TP2 NE324 - Le protocole PPP

Owen Rougé et Adrian Bonnet 05//19

Préparation : les questions effectuées en préparation du TP sont les questions 1, 2.a et 2.e, en plus de la question sur "`Linux cooked-mode capture`" sur `wireshark`.

0. Linux cooked-mode capture

Préparation : Sous Linux, `libpcap` ne permet pas de capturer les en-têtes correctes de la couche liaison pour le véritable protocole (Ethernet), mais fournit à la place des fausses en-têtes. On parle de pseudo-protocole.

Sur `wireshark`, le `Linux cooked-mode capture` est fonctionnel.

1. PPP sur Ethernet

Préparation : La taille maximum de la charge utile (i.e. le champs de données) sur Ethernet est de 1500 octets. Or l'en-tête PPPoE est de 6 octets et l'identificateur de protocole est de 2 octets. Donc le MTU de PPP ne doit pas dépasser 1492 octets.

Les fichiers modifiés par `pppoeconf` en root sont : `/etc/ppp/peers/dsl-provider`, `/etc/ppp/*-secrets` et `/etc/network/interfaces`.

Après la commande `pon dsl-provider`, nou observons bien grâce à ifconfig l'interface ppp0. Celle-ci disparaît après la commande `poff dsl-provider`.

2. Etude de PPP(oE)
    a. **Préparation** : PPP est un protocole de la couche liaison. Il doit donc résoudre des problèmes de transmission de données liés à cette couche. PPPoE utilise Ethernet pour la couche liaison. Il n'y a donc plus nécessité d'utiliser les fanions et les adresses, puisque ces services (synchronicité et destination) sont résolus par Ethernet.

    b. voir annexe 1

Ethernet II est associé au trafic PPPoE. Sur l'interface ppp0, le MTU est 1492 octets, imposé par `Linux cooked-mode`, tandis que sur eth0, le MTU est 1500 octets pour Ethernet II.

    c. voir annexe 2

Les valeurs possibles du champ code de PPPoED est `0x09` (PADI), `0x07` (PADO), `0x19` (PADR) et `0x65` (PADS).

La valeur du champ type de la trame Ethernet est `0x8863`

La valeur du champ session de PPPoE est `0x0000` tant que le lien n'est pas confirmé, puis `0x001f` dans la trame de confirmation du lien.

d. voir annexe 3

Les paramètres négociés sont le MRU, les champs `Protocol Field compression` et `Authentication Protocol`.

e. **Préparation** : Le champ protocol est reçu avec l'octet de poids fort en premier. Le bit de poids faible (LSB) de chaque octet est utilisé pour indiquer l'extension de protocole. Un 0 comme LSB signifie que le champs continue après cet octet, un 1 comme LSB marque la fin du champ protocol. Le champ protocol ne doit pas être compressé tant que l'option de configuration n'est pas négociée.

f. voir annexe 4

Le fonctionnement est le suivant : le serveur envoie une requête d'identification, le client envoie une réponse contenant l'identifiant. Le serveur envoie ensuite un request MD5-challenge, le client répond avec un response MD5-challenge contenant le HASH du mot de passe. Le serveur envoie un dernier message correspondant au succès ou à l'échec d'authentification.

g. voir annexe 5

Dans le fichier `/etc/ppp/peers/dsl-provider`, on précise `refuse-eap` pour refuser l'authentification en EAP.

Dans l'échange LCP, le client envoie un NAK, refusant le protocole EAP et proposant le protocole CHAP. Le serveur refait un `configuration request`, que le client acquitte.

h. voir annexe 6

Une fois refusé le protocole CHAP, le client propose le protocole MS-CHAP-V2. Si MS-CHAP-v2 est refusé, il propose MS-CHAP, puis de la même manière le protocole PAP. Si PAP est refusé, aucune authentification n'est utilisée.

i. Le serveur accepte les protocoles PAP, CHAP (ainsi que MS-CHAP et MS-CHAPv2), et EAP.

j. voir annexe 7

k. Pour forcer l'adresse IP du client, il faut rajouter dans `/etc/ppp/peers/dsl-provider` une ligne `<IP_client> : <IP_serveur>` (sans les chevrons).

