

Travaux Pratiques R&T 1^{ère} année
Durée : 3 heures
SAE21 - IPv6



Noms :
Groupe :
Date :

OBJECTIFS - TOPOLOGIE

⇒ Découverte du fonctionnement du protocole IPv6 (adressage, autoconfig, NDP)

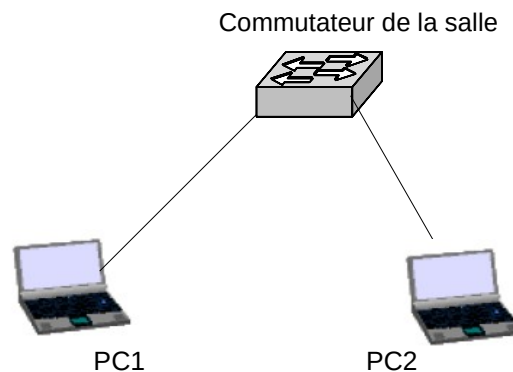
Vous disposez pour le TP de deux PCs en double boot **Windows** et **Linux Debian**, du switch de la salle et d'un routeur Cisco.

MANIPULATIONS

Pour l'ensemble des questions suivantes vous décrirez votre façon de procéder dans votre compte rendu de TP. Vous lancerez le logiciel **Wireshark** afin d'observer le trafic du réseau tout au long de ce TP.

1. Etude de la topologie initiale

Vous considérez ici vos 2 PC dans la topologie actuelle de la salle.



1. Mettez 1 poste sous Windows et l'autre sous linux, observez la configuration réseau IPv6.

Sur le poste windows :

```
enp0s31f6: flags=4163<UP,BROADCAST,RUNNING,MULTICAST> mtu 1500
    inet 172.25.0.54 netmask 255.255.0.0 broadcast 172.25.255.255
    inet6 fe80::d28e:79ff:fe10:d8e4 prefixlen 64 scopeid 0x20<link>
```

Sur le poste windows :

```

Carte Ethernet Ethernet 7 :

  Suffixe DNS propre à la connexion. . . : uca.local
  Adresse IPv6 de liaison locale. . . . : fe80::e282:92f0:a4f4:ba55%36
  Adresse IPv4. . . . . : 172.25.0.53
  Masque de sous-réseau. . . . . : 255.255.0.0
  Passerelle par défaut. . . . . : 172.25.255.254

Carte réseau sans fil Connexion au réseau local* 11 :

  Statut du média. . . . . : Média déconnecté
  Suffixe DNS propre à la connexion. . . :

```

2. Relever les 2 adresses obtenues, de quel type d'adresses s'agit-il ? Comment sont-elles obtenues ? Quelles sont leurs limites en terme d'utilisation ?

Sur le poste linux on relève l'adresse suivante : fe80::d28e:79ff:fe10:d8e4

Sur le poste Windows on relève l'adresse :

Ce sont des adresse unicast lien local

3. Faites un test de connectivité entre les 2 postes. Comment procédez vous ?

```

PS C:\WINDOWS\system32> ping fe80::d28e:79ff:fe10:d8e4

Envoi d'une requête 'Ping' fe80::d28e:79ff:fe10:d8e4 avec 32 octets de données :
Réponse de fe80::d28e:79ff:fe10:d8e4 : temps=1 ms
Réponse de fe80::d28e:79ff:fe10:d8e4 : temps<1ms

Statistiques Ping pour fe80::d28e:79ff:fe10:d8e4:
    Paquets : envoyés = 2, reçus = 2, perdus = 0 (perte 0%),
    Durée approximative des boucles en millisecondes :
        Minimum = 0ms, Maximum = 1ms, Moyenne = 0ms

```

4. Capturez et expliquez les échanges avec Wireshark

1 0.000000000	fe80::e282:92f0:a4f...	fe80::d28e:79ff:fe1...	ICMPv6	94 Echo (ping) request id=0x0001, seq=4, hop limit=128 (reply in 2)
2 0.000045086	fe80::d28e:79ff:fe1...	fe80::e282:92f0:a4f...	ICMPv6	94 Echo (ping) reply id=0x0001, seq=4, hop limit=64 (request in 1)
34 1.000469020	fe80::e282:92f0:a4f...	fe80::d28e:79ff:fe1...	ICMPv6	94 Echo (ping) request id=0x0001, seq=5, hop limit=128 (reply in 35)
35 1.000513395	fe80::d28e:79ff:fe1...	fe80::e282:92f0:a4f...	ICMPv6	94 Echo (ping) reply id=0x0001, seq=5, hop limit=64 (request in 34)
36 2.018697662	fe80::e282:92f0:a4f...	fe80::d28e:79ff:fe1...	ICMPv6	94 Echo (ping) request id=0x0001, seq=6, hop limit=128 (reply in 37)
37 2.018744231	fe80::d28e:79ff:fe1...	fe80::e282:92f0:a4f...	ICMPv6	94 Echo (ping) reply id=0x0001, seq=6, hop limit=64 (request in 36)

