

Architecture des Systèmes d'Exploitation – Laboratoire

1) Modalités du cours

Ce laboratoire est à réaliser de manière **individuelle**, il intègre une grande partie des techniques vues durant le cours d'Architecture des Systèmes d'Exploitation – Théorie.

2) Contexte

Un petit cabinet comptable (plus ou moins 20 employés) désire informatiser ses services et aimerait se moderniser technologiquement en proposant à ses employés une pléthore de services et d'outils collaboratifs internes afin d'améliorer leur productivité tels que le propose Microsoft avec Office365. Vu que les données sont des données sensibles, ils aimeraient bien héberger ces services en interne en utilisant des logiciels libres.

L'entreprise ne dispose pour le moment d'aucune infrastructure. Les utilisateurs (RH, techniciens, chef comptables et comptables) utilisent principalement leur ordinateur de bureau (Windows 10 ou 11) et leur smartphone pour travailler. D'un point de vue réseau, les utilisateurs sont directement reliés au routeur du FAI via un switch ou via le wifi. L'entreprise ne souhaite pas pour le moment réorganiser son infrastructure réseau, ce point ne sera donc pas évalué dans ce cours. Afin de mettre à bien votre architecture, l'entreprise est prête à faire des investissements dans des serveurs pour pouvoir la supporter.

Le client porte un point d'honneur à ce que l'architecture soit bien **sécurisée** vu qu'il héberge des données sensibles.

3) Implémentation

3.1) Architecture et services à implémenter

Un serveur dédié sous Fedora 36 permettra aux clients Windows de se connecter via un service type « Active Directory » mais donc sous Linux. Ce serveur aura besoin de plusieurs autres services pour son bon fonctionnement à vous de les installer. Attention, cette partie (Fedora 36) **ne peut pas tourner avec Docker**.

Un autre serveur dédié sous Debian 11 sera nécessaire pour faire tourner les outils suivants implémentés grâce à Docker :

- L'outil de versionnage GitLab pour l'équipe de chercheurs
- Le système de gestion de fichiers NextCloud
- L'outil de vidéoconférence Jitsi Meet

- Un service web pour ajouter/supprimer/modifier les utilisateurs dans l'« active directory » (voir plus loin dans le document)
- Un reverse proxy pour rediriger les requêtes vers les bons services

Tous ces services doivent utiliser le serveur « Fedora 36 » pour pouvoir authentifier les utilisateurs. Les utilisateurs présents dans l'« active directory » devront donc pouvoir se connecter via les mêmes identifiants.

Vous devez uniquement configurer les services pour que les utilisateurs puissent s'authentifier avec l'« AD », pas besoin de configuration supplémentaire.

Pour tous ces services (tournant donc sur le serveur Debian 11) l'utilisation de Docker est obligatoire. Si d'autres services doivent être installés pour le bon fonctionnement de l'architecture, ceux-ci devront se trouver sur le serveur « Fedora 36 », pour rappel, vous ne pouvez pas utiliser Docker sur la machine Fedora.

3.2) Service web de contrôle de l'« AD »

Afin de faciliter le contrôle de cet « AD » sous Linux, il vous est demandé de développer une interface web en Java qui devra tourner sous Docker. Cette interface web, devra contenir une page de login, permettant uniquement aux utilisateurs ayant le rôle « technicien » de pouvoir s'y connecter.

Cette interface permettra de :

- Ajouter un nouvel avec toutes ses informations associées (rôle, nom, identifiants, etc)
- Modifier un utilisateur
- Supprimer un utilisateur

Cette interface devra juste être fonctionnelle et sera créée dans le but de faciliter la gestion de l'« AD » dans le futur. Le style graphique ne sera pas donc évalué. La seule contrainte est qu'elle doit-être sécurisée (à vous de voir ce qu'il faut mettre ne place pour).

Lors de la création de ce conteneur **veillez à bien appliquer toutes les notions théoriques** que vous avez vues ! (Multi stage build, la bonne façon de créer un Dockerfile, etc)

3.3) Utilisation d'Ansible

Il vous est demandé d'utiliser Ansible pour automatiser le déploiement de l'architecture sur les serveurs Fedora et Debian. L'objectif étant, à partir d'images « vierges », de déployer l'ensemble des services (« Active Directory », Docker, ...), de déployer l'ensemble des conteneurs et de lancer l'exécution de ces derniers. Le tout devant être intégré dans un script de déploiement unique.

La création de ces images « vierges » est laissée à vos soins, seule la connexion de la machine de contrôle Ansible vers ces dernières doit être préconfigurée (déploiement des clés SSH,

connexion WinRM, ...).

4) Évaluation

Vous serez évalués de manière continue durant le quadrimestre à concurrence de **35 %** de la note, les **65 %** restants seront évalués lors de l'examen final. Pour rappel, cette UE est en évaluation intégrée, ce qui veut dire qu'il n'y a qu'une seule note pour l'intégralité l'UE (Théorie et Laboratoire). Vous êtes donc susceptible d'avoir des questions portant sur la théorie lors des évaluations. Les évaluations seront conjointement évaluées par monsieur Duvillié et monsieur Tillieux. Concernant la seconde session, l'énoncé reste le même, vous nous représenterez la ou les parties en échec le jour de l'examen.

Il vous est demandé de créer un scénario plausible le jour où vous nous présenterez votre solution. Créez donc quelques utilisateurs à l'avance pour qu'on puisse directement tester votre solution. Vos machines devront être démarrées et prêtes à être testées. Un client Windows 10 et Ubuntu 22.04 devront être préparés pour qu'on puisse tester votre solution (ils ne sont pas obligés de tourner en même temps si vous êtes limités en RAM).

Vous devrez nous présenter la première partie de votre travail à la date du **à définir, un mail vous sera envoyé**. Cette première partie reprend :

- La mise en place de l'« AD » sur le serveur Fedora
- L'authentification via l'« AD » sur les clients Windows 10 et Ubuntu
- La mise en place et la configuration des services GitLab, NextCloud, et le reverse proxy

Le reste (Ansible et la création du service web en Java et la mise en place de Jitsi) sera évalué le jour de l'examen. Attention, lors de cette deuxième évaluation, nous pouvons revenir sur des points de la première évaluation si ça nous semble nécessaire.

De façon générale, pensez à bien appliquer tous les concepts et les bonnes pratiques que vous avez apprises lors du cours théorique dans ce laboratoire, nous y serons particulièrement attentifs.

La solution n'est pas obligée d'être accessible depuis l'extérieur du réseau de l'entreprise.

Conseils

Pour pouvoir réaliser ce laboratoire nous mettons à votre disposition des serveurs (si vous en faites la demande). Vous pouvez aussi très bien utiliser un logiciel de virtualisation tel que « VirtualBox ». Normalement si votre ordinateur a au moins 8Gb de RAM ça ne devrait pas poser de problème.

Nous avons fait exprès de ne pas avoir été explicite dans cet énoncé, d'autres choses doivent être mises en place pour que la solution fonctionne. Vous le découvrirez par vous-même lors de la mise en place de la solution.

Si vous avez des questions vous pouvez les poser à monsieur Tillieux

(arnaud.tillieux@henallux.be) ou à monsieur Duvill  (guillaume.duvillie@henallux.be).

Pour des fins d'archivage il vous est demand  de cr er un repository GitLab dans votre espace personnel avec les scripts, docker-compose, dockerfile, etc, que vous avez cr   lors de la r alisation de ce laboratoire.

Bon travail.