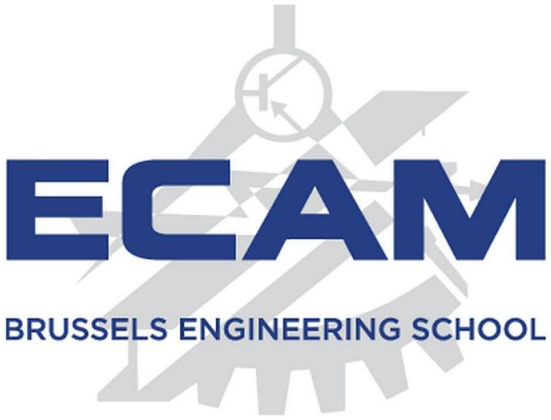


ECOLE CENTRALE DES ARTS ET MÉTIERS

RAPPORT DU PROJET DE SIMULATION RÉSEAU 4IN

Déploiement d'un réseau d'entreprise avec Cisco Packet Tracer

Wéry Benoît
14256

Enseignant : M G. Demaude

20 mai 2019

Table des matières

Chapitre 1

Introduction

L'objectif de ce travail consiste à simuler un réseau d'entreprise avec l'outil *Cisco Packet Tracer* en respectant certaines contraintes afin de se rapprocher de la réalité d'implémentation.

Les consignes à respecter sont les suivantes :

- Un réseau privé ECAM avec des sous-réseaux pour séparer les étudiants et l'administration
- (Optionnel) Implémentation des sous-réseaux au moyen de VLANs
- Implémentation d'un service DHCP pour l'attribution des adresses IP privées du réseau ECAM
- Implémentation d'une translation NAT pour communiquer avec le réseau public
- Un firewall pour contrôler le traffic vers le réseau ECAM
- Utilisation du protocole OSPF pour le réseau public reliant les deux sites privés
- Un serveur web accessible sur le réseau public depuis une adresse statique et depuis son url via un serveur DNS
- Un réseau ICHEC capable d'atteindre un serveur sur le réseau ECAM
- (Optionnel) un tunnel VPN pour relier les réseaux ECAM et ICHEC de façon sécurisée
- (Optionnel) une DMZ pour améliorer la gestion des contrôles d'accès au réseau ECAM

Tout d'abord, je commencerai en décrivant la topologie de mon réseau avec les différents choix d'implémentations effectués.

Ensuite, une série de tests via l'outil de simulation mettront en avant le bon comportement du réseau.

Finalement, une brève conclusion reviendra sur les objectifs atteints du travail et mes acquis.

Des exemples de configurations sont donnés en annexes à titre informatif pour illustrer certains points théoriques abordés.

Remarques

- Après plusieurs vérifications, les adresses IP dans le réseau ECAM peuvent prendre du temps à être distribuées par le service DHCP après le lancement du fichier *evaluation.pka*.
- Dans un souci de confort d'utilisation, et étant donné le cadre pédagogique de ce projet, j'ai choisi de ne pas appliquer les configurations "de sécurité de base" (à savoir, mdp pour configurer routeur/switch,...) pour ne pas devoir y encoder à répétitions les mdps. Dans un cas pratique, il est évident que cette mesure doit être prise pour éviter qu'un intru ne puisse modifier les paramètres des machines.

Chapitre 2

Implémentation du réseau

Cette section reprend les détails théoriques et les choix d'implémentation de la topologie globale ci-dessous.

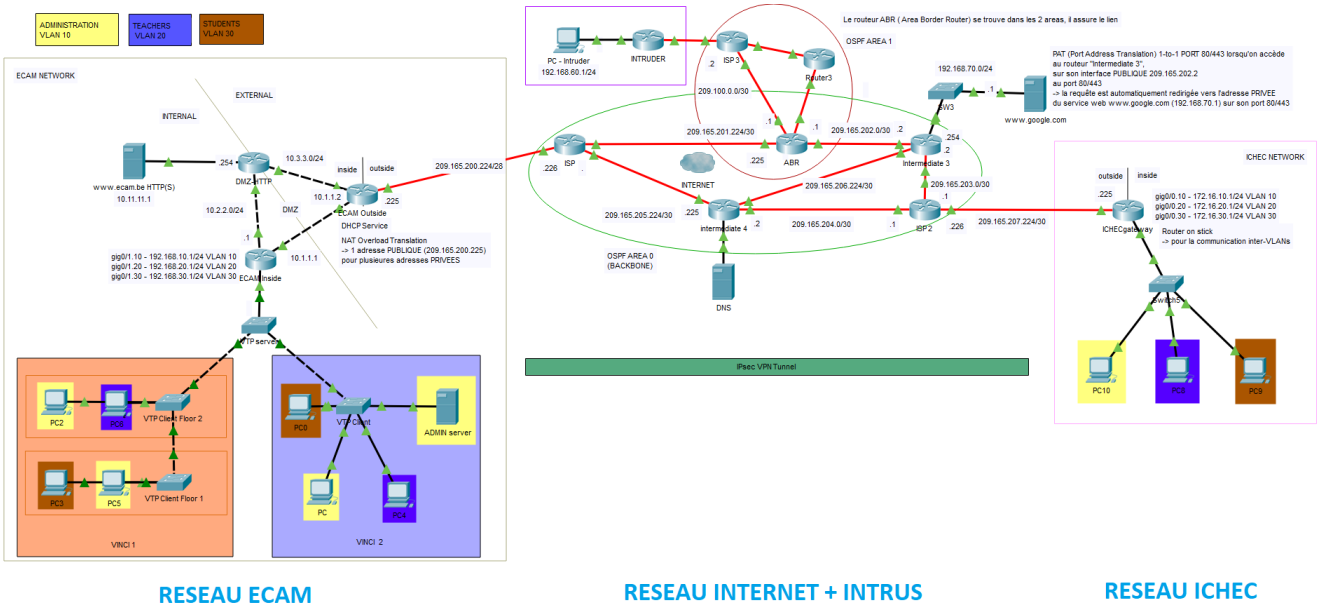


FIGURE 2.1 – Topologie globale du projet - les différentes sections sont détaillées ci-dessous

