

Durée : 3 jours soit 21 heures

Référence : IF-BASH

Public visé :

- Administrateurs Systèmes et Réseaux.
- Ingénieurs DevOps et SRE.
- Développeurs souhaitant automatiser leurs environnements de déploiement.
- Techniciens d'exploitation souhaitant fiabiliser leurs tâches récurrentes.

Pré-requis :

- Maîtrise des commandes de base de la navigation Linux (cd, ls, cp, mkdir).
- Notions élémentaires sur les droits d'accès et les éditeurs de texte (Vim, Nano).
- Une première expérience en programmation (quel que soit le langage) est un plus.

Objectifs pédagogiques :

À l'issue de cette formation, le stagiaire sera capable de :

- **Concevoir des scripts structurés** en respectant les standards professionnels (Shebang, modularité).
- **Maîtriser la manipulation de flux** et de données complexes avec les expressions régulières, Sed et Awk.
- **Fiabiliser l'exécution** via une gestion avancée des erreurs, des logs et du débogage.
- **Développer des outils interactifs** gérant des options en ligne de commande (getopts).
- **Interagir avec des formats de données modernes** (parsing JSON simple) et des processus parallèles.

Compétences acquises à l'issue de la formation :

- Concevoir des scripts structurés en respectant les standards professionnels (Shebang, modularité).
- Maîtriser la manipulation de flux et de données complexes avec les expressions régulières, Sed et Awk.
- Fiabiliser l'exécution via une gestion avancée des erreurs, des logs et du débogage.
- Développer des outils interactifs gérant des options en ligne de commande (getopts)
- Interagir avec des formats de données modernes (parsing JSON simple) et des processus parallèles.

Modalités pédagogiques :

Session dispensée en présentiel ou téléprésentiel, selon la modalité inter-entreprises ou intra-entreprises sur mesure.

La formation est animée par un(e) formateur(trice) durant toute la durée de la session et présentant une suite de modules théoriques clôturés par des ateliers pratiques validant l'acquisition des connaissances. Les ateliers peuvent être accompagnés de Quizz.

L'animateur(trice) présente la partie théorique à l'aide de support de présentation, d'animation réalisée sur un environnement de démonstration.

En présentiel comme en téléprésentiel, l'animateur(trice) accompagne les participants durant la réalisation des ateliers.

Moyens et supports pédagogiques :

Cadre présentiel

Salles de formation équipées et accessibles aux personnes à mobilité réduite.

- Un poste de travail par participant
- Un support de cours numérique ou papier (au choix)
- Un bloc-notes + stylo
- Vidéoprojection sur tableau blanc
- Connexion Internet
- Accès extranet pour partage de documents et émargement électronique

Cadre téléprésentiel

Session dispensée via notre solution iClassroom s'appuyant sur Microsoft Teams.

- Un compte Office 365 par participant
- Un poste virtuel par participant
- Un support numérique (PDF ou Web)
- Accès extranet pour partage de documents et émargement électronique

Informations sur l'accessibilité :

Nos formations sont, dans la mesure du possible, conçues pour être accessibles à toutes et à tous. Afin de garantir les meilleures conditions d'accueil et d'apprentissage pour les personnes en situation de handicap, nous vous invitons à contacter notre référente handicap certifiée :

Céline SOLATGES – 05 61 34 39 80 – csolatges@iform.fr

Nous vous remercions de bien vouloir nous communiquer toute information utile à ce sujet en amont de la formation, afin de mettre en place les adaptations nécessaires et d'assurer un accompagnement optimal.

Pour en savoir plus sur les dispositifs d'accompagnement existants, vous pouvez consulter les sites suivants :

- [AGEFIPH](#)
- [FIPHP](#)
- MDPH de votre département



Description / Contenu

Module 1 : L'Environnement de Scripting et Fondamentaux

- **Le Shell de nos jours** : Pourquoi Bash reste le standard malgré Python/Go.
- **Le processus d'exécution** : Filiation, Fork, Exec et l'importance du Shebang.
- **Mécanismes de substitution** : Accolades, Tilde, variables et évaluation arithmétique.
- **Quoting & Escaping** : Maîtriser les ' ', " ", et \ pour éviter les erreurs d'interprétation.
- **TP** : Configuration d'un environnement de développement propre et création d'un script d'inventaire système automatisé récupérant les informations de l'hôte.

Module 2 : Programmation Structurée et Flux

- **Variables et Portée** : Variables locales vs globales, constantes (readonly).
- **Structures de contrôle avancées** : * Tests modernes [[...]] vs [...].
- Boucles intelligentes (for, while, until) et lecture de fichiers ligne à ligne.
- **Gestion des Flux** : Redirections complexes, Descripteurs de fichiers (FD) et Pipes.
- **TP** : Création d'un script de surveillance d'espace disque qui génère une alerte par mail ou log uniquement si un seuil critique est atteint sur une partition spécifique.

Module 3 : Traitement de Texte et Expressions Régulières (Le moteur de Bash)

- **Regex** : Standards Basiques (BRE) et Étendus (ERE).
- **Grep & Find** : Le duo gagnant pour le filtrage de fichiers de masse.
- **Sed (Stream Editor)** : Substitutions, suppressions et transformations de flux.
- **Awk (Le langage de traitement)** : Extraction de colonnes, variables de calcul et rapports.
- **TP** : Analyseur de logs Apache/Nginx : extraire les 10 adresses IP les plus actives et calculer le volume total de données transférées.

Module 4 : Robustesse, Débogage et Sécurité

- **Gestion des erreurs** : Codes de retour (\$?), utilisation de set -e, set -u, set -o pipefail.
- **Trap** : Capturer les signaux (Ctrl+C) pour nettoyer les fichiers temporaires.
- **Débugage** : Mode trace (set -x), outils de linting (ShellCheck).
- **Sécurité** : Prévention de l'injection de commandes et gestion des données sensibles.
- **TP** : Sécurisation et fiabilisation d'un script de sauvegarde de base de données : ajout de logs horodatés, vérification de l'existence du répertoire cible et gestion du signal d'interruption.

Module 5 : Fonctions, Modularité et Interface

- **Bibliothèques de fonctions** : Créer des fonctions réutilisables et les "sourcer".
- **Interaction utilisateur** : Gestion des arguments (\$1, \$#) et analyse des options professionnelles avec getopt.
- **Interface Textuelle** : Introduction à whiptail ou dialog pour créer des menus.
- **Modernisation** : Introduction au parsing JSON avec jq (indispensable pour les API en 2026).

- **TP** : Création d'un outil d'administration "menu-driven" permettant de choisir entre plusieurs actions (backup, check santé, nettoyage) avec passage d'options en ligne de commande.

Module 6 : TP de Synthèse - Projet Fil Rouge (4h)

- **Sujet** : Développement d'un "Automate de Déploiement et de Monitoring".
- **Objectif** : Créer un script capable de télécharger une archive web, de vérifier son intégrité, de l'installer, de configurer les droits, et de mettre en place un monitoring de l'application.
- **Contraintes** : Le script doit être modulaire, documenté, gérer les erreurs proprement, et proposer un mode "Dry-run" (simulation).