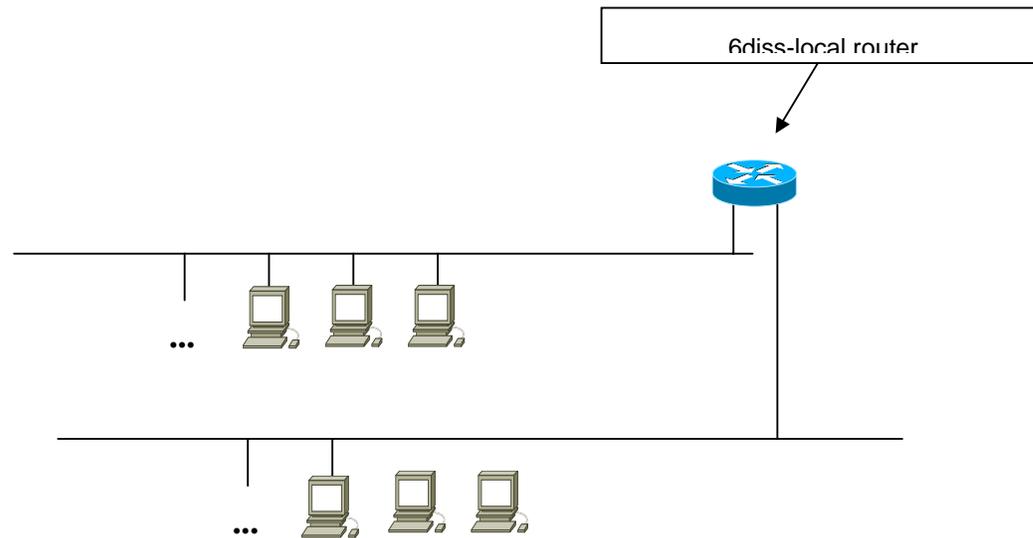


- Manually configured tunnels require:
 - Dual stack end points
 - Both IPv4 and IPv6 addresses configured at each end
2. Configure IPv4:
 - Configure IPv4 addresses and routing on the lab, between Cisco7200-1 and Cisco7200-2. This IPv4 configuration will simulate the IPv4 Internet or an IPv4-only network infrastructure.
 3. Verify the IPv4 connectivity:
 - Check, using the 'ping' command, that Cisco7200-1 can ping Cisco7200-2 and vice-versa
 4. Create the tunnel:
 - Add the tunnel configuration according to the diagram above. The IPv4 tunnel endpoints need to be modified in the configuration to reflect the actual configured addresses in the lab.
 5. Verify the IPv6 tunnel:
 - Execute some 'ping ipv6 x.x.x.x' commands to check if everything works correctly.

8 Multicast (using the Paris 6DISS Lab)

8.1 Part I - Multicast on the link

8.1.1 Lab Exercise



1. Set up the Multicast configuration:

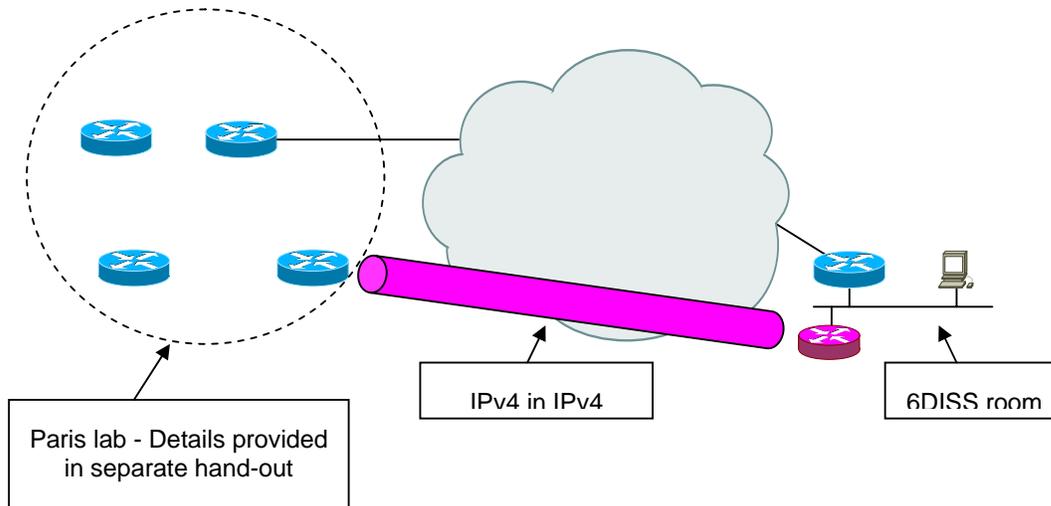
- Make sure IPv6 addressing unicast addressing is configured
- Choose one of the following applications and install it on your PC. Don't hesitate to install any other multicast application you may know.
 - vlc
 - vic/rat/sdr
 - Windows Media Player
 - Realplayer
- Observe messages exchanged on the link between routers and hosts. Observe what happens when you join a group, when you leave a group.
 - Capture MLD messages exchanged on the link using Ethereal.
 - See the states that MLD created on the router.
- Configure MLD filters, to make sure only certain prefixes are allowed.
- In case MLDv2 is used, configure filters based on groups and source addresses. As there is only a single router on the link, make sure you use the trainer's PC to connect to the router, so that all participants do not apply their configuration statements at the same time.