2.1 Réseau ECAM

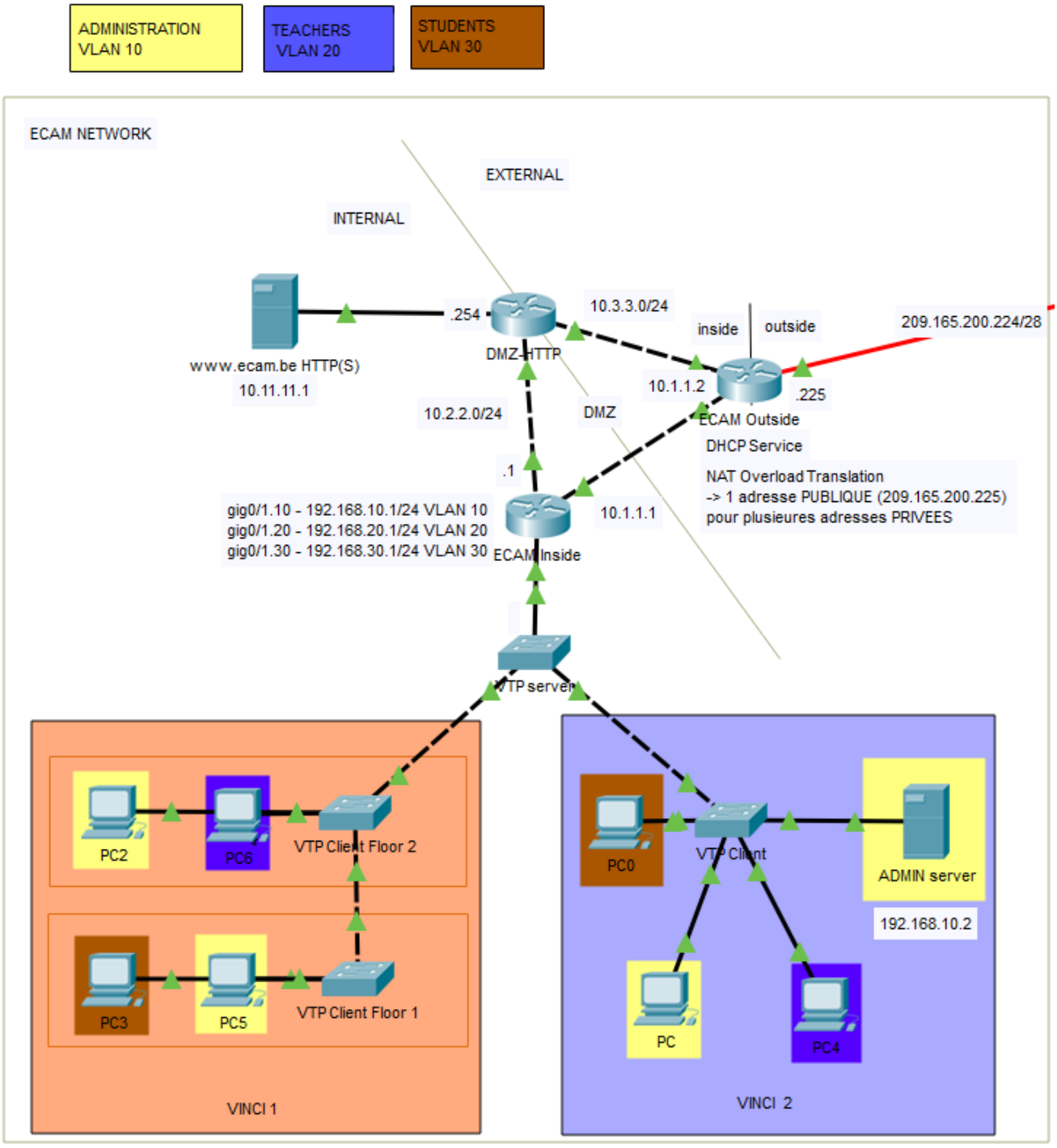


FIGURE 2.2 – Réseau ECAM

2.1.1 VLANs

Afin de scinder les différentes entités actives au sein du réseau, j'ai choisi de créer plusieurs réseaux virtuels (VLANs = *Virtual Local Area Networks*) [1] :

vlan 10 : Administration | vlan 20 : Teachers | vlan 30 : Students

1. Configuration : A.1

Ceci offre plusieurs avantages, dont le fait de s'affranchir de la contrainte d'emplacement physique d'une machine par rapport à son réseau. Comme on peut le voir sur la figure 2.3.(A), en l'absence de réseaux virtuels, les machines d'un même sous-réseau doivent se trouver à proximité les unes des autres et par rapport à l'interface du routeur. A contrario, en utilisant des VLANs, des machines peuvent faire partie d'un même sous-réseau tout en étant physiquement distantes comme sur la figure 2.3.(B)

De plus, le même réseau peut être obtenu en utilisant moins de matériel et moins d'interfaces réseaux, comme le montre la figure 2.3.(C)

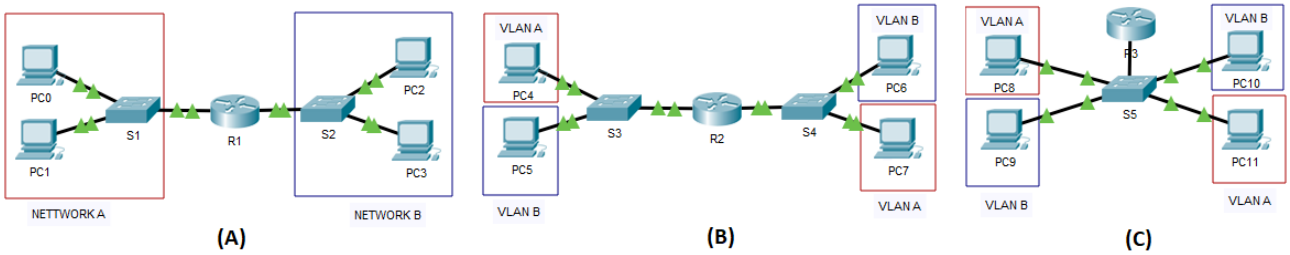


FIGURE 2.3 – Avantages des VLANs

J'ai cherché à mettre en avant ces avantages dans le projet, en distingant les bâtiments VINCI 1 et VINCI 2 de l'ECAM (fig 2.2). On observe alors que les machines PC0 et PC3 se trouvent chacune dans un des batîments mais font bien partie du même réseau virtuel *Students*.

Les VLANs permettent également de facilement former des groupes (sous-réseaux) et d'attribuer des règles d'accès spécifiques aux membres (voir ACL, 2.1.2). A titre d'exemple, l'administration (VLAN 10 - jaune) peut communiquer avec tout le monde, tandis que les étudiants (VLAN 30 - brun) ne peuvent pas accéder aux VLANs 10 et 20.

Dans ce cas-ci, on peut constater que le serveur "ADMIN" qui se trouve dans le VLAN 20, n'est donc accessible que par les membres du même VLAN.

Protocol VTP

Pour éviter de devoir configurer chaque switch individuellement, j'ai choisi d'implémenter le protocole VTP (*VLAN Trunking Protocol*). Ce dernier permet de faciliter la gestion des VLANs à travers plusieurs switchs d'un même réseau.

Le switch SERVEUR est celui sur lequel on configure les différents VLANs. Cette configuration est ensuite propagée vers les autres switchs CLIENTS, ce qui évite donc de devoir configurer manuellement des changements sur chacun. De plus, lorsqu'on ajoute un switch au sein du réseau, on peut alors très facilement hériter des paramètres VLANs existants.

Communication Inter-VLANs

Par défaut, les VLANs ne sont pas capables de communiquer entre eux puisqu'il s'agit de réseaux distincts (couche 3 du modèle OSI) implémentés sur des switchs (couche 2). Ceci représente un avantage certain dans le cas où l'on souhaite avoir des réseaux isolés.

Dans notre cas, afin que les VLANs puissent communiquer entre eux, il faut donc ajouter la gestion de la couche 3, ce qui a été fait grâce à une configuration dite ***"Router On Stick"***, implémentée au niveau du routeur ECAM INSIDE.

Cette configuration offre l'avantage d'utiliser seulement une interface sur le routeur (ex : gi0/1), en la divisant virtuellement en plusieurs sous-interfaces (gi0/1.10, gi0/1.20 et gi0/1.30). Chaque sous-interface est ensuite associée à une adresse IP qui correspond à la passerelle par défaut du VLAN

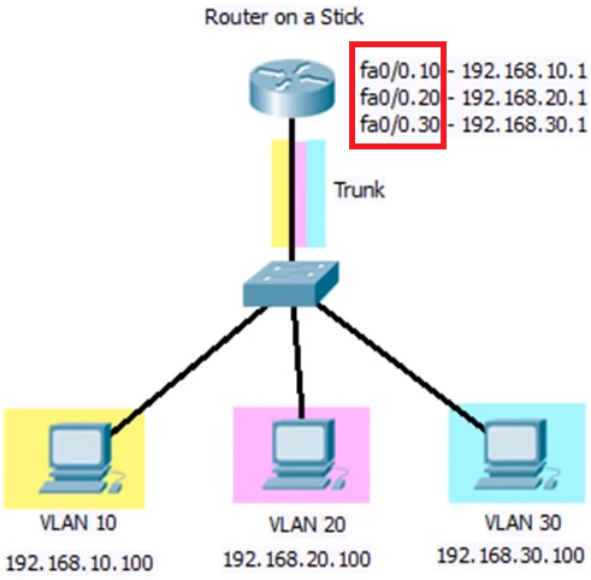


FIGURE 2.4 – Exemple simplifié de la communication inter-VLANs *Router On Stick*

Remarques :

1. Les ports des interfaces qui relient les switchs entre eux, ainsi que la liaison entre le routeur et le switch serveur doivent être en mode ***trunk***. Ceci permet de connecter les interfaces à plusieurs VLANs en même temps, contrairement à un port en mode *access* qui ne peut être associé qu'à un seul VLAN à la fois.
2. Le protocole VTP permet uniquement de propager les informations de configuration des VLANs, par exemple : création, suppression, modification,...
   Il faut sur chaque switch associer ses interfaces aux VLANs désirés.

