M3102 – Services et reseaux

EMAIL : CRISTINA.ONETE@GMAIL.COM

Des vrais réseaux vs. l'émulation

- Nos TPs l'année dernière :
  - Plusieurs machines, des équipements réseau (routeurs, switches, etc.)
  - Une machine virtuelle par machine
  - Les machines étaient connectées dans des sous-reseaux, éventuellement passant sur des passerelles et/ou via la machine prof.
- Quels sont les risques de cette approche ?

Dans ce module on va plutôt simuler des réseaux complèxes

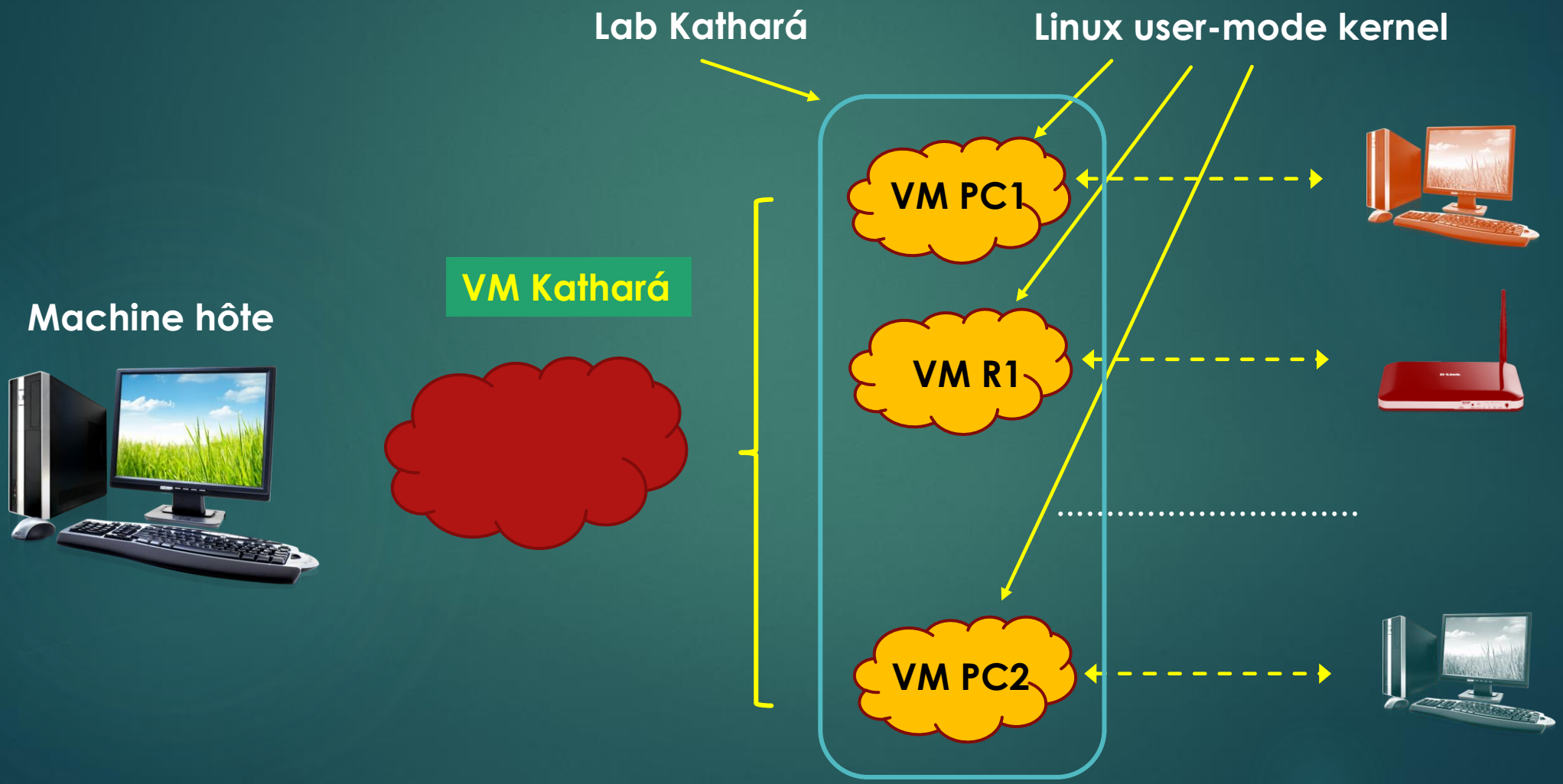
L'outil Kathará

- Permet de simuler le fonctionnement de tout un reseau sur une machine
- Un logiciel qui permet de créer des containeurs sur une **machine hôte**
  - Chaque containeur est une VM Linux
  - Chaque VM joue le rôle d'un élément d'un réseau : un PC, un routeur, un serveur...
  - Toutes ces VMs sont dans un seul VM dans lequel on exécute Kathará
  - On peut manipuler VM par VM, ou alternativement faire toute la simulation fonctionner à partir de Kathará
- Ceci nous permet de simuler le fonctionnement d'un réseau en toute sécurité

Une description de Kathará

LES V-COMMANDES, LES L-COMMANDES, ETC.