8.2 Part II - Configuring PIM / MBGP

8.2.1 Lab Exercise

The trainees will now be grouped in teams of 2 or 3 people. Each team will be in charge of managing one of the lab routers. Refer to the lab description hand-out to spread the routers between the participants.

1. Set up the PIM /MBGP configuration:



- Make sure IPv6 unicast is running and already configured in the lab. Early sections of this deliverable detail the procedures to do so.
- Configure an IPv6 over IPv4 tunnel between 7200-2 and the 6diss-local router
- Enable PIM-SMv2 in the Paris lab and on the 6diss-local router
- Configure a Rendezvous Point (RP) on GSR-3. Configure the RP statically on all the other routers.
- Add BGP IPv6 multicast address family support for all the peerings of the lab.
- Configure static multicast routing between the lab and the 6DISS room.
- Add some groups statically in the routers, and check that the appropriate trees are created. Send traffic from the 6DISS room and make sure it is forwarded on the trees.
- Change the static routing between the 6DISS room and the lab and use an MBGP peering instead. Use any private AS for 6DISS room.
- Configure an embedded-RP in the 6diss-local router and GSR-2. Make sure that proper RPs are configured based on the chosen multicast addresses.
- Check again that the trees are in place.

8.3 Part III - Monitoring and troubleshooting

8.3.1 Lab Exercise

1. Install Dbeacon:
 - Build a dbeacon matrix with all hosts in the room.
2. Install SSMping and ASM ping on all hosts in the room:
 - Check this tool is running and describe its usefulness.
3. Find the proper commands to check that:
 - PIM states are created on the routers
 - MLD states are created

- IPv6 multicast BGP prefixes are received and sent
- multicast RPF
- multicast packets are properly forwarded on the interfaces

8.4 Receiving video with IPv6 Multicast

8.4.1 Objective of the Exercise

We will now try to receive IPv6 multicast using *vlc*. For this we will use standard multicast (ASM not SSM) so a Rendezvous Point is needed. We did not configure one on the router, but the RP address is encoded in the group address, so that routers know immediately what the RP address is when they see the group address we are using e.g. `ff78:140:2001:630:d0:f000:feed:1` from which a router can derive the RP address `2001:630:d0:f000::1`.

8.4.2 Lab Exercise

1. Type `show ipv6 pim group-map` on the router to see that an RP is configured when *vlc* joins
2. Type `show ipv6 interfaces` to see that a tunnel interface has been created (the tunnel interface is used for PIM register messages).
3. For details of how to run *vlc*, see <http://www.ecstv.ecs.soton.ac.uk/ipv6lab.php>

9 Other Exercises

The following experiments /exercises have been designed and developed in conjunction with UKERNA.

- dhcpv6
- basic static v6 connectivity
- httpd
- access control
- autoconfiguration

These exercises can be run during onsite IPv6 trainings where the trainer has access to the equipment described in the lab topology. Some of the devices and equipment may be provided onsite and belong to the participants.

