INSA | INSTITUT NATIONAL DES SCIENCES APPLIQUÉES CENTRE VAL DE LOIRE

Proxy applicatifs et proxy socks

Département STI

Promo 2022 année 4

Module : Administration de réseaux V1

Proxies ou Firewalls (pare-feux) :

- Le terme « proxy » → « mandataire » :
  the authority to represent someone else, especially in voting.(Google translate)
- Les proxies sont parfois qualifiés de « firewall applicatif »

Deux types de Firewall :

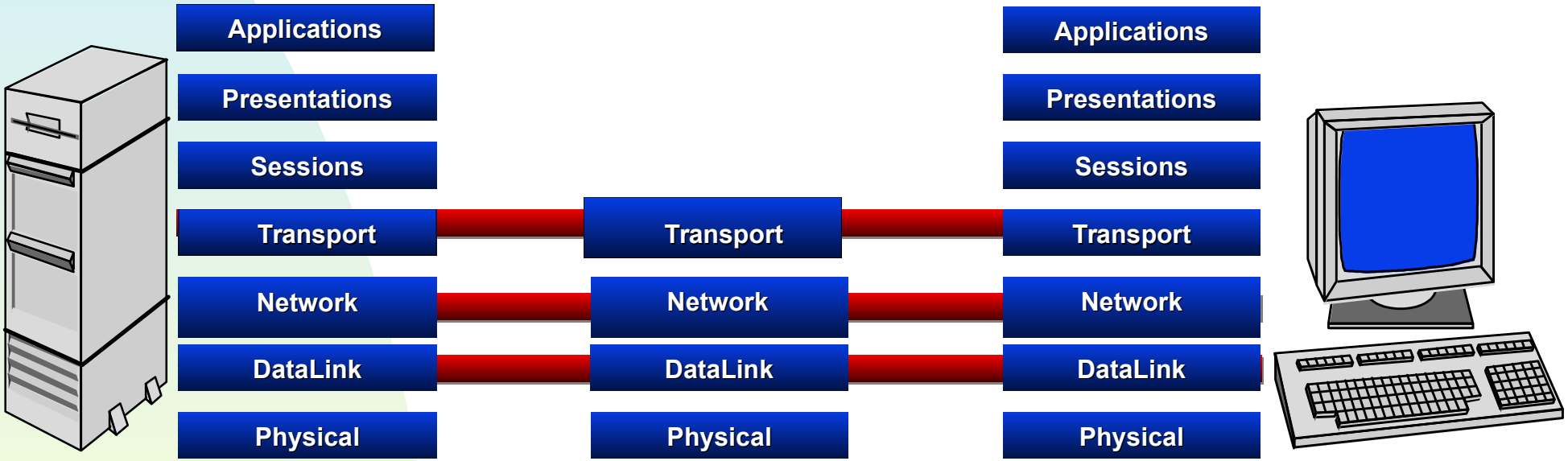
- Les pare-feux IP ou filtrants qui ont pour objectif de bloquer tout le trafic sauf celui sélectionné (Voir cours « iptables »)

- Les serveurs mandataires (PROXIES) qui ont pour objectif de réaliser, en particulier, les connexions réseaux pour vous.

  Les clients ne se connectent pas directement à l'extérieur, mais par l'intermédiaire du serveur mandataire.

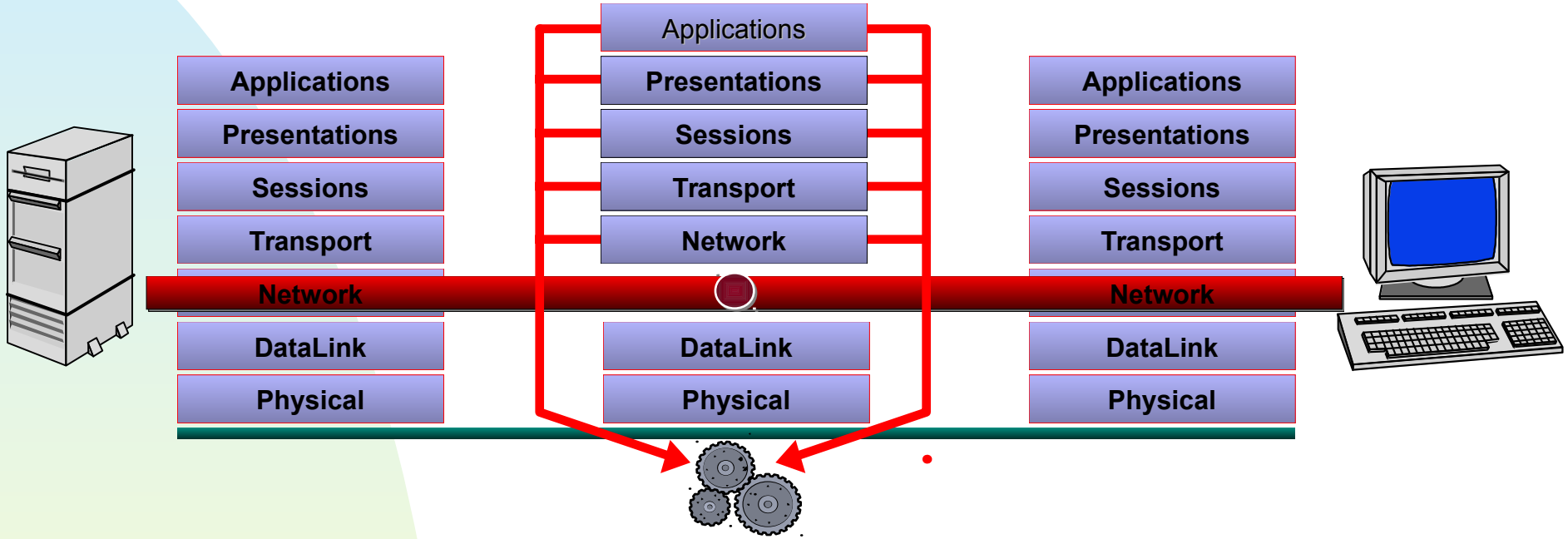
Les deux types de firewall sont complémentaires.