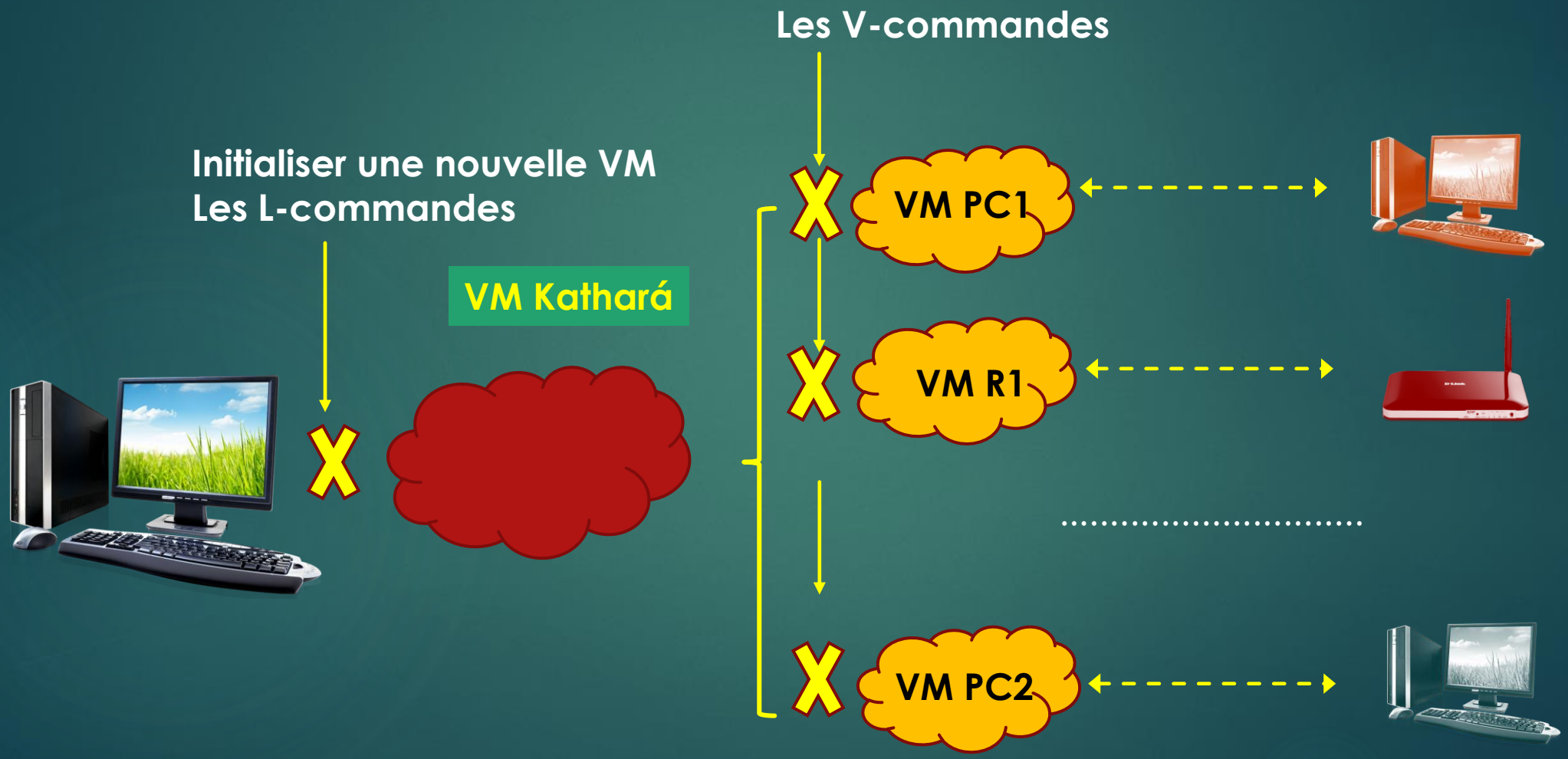
L'infrastructure avec Kathará



Deux types de commandes

- Kathará nous donne la possibilité de "programmer" un réseau
  - Spécifier quels rôles les machines jouent-elles
  - Détailler comment des réseaux sont connectés
  - Simuler la communication entre des machines, des reseaux, ...
- Essentiellement deux types de commandes :
  - Les V-commandes (vcommands) : des commandes pour chaque machine virtuelle (d'où le v)
  - Les L-commandes (lcommands) : des commands pour le lab en entier (d'où le L)

Faire entrer les commandes



Quelques V-commandes

- **kathara vstart** : fait démarrer une nouvelle machine virtuelle
- **kathara vlist** : liste des VMs actuelles
- **kathara vconfig** : permet d'ajouter dynamiquement des Nouvelles interfaces à une VM déjà existente
- **kathara vhalt** : permet d'arrêter une VM sans causer des crashes
- **kathara vcrash** : provoque un crash
- **kathara vclean** : permet de nettoyer tout Netkit et la configuration de la machine hôte
- **kathara wipe** : commande essentielle, nettoye les traces des labs exécutées précédemment

