

Client-serveur : des sockets TCP aux requêtes HTTP

I. Introduction aux réseaux [NSIP]

A. Terminologie et généralités [NSIP Chap 26.1]

Définition 1 Un réseau est un ensemble de nœuds ou composants reliés entre eux par des liens.

Définition 2 Un nœud ou composant est un équipement informatique (ordinateur, routeur, concentrateur, etc.)

Définition 3 Un lien est le vecteur de transmission entre nœuds par exemple (câbles de cuivre, fibre optique, liaison satellites, ondes radios, etc.)

Définition 4 Une interface est le point de raccordement entre un lien et un nœud. Elle peut être matérielle (carte réseau) ou logicielle (pilote noyau).

Définition 5 Un protocole est un ensemble de règles permettant d'établir, mener et terminer une communication entre deux entités.

B. Modèle en couche [NSIP Chap 26.2 26.3]

Remarque 6 Les réseaux informatiques sont organisés en niveaux d'abstraction nommés couches, supposés indépendants les uns des autres. Ces organisations sont appelées modèle.

Définition 7 le modèle OSI

Nom	PDU	Exemple
Application	Données	HTTP, IMAP, ...
Présentation	Données	TLS, PGP, ...
Session	Données	RTP, NetBIOS
Transport	Segments	TCP, UDP, ...
Réseau	Paquets	IPv4, IPv6, ...
Liaison	Trames	Ethernet, Wifi
Physique	Bits	Ethernet, Wifi

Définition 8 le modèle TCP/IP

Nom	Exemple
Application	HTTP, IMAP, ...
Transport	TCP, UDP, ...
Réseau	IPv4, IPv6, ...
Liaison	Ethernet, Wifi

Remarque 9 Chaque couche offre des fonctionnalités que le niveau supérieur peut utiliser. Elle est censée ignorer le contenu des messages qu'elle transmet, on dit qu'elle encapsule ce contenu.

II. Couches liaison et réseau : comment transmettre des paquets.

A. La couche liaison / physique [NSIP Chap 26.3 p347]

Définition 10 La couche liaison regroupe tous les aspects physiques du réseau. On ne considère alors que les machines directement reliées par la même technologie. On appelle ces réseaux LAN (Local Area Network)

Exemple 11 Par exemple, un réseau de la couche physique peut être un réseau Wi-Fi ou d'un réseau câblé utilisant le protocole Ethernet.

Définition 12 Dans un réseau LAN de type Ethernet, chaque interface matérielle possède une adresse propre de 48 bits appelée adresse MAC. On utilise ces adresses pour communiquer entre machines.

Définition 13 Trame Ethernet

préfixe	adresse destination	adresse source	longueur	données	suffixe
8 octets	6 octets	6 octets	2 octets		16 octets

B. La couche réseau dans internet [NSIP Chap 26.3 p348]

Définition 14 La couche réseau a pour objectif de relier différents réseaux LAN pour permettre d'envoyer des paquets entre n'importe quelle machine, peu importe le réseau physique.

Rôles de l'IP 15

- distribuer des adresses uniques à chaque machine du réseau (IPv4: 32bits, IPv6: 128bits)
- le routage des paquets de proche en proche de la source à la destination

Exemple 16 Il existe divers algorithmes de routages comme RIP ou encore OSPF.

Définition 17 Paquet IPv4

préfixe	adresse IP source	adresse IP destination	données
12 octets	4 octets	4 octets	

Remarque 18 Afin d’éviter de retenir et d’utiliser au quotidien des adresses IP, on utilise plutôt des noms de domaines.

Définition 19 Le système de résolution de noms DNS (Domain Name System), est un protocole de la couche application qui permet d’obtenir une adresse IP à partir d’un nom de domaine comme « agreg-info.org ».

III. Couche transport : la communication entre ordinateurs.

Remarque 20 Dans la couche transport, les paquets sont principalement caractérisé par les numéros de ports qui permettent à la machine distance de distribuer le contenu au service associé.

A. Les protocoles UDP et TCP

1. Le protocole UDP [RESO 6.4]

Définition 21 Le protocole UDP pour User Datagram Protocol, est un protocole sans connexion qui permet exclusivement l’émission de paquets entre applications.

