



Ressourcesinformatiques

# APACHE 2.4

## Installation et configuration

Nicolas MARTINEZ

Téléchargement  
[www.editions-eni.fr](http://www.editions-eni.fr)



Les éléments à télécharger sont disponibles à l'adresse suivante :  
**<http://www.editions-eni.fr>**  
Saisissez la référence ENI du livre **RI24APA** dans la zone de recherche et validez. Cliquez sur le titre du livre puis sur le bouton de téléchargement.

## Avant-propos

- 1. Introduction ..... 15
- 2. À qui cet ouvrage s'adresse-t-il ? ..... 15
- 3. Niveau de compétences requis ..... 16
- 4. Structure de cet ouvrage ..... 16

## Chapitre 1 Introduction

- 1. Présentation du protocole HTTP ..... 19
  - 1.1 Introduction ..... 19
  - 1.2 La communication ..... 21
    - 1.2.1 Les méthodes utilisées par le client ..... 21
    - 1.2.2 Les en-têtes HTTP ..... 24
    - 1.2.3 La réponse HTTP ..... 25
- 2. Présentation d'Apache ..... 28
  - 2.1 Qu'est-ce qu'un serveur web ? ..... 28
  - 2.2 Apache ..... 29
  - 2.3 Historique des versions stables d'Apache HTTP Server et événements marquants ..... 30
- 3. Nouveautés de la version 2.4 ..... 31
  - 3.1 Améliorations du noyau ..... 31
  - 3.2 Nouveaux modules ..... 32
  - 3.3 Améliorations des modules ..... 32
  - 3.4 Améliorations et nouveaux outils Apache ..... 33

3.5	Modification des paramètres de compilation et d'activation des modules par défaut . . . . .	33
3.6	Modifications des syntaxes de configuration . . . . .	34

## Chapitre 2

### Installation d'Apache sous Linux

1.	Distribution Linux Debian . . . . .	37
1.1	Modification des sources d'installation . . . . .	38
1.2	Installation d'Apache . . . . .	40
1.3	Localisation des fichiers de configuration . . . . .	41
1.4	Gestion du service Apache . . . . .	42
2.	Distribution Linux CentOS . . . . .	43
2.1	Localisation des fichiers de configuration . . . . .	46
2.2	Gestion du service Apache . . . . .	47
3.	Compilation depuis les sources . . . . .	47
3.1	Téléchargement des sources . . . . .	48
3.2	Configuration de la compilation . . . . .	49
3.3	Création d'un layout d'installation personnalisé . . . . .	57
3.4	Compilation et vérifications de fonctionnement . . . . .	59
3.5	Création du script de démarrage . . . . .	61
3.6	Activation du service au démarrage . . . . .	62
3.7	Ajouter les binaires dans le path . . . . .	63
3.8	Modifier le path pour sudo . . . . .	64
3.9	Gestion du service Apache . . . . .	65

## Chapitre 3

### Configuration de base d'Apache

1.	Fichiers de configuration . . . . .	67
1.1	Syntaxe des fichiers de configuration . . . . .	68
1.2	Fichiers .htaccess . . . . .	68
1.3	Fichier MIME . . . . .	70

