



Table des matières

1) Recherche sur différents paquets.....	1
2) PPP monoclient.....	4
Configuration côté client :.....	4
Configuration côté serveur :.....	4
Test de connexion entre le client et le serveur.....	5
1) Connexion entre le client et le serveur.....	5
2) Ping entre le client et le serveur (interface ppp0).....	9
3) Ping entre le client et le serveur (interface eno1).....	10
4) Ping entre le client et la Gateway.....	11

1) Recherche sur différents paquets

En faisant `apt show [paquet]` on peut voir pas mal d'informations sur un paquet choisit. Ici en faisant un `show` du paquet `ppp` on peut voir des informations très utiles comme :

- Sa version
- Ses dépendances (les paquets installés avec ce paquet là)
- Sa description

```
root@213-9:/home/test# apt show ppp
Package: ppp
Version: 2.4.7-2+4.1
Priority: optional
Section: admin
Maintainer: Chris Boot <bootc@debian.org>
Installed-Size: 1 044 kB
Depends: libpam-modules, libpam-runtime, lsb-base, procps, libc6 (>= 2.15),
libpam0g (>= 0.99.7.1), libpcap0.8 (>= 0.9.8), libssl1.1 (>= 1.1.0)
Breaks: network-manager (<< 0.9.8.8-7~), network-manager-pptp (<< 0.9.8.4-
3~), pppdcapiplugin (<< 1:3.25+dfsg1-3.4~)
Homepage: http://ppp.samba.org/
Tag: hardware::modem, implemented-in::c, interface::daemon, network::server,
protocol::ip, protocol::ipv6, role::program, security::authentication,
```

```
use::dialing, use::login, works-with::network-traffic
Download-Size: 353 kB
APT-Manual-Installed: yes
APT-Sources: http://debian.iutbeziers.fr/debian buster/main amd64 Packages
Description: protocole point à point (PPP) - démon
Le protocole point-à-point (PPP) fournit une méthode normalisée de
transmission de datagrammes via un lien série et de négociation des
caractéristiques du lien entre les machines aux deux extrémités.
.
Ce paquet fournit un démon qui est en général utilisé pour gérer un modem
(RTC ou haut débit) et établir une connexion ponctuelle.
```

Maintenant en faisant un `man -k [paquet]` on peut avoir des informations supplémentaires sur un paquet :

Ici nous faisons un `man -k ppp` :

```
root@213-9:/home/test# man -k ppp
plog (1)          - starts up, shuts down or lists the log of PPP connections
poff (1)          - starts up, shuts down or lists the log of PPP connections
pon (1)           - starts up, shuts down or lists the log of PPP connections
pppd (8)          - Point-to-Point Protocol Daemon
pppd-radattr (8)  - RADIUS utility plugin for pppd (8)
pppd-radius (8)   - RADIUS authentication plugin for pppd (8)
pppdump (8)       - convert PPP record file to readable format
pppoe (8)         - user-space PPPoE client.
pppoe-connect (8) - Shell script to manage a PPPoE link
pppoe-discovery (8) - perform PPPoE discovery
pppoe-relay (8)   - user-space PPPoE relay agent.
pppoe-server (8)  - user-space PPPoE server
pppoe-setup (8)   - Shell script to configure Roaring Penguin PPPoE client
pppoe-sniff (8)   - examine network for non-standard PPPoE frames
pppoe-start (8)   - Shell script to bring up a PPPoE link
pppoe-status (8)  - Shell script to report on status of PPPoE link
pppoe-stop (8)    - Shell script to shut down a PPPoE link
pppoe.conf (5)    - Configuration file used by pppoe-start(8), pppoe-stop(...)
pppoeconf (8)     - configures a PPPoE (ADSL) connection
pppstats (8)      - print PPP statistics
```

Après avoir fait cette commande on peut voir que certains paquets nous intéressent comme le `pppd` ou bien `pppoe` et `pppoeconf`.

Nous commençons par l'apt show de **pppd** (Point-To-Point Protocol Deamon) :

Le pppd travail avec le kernel ppp pour établir et maintenir un lien ppp avec d'autres systèmes.

Description: protocole point à point (PPP) - démon
Le protocole point-à-point (PPP) fournit une méthode normalisée de transmission de datagrammes via un lien série et de négociation des caractéristiques du lien entre les machines aux deux extrémités. Ce paquet fournit un démon qui est en général utilisé pour gérer un modem (RTC ou haut débit) et établir une connexion ponctuelle.

