Table des matières

Avant-propos

Chapitre 1
Présentation de Linux

1. Bienvenue dans le monde Linux 23
   - 1.1 Un système en évolution 23
   - 1.2 Le système d'exploitation 23
   - 1.3 Le système Unix, une brève histoire 26
     - 1.3.1 De Multics à Unix 26
     - 1.3.2 Le langage C 26
     - 1.3.3 Les différents types d'Unix 27
2. Le logiciel libre 27
   - 2.1 Les origines du logiciel libre 27
   - 2.2 GNU/Linux 29
     - 2.2.1 Linus Torvalds 29
     - 2.2.2 Le succès communautaire 30
     - 2.2.3 Les années 1994-1997 30
     - 2.2.4 Linux aujourd'hui 30
3. Les distributions 31
   - 3.1 Qu'est-ce qu'une distribution Linux ? 31
   - 3.2 Debian 32
   - 3.3 Ubuntu 33
   - 3.4 Les distributions de type Red Hat 34
   - 3.5 openSUSE 36
   - 3.6 Les autres distributions 36
   - 3.7 Tester une distribution : LiveCD, LiveDVD ou LiveUSB 37
   - 3.8 Distribution de secours 37
4. Quel matériel pour Linux ? 37
   - 4.1 L'architecture 37
   - 4.2 Configuration matérielle de base 38
   - 4.3 Compatibilité du matériel 39
5. Obtenir des informations et de l'aide concernant Linux 40

Chapitre 2
Installation de Linux et des logiciels

1. Installer une distribution. 41
   1.1 Déterminer les caractéristiques d'installation 41
   1.2 Paramètres d'installation 42
   1.3 Procédure d'installation 42
   1.4 Partitionnement des disques 43
   1.5 Configuration des interfaces réseau 44
   1.6 Sélection des paquets logiciels 44
   1.7 Redémarrage 44
2. Les gestionnaires de paquets logiciels 44
   2.1 Notion de paquet logiciel (package) 45
3. Les paquets logiciels Red Hat 46
   3.1 Le gestionnaire RPM 46
   3.2 Installation, mise à jour et suppression 46
   3.3 Cas du noyau 48
   3.4 Requêtes RPM 48
   3.5 Vérification des paquets logiciels 50
   3.6 Les dépendances 51
   3.7 Extraction du contenu cpio d'un paquet logiciel 51
   3.8 Mises à jour automatisées 52
4. Le gestionnaire de paquets YUM 52
   4.1 Configuration des dépôts 53
   4.2 Utilisation des dépôts 55
      4.2.1 Rafraîchir le cache 55
      4.2.2 Lister les paquets logiciels 55
      4.2.3 Installer des paquets logiciels 57
      4.2.4 Mises à jour 57
      4.2.5 Rechercher un paquet logiciel 58
      4.2.6 Désinstaller un paquet logiciel 59
      4.2.7 Télécharger un fichier paquet logiciel. 59
5. Le gestionnaire de paquets DNF 60

6. Les paquets logiciels Debian 60
6.1 dpkg : la commande de gestion de paquets Debian. 60
6.2 Installation, mise à jour et suppression de paquets logiciels 61
6.3 Requêtes de recherche et sélection de paquets 62
6.3.1 Lister les paquets 62
6.3.2 Trouver un paquet contenant un fichier 63
6.3.3 Lister le contenu d'un paquet 63
6.4 Reconfigurer un paquet logiciel 64
7. Le gestionnaire de paquets APT 64
7.1 Les dépôts de paquets logiciels 65
7.1.1 Configuration 65
7.1.2 Mise à jour de la base 65
7.2 Mise à jour de la distribution 66
7.3 Rechercher et installer un paquet logiciel individuel. 67
7.4 Client graphique 67
8. Le gestionnaire de paquets aptitude. 68
8.1 apt ou aptitude ? 68
8.2 Installation d'aptitude. 69
8.3 Utilisation 69
9. Le gestionnaire de paquets logiciels Zypper. 70
9.1 Gestion des dépôts. 70
9.2 Gérer les paquets logiciels 71
10. Gérer les bibliothèques partagées 73
10.1 Lieu de stockage. 74
10.2 Identifier les bibliothèques liées à un pogramme 75
10.3 Configurer le cache de l'éditeur de liens. 75
10.4 Recherche des bibliothèques partagées 75

Chapitre 3
Le shell et les commandes GNU

1. Le shell bash 77
   1.1 Rôle du shell 77
   1.2 Bash : le shell Linux par défaut 78
      1.2.1 Un shell puissant et libre 78
      1.2.2 L'invite de commandes 79
   1.3 Utiliser le shell 79
      1.3.1 La saisie sur la ligne de commande 79
      1.3.2 Syntaxe générale des commandes 80
      1.3.3 Exemple de commande : cal 80
      1.3.4 Enchaîner les commandes 82
      1.3.5 Afficher du texte 82
      1.3.6 Commandes internes et externes 83
      1.3.7 Séquences de contrôle 84
   1.4 Historique des commandes 84
2. La gestion des fichiers 85
   2.1 Le système de fichiers 85
   2.2 Les différents types de fichiers 86
      2.2.1 Les fichiers ordinaires ou réguliers 86
      2.2.2 Les répertoires 87
      2.2.3 Les fichiers spéciaux 87
   2.3 Nommage des fichiers 88
   2.4 Chemins d'accès 88
      2.4.1 Structure d'un chemin d'accès 88
      2.4.2 Chemin d'accès absolu 89
      2.4.3 Répertoire de connexion et répertoire courant 89
      2.4.4 Chemin d'accès relatif 89
      2.4.5 Le caractère tilde 91
      2.4.6 Changer de répertoire courant 91
   2.5 Les commandes de base 92
      2.5.1 Aide pour la syntaxe des commandes 92
      2.5.2 Lister les fichiers et les répertoires 93
      2.5.3 Gérer les fichiers et les répertoires 95
      2.5.4 Les caractères génériques 102

3. Rechercher des fichiers avec la commande find 105
3.1 Critères de recherche 106
3.1.1 Recherche par nom 106
3.1.2 Recherche par type 107
3.1.3 Recherche par propriétaire ou groupe associé 107
3.1.4 Recherche par taille 108
3.1.5 Recherche par date 109
3.1.6 Recherche par permissions d'accès 109
3.2 Commandes exécutées avec les fichiers recherchés 111
3.2.1 Recherche avec liste détaillée 111
3.2.2 Recherche avec exécution d'une commande 112
3.3 Combinaison logique de critères 113
3.4 Rechercher des informations sur une commande 113
3.4.1 whereis 113
3.4.2 which 114
3.4.3 locate 114
4. L'éditeur vi 114
4.1 Présentation 115
4.2 Fonctionnement 115
4.3 Les commandes de base de vi 115
4.3.1 Passer en mode saisie 115
4.3.2 Ouvrir la ligne de commande de vi 116
4.3.3 Quitter l'éditeur 116
4.3.4 Déplacement dans le fichier 116
4.3.5 Modification de texte 117
4.3.6 Expressions régulières 118
4.3.7 Recherche dans le texte 119
4.3.8 Remplacement de texte 119
4.3.9 Copier-coller 120
4.3.10 Substitution 120
4.3.11 Autres commandes de vi 121
5. Les redirections des entrées/sorties standards 122
5.1 Les entrées/sorties standards 122
5.2 Les entrées/sorties standards par défaut 122

6. La redirection 123
6.1 Redirection de la sortie standard. 123
6.2 Redirection de la sortie d'erreur standard. 125
6.3 Redirection de la sortie et de la sortie d'erreur standards dans un même fichier 126
6.4 Redirection de l'entrée standard 127
6.5 Documents en ligne 128
6.6 Ouverture de descripteurs de fichiers supplémentaires. 128
6.7 Fermeture de descripteurs de fichiers 129
6.8 Les tubes (pipes) 129
7. Les commandes filtres 130
7.1 Compter des lignes, des mots, des caractères. 130
7.2 Sélection de lignes 131
7.2.1 grep 131
7.2.2 egrep 132
7.2.3 fgrep 133
7.2.4 sed 133
7.3 Sélection de parties de lignes. 135
7.3.1 Sélection par position 135
7.3.2 Sélection par champs 136
7.4 Tri de lignes 137
7.5 Suppression des doublons 139
7.6 Jointure de deux fichiers triés 140
7.6.1 Concaténation de fichiers ligne à ligne. 140
7.7 Découpage d'un fichier en plusieurs fichiers 141
7.8 Substitution et suppression de caractères dans un fichier 142
7.8.1 La commande tr 142
7.9 xargs 144
7.10 Affichage de texte 144
7.10.1 Affichage page par page. 144
7.10.2 Affichage des premières lignes d'un fichier 146
7.10.3 Affichage des dernières lignes d'un fichier 146
7.10.4 Formater l'affichage 147
7.11 Duplication de la sortie standard 147

8. Autres commandes utilitaires 148
8.1 Extraction d'une partie d'un chemin d'accès 148
8.2 Comparaison de fichiers 148
8.2.1 diff 148
8.2.2 cmp 150
8.3 Mise en attente 150
8.4 Les sommes de contrôle 151
9. La gestion des processus 152
9.1 Attributs d'un processus 152
9.2 États d'un processus 153
9.3 Lancement d'une commande en tâche de fond 153
9.4 Tâches en avant-plan et en arrière-plan 154
9.5 Liste des processus 155
9.6 Envoi d'un signal à un processus 157
9.7 nohup 158
9.8 nice et renice 159
9.9 time 160
9.10 exec 161
10. Plus loin avec le bash 161
10.1 Les alias 161
10.2 Groupement de commandes 162
10.3 Liaison conditionnelle 163
11. Les variables 164
11.1 Nom de variable 164
11.2 Déclaration et affectation 164
11.3 Accès et affichage 165
11.4 Suppression de variable et protection en écriture 166
11.5 Export 167
11.6 Délimitation du nom de variable 168
11.7 Remplacement conditionnel de variable 168
11.8 Variables système 169
11.9 Variables spéciales 170
11.10 Longueur d'une chaîne 171
11.11 Tableaux et champs 171
11.12 Variables numériques et calcul 172

12. Configuration de bash 173
12.1 Fichiers de configuration 173
12.1.1 Shell de connexion 173
12.1.2 Shell simple 173
12.2 Configuration du shell par la commande set 174
13. Programmation shell 174
13.1 Structure et exécution d'un script shell 174
13.2 Arguments d'un script 177
13.2.1 Paramètres de position 177
13.2.2 Redéfinition des arguments 178
13.2.3 Parcours des arguments 178
13.2.4 Terminaison de script 179
13.3 Environnement d'un processus 179
13.4 Substitution de commande 180
13.5 Tests logiques 180
13.5.1 Tests sur une chaîne 181
13.5.2 Tests sur les valeurs numériques 182
13.5.3 Tests sur les fichiers 182
13.5.4 Tests combinés par des opérateurs logiques 183
13.5.5 Nouvelle syntaxe 183
13.6 Structures de contrôle conditionnelles 184
13.6.1 if ... then ... else 184
13.6.2 Structure de choix multiples 185
13.6.3 Saisie utilisateur 187
13.6.4 Les boucles 188
13.6.5 Les fonctions 193
13.6.6 Calculs et expressions 194
13.6.7 Traitement des signaux 196
14. Multiplexeurs de terminal 197
14.1 Utilisation 198
14.1.1 Installation et aide 198
14.1.2 Fenêtres 198
14.1.3 Détacher et rattacher 198
14.1.4 Terminer la session 199
14.2 Autres multiplexeurs 199