- 2. Première configuration . . . . . 71
  - 2.1 Création du compte utilisateur d'Apache . . . . . 71
  - 2.2 Modification du compte utilisateur d'Apache . . . . . 72
  - 2.3 Premier fichier de configuration . . . . . 73
  - 2.4 Tests et affinage de la configuration . . . . . 78
- 3. Directives Core . . . . . 82
  - 3.1 Directives globales . . . . . 82
    - 3.1.1 ServerRoot . . . . . 82
    - 3.1.2 ServerName . . . . . 82
    - 3.1.3 ServerAlias . . . . . 83
    - 3.1.4 ServerAdmin . . . . . 83
    - 3.1.5 ServerSignature . . . . . 83
    - 3.1.6 ServerTokens . . . . . 83
    - 3.1.7 LoadModule . . . . . 84
    - 3.1.8 DocumentRoot . . . . . 84
    - 3.1.9 Error . . . . . 85
    - 3.1.10 ErrorLog . . . . . 85
    - 3.1.11 ErrorLogFormat . . . . . 86
    - 3.1.12 LogLevel . . . . . 87
    - 3.1.13 ErrorDocument . . . . . 88
    - 3.1.14 Include . . . . . 88
    - 3.1.15 UseCanonicalName . . . . . 88
    - 3.1.16 UseCanonicalPhysicalPort . . . . . 89
    - 3.1.17 TimeOut . . . . . 89
    - 3.1.18 KeepAlive . . . . . 89
    - 3.1.19 MaxKeepAliveRequests . . . . . 89
    - 3.1.20 KeepAliveTimeout . . . . . 90
    - 3.1.21 HostnameLookups . . . . . 90
    - 3.1.22 AccessFileName . . . . . 90
    - 3.1.23 AllowOverride . . . . . 91
  - 3.2 Directives de type bloc . . . . . 93
    - 3.2.1 <Directory> . . . . . 93
    - 3.2.2 <DirectoryMatch> . . . . . 93

3.2.3	<Files>	94
3.2.4	<FilesMatch>	94
3.2.5	<Location>	94
3.2.6	<LocationMatch>	95
3.2.7	<VirtualHost>	95
3.2.8	<If>	96
3.2.9	<IfDefine>	96
3.2.10	<IfModule>	96
3.3	Directives MPM	97
3.3.1	PidFile	97
3.3.2	Port d'écoute et protocole	97
3.3.3	ServerLimit	98
3.3.4	ThreadLimit	98
3.3.5	ThreadsPerChild	98
3.3.6	MaxRequestWorkers (anciennement MaxClients)	99
3.3.7	StartServers	99
3.3.8	MaxSpareServers (MPM Prefork)	100
3.3.9	MinSpareServers (MPM Prefork)	100
3.3.10	MaxSpareThreads	100
3.3.11	MinSpareThreads	101
4.	Modules	101
4.1	mod_unixd (unixd_module)	101
4.1.1	User	102
4.1.2	Group	102
4.1.3	ChrootDir	102
4.1.4	Suexec	103
4.2	mod_authz_core (authz_core_module)	103
4.2.1	Require	103
4.2.2	RequireAll	105
4.2.3	RequireAny	106
4.2.4	RequireNone	106

- 4.3 mod\_authn\_core (authn\_core\_module) ..... 107
  - 4.3.1 AuthName ..... 107
  - 4.3.2 AuthType ..... 107
  - 4.3.3 AuthnProviderAlias ..... 108
- 4.4 mod\_auth\_basic (auth\_basic\_module) ..... 109
  - 4.4.1 AuthBasicProvider ..... 109
  - 4.4.2 AuthBasicAuthoritative ..... 109
  - 4.4.3 AuthBasicProvider ..... 110
  - 4.4.4 AuthBasicUseDigestAlgorithm ..... 110
- 4.5 mod\_authn\_file (authn\_file\_module) ..... 111
  - 4.5.1 AuthUserFile ..... 111
- 4.6 mod\_authz\_user (authz\_user\_module) ..... 112
- 4.7 mod\_mime (mime\_module) ..... 112
  - 4.7.1 AddCharset ..... 113
  - 4.7.2 AddLanguage ..... 113
  - 4.7.3 AddEncoding ..... 113
  - 4.7.4 AddHandler ..... 114
  - 4.7.5 AddType ..... 114
  - 4.7.6 AddInputFilter ..... 115
  - 4.7.7 AddOutputFilter ..... 115
  - 4.7.8 DefaultLanguage ..... 115
  - 4.7.9 RemoveCharset ..... 116
  - 4.7.10 RemoveEncoding ..... 116
  - 4.7.11 RemoveHandler ..... 116
  - 4.7.12 RemoveInputFilter ..... 117
  - 4.7.13 RemoveOutputFilter ..... 117
  - 4.7.14 RemoveLanguage ..... 117
  - 4.7.15 RemoveType ..... 118
  - 4.7.16 TypesConfig ..... 118
- 4.8 mod\_negotiation (negotiation\_module) ..... 118
  - 4.8.1 ForceLanguagePriority ..... 119
  - 4.8.2 LanguagePriority ..... 119