2.1.2 DMZ (De-Militarized Zone)

Afin d'assurer la sécurité du réseau par rapport à l'environnement extérieur, j'ai implémenté une DMZ. Cette configuration permet à des utilisateurs externes d'accéder aux services partagés du réseau d'entreprise (serveur web, serveur email,...) tout en empêchant une intrusion à l'intérieur du réseau interne à proprement parler.

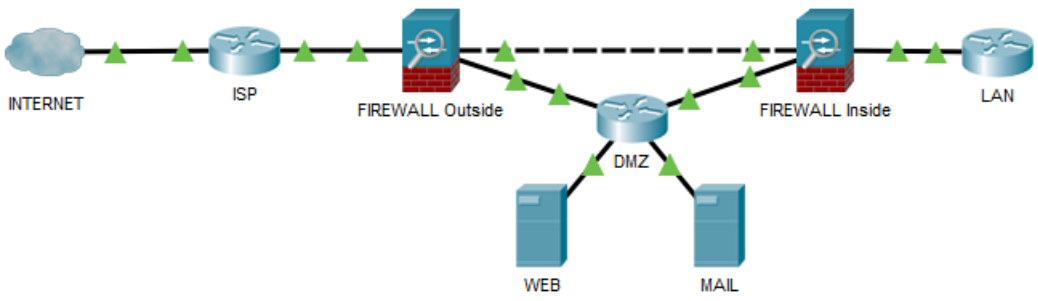


FIGURE 2.5 – Exemple simplifié d'une DMZ

Dans le projet, la DMZ est composée de deux routeurs (ECAM OUTSIDE et ECAM INSIDE) auxquels sont associés les *firewalls* avec des règles différentes.

ACL (Access Control List)

Ces règles ou droits accès, sont assurés au moyen des *Access Control Lists*. En mode *Extended*, celles-ci permettent de définir très précisemment les permissions de passages au travers des interfaces, en spécifiant des règles sur les adresses sources/destinations, les protocoles, les ports,...

Ainsi, pour la DMZ, le routeur qui donne sur le monde extérieur autorise la plupart des connexions entrantes. Tandis que le routeur connecté à l'intérieur du réseau filtre les requêtes en autorisant seulement celles provenant des machines internes à le traverser [2].

2.1.3 Service DHCP

Le service DHCP (*Dynamic Host Configuration Protocol*) est assuré par le routeur ECAM OUTSIDE. Il permet d'associer de façon dynamique les adresses IP aux différentes machines du réseau.

Pour le configurer, on crée des *pools* (ex : ip dhcp pool VLAN10) que l'on associe aux différents sous-réseaux respectifs afin de définir les plages d'adresses à distribuer [3].

Dans l'exemple suivant, le PC2 (faisant partie du VLAN 10) fait une requête d'adresse IP au routeur :

```
C:\>ipconfig /release

   IP Address......................: 0.0.0.0
   Subnet Mask.....................: 0.0.0.0
   Default Gateway.................: 0.0.0.0
   DNS Server......................: 0.0.0.0

C:\>ipconfig /renew

   IP Address......................: 192.168.10.8
   Subnet Mask.....................: 255.255.255.0
   Default Gateway.................: 192.168.10.1
   DNS Server......................: 8.8.8.8
```

Il est également possible de définir des adresses qui ne seront pas distribuées pour pouvoir les utiliser de façon statique par exemple. Dans mon cas, j'ai exclu les adresses allant de 192.168.10.1 à 192.168.10.5 pour le vlan 10. Ceci m'a alors permis d'attribuer une adresse statique au serveur de l'administration pour qu'il soit joignable à tout moment.

Remarque : les requêtes DHCP des machines sont redirigées par le routeur ECAM INSIDE via "ip helper-address"

2. Configuration : A.2
3. Configuration : A.3

2.1.4 NAT (Network Adress Translation)

La NAT permet de convertir une adresse privée en une adresse publique pour accéder au réseau Internet. Elle est assurée par le point de jonction entre le réseau privé et le réseau public, c'est-à-dire le routeur ECAM OUSIDE.

Dans ce projet, j'ai choisi d'utiliser une *NAT statique* pour le serveur web et une *NAT Overload Translation* pour les PCs [4].

NAT statique et PAT

La NAT statique permet d'associer de façon permanente une adresse privée du réseau d'entreprise à une adresse publique. Un service plus intéressant dans notre cas est le PAT (*Port Address Translation)* qui permet d'associer une adresse IP publique ET un port d'accès à une adresse privée.
Ainsi, par exemple, lorsqu'une machine extérieure effectue une requête HTTP vers le port 80 (ou 443 pour une connexion sécurisée HTTPS) à l'adresse IP 209.165.200.225, celle-ci est automatiquement redirigée vers le serveur web "www.ecam.be".

NAT Overload Translation

Il s'agit d'une NAT **dynamique** qui permet d'associer une seule et même adresse IP publique (209.165.200.225) à toutes les machines du réseau interne. Pour identifier les hôtes, chacun est alors associé à un numéro de port au moment de la translation.

4. Configuration : A.4

2.2 Réseau public

Cette section décrit l'ensemble du réseau public implémenté. On constatera que pour tous les réseaux point-à-point entre les routeurs "Internet", les masques de sous-réseaux 255.255.255.252 (/30) ont été choisis afin d'optimiser l'utilisation des adresses publiques.
En effet, un tel réseau connecte uniquement les interfaces des deux routeurs impliqués. Dès lors, il est intéressant de choisir la plus petite plage d'adresses possible afin de ne pas gaspiller d'adresses. La plage la plus petite étant de 4 adresses :

1 ad. réseau | 1 ad. broadcast | 2 ad. machines

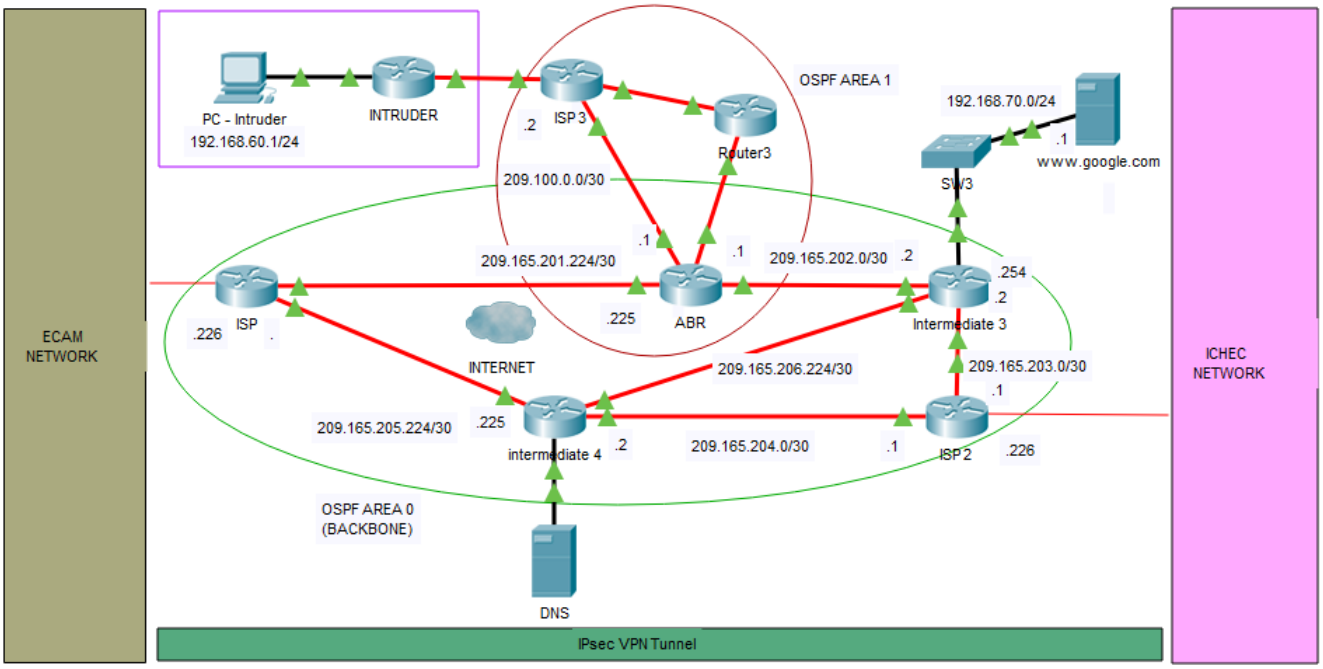


FIGURE 2.6 – Topologie du réseau publique (Internet + Intrus)

2.2.1 Configuration de l'OSPF

Le protocole de routage OSPF (*Open Shortest Path First*) est utilisé afin de définir le meilleur chemin à emprunter pour joindre deux réseaux. Il s'agit d'un *Link State Routing Protocol* qui va permettre de construire de façon dynamique la topologie du réseau (utile pour calculer un nouveau chemin lorsqu'un routeur est inacessible).

