

Examen de Réseau

Licence – Licence GMI – Maîtrise

(Tous documents autorisés)

Contexte du sujet :

Une entreprise spécialisée dans la production de matériel de mesure et d'analyse météorologique propose des stations de mesure automatisées qui permettent d'obtenir en « temps réel » les conditions de pression, de température et d'hygrométrie (entre autre) sur le lieu géographique où est installé le système. Ces automates sont utilisés par exemple, par les opérateurs des tours de contrôles des aéroports, par les laboratoires d'étude de l'évolution du climat ou encore par les grandes fermes de l'industrie agro-alimentaire.

Récemment employé par cette société, il vous est demandé de permettre l'exploitation à distance des stations de mesure. Les clients souhaiteraient pouvoir consulter les mesures effectuées par les stations automatiques (et les reconfigurer) directement sur leurs navigateurs WEB. Il vous faut donc équiper les stations automatiques de mesure d'un serveur WEB. (Elles possèdent déjà une mini VM Java.)

Pour cela les stations de mesure seront équipées par les électroniciens de l'entreprise d'un modem GSM respectant une norme dédiée à l'échange de données sur réseau mobile : 9600 Bauds 2 niveaux de fréquence (9600Hz et 19200Hz) et 2 niveaux de phases (0 et 180°). Les documentations techniques vous indiquent aussi que les octets sont transmis du bit de poids le plus fort au bit de poids le plus faible, en encodant le premier bit par la fréquence et le second par la phase du signal sur un intervalle de temps élémentaire. L'émission (et la réception) des octets est « fiabilisée » par le matériel grâce à l'utilisation d'un bit de parité (parité paire du nombre de 1).

Pour intégrer les stations de mesure dans le réseau de l'entreprise vous utilisez le protocole SLIP pour relayer les paquets IPs depuis les clients « web » jusqu'aux stations de mesures automatiques via des liaisons modem « téléphone sans fil ». Une machine spécialisée jouera le rôle de routeur entre les ordinateurs de l'entreprise et les stations de mesure déployées dans la nature.

Pour connecter les stations de mesure automatiques aux réseaux des entreprises dans lesquelles elles s'intègrent, vous utilisez une machine dédiée appelée « passerelle stations mesure ». Cette passerelle utilise des modems pour établir la liaison physique avec les stations de travail, et une carte Ethernet pour se connecter au réseau des entreprises.

La couche TCP installée sur les stations de mesure utilise une fenêtre glissante réduite au minimum (255 octets). On suppose néanmoins que cela suffit pour recevoir par exemple une requête HTTP complète. En terme d'émission, un paquet TCP est transmis pour chaque ligne de texte (associée à un println, ou terminée par les caractères '\n', '\r') car il n'y a pas de buffer d'émission très important. Un seul paquet est émis puis la couche TCP attend réception d'un ACK pour émettre le paquet suivant (cela lui permet de ne pas garder en mémoire des paquets déjà émis (faute d'avoir suffisamment de RAM sur la station de mesure)).

Couche Physique

Le mécanisme de modulation retenu par les électroniciens se veut plus adapté aux communications des téléphones mobiles qui sont très fortement perturbées par les conditions extérieures (orages, murs, ligne électriques).

Question 1 :

Quel type de modulation possible n'est pas utilisée par ce modem ? Comment expliquez-vous ce choix des électroniciens ?

Question 2 :

Combien de bits d'information sont transmis durant un intervalle de modulation élémentaire pour le modem ? Combien de temps prendra l'émission d'un octet ? Sachant que les prestataires de téléphonie mobile facturent automatiquement la première minute de communication, combien d'octets peuvent être transférés durant cette première minute depuis la station automatique vers le serveur central ?

Question 3 :

Donnez le signal généré par le modem pour transmettre l'octet dont la valeur est 78.

Question 4 :

Dans certaines configurations, les stations de mesures automatiques sont installées dans des zones où il n'existe aucune couverture téléphonique. Dans ce cas il est prévu d'utiliser le même système de modulation, dirigé vers des satellites géostationnaires situés à 36000 km de la terre. En supposant que le signal est transmis à la vitesse de 150000 km/s, combien de temps mettra le premier octet d'un bloc de données pour arriver au satellite en partant de la station de mesure ? Combien de temps faudra-t'il pour transmettre un paquet de 256 octets ?

