

[PHPLinux版] 監視モニター I

初期化: どんなものか作ってみよう

[PHPLinux] 監視モニター初期化

ビジュアル監視の初期化の必要性

監視アプリを正常に動作させるため、動作環境の整合性を確保する

初期化はいつ実施するのか

OSおよび監視アプリに必要な関連ソフトのインストール時

監視アプリのインストールハードウェアの変更時

データベース再構築時

監視アプリおよび関連ソフトの移行時

その他、必要に応じて実行

初期化する手段

変数、パラメータを設定、初期化アプリを実行する

難易度

初期化に必要なPHPパッケージインストールが出来れば構築出来ますが
トラブル発生時の対応にはPHPおよびシェルスクリプトの知識も必要です
また、拡張機能カスタマイズには次の知識が必要です

snmpの知識

監視およびグラフ取得する場合は、監視対象にsnmpマネージャと
snmpエージェントサービスが必要です

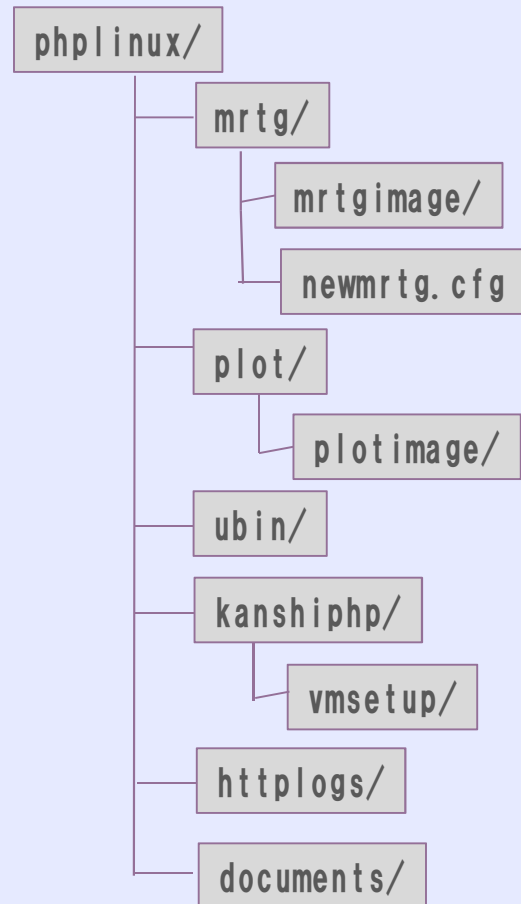
メールの知識

イベントをメールサーバへ送信するためのメールサーバーの
構築、設定が必要です、但し認証なしの25ポートまたは
SMTP認証の587ポートのサーバがあれば利用出来ます

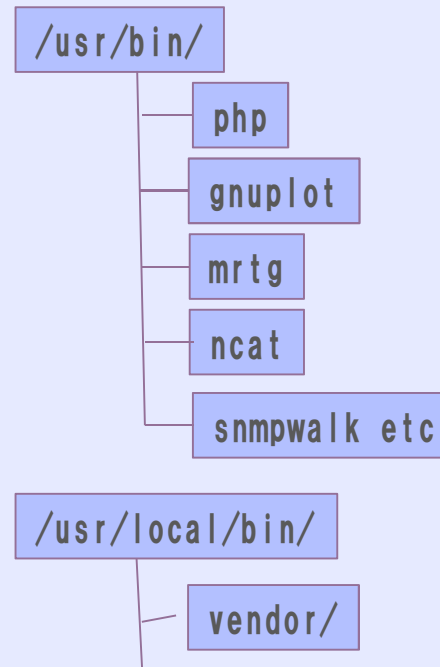


[PHPLinux] 監視ディレクトリ構成

監視アプリ



パッケージ



動作確認パッケージバージョン

php-5.6
gnuplot-4.6.2
mrtg-2.17.4
httpd-2.4.6
PHPMailer
Net-Snmp-5.7.2
MariaDB-5.5

[PHPLinux] 監視システム構築手順

1. GithubからCLONE
2. パッケージインストール
3. 監視アプリをコピー
4. 変数、パラメータ設定
5. 監視アプリで初期化
6. 最初のログイン
7. 構築後の機能制限
8. 機能拡張
9. 初期化後画面と構成

[PHPLinux] 監視システム構築手順

1. Github からCLONE

CLONE用のディレクトリvisualmonitorを作成、そこへCLONEする

```
$ sudo git clone https://github.com/ossiansunny/visualmonitor.git
```

必要なディレクトリは、

```
visualmonitor/phplinux/kanshiphp/  
visualmonitor/phplinux/ubin/  
visualmonitor/phplinux/mrtg/  
visualmonitor/phplinux/plot/
```

2. インストール先パッケージとアプリケーションの配置 例

/var/www/html/kanshiphp/	監視アプリ
/var/www/html/kanshiphp/vmsetup/	監視アプリ初期化
/var/www/html/mrtg/newmrtg.cfg	mrtgコンフィグ
/var/www/html/mrtg/mrtgimage/	mrtgグラフィイメージ
/var/www/html/plot/plotimage/	gnuplotグラフィイメージ
/var/www/html/httplogs/	webログ
/var/www/html/ubin/	シェルスクリプト
/usr/bin/mrtg	mrtgプログラム
/usr/bin/php	phpプログラム
/usr/bin/gnuplot	gnuplotプログラム
/usr/bin/snmpwalk ...	net-snmpプログラム
/usr/local/bin/vendor/	PHPMailer

