# Les faiblesses de la suite TCP/IP

16 Juin 2023




claude@lemarson.com
https://www.lemarson.com

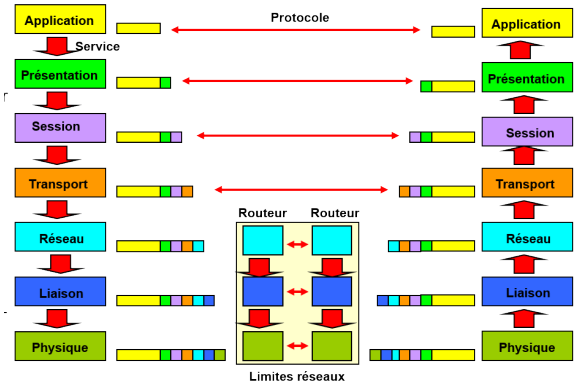
## Sommaire


***Les faiblesses de TCP/IP***

- *De quoi parle-t-on ?*
- *D'où vient la suite TCP/IP, un peu d'histoire*
- *Les motivations d'Arpanet : indépendance et ouverture*
- *Rappel de la structure, positionnement par rapport au modèle OSI*
- *Les faiblesses natives de la pile TCP/IP*
- *Du pain bénit pour les hackers*
- *Très compliqué de modifier l'ordre établi*
- *Quelques solutions quand même*

Plus de 5 milliards d'utilisateurs sont connectés à Internet en 2023, qui Exploitent la suite TCP/IP… Difficile de revenir en arrière.

# De quoi parle-t-on
## (modèle de réseau en couches)

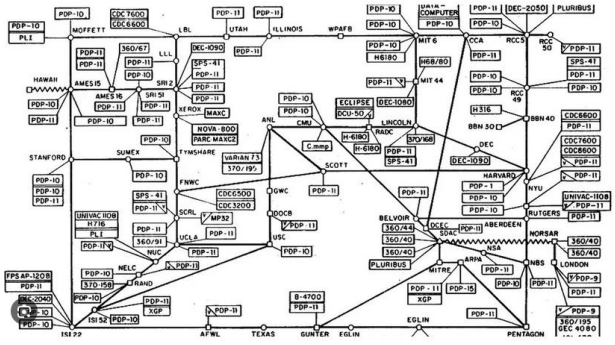



Modèle OSI en 7 couches

- Les fonctionnalités du réseau sont réparties entre différentes couches, dont le nombre dépendra de ce qu'il doit faire
- 7 couches pour le modèle OSI de Zimmermann (arrivé après TCP/IP)
- Chaque couche activée de l'émetteur communique avec la couche équivalente du destinataire.



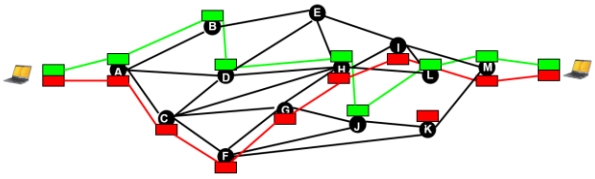
# D'où vient la suite, un peu d'histoire



- Fin des années 60
- Une alternative, deux solutions
  - On reste dépendant d'une architecture propriétaire : SNA d'IBM apparu en 1964 avec obligation d'implémenter le protocole SDLC maison et des nœuds d'architecture propriétaires (frontaux, routeurs)
  - Terminal de la même "marque"
  - Ou solution ouverte, avec protocoles d'interconnexion et indépendance par rapport à l'identité des machines hôtes
- On manque d'expérience, les organisations indépendantes étant vues d'un "mauvais œil" par IBM et les autres majors (Sperry Univac, Control Data…)
- Contexte très "léger", Arpanet ne concernait à ses débuts que quelques universités, centres de recherche, les militaires, des entreprises "amies" convaincues de l'intérêt des systèmes ouverts.