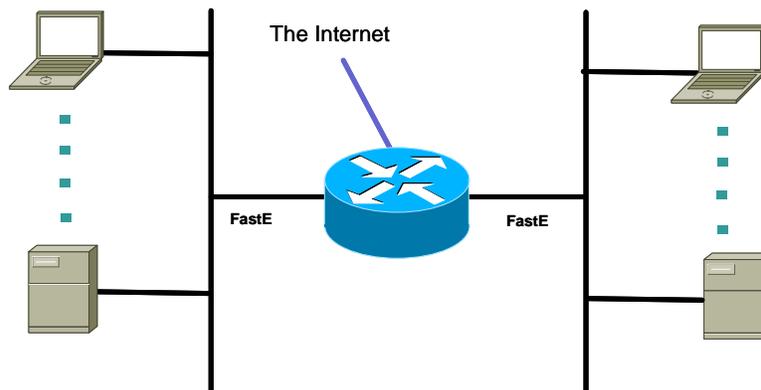
9.1 Lab Topology

Router: Cisco 2821 router with 2 Gigabit ports

Clients:

- 2x Windows XP PCs, with Service Pack 2
- 2x RedHat Enterprise Workstations
- 4x Linux PCs
- ... split into two subnets

Full IPv4 connectivity



9.2 DHCPv6 experiments

9.2.1 Objective of the Exercise

The goal of this lab is to teach the trainees to enable IPv6 DHCP on their clients and to configure a DHCPv6 server and relay-agent.

9.2.2 Lab Exercise

1. Install and configure a DHCPv6 server (eg. Dnsmasq):

- Either a DHCPv6 server, Client and Relay Agent (RA) can be downloaded from:
 - <http://www.tahi.org/dhcpv6/> or
 - [http://www.pl.ipv6tf.org/index.php?id=19&L=1&tx_ttnews\[tt_news\]=22&tx_ttnews\[backPid\]=8&cHash=9939935511](http://www.pl.ipv6tf.org/index.php?id=19&L=1&tx_ttnews[tt_news]=22&tx_ttnews[backPid]=8&cHash=9939935511)
- 2. Configure clients according to the details from the DHCPv6 documentation
- 3. Scenarios:
 - a. No DHCPv6, with RA but without network prefix
In this case, hosts will have:
 - Link-local address
 - Default gateway
 - b. DHCPv6 client enable, with RA but without network prefix
In this case, hosts will have:
 - Link-local address
 - Default gateway
 - Global address (from DHCPv6)
 - c. No DHCPv6, with RA and network prefix (typical stateless autoconfiguration)
In this case, hosts will have:
 - Link-local address
 - Default gateway
 - Global address (from RA message)
 - d. DHCPv6 enabled, with RA and network prefix
In this case, hosts will have:
 - Link-local address
 - Default gateway
 - Global addresses (from DHCPv6 and RA message)
 - e. DHCPv6 enabled, without RA
In this case, hosts will have:
 - Link-local address
 - Global addresses (from DHCPv6 and RA message)

Note that in this scenario, a default gateway must be manually configured in order to have global connectivity.

9.3 Basic Static IPv6 connectivity experiments

9.3.1 Objective of the Exercise

Establish basic IPv6 connectivity from the clients and configure the router to allow clients to connect.

9.3.2 Lab Exercise

1. Observe network configuration, interface statuses, etc.
2. Install/enable IPv6 on clients

- Get the routing infrastructure ready
 - start advertising prefixes for SLAAC (stateless autoconfiguration), or enabling stateful-DHC service
 - Observe changes
 - Observe bits on the wire as a result of enabling
 - SLAAC, DAD, ND, etc.
 - Observe ping tests on a local link
3. Install/enable IPv6 on router
 - On the interface facing the upstream (ge0/0):
 - Enable IPv6 protocol
 - Configure address
 - On the interface facing the subnets (ge0/1.x):
 - Static routing (add a default route toward the head-router)
 - Observe ping tests and trace(path|route) around the lab, to other groups, and beyond
 - type netstat -nr on clients, etc. to examine routes on clients
 4. Global configuration settings needed for Cisco routers
 - Enable IPv6 datagram forwarding
 - Enable IPv6 CEF (optional)
 - Configure an IPv6-transport name server (optional)
 5. IPv6 is now native to the desktop
 - Leave Ethereal running to see the traffic
 - Try: DNS lookups, traceroute6
 - Browse to: ipv6lab.ecs.soton.ac.uk, www.uk6x.com, www.kame.net
 - SSH into neighbouring machines, observe source addresses