Quelques L-commandes

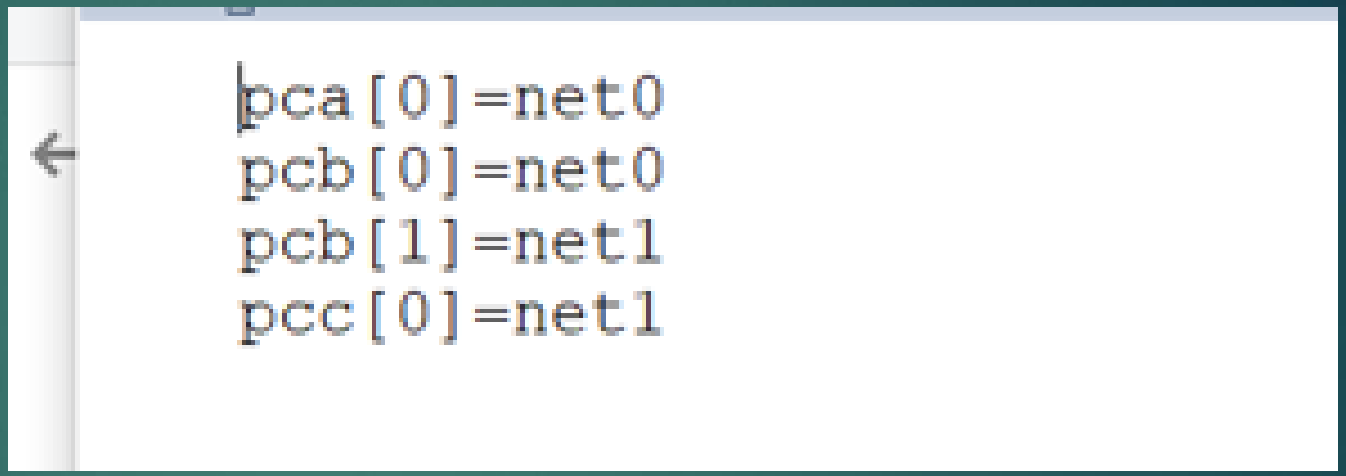
- **kathara lstart** : démarre un labo
- **kathara lhalt** : arrêt
- **kathara lcrash** : provoque un crash sur toutes les VMs du labo
- **kathara lclean** : enlève les fichiers temporaires du repertoire d'un labo
- **kathara linfo** : donne des informations concernant un labo, sans le démarrer
- **kathara ltest** : permet l'exécution de tests, pour savoir si un labo fonctionne correctemment

Les labs Kathará

- Décrivent une certainte topologie,spécifiée dans un fichier lab.conf
  - La topologie spécifie des machines avec des interfaces réseau
  - Les reseaux forment des "domaines de collision" (collision domain)
  - On a la possibilité de donner des specifications pour chaque machine (par exemple combien de RAM on a, etc.)
- Chaque machine virtuelle sera associée à un sous-répertoire
- Des fichiers pour décrire le comportement des machines au start et au halt

La configuration d'un lab : lab.conf

Trois machines :
pca, pcb, pcc



<nom_machine>[<numéro>] : se réfère à l'interface <numéro> de la machine <nom_machine>

net0, net1 : deux domaines de collision distincts, domain0, domain1

mem : pca[mem] = 64 veut dire la machine pca aura 64 Megabyte de mémoire

Dessinez la topologie indiquée par ce script

Configurer les interfaces des VMs

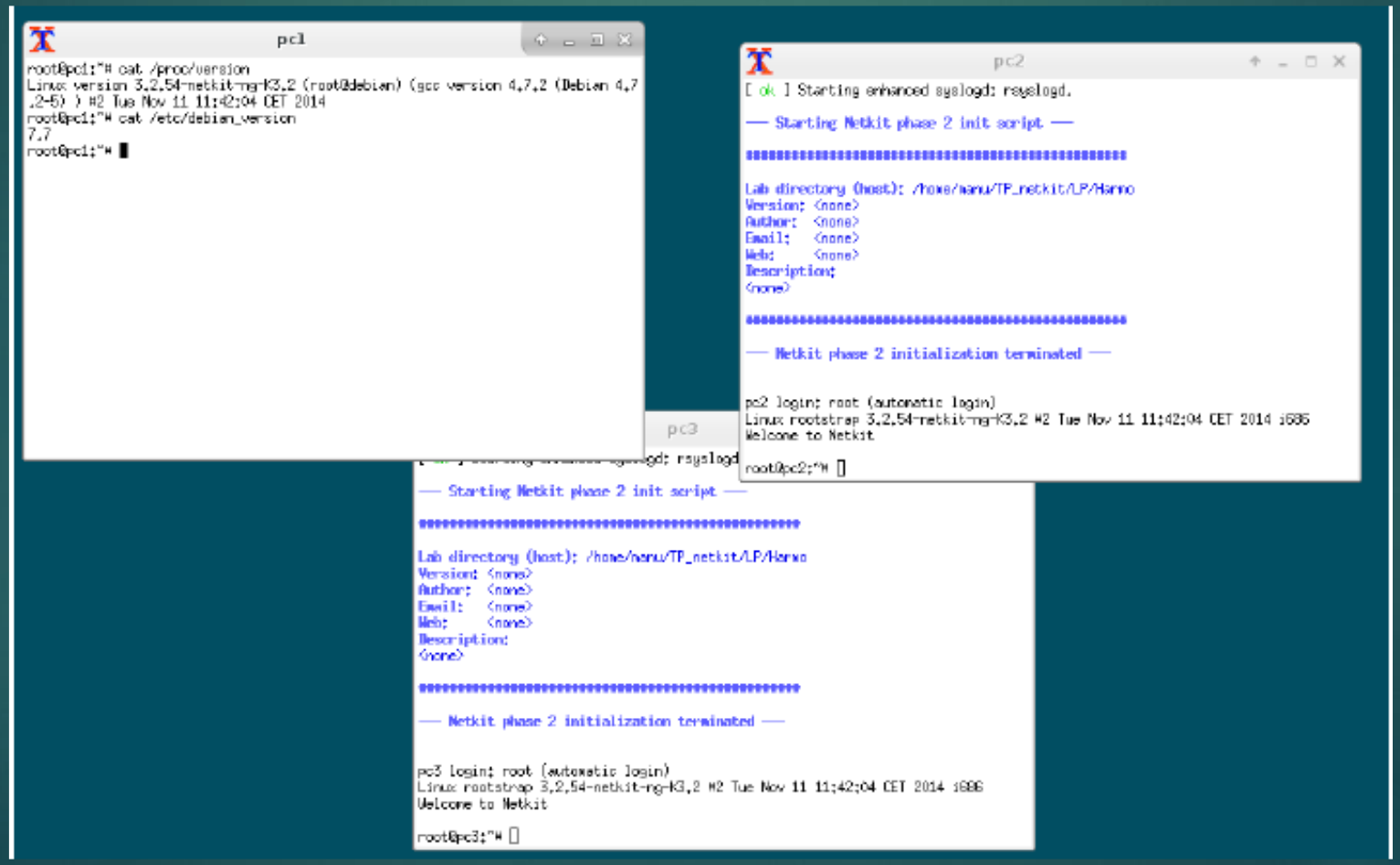
- Se fait en Kathara via les scripts de startup et de shutdown
- Deux possibilités :
  - Chaque VM peut avoir un fichier <nom machine>.startup
    - Une instruction typique dans un tel fichier pourait être