On observe que les échanges wireshark sont similaire au ping de ipv4 mais ici la source et la destination sont sous ipv6

5. Mettez en place une configuration IPv6 différente (adresses différentes mais de même type) de manière manuelle en ligne de commande.

Pour windows : fe80::2

Pour linux : fe80::3

On change la config ipv6 du linux :

```

root@iutclrtclb14:~# ip a del fe80::d28e:79ff:fe10:d8e4 dev enp0s31f6
RTNETLINK answers: Cannot assign requested address
root@iutclrtclb14:~# ipconfig
bash: ipconfig : commande introuvable
root@iutclrtclb14:~# ip a add fe80::3/64 dev enp0s31f6
Et sur le windows :

```

[SCREEN CONFIG IP WINDOWS]

On ping les deux et cela fonctionne :

```

RTNETLINK answers: File exists
root@iutclrtclb14:~# ping fe80::2
PING fe80::2(fe80::2) 56 data bytes
64 bytes from fe80::2:enp0s31f6: icmp_seq=1 ttl=128 time=0.814 ms
64 bytes from fe80::2:enp0s31f6: icmp_seq=2 ttl=128 time=0.851 ms
64 bytes from fe80::2:enp0s31f6: icmp_seq=3 ttl=128 time=0.473 ms
64 bytes from fe80::2:enp0s31f6: icmp_seq=4 ttl=128 time=1.14 ms
64 bytes from fe80::2:enp0s31f6: icmp_seq=5 ttl=128 time=0.929 ms
64 bytes from fe80::2:enp0s31f6: icmp_seq=6 ttl=128 time=1.04 ms
64 bytes from fe80::2:enp0s31f6: icmp_seq=7 ttl=128 time=0.694 ms
64 bytes from fe80::2:enp0s31f6: icmp_seq=8 ttl=128 time=0.876 ms
64 bytes from fe80::2:enp0s31f6: icmp_seq=9 ttl=128 time=1.10 ms
64 bytes from fe80::2:enp0s31f6: icmp_seq=10 ttl=128 time=0.966 ms

```

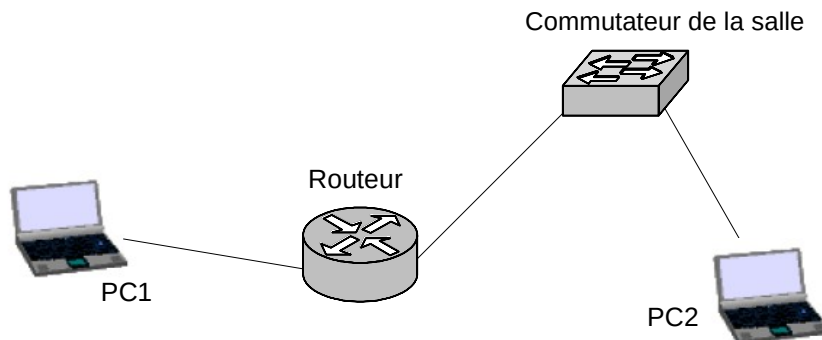
6. Faites un nouveau test de connectivité entre les 2 postes. Capturez et expliquez les échanges avec Wireshark

On prend un ping en même temps on fait une capture wireshark :

2 0.365514744	fe80::3	fe80::2	ICMPv6	118 Echo (ping) request id=0x709e, seq=11, hop limit=64 (reply in 3)
3 0.366592801	fe80::2	fe80::3	ICMPv6	118 Echo (ping) reply id=0x709e, seq=11, hop limit=128 (request in 2)
4 1.366975396	fe80::3	fe80::2	ICMPv6	118 Echo (ping) request id=0x709e, seq=12, hop limit=64 (reply in 5)
5 1.367924027	fe80::2	fe80::3	ICMPv6	118 Echo (ping) reply id=0x709e, seq=12, hop limit=128 (request in 4)
6 2.368202312	fe80::3	fe80::2	ICMPv6	118 Echo (ping) request id=0x709e, seq=13, hop limit=64 (reply in 7)
7 2.369099136	fe80::2	fe80::3	ICMPv6	118 Echo (ping) reply id=0x709e, seq=13, hop limit=128 (request in 6)
8 3.369366377	fe80::3	fe80::2	ICMPv6	118 Echo (ping) request id=0x709e, seq=14, hop limit=64 (reply in 9)
9 3.370324807	fe80::2	fe80::3	ICMPv6	118 Echo (ping) reply id=0x709e, seq=14, hop limit=128 (request in 8)