# Les motivations d'Arpanet, un réseau ouvert

- Arpanet est mis en service le 29 octobre 1969 à UCLA.
- Les pères fondateurs : Vinton Cerf, Steve Crocker, Jon Postel, Mike Wingfield, Charlie Kline
- Usage d'un même application par plusieurs utilisateurs de machines différentes
  - Indépendance vis-à-vis de l'OS (gros changement par rapport au contexte IBM et SNA).
- Réseau à commutation de paquets
  - Les messages sont découpés en paquets individuels et acheminés. séparément, ce qui permet d'optimiser l'usage des liens.
  - Diversité des protocoles de routage.
  - Limitation de la charge induite sur l'infrastructure.
- Mode "passif" : le serveur est à l'écoute des demandes.
- Système de livraison de bout en bout.
- Mécanisme de contrôle de flux pour éviter les congestions (système de fenêtrage).
- La suite TCP/IP est finalisée en 1983, avec l'aide de partenaires dont les français de Cyclade (Louis Pouzin), mais divergence par la suite.

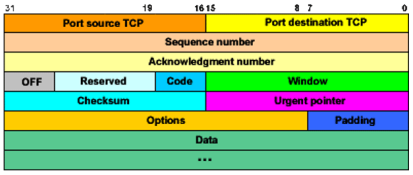



Les paquets appartenant au même message empruntent des chemins différents en fonction du caractère du routage : statique, adaptatif, dynamique

- En mode commutation de paquets, ce n'est pas le chemin entre l'émetteur et le récepteur qui est garanti, mais l'acheminement du message, indépendamment du chemin physique emprunté.
- Chaque paquet comporte un "en-tête" qui contient des informations de routage (adresse du destinataire…).
- Les paquets empruntent des chemins différents.
- A l'arrivée, le destinataire recompose le message, avec des paquets en bon ordre.



# TCP (Transmission Control Protocol)




- **Séquence number**

Position des données à transmettre par rapport au segment initial : TCP numérote chaque octet transmis en incrémentant ce nombre 32 bits, non signé : il revient à 0 après $2^{32} - 1$

- **Acknowledgment Number**

Position du dernier octet reçu dans le flux entrant

- **Code**

Six bits qui influencent le comportement du transport TCP

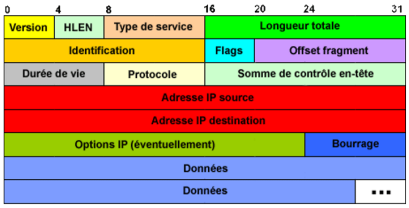
  - **URG** : le champ Urgent Pointer doit être exploité
  - **ACK** : le champ Acknowledgment Number doit être exploité
  - **RST** : Réinitialisation de la connexion
  - **SYN** : le champ Sequence Number contient la valeur de début de connexion
  - **FIN** : l'émetteur a fini d'émettre

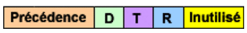
- **Window**

Le flux TCP est contrôlé par un système de fenêtres, dont la taille est fixé dans l'en-tête TCP : entier non signé 16 bits

# Le protocole IP

- Remise des paquets sans garantie.
- Les paquets sont traités indépendamment les uns des autres (connectionless).
- Best effort : les paquets ne sont pas éliminés sans raison.




- **Type de service**
- Indique comment le datagramme doit être géré :
  - PRECEDENCE (3 bits) : définit la priorité du datagramme ; en général ignoré par les machines et passerelles (pb de congestion).
  - Bits D, T, R : indiquent le type d'acheminement désiré du datagramme, ce qui permet à un routeur de choisir entre plusieurs routes (si elles existent) : D pour délai court, T pour débit élevé et R pour grande fiabilité.

- **Version** : numéro de version du protocole
- **HLEN** : longueur de l'en-tête en mots de 32 bits
- **Longueur totale** : longueur totale du datagramme en octets (en-tête + données)
- **Identification, Flags et Offset fragment** : fragmentation liée à sa manière de collaborer avec TCP et à son MTU : Maximum Transfer Unit.
- **Durée de vie** : ce champ indique en secondes, la durée maximale de transit du datagramme sur Internet. La machine qui émet le datagramme définit sa durée de vie.
  - Les routeurs qui traitent le datagramme doivent décrémenter sa durée de vie du nombre de secondes (1 au minimum) que le datagramme a passé pendant son séjour dans le routeur ; lorsque celle-ci expire, le datagramme est détruit et un message d'erreur est renvoyé à l'émetteur.
- **Protocole** : ce champ identifie le protocole de niveau supérieur dont le message est véhiculé dans le champ données du datagramme : 6 = TCP, 17 = UDP, 1 = ICMP.
- **Somme de contrôle** : ce champ permet de détecter les erreurs survenant dans l'en-tête du datagramme, et par conséquent l'intégrité du datagramme.
  Le total de contrôle d'IP porte sur l'en-tête du datagramme et non sur les données véhiculées.