Chapitre 4
Les disques et le système de fichiers

1. Représentation des disques 201
   - 1.1 Nomenclature 201
     - 1.1.1 Disques IDE 201
     - 1.1.2 Disques SCSI, SATA, USB, FIREWIRE, etc. 202
   - 1.2 Cas particuliers 203
     - 1.2.1 Virtualisation 203
     - 1.2.2 SAN, iSCSI, multipathing 203
2. Opérations de bas niveau 204
3. Choisir un système de fichiers 205
   - 3.1 Principe 205
     - 3.1.1 Représentation 205
     - 3.1.2 Les métadonnées 206
     - 3.1.3 Les noms des fichiers : les liens physiques 206
     - 3.1.4 Les systèmes de fichiers journalisés 206
   - 3.2 Les types de systèmes de fichiers sous Linux 207
     - 3.2.1 Systèmes de fichiers de type ext* 207
     - 3.2.2 Systèmes de fichiers de type XFS 208
     - 3.2.3 Systèmes de fichiers de type BTRFS 209
     - 3.2.4 VFAT (FAT32) 210
     - 3.2.5 exFAT 210
     - 3.2.6 FUSE 211
4. Partitionnement 211
   - 4.1 Les méthodes de partionnement 211
   - 4.2 Partitionnement MBR (Master Boot Record) 212
     - 4.2.1 MBR et BIOS 212
     - 4.2.2 MBR 212
     - 4.2.3 Les partitions 212
     - 4.2.4 Types de partitions 213
   - 4.3 Partitionnement GPT 214
     - 4.3.1 GPT et UEFI 214
     - 4.3.2 GUID 215
     - 4.3.3 LBA 0 215
     - 4.3.4 LBA 1 216
     - 4.3.5 LBA 2 à 33 216

4.3.6 Types de partitions 217
4.3.7 UEFI Boot manager 217
4.3.8 La partition système EFI 218
4.4 Manipuler les partitions 219
4.4.1 Manipuler les partitions MBR 219
4.4.2 Manipuler les partitions GPT 223
5. Manipuler les systèmes de fichiers 225
5.1 Définitions de base. 225
5.1.1 Bloc 225
5.1.2 Superbloc 225
5.1.3 Table d'inodes. 226
5.1.4 Les répertoires 227
5.1.5 Lien physique 228
5.2 Créer un système de fichiers 229
5.2.1 Création d'un système de fichiers ext* 231
5.2.2 Création d'un système de fichiers XFS. 233
5.2.3 Création d'un système de fichiers BTRFS 233
5.2.4 Création d'un système de fichiers VFAT 235
6. Accéder aux systèmes de fichiers 236
6.1 La commande mount. 236
6.1.1 Options de montage 239
6.1.2 umount. 240
6.1.3 Remonter un système de fichiers 241
6.1.4 Le fichier /etc/fstab 241
6.1.5 Systèmes de fichiers de CD/DVD et images ISO 243
7. Contrôler le système de fichiers 244
7.1 Suivi de l'espace disque par système de fichiers. 244
7.1.1 Suivi de l'espace disque par arborescence. 245
7.2 Vérifier et réparer les systèmes de fichiers 245
7.2.1 fsck 245
7.2.2 badblocks 246
7.2.3 dumpe2fs 247
7.2.4 tune2fs 247

7.3 XFS 249
7.3.1 xfs_info 249
7.3.2 xfs_growfs 249
7.3.3 xfs_repair 249
7.3.4 xfs_db et xfs_admin 251
7.3.5 xfs_fsr 251
8. Le swap 251
8.1 Taille optimale de l'espace de swap 252
8.2 Créer une partition de swap 252
8.3 Activer et désactiver le swap 253
8.3.1 Activation/désactivation dynamique 253
8.3.2 Déclaration des zones de swap dans /etc/fstab 253
8.4 Zone de swap dans un fichier 254
8.5 État de la mémoire 254
8.5.1 free 254
8.5.2 /proc/meminfo 255
9. Les droits d'accès 256
9.1 Les droits de base 256
9.1.1 Droits et compte utilisateur 256
9.1.2 Droits d'accès 257
9.2 Représentation des droits d'accès 258
9.2.1 Notation symbolique 258
9.2.2 Notation octale 259
9.3 Modification des droits 259
9.3.1 Notation symbolique 260
9.3.2 Notation octale 260
9.4 Masque des droits par défaut 261
9.5 Changer de propriétaire et de groupe propriétaire 262
9.6 Droits d'accès étendus 263
9.6.1 SetUID et SetGID 263
9.6.2 Sticky bit sur un répertoire 264
9.6.3 SetGID sur un répertoire 265

Chapitre 5
Boot, services, noyau et périphériques

1. Processus de démarrage 267
   1.1 Le BIOS et l'UEFI 267
      1.1.1 BIOS 267
      1.1.2 UEFI 268
      1.1.3 Choix du périphérique de démarrage 269
   1.2 Le chargeur de démarrage 269
   1.3 GRUB 270
      1.3.1 Configuration de GRUB legacy 270
      1.3.2 Installation 271
      1.3.3 Démarrage et édition d'un choix de menu 272
   1.4 GRUB2 272
      1.4.1 Configuration 273
      1.4.2 Démarrage et édition 276
      1.4.3 Cas de GPT et UEFI 276
   1.5 Initialisation du noyau 278
2. init System V 279
   2.1 Rôle d'init 279
   2.2 Niveaux d'exécution 280
   2.3 /etc/inittab 281
   2.4 Changement de niveau d'exécution 283
   2.5 Paramétrage système de base 283
   2.6 Niveaux d'exécution 284
   2.7 Gestion des niveaux et des services 284
      2.7.1 Services dans init.d 284
      2.7.2 Contrôle des services 286
      2.7.3 Modification des niveaux d'exécution 287
   2.8 Consoles virtuelles 289
   2.9 La procédure de connexion (login) 290
   2.10 Arrêt du système 290
3. systemd 291
   3.1 Unités cibles et services 292
   3.2 Configuration 292

3.3 Cibles 293
3.3.1 Équivalence avec init System V 293
3.3.2 Cible par défaut 293
3.3.3 Changer de cible par défaut 294
3.3.4 Passer d'une cible à l'autre 294
3.3.5 Mode secours et urgence 294
3.3.6 Cibles actives et dépendances 294
3.3.7 Lister toutes les cibles 296
3.4 Services 296
3.4.1 Actions 296
3.4.2 Statut 297
3.4.3 Activation 298
3.4.4 Dépendances 298
3.5 Actions système 300
3.6 Gestion de la console 300
4. upstart 301
4.1 Configuration 301
4.2 Niveau par défaut 302
4.3 Compatibilité System V 302
4.4 Commandes de contrôle 302
5. Consulter les traces du système 304
5.1 dmesg 304
5.2 /var/log/messages ou /var/log/syslog 305
5.3 journalctl 305
6. Le noyau et ses modules 306
6.1 uname 307
6.2 Gestion des modules 308
6.2.1 lsmod 309
6.2.2 modinfo 310
6.2.3 insmod 311
6.2.4 rmmod 312
6.2.5 modprobe 312
6.2.6 modprobe.d 313
6.3 Paramètres dynamiques 314

7. Les fichiers associés aux périphériques 317
7.1 Fichiers spéciaux 318
7.2 Créer un fichier spécial 319
7.3 Déterminer les composants matériels du système. 319
7.3.1 Bus PCI 319
7.3.2 Bus USB 320
7.3.3 Systèmes de fichiers virtuels 321
7.3.4 udev 323

Chapitre 6
Les tâches administratives

1. Administration des utilisateurs. 325
1.1 Les utilisateurs 325
1.2 Les groupes 327
1.3 Les mots de passe 328
1.4 Les fichiers de configuration des utilisateurs et des groupes. 328
1.4.1 /etc/passwd 328
1.4.2 /etc/group. 329
1.4.3 /etc/shadow 329
1.4.4 /etc/gshadow 330
1.5 Gestion des utilisateurs 331
1.5.1 Création d'un compte utilisateur 331
1.5.2 Gestion des mots de passe. 333
1.5.3 Modification d'un compte utilisateur 336
1.5.4 Suppression d'un compte utilisateur 337
1.6 Gestion des groupes d'utilisateurs 337
1.6.1 Modification d'un groupe d'utilisateurs. 338
1.6.2 Suppression d'un groupe d'utilisateurs 338
1.7 Commandes additionnelles 338
1.7.1 Vérifier la cohérence des fichiers de configuration 338
1.7.2 Vérifier l'historique des connexions 339
1.7.3 Modifications par l'utilisateur. 340
1.7.4 Interroger les annuaires 344
1.8 Configuration par défaut des comptes utilisateurs 344

1.9 Notifications à l'utilisateur 346
1.9.1 /etc/issue 346
1.9.2 /etc/motd 346
1.9.3 Envoi de messages écran aux utilisateurs 347
1.10 L'environnement utilisateur 347
1.10.1 Le répertoire /etc/skel 347
1.10.2 Scripts de configuration 348
1.11 Les modules PAM 349
2. L'impression 350
2.1 Principe 351
2.2 Le système d'impression LPD BSD 351
2.3 CUPS 352
2.3.1 Ajout d'une imprimante 354
3. Automatisation des tâches 359
3.1 Le service cron 359
3.1.1 Format d'une ligne de tâche crontab 359
3.1.2 La crontab système 360
3.1.3 Contrôle d'accès au service cron 362
3.2 La commande at 362
3.2.1 Format de spécification de la tâche différée 363
3.2.2 Contrôle des tâches 364
3.2.3 Contrôle d'accès à la commande at 365
3.3 Les timers systemd 365
4. Les fichiers journaux du système 369
4.1 Les messages 370
4.2 Configuration de rsyslog 370
4.3 Le service journald de systemd 373
4.4 Les fichiers journaux 376
4.5 La commande journalctl 376
4.6 Émettre des messages vers journald 378
4.7 Rotation des fichiers journaux 378
4.7.1 logrotate 378
4.7.2 journald 380

5. Sauvegarde et restauration 381
5.1 La commande tar 381
5.1.1 Archivage 381
5.1.2 Lister le contenu d'une archive 382
5.1.3 Restauration 383
5.1.4 Compression des fichiers d'archive 384
5.2 La commande cpio 384
5.2.1 Archivage 385
5.2.2 Lister le contenu d'une archive 385
5.2.3 Restauration 386
5.3 la commande dd 386
6. Gestion de la date et heure système 387
6.1 La commande date 387
6.2 Utiliser le protocole NTP 390
6.2.1 Client NTP 390
6.2.2 Dérive temporelle 392
6.3 timedatectl 392
6.4 chrony 393
7. Les paramètres régionaux 395
7.1 L'internationalisation (i18n) et la régionalisation (l10n) 395
7.2 Réglages régionaux 396
7.2.1 Variables d'environnement 396
7.2.2 Fuseaux horaires 399
7.3 Codage des caractères 400