2. Modification de la topologie

1. Faites les branchements nécessaires pour passer dans la topologie suivante :

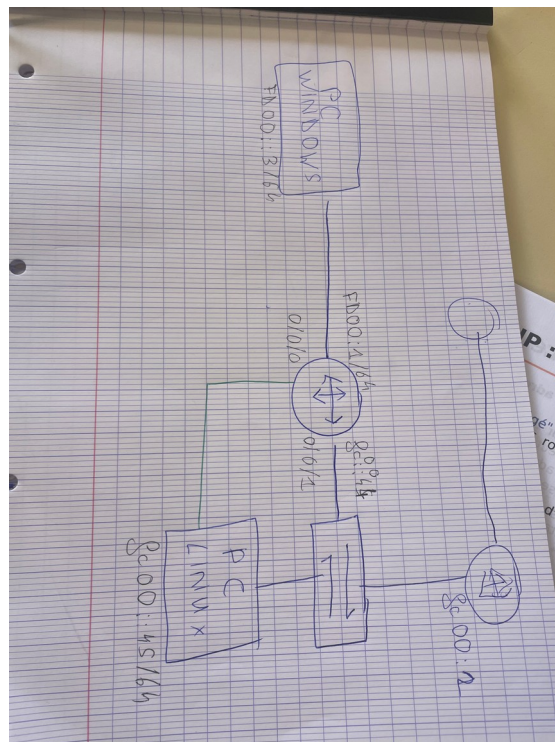


2. Vous allez devoir utiliser maintenant des adresses routables.

Quels sont les types d'adresses IPv6 possibles ? et celui qui vous paraît le mieux adapté à votre cas ? Justifiez votre choix.

Pour notre cas on choisit pour le pc windows l'adresse fd00 :: et pour le linux l'adresse fc00::45/64

Définissez votre plan d'adressage en prévoyant une future mise en commun.



3. Configurez le routeur pour qu'il ait une configuration d'adressage IPv6.

On active l'ipv6 :

```
Router(config)#ipv6 unicast-routing
Router(config)#
```

On configure l'interface du cotés de notre réseau :

```
Router(config)#int gi0/0/0
Router(config-if)#ipv6 addre
Router(config-if)#ipv6 address fd00::1/7
Router(config-if)#
```

et du réseau commun :

```
Router#show ipv6 interface brief
GigabitEthernet0/0/0 [up/up]
FE80::2681:3BFF:FE59:C50
FD00::1
GigabitEthernet0/0/1 [up/up]
FE80::2681:3BFF:FE59:C51
FC00::44
```

On oublie d'activer les interfaces

Enfin, le pc peut bien communiquer avec leurs passerelles :

```
root@iutclrtc1b14:~# ping fc00::44
PING fc00::44(fc00::44) 56 data bytes
64 bytes from fc00::44: icmp_seq=1 ttl=64 time=0.831 ms
64 bytes from fc00::44: icmp_seq=2 ttl=64 time=0.867 ms
64 bytes from fc00::44: icmp_seq=3 ttl=64 time=0.896 ms
^^
```

4. Faites la configuration nécessaire pour que les 2 PC puissent communiquer.

Sur chaque pc on configure une route par default :

```
root@iutclrtc1b14:~# ip -6 r
fc00::/64 dev enp0s31f6 proto kernel metric 256 pref medium
default via fc00::44 dev enp0s31f6 metric 1024 pref medium
```