Maintenant l'apt show de **pppoe** (ppp over Ethernet) :

Description: Pilote PPP sur Ethernet
PPP sur Ethernet (PPPoE) est un protocole utilisé par de nombreux fournisseurs d'accès internet ADSL. Ce paquet vous permet de vous connecter à ces fournisseurs de service PPPoE.

Pour finir l'apt show de **pppoeconf** (pppoe configuration) :

Description: configure PPPoE/ADSL connections
User-friendly tool for initial configuration of a DSL (PPPoE) connection.

Il permet de faire la configuration de la connexion en DSL du client vers le serveur.

2) PPP monoclient

Pour commencer nous devons installer les paquets utiles pour permettre une connexion en DSL entre le client et le serveur. Pour cela on installe les paquets suivants :

Sur le client	Sur le serveur
ppp	ppp
pppoe	pppoe
pppoeconf	

Configuration côté client :

Une fois les paquets installés sur le client on tape pppoeconf pour configurer manuellement le client. (**à faire dès que le démon du serveur est lancé**)

Dans la configuration on laisse tout comme on nous propose de base à part l'username et le mot de passe que nous changeons (ici c'est hugo et test)

Configuration côté serveur :

On branche le dongle usb sur le pc (serveur) puis on branche le pc client sur le dongle

On définit une adresse ip sur l'interface du dongle usb branché au pc (serveur). Ici on utilise 10.213.11.2

Ensuite on active l'interface avec la commande suivante :

```
root@213-2:/etc/ppp# ip link set up enx3c18a002b9e9
```

Commande pour lancer le démon du serveur :

```
root@232-22:/home/test# pppoe-server -I enx3c18a002b9b1 -L 10.213.11.1 -R 10.213.11.101
```

Options : -I (Interface) : Choix de l'interface

-L (local) : Choix de l'IP donnée au serveur

-R (Remote) : Choix de la première IP donnée au client qui vient se connecter au serveur

Commande pour vérifier que le démon du serveur est bien lancé :

```
root@232-22:/home/test# ps -A |grep pppoe
8169 ?      00:00:00 pppoe-server
```

Configuration des fichiers :

fichier pppoe-server-options :

```
# PPP options for the PPPoE server
require-pap
```

Dans ce fichier on dit que le protocole pour s'authentifier est pap

fichier pap-secrets :

```
#User      #Server      #Password      #IP
hugo        *            test           *
```

Dans ce fichier on ajoute les clients qui vont s'authentifier grâce au protocole PAP (Password Authentication Protocol). On y met les Users, Servers, Password et IP

Test de connexion entre le client et le serveur

1) Connexion entre le client et le serveur

Sur le serveur le démon doit être lancé. Dès que le serveur est lancé on va utiliser la commande suivante pour lancer une connexion vers le serveur :

```
root@213-12:/etc/ppp# pon dsl-provider
Plugin rp-pppoe.so loaded
```

Après avoir lancé le démon du serveur et configuré ses fichiers on va lancer wireshark et faire une capture de trames sur l'interface du dongle usb (**Les captures de trames sont faites pendant la connexion entre le client et le serveur**) :

No.	Time	Source	Destination	Protocol	Length	Info
10	17.444125150	Dell_e0:7e:ce	Broadcast	PPPoED	60	

Active Discovery Initiation (PADI)

Frame 10: 60 bytes on wire (480 bits), 60 bytes captured (480 bits) on interface 0

Ethernet II, Src: Dell_e0:7e:ce (98:90:96:e0:7e:ce), Dst: Broadcast (ff:ff:ff:ff:ff:ff)
PPP-over-Ethernet Discovery

No.	Time	Source	Destination	Protocol	Length	Info
11	17.444253408	Luxshare_02:b9:e9	Dell_e0:7e:ce	PPPoED	57	

Active Discovery Offer (PADO) AC-Name='213-2'

Frame 11: 57 bytes on wire (456 bits), 57 bytes captured (456 bits) on interface 0

Ethernet II, Src: Luxshare_02:b9:e9 (3c:18:a0:02:b9:e9), Dst: Dell_e0:7e:ce (98:90:96:e0:7e:ce)
PPP-over-Ethernet Discovery

No.	Time	Source	Destination	Protocol	Length	Info
12	17.444708240	Dell_e0:7e:ce	Luxshare_02:b9:e9	PPPoED	60	

Active Discovery Request (PADR)

Frame 12: 60 bytes on wire (480 bits), 60 bytes captured (480 bits) on interface 0