4.9	mod_log_config (log_config_module)	120
4.9.1	CustomLog	120
4.9.2	LogFormat	120
4.9.3	TransferLog	123
4.10	mod_dir (dir_module)	124
4.10.1	DirectoryIndex	124
4.10.2	DirectoryIndexRedirect	124
4.10.3	DirectorySlash	125
4.10.4	FallbackResource	125
4.11	mod_rewrite (rewrite_module)	126
4.11.1	RewriteBase	126
4.11.2	RewriteCond	128
4.11.3	RewriteEngine	131
4.11.4	RewriteMap	131
4.11.5	RewriteRule	133
4.12	mod_alias (alias_module)	136
4.12.1	Alias	136
4.12.2	AliasMatch	136
4.12.3	Redirect	137
4.12.4	RedirectMatch	138
4.12.5	RedirectPermanent	139
4.12.6	RedirectTemp	139
4.12.7	ScriptAlias	139
4.12.8	ScriptAliasMatch	140
5.	Cas concret : création de l'environnement applicatif d'un site web	140
5.1	Fichier de configuration principal d'Apache httpd.conf	141
5.2	Fichier de configuration modules.conf	143
5.3	Fichier de configuration globale server.conf	143
5.4	Fichier de configuration du MPM Event mpm-event.conf	144
5.5	Répertoire applicatif et son fichier de configuration	144

**Chapitre 4**  
**Configuration avancée**

- 1. Modules et fonctionnalités avancés ..... 147
  - 1.1 mod\_setenvif ..... 147
    - 1.1.1 BrowserMatch ..... 147
    - 1.1.2 BrowserMatchNoCase ..... 148
    - 1.1.3 SetEnvIf ..... 149
    - 1.1.4 SetEnvIfExpr ..... 150
    - 1.1.5 SetEnvIfNoCase ..... 150
  - 1.2 mod\_headers ..... 150
    - 1.2.1 Header ..... 151
    - 1.2.2 RequestHeader ..... 156
  - 1.3 mod\_expires ..... 157
    - 1.3.1 ExpiresActive ..... 157
    - 1.3.2 ExpiresByType ..... 157
    - 1.3.3 ExpiresDefault ..... 159
  - 1.4 mod\_sed ..... 159
    - 1.4.1 InputSed ..... 159
    - 1.4.2 OutputSed ..... 159
  - 1.5 mod\_macro ..... 160
    - 1.5.1 Macro ..... 160
    - 1.5.2 Use ..... 160
    - 1.5.3 UndefMacro ..... 161
  - 1.6 mod\_session ..... 161
    - 1.6.1 Session ..... 161
    - 1.6.2 SessionEnv ..... 161
    - 1.6.3 SessionExclude ..... 162
    - 1.6.4 SessionInclude ..... 162
  - 1.7 mod\_session\_cookie ..... 162
    - 1.7.1 SessionCookieName ..... 162
    - 1.7.2 SessionCookieRemove ..... 163
  - 1.8 mod\_request ..... 164
  - 1.9 mod\_unique\_id ..... 164

1.10	mod_auth_form	164
1.10.1	AuthFormProvider	165
1.10.2	AuthFormUsername	165
1.10.3	AuthFormPassword	165
1.10.4	AuthFormLocation	166
1.10.5	AuthFormLogoutLocation	166
2.	Installation de modules supplémentaires	170
2.1	mod_session_crypto	170
2.1.1	SessionCryptoCipher	170
2.1.2	SessionCryptoPassphrase	171
2.1.3	SessionCryptoPassphraseFile	171
2.1.4	Installation	171
2.1.5	Compilation des bibliothèques APR et APR-util	173
2.1.6	Recompilation d'Apache	174
2.2	mod_security	176
2.2.1	Installation	176
2.2.2	Activation du module	177
3.	Cas concret : hébergement de plusieurs sites web sur un même serveur à partir de modèles de configuration	179