      ip address add 192.168.1.2/24 dev eth1

      ip link set dev eth1 up
  - Les configurations qui affectent toutes les VMs sont dans deux fichiers

    shared.startup et shared.shutdown
- Au démarrage, la VM exécute shared.startup, puis <nom de la VM>.startup
- Le fichier de shutdown est utilisé au cas des opérations de reboot ou de halt
  - On ne l'utilise pas dans le cas d'un crash

Les trois VMs initialisées



La commande vstart

- Elle peut être exécutée dans un script ou comme ligne de commande

  kathara **`vstart <options> <nom_machine>`**

- Quelques options :

  -n <valeur> : nom de la machine

  -mem <valeur> : associe la valeur de Mo indiquée pour la VM

  --eth<numéro> : <nom_domaine> : associe l'interface eth<numéro> (par ex. eth0 ou eth1) avec le domaine de collision

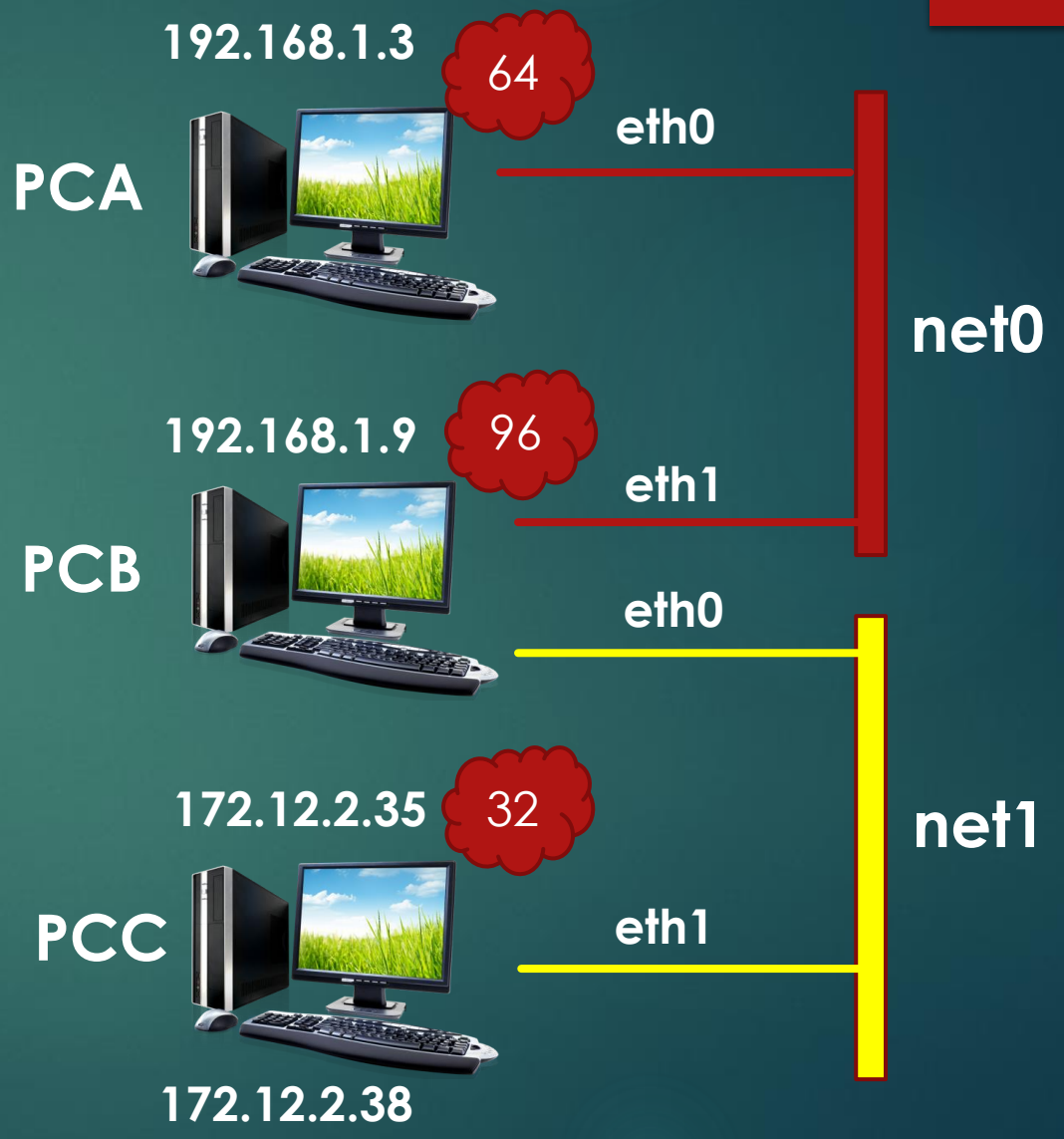
  --bridged : function avancée, permet d'obtenir accès à l'Internet via l'interface ethN

- Exemple :

```
kathara vstart -n pca -mem 64 --eth0:net0 --eth1:net1
```

Exercices

- Ecrivez un fichier lab.conf qui décrit la configuration à droite
- Quels seraient les contenus des fichiers PCB.startup et PCB.shutdown ?