Ethernet II, Src: Dell_e0:7e:ce (98:90:96:e0:7e:ce), Dst: Luxshare_02:b9:e9 (3c:18:a0:02:b9:e9)
PPP-over-Ethernet Discovery

No.	Time	Source	Destination	Protocol	Length	Info
13	17.445379033	Luxshare_02:b9:e9	Dell_e0:7e:ce	PPPoED	24	

Active Discovery Session-confirmation (PADS)

Frame 13: 24 bytes on wire (192 bits), 24 bytes captured (192 bits) on interface 0
Ethernet II, Src: Luxshare_02:b9:e9 (3c:18:a0:02:b9:e9), Dst: Dell_e0:7e:ce (98:90:96:e0:7e:ce)
PPP-over-Ethernet Discovery

En vert au dessus on peut voir les différentes étapes lors de la connexion du client au serveur. Le PADI fait que le client demande une **connexion au serveur**. Le PADO est la **réponse du serveur** vers le client avec une adresse IP pour dire qu'il écoute le client. Le PADR est la **demande de connexion et l'obtention d'une IP du client** sur le serveur. Le PADS est la **confirmation du serveur** pour que le client se connecte dessus avec une IP qu'il lui donne. Ce processus ressemble à celui du DHCP.

No.	Time	Source	Destination	Protocol	Length
Info					
25	11.173040258	Dell_e0:81:37	Luxshare_02:b9:b1	PPP PAP	
60		Authenticate-Request (Peer-ID='hugo', Password='test')			

Frame 25: 60 bytes on wire (480 bits), 60 bytes captured (480 bits) on interface 0
Ethernet II, Src: Dell_e0:81:37 (98:90:96:e0:81:37), Dst: Luxshare_02:b9:b1 (3c:18:a0:02:b9:b1)
PPP-over-Ethernet Session
Point-to-Point Protocol
PPP Password Authentication Protocol

Ci-dessus en orange on peut voir que le protocole utilisé pour l'authentification est bien PAP et en rouge on peut voir que l'ID et le mot de passe qui passent en clair sont bien ceux du client.

No.	Time	Source	Destination	Protocol	Length
Info					
28	11.173382277	Luxshare_02:b9:b1	Dell_e0:81:37	PPP PAP	
35		Authenticate-Ack (Message='Login ok')			

Frame 28: 35 bytes on wire (280 bits), 35 bytes captured (280 bits) on interface 0
Ethernet II, Src: Luxshare_02:b9:b1 (3c:18:a0:02:b9:b1), Dst: Dell_e0:81:37 (98:90:96:e0:81:37)
PPP-over-Ethernet Session
Point-to-Point Protocol
PPP Password Authentication Protocol

La trame ci-dessus montre que l'authentification a bien été effectuée avec succès et que le client est donc bien connecté.

Lors de la fin d'une connexion on a un PADT qui apparaît dans les trames. Il correspond à la fin d'une connexion entre le client et le serveur :

No.	Time	Source	Destination	Protocol	Length	Info
6	2.223294230	Dell_e0:7e:ce	Luxshare_02:b9:e9	PPP LCP	60	Termination Request

Frame 6: 60 bytes on wire (480 bits), 60 bytes captured (480 bits) on interface 0
Ethernet II, Src: Dell_e0:7e:ce (98:90:96:e0:7e:ce), Dst: Luxshare_02:b9:e9 (3c:18:a0:02:b9:e9)
PPP-over-Ethernet Session
Point-to-Point Protocol
PPP Link Control Protocol

Sur la trame ci-dessus on peut voir en vert l'adresse du pc (client), en violet l'adresse du pc (serveur) et en rouge l'information (demande de fin de connexion entre les deux pc).

No.	Time	Source	Destination	Protocol	Length	Info
7	2.232263908	Luxshare_02:b9:e9	Dell_e0:7e:ce	PPP LCP	26	Termination Ack

Frame 7: 26 bytes on wire (208 bits), 26 bytes captured (208 bits) on interface 0
Ethernet II, Src: Luxshare_02:b9:e9 (3c:18:a0:02:b9:e9), Dst: Dell_e0:7e:ce (98:90:96:e0:7e:ce)
PPP-over-Ethernet Session
Point-to-Point Protocol
PPP Link Control Protocol

Sur la trame au dessus on peut voir que la demande de fin de connexion est acceptée

No.	Time	Source	Destination	Protocol	Length	Info
8	2.276131500	Dell_e0:7e:ce	Luxshare_02:b9:e9	PPPoED	60	Active Discovery Terminate (PADT)