On obtient ainsi effectivement l'adresse IP que l'on souhaite dans la communication.

Nous avons ouvert deux liens PPP avec la même adresse IP. Ceci ne pose aucun problème de connexion au serveur.

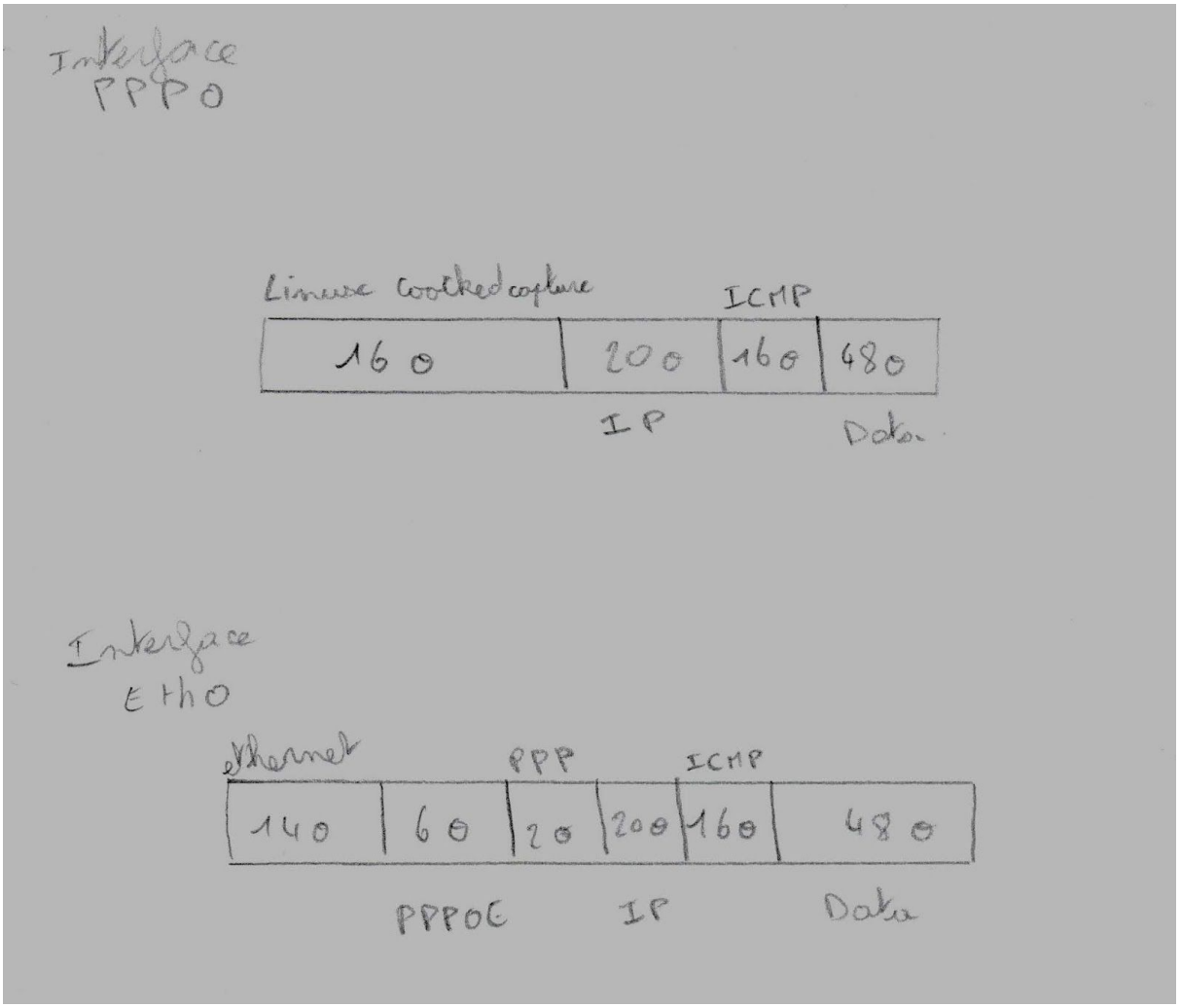
Il se trouve que PPP est un protocole de la couche liaison, donc il n'assure pas la sécurité des adresses IP. Ce n'est donc pas un bug de spécification de PPP. PPP assure la possibilité d'échange entre les deux entités, rien de plus.

l. En mettant dans `<IP_serveur>` l'adresse que l'on souhaite, on peut demander au serveur de changer d'adresse IP. Le serveur accepte de le faire (dans ce cas présent, mais cela dépend de sa configuration).