Des fonctions avancées

L'INTERFACE TAP ET L'ACCES INTERNET

Une connexion Internet

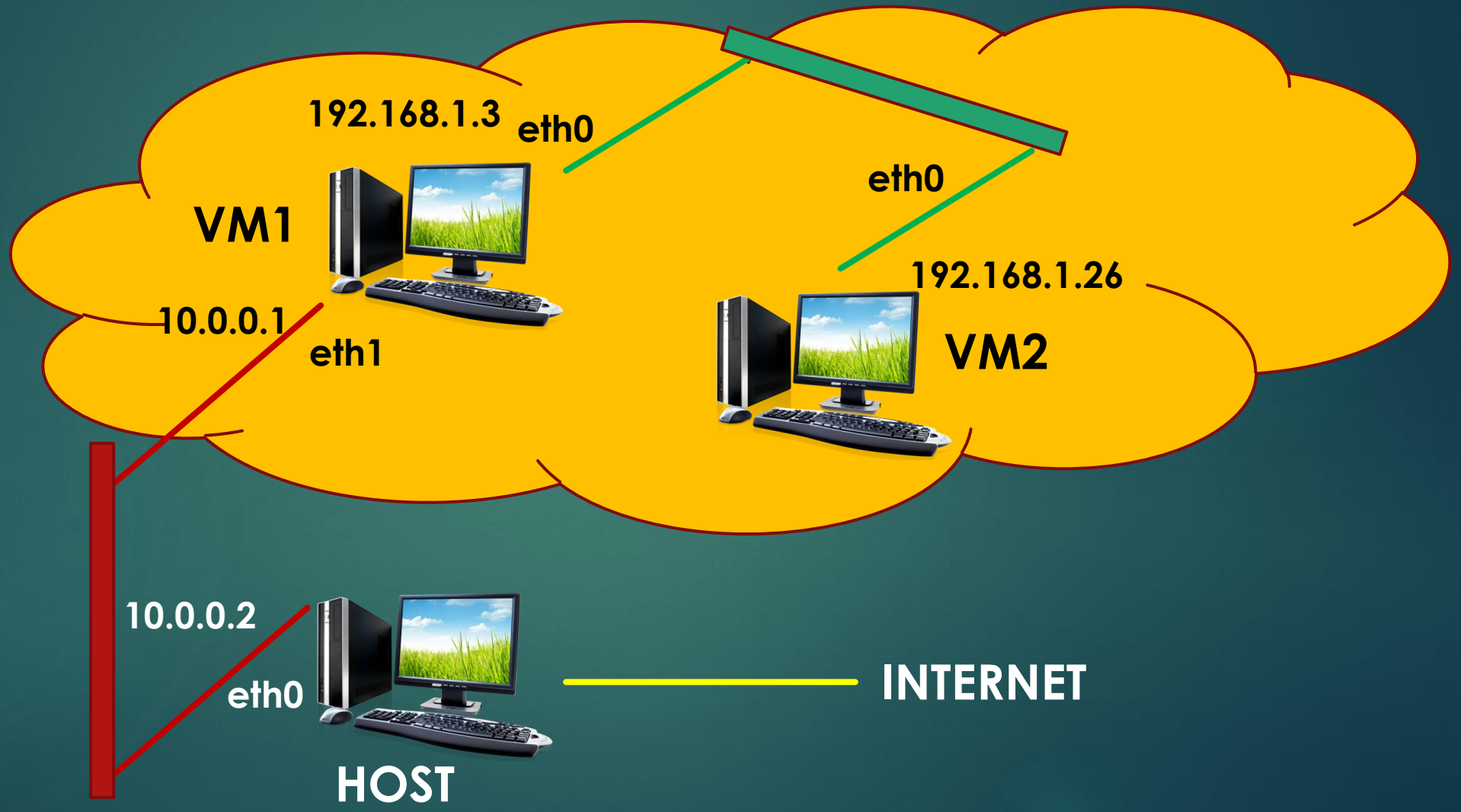
- Le logiciel Kathará sert à simuler des configurations de réseau avant de les mettre en pratique
  - Par exemple pour anticiper des crashes, des difficultés concernant le routage…
- Parfois, même pour une simulation il nous faut une connexion Internet

Pourquoi ?

Un pont Internet

- Réalise une connexion entre une VM et l'Internet en utilisant la machine hôte
  - Si la VM est connectée à d'autres machines, la VM et la machine hôte devront jouer le rôle d'un routeur
- Le pont crée une interface "artificielle" sur la machine hôte et sur la VM
  - ... qui seront dans un même domaine de collision

Un exemple


192.168.1.3
eth0
VM1
10.0.0.1
eth1
eth0
192.168.1.26
VM2
10.0.0.2
eth0
HOST
INTERNET

Configurer un pont

- En ligne de commande :

  vstart <autres options> --bridged

  - Par exemple :