Liaison de données

Le responsable de l'installation de l'électronique dans la station de mesure vous explique que la transmission est « fiabilisée » matériellement car le modem utilise un bit de parité associé à chaque mot de huit bits transmis.

Question 1 :

Si le modem utilise un bit de parité pour permettre de vérifier chaque octet transmis, combien d'octets peuvent être transmis par la couche liaison en une minute ?

Question 2 :

Calculez le bit de parité des octets 78, 32, 131, 24 et 19. En supposant que le dernier bit reçu est le bit de parité, donnez la valeur décimale des octets reçus lorsque le modem décode les séquences de 9 valeurs binaires suivantes :

1. « 000000011 »
2. « 011000110 »
3. « 100111100 »
4. « 010001111 »

Question 3 :

Dans le cas où la transmission est effectuée vers un satellite (Question I.3), quelle est la perte de temps qu'occasionne l'émission d'un paquet de 256 octets transmis avec ce système de bit de parité ?

Pour chaque paquet reçu, le satellite retourne un accusé de réception (1 seul octet) : négatif s'il y a une erreur, positif si tout est OK. Combien de temps prend alors l'émission du paquet + accusé de réception ? Que peut-on faire au niveau de la couche liaison pour éviter d'attendre l'aller/retour entre la station et le satellite ? (Préciser la configuration optimale de votre solution pour ce contexte)

La machine virtuelle java sur laquelle repose les applications embarquées dans la station de mesure automatique, fournit une interface d'accès au modem. Vous ne disposez initialement que d'une classe `PhoneLink` décrite ci-dessous :

```
Class PhoneLink
{
    public PhoneLink(String PhoneNumber) throws IOException;
    public static PhoneLink acceptACall() throws IOException;
    public static PhoneLink getTheCurrentLink();

    public void sendAByte(int aByte) throws IOException ;
    public int getAByte() throws IOException ;

    public String getPhoneNumber() ;
    public void finish() ;
    public boolean isFinish() ;
}
```

Le constructeur permet d'obtenir une nouvelle connexion avec le destinataire dont le numéro de téléphone est indiqué dans la chaîne de caractères passée en paramètre (Exception si la ligne est occupée ou hors service).

La méthode de classe `acceptACall()` permet d'attendre qu'une connexion distante soit initiée. Elle retourne une nouvelle liaison (`PhoneLink`).

Enfin si une liaison existe déjà il n'est pas possible d'en créer une autre (il n'y a qu'un téléphone de disponible !). Pour obtenir une référence sur une liaison déjà existante il est possible d'utiliser la méthode statique `getTheCurrentLink`. Cette méthode retourne `null` s'il n'y a aucune liaison téléphonique en cours.

En disposant d'un objet `PhoneLink` il est possible de recevoir (`getAByte`) ou d'émettre un octet (`sendAByte`). Ce sont des `int` qui sont proposés par la classe `PhoneLink` car les valeurs entre 0 et 255 codent le contenu d'un octet (alors qu'un `byte` « java » permet de coder des valeurs dans l'intervalle [-128 ; +127]). La valeur -1 retournée par `getByte` indique la réception d'un octet dont le bit de parité est incorrect.

Il est encore possible de connaître le numéro de l'interlocuteur sur une liaison téléphone déjà établie. C'est la méthode `getPhoneNumber` qui retourne ce numéro dans une chaîne de caractère.

Question 4 :

Comme cette classe ne correspond pas aux fonctionnalités d'une couche liaison de données pour le modèle TCP/IP, il vous faut réaliser le logiciel associé à la couche

manquante. Il s'agit simplement de définir une sous classe de la classe `LinkLayer` qui implante la méthode `SendIPPacket`. Pour faire « au plus simple », vous avez choisi le protocole SIIP pour votre couche liaison.