Pour ce faire, les routeurs communiquent entre eux des informations sur les réseaux auxquels ils sont directement connectés. Ces données sont regroupées dans un message LSA (*Link State Advertisement*) pour chacune des interfaces actives d'un routeur et qu'il partage régulièrement aux autres routeurs pour mettre à jour les chemins existants.
Ainsi, ce contenu d'information permet à chaque routeur de se créer une image des interconnexions du réseau (= LSDB, *Link State DataBase*). Une fonction de coût permet alors de calculer le chemin à emprunter par les paquets entre la source et la destination afin de minimiser ce coût (sur base de différents critères possibles).

Ci-dessous, on peut voir (en partie) que le routeur ISP a bien connaissance de tous les sous-réseaux qu'il peut joindre et les routes pour y parvenir [5]

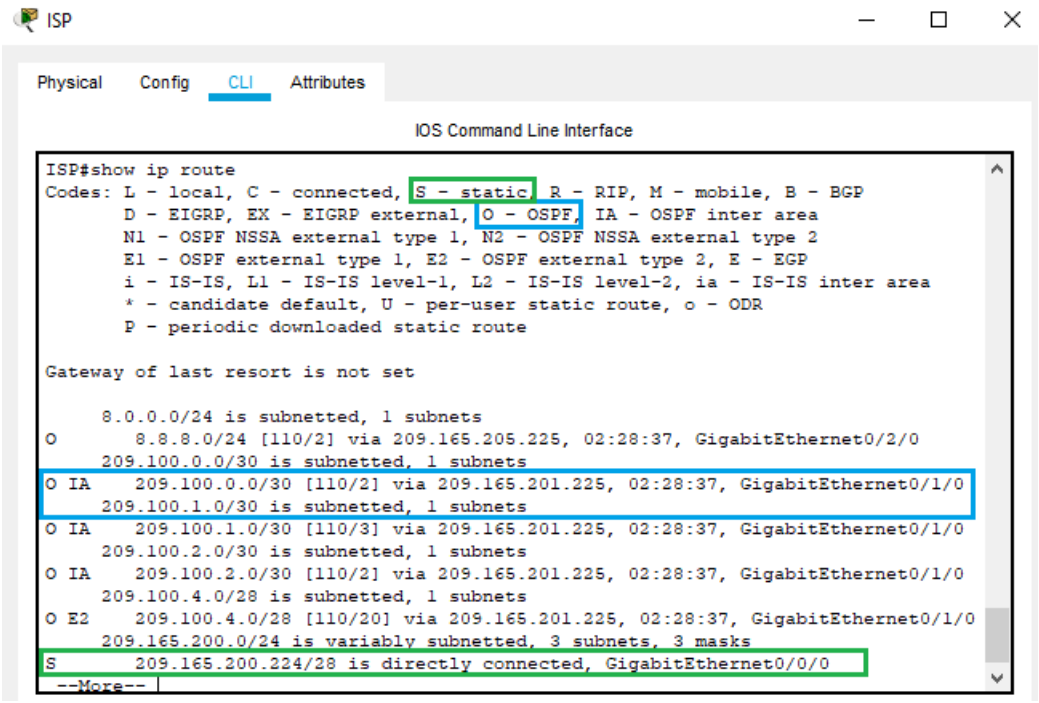


FIGURE 2.7 – Exemple du protocole OSPF

Areas

Dans le cas d'un réseau avec de nombreux routeurs, on divise celui-ci en *areas* pour simplifier les calculs du protocole OSPF. Il existe toujours une ***Backbone Area*** (par défaut l'area 0) qui doit être connectée à toutes les autres *areas*.

2.2.2 Service DNS

Le service DNS (*Domain Name system*) est assuré par un serveur indépendant sur le réseau, à l'adresse 8.8.8.8. Ce service permet de traduire une adresse URL (facile à retenir pour l'humain) en l'adresse IP qui y est associée.

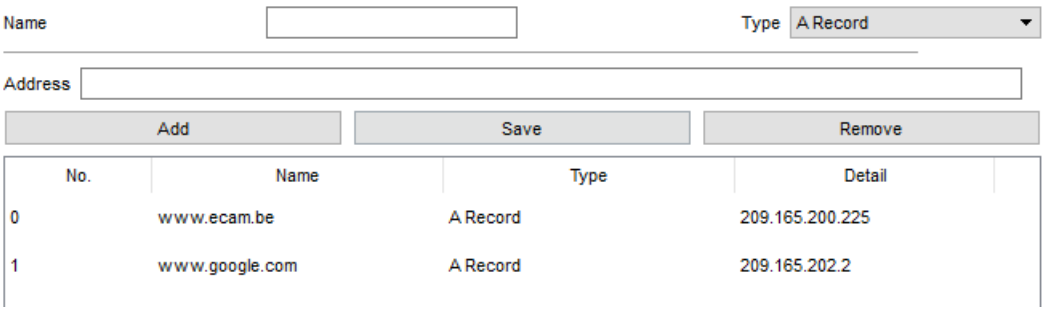


FIGURE 2.8 – Exemple de configuration du DNS

5. Configuration : A.5

Les machines (PCs) des réseaux privés connaissent l'adresse d'un serveur DNS (distribuée par le service DHCP dans notre cas). Ainsi, lorsque le PC2 veut joindre la page web www.google.com, sa requête est d'abord adressée au serveur DNS (8.8.8.8) qui lui fournit l'adresse IP 209.165.202.2 (port 80/443).

2.2.3 Serveur web

Le serveur web www.google.com (192.168.70.1) est joignable via l'adresse IP publique 209.165.202.2 grâce à une translation NAT statique comme expliquée au paragraphe 2.1.4

2.2.4 Tunnel VPN

Un Tunnel VPN permet d'établir une connexion sécurisée entre les deux sites LANs ECAM et ICHEC. Cette connexion qui est faite sur le réseau Internet doit paraître transparente pour les machines de ces deux réseaux.

Dans ce projet, j'ai choisi de configurer le VPN de façon à ce que seuls les VLANs "Administration" de chaque site puissent communiquer directement [6].

2.2.5 Réseau Intrus

Le réseau intrus permet de tester que les accès soit bien restreints au niveau du réseau ECAM. Le PC intru devrait avoir accès au serveur web du réseau ECAM mais ne doit pas être capable de communiquer avec les machines internes à ce même réseau.