Définition 22 En-tête UDP

2 octets	2 octets
Port source	Port de destination
Longueur UDP	Total de contrôle UDP

Remarque 23 Le protocole UDP ne permet pas de garantir que les données arrivent dans le bon ordre et de manière fiable.

2. Le protocole TCP [RESO 6.5]

Définition 24 Le protocole TCP pour Transmission Control Protocol permet d’envoyer à un programme destination des données de taille arbitraires, dans l’ordre, de détecter les erreurs de transmissions et de retransmettre les données si besoin.

Définition 25 Un socket est une interface logicielle c’est-à-dire un objet qui représente la connexion entre notre programme et le programme de la machine distante sur le réseau.

Appels systèmes 26 On utilise les appels systèmes suivants :

- (SERVEUR) socket()
- (SERVEUR) bind()
- (SERVEUR) listen()
- (SERVEUR) accept()
- (SERVEUR) close()
- (CLIENT) socket()
- (CLIENT) connect()
- (SERVEUR) close()

3. Le protocole QUIC [RESO 6.6.1]

Remarque 27 La simplicité du protocole UDP permet aux applications d’y ajouter leurs propres protocoles.

Définition 28 Le protocole QUIC pour Quick UDP Internet Connections fonctionne sur UDP, il permet de fusionner les connexions avec un même flux. Cela évite l’établissement de plusieurs connexions en parallèle.

B. L’interaction client-serveur

Définition 29 Un serveur [NSIT 23] est un programme qui fournit un service réseau à des clients. On les assimiles parfois à la machine qui exécute le programme.

Définition 30 Le paradigme client-serveur [KURO 2.1.1] est un modèle de communication où un serveur central attend et répond à des requêtes de clients. Les différents clients d’un serveur ne communiquent pas directement entre eux.

Exemple 31 [KURO 2.1.1] Un exemple classique d’architecture client-serveur est le Web : un site web est hébergé par un serveur web qui répond aux requêtes de navigateurs.

Remarque 32 [KURO 2.1.1] Il existe d’autres types d’interactions de communications comme par exemple le modèle pair-à-pair. Ici il n’y a pas de serveur central, les communications se font directement entre les utilisateurs (appelées pairs).

Exemple 33 [KURO 2.1.1] Le modèle pair-à-pair est utilisé par exemple dans des applications qui transfèrent de grandes quantités de données comme le partage de fichiers (BitTorrent) ou les appels vidéo (Skype avant 2012).

Remarque 34 CDN [RESO 1.2.3] Pour palier à la centralisation des ressources dans le modèle client-serveur, les fournisseurs de services peuvent faire appel à des CDN (content delivery network).

IV. Couche application : l'exemple du Web

A. Le protocole HTTP [RESO Chap 7.3.4]

Définition 35 URL pour Uniform Resource Locator est une chaîne de caractères permettant d'accéder à un contenu sur Internet. La syntaxe simplifiée est « protocole://nom-ou-adresse/document »

Remarque 36 Le protocole utilisé dans le Web sera souvent http ou https, la partie nom-ou-adresse le nom de domaine ou l'adresse IP, et le document est le chemin permettant de localiser une ressource.

Définition 37 Le protocole HTTP pour HyperText Transfer Protocol, est un standard qui définit le format d'encapsulation des messages envoyés entre un navigateur web et le serveur Web. Les messages du client sont appelés requêtes et ceux du serveur réponses.

Définition 38 Le format HTML pour HyperText Markup Language est un langage de balisage (qui décrit le formatage d'un document) principalement utilisé pour écrire des pages Web statiques (qui ne sont pas modifiées à chaque requête).

Définition 39 Les Méthodes du protocole HTTP permettent d'autres opérations que la requête de lecture d'une page, méthode GET. Les plus utilisées sont aussi POST pour l'ajout de contenu à la requête mais aussi PUT pour les requêtes de stockage de page.

Définition 40 Chaque requête est suivie d'une réponse représentée par un code, par exemple le code 200 pour un Succès et 404 pour une page inexistante.

DEV **Exemple 41** Accès à une page Web, les différents protocoles et requêtes effectuées.

B. La sécurité avec HTTPS [NSIT Chap 24]

Définition 42 le protocole HTTPS pour HyperText Transfer Protocol Secure, est le protocole HTTP auquel on a ajouté une couche de cryptographie, c'est la combinaison de HTTP sur TLS, en anglais HTTP over TLS.