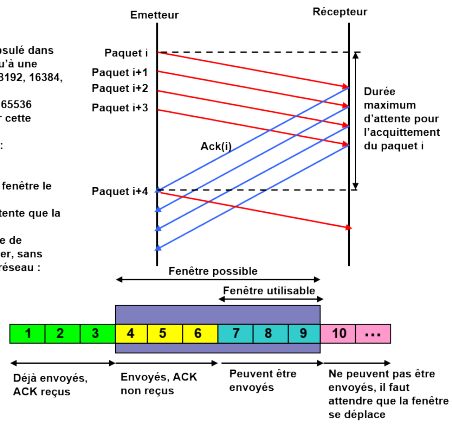


# Le fenêtrage TCP

- Pour éviter les temps d'attente de validation d'un paquet, avant d'en réémettre d'autres, TCP exploite un mécanisme de fenêtres glissantes qui optimise l'usage de la bande passante réseau.
- Chaque paquet est associé à une horloge qui mesure le temps d'attente de l'accusé de réception ACK.
- Si le délai limite est atteint, le paquet est réémis.

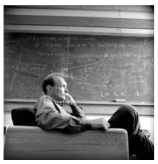
- Sans attendre l'acquittement d'un paquet (qui va être encapsulé dans un datagramme IP), l'émetteur envoie d'autres paquets, jusqu'à une limite fixée par le paramètre window de l'en-tête TCP : 4096, 8192, 16384, qui exprime le nombre d'octets transmissible.
- Window est un paramètre sur 16 bits, normalement limité à 65536 octets, mais une option de l'en-tête TCP, permet de dépasser cette limite.
- A un moment donné, il y a donc quatre natures de paquets :
  - Les paquets déjà envoyés, qui ont reçus leur ACK
  - Les paquets envoyés, en attente de leur ACK
  - Les paquets qui peuvent être envoyés, parce que la fenêtre le permet
  - Les paquets qui ne peuvent pas être envoyés, en attente que la fenêtre se déplace
- Chaque ACK est accompagné d'une nouvelle valeur de taille de fenêtre, qui permet d'ajuster le nombre de segments à envoyer, sans attente de ACK, en fonction des conditions instantanées du réseau : c'est le contrôle de flux, l'un des gros avantages de TC P.

La fenêtre glisse en fonction des paquets validés (ACK). Plus la taille de la fenêtre est grande, plus le débit sera rapide, mais plus le risque augmente de perdre des paquets ou de les dupliquer.



# HTTP

```
GET /page.html HTTP/1.0
Host: example.com
Referer: http://example.com/
User-Agent: CERN-LineMode/2.15 libwww/2.17b3
```

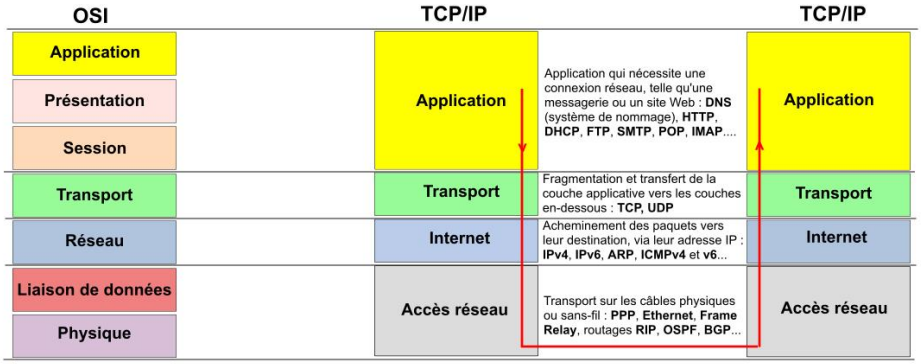


```
HTTP/1.0 200 OK
Date: Fri, 31 Dec 1999 23:59:59 GMT
Server: Apache/0.8.4
Content-Type: text/html
Content-Length: 59
Expires: Sat, 01 Jan 2000 00:59:59 GMT
Last-modified: Fri, 09 Aug 1996 14:21:40 GMT
<TITLE>Exemple</TITLE>
<P>Ceci est une page d'exemple.</P>
```