Chapitre 7
Le réseau

1. TCP/IP 403
1.1 L'adressage IPv4 404
1.1.1 Sous-réseaux 405
1.1.2 Routage 406
1.1.3 IPv6 407

1.2 Configuration de base du réseau 408
1.2.1 Nommage des interfaces 408
1.2.2 NetworkManager 409
1.3 Commandes de configuration 409
1.3.1 Anciennes versions des distributions de type Red Hat 410
1.3.2 Anciennes versions des distributions de type Debian 412
1.3.3 Routage 413
1.3.4 La commande ip 414
1.3.5 Configuration avec NetworkManager 416
1.3.6 Les numéros de ports 419
1.4 Outils réseau 421
1.4.1 La commande ping 421
1.4.2 La commande traceroute 422
1.4.3 La commande tracepath 423
1.4.4 La commande whois 424
1.4.5 La commande nc (netcat) 425
1.4.6 La commande netstat 426
1.4.7 La commande ss 428
1.4.8 La commande IPTraf 429
1.5 Les fichiers de configuration 430
1.5.1 /etc/resolv.conf 430
1.5.2 /etc/hosts et /etc/networks 432
1.5.3 /etc/nsswitch.conf 432
1.5.4 /etc/services 433
1.5.5 /etc/protocols 434
1.6 Contrôle de la résolution de noms 434
1.6.1 La commande dig 434
1.6.2 La commande host 435
1.6.3 La commande getent 436
2. Services réseau xinetd 436
2.1 Configuration 437
2.2 Démarrage et arrêt des services 439
3. OpenSSH 440
3.1 Configuration du serveur ssh 440
3.2 Utilisation de ssh 441

3.3 Clés et connexion automatique 441
3.3.1 Type de chiffrement 441
3.3.2 Exemple de configuration côté client 442
3.3.3 Côté serveur 443
3.3.4 Copie automatique de la clé publique 443
3.4 Passphrase et agent SSH 444
3.5 Authentification de l'hôte 445
4. Courrier électronique 446
4.1 postfix 447
4.1.1 Alias d'utilisateurs 447
4.1.2 exim 448
4.1.3 qmail 448

Chapitre 8
La sécurité

1. Les bases de la sécurité 449
1.1 Contrôler les droits d'endossement SUID et SGID 450
1.2 Vérifier les paquets logiciels 451
1.3 Politique de mot de passe 452
1.4 Interdire les connexions 453
1.4.1 Shell de connexion /bin/false ou /sbin/nologin 453
1.4.2 /etc/nologin 453
1.4.3 /etc/securetty 454
1.5 Limiter les ressources pour un compte utilisateur 454
1.6 Les droits SUDO 455
2. Sécurité des services et du réseau 458
2.1 Vérifier les ports ouverts 458
2.1.1 Informations depuis netstat 458
2.1.2 L'outil nmap 459
2.2 Désactiver les services inutiles 460
2.2.1 Services autonomes 460
2.2.2 Services xinetd 461
2.3 Les TCP wrappers 461

2.4 GPG 463
2.4.1 Générer les clés 464
2.4.2 Générer une clé de révocation 465
2.4.3 Gérer le trousseau 466
2.4.4 Exporter la clé publique 467
2.4.5 Importer une clé 469
2.4.6 Signer une clé 469
2.4.7 Signer et chiffrer un message 471

Chapitre 9
Interfaces graphiques

1. Comment fonctionne un environnement graphique ? 475
1.1 Le système X Window 475
1.1.1 Le gestionnaire de fenêtres 477
1.1.2 Les widgets et les toolkits 478
1.1.3 Les bureaux virtuels 479
1.2 Les environnements de bureau 479
2. Wayland 480
3. Xorg 481
3.1 Installation 482
3.2 Configuration de Xorg 483
3.2.1 Via la distribution 483
3.2.2 Xorgcfg 484
3.2.3 Xorgconfig 485
3.2.4 X 485
3.3 Structure de xorg.conf 485
3.3.1 Sections et sous-sections 485
3.3.2 Valeurs booléennes 486
3.3.3 Section InputDevice ou InputClass 486
3.3.4 Section Monitor 487
3.3.5 Section Device 487
3.3.6 Section Screen 488
3.3.7 Section ServerLayout 489
3.3.8 Section Files 489
3.3.9 Section Modules 490

3.3.10 Section ServerFlags 490
3.3.11 xorg.conf.d 491
3.4 Tester et lancer X 491
3.4.1 Vérifier la configuration 491
3.4.2 Les fichiers journaux 492
3.4.3 Tester le serveur 493
4. Le gestionnaire d'affichage (Display Manager) 494
4.1 XDM 495
4.1.1 Setup : Xsetup 496
4.1.2 Chooser : RunChooser 498
4.1.3 Startup : Xstartup 498
4.1.4 Session : Xsession 498
4.1.5 Reset : Xreset 499
4.1.6 Resources : Xresources 500
4.1.7 Servers : Xservers 500
4.1.8 AccessFile : Xaccess et XDMCP 500
4.2 GDM et KDM 501
4.3 Gestionnaire d'affichage au démarrage 503
4.3.1 System V et inittab 503
4.3.2 System V et services 503
4.3.3 Cible systemd 503
5. Gestionnaire de fenêtres et l'environnement personnel 504
5.1 Via le gestionnaire d'affichage 504
5.2 startx 505
5.3 Les terminaux en mode graphique 505
5.4 Les gestionnaires de fenêtres 507
5.5 Exporter ses fenêtres 509
6. Bureau distant 511
6.1 RDP 511
6.2 VNC 513
6.3 Spice 514
7. Accessibilité 515
7.1 Assistance au clavier et à la souris 515
7.2 Assistance visuelle et auditive 517

Chapitre 10
Machines virtuelles, containers et Cloud

1. La virtualisation 519
   - 1.1 Le Cloud 520
   - 1.2 Intérêt 520
   - 1.3 Compétence 522
   - 1.4 Choix de la solution 522
2. Méthodes de virtualisation 522
   - 2.1 L'isolation 522
   - 2.2 Noyau en espace utilisateur 524
   - 2.3 Hyperviseur de type 2 524
   - 2.4 Hyperviseur de type 1 525
   - 2.5 Virtualisation matérielle 526
3. Paravirtualisation 526
   - 3.1 Principe 526
   - 3.2 VirtIO 527
   - 3.3 Virtualisation de la mémoire 527
   - 3.4 Virtualisation des périphériques 528
   - 3.5 Sécurité 529
   - 3.6 Configuration particulière 529
4. Les containers 530
   - 4.1 Principe 530
   - 4.2 Container et machine virtuelle 531
   - 4.3 Les espaces de nommage 532
   - 4.4 Les groupes de contrôle 533
   - 4.5 Docker 533
   - 4.6 Un exemple complet 534
     - 4.6.1 Créer une image 535
     - 4.6.2 Démarrer un container 536
     - 4.6.3 Arrêt du container 537
     - 4.6.4 Exposition du container 537
     - 4.6.5 Fichiers journaux du container 538
     - 4.6.6 Supprimer le container et l'image 538
   - 4.7 Sécurité 539

5. Le Cloud 539
5.1 Services Cloud 540
5.2 Fournisseurs 541
5.3 Exemple d'AWS 541
5.4 Zones géographiques 542
5.5 Tester 543
5.6 Cloud-init 547
6. Systèmes invités 548
6.1 Hyperviseur et additions 548
6.2 L'accès à la console ou l'affichage 550
6.2.1 Spice et KVM 550
6.2.2 Client Spice 551
6.2.3 Autres cas 552

Index 553

Les éléments à télécharger sont disponibles à l'adresse suivante :
http://www.editions-eni.fr
Saisissez la référence de l'ouvrage **EI3KUB** dans la zone de recherche et validez. Cliquez sur le titre du livre puis sur le bouton de téléchargement.

Avant-propos

1. Présentation de Kubernetes 43
1.1 Un peu d'histoire 43
1.2 Qu'est-ce qu'un conteneur ? 44
1.3 Les conteneurs avant Docker 44
1.4 Pourquoi utiliser des conteneurs ? 45
1.5 Problèmes introduits avec les conteneurs 46
1.6 À quoi va servir Kubernetes ? 46
1.7 Ressources externes 47
2. Un mot sur l'application 47
2.1 Rien ne sert de courir 47
2.2 Les douze facteurs applicatifs 48
2.3 Microservices vs Monolithes 48

Chapitre 1
Introduction

1. Cibles et objectifs de l'ouvrage 49
2. Prérequis techniques et ressources documentaires 50
2.1 Prérequis techniques 50
2.2 Ressources documentaires 50
2.3 Récupération des fichiers d'exemples 51
3. Présentation générale 51
3.1 Prérequis 51
3.2 Utilisation de Kubernetes 52
3.3 Installation et configuration de Kubernetes 52
3.4 Extension du cluster Kubernetes et notions avancées 53

3.5 Déploiement et intégration continue 53
3.6 Conventions utilisées 54

Chapitre 2
Installation de l'environnement Kubernetes

1. Objectifs du chapitre et prérequis 55
2. Alternative à l'installation en local 56
   2.1 Pourquoi ces alternatives ? 56
   2.2 Utilisation d'un service managé 56
   2.3 Service Killercoda 57
3. Mise en place de la commande kubectl 58
   3.1 À quoi sert kubectl ? 58
   3.2 Installation de kubectl 58
      3.2.1 Installation sous Debian/Ubuntu 58
      3.2.2 Installation sous CentOS/RHEL ou Fedora 59
      3.2.3 Installation à l'aide d'Arkade 59
   3.3 Vérification de l'installation 62
   3.4 Configuration de l'autocomplétion 62
      3.4.1 Présentation du mécanisme d'autocomplétion 62
      3.4.2 Fichier profile à modifier 62
      3.4.3 Autocomplétion sur kubectl 63
      3.4.4 Utilisation de la variable SHELL 63
      3.4.5 Mise en place d'un alias 64
4. Mise en place de Minikube 64
   4.1 Pourquoi faire appel à Minikube ? 64
   4.2 Téléchargement et installation de Minikube 65
   4.3 Vérification de l'installation de Minikube 65
   4.4 Mise en place de l'autocomplétion 66

5. Installation du cluster Kubernetes avec Minikube 66
5.1 Options de lancement 66
5.2 Installation de Docker 67
5.2.1 Installation de Docker Community Edition sur Ubuntu 67
5.2.2 Installation alternative 68
5.2.3 Configuration des accès à Docker 68
5.2.4 Vérification de l'installation de Docker 68
5.3 Installation de l'hyperviseur VirtualBox 69
5.4 Installation de l'hyperviseur KVM/libvirt 70
5.5 Configuration de l'utilisateur courant 70
5.6 Déploiement du cluster avec Minikube 71
5.6.1 Création de la machine Minikube 71
5.6.2 Arrêt/démarrage de la machine Minikube 75
5.6.3 Choix du conteneur runtime 76
5.6.4 Extensions de Minikube 76
5.6.5 Suppression de la machine Minikube 78
6. Quelques notions sur le format YAML 79
6.1 Déclaration de couples clés/valeurs 79
6.2 Les tableaux en YAML 80
6.3 Les structures clé/valeur ou table de hachage 81
6.4 Tableau de table de hachage 83