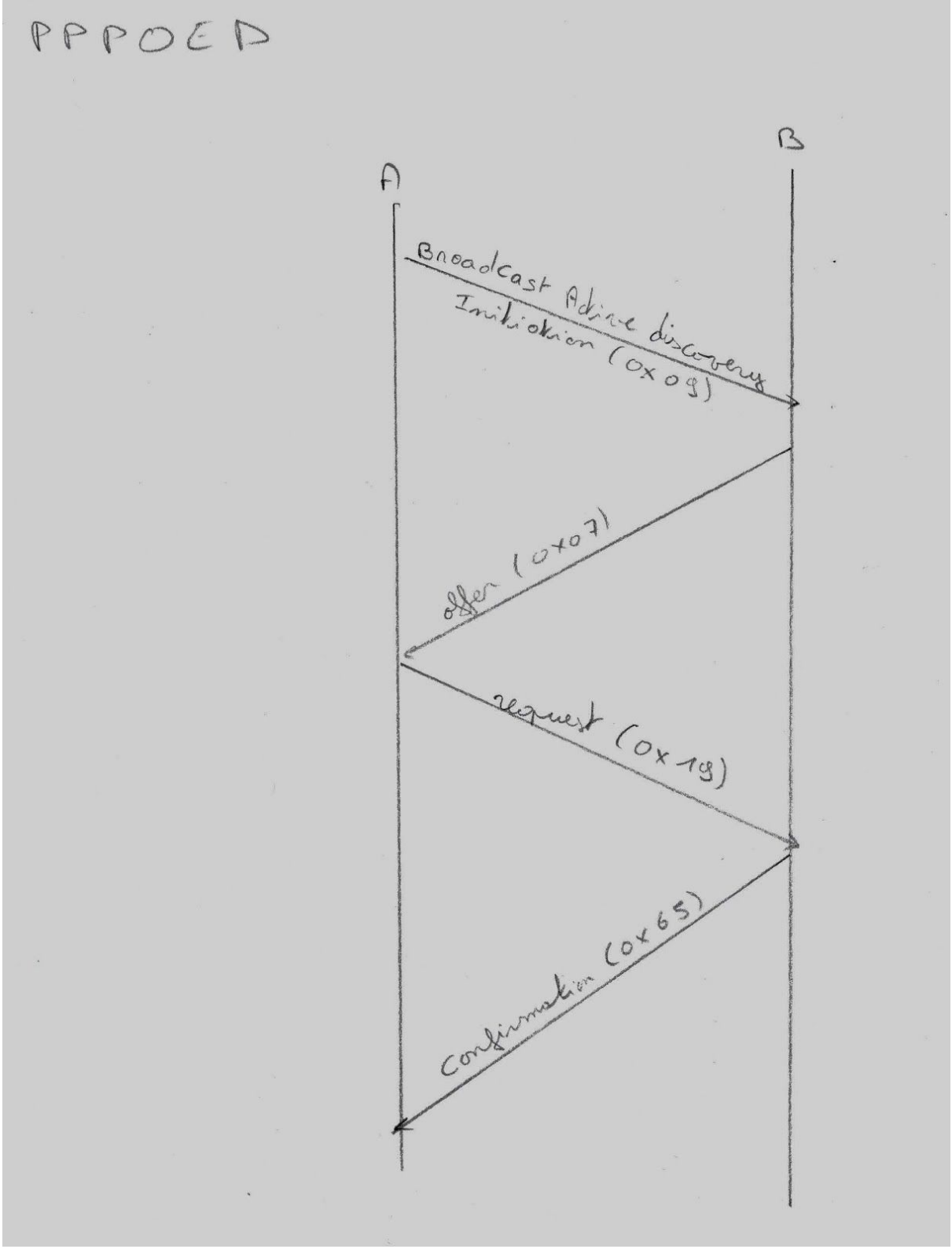
Dans le protocole PPP, les adresses IP n'ont pas réellement d'importance car la communication se fait en point à point.