## Chapitre 5

### Gestion des logs sous Apache

1.	Activation et configuration	185
1.1	Journaux d'erreur	186
1.1.1	Utilisation de LogLevel	186
1.1.2	Utilisation de ErrorLogFormat	189
1.2	Module mod_log_config	190
1.2.1	Utilisation de la directive LogFormat	190
1.3	Module mod_logio	192
1.4	Logs Apache au format JSON	192

- 2. Rotation ..... 193
  - 2.1 Rotation avec Logrotate ..... 194
  - 2.2 Rotation avec rotatelogs ..... 195
    - 2.2.1 Rotation basée sur un intervalle de temps ..... 196
    - 2.2.2 Rotation basée sur une taille de fichier ..... 196
- 3. Centralisation ..... 197
  - 3.1 Principe ..... 197
  - 3.2 Serveur Rsyslog ..... 197
    - 3.2.1 Installation de Rsyslog ..... 197
    - 3.2.2 Configuration du serveur Apache ..... 197
    - 3.2.3 Configuration du client Rsyslog ..... 198
    - 3.2.4 Configuration du serveur Rsyslog central ..... 198
- 4. Outil d'interprétation et de statistiques : AWStats ..... 200
  - 4.1 Installation ..... 200
  - 4.2 Configuration d'Apache ..... 202
  - 4.3 Configuration d'AWStats ..... 203

**Chapitre 6**  
**Apache en tant que serveur web dynamique**

- 1. Introduction ..... 207
- 2. PHP avec Apache en MPM Prefork ..... 208
  - 2.1 Installation de PHP ..... 208
    - 2.1.1 Installation avec le gestionnaire de paquets Debian ..... 208
    - 2.1.2 Installation depuis les sources ..... 216
  - 2.2 Activation et configuration du module PHP ..... 223
    - 2.2.1 Activation du module PHP ..... 223
    - 2.2.2 Configuration d'Apache pour PHP ..... 226
- 3. PHP avec Apache en MPM Worker ..... 229
  - 3.1 Installation de PHP ..... 230
  - 3.2 Configuration d'Apache en MPM Worker ..... 232
  - 3.3 Installation d'Apache en MPM Worker (depuis les sources) ..... 233

3.4	Installation et configuration de mod_fcgid	236
3.4.1	Installation du module	236
3.4.2	Configuration du module	238
4.	PHP avec Apache en MPM Event	242
4.1	Installation de PHP-FPM	242
4.1.1	Distribution Linux Ubuntu/Debian	243
4.1.2	Installation depuis les sources	244
4.2	Installation Apache en MPM Event	249
4.3	Activation et configuration de mod_proxy_fcgi	251
5.	Installation de MySQL	253
5.1	Distribution Ubuntu et Debian/Linux	253
5.2	Distribution CentOS/Red Hat	256

## Chapitre 7

### Apache en tant que serveur reverse proxy

1.	Introduction	259
2.	Mise en place d'un reverse proxy	260
2.1	Activation du module	260
2.2	Configuration du module mod_proxy	260
2.2.1	ProxyPass	260
2.2.2	ProxyPassReverse	262
2.2.3	ProxyAddHeaders	263
2.2.4	ProxyVia	264
2.2.5	ProxyPassMatch	265
2.2.6	ProxyPassReverseCookieDomain	265
2.2.7	ProxyPassReverseCookiePath	266
2.2.8	ProxyPreserveHost	266