Chapitre 3
Tableau de bord et ligne de commande

1. Objectifs du chapitre et prérequis 85
2. Préambule 86
2.1 Origine du nom et du logo 86
2.2 Pourquoi utiliser Kubernetes ? 86
2.3 Origine de Kubernetes 87
2.4 Fondation CNCF 87
2.5 Les orchestrateurs du marché 88

3. Le tableau de bord de Kubernetes (dashboard) 89
3.1 Présentation 89
3.2 Tableau de bord Kubernetes sur service managé 89
3.3 Déploiement du dashboard sur Minikube 89
3.4 Accès au dashboard sur Minikube 90
3.5 Structure du tableau de bord 91
3.6 Création d'un déploiement 92
3.6.1 Un petit mot sur Mailpit 92
3.6.2 Lancement du déploiement 93
3.7 État d'un déploiement 95
3.7.1 Consultation de l'état du déploiement 95
3.7.2 Consultation du gestionnaire de réplicas 96
3.7.3 Consultation de l'état d'un pod 98
3.7.4 Journal d'activité du conteneur 99
3.7.5 Scalabilité 99
3.7.6 Mise à jour de l'application 101
3.7.7 Pour résumer 103
4. Présentation de l'outil kubectl 103
4.1 Préambule 103
4.2 Consultation des éléments 103
4.3 Liste des pods 105
4.4 Liste des machines d'un cluster 106
4.4.1 Connexion à la machine Minikube 106
4.4.2 Liste des nœuds d'un cluster 106
4.4.3 Affichage des caractéristiques étendues 107
5. Le moteur Containerd de Minikube 108
5.1 Initialisation de l'environnement 108
5.2 Les conteneurs associés aux pods 109

Chapitre 4
Automatisation et publication d'une application

1. Objectifs du chapitre et prérequis 111
2. Gestion par kubectl d'une application 111
   2.1 Suppression d'un déploiement 111
   2.2 Création d'un déploiement 112
   2.3 État du déploiement 113
   2.4 Mécanisme des réplicas 115
      2.4.1 Consultation des réplicas 115
      2.4.2 Description des réplicas 116
   2.5 État du pod 117
      2.5.1 Liste des pods 117
      2.5.2 Détails de l'état d'un pod 118
   2.6 Accès aux logs des conteneurs 119
   2.7 Accès à l'application Mailpit 120
3. Exposition de services 121
   3.1 Pourquoi utiliser un service ? 121
   3.2 Exposition d'un déploiement via un service 122
   3.3 Vérification du service mailpit 122
   3.4 Que faire en cas d'absence de shell ? 124
      3.4.1 Contexte 124
      3.4.2 Utilisation d'un pod éphémère 125
      3.4.3 Lancement d'un pod de test 126
   3.5 Résilience et scalabilité 127
      3.5.1 Origine du besoin 127
      3.5.2 Scalabilité manuelle 128
      3.5.3 Nombre de pods associés à un déploiement 128
      3.5.4 Arrêter temporairement une application 128

4. Automatisation de déploiement par fichier YAML 130
4.1 Mécanisme de création et mise à jour 130
4.2 Structure YAML d'un déploiement 130
4.2.1 Quelques rappels 130
4.2.2 Récupération d'une structure au format YAML 131
4.2.3 Édition d'un déploiement 132
4.2.4 Squelette pour un déploiement 133
4.2.5 Création d'un déploiement à l'aide d'un fichier 135
4.2.6 Suppression des éléments d'un fichier 136
4.2.7 Gestion de l'idempotence et de la réentrance 136
4.3 Création du service 138
4.3.1 Définition du service 138
4.3.2 Application de la définition du service 139
4.3.3 Gestion de la réentrance 139
4.4 Mécanisme de sélecteur et labels 140
4.5 Regroupement de la création des éléments 142
4.5.1 Création d'un groupe d'objets 142
4.5.2 Consultation de l'état d'un groupe d'objets 143
4.6 Structure des objets 143
4.6.1 Interrogation de Kubernetes avec kubectl 143
4.6.2 Référence de l'API en ligne 144

5. Ingress et reverse proxy 145
5.1 Origine du besoin 145
5.2 Rôle d'un proxy inverse 145
5.3 Activation du contrôleur Ingress dans Minikube 146
5.4 Déclaration d'une règle Ingress 147
5.5 Consultation des règles Ingress 148
5.6 Hôte virtuel et nip.io 150
5.6.1 Hôte virtuel par défaut 150
5.6.2 Présentation du mécanisme de nip.io 150
5.6.3 Configuration du serveur DNS 151
5.6.4 Création d'un hôte virtuel pour Mailpit 153

Chapitre 5
Cycle de vie d'un conteneur dans Kubernetes

1. Objectifs du chapitre et prérequis 157
2. Gestion des crashs d'application 158
   2.1 Consultation de l'état des pods 158
   2.2 Connexion au pod 158
   2.3 Conteneur associé à Mailpit 159
   2.4 Comportement en cas de crash 161
   2.5 État du conteneur après redémarrage du pod 162
   2.6 Container vu depuis Containerd (Minikube) 162
   2.7 Attention au nettoyage 164
3. État d'un conteneur 164
   3.1 Pourquoi scruter l'état d'un conteneur ? 164
   3.2 Readiness vs Liveness 165
   3.3 Utilisation et bonne pratique 166
   3.4 Structure des champs de surveillance 166
   3.5 Vérification de la présence d'un port 168
      3.5.1 Définition de la surveillance 168
      3.5.2 Test d'indisponibilité sur un pod non prêt 169
      3.5.3 État des pods en cas d'indisponibilité 170
      3.5.4 Test d'indisponibilité sur un pod en mauvaise santé . . . 170
      3.5.5 État des pods en cas de problème sur un pod 171
      3.5.6 Attention à la consistance des tests 171
      3.5.7 Uniformisation des tests 173
   3.6 Surveillance HTTP 174
      3.6.1 Pourquoi privilégier ce type de surveillance ? 174
      3.6.2 Surveillance de l'application Mailpit 174
   3.7 Point d'entrée de surveillance HTTP d'une application 175
      3.7.1 Un mot sur les frameworks modernes 175
      3.7.2 Présentation de l'application Flask 175
      3.7.3 Exemple de déclaration 176
      3.7.4 Déploiement de l'application Flask 177

3.7.5 Consultation de l'état de l'application 177
3.8 Lancement d'un shell 179
3.8.1 Principe de fonctionnement 179
3.8.2 Exemple de surveillance d'une base Postgres 179
3.8.3 Déclaration de la commande 180
4. Définition de la capacité d'un pod 180
4.1 Pourquoi définir une capacité ? 180
4.2 Réservation et surallocation 181
4.3 Allocation de ressources à un conteneur 181
4.4 Allocation de ressources à l'application Mailpit 182
4.5 Comportement en cas de saturation des ressources 183
4.5.1 Demande trop importante de CPU 183
4.5.2 Dépassement de la mémoire allouée 184
4.6 Priorité d'un pod 185
4.6.1 Présentation du mécanisme 185
4.6.2 Consultation des types par défaut 186
4.6.3 Consultation des priorités des pods 186
4.6.4 Création d'une classe de priorité 188
4.6.5 Affectation d'une classe de priorité personnalisée 189
4.6.6 Remarque sur les classes de priorité par défaut 190

Chapitre 6
Persistance des données

1. Objectifs du chapitre et prérequis 191
2. Persistance des données 191
2.1 Origine du besoin 191
2.2 Utilisation d'un volume persistant externe 192
2.3 Volumes persistants 193
2.3.1 Structure du volume persistant 193
2.3.2 Création du volume persistant 194

2.4 Persistance de données avec Mailpit 194
2.4.1 Opérations à réaliser 194
2.4.2 Déclaration de l'objet PersistentVolumeClaim 195
2.4.3 État des objets de volume persistant 196
2.4.4 État de la demande de volume persistant 197
2.4.5 Déclaration du point de montage 197
2.4.6 Ajout d'un point de montage sur le conteneur 198
2.4.7 Options de lancement de Mailpit 198
2.4.8 Déclaration entière suite aux modifications 199
2.5 Test de la persistance 201
2.5.1 Utilisation de Mailpit pour l'envoi d'e-mail 201
2.5.2 Ouverture de la communication avec le port SMTP . . . 202
2.5.3 Envoi d'un mail 202
2.5.4 Réduction des droits d'exécution et initialisation 203
2.5.5 Gestion des droits du répertoire de persistance 205
2.5.6 Consultation de l'interface de Mailpit 208
2.5.7 Suppression des pods 209
2.5.8 Vérification du fonctionnement de la persistance 210
3. Classes de stockage 210
3.1 Origine du besoin 210
3.2 Liste des classes de stockage 211
3.3 Détail d'une classe de stockage 212
3.4 Classe de stockage par défaut 212
3.5 Les différentes classes de stockage 213
3.5.1 Les différentes familles 213
3.5.2 Origine de ces familles 213
3.6 Caractéristiques des classes de stockage 214
3.6.1 Modes d'accès 214
3.6.2 Caractéristiques de certains pilotes 215
3.6.3 Liste des pilotes chargés 216
3.7 Déclaration d'une classe de stockage 217
3.7.1 Structure de la déclaration 217
3.7.2 Exemple de déclaration 218

3.8 Test de création automatique d'un volume persistant 219

Chapitre 7
Hébergement d'application en cluster

1. Objectifs du chapitre et prérequis 223
2. Déploiement d'une base de données MariaDB 223
   2.1 Origine du besoin 223
   2.2 Déploiement 224
      2.2.1 Choix de l'image Docker 224
      2.2.2 Version initiale du fichier de déploiement 224
      2.2.3 Gestion de la réentrance 225
   2.3 Volume persistant 226
      2.3.1 Demande de volume persistant 226
      2.3.2 État de la demande de volume persistant 226
      2.3.3 Ajout d'une persistance sur le conteneur de MariaDB . . 227
      2.3.4 Consultation de l'état du déploiement 228
   2.4 Configuration de la base de données 229
   2.5 Consultation de l'état du pod 230
      2.5.1 Liste des pods 230
      2.5.2 Connexion au conteneur 231
      2.5.3 Création de l'entrée de service 231
   2.6 Surveillance de la base de données 232
      2.6.1 Définition des commandes de surveillance 232
      2.6.2 Application de la modification 234
      2.6.3 Vérification du déploiement 234
   2.7 Mécanisme de déploiement 235
3. Mise en place d'un StatefulSet 237
   3.1 Augmentation du nombre de pods associés au déploiement . . 237
   3.2 Présentation du type StatefulSet 239
      3.2.1 Caractéristiques 239
      3.2.2 Limitations 239

3.3 Déclaration du premier objet StatefulSet 240
3.3.1 Purge de l'ancien déploiement 240
3.3.2 Modifications à réaliser 240
3.3.3 Création du StatefulSet 242
3.3.4 État des volumes persistants 242
3.3.5 Suppression des anciens objets PV/PVC 243
3.4 Scalabilité de l'objet StatefulSet 243
3.5 Pods et volumes persistants d'un objet StatefulSet 244
3.6 Réduction de la taille du StatefulSet 245
4. Base et compte de test 246
4.1 Variables d'environnement du conteneur 246
4.2 ConfigMap et secret 246
4.2.1 Pourquoi y faire appel ? 246
4.2.2 Structure d'un objet ConfigMap 247
4.2.3 Déclaration d'un objet Secret 247
4.2.4 Rattachement au conteneur 249