- Le protocole HTTP : port 80, permet de transférer des fichiers HTML, identifiés grâce à la chaîne de caractères URL, entre un navigateur client et un serveur web
  - Le transfert s'effectue au dessus de TCP
  - Fondé sur un mécanisme de requêtes / réponses
- Les requêtes
  - Une requête est un bloc de lignes adressé au serveur par le navigateur, qui comprend :
  - **Une ligne de requête**, qui précise le type de document demandé, la méthode à utiliser (GET, POST, PUT, TRACE …)
  - La version du protocole
  - **Les champs d'en-tête :** un ensemble de lignes facultatives permettant de donner des informations supplémentaires sur la requête et/ou le client (Navigateur, système d'exploitation …), chacune des lignes étant composée d'un nom qualifiant le type d'en-tête, suivi de deux points (:) et de la valeur de l'en-tête
  - **Le corps de la requête**: un ensemble de lignes optionnelles devant être séparées des lignes précédentes par une ligne vide et permettant par exemple un envoi de données par une commande POST
- Les réponses
  - Une réponse est un bloc de lignes envoyées par le serveur au navigateur, qui comprend :
  - **Une ligne de statut**, qui précise la version du protocole utilisé et l'état du traitement de la requête à l'aide d'un code et d'un texte explicatif. La ligne comprend trois éléments devant être séparés par un espace : la version du protocole utilisé, le code de statut et la signification du code
  - **Les champs d'en-tête de la réponse** : des lignes facultatives qui fournissent des informations supplémentaires sur la réponse et/ou le serveur. Chacune de ces lignes est composée d'un nom qualifiant le type d'en-tête, suivi de deux points (:) et de la valeur de l'en-tête
  - **Le corps de la réponse**, avec le document demandé



# TCP/IP par rapport au modèle OSI

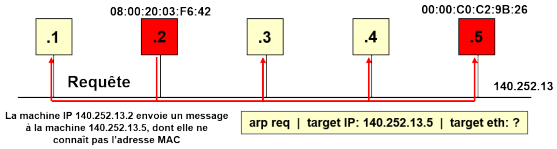


# TCP/IP pain bénit pour les hackers

- Les protocoles de 1983 ont été conçus pour des infrastructures qui n'ont rien à voir avec ce qu'elles sont aujourd'hui.
- Le contexte sécuritaire n'est plus le même :
  - Il s'est mondialisé et durci.
  - On cherche à se défendre avec des boucliers et des arcs, contre des missiles nucléaires.
- La totalité des briques de la suite TCP/IP sont vulnérables.
- La découverte de l'infrastructure d'un réseau TCP/IP est très facile à effectuer : principe même des systèmes passifs (en attente).
  - De l'extérieur, on peut lister les adresses IP associées à des services actifs, il suffit de lister les services à l'écoute. Le "handshake" TCP/IP est conçu pour permettre au client de se connecter sans délai.
  - A chaque nature de service correspondent des attaques spécifiques, efficaces et disponibles.
- Exposition des services
  - On accède à un service par une adresse IP sur 4 ou 16 octets (IPv4 et IPv6), associée à un numéro de port sur 2 octets. Très facile
- La couche Internet ne garantit pas la livraison des paquets, c'est TCP qui joue ce rôle.
- TCP et IP n'ont pas de mécanismes propres d'authentification et de chiffrement : il faut les implémenter dans la couche application.
- Pas de garantie de délai, ni de bande passante minimale imposée.
- De nombreux utilitaires d'attaques sont disponibles : NMAP…
- On cautérise des jambes de bois…



# TCP/IP pain bénit pour les hackers




## ARP

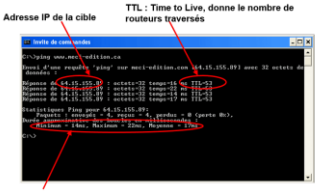
- ARP (Address Resolution Protocol) permet d'obtenir l'adresse MAC physique 48 bits correspondant à une adresse logique IP.
- ARP construit une table de correspondances en interrogeant toutes les machines, qu'il met en cache.
- RARP (Reverse Address Resolution Protocol) fait l'inverse : fournit l'adresse IP d'une adresse MAC.