3. 監視アプリのコピー

任意のディレクトリで構築する場合はGithub CLONE構成の監視アプリをコピーする

コピー元	コピー先
/visualmonitor/phplinux/kanshiphp/	/var/www/html/kanshiphp/
/visualmonitor/phplinux/mrtg/	/var/www/html/mrtg/
/visualmonitor/phplinux/plot/	/var/www/html/plot/
/visualmonitor/phplinux/ubin/	/var/www/html/ubin/

[PHPLinux] 監視システム構築手順

4. 変数、パラメータ設定

4.1 シェルPATH変数にパッケージ関連のパスを追加 例
/usr/local/bin etc

4.2 監視アプリPATH変数設定 例

```
$vpath_kanshiphp/vmsetup/kanshiphp.ini
vpath_php      = "/usr/bin/php"
vpath_base     = "/var/www/html"
vpath_kanshiphp = "/var/www/html/kanshiphp"
vpath_mrtgbase = "/var/www/html/mrtg"
vpath_mrtghome = "/var/www/html/mrtg"
vpath_plothome = "/var/www/html/plot"
vpath_weblog   = "/var/www/html/httplogs"
vpath_htdocs   = "/var/www/html"
vpath_kanshibin = "/var/www/html/ubin"
vpath_phpmailer = "/usr/local/bin"
vpath_gnuplotbin = "/usr/bin"
```

4.3 Mysql初期化変数

4.3.1 kanshiphp用設定パラメータ

```
$vpath_kanshiphp/vmsetup/mysqlsetup.cfg
[mysql]
kanshi_host= "localhost"    ... Kanshi用ホスト (変更可)
kanshi_user= "kanshiadmin"  ... Kanshi用ユーザー (変更可)
kanshi_pass= "kanshipass"   ... Kanshi用ユーザーパスワード注 (変更可)
kanshi_db= "kanshi"        ... Kanshi用DB名 "kanshi" であること (変更不可)
```

注: validate_password_policy= "LOW",
validate_password_special_char_count=0 になっていること

4.3.2 既存のmysql設定パラメータ

\$vpath_kanshiphp/vmsetup/sqlexisting.cfg パーミッションは400

```
[client]
host= "xxxxxxx"           ... 既存のHost名
user= "xxxxxx"            ... 既存のuser
password = "xxxxxxx"      ... 既存のuserパスワード (パスワード無い場合はこのキーワード不要)
```


[PHPLinux] 監視システム構築手順

5. 監視アプリの初期化

5.1 初期化するもの

○mysql 監視ユーザ、パスワード、DB、テーブル作成、初期データ挿入、mysqlkanshi.php作成、mrtgのWorkDir作成
日本語を扱う場合、mysql構成パラメータに設定して置くこと（例）

```
mysql> show variables like "char%";
+-----+-----+
| Variable_name | Value                                |
+-----+-----+
| character_set_client | cp932                               |
| character_set_connection | cp932                               |
| character_set_database | utf8 または utf8mb4                |
| character_set_filesystem | binary                             |
| character_set_results | cp932                               |
| character_set_server | utf8 または utf8mb4                |
| character_set_system | utf8                                |
+-----+-----+
```

5.2 シェルで初期化実行

phpパッケージにphp-mysqlが必要

```
$ cd $vpath_kanshiphp/vmsetup/
$ sudo php init-mysql.php           ... ユーザ、パスワード、DB、mysqlkanshitmp.php作成
$ sudo php init-mysqldbset.php      ... テーブル作成
$ sudo php init-mysqlinsert.php     ... データ挿入
$ sudo php init-workdir.php         ... Workdir設定
$ sudo php init-instcheck.php       ... mysqlkanshi.php再作成 再度実行する時は、init-mysql.phpから
```

5.3 webのログをローテイトさせる為、httpd.confを下記部分を変更する

```
ErrorLog "| bin/rotatelog.exe <$vpath_weblog/error_%Y%m%d.log 86400 540 "
CustomLog "| bin/rotatelog.exe <$vpath_weblog>/access_%Y%m%d.log 86400 540" combined
```

5.4 パーMISSIONの変更

次のディレクトリおよび配下をアプリケーションサーバの所有者:グループにし、実行権限を777にする

次の例は、apacheアプリケーションサーバの場合

```
$ cd $vpath_htdocs
$ sudo chown -R apache:apache kanshiphp ; sudo chmod -R 777 kanshiphp
$ sudo chown -R apache:apache mrg       ; sudo chmod -R 777 mrtg
$ sudo chown -R apache:apache plot      ; sudo chmod -R 777 plot
$ sudo chown -R apache:apache httplogs  ; sudo chmod -R 777 httplogs
$ sudo chown -R apache:apache ubin      ; sudo chmod -R 777 ubin
```

[PHPLinux] 監視システム構築手順

6. 最初のログイン

httpd, mysql 起動、<http://localhost>でhttpd起動確認
確認出来たら、
ログイン

<http://localhost/kanshiphp/login.php>

user:admin password:manager

7. 構築後の機能制限

○メール送信は不可

メニュー「メール設定：送信」

○ホスト死活監視は、「ping監視」のみ

メニュー「監視ホスト追加」「監視ホスト修正」の死活

○グラフ関連は不可

メニュー「リソースグラフ」「プロットグラフ」「プロットログ」

8. 機能拡張

documentsのPDF、監視アプリ内マニュアルページ参照

[PHPLinux] 監視システム構築手順

9. 初期化後画面と構成