- 3. Mise en place d'un reverse proxy/load balancer . . . . . 267
  - 3.1 Activation du module . . . . . 267
  - 3.2 Configuration du module mod\_proxy\_balancer . . . . . 268
    - 3.2.1 Répartition de charge . . . . . 269
    - 3.2.2 Pondération dans la répartition de charge . . . . . 271
  - 3.3 Serveur de secours. . . . . 272
    - 3.3.1 En cas d'indisponibilité de services . . . . . 272
    - 3.3.2 En cas d'erreur applicative/code HTTP . . . . . 274
  - 3.4 Répartition de charge avec abonnement utilisateur (stickyness) . . . . . 276
  - 3.5 Interface de gestion Load Balancer Manager . . . . . 281

**Chapitre 8**  
**Notions de sécurité**

- 1. Introduction . . . . . 283
- 2. Permissions de répertoire . . . . . 284
- 3. Chroot d'Apache . . . . . 284
  - 3.1 Installation d'Apache dans le chroot . . . . . 285
  - 3.2 Configuration d'Apache et VirtualHost . . . . . 288
- 4. Protection d'accès . . . . . 293
  - 4.1 Autorisation d'accès . . . . . 293
    - 4.1.1 Contrôle d'accès en fonction du réseau . . . . . 293
    - 4.1.2 Contrôle d'accès en fonction de variables . . . . . 295
  - 4.2 Authentification Basic et Digest . . . . . 296
    - 4.2.1 Module mod\_auth\_basic. . . . . 296
    - 4.2.2 Module mod\_auth\_digest . . . . . 297
  - 4.3 Les fournisseurs d'authentification . . . . . 297
    - 4.3.1 Authentification par fichier de mot de passe . . . . . 297
    - 4.3.2 Authentification à la manière des sites FTP anonymes . . . . . 299
    - 4.3.3 Authentification avec une base de données MySQL . . 300

5.	Protection contre les attaques et limitation du trafic . . . . .	305
5.1	Protection contre les attaques de type "dédi de service" . . . . .	305
5.1.1	Ajustement du paramétrage d'Apache . . . . .	306
5.1.2	Module mod_reqtimeout . . . . .	306
5.1.3	Module mod_security . . . . .	307
5.2	Protection contre des attaques diverses et variées . . . . .	315
5.2.1	Jeux de règles de base . . . . .	316
5.2.2	Jeux de règles orientés application . . . . .	317
5.2.3	Jeux de règles orientés spam et usurpation d'identité . . . . .	318
5.3	Limitation de trafic. . . . .	319
5.3.1	Module Apache mod_ratelimit . . . . .	319
6.	Protocole de sécurisation SSL/TLS . . . . .	320
6.1	Présentation et historique . . . . .	320
6.2	Utilisation avec Apache . . . . .	322
6.2.1	Génération de certificats . . . . .	322
6.2.2	Activation du module mod_ssl . . . . .	325
6.2.3	Configuration du module mod_ssl . . . . .	325

## Chapitre 9

### Optimisation du service Apache

1.	Introduction . . . . .	329
2.	Gestion des ressources (matérielles, système, réseau) . . . . .	330
2.1	Mémoire vive . . . . .	330
2.1.1	Charger les modules strictement nécessaires . . . . .	330
2.1.2	Limiter les ressources consommées par Apache . . . . .	333
2.2	Stockage/disques durs . . . . .	335
2.2.1	Directives FollowSymLinks et SymLinksIfOwnerMatch . . . . .	335
2.2.2	Directive AllowOverride . . . . .	337
2.2.3	Directive DirectoryIndex . . . . .	337
2.2.4	Optimiser ou éliminer l'écriture de logs . . . . .	339

**Chapitre 10**  
**Les outils d'Apache**

- 1. Introduction ..... 341
- 2. apachectl ..... 342
- 3. ab ..... 345
- 4. apxs ..... 346
- 5. dbmmanage ..... 347
- 6. fcgistarter ..... 348
- 7. htcacheclean ..... 348
- 8. htdigest ..... 350
- 9. htdbm ..... 351
- 10. htpasswd ..... 352
- 11. httxt2dbm ..... 353
- 12. logresolve ..... 353
- 13. rotatelog ..... 353