Annexes :

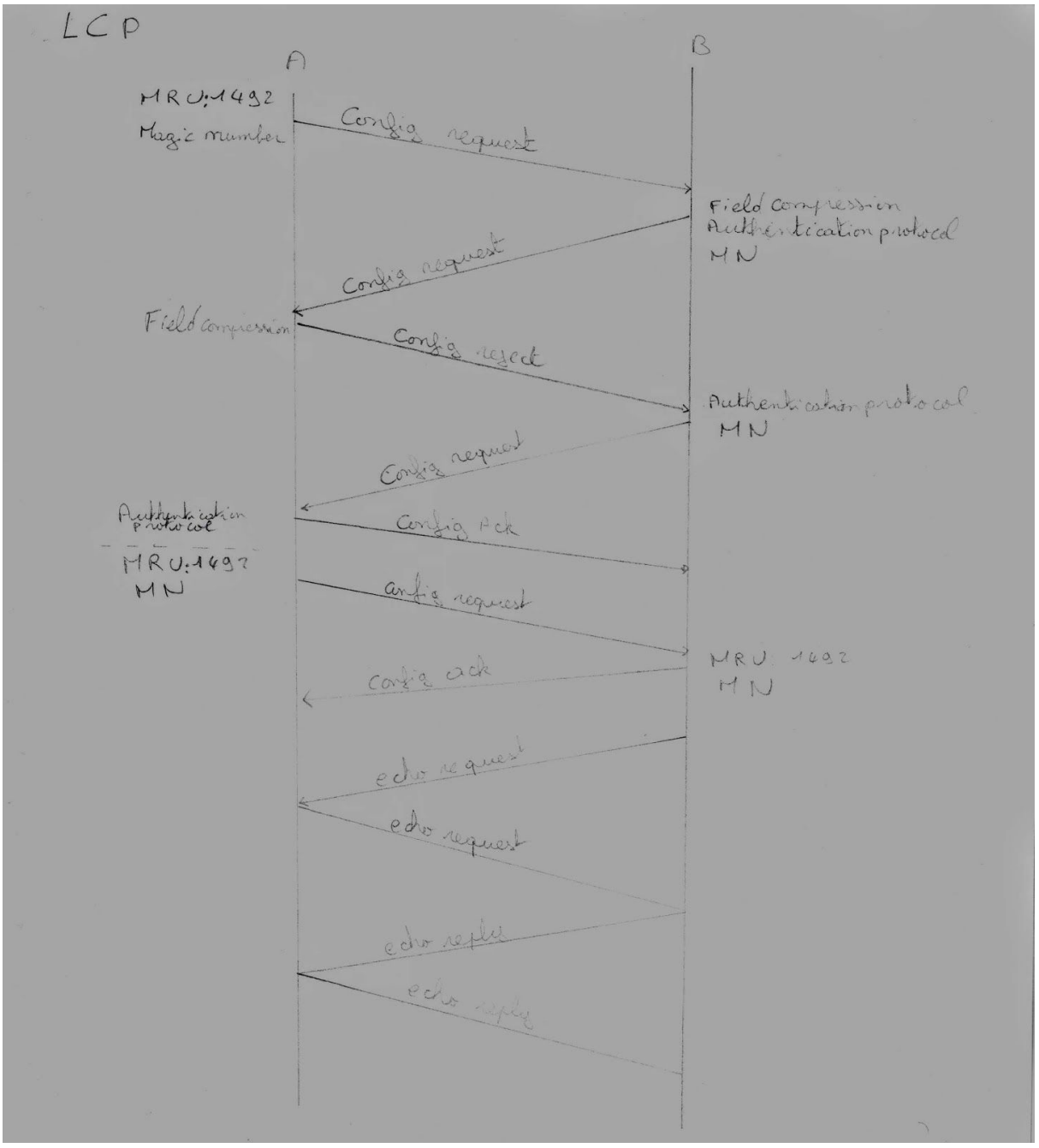
annexe 1 : encapsulation



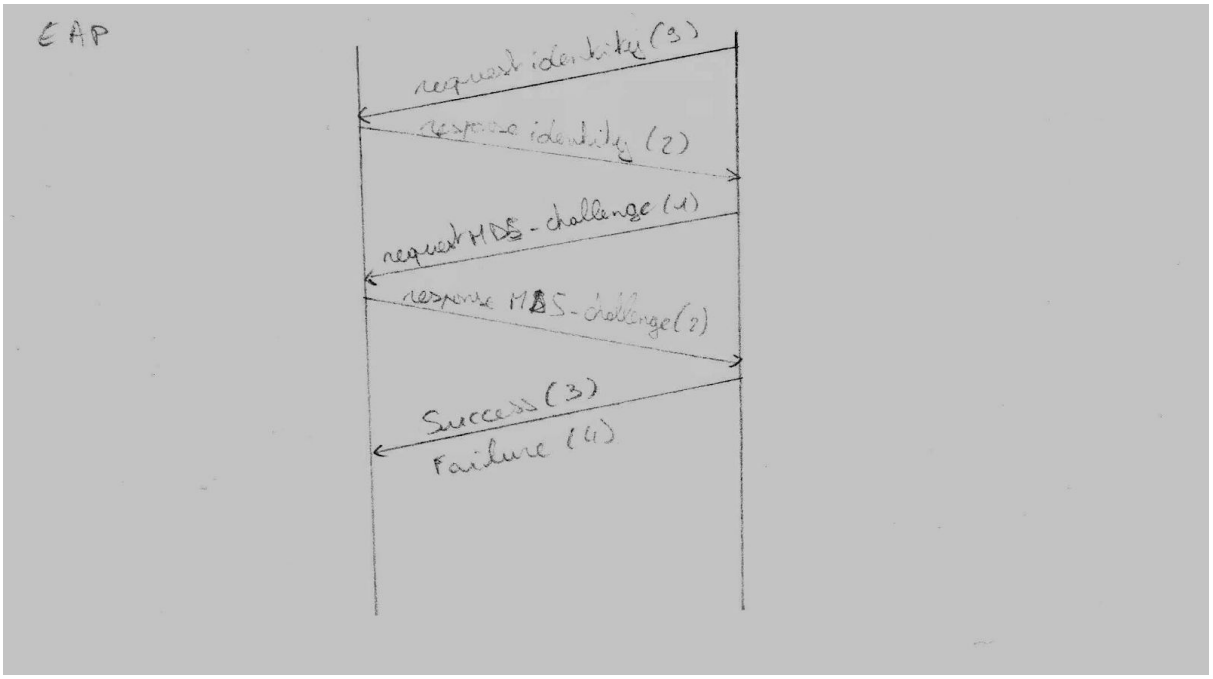