Frame 8: 60 bytes on wire (480 bits), 60 bytes captured (480 bits) on interface 0
Ethernet II, Src: Dell_e0:7e:ce (98:90:96:e0:7e:ce), Dst: Luxshare_02:b9:e9 (3c:18:a0:02:b9:e9)
PPP-over-Ethernet Discovery

No.	Time	Source	Destination	Protocol	Length	Info
9	2.276418281	Luxshare_02:b9:e9	Dell_e0:7e:ce	PPPoED	37	

Active Discovery Terminate (PADT)

Frame 9: 37 bytes on wire (296 bits), 37 bytes captured (296 bits) on interface 0
Ethernet II, Src: Luxshare_02:b9:e9 (3c:18:a0:02:b9:e9), Dst: Dell_e0:7e:ce (98:90:96:e0:7e:ce)
PPP-over-Ethernet Discovery

Sur les deux dernières trames on peut voir que la fin de la connexion est acceptée par les deux pc

Pour vérifier qu'il n'y a pas eu de problèmes pendant la connexion on peut faire un plog (si rien ne s'affiche c'est bon)

On peut aussi voir plus de détails dans /var/log/syslog

2) Ping entre le client et le serveur (interface ppp0)

Maintenant le client est bien connecté au serveur et on peut le ping :

```
root@213-1:/home/test# ping 10.213.2.10
PING 10.213.2.10 (10.213.2.10) 56(84) bytes of data.
64 bytes from 10.213.2.10: icmp_seq=1 ttl=64 time=0.982 ms
64 bytes from 10.213.2.10: icmp_seq=2 ttl=64 time=0.999 ms
64 bytes from 10.213.2.10: icmp_seq=3 ttl=64 time=1.71 ms
64 bytes from 10.213.2.10: icmp_seq=4 ttl=64 time=1.18 ms
^C
--- 10.213.2.10 ping statistics ---
4 packets transmitted, 4 received, 0% packet loss, time 7ms
rtt min/avg/max/mdev = 0.982/1.219/1.713/0.296 ms
```

No.	Time	Source	Destination	Protocol	Length	Info
1	0.000000000	10.213.2.101	10.213.2.10	ICMP	100	

Echo (ping) request id=0x0528, seq=1/256, ttl=64 (reply in 2)

Frame 1: 100 bytes on wire (800 bits), 100 bytes captured (800 bits) on interface 0

Linux cooked capture

Internet Protocol Version 4, Src: 10.213.2.101, Dst: 10.213.2.10
Internet Control Message Protocol

No.	Time	Source	Destination	Protocol	Length	Info
2	0.000916825	10.213.2.10	10.213.2.101	ICMP	100	

Echo (ping) reply id=0x0528, seq=1/256, ttl=64 (request in 1)

Frame 2: 100 bytes on wire (800 bits), 100 bytes captured (800 bits) on interface 0
Linux cooked capture
Internet Protocol Version 4, Src: 10.213.2.10, Dst: 10.213.2.101
Internet Control Message Protocol

En violet est surligné l'adresse source (client) et en orange l'adresse de destination (serveur). On voit bien que le client envoie un echo Request et que le serveur répond avec un echo Reply

3) Ping entre le client et le serveur (interface eno1)

Maintenant nous allons essayer de faire un ping entre l'interface ppp0 du client (10.213.1.101) et l'interface eno1 du serveur (10.213.2.1)

```
root@213-1:/home/test# ping 10.213.2.1
PING 10.213.2.1 (10.213.2.1) 56(84) bytes of data.
64 bytes from 10.213.2.1: icmp_seq=1 ttl=64 time=1.21 ms
64 bytes from 10.213.2.1: icmp_seq=2 ttl=64 time=1.12 ms
64 bytes from 10.213.2.1: icmp_seq=3 ttl=64 time=0.694 ms
64 bytes from 10.213.2.1: icmp_seq=4 ttl=64 time=1.09 ms
64 bytes from 10.213.2.1: icmp_seq=5 ttl=64 time=0.815 ms
64 bytes from 10.213.2.1: icmp_seq=6 ttl=64 time=0.943 ms
```

Sur Wireshark on peut voir les trames suivantes :