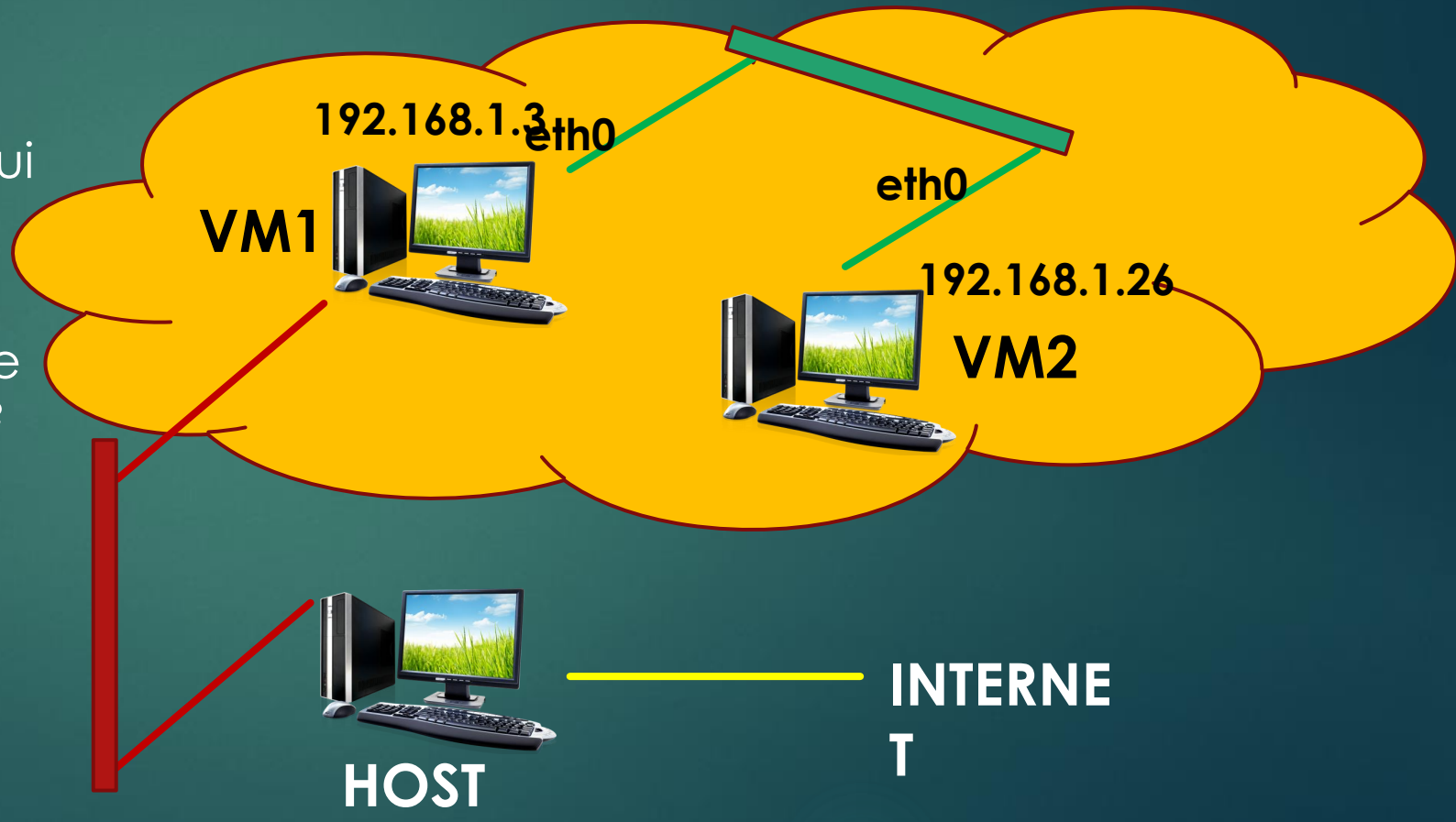
    vstart -n pca --eth1:net0 --eth1:net1 --bridged

- En lab.conf :

  <nom VM>[bridged] = true

Exercices

- Ecrivez le fichier lab.conf qui correspond à cette figure
- Quelles instructions doit-on fournir pour que la machine VM2 ait accès à l'Internet ?
- Où va-t-on écrire ces instructions ?



Rappel sur DNS

(POUR LE PREMIER TD)

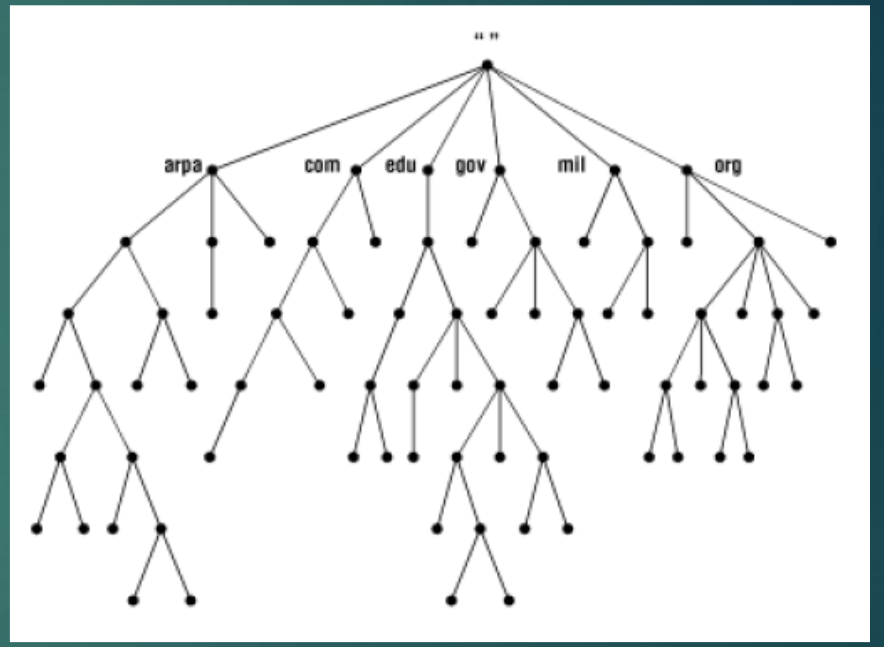
Le but du protocole DNS

- Puisque les humains n'aiment pas devoir retenir des centaines d'adresses IP...
  - On peut utiliser des noms de domaine plutôt que des adresses IP
- Correspondance adresse IP -- nom de domaine : DNS (ou fichier hosts)
  - Résolution directe
  - Résolution inverse
- Le protocole DNS s'exécute entre un client et un serveur
  - Le serveur est toujours un serveur, mais un client peut également être un serveur DNS