annexe 2 :PPPoE Discovery



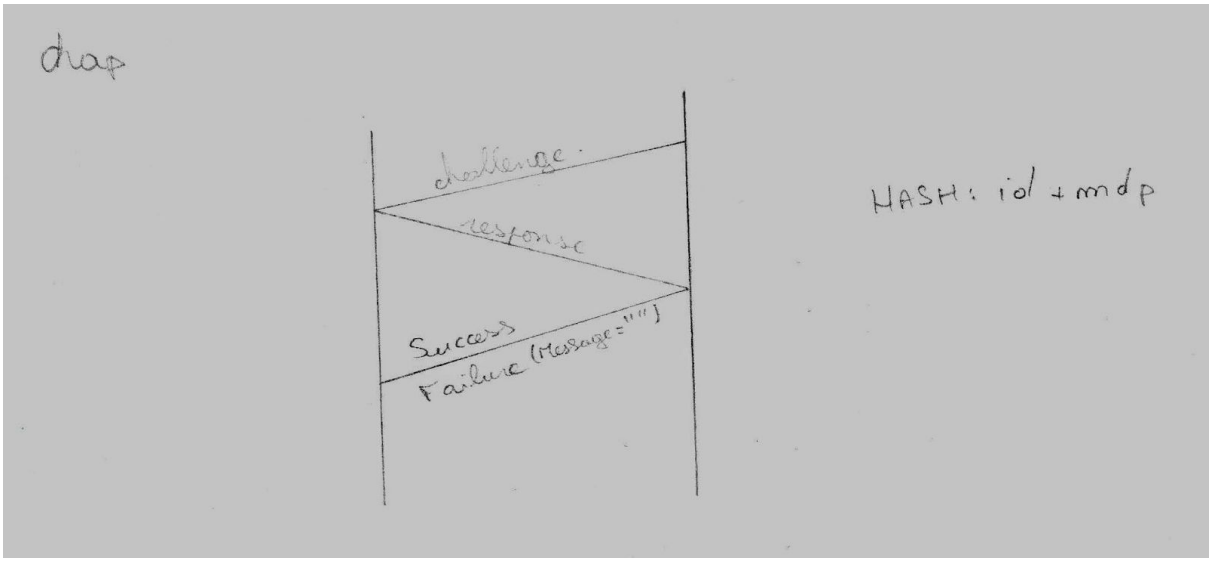
annexe 3 : echange LCP