## ICMP

- ICMP (Internet Control Message Protocol) gère les erreurs générées par les machines connectées
  - Il ne les corrige pas, mais avertit les couches voisines
  - Un message ICMP comporte un type (18 possibles), un code et un message
  - Le message le plus courant : type 8, Code 0, est une demande d'ECHO, utilisée par les PING
- Ping (Packet Internet Grouper) vérifie qu'une machine distante est accessible par le réseau
  - A intervalles réguliers (1 sec souvent), l'émetteur envoie un "echo request" et attend un "echo reply"
  - Utilisé par les outils d'administration de réseaux, mais aussi par les hackers pour inonder les machines et réseaux cibles

| Types de messages ICMP | Message |
|---|---|
| 0 | Réponse à une demande d'écho |
| 3 | Destination inaccessible |
| 4 | Encombrement du réseau |
| 5 | Réorientation |
| 6 | Alternative d'adresse d'un hôte |
| 8 | Demande d'écho |
| 9 | Avertissement d'un routeur |
| 10 | Sélection d'un routeur |
| 11 | Dépassement de temps |
| 12 | Problème de paramètre |
| 13 | Demande d'horodateur |
| 14 | Réponse à une demande d'estampillage |
| 15 | Demande d'information |
| 16 | Réponse à une demande d'information |
| 17 | Demande de masque d'adresse |
| 18 | Réponse à une demande de masque d'adresse |

Exemples de messages ICMP




Temps de propagation en boucle : durée en en ms, d'un aller-retour entre l'émetteur et le récepteur. Doit être en général inférieur à 200 ms.

# Les actions malveillantes


**ARP Spoofing**

- L'attaquant envoie des messages ARP falsifiés, pour associer une adresse MAC à une IP légitime.
- Une fois connecté, l'attaquant reçoit tout le trafic qui ne lui est pas destiné.
- Utilisé pour voler des données sensibles, bloquer des ressources par DoS, voler des identifiants de sessions, etc


**Attaques ICMP**

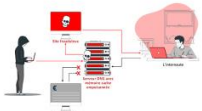
- Le protocole ICMP est utilisé pour submerger une cible.
- "Ping de la mort" : l'attaquant envoie des paquets plus grands que la taille autorisée (65 536 par défaut).
- La victime tombe en "buffer overflow" et devient vulnérable à l'injection de code malicieux.


**Port scanning**

- Pour détecter l'identité des services actifs et leur appliquer des traitements de récupération malveillants : identité des utilisateurs, profils d'accès, etc.
- On retrouve le "port scanning" dans de nombreuses techniques : ARP, SYN...

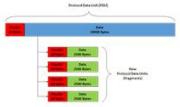
**Attaque MitM ("Man in the Middle")**

- L'attaquant s'insère entre le client et le serveur pour intercepter, lire et modifier les messages entre les correspondants.



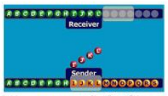
**Empoisonnement DNS**

- L'attaquant fait croire au service DNS que les fausses données qui lui sont envoyées sont légitimes.
- Il "emposonne" sa mémoire cache.
- Les données fausses restent en cache jusqu'à l'expiration du TTL (Time to Live).



**Désynchronisation**

- Les messages sont découpés en paquets et réassemblés à l'arrivée.
- Un attaquant peut s'insérer dans le processus et empêcher le destinataire de retrouver les données dans le bon ordre.
- L'attaquant peut insérer ses propres trames.
- En 2012, l'IETF a proposé un algorithme qui contribue à éviter la supercherie.



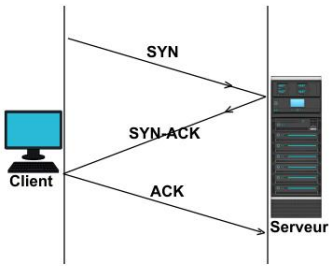

**Attaques sur le fenêtrage**

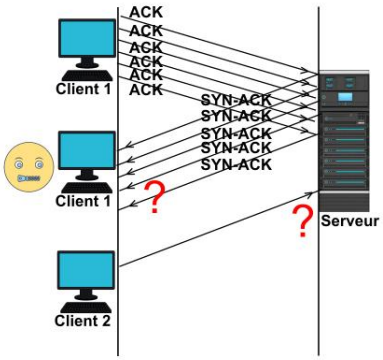
- Le champ Window d'un paquet TCP précise la taille de la fenêtre pour enregistrer les numéros de séquence entrants.
- Quand le système arrive en bout de capacité, il diminue le champ Window pour ralentir l'activité.
- C'est à ce moment là que l'attaquant se connecte et prolonge artificiellement la durée de session.
- La machine distante est surchargée.