**Chapitre 11**  
**Cas pratiques**

- 1. Introduction ..... 355
- 2. Environnement et contexte d'installation ..... 355
  - 2.1 Prérequis système ..... 355
  - 2.2 Apache ..... 356
  - 2.3 MySQL ..... 356
- 3. Hébergement d'un CMS PHP ..... 356
  - 3.1 CMS eZ Publish ..... 356
  - 3.2 Prérequis du CMS ..... 357
  - 3.3 Préparation du système ..... 358
  - 3.4 Installation de PHP avec les prérequis ..... 359

3.5	Configuration d'Apache	363
3.6	Création de la base de données et du compte utilisateur MySQL	366
3.7	Installation d'eZ Publish	366
4.	Utilisation de plusieurs versions de PHP sur un même serveur Apache	376
4.1	CMS WordPress	376
4.2	Prérequis de WordPress	377
4.3	Installation de PHP	377
4.4	Préparation du système	381
4.5	Configuration d'Apache	382
4.6	Création de la base de données et du compte utilisateur MySQL	384
4.7	Installation de WordPress	384
	Index	391

# Chapitre 5

## Gestion des logs sous Apache

### 1. Activation et configuration

Le service Apache possède nativement des fonctions de journalisation (écriture dans des fichiers logs) permettant de collecter diverses informations, telles que :

- l'état du service,
- les erreurs de configuration,
- les dépendances non satisfaites,
- d'autres informations liées aux applications hébergées.

La journalisation est primordiale pour assurer un suivi opérationnel du service, mais peut également constituer une véritable problématique en matière de stockage et de performance.

Il s'agit donc de paramétrer cette journalisation en fonction de ses propres besoins, de l'environnement (production, intégration, recette) et d'utiliser des modules d'Apache spécifiques apportant une granularité supplémentaire.

## 1.1 Journaux d'erreur

Par défaut, Apache enregistre des informations liées aux arrêts/démarrage du service ainsi qu'aux erreurs via la directive `ErrorLog`.

Cette directive est documentée dans le chapitre Configuration de base d'Apache, à la section Directives Core.

```
■ ErrorLog "/var/log/apache2.4/error_log"
```

La directive `ErrorLog` nous permet ici de préciser où seront enregistrées les informations du service.

### 1.1.1 Utilisation de LogLevel

Par défaut, lors d'un redémarrage de service, voici le niveau de journalisation observé dans le fichier `/var/log/apache2.4/error_log` :

```
[Mon Oct 20 12:21:32 2014] [notice] [pid 20911] event.c(2855):  
[client AH00494: SIGHUP received. Attempting to restart  
[Mon Oct 20 12:21:33.001849 2014] [mpm_event:notice] [pid  
20911:tid 139883230811904] AH00489: Apache/2.4.10 (Unix)  
ENI_server configured -- resuming normal operations  
[Mon Oct 20 12:21:33.001875 2014] [core:notice] [pid 20911:tid  
139883230811904] AH00094: Command line:  
'/opt/prod/apache2.4/bin/httpd'
```

Nous pouvons constater que seules des informations de niveau « notice » sont enregistrées dans le fichier de logs.

Les principaux niveaux sont :

Niveau	Description
emerg	Urgences - le système est inutilisable
alert	Des mesures doivent être prises immédiatement
crit	Conditions critiques
error	Erreurs
warn	Avertissements
notice	Événements importants mais normaux

Niveau	Description
info	Informations
debug	Messages de débogage

Passons à présent en mode `info` grâce à la directive `LogLevel` :

```
LogLevel info
```