Avec le pc windows on fait les test de connexion avec la passerelle du réseau reliée au commutateur de la salle et du pc linux.

```

PS C:\WINDOWS\system32> ping fd00::1

Envoi d'une requête 'Ping' fd00::1 avec 32 octets de données :
Réponse de fd00::1 : temps=3 ms
Réponse de fd00::1 : temps=1 ms

Statistiques Ping pour fd00::1:
    Paquets : envoyés = 2, reçus = 2, perdus = 0 (perte 0%),
Durée approximative des boucles en millisecondes :
    Minimum = 1ms, Maximum = 3ms, Moyenne = 2ms
Ctrl+C
PS C:\WINDOWS\system32> ping fc00::44

Envoi d'une requête 'Ping' fc00::44 avec 32 octets de données :
Réponse de fc00::44 : temps<1ms
Réponse de fc00::44 : temps<1ms

Statistiques Ping pour fc00::44:
    Paquets : envoyés = 2, reçus = 2, perdus = 0 (perte 0%),
Durée approximative des boucles en millisecondes :
    Minimum = 0ms, Maximum = 0ms, Moyenne = 0ms
Ctrl+C
PS C:\WINDOWS\system32> ping fc00::45

Envoi d'une requête 'Ping' fc00::45 avec 32 octets de données :
Réponse de fc00::45 : temps<1ms
Réponse de fc00::45 : temps=1 ms
Réponse de fc00::45 : temps=1 ms

Statistiques Ping pour fc00::45:
    Paquets : envoyés = 3, reçus = 3, perdus = 0 (perte 0%),
Durée approximative des boucles en millisecondes :
    Minimum = 0ms, Maximum = 1ms, Moyenne = 0ms

```

5. Quel est le rôle du protocole NDP ? Quels sont les différents types de messages utilisés par ce protocole ? Relevez certains de ces messages sur votre capture Wireshark.

Le protocole ndp est l'équivalent de ARP mais pour l'ipv6. Il permet la découverte des autres hôtes sur le même nœud.

Dans notre cas des pings ont déjà été effectués il n'est donc pas possible d'observer des trames ndp, dans la mesure où il n'y a pas besoin de découvrir des hôtes.

6. Quelles sont les commandes Cisco pour avoir les informations IPv6, de configuration des interfaces, table de routage, postes voisins ? Montrez les résultats obtenus.

Pour les interfaces :

```

Router#show ipv6 interface brief
GigabitEthernet0/0/0 [up/up]
    FE80::2681:3BFF:FE59:C50
    FD00::1
GigabitEthernet0/0/1 [up/up]
    FE80::2681:3BFF:FE59:C51
    FC00::44
GigabitEthernet0/0/2 [down/down]
    unassigned
GigabitEthernet0/0/3 [down/down]
    unassigned

```

Pour les voisins :

```

Router#show ipv6 neighbors
IPv6 Address                               Age Link-layer Addr State Interface
FD00::45C9:3F3A:28A5:FCC8                  0 d08e,7915,d1fb REACH Gi0/0/0
FE80::E282:92F0:A4F4:B455                  0 d08e,7915,d1fb DELAY Gi0/0/0
FC00::45                                    0 d08e,7910,d8e4 REACH Gi0/0/1
FC00::331:5918:A7C9:2006                   23 d08e,7915,d353 STALE Gi0/0/1
FC00::7DE:3DD1:60C4:290F                   17 d08e,7915,d353 STALE Gi0/0/1
FC00::1554:E27B:8011:2AE9                  26 d08e,7915,d215 STALE Gi0/0/1
FC00::35D4:245:ED99:4382                   25 d08e,7915,d1d8 STALE Gi0/0/1
FC00::45D0:2CCA:7C86:53F8                   22 d08e,7915,d239 STALE Gi0/0/1
FC00::4CA5:D427:9C45:B150                   23 d08e,7915,d25b STALE Gi0/0/1
FC00::5413:61C3:4215:D14F                   26 b496,91cb,3d50 STALE Gi0/0/1
FC00::6511:8BE4:9067:B6B0                   27 d08e,7915,d4a1 STALE Gi0/0/1
FC00::651F:7141:5A21:3DF                    26 d08e,7915,d215 STALE Gi0/0/1
FC00::6663:229:9A9C:67B7                   16 d08e,7915,d4a1 STALE Gi0/0/1
FC00::6B71:33AC:A1C5:5CAC                   14 d08e,7915,d4a1 STALE Gi0/0/1
FC00::7894:9EF6:E5BD:D7EE                   16 d08e,7915,d353 STALE Gi0/0/1
FC00::7993:5B5C:8B97:45E8                   14 b496,91cb,3d50 STALE Gi0/0/1
FC00::CAC0:C16B:E03B:894C                    1 d08e,7915,d353 STALE Gi0/0/1
FC00::E159:8D6D:925E:8A7F                   27 d08e,7915,d272 STALE Gi0/0/1
FC00::F4BE:3619:9981:F009                    1 d08e,7915,d353 STALE Gi0/0/1
FE80::5                                    1 d08e,7915,d1d8 STALE Gi0/0/1
FE80::12                                   27 d08e,7915,d272 STALE Gi0/0/1
FE80::67                                   14 d08e,7915,d29c STALE Gi0/0/1
FE80::73                                   14 d08e,7915,d351 STALE Gi0/0/1
FE80::2681:3BFF:FE58:B5B1                    0 2481,3b58,b5b1 STALE Gi0/0/1
FE80::2681:3BFF:FE59:1940                   26 2481,3b59,1940 STALE Gi0/0/1
FE80::4665:36D2:8A8E:525F                   14 b496,91cb,3d50 STALE Gi0/0/1
FE80::58EE:F5C3:782B:61D3                   27 d08e,7915,eed1 STALE Gi0/0/1
FE80::7C8C:6344:1D4C:44B9                   27 d08e,7915,d505 STALE Gi0/0/1
FE80::A10B:6D44:CF6A:A9A                    27 d08e,7915,d2a4 STALE Gi0/0/1
FE80::B5E7:DCBF:DF9F:A042                   27 d08e,7915,d244 STALE Gi0/0/1
FE80::B61D:35F8:1C8E:B85D                   27 d08e,7915,d34c STALE Gi0/0/1
FE80::BF02:E85D:E972:6B1D                   27 d08e,7915,d237 STALE Gi0/0/1
FE80::D28E:79FF:FE15:D25B                   23 d08e,7915,d25b STALE Gi0/0/1
FE80::D28E:79FF:FE15:D353                    1 d08e,7915,d353 STALE Gi0/0/1
FE80::D28E:79FF:FE15:D4A1                   14 d08e,7915,d4a1 STALE Gi0/0/1
FE80::D965:A2F2:B608:4C8C                   26 d08e,7915,d215 STALE Gi0/0/1
FE80::DDDF:B4A3:EFD9:B44A                   22 d08e,7915,d239 STALE Gi0/0/1
FE80::EEF0:BABE:6C4D:D5FD                   16 d08e,7915,d1d8 STALE Gi0/0/1
FE80::FCA9:EBF7:4707:8885                   27 d08e,7915,d332 STALE Gi0/0/1

