

# Formation Écriture de script en Shell

## Description de la formation Shell :

Objectif Formation Shell : Le Shell est le programme auquel nous soumettons nos instructions quand nous dialoguons avec un système d'exploitation, soit à travers une simple commande tapée au clavier, soit à travers des fichiers appelés Shell scripts. Son fonctionnement et sa batterie de caractères spéciaux (\*, [, &, #, `,(,|, \, « , \$, ;, !, etc.) le rendent très puissant (mais pas très lisible au début), d'autant plus qu'il est programmable (if, while, fonction, etc.).

Cette formation présente les mécanismes détaillés du fonctionnement du Shell standard sous Linux (bash), puis de sa programmation. Vous serez ainsi totalement à l'aise quand vous taperez des commandes complexes.

Surtout, vous serez autonome dans la programmation de Shell scripts dans des domaines d'applications du quotidien : surveillance des ressources du système, occupation des filesystems, processus consommateurs, connexions en cours, déclenchement d'alertes, génération de ses propres journaux sous forme texte ou HTML pour publication Web, envoi de mails ou de SMS, transfert de fichiers automatisés, etc.

## Objectifs de la formation Shell :

Objectif opérationnel :

Écrire des commandes et programmer en Shell.

Objectifs pédagogiques :

- Comprendre les mécanismes de substitutions du shell (\$, `, ` , \*, \, etc.)
- Lancer un Shell (PATH, filiation, exec, fork, &, |, ;, etc.)

- Maîtriser les structures de contrôles (if, while, ...)
- Programmer et debugger un Shell script
- Maîtriser les expressions régulières
- Utiliser les outils sed et awk dans des Shell scripts
- Faire le lien entre les concepts et outils présentés dans cette formation avec les besoins du quotidien des développeurs de Shell scripts

## À qui s'adresse cette formation ?

Public :

Ce cours s'adresse aux informaticiens chargés de développer des Shell-scripts, mais aussi aux correspondants informatiques ayant la responsabilité de grosses applications et aux responsables système ou d'exploitation.

Prérequis :

Les participants à ce stage doivent avoir une formation de base à l'utilisation d'un système UNIX/Linux.

## Contenu du cours Shell

Présentation et rappels

Principes

Les différents interpréteurs : Bourne Shell, Korn Shell, Bash, C Shell, Tesh...

Disponibilité des interpréteurs sur les divers systèmes.

Le point sur la normalisation (impacts sur l'écriture des scripts)

Les apports GNU (gawk, gsed...)

Différences Bourne Shell/Korn Shell/Bash

Programmation par scripts

Outils de développement

Mécanisme d'exécution des scripts

Règles de recherche des commandes

Principes d'exécution d'une commande (exec, pipeline, sous-Shell, background...)

Mécanismes de base

Lecture et analyse de la ligne de commande

Expansion des accolades, développement du tilde, remplacement des paramètres

Substitution des commandes et évaluation arithmétique

Procédés d'échappement (banalisation)

Les redirections (entrée et sortie standards, fichiers, tubes, document en ligne)

Fonctionnement en interactif

Invocation du Shell (options).

Les différents fichiers de démarrage.

Notions d'environnement (variables, alias, fonctions).

Historique et rappel des commandes. Contrôle de jobs.

La complémentation des noms. Terminaison du Shell.

Construction de Shell-scripts portables (ksh/bash)

Interface avec un Shell-script.

Structuration d'un Shell-script.

Appel d'un Shell-script

Rappels sur les différentes méthodes.

Options utilisables.

Notions de sous-Shell.

Compléments sur l'environnement.

Options et arguments.

Préambule du Shell-script

Qui interprète le Shell-script ?

Commentaires.

Paramètres de position (initialisation, sauvegarde, décalages).

Variables locales.

Variables globales.

Déclaration et visibilité des fonctions.

Postambule et retour de Shell-script

Sortie du Shell-script.

Fonction de sortie.

Conventions utilisées.

Valeur de retour.

Enchaînement de Shell-scripts.

Structures de contrôle du Shell

Commandes simples, pipelines, et listes de pipelines

Commandes composées, sous-Shell et fonctions

Mécanismes de sélection et d'itération

Menus

Commandes internes et externes

Entrées/Sorties.

Interactions avec le système.

Arguments en ligne de commande.

Opérations de tests.

Compléments sur les instructions.

Redirection locale dans les fichiers de commande

Mécanismes complémentaires

Debugging d'un Shell-script

Commandes de debugging.

Signaux de trace.

Journalisation.

Robustesse d'un Shell-script

Vérifier l'initialisation des variables.

Gestion avancée des arguments en ligne de commande (getopts).

Tests sur le type des variables (expr).

Gestion des signaux

Autres points

Nommage et destruction des fichiers temporaires.

Cas particulier d'exécution d'un Shell-script par cron.

La commande eval

Le bêtisier du Shell.

Extensions du Korn Shell et Bash

Tableaux de variables. Notations spécifiques.

Opérations arithmétiques. Les alias suivis.

Notion de Co processus.

Les commandes internes spécifiques.

Outils supplémentaires

Outils d'assistance pour la création de scripts

Utilitaires système SUSv3

Expression rationnelles : outil grep

Recherche et traitement de fichiers : outil find

Manipulation de flux de texte avec sed

Principes et aspects des scripts.

Commandes simples de recherche, remplacement, insertion.

Aperçu des commandes avancées (traitement multiligne, mémoire secondaire...)

Automatisation de tâches avec awk

Éléments généraux de programmation avec awk.

Utilisation des variables et des fonctions

Présentation des fonctions intégrées : mathématique, traitement de chaîne, interaction avec le système...

Exemples complets de scripts Awk (statistiques système, calculs...)