

Editions ENI

Les réseaux avec Cisco

Connaissances approfondies sur les réseaux

(3^e édition)

Collection
Ressources Informatiques

Extrait

Chapitre 3

Les concepts fondamentaux

1. Les systèmes de numération

1.1 Le système binaire

Le système binaire est essentiel car c'est l'unique représentation que les systèmes informatiques sont en mesure d'interpréter.

Ainsi, qu'il s'agisse de caractères alphanumériques ou de représentations graphiques, un système informatique ne peut les traiter qu'à la condition qu'ils lui soient soumis sous forme binaire.

Cette limitation des systèmes informatiques est due au fait qu'il ne s'agit, après tout, que de systèmes électroniques qui ne connaissent que les deux états fondamentaux : 0 et 1.

Dans le cadre des réseaux informatiques, le système binaire fait partie des concepts fondamentaux à maîtriser car il sert de base à un certain nombre de fonctions importantes comme l'adressage.

Le système binaire est une méthode de numération qui semble très simple puisqu'elle ne dispose que de deux valeurs : 0 et 1.

Il est également appelé système en base 2 (puisque'il ne contient que deux possibilités de valeurs). On dénomme "bit" un chiffre binaire, un bit ne peut donc avoir que deux valeurs : 0 ou 1.

Si ce système ne comporte que deux valeurs, comment est-il possible de représenter des valeurs décimales voire des caractères alphanumériques ? Réponse : la combinaison de plusieurs bits permet d'exprimer des valeurs plus complexes.

Il est important de comprendre également la notion d'octet (byte en anglais). Un octet est le regroupement de 8 bits dans le but de coder une information. Chacun de ces 8 bits pouvant se caractériser en deux états différents (0 ou 1), il est alors possible de coder 2^8 combinaisons différentes soit 256 valeurs.

Mathématiquement plusieurs méthodes sont valables pour coder un chiffre en binaire.

Une première méthode est de diviser le nombre à convertir par le nombre de valeurs possibles d'un caractère binaire (un bit), soit 2.

- Si le résultat de la division est un nombre à virgule, on note 1.
- Si le résultat de la division est un nombre juste, on note 0.
- La notation reprend ainsi les résultats obtenus de droite à gauche.

Exemple : Conversion d'une valeur décimale en binaire, première méthode

Soit le nombre 97 à convertir :

$97/2 = 48,5$. On note alors 1 car la division n'est pas juste.

$48/2 = 24$. On note alors 0 car la division est juste.

$24/2 = 12$. On note également 0 car la division est juste.

$12/2 = 6$. On note 0 car la division est juste.

$6/2 = 3$. On note 0.

$3/2 = 1,5$. On note 1.

$1/2 = 0,5$. On note 1.

On reprend les résultats obtenus de droite à gauche, du dernier au premier : 1100001.

Le nombre 97 se code en binaire en utilisant seulement 7 bits.

La seconde méthode semble plus rapide pour les calculs qui concernent les adressages réseau.

Il s'agit de considérer la valeur décimale de chacun des bits composant un octet. Un octet est équivalent à 8 bits :

Bit 8	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1
128	64	32	16	8	4	2	1

À partir de ce tableau, il est possible de coder 97 en binaire.

Exemple : Conversion d'une valeur décimale en binaire, seconde méthode

Soit le nombre 97 à convertir.

97 est inférieur à 128, on note 0.

97 est supérieur à 64, on note 1 et on effectue la soustraction $97-64$, il reste 33.

33 est supérieur à 32, on note 1 et on effectue la soustraction $33-32$, il reste 1.

16 est supérieur à 1, on note 0.

8 est supérieur à 1, on note 0.

4 est supérieur à 1, on note 0.

2 est supérieur à 1, on note 0.

1 est égal à 1, on note 1 et il ne reste rien.

La notation donne 01100001, le zéro initial peut être omis.