Chapitre 8
Mise en place d'une réplication entre pods

1. Objectifs du chapitre et prérequis 251
2. Synchronisation des pods MariaDB 252
2.1 Exposition de la problématique 252
2.2 Principe de fonctionnement de la synchronisation 252
2.2.1 Opérations à réaliser 252
2.2.2 Nombre de réplicas 253
2.3 Identifiants des serveurs 253
2.3.1 Connexion aux pods 253
2.3.2 Connexion à la base de données 253
2.3.3 Identifiants des serveurs 254
2.3.4 ID du maître 254
2.3.5 Création du compte de réplication sur le maître 255
2.3.6 Configuration de l'esclave 256

2.4 Activation de la synchronisation 256
2.4.1 Activer les journaux pour la réplication 256
2.4.2 Commande docker-entrypoint.sh 257
2.4.3 Consultation de l'état du maître 258
2.4.4 Configuration de l'esclave 259
2.5 Test de la réplication 260
2.5.1 Connexion au maître 260
2.5.2 Création d'une table 261
2.5.3 Connexion à l'esclave 261
3. Automatisation de la synchronisation 262
3.1 Scripts de démarrage et synchronisation 262
3.1.1 Script de démarrage 262
3.1.2 Configuration de la synchronisation 263
3.1.3 Scripts SQL additionnels 264
3.1.4 Script d'arrêt de la base 264
3.2 Scripts et objet ConfigMap 265
3.3 Création du ConfigMap 265
3.4 Montage du ConfigMap 268
3.4.1 Référencement du ConfigMap dans la liste des volumes 268
3.4.2 Point de montage du ConfigMap 269
3.5 Démarrage et arrêt du conteneur 269
3.5.1 Commande de démarrage 269
3.5.2 Commande d'arrêt de la base 270
3.6 Résumé des modifications 270
3.7 État du déploiement 272
3.7.1 État des pods 272
3.7.2 Journaux d'activité du pod esclave 272
3.7.3 Test de la synchronisation 272
3.7.4 Vérification du fonctionnement de la synchronisation . . 273

Chapitre 9
Gestion des briques internes de Kubernetes

1. Objectifs du chapitre et prérequis 275
2. Espace de noms kube-system 275
   2.1 Pods présents dans l'espace de noms kube-system 275
   2.2 CoreDNS 276
   2.3 etcd 277
   2.4 Le gestionnaire de réseau Kindnet 277
   2.5 Le gestionnaire d'extensions de Minikube 277
   2.6 Le serveur d'API 278
   2.7 Le proxy Kubernetes (kube-proxy) 278
   2.8 Le gestionnaire de tâches (scheduler) 278
   2.9 Le gestionnaire de contrôle (controller manager) 278
   2.10 Kubelet 278
3. Configuration des serveurs maîtres 279
   3.1 Principe de lancement des pods système 279
   3.2 Contenu du répertoire /etc/kubernetes/manifests 279
   3.3 Contenu des fichiers 280
   3.4 Désactivation d'un pod système 281
   3.5 Réactivation du pod système 282
4. Monitoring des conteneurs du cluster avec Glances 283
   4.1 Origine du besoin 283
   4.2 Consultation des DaemonSets 283
   4.3 Présentation de Glances 284
   4.4 Définition du DaemonSet 284
      4.4.1 Structure de la déclaration 284
      4.4.2 Champ volumes 285
      4.4.3 Champ containers 285
   4.5 Création du DaemonSet 286
      4.5.1 Déclaration complète 286
      4.5.2 Création du DaemonSet 287
      4.5.3 Consultation des pods 287

4.6 Annotations de tolérance 288
4.6.1 Présentation du mécanisme 288
4.6.2 Récupération des annotations taints 288
4.6.3 Tolérances de lancement 290
4.6.4 Modification du DaemonSet 290
4.7 Arrêt d'un DaemonSet 292
4.8 Connexion à Glances 294

Chapitre 10
Helm - Gestionnaire de package

1. Objectifs du chapitre et prérequis 295
2. Présentation de Helm 295
2.1 Pourquoi faire appel à Helm ? 295
2.2 Principe de fonctionnement 296
3. Déploiement de Helm 296
3.1 Installation du client Helm 296
3.1.1 Installation à l'aide d'Arkade 296
3.1.2 Installation manuelle 297
3.2 Consultation de la version de Helm 298
3.3 Configuration du client Helm 298
4. Déploiement d'une application avec Helm 299
4.1 Déterminer le package à déployer 299
4.1.1 Recherche d'un chart Helm 299
4.1.2 Gestion des sources de charts Helm 299
4.1.3 Recherche et gestion du cache de Helm 301
4.2 Installation du package WordPress 302
4.2.1 Un peu de vocabulaire 302
4.2.2 Lancement de l'installation 302
4.2.3 Installation sans accès direct 304

4.3 Corrections de l'installation 304
4.3.1 Quelques remarques 304
4.3.2 Spécification du nom et espace de noms 304
4.3.3 Lancement de l'installation 305
4.3.4 Mise à jour et réentrance 305
4.3.5 Généralisation des options d'installation 311
4.4 Éléments déployés avec Helm 311
4.5 Suppression d'un déploiement 312
4.6 Annulation de la suppression 313
4.7 Purge d'un chart Helm 314
5. Cycle de vie d'une application déployée avec Helm 315
5.1 Ouverture du port vers WordPress 315
5.2 Connexion à WordPress 316
5.3 Configuration d'un chart Helm 317
5.3.1 Consultation des options d'un chart 317
5.3.2 Configuration de la publication (Minikube) 319
5.4 Historique de déploiement 321
5.5 Visualisation des différences avant installation 322
5.5.1 Origine du besoin 322
5.5.2 Installation de l'extension diff pour Helm 322
5.5.3 Visualisation des modifications dans l'historique 324
5.5.4 Visualisation des modifications avant installation 325
5.6 Retour arrière 327
5.7 Helm Dashboard 328
5.8 Portail Artifact Hub 330

Chapitre 11
Contextes et outils tiers Kubernetes

1. Objectifs du chapitre et prérequis 333
2. Gestion des contextes avec kubectl 334
2.1 Origine du besoin 334
2.2 Lister les contextes 334

2.3 Variable d'environnement KUBECONFIG 335
2.3.1 Spécifier l'emplacement du fichier 335
2.3.2 Spécifier plusieurs fichiers 335
2.4 Changement de contexte 336
2.5 Créer un contexte 337
2.6 Supprimer un contexte 338
2.7 Outils de gestion de contexte 339
2.7.1 Présentation de kubectx et kubens 339
2.7.2 Installation de kubectx et kubens 339
2.7.3 Mise en place de l'autocomplétion 340
2.7.4 Test des commandes 341
2.8 Contexte dans le prompt utilisateur 341
2.8.1 Pourquoi afficher le contexte ? 341
2.8.2 Activation à l'aide de oh-my-zsh 342
2.8.3 Activation avec bash 342
2.8.4 Exemple d'affichage 342
2.9 Changement des couleurs du terminal 343
2.9.1 Principe de fonctionnement 343
2.9.2 Création de la fonction 344
2.9.3 Ajout de l'appel dans l'invite de commandes 345

3. Utilitaires Kubernetes 346
3.1 Krew : gestionnaire d'extensions 346
3.1.1 Présentation du mécanisme d'extensions 346
3.1.2 Installation de Krew 346
3.1.3 Test de l'extension 347
3.2 node-shell : lancer une session sur un nœud du cluster 348
3.2.1 Contexte 348
3.2.2 Installation de node-shell 349
3.2.3 Utilisation de node-shell 350
3.3 k9s : interface texte de suivi 350
3.3.1 Contexte 350
3.3.2 Installation de k9s 351
3.3.3 Lancement de k9s 351

3.3.4 Personnalisation de k9s 352
3.4 Kubespy : espionnage de l'activité 353
3.4.1 Présentation de Kubespy 353
3.4.2 Installation de Kubespy 354
3.4.3 Observation d'un déploiement avec Kubespy 354
3.5 Sniff : capture du trafic réseau d'un pod 355
3.5.1 Principe de fonctionnement 355
3.5.2 Installation de Sniff et Wireshark 355
3.5.3 Lancement d'une séance de capture 356

Chapitre 12
Analyse et sécurisation d'un cluster

1. Objectifs du chapitre et prérequis 359
2. Trivy : analyse de failles de sécurité 360
2.1 Trivy : outil d'analyse du cluster 360
2.1.1 Présentation de Trivy 360
2.1.2 Installation de Trivy 360
2.1.3 Lancement de l'analyse 361
2.1.4 Automatisation du lancement de l'analyse 363
2.1.5 Résultat de l'analyse 366
2.1.6 Lancement à intervalles réguliers 367
2.2 Mise en place de l'opérateur Trivy 370
2.2.1 Problèmes liés au rapport généré par Trivy 370
2.2.2 Installation de l'opérateur 370
2.2.3 Utilisation des nouveaux types de l'opérateur Trivy . . . 371
3. OPA (Open Policy Agent) Gatekeeper : le gardien du cluster 374
3.1 Présentation d'OPA Gatekeeper 374
3.2 Installation de Gatekeeper 374
3.3 Principe de fonctionnement 376
3.4 Création d'une règle OPA 376
3.5 Ressources à disposition de la communauté 380

Chapitre 13
Services managés Kubernetes

1. Objectifs du chapitre et prérequis 381
2. Service managé de Google : GKE 382
   2.1 Présentation du service Google 382
   2.2 Administration depuis la console Google 382
   2.3 Installation de la commande gcloud en local 384
      2.3.1 Installation sur Debian/Ubuntu 384
      2.3.2 Mise en place de l'autocomplétion 385
   2.4 Configuration de l'environnement 386
      2.4.1 Authentification auprès de Google Cloud 386
      2.4.2 Projet associé avec le contexte courant 388
      2.4.3 Activation de l'API 388
   2.5 Gestion du cluster GKE 389
      2.5.1 Consultation de la liste des clusters 389
      2.5.2 Versions et régions disponibles 390
   2.6 Création d'un cluster 390
      2.6.1 Options de création 390
      2.6.2 Lancement de la création du cluster 391
      2.6.3 Récupération du fichier d'accès au cluster 393
   2.7 Consultation du cluster 393
      2.7.1 Liste des nœuds 393
      2.7.2 Services démarrés 393
   2.8 Délégation des droits d'accès 396
      2.8.1 Configuration des accès 396
      2.8.2 Principe du mécanisme sous-jacent 397
   2.9 Suppression d'un cluster GKE 398
3. Service managé Microsoft Azure : AKS 399
   3.1 Présentation du service Azure 399
   3.2 Administration depuis la console Azure 399
      3.2.1 Présentation de la console 399
      3.2.2 Consultation du tableau de bord Kubernetes 401

