AC11.04 Maîtriser les rôles et les principes fondamentaux des systèmes d'exploitation afin d'interagir avec ceux-ci pour la configuration et l'administration des réseaux et services fournis

Ressources : SAE R108 R102 SAé1.01 Saé1.2

R1.02

Que se passe-t-il ?
Dans le cadre de l'étude du protocole FTP, une topologie réseau a été mise en place pour connecter un client à un serveur FTP. Cette configuration a permis d'analyser les paquets échangés et de mieux comprendre les interactions entre les différentes couches du modèle OSI.

Comment cela se passe (déroulement, circonstances) ?
Le client FTP a été configuré pour se connecter au serveur à l'adresse IP 172.16.254.3. Une requête FTP a été envoyée pour récupérer un fichier nommé s1-central. À l'aide d'un outil d'analyse réseau, comme Wireshark, les paquets échangés ont été capturés. Ces paquets ont ensuite été interprétés pour identifier les interactions entre les couches OSI, depuis la commande FTP (couche application) jusqu'à la transmission des données sur le support réseau (couche physique).

Pourquoi cela se passe-t-il ainsi ?
Ce processus a permis de mettre en pratique les concepts théoriques du modèle OSI et de comprendre le fonctionnement du protocole FTP. L'analyse des paquets a également renforcé les compétences en diagnostic réseau et en interprétation des échanges entre un client et un serveur.

Que peut-on améliorer ?
Il serait intéressant d'étendre l'analyse à d'autres modes de transfert FTP, comme le mode actif et le mode passif, pour mieux comprendre leurs différences. De plus, une exploration des erreurs possibles et des méthodes de résolution pourrait enrichir l'expérience.

Comment peut-on améliorer ?
En configurant des scénarios plus complexes impliquant des transferts sécurisés (via FTPS ou SFTP), et en utilisant des outils avancés d'analyse réseau pour capturer des données supplémentaires. Des études de cas sur des transferts de fichiers volumineux ou sur des environnements réseau contraints pourraient également être bénéfiques.

Que se passe-t-il ?
Dans un environnement de machine virtuelle sous Linux Ubuntu, l'outil Python Scapy a été utilisé pour créer et analyser des paquets réseau. Cette expérimentation visait à approfondir la compréhension des couches du modèle OSI et de la structure des trames Ethernet.

Comment cela se passe (déroulement, circonstances) ?
La trame Ethernet a été créée à l'aide de la commande ma_trame = Ether() dans Scapy, permettant d'ajouter des champs personnalisés comme dst (adresse MAC de destination), src (adresse MAC source) et type (protocole encapsulé). Les détails de la trame ont ensuite été affichés avec la commande ma_trame.show(). Lors de l'analyse des paquets, un avertissement a été généré (Mac address to reach destination not found. Using broadcast.), car aucune adresse de destination spécifique n'avait été définie, ce qui a entraîné l'utilisation de l'adresse de diffusion.

Pourquoi cela se passe-t-il ainsi ?
Cette démarche a permis de mettre en pratique les concepts théoriques des couches OSI et de la structure des trames Ethernet. L'utilisation de Scapy a offert une approche pratique pour personnaliser et analyser les différents champs d'une trame, renforçant les compétences en manipulation de paquets réseau.

Que peut-on améliorer ?
Il serait intéressant d'approfondir l'expérimentation en spécifiant une adresse de destination valide au lieu de l'adresse de diffusion par défaut. De plus, explorer d'autres protocoles encapsulés, comme ARP ou IP, permettrait de mieux comprendre leur intégration dans les trames Ethernet.

Comment peut-on améliorer ?
En testant des scénarios plus complexes, comme l'interaction entre plusieurs machines virtuelles, ou en utilisant des captures réseau réelles pour analyser des paquets au-delà des trames créées manuellement. Une documentation approfondie sur les avertissements et erreurs de Scapy pourrait également faciliter leur résolution et leur interprétation.

R1.08

Que se passe-t-il ?
Une communication locale a été établie entre un serveur Ubuntu (shemsserv), un client Ubuntu (client-nfs), et une machine Windows (windows). Cette connexion repose sur des adresses IP internes et une configuration des fichiers réseau pour permettre une résolution locale des noms.

Comment cela se passe (déroulement, circonstances) ?
Les fichiers /etc/hosts des machines Ubuntu ont été modifiés pour inclure les noms des machines et leurs adresses IP internes. Des adresses IP statiques ont été attribuées à chaque machine pour garantir une connectivité stable. Enfin, des tests de connectivité, via les commandes réseau ping et nslookup, ont été réalisés pour valider la communication entre les machines.

Pourquoi cela se passe-t-il ainsi ?
Cette démarche permet de garantir une résolution de noms locale et fiable sans dépendre d'un serveur DNS externe. Elle facilite également la gestion et l'identification des machines au sein du réseau local, tout en assurant une connectivité efficace entre elles.

Que peut-on améliorer ?
Il serait intéressant d'automatiser la configuration des fichiers /etc/hosts et des adresses IP à l'aide d'outils comme Ansible. De plus, l'intégration d'un serveur DNS interne pourrait simplifier la gestion des noms lorsque le réseau s'étend.

Comment peut-on améliorer ?
En explorant des solutions avancées comme la mise en place d'un serveur DHCP pour attribuer automatiquement les adresses IP, ou en utilisant un gestionnaire de configuration pour uniformiser les paramètres réseau sur toutes les machines.

SAÉ1.02

Que se passe-t-il ?
Dans le cadre de l'administration réseau, plusieurs services critiques ont été configurés et gérés, notamment un serveur DHCP pour l'attribution automatique des adresses IP, pour la résolution de noms, ainsi que SSH pour une administration sécurisée à distance.

Comment cela se passe (déroulement, circonstances) ?
Un serveur DHCP a été mis en place pour attribuer automatiquement des adresses IP aux clients du sous-réseau 192.168.1.0/24, avec des réservations configurées pour les équipements critiques. Parallèlement, un serveur DNS a été déployé pour gérer la zone directe du domaine exemple.local, assurant une résolution de noms efficace pour les clients. Enfin, SSH a été activé et sécurisé, avec la génération et la gestion de clés SSH pour un accès restreint aux utilisateurs autorisés.

Pourquoi cela se passe-t-il ainsi ?
Ces actions permettent d'assurer une gestion efficace et sécurisée des ressources réseau. Le serveur DHCP automatise la configuration des adresses IP, réduisant les erreurs manuelles. SSH garantit une administration distante sécurisée, limitant les accès aux seuls utilisateurs autorisés.

Que peut-on améliorer ?
Il serait possible d'ajouter des mécanismes de redondance pour le serveur DHCP afin de garantir leur disponibilité en cas de panne. De plus, l'activation de SSH sur tous les équipements réseau, accompagnée d'un monitoring centralisé, renforcerait la sécurité globale.

Comment peut-on améliorer ?
En explorant des solutions avancées comme la mise en cluster des serveurs DHCP ou en utilisant des outils comme Ansible pour automatiser la configuration des clés SSH sur l'ensemble des systèmes. Des audits réguliers de sécurité et des mises à jour des services contribueraient également à améliorer la fiabilité et la sécurité des services.