# L'exemple malveillant du piratage SYN

- L'inondation par TCP SYN exploite le mécanisme de "handshaking" en 3 phases, pour consommer des ressources chez le destinataire et l'empêcher de fonctionner.




- L'attaquant envoie des paquets SYN à tous les ports actifs identifiés, généralement à partir d'une fausse adresse IP
- Le serveur répond à chaque SYN par un accusé de réception SYN ACK, pour chaque port ouvert
- L'attaquant ne retourne pas le ACK de réponse au SYN ACK
- Mais côté serveur, la session reste ouverte
- Avec un grand nombre de sessions ouvertes, le serveur est bloqué et ne peut plus répondre aux demandes légitimes nouvelles

# TCH Hydra

- L'horreur absolue.
- L'outil s'attaque aux « credentials » d'authentification de plus de 50 protocoles : Cisco, FTP, HTTPS, IMAP, IRC, LDAP, MS-SQL, MYSQL, NNTP, plusieurs protocoles Oracle, POP3, RDP, SAP/R3, SIP, SMTP, SSH, Subversion, Telnet, VMWare-Auth, VNC, etc.
- Il est le seul à proposer une telle diversité.
- Type d'outil d'attaque par force brute.
  - Nécessite des interventions précises de l'utilisateur, pour aider TCH Hydra dans son cheminement pour aboutir (peut-être) à l'obtention d'un mot de passe.
  - Il faudra lui indiquer soit une adresse IP, soit une famille d'IP, le protocole qui nous intéresse, voire un fichier contenant des mots de passe que TCH Hydra pourra tester en priorité, voire limiter le périmètre d'introspection pour des mots de passe ayant une certaine longueur, 8 caractères par exemple, mais pas 6 ou 7, la position des caractères numériques, spéciaux et majuscules, etc.
- L'outil est une console de commandes très puissante.
- Attention, éviter de le mettre entre n'importe quelles mains.



# John the Ripper

- John The Ripper est au hacking ce qu'est Pelé au soccer. La référence absolue.
- A l'origine, un cracker de mots de passe, mais il est devenu une véritable suite, capable de mener des attaques par force brute, grâce à toute la panoplie des outils et techniques, connus à ce jour.
- Pour retrouver les mots de passe, il se sert des artefacts habituels : des dictionnaires (le sien par défaut comporte plus de 3 000 occurences). Il tente toutes les combinaisons possibles en associant l'identifiant et en effectuant toutes sortes de permutations, avec des caractères majuscules, minuscules, numériques et des symboles.
- JTR, est très à l'aise avec les techniques de chiffrement et il est capable, dans certains cas, de déchiffrer une empreinte (hash) MD5, par exemple et intervenir dans un processus de fédération d'identités de type ADFS de Microsoft ou Kerberos. Dans certains contextes et avec de la patience, il peut y parvenir…
- Avantage : sa disponibilité sur une cinquantaine de plates-formes.

# Nmap

- Nmap peut faire tomber les protections d'un réseau et constituer un excellent outil de maintenance, tests de vulnérabilités et simuler des attaques.
- Nmap est un « sniffer » de cibles, qui dresse une carte des ressources réseaux, avec de nombreuses informations sur chacune d'elles.
- Nmap peut être actionné par une interface graphique (il en existe plusieurs), mais aussi en mode commande.

`nmap -option1 -option2 -optionN [cible]`, la cible pouvant être un DNS, une adresse IP, une plage d'adresses IP, une adresse réseau avec son filtre de sous-réseau.