Après redémarrage du service, on a :

```
[Mon Oct 20 12:32:19.475407 2014] [mpm_event:notice] [pid 20911:tid 139883230811904] AH00494: SIGHUP received. Attempting to restart
[Mon Oct 20 12:32:19.478174 2014] [unique_id:info] [pid 20911:tid 139883230811904] AH01566: using ip addr 127.0.1.1
[Mon Oct 20 12:32:20.000686 2014] [session_crypto:info] [pid 20911:tid 139883230811904] AH01849: The crypto library 'openssl' was loaded successfully
[Mon Oct 20 12:32:20.001547 2014] [mpm_event:notice] [pid 20911:tid 139883230811904] AH00489: Apache/2.4.10 (Unix) ENI_server configured -- resuming normal operations
[Mon Oct 20 12:32:20.001557 2014] [mpm_event:info] [pid 20911:tid 139883230811904] AH00490: Server built: Nov 16 2014 14:12:11
[Mon Oct 20 12:32:20.001565 2014] [core:notice] [pid 20911:tid 139883230811904] AH00094: Command line:
'/opt/prod/apache2.4/bin/httpd'
```

Avec un `LogLevel debug`, on a :

```
[Mon Oct 20 12:23:29.238600 2014] [mpm_event:notice] [pid 20911:tid 139883230811904] AH00494: SIGHUP received. Attempting to restart
[Mon Oct 20 12:23:29.241396 2014] [unique_id:info] [pid 20911:tid 139883230811904] AH01566: using ip addr 127.0.1.1
[Mon Oct 20 12:23:30.000857 2014] [session_crypto:info] [pid 20911:tid 139883230811904] AH01849: The crypto library 'openssl' was loaded successfully
[Mon Oct 20 12:23:30.000903 2014] [:debug] [pid 20911:tid 139883230811904] mod_security2.c(595): SecServerSignature: Changed server signature to "ENI_server".
[Mon Oct 20 12:23:30.001706 2014] [mpm_event:notice] [pid 20911:tid 139883230811904] AH00489: Apache/2.4.10 (Unix) ENI_server configured -- resuming normal operations
[Mon Oct 20 12:23:30.001717 2014] [mpm_event:info] [pid 20911:tid
```

```
139883230811904] AH00490: Server built: Nov 16 2014 14:12:11
[Mon Oct 20 12:23:30.001725 2014] [core:notice] [pid 20911:tid
139883230811904] AH00094: Command line:
'/opt/prod/apache2.4/bin/httpd'
[Mon Oct 20 12:23:30.002052 2014] [mpm_event:debug] [pid 15151:tid
139883131455232] event.c(2009): AH02471: start_threads: Using
epoll
[Mon Oct 20 12:23:30.002212 2014] [mpm_event:debug] [pid 15150:tid
139883131455232] event.c(2009): AH02471: start_threads: Using
epoll
[Mon Oct 20 12:23:30.002680 2014] [mpm_event:debug] [pid 15152:tid
139883131455232] event.c(2009): AH02471: start_threads: Using epoll
```

La quantité d'information collectée n'est donc pas la même en fonction du niveau choisi via la directive `LogLevel`.

En effet, pour un simple redémarrage de service :

- Trois lignes sont écrites dans le fichier de logs avec un `LogLevel` par défaut (`warn`).
- Six lignes pour un `LogLevel` positionné à `info`.
- Neuf lignes pour un `LogLevel` positionné à `debug`.

Bien entendu, il s'agit de sélectionner le niveau de journalisation en fonction du besoin. Ainsi, compte tenu de la quantité d'information collectée par les différents `LogLevel`, et notamment en mode `debug`, il convient de l'activer uniquement dans le cadre d'une analyse « bas niveau » du service pour effectuer un diagnostic.

Car un fichier log d'erreur est efficace à condition d'y trouver uniquement les informations utiles, voire en cas de fort trafic, uniquement celles qui sont critiques.

Il est important de noter que la volumétrie d'événements journalisés a également un impact direct sur la performance et la maintenance des plateformes :

- nombre d'accès disque lors de l'écriture des logs,
- trafic réseau, dans le cas d'une externalisation logs,
- volumétrie importante des fichiers logs, et donc utilisation de l'espace disque

### 1.1.2 Utilisation de ErrorLogFormat

Un autre moyen pour optimiser la journalisation des erreurs d'Apache est d'utiliser la directive `ErrorLogFormat`.