Firewalls réseau filtrant



Firewall réseau

Firewalls proxy



Proxy principes

- Relais entre un client et un serveur

  `Client-->Proxy-->Serveur`

- Deux niveaux
  - Niveau application (proxy)
  - Niveau transport et réseau (proxy socks)
    SOCKS signifie Socket Secure

Proxy niveau application

- Gère des applications spécifiques (client web,ftp). Principalement utilisé pour le WEB.
- Limite les connexions disponibles.
- Cache l'identité réelle des deux éléments connectés.
- Un processus par connexion (gourmand en ressources).
- Peuvent faire office de caches de données (HTTP cache de pages Web)

Proxy niveau application

- Plus approprié pour le mode connecté.
- Ne concerne pas les protocoles de couche inférieure à la couche transport.

- Supporte le principe (tout ce qui n'est pas explicitement autorisé est interdit)
- Si de nouvelles applications apparaissent il faut faire évoluer le proxy applicatif

Proxy niveau application

- Les postes clients doivent être configurés pour utiliser le proxy.
- Possibilité de proxy « transparent »

Proxy niveau application

Avantages:

- Monitoring réseau : Centralisation des accès INTERNET
- Authentification de l'utilisateur et/ou de la machine
  - Possibilité d'exiger l'authentification de l'utilisateur, exemples

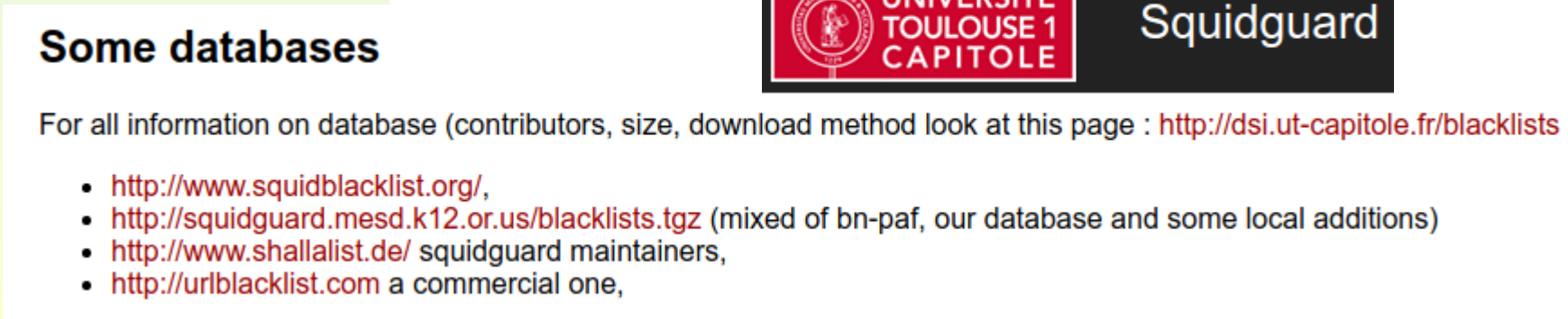
1. LDAP : se base sur le protocole Linux Lightweight Directory Access
2. NCSA : utilise un fichier de noms d'utilisateur et de mots de passe de style NCSA
3. SMB : utilise un serveur SMB comme SAMBA ou Windows NT
4. MSNT : utilise l'authentification de domaine Windows NT
5. PAM : utilise les modules Linux "Pluggable Authentication Modules"
6. getpwam : utilise le fichier passwd de Linux.
7. A partir de Squid-2.6 gestion de Kerberos (Actuellement Squid 4.13 année 2020)

- Translation d'adresse
- Cache : optimisation de la navigation mais problème pour les pages trop dynamiques.
- Log : Enregistrer les accès des utilisateurs aux sites.