9.4 HTTPD experiments

9.4.1 Objective of the Exercise

Enable IPv6 HTTP server services on an Apache server

9.4.2 Lab Exercise

1. Convert IPv4-only Apache (that was built with unconfigured IPv6 support) to speak IPv6
 - Nearly all Unix distributions and pre-built downloadable Apache packages come with IPv6 code compiled in, just not configured
2. Experiment with the “listen” directive
 - This determines the nature of the socket connection
3. Use simple Server Side Includes (SSI) to distinguish the inbound connecting protocol
4. Configure Apache to allow SSIs on .shtml files
5. Create a .shtml file showing server and client addresses
6. Observe the behaviour accessing this using web browsers on the other client machines

9.5 Access Control experiments

9.5.1 Objective of the Exercise

Check and make sure that the router allows IPv6 security data-filters and IPv6 terminal access

9.5.2 Lab Exercise

1. Protect the router configurations (eg. ACLs on line vty):
 - Define an access-class in the same way as for IPv4 (use symbolic names rather than class indexes of a particular range)
 - Bind that access-class to a line definition
 - Test
2. Configure router traffic filters, as with IPv4 access lists, eg. use ACLs to achieve three filters
 - block SSH connections coming in from some of the other teams, but not all
 - block HTTP access to anyone outside of the workshop
 - block all outbound SMTP traffic from your first subnet, but not your second
3. Perform per-node filtering
 - Deny 23/tcp at the edge (i.e. workgroup router)
 - Pick other protocols individually on nodes, e.g. http on client 1, ssh on 2, etc.

9.6 Stateless Autoconfiguration experiments

9.6.1 Objective of the Exercise

Make sure that SLAAC works on the stack of the participants

9.6.2 Lab Exercise

1. On hosts, see the link-local address
 - FE80::EUI-64 address
2. Configure Router Advertisements on Routers
3. With a sniffer, see the RA messages
4. On hosts, see the configuration of global addresses
 - Prefix::EUI-64 address
5. On hosts, see the default gateway
 - It should be the router's link-local address

10 Other Technical Training Material available from 6DISS

10.1 Cisco IPv6 technical e-learning material

Cisco has developed an e-learning course, which complements the 6DISS e-learning package with a deeper level of training on IPv6, but oriented towards Cisco customers and Cisco devices. An arrangement has been made whereby this course can be accessed free of charge, for non-commercial purposes, using a specific username and password combination.

11 Conclusion

Dissemination through workshops is one of the major activities in the project, but dedicated “hands-on” courses are also given for engineers that will have to work with IPv6 equipment. These courses will be given either in Brussels by Cisco staff, in Paris, by RENATER staff, or on site.

This deliverable has explained how attendees will be trained how to configure and operate IPv6 clients and routers. Support will be made available from the appropriate engineers for answering specific technical questions.

The course comprises both slides and hands-on sessions. It is intended for engineers and network managers (especially from ISPs) who will work with equipment on a daily basis, and who want a deeper technical training on IPv6 configuration and management. The main objectives of this “complementary, non-workshop training” are:

- To develop an IPv6 training course for engineers (e.g. deployment engineers, maintenance engineers, NOC personnel)
- To give IPv6 training to engineers (e.g. deployment engineers, maintenance engineers, NOC personnel) in a testbed laboratory

The training course will generally last 1 week and will cover the same items as in the workshops, but with more focus on hands-on practical examples. Equipment from Cisco, Alcatel and Juniper is available. Typically, the course is suitable for up to 20 people.

This document has described the topics covered, through a detailed description of the exercises that will be performed to demonstrate the procedures necessary to establish an IPv6 network. ie:

Basic IPv6 commands

Enabling IPv6 on client terminals

IPv6 routing protocols (OSPF, IS-IS, Aggregation, BGP, Prefix filters, Multicast and Tunnelling)

Enabling IPv6 addressing and routing in the Paris and Brussels labs

Transitioning with tunnels

Multicast using the Paris 6DISS Lab

Other exercises (DHCPv6, Basic IPv6, HTTPD, Traffic Filters, Stateless configuration for clients)