Les noms de domaine

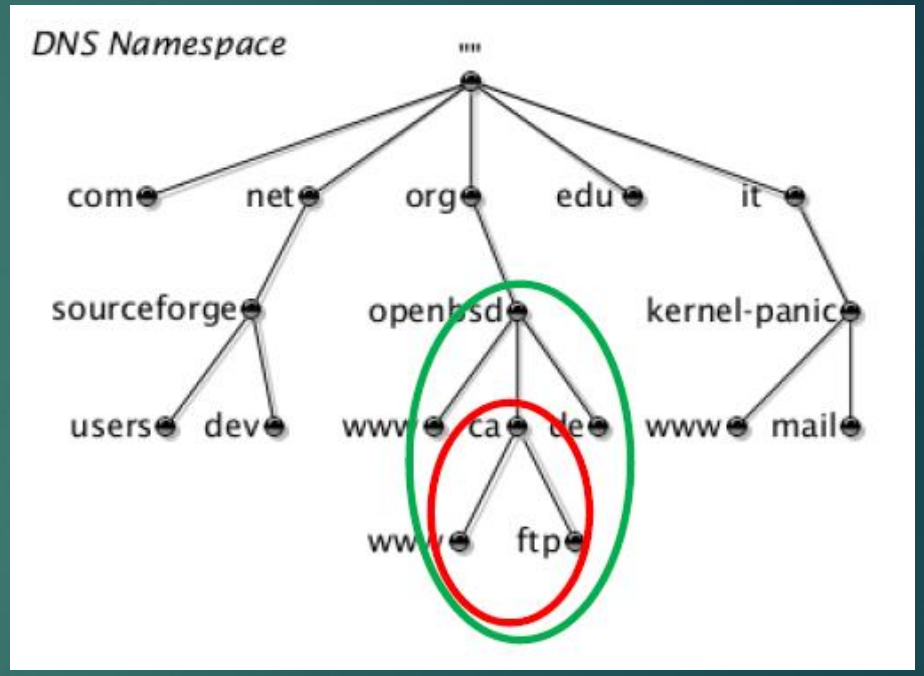
- Namespace : l'espace de nommage
- Arbre de noms avec racine commune
  - jusqu'à 127 niveaux (max 253 char ASCII)
  - Chaque noeud à un nom...
  - ... sauf la racine: "."

- Le nom d'un domaine se reconstitue en ordre inverse, du nom du noeud au plus en bas vers celui le plus en haut (finissant par .)



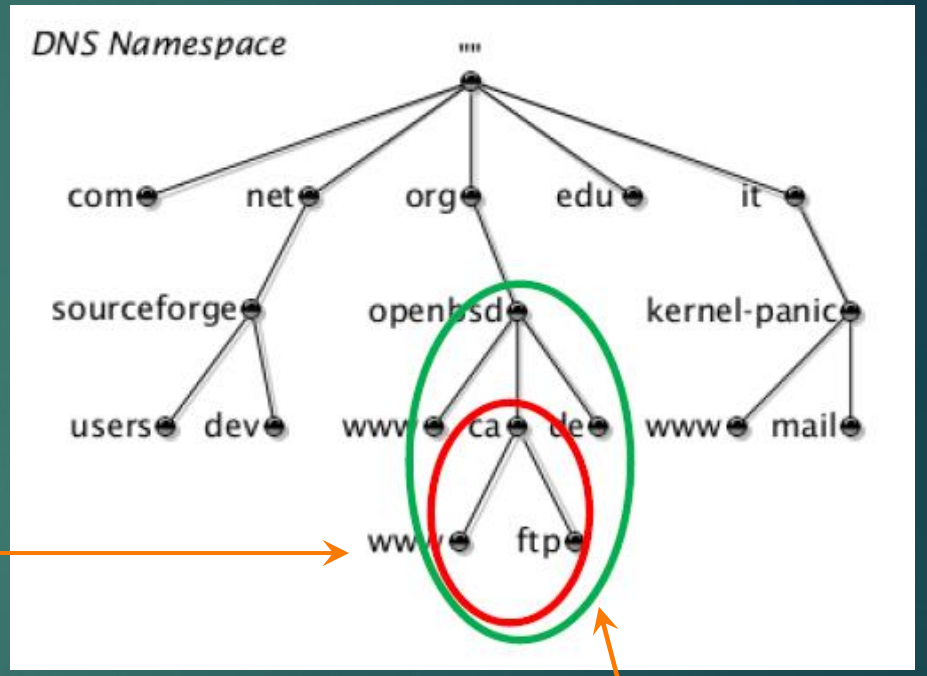
Exemple

- Domaine en rouge : ca.openbsd.org.
  - Avec deux machines :
    - www.ca.openbsd.org
    - ftp.ca.openbsd.org
- Domaine en vert : openbsd.org.
  - plusieurs sous-domaines



Exemple

- Domaine en rouge : ca.openbsd.org.
  - Avec deux machines :
    - www.ca.openbsd.org
    - ftp.ca.openbsd.org
- Les pings sur une machine dans un domaine s'interprètent premièrement "localement"