No.	Time	Source	Destination	Protocol	Length	Info
1	0.000000000	10.213.2.101	10.213.2.1	ICMP	100	Echo (ping) request id=0x2bea, seq=1/256, ttl=64 (reply in 2)
Frame 1: 100 bytes on wire (800 bits), 100 bytes captured (800 bits) on interface 0 Linux cooked capture Internet Protocol Version 4, Src: 10.213.2.101, Dst: 10.213.2.1 Internet Control Message Protocol						
2	0.001155629	10.213.2.1	10.213.2.101	ICMP	100	Echo (ping) reply id=0x2bea, seq=1/256, ttl=64 (request in 1)
Frame 2: 100 bytes on wire (800 bits), 100 bytes captured (800 bits) on interface 0 Linux cooked capture Internet Protocol Version 4, Src: 10.213.2.1, Dst: 10.213.2.101 Internet Control Message Protocol						

On a encore une fois un echo Request et un echo Reply qui montrent que les deux interfaces arrivent bien à communiquer.

Ceci est dû à la machine elle-même. Comme les deux interfaces sont situées sur la même machine elles peuvent communiquer entre elles.

4) Ping entre le client et la Gateway

Maintenant on voit que le client n'a pas accès à internet (en faisant un ping ou en ouvrant une page internet) :

Pour que le client puisse avoir accès à internet il faut activer le routage sur le serveur. Il se fait de la façon suivante :

```
root@213-2:/etc/ppp# echo 1 > /proc/sys/net/ipv4/ip_forward
```

À partir de là si on fait un ping vers l'extérieur on ne doit voir que des echo Request et du ARP :

No.	Time	Source	Destination	Protocol	Length	Info
1	0.000000000	10.213.2.101	8.8.8.8	ICMP	98	Echo (ping) request id=0x4904, seq=1/256, ttl=63 (no response found!)

Frame 1: 98 bytes on wire (784 bits), 98 bytes captured (784 bits) on interface 0

Ethernet II, Src: Dell_e0:84:b2 (98:90:96:e0:84:b2), Dst: HewlettP_2d:df:4c (d0:7e:28:2d:df:4c)

Internet Protocol Version 4, Src: 10.213.2.101, Dst: 8.8.8.8

Internet Control Message Protocol

No.	Time	Source	Destination	Protocol	Length	Info
2	0.008808550	HewlettP_2d:df:4c	Broadcast	ARP	60	Who has 10.213.2.101? Tell 10.213.255.254

Frame 2: 60 bytes on wire (480 bits), 60 bytes captured (480 bits) on interface 0

Ethernet II, Src: HewlettP_2d:df:4c (d0:7e:28:2d:df:4c), Dst: Broadcast (ff:ff:ff:ff:ff:ff)

Address Resolution Protocol (request)

On a pas de retour car l'arp ne connaît pas la destination de retour

Il nous reste donc encore à activer le retour des réponses avec la commande suivante :

```
root@213-2:/etc/ppp# echo 1 > /proc/sys/net/ipv4/conf/eno1/proxy_arp
```

Après avoir fait ceci le client a accès à internet (**Modifier le DNS de l'IUT s'il n'est pas comme le suivant : 10.255.255.200**) :

No.	Time	Source	Destination	Protocol	Length	Info
68	28.2344	10.213.2.104	8.8.8.8	ICMP	106	Echo (ping)
request id=0x4579, seq=4/1024, ttl=64 (reply in 69)						
Frame 68: 106 bytes on wire (848 bits), 106 bytes captured (848 bits) on interface 0						
Ethernet II, Src: Dell_e0:7e:ce (98:90:96:e0:7e:ce), Dst: Luxshare_02:b9:e9 (3c:18:a0:02:b9:e9)						
PPP-over-Ethernet Session						
Point-to-Point Protocol						
Internet Protocol Version 4, Src: 10.213.2.104, Dst: 8.8.8.8						
Internet Control Message Protocol						
No.	Time	Source	Destination	Protocol	Length	Info
69	28.2409	8.8.8.8	10.213.2.104	ICMP	106	Echo (ping)
reply id=0x4579, seq=4/1024, ttl=53 (request in 68)						
Frame 69: 106 bytes on wire (848 bits), 106 bytes captured (848 bits) on interface 0						
Ethernet II, Src: Luxshare_02:b9:e9 (3c:18:a0:02:b9:e9), Dst: Dell_e0:7e:ce (98:90:96:e0:7e:ce)						
PPP-over-Ethernet Session						
Point-to-Point Protocol						
Internet Protocol Version 4, Src: 8.8.8.8, Dst: 10.213.2.104						
Internet Control Message Protocol						

On peut voir sur les deux trames au dessus qu'après avoir activé le routage et le retour des paquets le client arrive bien à accéder à internet