Ecrivez la méthode `SendIPPacket` de la classe. Pour cela, utilisez la liaison téléphonique en cours si elle existe, ou créez en une vers le numéro de téléphone par défaut contenu dans la variable de classe `String DefaultPhoneNumber` de la classe `LinkLayer`. La méthode `SendIPPacket` prend un paquet IP sous la forme d'un tableau d'octets (`byte[]`) de taille variable et transmet ce paquet IP, octet par octet sur la liaison Modem (via la méthode `SendAByte()`) selon le protocole SIIP. Pour envoyer les octets elle utilise les méthodes offertes par les instances de la classe `PhoneLink`. Voici la déclaration de la méthode à réaliser :

```
synchronized void SendIPPacket(byte[] aPacket) throws  
IOException;
```

Routage

Un client a commandé l'installation d'une dizaine de station de mesure et du système informatique qui permet de les consulter. Ce système informatique est un ordinateur particulier qui est installé parmi les ordinateurs qui composent le réseau LAN du client (fig. 1). Cette machine a pour rôle de permettre le routage des paquets IP depuis les navigateurs des clients jusqu'aux stations de mesures automatiques.

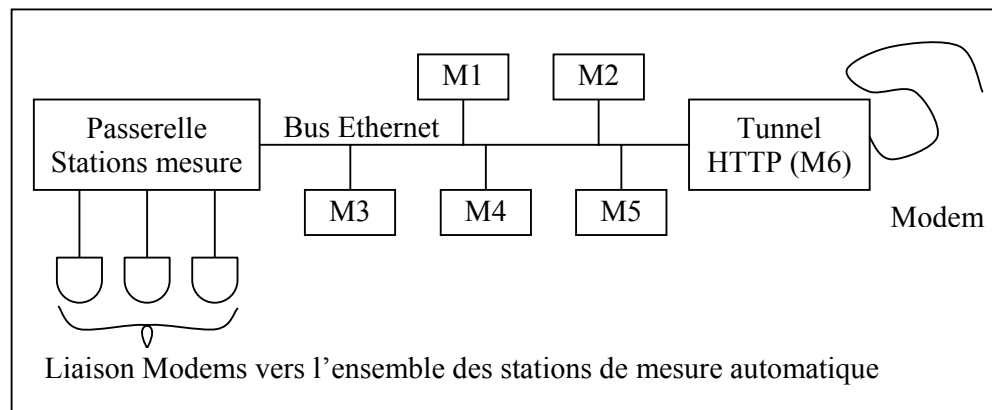


Fig. 1 : Organisation du réseau local du client

La machine « passerelle Stations mesure » est le système installé chez le client pour interconnecter son réseau local avec les stations de mesures distantes. Les machines M1 à M5 composent le réseau local, la machine M6 « tunnel Internet » est en fait une machine comme les autres qui possède une connexion Internet. Elle permet aux autres machines d'accéder au Web en jouant le rôle de tunnel de protocole HTTP (Proxy Web).

Question 1 :

Le client à utiliser pour configurer les machines de son réseau local l'adresse par défaut des systèmes windows. Il s'agit des plages d'adresse IP réservées pour les réseaux intranet (non visible sur l'Internet) de classe C. Donner les adresses des machines (Mx) en supposant que les identifiants de machines correspondent au

numéro des machines dans le réseau LAN (Machine M1 \Rightarrow identifiant de machine = 1, Machine M2 \Rightarrow identifiant de machine = 2, ...).

Question 2 :

La « passerelle stations de mesure » doit permettre aux machines M1 à M6 de se connecter simplement aux stations de mesure via leurs intranets. Proposez une re-organisation du réseau local du client, des sous réseaux, leurs masques d'identification, enfin attribuer une adresse IP à chacune des dix stations de mesure et aux machines M1 à M6 si nécessaire.

Question 3 :

La passerelle vers les stations de mesure est une machine. Donnez ses adresses IP dans votre organisation. Indiquez les commandes système nécessaires pour configurer entièrement les tables de routage de la machine qui joue le rôle de passerelle vers les stations de mesure (on suppose qu'initialement ses tables de routage sont vides). La liaison Ethernet est « IF 1 » la liaison modem vers n'importe quelle station est « IF 2 ».