6. Configuration : A.6

2.3 Réseau ICHEC

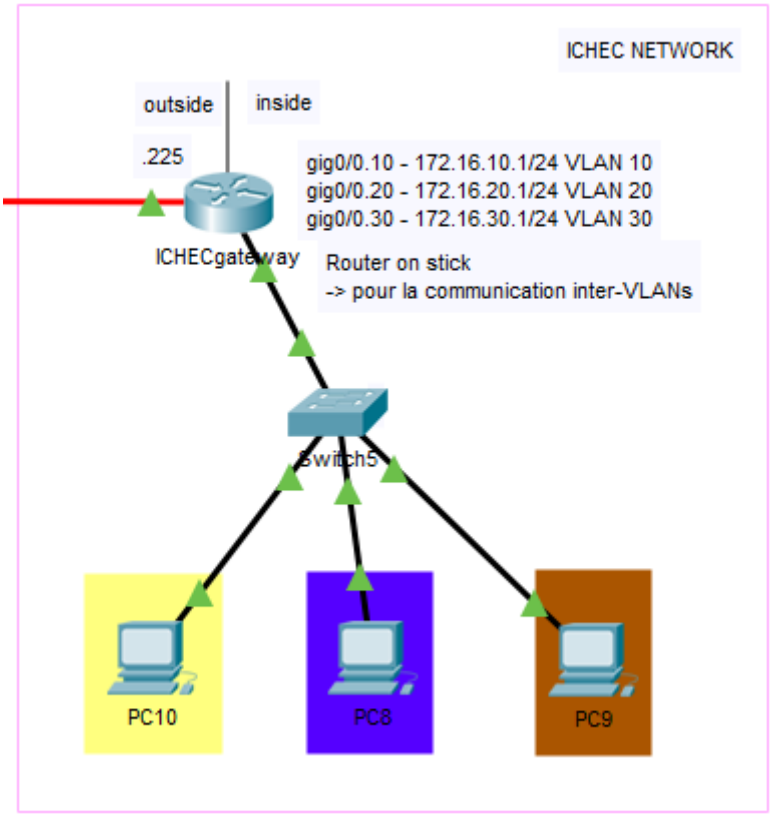


FIGURE 2.9 – Topologie du réseau ICHEC

Pour le réseau ICHEC, je me suis limité à un minimum fonctionnel pour effectuer les différents tests. On retrouve donc :

- une communication inter-VLANs montée en *Router On Stick*
- un service DHCP
- une NAT Overload Translation
- (pas de firewall)

Chapitre 3

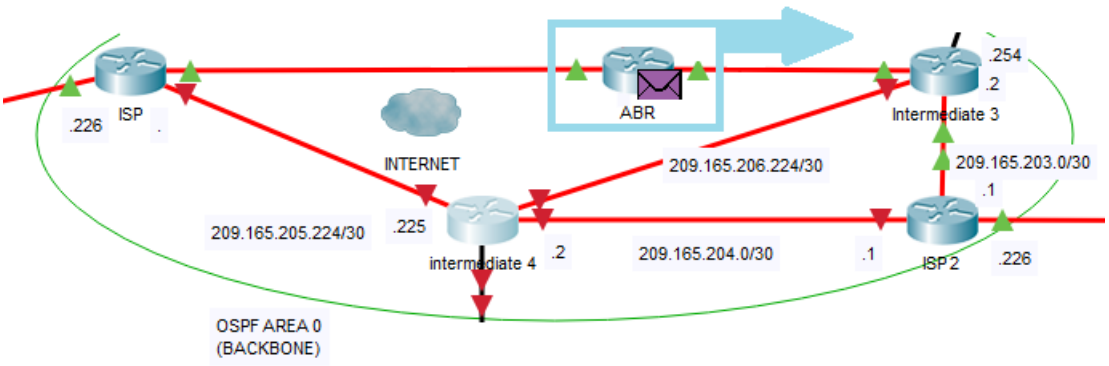
Simulation

Ce chapitre permet de mettre en évidence le comportement du réseau à travers différents tests spécifiques.

3.1 OSPF

Dans le test suivant, j'effectue un ping vers le PC10 avec la configuration initiale et on s'aperçoit que le paquet passe par le routeur *Intermediate 4*. Après cela, j'éteins ce routeur pour vérifier que le paquet sait néanmoins emprunter un autre chemin pour atteindre sa destination.

Après un certains temps, on constate que le protocole OSPF a effectué ses transmissions de paquets LSA, ce qui a permis de mettre à jour la LSB du routeur ISP qui a donc bien redirigé le paquet ICMP vers le routeur ABR.



Intermediate 4 ON

Time(sec)	Last Device	At Device	Type
0.000	--	PC	ICMP
0.001	PC	VTP Client	ICMP
0.002	VTP Client	VTP server	ICMP
0.003	VTP server	ECAM Inside	ICMP
0.004	ECAM Inside	ECAM Outside	ICMP
0.005	ECAM Outside	ISP	ICMP
0.006	ISP	intermediate 4	ICMP
0.007	intermediate 4	ISP 2	ICMP
0.008	ISP 2	ICHECgateway	ICMP
0.009	ICHECgateway	Switch5	ICMP
0.010	Switch5	PC10	ICMP

Intermediate 4 OFF

Time(sec)	Last Device	At Device	Type
0.000	--	PC	ICMP
0.001	PC	VTP Client	ICMP
0.002	VTP Client	VTP server	ICMP
0.003	VTP server	ECAM Inside	ICMP
0.004	ECAM Inside	ECAM Outside	ICMP
0.005	ECAM Outside	ISP	ICMP
0.006	ISP	ABR	ICMP
0.007	ABR	Intermediate 3	ICMP
0.008	Intermediate 3	ISP 2	ICMP
0.009	ISP 2	ICHECgateway	ICMP
0.010	ICHECgateway	Switch5	ICMP
0.011	Switch5	PC10	ICMP

FIGURE 3.1 – Test de l'OSPF

3.2 NAT dynamique

Dans cette simulation, je teste le bon fonctionnement de la translation d'adresse dynamique. Pour cela, j'effectue un simple ping avec un PC interne au réseau ECAM vers le réseau public. On constate bien que son adresse privée a été convertie en adresse publique en empruntant l'adresse publique 209.165.200.225 de l'intreface gi0/2/0.

3. The packet is going from an inside to an outside network. The device looks up its NAT table for necessary translations.
4. The packet matches an inside source list and creates a new entry for source local address.
5. The device translates the packet from local to global addresses with the matched entry

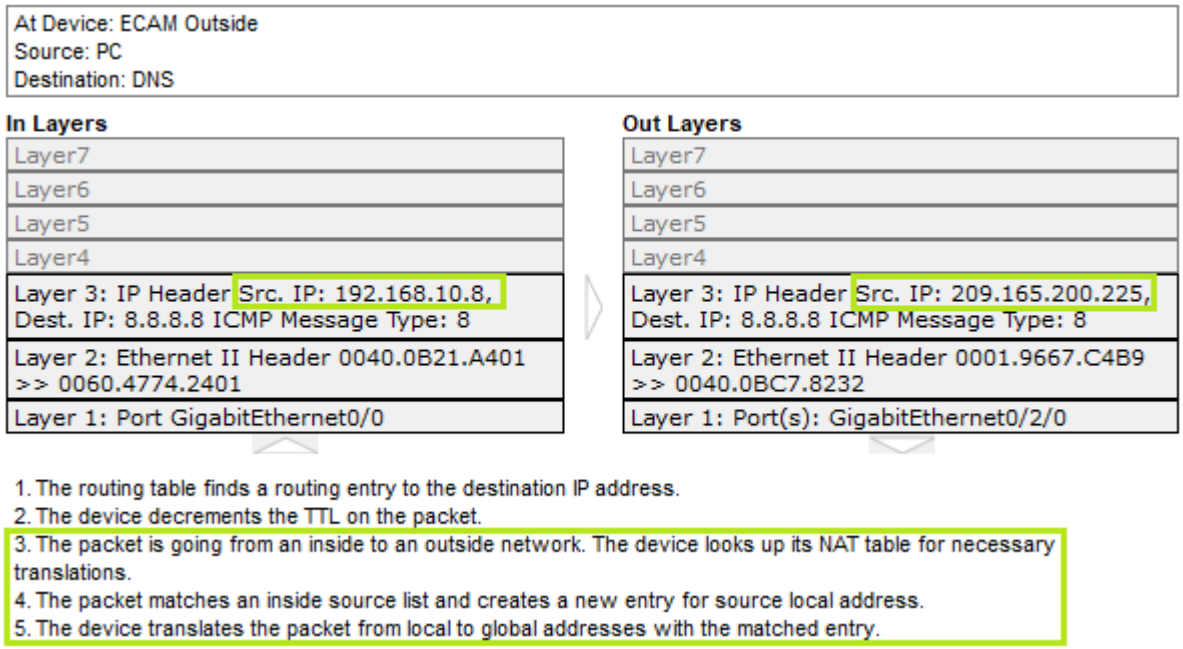


FIGURE 3.2 – NAT Overload Translation - Routeur ECAM Outside

Une vérification de la table de translation du routeur ECAM OUTSIDE nous permet également de vérifier que l'adresse privée 192.168.10.8 à bien été traduite en adresse publique 209.165.200.225 associée au port 16 :

```
ECAMgateway#show ip nat translations
Pro  Inside global      Inside local       Outside local      Outside global
icmp 209.165.200.225:16 192.168.10.8:16    8.8.8.8:16         8.8.8.8:16
---  209.165.200.224    192.168.90.1       ---                ---
tcp 209.165.200.225:443 10.11.11.1:443     ---                ---
tcp 209.165.200.225:80  10.11.11.1:80      ---                ---
```