Proxy niveau application

Avantages:

- Bloquer certains accès
  - Par adresses IP ou plages d'adresses,
  - Par noms d'utilisateurs,
  - Par procédure d'authentification.
- Bloquer des URL de sites en particulier (Blacklist).
  Exemple blacklits téléchargeables : https://dsi.ut-capitole.fr/documentations/cache/squidguard_en.html#contrib



- ...

Proxy niveau application

- Avantages :
- Le proxy visualise l'ensemble des données transmises (avantage ou inconvénient ?).
- Les log sont complets puisque toutes les données applicatives sont analysées par le proxy.
- On peut filtrer suivant les URL, ou simplement l'apparition de mots désignés comme interdits.
- Pour les mandataires FTP, on peut scanner la présence ou non de virus dans le fichier téléchargé.

Proxy niveau application

- Une distribution avec un serveur proxy préinstallé (mais pas seulement) :

http://smeserver.fr/index.php

Proxy niveau transport (socks)

- Ces proxies gèrent plus de services que le proxy applicatif.
- Contrôle moins la charge utile du paquet.
- Certains protocoles en mode connecté sont difficiles à gérer (Ftp mode passif)
- Les clients doivent être conscients qu'ils utilisent un circuit au niveau du proxy

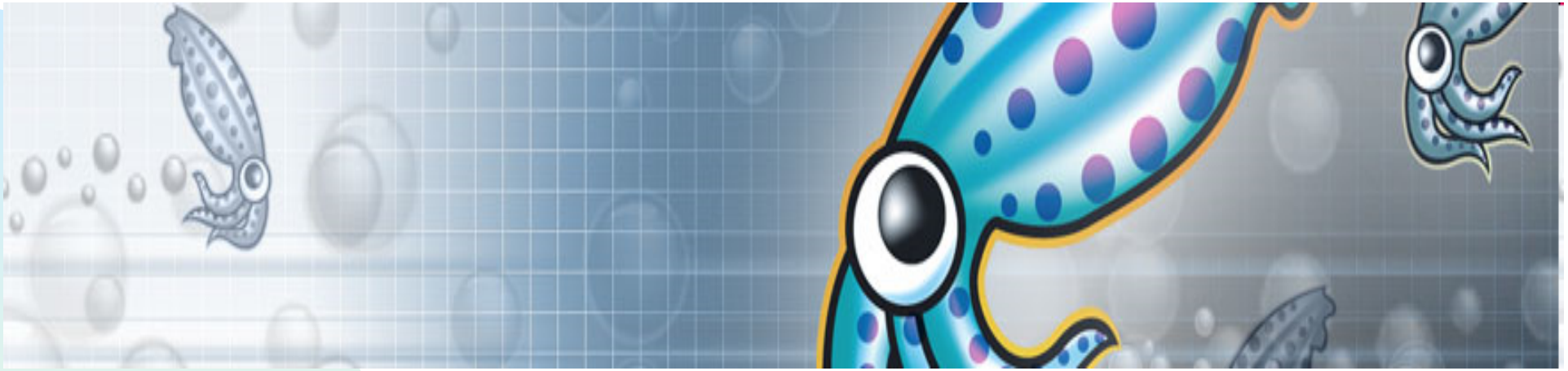
Proxy niveau transport (socks)

- Supporte avant tout le mode TCP
- SOCKS v5 supporte le mode UDP

Proxy niveau transport (socks)

- Avantages :
  - Vérifier l'adresse IP du client et du serveur
  - Autoriser une connexion sur un port pour une durée maximale fixée
  - N'autoriser la réutilisation d'un même port qu'après un certain délai
  - Enregistrer la destination de la connexion de chaque utilisateur
  - Enregistrer l’utilisateur qui a demandé la connexion.

Présentation d'un serveur proxy



- SQUID
  - Squid est le serveur proxy référence sur les OS Linux

http://www.squid-cache.org/

Présentation d'un serveur proxy

- SQUID
  - Squid est le serveur proxy référence sur les OS Linux

  - Squid peut-être avantageusement complété
    - **Cachemgr** : livré avec squid, il s'agit d'un script qui permet de visualiser en temps réel l'état et les performances de SQUID.
    - **SquidGuard** : permet le filtrage à partir de listes noires thématiques (sites pornographiques, warez, etc.) ou d'expressions.

    - **Prostat** : permet de traiter les logs et de générer une page de statistiques
    - **MRTG** : permet de récupérer des données via SNMP afin d'évaluer les performances du cache.