3.3 Installation de la commande az en local 401
3.3.1 Installation sur Debian/Ubuntu 402
3.3.2 Mise en place de l'autocomplétion 402
3.4 Authentification auprès du service Azure. 403
3.5 Emplacement de déploiement 404
3.5.1 Liste des emplacements 404
3.5.2 Versions disponibles de Kubernetes 405
3.6 Création d'un cluster 406
3.6.1 Création d'un groupe de ressources. 406
3.6.2 Lancement de la création du cluster 407
3.6.3 Récupération du fichier de connexion. 409
3.6.4 Zone DNS par défaut 409
3.7 Consultation de la liste des clusters 410
3.8 Délégation des droits d'accès 410
3.9 Suppression d'un cluster AKS 411
4. Service managé d'Amazon : EKS 411
4.1 Présentation du service Amazon AWS 411
4.2 Introduction de la commande eksctl. 412
4.3 Configuration des accès Amazon 412
4.4 Installation des binaires 416
4.4.1 Installation d'eksctl 416
4.4.2 Installation de l'outil aws cli 416
4.4.3 Vérification de la communication avec AWS 417
4.5 Création du cluster EKS 418
4.5.1 Aide en ligne d'eksctl 418
4.5.2 Options intéressantes à la création d'un cluster. 419
4.5.3 Lancement de la création du cluster 420
4.6 Configuration des accès kubectl 424
4.7 Mécanisme de communication 424
4.8 Délégation des droits d'accès 425
4.8.1 Configuration des accès 425
4.8.2 Principe du mécanisme sous-jacent. 426
4.9 Suppression du cluster 427

5. Service Kubernetes OVHcloud 429
5.1 Présentation d'OVHcloud 429
5.2 Méthode de création du cluster et prérequis 429
5.3 Réseau des nœuds du cluster 430
5.4 Création du cluster Kubernetes 431
5.5 Suppression du cluster Kubernetes 432
6. Infrastructure as Code 433
6.1 Origine du besoin 433
6.2 Installation d'OpenTofu 433
6.3 Génération des identifiants d'accès OVH 434
6.4 Déclaration des objets dans OpenTofu/Terraform 437
6.5 Téléchargement des dépendances 441
6.6 Création de la plateforme 442
6.7 Test et utilisation du cluster 444
6.8 Destruction du cluster 444
7. Accès en lecture-écriture multiple 446
7.1 Origine du besoin 446
7.2 Serveur NFS déployé dans Kubernetes 446
7.2.1 Limitations 446
7.2.2 Déploiement d'un serveur NFS 447
7.2.3 Vérification du déploiement 448
7.2.4 Test de création de volume persistant (PVC) 448
7.3 Service managé Amazon : EFS 449
7.3.1 Consultation des instances EFS 449
7.3.2 Création d'une instance EFS 449
7.4 Service managé Google : Filestore 450
7.4.1 Consultation des instances Filestore 450
7.4.2 Création d'une instance Filestore 451
7.5 Classe de stockage NFS 452
7.5.1 Installation du chart 452
7.5.2 Vérification de l'installation du chart 452
7.5.3 Test de création d'une demande de volume persistant (PVC) 453

7.6 Classe de stockage Azure 453

Chapitre 14
Installation de Kubernetes en interne

1. Objectifs du chapitre et prérequis 455
2. Installation à l'aide de Kubespray 456
2.1 Origine du besoin 456
2.2 Pourquoi Kubespray ?. 456
2.3 Principe de Kubespray 456
2.4 Prérequis des machines à administrer avec Ansible 457
2.4.1 Échange de clés SSH 457
2.4.2 Escalade de droits sudo. 457
2.5 Structure du cluster 459
2.5.1 Architecture du cluster Kubespray 459
2.5.2 Groupes de machines 459
2.6 Préparation des éléments d'installation 460
2.6.1 Clonage du dépôt Git 460
2.6.2 Installation des prérequis 460
2.7 Préparation de l'inventaire Ansible 461
2.7.1 Qu'est-ce qu'un inventaire ? 461
2.7.2 Fichier d'inventaire d'exemple. 461
2.7.3 Test de la communication Ansible/SSH 463
2.8 Installation du cluster 464
2.8.1 Configuration du cluster 464
2.8.2 Lancement de l'installation 465
2.8.3 Mise à jour du cluster 465
2.9 Accès au cluster. 466
2.9.1 Configuration de l'accès au cluster 466
2.9.2 Utilisation d'un répartiteur de charge. 466
2.9.3 Vérification de la communication avec le cluster. 467

3. Environnement embarqué avec k3s. 468
3.1 Présentation et but du projet. 468
3.2 Installation de k3s 469
3.3 Communication avec le cluster 470
3.4 Ajout de nœuds au cluster 471

Chapitre 15
Exposition des applications sur Internet

1. Objectifs du chapitre et prérequis 473
2. Gestion des entrées DNS 474
2.1 Principe de fonctionnement. 474
2.2 Prérequis 475
2.3 Retour sur le fonctionnement du domaine DNS nip.io. 475
2.4 Activation du service DNS. 476
2.4.1 Console Cloud DNS Google 476
2.4.2 Service DNS Zones d'Azure. 479
2.4.3 Service Route 53 d'AWS. 480
2.5 Configuration DNS 481
2.5.1 Opération à réaliser 481
2.5.2 Console DNS OVH 481
2.5.3 Vérification de la délégation 483
2.6 Gestionnaire de DNS 484
2.6.1 Présentation de la brique external-dns 484
2.6.2 Création du compte d'administration DNS de Google Cloud 484
2.6.3 Création du compte d'administration DNS service Azure. 486
2.6.4 Création du compte d'administration DNS Amazon (Route 53) 488
2.6.5 Création d'un secret (Google et Azure). 490
2.6.6 Déploiement d'external-dns 491
2.6.7 Vérification du fonctionnement d'external-dns. 493

3. Exposition de services et répartition de charge 494
3.1 Présentation du mécanisme 494
3.2 Retour sur la notion de service 494
3.2.1 Rôle d'un service 494
3.2.2 Structure d'un service 495
3.2.3 Type ClusterIP 495
3.2.4 Type NodePort et LoadBalancer 496
3.2.5 Type ExternalName 496
3.3 Service associé au proxy inverse 496
4. Le contrôleur Ingress 497
4.1 Principe de fonctionnement 497
4.2 Le rôle du contrôleur Ingress 498
4.3 Structure d'une règle Ingress 498
4.4 Droits nécessaires pour un contrôleur 500
5. Le contrôleur Ingress Google 500
5.1 Prérequis 500
5.2 Présentation du contrôleur GLBC 501
5.3 Déploiement de Mailpit 501
5.3.1 Préparation de la règle Ingress 501
5.3.2 Déploiement de l'application Mailpit 502
5.3.3 Consultation de l'état de l'objet Ingress 502
5.3.4 Consultation du journal d'activité d'external-dns 502
5.3.5 Consultation de Mailpit 504
6. Le contrôleur Ingress Nginx 507
6.1 Pourquoi changer de contrôleur Ingress ? 507
6.2 Présentation du logiciel Nginx 507
6.3 Installation d'Ingress Nginx sur GKE (Google) 508
6.3.1 Détermination du chart Helm à installer 508
6.3.2 Espace de noms et configuration du chart 508
6.4 Utilisation du contrôleur 509
6.4.1 Utilisation du champ spec.ingressClassName 509
6.4.2 Vérification du déploiement 510

6.5 Annotations Ingress Nginx 511
6.6 Ajustement de la configuration du contrôleur Nginx 512
6.6.1 Récupération de l'origine du trafic 512
6.6.2 Haute disponibilité et règles d'affinité/anti-affinité . . . 514
6.6.3 Gestion des contrôleurs Ingress 517
6.6.4 Réservation et limitation des ressources disponibles . . . 519
7. Le contrôleur Ingress Traefik 521
7.1 Présentation de Traefik 521
7.2 Installation du chart Helm 521
7.3 Utilisation du nouveau contrôleur Ingress 524
7.3.1 Sélectionner le contrôleur Ingress Traefik 524
7.3.2 Création de la règle Ingress faisant appel à Traefik 524
7.3.3 État des règles Ingress. 525
7.4 Tableau de bord de Traefik 526
7.5 Annotations Ingress Traefik 528
7.6 Distribution de la charge et haute disponibilité 529
7.7 Réservation et limitation des ressources disponibles 530
7.8 Pour aller plus loin 530

Chapitre 16
Sécurisation : accès aux applications

1. Objectifs du chapitre et prérequis 533
2. Mise en place de Let's Encrypt. 534
2.1 Présentation de Let's Encrypt 534
2.2 Installation du chart Helm cert-manager 534
2.2.1 Présentation du chart Helm. 534
2.2.2 Prérequis avant installation 535
2.3 L'émetteur de certificats (issuer) 536
2.3.1 Principe du protocole ACME 536
2.3.2 Structure de la déclaration d'un émetteur 536
2.3.3 Exemple de déclaration Issuer Google. 538
2.3.4 Exemple de déclaration Issuer Azure. 539

2.3.5 Exemple de déclaration Issuer Amazon 540
2.3.6 Limitations et certificats avec joker 540
2.4 Exemples de déclarations 541
2.4.1 Serveur Let's Encrypt de test 541
2.4.2 Serveur Let's Encrypt de production 542
2.5 Déclaration des certificats 543
2.5.1 État des émetteurs de certificats 543
2.5.2 Structure d'un certificat 543
2.5.3 Certificat de test 544
2.5.4 État du certificat 545
2.5.5 Journal d'activité de cert-manager 545
2.5.6 Consultation du secret 547
2.5.7 Certificat de production 549
2.5.8 Marche à suivre en cas de problèmes 550
2.6 Rattachement du certificat à la règle Ingress 550
2.7 Automatisation de la gestion des certificats 552
2.7.1 Certificat par défaut du contrôleur Ingress Nginx 552
2.7.2 Mécanisme d'annotations 553
2.7.3 Émetteur de certificats par défaut 554

3. Protection de l'accès aux applications 555
3.1 Origine du besoin 555
3.2 Mot de passe simple (HTTP basic) 555
3.2.1 Principe de fonctionnement 555
3.2.2 Création du secret à l'aide de htpasswd 556
3.2.3 Import du secret 556
3.2.4 Configuration de l'authentification 557

4. Authentification basée sur OAuth2 561
4.1 À propos du protocole OAuth2 561
4.2 Principe de la solution 561
4.3 Création d'un identifiant GitHub 562
4.4 Déploiement du proxy 564
4.4.1 À propos du proxy 564
4.4.2 Configuration du chart Helm 565

4.4.3 Déploiement du chart Helm 566
4.4.4 État du déploiement 566
4.5 Déclaration des règles Ingress 567
4.5.1 Description des règles Ingress 567
4.5.2 Annotations Ingress de Mailpit 567
4.5.3 Description Ingress du proxy OAuth 567
4.5.4 Déclaration des règles Ingress 567
4.6 Tests de connexion 569
4.7 Restriction des accès 570
4.7.1 Mécanisme d'autorisation 570
4.7.2 Restriction par domaine e-mail 571
4.7.3 Restriction par organisation GitHub 571