- On peut détecter l'OS qui équipe la cible :

`nmap -A domaine` : OS de la cible

`nmap -sV domaine` : services actifs avec n° de port

- Tests d'intrusion : on pourra associer la commande Nmap à un véritable script, écrit en Lua

`nmap -Pn -script de vulnérabilité domaine`

- C'est la fonctionnalité la plus utilisée via un script publié par la communauté.
- Simulations d'attaques de type DOS (Denial of Service), sur les cibles sensibles à la faiblesse Slowloris, par exemple :

```
nmap 192.168.X.Y -max-parallelism 800 -Pn --script http-slowloris --script-args
http-slowloris.runforever=true
```

- Ex d'attaques par force brute :
  - Sur une cible WordPress

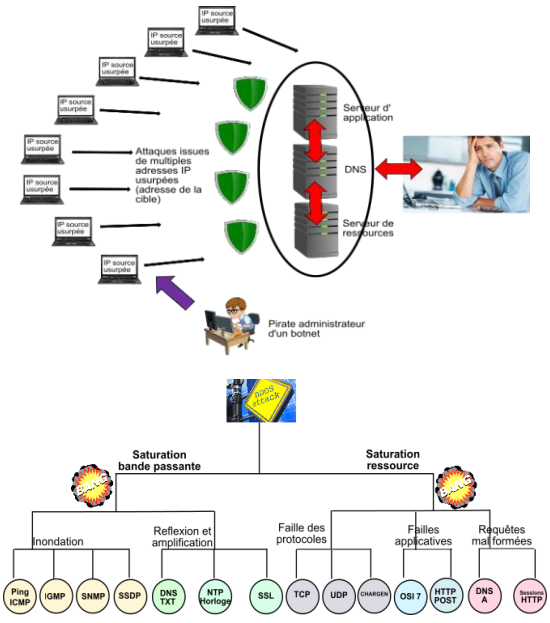
```
nmap -sV --script http-wordpress-brute --script-args
'userdb=users.txt,passdb=passwds.txt,http-wordpress-brute.hostname=domain.com,
http-wordpress-brute.threads=3,brute.firstonly=true' 192.168.X.Y
```

  - ou sur une base de données SQL

```
nmap -p 1433 --script ms-sql-brute --script-args
userdb=customuser.txt,passdb=custompass.txt     192.168.X.Y
```

- Nmap est une sorte de couteau suisse, dangereux d'usage, mais accessible librement

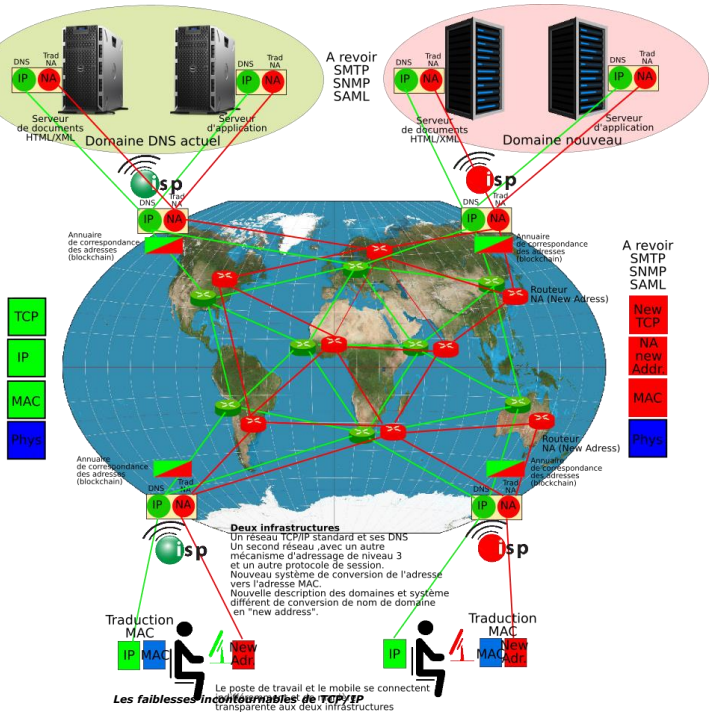



# Le DDoS : on a dépassé le Tbps

- Le « ping de la mort » : protocole ICMP
- Inondation IGMP : Internet Group Management Protocol, protocole qui permet à des routeurs IPv4 d'établir de façon dynamique des groupes de plusieurs hôtes pour qu'ils puissent s'insérer dans des diffusions multipoint.
- Attaque SNMP : Simple Network Management Protocol, permet aux administrateurs de gérer les équipements du réseau, de superviser et de diagnostiquer les problèmes à distance.
- Attaque par réflexion SSDP : Simple Service Discovery Protocol, pour la diffusion et la découverte de services de réseau et d'informations de présence
- Demande de fichiers volumineux.
- Attaques NTP (synchronisation d'horloges).
- Attaques SSL.
- Attaques génériques TCP et UDP.
- Attaques CHARGEN (protocole peu utilisé de test, débogage et mesure), mal conçu.
- Inondation ACK.
- Attaques par saturation applicative.
- Flooding HTTP POST.
- Attaques sur requêtes mal formées.