Cette dernière méthode est également utilisable pour réaliser l'opération inverse, c'est-à-dire convertir une valeur binaire en son équivalent décimal. Puisque les valeurs décimales de chacun des bits sont connues, il est possible de les additionner pour trouver l'équivalence décimale.

Ainsi si vous reprenez l'exemple précédent avec l'octet 0110 0001, vous pouvez calculer avec le tableau de valeur décimale des bits que l'octet vaut $0 + 64 + 32 + 0 + 0 + 0 + 0 + 1$, soit 97.

Si le chiffre ne peut pas être codé en binaire sur 8 bits, il est alors nécessaire d'ajouter autant de bits que nécessaire. Rappelez-vous que le nombre maximum de combinaisons avec 8 bits est de 256 (256 est le résultat de 2^8). Rien ne vous interdit d'en utiliser 9 ou plus selon le besoin. Pour 9 bits le nombre maximum de combinaisons est de 512 (512 est le résultat de 2^9).

Dans ce cas la valeur décimale des bits ajoutés est supérieure à celle des bits précédents :

Bits 10	Bits 9	Bit 8	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1
512	256	128	64	32	16	8	4	2	1

1.2 Le système hexadécimal

Le système hexadécimal est aussi dénommé système en base 16 car il contient 16 caractères permettant de coder l'information.

La correspondance hexadécimale-numérique est très simple et elle implique l'utilisation de lettres :

Numérique	Hexadécimal
0	0
1	1
2	2
...	...
10	A

Numérique	Hexadécimal
11	B
12	C
13	D
14	E
15	F

C'est sur cette base que les informations sont codées. Ainsi le chiffre décimal 12 est noté de manière hexadécimale 0x0C.

Comme il y a 16 valeurs hexadécimales dans une valeur hexadécimale et qu'elles sont représentées par deux, cela signifie qu'on peut y coder 16×16 possibilités soit 256 valeurs.

On peut donc en déduire que chaque caractère hexadécimal vaut 4 bits et que deux caractères hexadécimaux valent 1 octet.

La conversion hexadécimale vers binaire est assez simple puisque chaque caractère hexadécimal vaut 4 bits. Reprenons l'exemple précédent avec 0x0C.

Les deux caractères hexadécimaux sont respectivement 0 et C. Le premier caractère vaut 0 en décimal et le second caractère représente 12 en décimal.

On pourra ainsi écrire 0x0C en binaire sous la forme 0000 1100.

Mathématiquement il est possible de formuler l'opération en élevant 16 à la puissance de la valeur binaire du caractère hexadécimal.

Prenons par exemple la valeur hexadécimale FE qui vaut en binaire 1111 1110 soit 254.

L'opération est la suivante :

$$(15 \times 16^1) + (14 \times 16^0) \text{ soit } 240 + 14 = 254$$

Prenons un nombre hexadécimal plus complexe, par exemple FE 34 89.

L'opération est la suivante :

$$(15 \times 16^5) + (14 \times 16^4) + (3 \times 16^3) + (4 \times 16^2) + (8 \times 16^1) + (9 \times 16^0)$$

$$15728640 + 917504 + 12288 + 1024 + 128 + 9 = 16659593$$

Ou encore en binaire en partant du fait que chaque caractère vaut 4 bits :

1111 1110 0011 0100 1000 1001

Comme on peut le constater, l'avantage principal de la notation hexadécimale est qu'elle permet d'écrire des valeurs importantes de manière réduite.

Cette notation devient de plus en plus importante de nos jours dans le monde du réseau à cause d'IPv6 dont la représentation de l'adressage s'effectue en hexadécimal.

1.3 Le système décimal

L'origine de ce système est évidemment liée au fait que nous possédons dix doigts.

Le nombre 10, caractéristique de la base décimale, intervient par ses différentes puissances appelées « poids ». Au niveau de l'écriture, seuls apparaissent les coefficients. La multiplication des coefficients par l'indice de base est implicite. Les coefficients peuvent prendre toutes les valeurs inférieures à 10, c'est-à-dire inférieures à la base, soit 0, 1, 2, 3, 4, 5, 6, 7, 8 et 9 (dix symboles).