Chapitre 17
Polices réseau

1. Objectifs du chapitre et prérequis 573
2. Les polices réseau (network policies) 573
2.1 Présentation du mécanisme 573
2.2 Kubernetes Network Plugins 574
2.3 Polices réseau sur services managés et installation maison . . . 575
2.4 Installation de Calico sur Minikube 575
2.4.1 Activation de CNI sur Minikube 575
2.4.2 Installation de Calico 575
2.5 Connexions entrantes 576
2.5.1 Test de la connexion en interne 576
2.5.2 Bloquer les accès internes 576
2.5.3 Test de la règle 578
2.6 Connexions sortantes 579
2.6.1 Test des connexions externes 579
2.6.2 Restriction sur les règles sortantes 579
2.6.3 Test de la règle 580

3. Protection de l'application WordPress. 580
3.1 Contexte 580
3.2 Déploiement de WordPress 580
3.3 Restriction des accès. 582
3.3.1 Référencement de l'ensemble des flux 582
3.3.2 Restriction de tous les accès. 582
3.3.3 Autorisation de l'accès du contrôleur Ingress sur WordPress 583
3.3.4 Autorisation de l'accès entre WordPress et MariaDB . . . 584
3.3.5 Test des règles réseau 585
3.4 Ressources externes 587

Chapitre 18
Montée en charge automatique

1. Objectifs du chapitre et prérequis 589
2. Le serveur de métriques 590
2.1 Présentation de la brique metrics-server 590
2.2 Activation du serveur de métriques. 590
2.2.1 Vérification de la présence du serveur de métriques . . . 590
2.2.2 Activation sous Minikube 591
2.2.3 Activation sur Amazon EKS 591
2.3 Consultation de la consommation des pods et des nœuds . . . 592
2.3.1 État des nœuds d'un cluster. 592
2.3.2 État des pods. 592
3. Activation de la montée en charge automatique 593
3.1 Test avec l'application Mailpit 593
3.2 Lancement d'un bench. 594
3.2.1 Présentation d'Apache Bench 594
3.2.2 Installation d'Apache Bench 595
3.2.3 Lancement du test initial. 595
3.3 Gestion de la montée en charge. 596

3.4 Lancement du test 597
3.4.1 Test à froid : montée en charge des pods 597
3.4.2 Test à chaud 598
3.4.3 Diminution du nombre de pods 599
4. Scalabilité des nœuds d'un cluster. 599
4.1 Contexte 599
4.2 Présentation de l'autoscaler 600
4.3 Activation de l'autoscaler avec Google Cloud 600
4.4 Activation de l'autoscaler avec Azure AKS 601
4.5 Activation de l'autoscaler avec Amazon EKS 601
4.5.1 Présentation du mécanisme de l'ASG 601
4.5.2 Affichage des tags d'un groupe ASG 602
4.5.3 Activation de l'autoscaler. 603
4.5.4 Vérification de l'activation du mécanisme de l'autoscaler 604
4.6 Test de montée en charge 604
4.6.1 Déploiement de WordPress 604
4.6.2 Pods en attente de ressources. 605
4.6.3 État des nœuds 605
4.6.4 Démarrage du pod 606
4.6.5 Nettoyage des déploiements 606
5. Définition de groupes de machines 607
5.1 Contexte 607
5.2 Les différents types de machines 608
5.2.1 Les machines préemptives (Spot) 608
5.2.2 Réservation d'instances 609
5.2.3 Familles de machines 609
5.2.4 Quelques exemples de familles de machines 610
5.2.5 Utilisation d'accélérateurs graphiques (GPU). 612
5.2.6 Quelques exemples de coûts 613
5.3 Création de groupes de machines 615
5.3.1 Contexte 615
5.3.2 Création d'un groupe avec Google 616

5.3.3 Création d'un groupe avec Azure 617
5.3.4 Création d'un groupe avec AWS EKS 619
5.3.5 Création d'un groupe avec OVH 621
5.4 Mécanisme de tolérance et sélection de nœuds 623

Chapitre 19
Surveillance à l'aide de Prometheus

1. Objectifs du chapitre et prérequis 627
2. Mise en place de Prometheus 628
2.1 À propos de Prometheus 628
2.2 Fonctionnement de Prometheus 629
2.2.1 Architecture de Prometheus 629
2.2.2 Le moteur Prometheus 629
2.2.3 Les exporteurs Prometheus 630
2.3 Installation de Prometheus 631
2.3.1 Choix du chart Prometheus 631
2.3.2 Qu'est-ce qu'un opérateur ? 631
2.3.3 Déploiement de l'opérateur Prometheus 632
2.3.4 Pods démarrés 634
2.3.5 Objets déploiements 634
2.3.6 Nouvelles ressources Prometheus 635
2.3.7 DaemonSet : node exporter 636
2.4 Priorisation des briques de surveillance 637
2.4.1 Problème de la surveillance 637
2.4.2 Déclaration des classes de priorité 637
2.4.3 Modification du déploiement de Prometheus 638
3. Utilisation de Prometheus 638
3.1 Fonctionnement des métriques 638
3.1.1 Consultation des métriques de Prometheus 638
3.1.2 Présentation de l'interface de Prometheus 639
3.1.3 Métriques de Kubernetes 641
3.1.4 Déclaration des points de collecte dans Kubernetes 642

3.1.5 Consultation des points de collecte dans Prometheus . . 643
3.2 Définition des alertes 644
3.2.1 Consultation de la liste des alertes 644
3.2.2 Structure d'une règle d'alerte 645
3.2.3 Définition d'alertes 646
3.3 Gestionnaire d'alertes 647
3.3.1 Rôle du gestionnaire d'alertes 647
3.3.2 Consultation du gestionnaire d'alertes 648
3.3.3 Configuration des alertes 648
3.3.4 Désactivation des alertes scheduler et manager (clusters managés) 649
3.3.5 Configuration de l'envoi des notifications 651
3.3.6 Adresse de l'API de Slack 652
4. Tableaux de bord Grafana 655
4.1 Présentation de Grafana 655
4.2 Configuration de Grafana 655
4.2.1 Branchement au moteur Prometheus 655
4.2.2 Définition des tableaux de bord 656
4.3 Interface Grafana 656
4.4 Sécurisation de l'accès à Grafana 658
5. Suppression du chart de Prometheus 659

Chapitre 20
Centralisation des journaux d'activité

1. Objectifs du chapitre et prérequis 661
2. Principe de la centralisation des journaux 662
2.1 Architecture 662
2.2 Caractéristiques de l'agent déployé 663
2.3 Mécanisme de centralisation des journaux 664

3. Centralisation dans le cloud 664
3.1 Centralisation à l'aide d'un service managé 664
3.2 Google Stackdriver 665
3.2.1 Présentation de Stackdriver 665
3.2.2 Pod Fluent Bit (cluster GKE) 665
3.2.3 Consultation des journaux 665
3.3 Azure Monitor 666
3.3.1 Présentation d'Azure Monitor 666
3.3.2 Consultation des journaux 667
3.4 Amazon Cloudwatch 668
3.4.1 Présentation de Cloudwatch 668
3.4.2 Activation de Cloudwatch sur le centre de contrôle . . . 668
3.4.3 Configuration de Cloudwatch 669
3.4.4 Création de la police de communication avec Cloudwatch 670
3.4.5 Création d'un compte et rattachement à la police 671
3.4.6 Création d'une clé d'accès 672
3.4.7 Envoi des journaux dans Cloudwatch 672
3.4.8 État des pods déployés 674
3.4.9 Consultation des journaux dans Cloudwatch 675
4. Centralisation des journaux avec Loki 676
4.1 Présentation de Loki 676
4.1.1 Origine de Loki 676
4.1.2 Loki vs Opensearch/Elasticsearch 676
4.1.3 Conseil d'utilisation 676
4.2 Installation de Loki 677
4.3 Configuration de la source de données Grafana 678
4.4 Consultation des journaux dans Grafana 680
4.4.1 Vérification des sources de données 680
4.4.2 Consultation des journaux 680
5. Centralisation des journaux avec Opensearch/Elasticsearch 682
5.1 Avertissements et limitations 682
5.2 À propos d'Opensearch et d'Elasticsearch 682

5.3 Déploiement des briques Opensearch/Elasticsearch 683
5.3.1 Installation d'Opensearch 683
5.3.2 Installation de l'agent Fluent-bit 684
5.3.3 Installation de Kibanad/Opensearch Dashboards 687
5.4 État des différentes briques 687
5.4.1 État du moteur Elasticsearch 687
5.4.2 Agent Fluent-bit 688
6. Gestion d'Opensearch/Elasticsearch 688
6.1 Utilisation de Kibana/Opensearch Dashboards 688
6.1.1 Accéder à l'interface de Kibana ou Opensearch Dashboards 688
6.1.2 Création de l'index 690
6.2 Branchement sur Grafana 692
6.2.1 Source de données Opensearch/Elasticsearch 692
6.2.2 Création d'un objet ConfigMap 692

Chapitre 21
Maillage de services

1. Objectifs du chapitre et prérequis 695
2. Présentation d'Istio 696
2.1 Micro-services et mise en réseau de services 696
2.2 Présentation d'Istio 697
2.3 Principe de fonctionnement 697
3. Installation d'Istio 699
3.1 Configuration d'external-dns 699
3.2 Dépôt des charts Istio 700
3.3 Installation d'istio 701
3.4 Configuration d'Istio 701
3.4.1 Activation des tableaux de bord 701
3.4.2 Configuration d'Istio 702

3.5 Configuration du composant Gateway Istio 703
3.5.1 Présentation du mécanisme 703
3.5.2 Installation du composant Gateway 703
3.5.3 Génération du certificat 705
4. Utilisation d'Istio 706
4.1 Injection de pods dans le maillage de services 706
4.1.1 Installation d'istioctl 706
4.1.2 Injection du sidecar à l'aide d'istioctl 707
4.1.3 Injection du sidecar par annotation 708
4.1.4 Désactivation du sidecar par annotation 708
4.2 Déploiement d'une application de test 709
4.2.1 Principe de l'exposition d'application avec Istio 709
4.2.2 Préparation du fichier de déploiement Mailpit 710
4.2.3 Déclaration du service Mailpit 713
4.2.4 Création de l'objet Gateway 713
4.2.5 Création du service virtuel (VirtualService) 715
4.3 Sécurisation des flux 716
4.3.1 Activation de Mutual TLS (mTLS) 716
4.3.2 Consultation des polices du service Mailpit 717
4.3.3 Forcer l'utilisation de mTLS 719
5. API Gateway 722
5.1 Origine du besoin 722
5.2 Principe de fonctionnement 723
5.3 Installation de l'API Gateway 723
5.4 Définition de l'objet Gateway 724
5.5 Définition de l'objet HTTPRoute 728
5.6 Configuration du module external-dns 730
5.7 Un mot sur les limitations actuelles 731
6. Tableaux de bord 731
6.1 Présentation des différentes briques 731
6.2 Interface Kiali 732
6.3 Interface Grafana 733