Remarque : dans la table NAT, *Inside* correspond au routeur ECAM OUTSIDE (global=int gi0/2/0 et local=int gi0/0) et *Outside* est le routeur INTERMEDIATE 4 (local=int gi0/0 et global=int gi0/3/0)

3.3 VPN

Dans le test du VPN, le PC10 du réseau ICHEC effectue un ping vers PC du réseau ECAM en utilisant l'adresse privée de celui-ci, comme s'ils se trouvaient sur le même réseau interne.

On constate bien sur la figure 3.3 que le paquet effectue un aller retour sans encombre entre les deux réseaux. Il est à noter que pour passer vers le VLAN 10 du réseau ECAM, il a fallu mettre à jour l'ACL sur l'interface gi0/0 du routeur ECAM INSIDE afin d'autoriser le traffic entrant depuis le réseau 172.16.10.0 de l'ICHEC.

Time(sec)	Last Device	At Device	Type
0.000	--	PC10	ICMP
0.001	PC10	Switch5	ICMP
0.002	Switch5	ICHECgateway	ICMP
0.003	ICHECgateway	ISP 2	ICMP
0.004	ISP 2	intermediate 4	ICMP
0.005	intermediate 4	ISP	ICMP
0.006	ISP	ECAM Outside	ICMP
0.007	ECAM Outside	ECAM Inside	ICMP
0.008	ECAM Inside	VTP server	ICMP
0.009	VTP server	VTP Client	ICMP
0.010	VTP Client	PC	ICMP
0.011	PC	VTP Client	ICMP
0.012	VTP Client	VTP server	ICMP
0.013	VTP server	ECAM Inside	ICMP
0.014	ECAM Inside	ECAM Outside	ICMP
0.015	ECAM Outside	ISP	ICMP
0.016	ISP	intermediate 4	ICMP
0.017	intermediate 4	ISP 2	ICMP
0.018	ISP 2	ICHECgateway	ICMP
0.019	ICHECgateway	Switch5	ICMP
0.020	Switch5	PC10	ICMP

FIGURE 3.3 – Résumé d'étapes du ping de 192.168.10.4 (ECAM) depuis 172.16.10.6 (ICHEC)

Les routeurs ICHEC GATEWAY et ECAM OUTSIDE assurent le chiffrement du paquet lorsqu'il quitte le réseau privé et le déchiffrement lorsqu'il y entre.

Constatant cela, une meilleure pratique aurait été de configurer le VPN sur le routeur ECAM INSIDE puisque la DMZ donne accès au réseau intermédiaire et donc aux paquets déchiffrés !

Une vérification sur le routeur ICHECGATEWAY nous permet également de confirmer qu'un paquet a bien été chiffré avant d'être envoyé :

```
1 ICHECgateway#show crypto ipsec sa
2
3 interface: GigabitEthernet0/3/0
4     Crypto map tag: VPNMAP, local addr 209.165.207.225
5
6    protected vrf: (none)
7    local  ident (addr/mask/prot/port): (172.16.10.0/255.255.255.0/0/0)
8    remote  ident (addr/mask/prot/port): (192.168.10.0/255.255.255.0/0/0)
```

```
   current_peer 209.165.200.225 port 500
    PERMIT, flags={origin_is_acl,}
   #pkts encaps: 1, #pkts encrypt: 1, #pkts digest: 0
^^I[...]

     local crypto endpt.: 209.165.207.225, remote crypto endpt.:209.165.200.225
     path mtu 1500, ip mtu 1500, ip mtu idb GigabitEthernet0/3/0
     current outbound spi: 0x12207C0B(304118795)

^^I[...]
```

3.4 VLANs + ACL

Dans les tests suivants, je vérifie le bon fonctionnement des listes ACL pour la communication inter-VLANs

VLAN Students vers VLAN Administration : en essayant de faire un *ping* le serveur "ADMIN", on constate bien que la requête est rejetée par l'*ACL extended* présente sur l'interface gi0/1.30.

1. The receiving port has an inbound traffic access-list with an ID of STUDENT. The device checks the packet against the access-list.
2. The packet matches the criteria of the following statement : deny ip 192.168.30.0 0.0.0.255 192.168.10.0 0.0.0.255. The packet is denied and dropped.

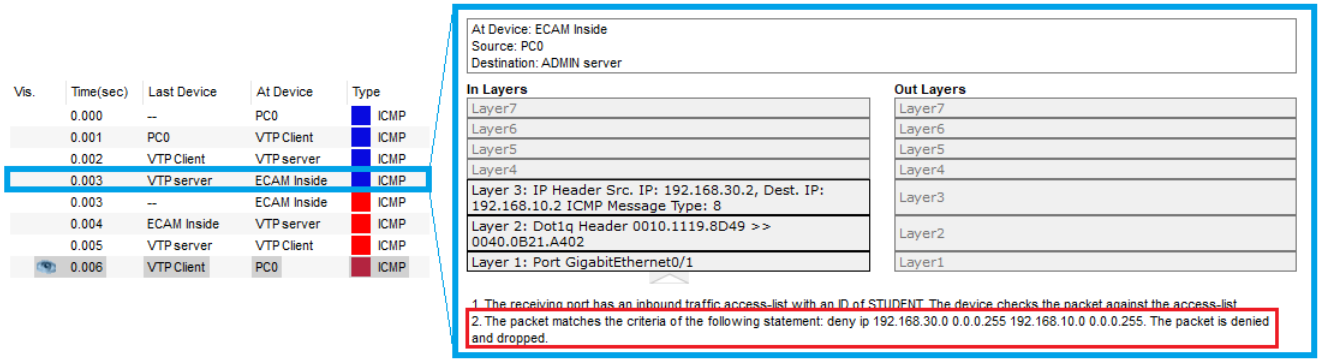


FIGURE 3.4 – Ping VLAN 10 depuis VLAN 30

VLAN Administration vers VLAN Students : en essayant de faire un *ping* depuis le VLAN "Administration" vers le VLAN "Student", on constate bien un retour de réponse après passage au travers l'ACL de gi0/1.30 qui accepte un "echo-reply"

1. The receiving port has an inbound traffic access-list with an ID of STUDENT. The device checks the packet against the access-list.
2. The packet matches the criteria of the following statement : permit icmp 192.168.30.0 0.0.0.255 any echo-reply. The packet is permitted.
3. The device looks up the destination IP address in the routing table.