SQUID et configuration

- SQUID et le port
  - Le fichier de configuration de squid : /etc/squid.conf
  - Partie de config du démon squid :

    # TAG: http_port
    # Usage: port
    # hostname:port
    # 1.2.3.4:port

    Le port conventionnellement admis pour un serveur proxy est le : 8080

SQUID et configuration

SQUID et le cache

```
#  TAG: cache_dir
#  Usage:
#  cache_dir Type Directory-Name Fs-specific-data [options]
#  You can specify multiple cache_dir lines to spread the
#  cache among different disk partitions.
# [...]
cache_dir ufs /var/spool/squid 100 16 256
```

SQUID et configuration

SQUID et les ACL
TAG: http_access
Allowing or Denying access based on defined access lists
Access to the HTTP port:
http_access allow|deny [!]aclname ...
#Default:
http_access deny all
#http_access allow our_networksacl

sti src 192.168.20.0/24   <== machines sti
acl Users proxy_auth REQUIRED <== Authentification requise.
http_access allow localhost <== autorisation machine locale.

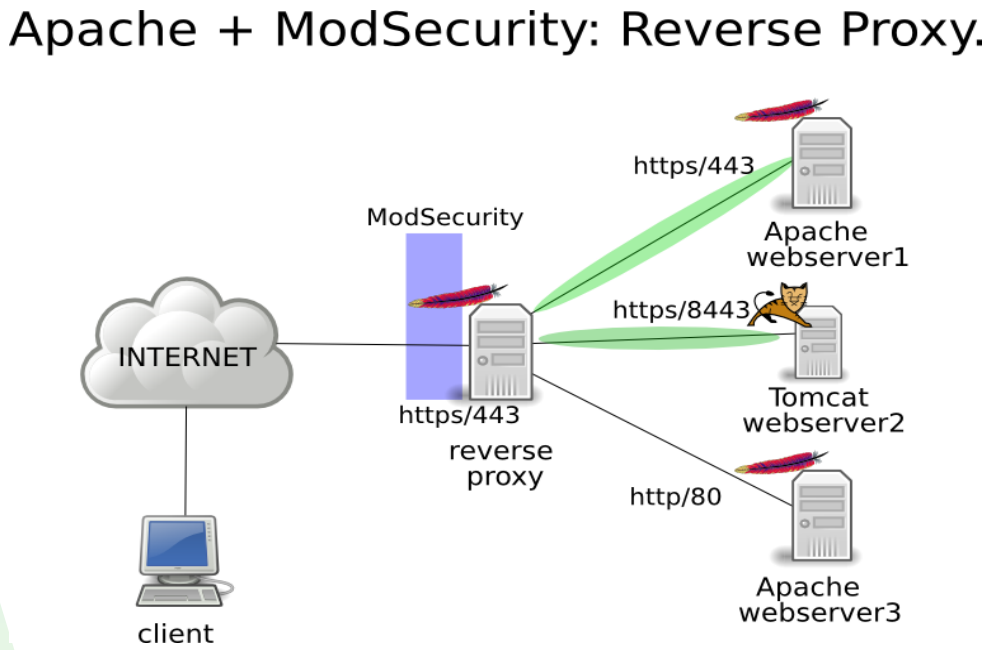
SQUID et configuration

SQUID et l'authentification
Paramètre auth_param :

```
#  TAG: auth_param
#   This is used to define parameters for the various authentication
#   schemes supported by Squid.
## format: auth_param scheme parameter [setting]
auth_param basic program /usr/lib/squid/ncsa_auth  /etc/squid/users
```

Squid fait appel à des programmes externes pour l'authentification.
Ici un exemple avec le programme ncsa_auth, les login et mots de passe autorisés sont dans /etc/squid/users
Le programme htpasswd permet de gérer ce fichier.

Proxy Inverse



Le proxy inverse est installé du côté serveurs Internet. L'utilisateur du Web passe par son intermédiaire pour accéder aux applications de serveurs internes. Le proxy inverse (reverse proxy) est parfois appelé substitut (surrogate).

Proxy Inverse

- Mémoire cache : le proxy inverse peut décharger les serveurs Web pages/objets statiques « accélérateur web » ou d'« accélérateur HTTP ».
- Intermédiaire de sécurité : le proxy inverse protège un serveur Web des attaques provenant de l'extérieur.
- La ré-écriture programmable d'URL permet de masquer et l'architecture d'un site web interne
- Chiffrement SSL : le proxy inverse peut être utilisé en tant que « terminateur SSL »
- Répartition de charge : le proxy inverse peut distribuer la charge d'un site unique sur plusieurs serveurs Web applicatifs.
- Compression : le proxy inverse peut optimiser la compression du contenu des sites.
- Virtual hosts : il peut rediriger les flux à partir de l'URL.

Proxy Inverse