La numération décimale est une numération de position car chaque rang (chiffre) est pondéré selon sa position, par opposition au code additionnel que les Romains utilisaient (exemple : VII correspondait à 5 + 1 + 1).

Editions ENI

Les réseaux

**Configurez routeurs et commutateurs
Exercices et corrigés**

(5^e édition)

Collection
Les TP Informatiques

Extrait

Chapitre 3

Configuration avancée d'un routeur

Durée : 4 heures 40

Mots-clés

TFTP, PPP, Frame Relay, NAT, PAT, DHCP, HSRP

Objectifs

- Utiliser des encapsulations avancées sur les liens série.
- Distribuer des adresses IP aux clients.
- Partager un accès réseau.

Après avoir configuré les options basiques d'un routeur, telles que le nom, l'horloge et surtout les adresses IP sur les interfaces, ce chapitre présente la mise en place de solutions avancées. Ces solutions permettent de sécuriser, d'optimiser ou de partager les connexions réseau. Certains protocoles permettent d'améliorer ou de répondre à des configurations spécifiques du réseau, tel PPP ou Frame Relay, d'autres permettent de dépasser le fonctionnement classique du routeur pour fournir d'autres services aux clients du réseau comme le NAT ou le DHCP.

Matériel à prévoir

Les travaux pratiques suivants sont basés sur les routeurs de la série 1900, 2600, 2800 ou 2900. Il est conseillé de faire tous les exercices à la suite, la plupart des TP réutilisent les configurations précédemment réalisées.

Un atelier se base sur le service TFTP, c'est un programme à installer sur l'ordinateur relié au routeur. Ce programme peut se trouver librement sur Internet en version freeware, pour tous les systèmes d'exploitation.

Dans les TP suivants, les liens entre les routeurs sont réalisés à l'aide de câbles série, il faut donc faire attention à la configuration de la bande passante (définie à l'aide de la commande `clock rate`) sur le côté DCE (*Data Communications Equipment*) du câble. Cette information est inscrite sur une étiquette près du connecteur, ou bien accessible à l'aide de la commande `show controllers serial`.

Prérequis

Pour valider les prérequis nécessaires avant d'aborder le TP, indiquez si les affirmations suivantes sont vraies ou fausses.

1. Le protocole TFTP nécessite une authentification de la part du client.
2. Le protocole de couche 2 par défaut sur les routeurs est le PPP.
3. La traduction d'adresse permet uniquement de partager un accès Internet.
4. Le protocole PPP prend en charge l'authentification.
5. Les liens série ne supportent qu'un protocole d'encapsulation à la fois.
6. Le service DHCP permet de distribuer des options aux clients (DNS, Passerelle).

Corrigé p. 177

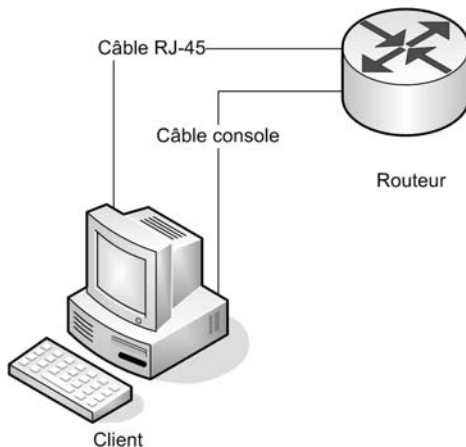
Énoncé 1 Chargement d'IOS sur TFTP

Durée : 20 minutes

Le serveur TFTP permet de sauvegarder la configuration du routeur ainsi que les IOS (*Internetworking Operating System*).

Lors du démarrage d'un routeur, l'IOS est directement chargé à partir de la mémoire flash par défaut. Il est possible de charger l'IOS à partir de deux endroits supplémentaires, la ROM (*Read Only Memory*) et un serveur TFTP.