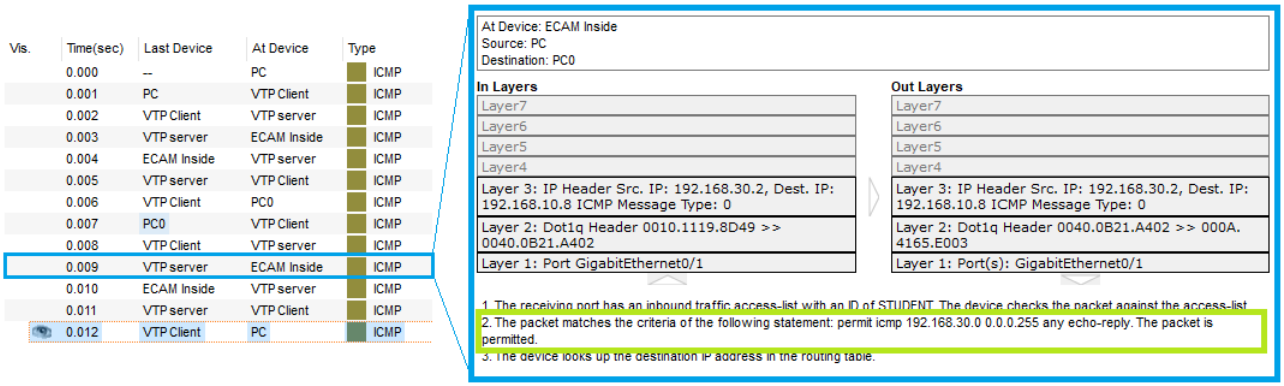


FIGURE 3.5 – Ping VLAN 30 depuis VLAN 10

3.5 DNS + NAT statique + DMZ

Dans le test suivant, le PC2 contacte le serveur web de google depuis son navigateur web.

1. Dans un premier temps, la requête est faite via l'adresse IP publique 209.165.202.2 (au port 80 puisque service web) du serveur. On constate donc que la NAT statique est bien fonctionnelle.
2. Ensuite, la même requête est effectuée en utilisant cette fois-ci l'URL du serveur. On constate ici que le service DNS (8.8.8.8) a bien effectué la conversion vers l'IP correspondante.
3. Ces deux tests précédents nous montrent également le bon fonctionnement de la DMZ puisque la réponse a pu joindre le réseau interne de l'ECAM. En effet, après avoir passé le premier routeur ECAM OUTSIDE l'ACL sur l'interface gi0/0 de ECAM INSIDE a filtré la requête comme étant valide puisque la condition suivante permet un retour lorsqu'une machine initie la connexion TCP (ce qui est bien notre cas pour le protocole HTTP) :

```
1 access-list 110 permit tcp any 192.168.0.0 0.0.255.255 established
```

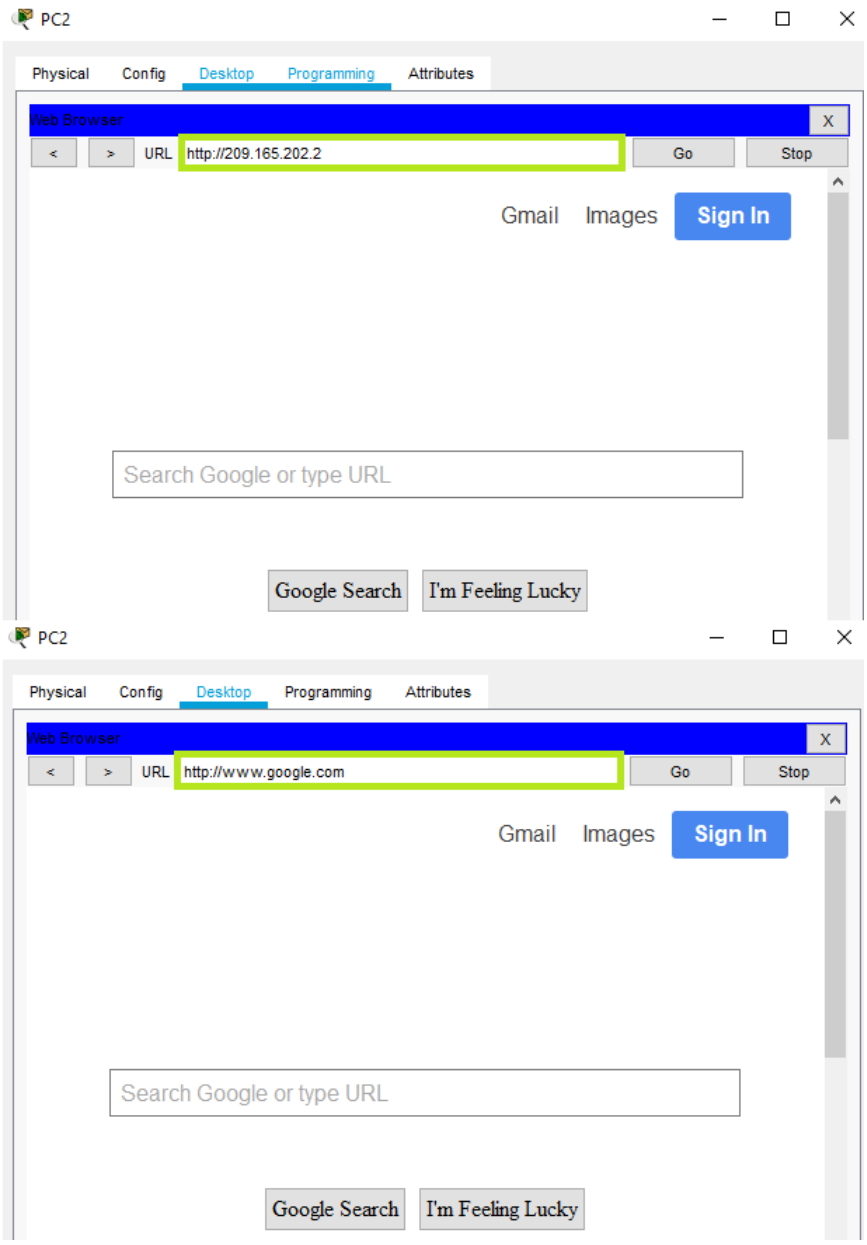
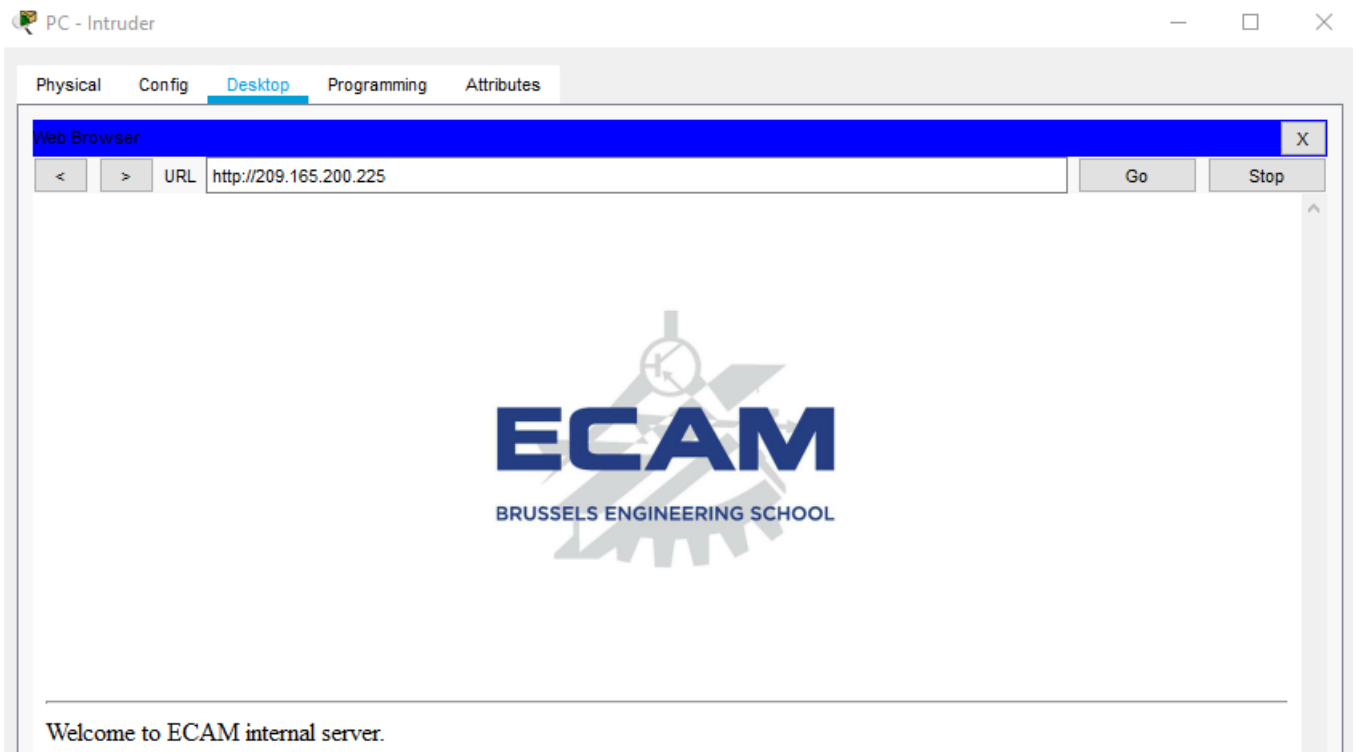


FIGURE 3.6 – DNS + NAT statique + DMZ

3.6 DMZ intrus

Dans cette simulation, je vérifie le bon comportement de la DMZ en effectuant une requête web vers le serveur web du réseau ECAM, à l'adresse publique 209.165.200.225 (NAT statique port 80/443).
A contrario, il n'y a aucun moyen pour l'intru de pénétrer le réseau interne (au-delà de l'intreface gi0/0 du routeur ECAM INSIDE).



