

APACHE PERFECTIONNEMENT

4 jours

OBJECTIFS & COMPETENCES

Héberger des applications PHP
Contrôler l'accès et l'authentification à un serveur Apache
Mettre en oeuvre les redirections, la réécriture d'URL et les filtres
Mettre en oeuvre un reverse-proxy haute disponibilité
Sécuriser les échanges avec HTTPS

PUBLIC VISE

Administrateurs de serveur Web, exploitants, intégrateurs ou architectes techniques.

PRE-REQUIS

Bonnes connaissances de l'administration du serveur Web Apache ou équivalentes à celles apportées par le stage "Apache, administrer un serveur Web" (réf. LIA).
Expérience souhaitable

PROGRAMME

Apache HTTPD 2.4 : rappels et nouveautés

Compilation, installation et test initial.
Configuration générale du serveur.
Choisir le bon MPM, gérer la charge et les limites.
Chargement des modules, quels modules activer ?
Les nouveaux types de contextes.
Panorama des modules d'Apache 2.4.
Support du protocole HTTP/2.

Travaux pratiques
Compilation et installation d'Apache HTTPD 2.4.

Héberger des applications PHP

Faire cohabiter PHP5 et PHP7.
CGI, CGID, Fast CGI et PHP-FPM.
Droits et identité dédiée, sessions.

Travaux pratiques

Gérer plusieurs versions de PHP5 et PHP7 via Fast CGI/PHP-FPM.

Contrôle d'accès et authentification

Contrôle d'accès : les modules mod_authz*.

Authentification LDAP avec mod_authnz_ldap.

Authentification externe et SGBD (dbm, mysql...).

Travaux pratiques

Mise en place de l'authentification basée sur un annuaire LDAP et sur une base MySQL.

Redirection, réécriture d'adresses, filtres

DocumentRoot et le module mod_alias.

Notion de répertoire virtuel et d'alias.

Les règles de ré-écriture d'URL et mod_rewrite.

Exemples de filtres avec mod_filter, mod_header.

Travaux pratiques

Mise en oeuvre de redirections et filtres.

Reverse Proxy et Cache

Les fonctionnalités de proxy et de Reverse Proxy.

Les modules mod_proxy* dans Apache HTTPD 2.4.

Répartition de charge et affinité de session.

Gestion de l'état des serveurs, fail-over, heartbeat.

KeepAlive et la haute disponibilité du Reverse-Proxy.

Gestion du cache mémoire et disque.

Travaux pratiques

Mise en oeuvre de reverse-proxy haute disponibilité.

Sécuriser les échanges avec HTTPS

Mise en oeuvre de HTTPS.

CA et gestion de certificats serveurs.

Authentifier les utilisateurs.

Création de certificats et d'hôtes virtuels HTTPS.

Travaux pratiques

Génération de certificats serveurs et mise en place d'un site virtuel HTTPS.

Authentification des visiteurs d'un site HTTPS par présentation de certificat client.

Sécurité et détection d'attaques

Qu'est-ce que mod_security ?

Le principe des règles de mod_security.

Détecter et bloquer les tentatives d'attaques.

Une alternative à un IDS comme Snort.