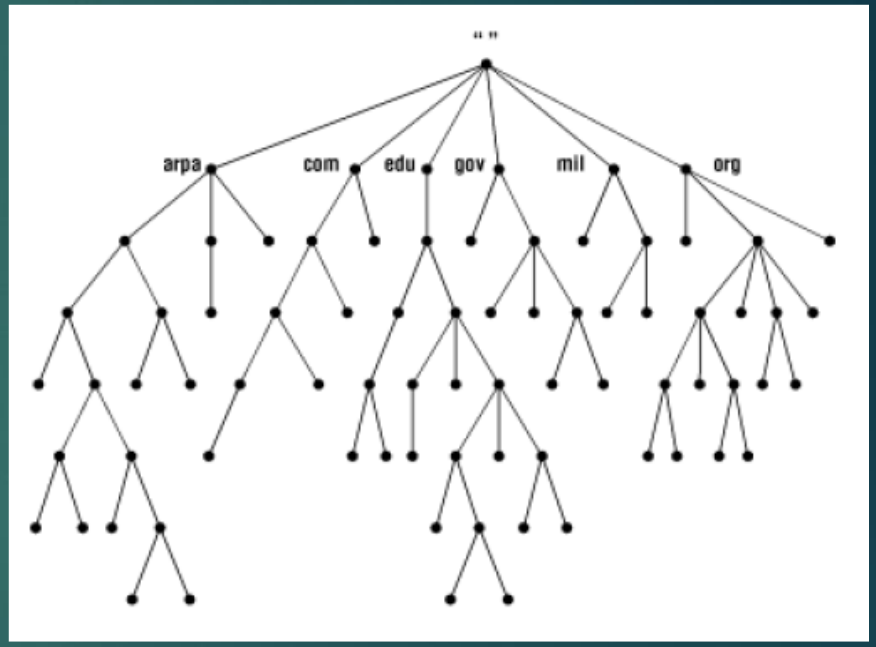


ping www sur machine ftp.ca.openbsd.org

Organisation des noms

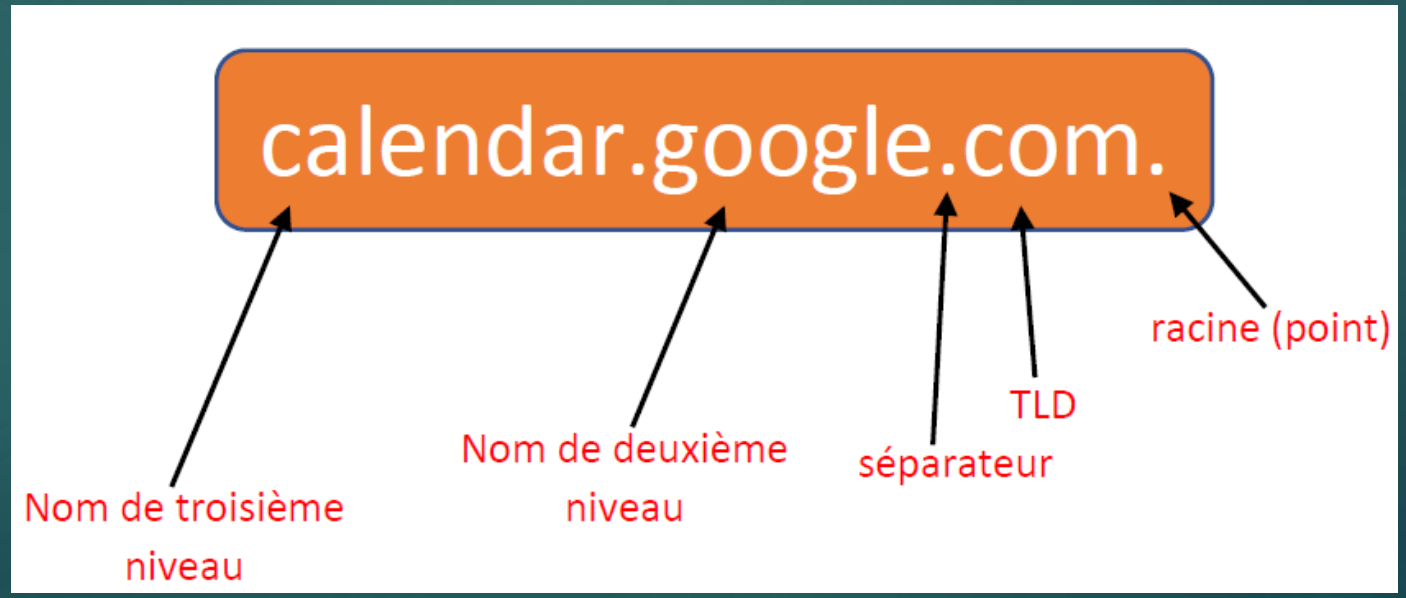
- TLD : Top-Level Domain
  - Niveau le plus haut du nom

- Plusieurs catégories :
  - .arpa : réservé, administratif
  - .ca, .be, .fr, etc. : country-code TLD (ccTLD)
  - génériques de 1er niveau : generic TLD : .com, .mil
  - génériques restreint : .biz, .name, ,pro
  - sponsorisés (sTLD) : .post



Nom rélatif, nom absolu

- Fully qualified domain name (FQDN) :
  - Nom complet d'une machine, finissant toujours par un point : .
  - La partie la plus significative est à droite, au contraire des adresses IP



La résolution DNS

- Résolution directe : parcourrir l'arbre par ordre inversée de la hiérarchie de son FQDN
- Résolution inverse : domaine spéciale : in-addr.arpa
  - Contient un arbre inversé : le noeud in-addr a 256 noeuds fils
  - Chaque noeud fils a encore 256 noeuds fils...
  - ... etc. (sur 4 niveaux)