Comme abordé dans le chapitre Configuration de base d'Apache, il est possible de sélectionner les chaînes de format que nous souhaitons voir apparaître dans les fichiers de logs.

Pour illustrer un cas d'erreur, nous avons chargé le module `mod_security` sans le module `mod_unique_id` qui est une dépendance.

Ci-dessous, la sortie de logs par défaut lors d'un redémarrage de service et après une requête HTTP :

```
[Mon Oct 20 13:26:27 2014] [notice] [pid 20911] [client AH00494:
SIGHUP received. Attempting to restart
[Mon Oct 20 13:26:27.342048 2014] [mpm_event:notice] [pid
20911:tid 139883230811904] AH00489: Apache/2.4.10 (Unix)
ENI_server configured -- resuming normal operations
[Mon Oct 20 13:26:27.342064 2014] [core:notice] [pid 20911:tid
139883230811904] AH00094: Command line:
'/opt/prod/apache2.4/bin/httpd'
[Mon Oct 20 13:26:36.780030 2014] [:error] [pid 16356:tid
139883125167872] ModSecurity: ModSecurity requires mod_unique_id
to be installed.
```

Utilisons la directive `ErrorLogFormat` ci-dessous :

```
ErrorLogFormat "[%t] [%l] [pid %P] [client %a] %M"
```

`[%t]` affiche l'heure entre [ et ], sans les microsecondes.

`[%l]` affiche la sévérité du message entre [ et ].

`[pid %P]` affiche « pid » suivi du numéro du PID entre [ et ].

`[client %a]` affiche adresse IP et port clients.

`%M` affiche le message (obligatoire).

Nous obtenons :

```
[Mon Oct 20 13:26:02 2014] [notice] [pid 20911] [client AH00494:
SIGHUP received. Attempting to restart
[Mon Oct 20 13:26:02 2014] [notice] [pid 20911] [client AH00489:
```

```
Apache/2.4.10 (Unix) ENI_server configured -- resuming normal
operations
[Mon Oct 20 13:26:02 2014] [notice] [pid 20911] [client AH00094:
Command line: '/opt/prod/apache2.4/bin/httpd'
[Mon Oct 20 13:26:09 2014] [error] [pid 16264] [client
ModSecurity: ModSecurity requires mod_unique_id to be installed.
```

L'information reste tout aussi lisible que précédemment, mais avec un message épuré.

Ici, l'objectif est de montrer qu'il est possible de formater les entrées du fichier `ErrorLog` de manière à ne stocker que l'essentiel.

Mais la directive `ErrorLogFormat` permet également de mettre en évidence des informations supplémentaires qui peuvent être importantes à journaliser.

Dans le cadre de l'hébergement de plusieurs sites web, avec des noms de domaine différents, l'ajout du nom du « `ServerName` » ayant généré une erreur permettra d'affiner le diagnostic :

```
ErrorLogFormat "[%t] [%v] [%l] [pid %P] [client %a] %M"
```

Le résultat obtenu :

```
[Mon Oct 20 13:34:56 2014] [vm-compilation-eni] [error] [pid
16467] [client ModSecurity: ModSecurity requires mod_unique_id to
be installed.
[Mon Oct 20 13:35:29 2014] [eni.labs] [error] [pid 16465] [client
ModSecurity: ModSecurity requires mod_unique_id to be installed.
```

## 1.2 Module `mod_log_config`

Le module `mod_log_config` permet de journaliser les requêtes traitées par le serveur en dehors de celles inscrites dans l'`ErrorLog`.

### 1.2.1 Utilisation de la directive `LogFormat`

Comme cela a été vu dans le chapitre Configuration de base d'Apache, la directive `LogFormat` permet de définir le formatage des fichiers de logs concernant les requêtes traitées par le serveur Apache. Tout comme pour les `ErrorLog`, il est possible d'optimiser la quantité d'information journalisée et de faire apparaître certaines informations en fonction de ses besoins.