Question 4 :

Le client souhaiterait avoir accès à ses stations de mesure via n'importe quelle machine disponible sur le réseau Internet. Cependant il ne souhaite pas changer la configuration du réseau telle qu'elle est actuellement. Quelle solution pourrait être envisagée en installant un nouveau programme sur la machine M6 par exemple (expliquer simplement le fonctionnement du programme, pas son implémentation...) ?

Mécanisme de transport

On s'intéresse maintenant au fonctionnement de la couche transport de la communication entre les stations de mesure et les navigateurs Web qui s'y connectent. Le serveur Web fonctionne selon le protocole HTTP/1.0 . Pour la requête étudiée on suppose qu'un seul paquet IP suffit pour transporter la requête HTTP. La réponse du serveur Web, elle, est composée de trois paquets. Il est rappelé que la couche TCP des stations de mesure attend l'accusé de réception d'un paquet envoyé avant d'envoyer le paquet suivant. (cf. Contexte du sujet).

Question 1 :

Proposez une trace de l'ensemble des paquets TCP échangés entre les deux machines M (client navigateur Web) et Station de mesure n°1 (serveur HTTP) associés à une requête/réponse respectant le protocole HTTP/1.0.

Question 2 :

Au cours de cet échange (requête HTTP, réponse HTTP), combien d'octets **la couche TCP** a-t-elle ajoutée à ceux transmis par le protocole HTTP ?

Question 3 :

Vous constatez dans la pratique que le taux de transfert de données est très sensiblement inférieur à celui que vous avez calculé. Comment tester le nombre de paquets que perd la liaison ?

Question 4 :

En moyenne, un paquet sur quatre est perdu. Proposez une nouvelle trace de l'échange de trame entre un client et une station de mesure en supposant que le troisième paquet envoyé par le serveur est perdu.

Applicatif

Les stations de mesure se comportent comme des serveurs HTTP. Les documents qu'elles retournent ne sont pas définis une fois pour toute mais ils sont en général recalculés lors de chaque nouvelle requête des clients.

Les deux principales pages fournies par le serveur sont :

1. La page d'accueil associée à l'URL `http://ipStationMesure/Page1.cgi` présente un formulaire HTML « de type GET » qui demande la saisie d'un mot de passe (le client souhaite être le seul bénéficiaire des appareils...)
2. La seconde page est celle associée au formulaire HTML de la première page. Son URL est `http://ipStationMesure/Page2.cgi&pass=UnMotDePasse`. Elle est donc calculée dynamiquement. Si le mot de passe passé en argument est incorrect le serveur http installé sur la borne retourne une page HTML indiquant « ACCES REFUSE ». Si le mot de passe est correct le serveur fournit une page HTML avec l'ensemble des mesures instantanées fournies par les appareils de mesure de la station. Date courante, température, pression, vent et luminosité.

Question 1 :

Proposez le code HTML associé à la seconde page lorsque le mot de passe est incorrect, et à la seconde page si le mot de passe est correct. Dans le dernier cas, remplacez les valeurs obtenues par le serveur avec ses instruments de mesures par des tags `<?= valeurCalculées>`.

Il n'est malheureusement pas possible d'installer sur les stations de mesures des logiciels aussi « imposants » qu'un serveur HTTP professionnel tant le système informatique est modeste. Vous devez donc réaliser le « mini-serveur » Web en Java.

Question 2 :

Proposez une classe appelée `Serveur` dont le programme principal attend et décode une requête HTTP 1.x « GET », et appelle la méthode `producePage1()` ou `producePage2()` (de la même classe) en fonction de l'URL indiquée. Pour la page 2, notez que l'argument contenu dans l'URL peut être quelconque puisqu'il dépend de ce que le visiteur aura indiqué dans le formulaire de la page 1... (Inutile de prévoir le traitement de plusieurs connexion simultanées)

Question 3 :

Définissez la méthode `void producePage2(String req, Socket s)` qui génère la page 2 via la liaison socket `s` en fonction de la requête http passée en paramètre (`req`) par le programme principal. On utilise la classe `Mesure` qui propose les méthodes publiques statiques `double temperature()`, `double vitesseVent()`, `double luminosite()`, `double pression()`, `String date()`.