Voici le schéma de l'atelier :



Pour cet exercice, configurez le routeur de manière à ce qu'il charge l'IOS depuis un serveur TFTP.

Corrigé p. 178

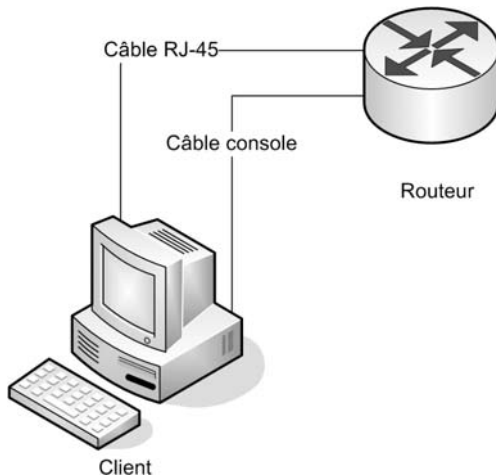
Énoncé 2 Modification de la séquence d'initialisation du routeur

Durée : 20 minutes

La séquence d'initialisation du routeur qui consiste à spécifier des options de démarrage du routeur est gérée par un nombre appelé "valeur de registre de configuration".

Cet exercice consiste à manipuler cette valeur afin de voir comment réagit le routeur en conséquence. Cet exercice suppose que le routeur est configuré au minimum (configuration par "setup").

Voici le schéma de l'atelier :



Suite au démarrage du routeur, en mode utilisateur ou privilégié, tapez la commande `show version`. Notez la valeur spécifiée en dernière ligne, valeur qui correspond à `0xXXXX` où X représente des chiffres (exemple : `0x2102`).

Entrez la valeur `0x2142` comme valeur de registre, puis redémarrez le routeur, que se passe-t-il ?

Configurez rapidement le routeur à l'aide du mode "setup" en spécifiant comme nom de routeur "`_routeur`", puis redémarrez à nouveau, que se passe-t-il ?

Ignorez le mode "setup", puis saisissez la valeur notée précédemment comme valeur de registre. Redémarrez le routeur, que se passe-t-il ?

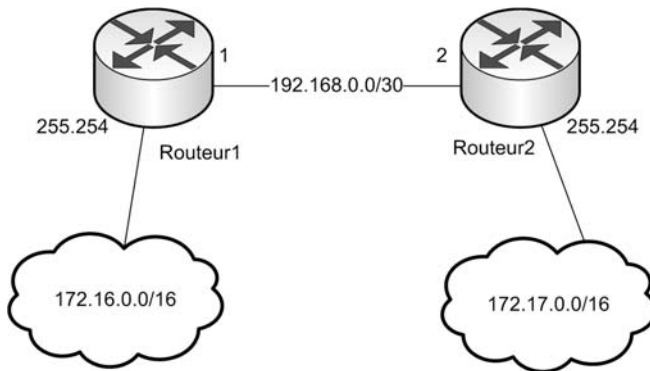
Corrigé p. 179

Énoncé 3 Encapsulation PPP

Durée : 30 minutes

Par défaut, sur un lien série, les routeurs emploient le protocole de liaison de données HDLC (*High-level Data Link Control*). Il est cependant possible, pour bénéficier d'options supplémentaires, de changer ce protocole pour le remplacer par PPP (*Point-to-Point Protocol*), relais de trame...

Voici le schéma de l'atelier :



Mettez en place le TP, puis configurez chaque routeur et ordinateur client afin de fournir un réseau fonctionnel. Remplacez le protocole HDLC par PPP. Vérifiez que le routage fonctionne toujours. Utilisez le protocole d'authentification CHAP (*Challenge Handshake Authentication Protocol*) à la place de PAP (*Password Authentication Protocol*) pour la liaison série. Vérifiez que le routage fonctionne toujours.