```

7. Avec un autre binôme, faites les configurations de routage statique nécessaires pour le succès d'un ping entre vos PC.

On va créer une route pour acheminer les informations :

```

Router(config)#ipv6 route fc00:1020::/64 fc00::2

```

Par la suite sur chaque pc, on fait une route sur chaque pc :

```
PS C:\WINDOWS\system32> netsh interface ipv6
Les commandes suivantes sont disponibles :

Commandes dans ce contexte :
6to4      - Modifications pour le contexte 'netsh interface ipv6 6to4'.
?         - Affiche une liste de commandes.
add       - Ajoute une entrée de configuration à une table.
delete    - Supprime une entrée de configuration d'une table.
dump      - Affiche un script de configuration.
help      - Affiche une liste de commandes.
isatap    - Modifications pour le contexte 'netsh interface ipv6 isatap'.
reset     - Réinitialiser les configurations IP.
set       - Définit les informations de configuration.
show      - Affiche les informations.

Les sous-contextes suivants sont disponibles :
6to4 isatap

Pour consulter l'aide d'une commande, entrez la commande, suivie par un
espace, et ensuite entrez ?.

PS C:\WINDOWS\system32> netsh interface ipv6 add route fc:1020::/64 "Ethernet 7" fc00::2
La commande suivante n'a pas été trouvée : interface ipv6 add route fc:1020::/64 "Ethernet 7" fc00::2.
PS C:\WINDOWS\system32> netsh interface ipv6 add route fc:1020::/64 "Ethernet 7" fc00::2
Ok.

PS C:\WINDOWS\system32> ping fc00:1020::1234:ad69

Envoi d'une requête 'Ping' fc00:1020::1234:ad69 avec 32 octets de données :
Réponse de fc00:1020::1234:ad69 : temps=1 ms
Réponse de fc00:1020::1234:ad69 : temps=2 ms
Réponse de fc00:1020::1234:ad69 : temps=1 ms
Réponse de fc00:1020::1234:ad69 : temps=1 ms

Statistiques Ping pour fc00:1020::1234:ad69:
    Paquets : envoyés = 4, reçus = 4, perdus = 0 (perte 0%),
    Durée approximative des boucles en millisecondes :
        Minimum = 1ms, Maximum = 2ms, Moyenne = 1ms
```

Le ping aboutit bien