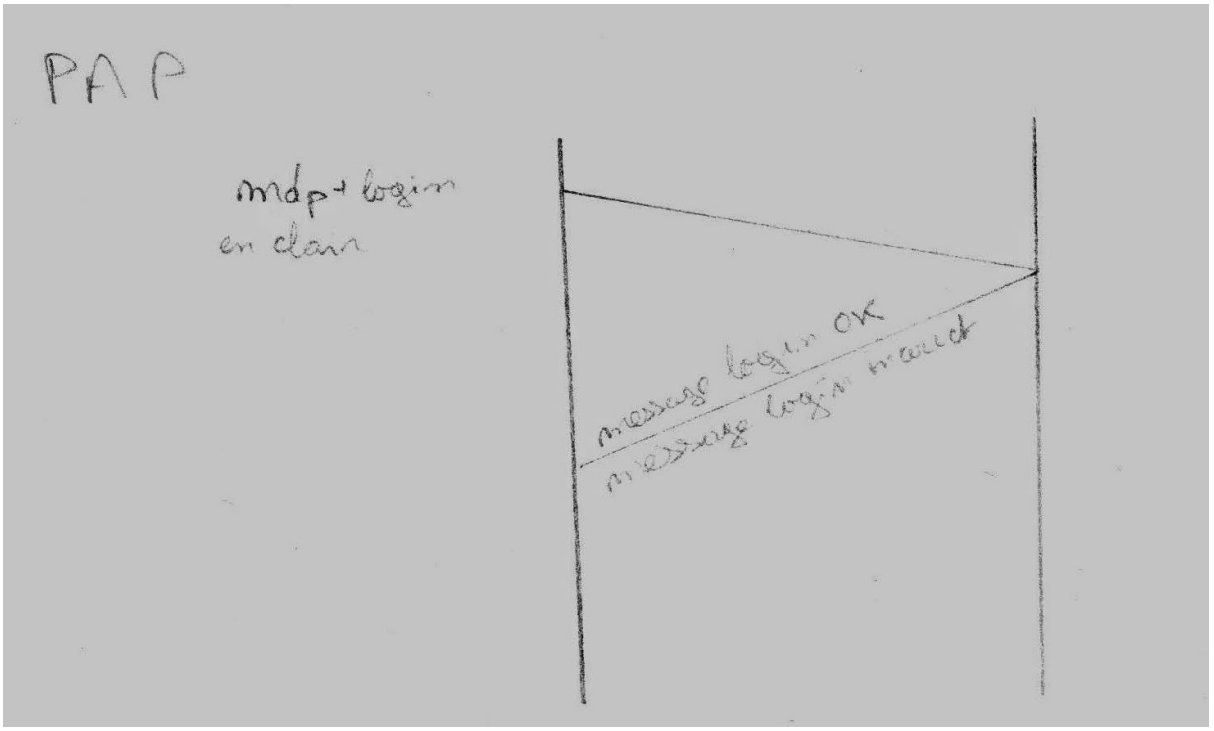
annexe 4 : EAP



annexe 5 : CHAP



annexe 6 : sans authentification



annexe 7 : IPCP