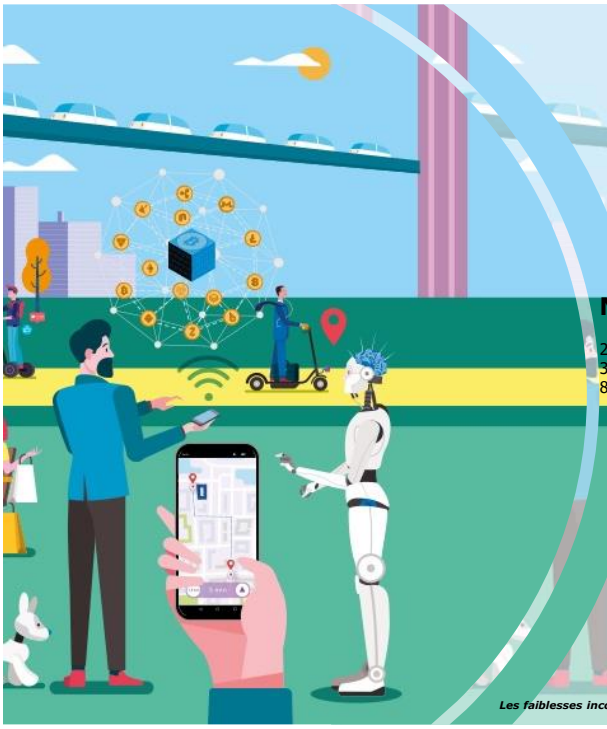
# Très compliqué de modifier l'ordre établi






# Des solutions quand même...

- Compte tenu des faiblesses natives de la suite TCP/IP, la sécurité n'est pas un projet, ni un objectif, c'est un voyage, une contrainte permanente.
- Avoir l'appui de la Direction Générale qui doit se positionner sur les grandes décisions : Cloud, télétravail, évolution du métier…
- Avoir une approche holistique (globale) du périmètre sécuritaire.
- Se méfier des discours rassurants des consultants : les néologismes ne règlent pas tout.
- Ne pas trop se faire trop d'illusions sur notre capacité à nous protéger à 100 %.
- Faire de la paranoïa une qualité : ce n'est pas peut-être, mais quand.
- Ne protéger que ce qui mérite de l'être.
- Isoler les données et ressources sur lesquelles est fondée notre activité (dans la mesure du possible) et leur appliquer des mesures drastiques, voire contraignantes.
- Exploiter les ressources récentes en matière de protections périmétriques : pare-feux, IDPS
- IPv6 plutôt que IPv4.
- Investir dans les technologies de chiffrement et les appliquer clairement dans les domaines clés de l'entreprise : messagerie, données sensibles, etc.
- Mettre en place une segmentation adaptée des réseaux.
- Traiter comme il se doit les problématiques d'authentification.
- Imposer une charte de comportement, mais pas pour orner les murs de bureaux.
- Faire confiance aux fournisseurs… dont c'est le métier.
- Appliquer les recommandations après analyse : patchs, mises à jour systèmes…
- Mettre les bonnes personnes au bon endroit.
- Scepticisme vis-à-vis de certaines contraintes règlementaires : "yakafocon…".
- La sécurité ne consiste pas à protéger uniquement le management : lui-même doit se conformer aux contraintes (moindre privilège…).
- Appliquer les principes du hacking éthique et s'organiser en conséquence.

# Les faiblesses de la suite TCP/IP

## 16 Juin 2023

### Nos prochains webinaires

| | |
|---|---|
| 23 Juin 2023 : | **La fédération d'identités : "you will never walk alone"...** |
| 30 Juin 2023 : | **Les grandes figures du TI... dont on parle moins** |
| 8 Septembre 2023 : | **Les grandes utopies du TI : capitaliser sur nos erreurs** |


claude@lemarson.com
https://www.lemarson.com