Chapitre 4

Conclusion

Ce projet m'a permis de faire beaucoup de liens avec la théorie de réseau et a éclairé de nombreux concepts grâce à la simulation. La "mise en pratique" (simulation) de ce cours a été pour moi le meilleur moyen de le comprendre dans sa globalité.

Après de nombreuses heures passées à configurer ce réseau, les objectifs initiaux ont été atteints et sont fonctionnels. Davantages de recherches m'ont égalemnet fait découvrir des nouveaux concepts, comme la connexion inter-vlans par la configuration *Router On Stick*, la DMZ, les protocoles de routage dynamiques,... qui m'ont été utiles pour améliorer progressivement l'implémentation. Tant d'aspects qui n'étaient pas abordés en théorie et qui représentent la réalité du terrain.

Ce travail m'a également permis de revoir les adresses ip (par essais et erreurs dans un premier temps, pour ensuite être capable de configurer par moi-même) et d'apprendre quelques règles de bonnes pratiques concernant le découpage des plages d'adresses, comme l'optimisation avec un masque de sous réseau /30 pour les connexions point-à point du réseau public.

L'outil de simulation, quant à lui, m'a aidé à de nombreuses reprises pour debugger des parties du réseau qui faisaient défaut. Le même travail de debuggage depuis un terminal, en ligne de commande, aurait été fastidieux pour une première fois. Toutefois, il faut bien garder à l'esprit que c'est de cette façon que l'ingénieur travaille sur le terrain.

Annexe A

Exemples de configurations

Les exemples qui suivent reprennent des parties de configurations pour illustrer les propos théoriques abordés dans le rapport. Ils ont été volontairement raccourci pour ne garder que les lignes intéressantes.

A.1 VLANs

Configuration VLANs :

```
STPserver(config)#vlan 10
STPserver(config-vlan)#name administration

STPserver(config)#interface range fastEthernet 0/4-10
STPserver(config-if-range)#switchport mode trunk
STPserver(config-if-range)#switchport access vlan 10
STPserver(config-if-range)#no shutdown

STPserver#show vlan brief
VLAN Name                             Status    Ports
---- -------------------------------- --------- -------------------------------
1    default                          active    Fa0/1, Fa0/2, Fa0/3, Gig0/1
                                                Gig0/2
10   administration                   active    Fa0/4, Fa0/5, Fa0/6, Fa0/7
                                                Fa0/8, Fa0/9, Fa0/10
20   teachers                         active    Fa0/11, Fa0/12, Fa0/13, Fa0/14
                                                Fa0/15
30   students                         active    Fa0/16, Fa0/17, Fa0/18, Fa0/19
                                                Fa0/20, Fa0/21, Fa0/22
1002 fddi-default                     active
1003 token-ring-default               active
1004 fddinet-default                  active
1005 trnet-default                    active

STPserver#sh vtp status
VTP Version                     : 2
Configuration Revision          : 8
Maximum VLANs supported locally : 255
Number of existing VLANs        : 9
VTP Operating Mode              : Server
VTP Domain Name                 : ecam
```

Router on stick :

```
ECAMInside#show run

[...]

interface GigabitEthernet0/1.10
 encapsulation dot1Q 10
 ip address 192.168.10.1 255.255.255.0
 ip helper-address 10.1.1.2
```

A.2 DMZ - ACL

```
ECAMInside#sh run | begin access-list

! ACL interface côté réseau interne
access-list 100 permit tcp 192.168.0.0 0.0.255.255 host 10.11.11.1 eq www
access-list 100 permit tcp 192.168.0.0 0.0.255.255 host 10.11.11.1 eq 443
access-list 100 permit udp 192.168.0.0 0.0.255.255 host 8.8.8.8 eq domain
access-list 100 permit udp 192.168.0.0 0.0.255.255 host 10.1.1.2 eq bootpc
access-list 100 permit udp 192.168.0.0 0.0.255.255 host 10.1.1.2 eq bootps

! ACL interface côté DMZ
access-list 110 permit ip 172.16.10.0 0.0.0.255 192.168.10.0 0.0.0.255
access-list 110 permit tcp any 192.168.0.0 0.0.255.255 established
access-list 110 permit icmp any 192.168.0.0 0.0.255.255 echo-reply
access-list 110 permit udp any any

! ACL pour isoler les étudiants des autres VLANs
ip access-list extended STUDENT
 permit icmp 192.168.30.0 0.0.0.255 any echo-reply
 deny ip 192.168.30.0 0.0.0.255 192.168.10.0 0.0.0.255
 deny ip 192.168.30.0 0.0.0.255 192.168.20.0 0.0.0.255
 permit ip any any
```

A.3 DHCP

```
ip dhcp excluded-address 192.168.10.1 192.168.10.5
!
ip dhcp pool ECAMinternalVLAN10
 network 192.168.10.0 255.255.255.0
 default-router 192.168.10.1
 dns-server 8.8.8.8

[...]
```

A.4 NAT

```
ECAMOutside#sh run
interface GigabitEthernet0/1
 ip address 10.3.3.1 255.255.255.0
 ip nat inside

interface GigabitEthernet0/2/0
 ip address 209.165.200.225 255.255.255.252
 ip nat outside

! NAT Overload Translation
ip nat inside source list NAT interface GigabitEthernet0/2/0 overload

! NAT statique pour l'accès au serveur web
ip nat inside source static tcp 10.11.11.1 80 209.165.200.225 80
ip nat inside source static tcp 10.11.11.1 443 209.165.200.225 443

! Les plages d'adresses à translater
ip access-list extended NAT
 ! Il faut exclure la communication VPN
 deny ip 192.168.10.0 0.0.0.255 172.16.10.0 0.0.0.255
 permit ip 192.168.10.0 0.0.0.255 any
 permit ip 192.168.20.0 0.0.0.255 any
 permit ip 192.168.30.0 0.0.0.255 any
```

A.5 Routage (statique + OSPF)

```
ISP#sh run | begin ospf
router ospf 1
 router-id 1.1.1.1
 log-adjacency-changes
 redistribute static subnets
 network 209.165.201.224 0.0.0.3 area 0
 network 209.165.205.224 0.0.0.3 area 0
!
ip classless
ip route 209.165.200.224 255.255.255.240 GigabitEthernet0/0/0
```

A.6 VPN

```
ECAMOutside#sh run
license boot module c2900 technology-package securityk9
!
crypto isakmp policy 10
 encr aes
 authentication pre-share
 group 5
!
crypto isakmp key KEY-VPN address 209.165.207.225
!
!
!
crypto ipsec transform-set CRYP-SET esp-aes esp-sha-hmac
!
crypto map VPN-MAP 10 ipsec-isakmp
 set peer 209.165.207.225
 set transform-set CRYP-SET
 match address VPN

interface GigabitEthernet0/2/0
 ip address 209.165.200.225 255.255.255.252
 ip nat outside
 crypto map VPN-MAP
```