Remarque 43 Pour des questions de vie privée, on peut appliquer la même méthode au protocole DNS, avec le DNS over TLS parfois remplacée de manière dissensuelle par DNS over HTTPS.

Définition 44 Les algorithmes de chiffrements symétriques utilisent la même clé pour chiffrer et déchiffrer.

Exemple 45 Les algorithmes AES et ChaCha20 sont considérés comme sûrs et sont efficaces. Mais ils supposent de pouvoir échanger la clé symétrique de façon sûre.

Remarque 46 On peut utiliser le protocole d'échange de clé Diffie-Hellman pour échanger les clés, mais cet algorithme est victime de l'attaque par intermédiaire (MITM en anglais).

Définition 47 Les algorithmes de chiffrements asymétriques sont caractérisés par la présence de deux clés, l'une pour chiffrer et l'autre pour déchiffrer.

Exemple 48 RSA est un protocole de chiffrement asymétrique.

Définition 49 Un certificat est une preuve d'identité, fournie par un tiers de confiance appelé autorité de certification.

Définition 50 Le protocole TLS ajoute une phase permettant l'authentification du serveur par certificat puis la mise en place de manière sécurisée d'une clé de chiffrement symétrique appelée clé de session.

Client-serveur : des sockets TCP aux requêtes HTTP		9	Rem
<u>I. Introduction aux réseaux</u> [NSIP]		<u>II. Couches liaison et réseau : comment transmettre des paquets.</u>	
<u>A. Terminologie et généralités</u> [NSIP Chap 26.1]		<u>A. La couche liaison / physique</u> [NSIP Chap 26.3 p347]	
1	Def Un réseau	10	Def La couche liaison
2	Def Un nœud	11	Ex
3	Def Un lien	12	Def
4	Def Une interface	13	Def Trame Ethernet
5	Def Un protocole	<u>B. La couche réseau dans internet</u> [NSIP Chap 26.3 p348]	
<u>B. Modèle en couche</u> [NSIP Chap 26.2 26.3]		14	Def La couche réseau
6	Rem Les réseaux informatiques	15	Rôles de l'IP
7	Def le modèle OSI	16	Ex
8	Def le modèle TCP/IP		
		25	Def Un socket
17	Def Paquet IPv4	26	Appels systèmes
18	Rem		
19	Def Le système de résolution de noms	27	Rem
<u>III. Couche transport : la communication entre ordinateurs.</u>		28	Def Le protocole QUIC
20	Rem	<u>B. L'interaction client-serveur</u>	
<u>A. Les protocoles UDP et TCP</u>		29	Def Un serveur [NSIT 23]
21	Def Le protocole UDP	30	Def Le paradigme client-serveur [KURO 2.1.1]
22	Def En-tête UDP	31	Ex [KURO 2.1.1]
23	Rem	32	Rem [KURO 2.1.1]
24	Def Le protocole TCP		
		40	Def
33	Ex [KURO 2.1.1]	41	Ex Accès à une page Web,
34	Rem CDN [RESO 1.2.3]	<u>B. La sécurité avec HTTPS</u> [NSIT Chap 24]	
<u>IV. Couche application : l'exemple du Web</u>		42	Def le protocole HTTPS
<u>A. Le protocole HTTP</u> [RESO Chap 7.3.4]		43	Rem
35	Def URL	44	Def Les algorithmes de chiffrements symétrique
36	Rem	45	Ex
37	Def Le protocole HTTP	46	Rem
38	Def Le format HTML	47	Def Les algorithmes de chiffrements asymétriques
39	Def Les Méthodes	48	Ex
		49	Def
		50	Def Le protocole TLS

Remarque

- Web server / client [NSIP 29 30]

Bibliographie

[NSIP] T. Balabonski & S. Conchon & J. Filliâtre & K. Nguyen, *Numériques et Sciences Informatiques 1er*.

[RESO] A. Tanenbaum & N. Feamster & D. J. Wetherall, *Réseaux, 6° édition, en français*.

[NSIT] T. Balabonski & S. Conchon & J. Filliâtre & K. Nguyen, *Numériques et Sciences Informatiques Terminale*.

[KURO] J. F. Kurose & K. W. Ross, *Computer Networks, a top-down approach, 8th edition*.