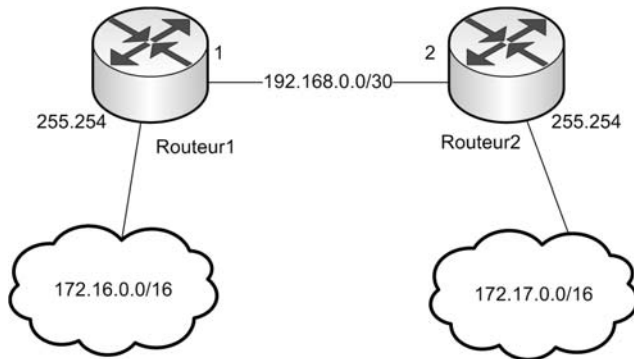
Corrigé p. 180

Énoncé 4 Encapsulation Frame Relay

Durée : 30 minutes

Par défaut, sur un lien série, les routeurs emploient le protocole de liaison de données HDLC (*High-level Data Link Control*). Il est cependant possible d'utiliser d'autres protocoles comme par exemple le relais de trame (*Frame Relay*) selon la topologie du réseau.

Voici le schéma de l'atelier :



Mettez en place le TP, puis configurez chaque routeur et ordinateur client afin de fournir un réseau fonctionnel. Remplacez le protocole HDLC par Frame Relay. Vérifiez que le routage fonctionne toujours. Le numéro DLCI est 50 entre les deux routeurs. Le type lmi est cisco. Vous pouvez utiliser la topologie de TP précédente.

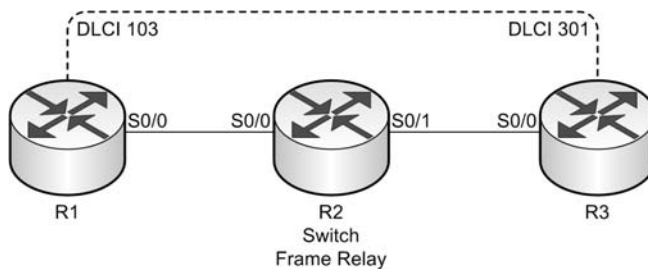
Corrigé p. 181

Énoncé 5 Configuration d'un commutateur Frame Relay

Durée : 30 minutes

Dans le précédent atelier, l'encapsulation Frame Relay a été mise en place entre deux routeurs directement reliés. Dans la réalité, les routeurs sont reliés par des commutateurs (switchs) Frame Relay gérés par les fournisseurs d'accès.

Cet atelier vous propose de configurer un switch Frame Relay, ainsi que deux routeurs. Voici le schéma de l'atelier :



Mettez en place le TP puis configurez le commutateur Frame Relay comme ci-après :

```
R2>enable
R2#configure terminal
R2(config)#frame-relay switching
R2(config)#interface Serial0/0
R2(config-if)#no ip address
R2(config-if)#encapsulation frame-relay
R2(config-if)#clockrate 56000
R2(config-if)#frame-relay intf-type dce
R2(config-if)#frame-relay route 103 interface Serial0/1 301
R2(config-if)#interface Serial0/1
R2(config-if)#no ip address
R2(config-if)#encapsulation frame-relay
R2(config-if)#clockrate 56000
R2(config-if)#frame-relay intf-type dce
R2(config-if)#frame-relay route 301 interface Serial0/0 103
R2(config-if)#exit
R2(config)#
```

Suite à la configuration du commutateur, configurez R1 et R3 avec comme adresse IP respective 192.168.0.1/24 et 192.168.0.3/24, puis vérifiez que R1 communique avec R3.

Note

Notez que le commutateur Frame Relay indique le signal d'horloge à R1 et R3, et qu'il ne possède pas d'IP.

Corrigé p. 182

Énoncé 6 Configuration d'un commutateur Frame Relay et de trois routeurs

Durée : 30 minutes

Dans le précédent atelier, un commutateur Frame Relay a été mis en place pour relier deux routeurs.

Cet atelier vous propose de configurer un switch Frame Relay, ainsi que trois routeurs. Pour avoir une connectivité entre les routeurs R3 et R4, un mappage statique doit être mis en place.