6.4 Interface Jaeger 736

Chapitre 22
Compilation et stockage d'images Docker

1. Objectifs du chapitre et prérequis 739
2. Création d'une image Docker 740
   2.1 Application d'exemple Flask HealthCheck 740
      2.1.1 Présentation de l'application 740
      2.1.2 Présentation des dépendances 740
      2.1.3 Description des dépendances 740
      2.1.4 Installation des dépendances 741
      2.1.5 Initialisation de l'application 742
      2.1.6 Fonction racine 742
      2.1.7 Lancement du programme 742
   2.2 Environnement de compilation 745
   2.3 Un mot sur les alternatives à Docker 745
   2.4 Le fichier Dockerfile 746
      2.4.1 Présentation du format 746
      2.4.2 Création de l'image 747
      2.4.3 Compilation et tag de l'image 748
      2.4.4 Authentification sur le registre d'images Docker 749
      2.4.5 Pousser l'image sur le registre 749
3. Image Docker multi-étape (multi-stage) 751
   3.1 Origine du besoin 751
   3.2 Exemple de compilation avec Maven 751
4. Analyse d'images 752
   4.1 Historique des commandes 752
   4.2 Analyse de l'image : Dive 753
      4.2.1 Présentation de Dive 753
      4.2.2 Installation de Dive 753
      4.2.3 Consultation du contenu d'une image 753

5. Analyse de failles de sécurité avec Trivy 754
5.1 Origine du besoin 754
5.2 Installation de Trivy 755
5.3 Lancement de l'analyse 755
5.4 Configuration de l'analyse 756
6. Utilisation de registres Docker privés 758
6.1 Origine du besoin 758
6.2 Déploiement d'image d'un registre privé 758
6.2.1 Exposition de la problématique 758
6.2.2 Création du secret 758
6.2.3 Utilisation du secret 759
6.2.4 Utilisation d'un compte de service 760
6.2.5 Recopie d'un secret entre espaces de noms 762
6.3 Erreurs de récupération des images 763
6.3.1 Détection des erreurs 763
6.3.2 Erreur sur le nom de l'image 763
6.3.3 Secret absent ou non spécifié 764
6.3.4 Identifiants invalides 764
6.4 Services de registres managés 764
6.4.1 Solutions managées 764
6.4.2 Service Google Container Registry 765
6.4.3 GitLab.com 767

Chapitre 23
Usine logicielle

1. Objectifs du chapitre et prérequis 769
2. Compilation à l'aide de GitLab 770
2.1 Application à compiler 770
2.2 Mécanisme de pipeline GitLab 770
2.3 Adresse et contenu du dépôt 770
2.4 Structure du fichier .gitlab-ci.yml 771
2.5 Exemple de fichier .gitlab-ci.yml de compilation d'image 772

2.6 Construction d'image à l'aide de Kaniko 774
2.7 Pour la suite 776
3. Déploiement continu avec Jenkins 776
3.1 À propos de Jenkins 776
3.2 Installation de Jenkins 777
3.2.1 Configuration du chart 777
3.2.2 Vérification de l'installation 778
3.2.3 Connexion à l'interface de Jenkins 779
3.2.4 Installation d'extensions 779
4. Pipeline de déploiement continu avec Jenkins 780
4.1 Prérequis 780
4.2 Présentation du mécanisme de déploiement continu 780
4.3 Stockage des identifiants Docker 781
4.4 Création de l'environnement develop 781
4.5 Création du pipeline 782
4.5.1 Création du pod de compilation 782
4.5.2 Squelette du pipeline de déploiement 784
4.5.3 Récupération du code source 785
4.5.4 Vérifications et tests 785
4.5.5 Compilation de l'image Docker 786
4.5.6 Connexion au registre et publication 786
4.5.7 Mise à jour du déploiement de test 787
4.5.8 Programme complet 788
4.6 Lancement de la compilation 790
4.6.1 Création du job 790
4.6.2 Lancement du build 791
4.7 Compte de service 793
4.7.1 Opérations à réaliser 793
4.7.2 Création d'un compte de service 793
4.7.3 Création du rôle de mise à jour 793
4.7.4 Affectation du rôle au compte de service 794
4.7.5 Affectation du compte de service 795
4.7.6 Relance de la compilation 795

4.8 Utilisation de Kaniko avec Jenkins 796
4.9 Mécanisme de Webhook 798
4.9.1 Présentation du mécanisme 798
4.9.2 Déclenchement du Webhook. 798
4.9.3 Création du Webhook 799
5. Un mot sur Jenkins X. 800

Chapitre 24
Packager son application avec Helm

1. Objectifs du chapitre et prérequis 801
2. Helm 802
2.1 Origine du besoin 802
2.2 Création d'un chart 802
2.3 Contenu d'un package 803
2.3.1 Structure d'un package. 803
2.3.2 Variables .Values, .Chart et .Release 803
2.4 Adaptation à l'application Mailpit 804
2.4.1 Résumé des travaux à réaliser 804
2.4.2 Utilisation de l'image de Mailpit 804
2.4.3 Correction sur les ports de service. 804
2.4.4 Ajout d'un ConfigMap. 806
2.4.5 Ajout du volume persistant 806
2.4.6 Montage du volume dans le conteneur. 807
2.4.7 Test de déploiement 808
2.5 Dépendances 809
2.5.1 Déclaration des dépendances 809
2.5.2 Récupération des dépendances 810
2.5.3 Déploiement du chart avec les dépendances. 811
3. Template Go 812
3.1 Principe de fonctionnement. 812
3.2 Substitution du contenu d'une variable 812

3.3 Blocs conditionnels 813
3.4 Gestion des conditions 813
3.4.1 Les valeurs de vrai ou faux 813
3.4.2 Opérateurs de comparaison 814
3.4.3 Conditions multiples et négation 815
3.5 Itération 815
3.5.1 Affichage du contenu d'un tableau 815
3.5.2 Suppression des sauts de lignes surnuméraires 816
3.5.3 Itération sur un groupe fixe d'éléments 817
3.5.4 Itération sur un tableau de hachage 817
3.5.5 Accéder à une variable globale depuis une boucle 818
3.6 Filtres 818
3.7 Valeurs par défaut 819
3.8 Fonction template 819
3.8.1 Exemple de définition de fonction 819
3.8.2 Passage de paramètres et contexte global 820
3.9 Redéploiement sur changement de configuration 821
3.9.1 Exposition de la problématique 821
3.9.2 Principe de la solution 821
3.9.3 Exemple d'implémentation 821
3.10 Génération de secret aléatoire « stable » 823
3.10.1Exposition de la problématique 823
3.10.2Principe de la solution et mise en œuvre 823

Chapitre 25

Restriction et délégation d'accès

1. Objectifs du chapitre et prérequis 825
2. Mise en place de quotas 826
2.1 Origine du besoin 826
2.2 Quotas par défaut sur un espace de noms 826
2.2.1 Création d'un espace de noms 826
2.2.2 Structure d'un objet LimitRange 827

2.2.3 Exemple de définition de limitations 828
2.2.4 Vérification de l'application des limitations 829
2.2.5 Test du mécanisme 830
2.2.6 Analyse du problème 830
2.3 Quotas globaux sur un espace de noms 831
2.3.1 Présentation des quotas de ressources (ResourceQuota) . 831
2.3.2 Structure d'un quota de ressources 832
2.3.3 Exemples de restriction de consommation CPU et mémoire 832
2.3.4 Champs pour positionner des limitations (champ hard) 833
2.3.5 Test du mécanisme 834
2.3.6 Nettoyage en fin d'exercice 835
3. Authentification et autorisation 836
3.1 Origine du besoin 836
3.2 Prérequis 837
3.3 Activation des accès anonymes 838
3.3.1 Activation des accès anonymes sur Minikube 838
3.3.2 Création du fichier d'accès 839
3.3.3 Affectation des droits en lecture 840
3.3.4 Suppression des droits d'accès anonymes 841
3.4 Principe de l'authentification par certificat 841
3.5 Problème de la révocation des certificats 842
3.6 Génération du certificat 842
3.6.1 Création de la clé et du certificat client 842
3.6.2 Emplacement de la PKI 843
3.6.3 Signature du certificat 844
3.7 Authentification par certificat 845
3.7.1 Récupération des informations de connexion au cluster 845
3.7.2 Utilisation du certificat pour l'authentification 846
3.7.3 Test de la connexion 848

3.8 Quelques exemples d'erreurs de manipulation 848
3.8.1 Problème d'autorité de certification 849
3.8.2 Problème de couple clé/certificat 849
3.8.3 Pas d'identifiant renseigné 849
3.8.4 Utilisation de la demande de certificat à la place du certificat 849
3.9 Attributions de droits administrateur sur le cluster 850
3.9.1 Contexte et opérations à réaliser 850
3.9.2 Affectation des droits administrateur à un utilisateur . 850
3.9.3 Test du nouvel administrateur 852
3.9.4 Création de nouveaux utilisateurs 852
3.9.5 Affectation des droits administrateur à un groupe 853
3.9.6 Administrateur d'un espace de noms 853

4. Mécanismes d'authentification externes 855
4.1 Présentation du mécanisme 855
4.2 Communication entre le fournisseur OAuth2 et le cluster . . . 856
4.3 Création des identifiants 857
4.4 Modification des options de démarrage (Minikube) 858
4.5 Configuration des accès clients 860
4.5.1 Présentation de l'outil kubelogin/oidc-login 860
4.5.2 Installation de kubelogin/oidc-login 860
4.5.3 Génération des identifiants d'accès 860
4.5.4 Renseignement du cluster et contexte 863
4.5.5 Test de connexion 864
4.6 Attribution des droits d'accès 865

5. Mise en place de comptes de service 865
5.1 Contexte 865
5.2 Création du compte de service 866
5.3 Attribution des droits d'administrateur au compte de service . . 868
5.4 Création du fichier Kubeconfig 869
5.4.1 Recopie des informations de connexion 869
5.4.2 Jeton de connexion du compte de service 870
5.4.3 Affectation de l'utilisateur 871

5.4.4 Test de la connexion 872
5.4.5 Révocation des accès 872

Chapitre 26
Les opérateurs Kubernetes

1. Objectifs du chapitre et prérequis 875
2. Utilisation des opérateurs 876
2.1 Présentation du principe 876
2.2 L'opérateur de Prometheus 876
2.2.1 Retour sur le chart prometheus-operator 876
2.2.2 Structure d'un objet Prometheus 877
2.3 Ressources externes sur les opérateurs existants 878
2.4 Présentation des opérateurs MySQL et MariaDB 879
2.4.1 Choix de l'opérateur 879
2.4.2 Déploiement de l'opérateur 879
2.5 Création d'une instance MariaDB 880
2.5.1 Création du cluster 881
2.5.2 Problèmes de droits sur le volume persistant 882
2.6 Objets créés au déploiement du cluster 882
2.6.1 Volumes persistants 883
2.6.2 Services MariaDB 883
2.7 Test de la réplication 883
2.7.1 Initialisation de l'environnement de test 883
2.7.2 Connexion aux instances maître et esclave 885
2.7.3 Création d'une table 886
2.7.4 Alimentation de la table 886
2.7.5 Changement du nombre de réplicas 887
2.8 Pour conclure 888

Annexes

1. bash vs zsh 889
   1.1 Les shells Unix 889
   1.2 zsh et oh-my-zsh 890
   1.3 Tester zsh et oh-my-zsh. 890
   1.4 Configurer zsh comme shell pour l'utilisateur courant 891
2. Déploiement du tableau de bord Kubernetes 892
   2.1 Installation du tableau de bord (application dashboard) 892
   2.2 Création du compte d'accès. 893
   2.3 Récupération du jeton de connexion 894
   2.4 Décodage du contenu du jeton 894
   2.5 Connexion au tableau de bord. 896

Index 899