

Apache

概要

本項目では、Apacheの構築とログの確認手法を習得する。
それに伴う、デーモン管理コマンドやFirewallの操作についても習得する。

目次

1. Apacheのテストページ表示
 - apacheインストール
 - apacheの起動と停止
 - ポート開放
 - ブラウザでテストページ確認
2. index.html、test.php表示
 - 設定ファイルの確認とバックアップの作成
 - サイトのホームディレクトリの変更の仕方
 - AddHandlerの追加によるCGIファイルの認識
 - autoindex.confを編集
 - index.htmlをブラウザで表示
 - test.phpを表示
3. ログの確認
 - Webサーバのログ
 - アクセスログ
 - エラーログ

手順1：Apacheのテストページ表示

Apacheとは

Webサイトを配信するサーバを WWWサーバ と呼ぶ。
その中の1つとして、Apache HTTP Serverがある。
機能が豊富で柔軟な設定ができる。

Apacheインストール

- **\$ sudo yum -y update**
- **\$ sudo yum -y install httpd**
 - ユーザのパスワードを要求されるので入力
 - complete! と表示されればOK

```
[furuhamama@localhost ~]$ sudo yum -y install httpd

We trust you have received the usual lecture from the local System
Administrator. It usually boils down to these three things:

    #1) Respect the privacy of others.
    #2) Think before you type.
    #3) With great power comes great responsibility.

[sudo] password for furuhama: _
```

余談であるが、yumコマンドはLinuxのパッケージを管理するためのコマンドである。

他にもLinuxのパッケージを管理するコマンドにはaptがある。yumとの違いは**パッケージ形式の違い**である。

yumのパッケージ形式はrpm形式であり、RedHatやCentOS形式のディストリビューションで用いられる。aptのパッケージ形式は.deb形式であり、DebianやUbuntu形式のディストリビューションで用いられる。

- **\$ httpd -version**
 - インストールされた Apache httpd のバージョンを確認

```
[furuhamama@localhost ~]$ httpd -version
Server version: Apache/2.4.6 (CentOS)
Server built:   Apr  2 2020 13:13:23
```

Apacheの起動と停止

- Apacheを起動
- **\$ sudo systemctl start httpd.service**
 - ユーザのパスワードを要求されるので入力
 - もし起動しない場合、CentOS7をシャットダウンし、もう一度起動すると良い

```
[furuhamama@localhost ~]$ sudo systemctl start httpd.service
[sudo] password for furuhama:
```

- Apacheが起動できたかどうか確認
- **\$ sudo systemctl status httpd.service**
 - Active: active (running)と表示されていることを確認

```
[furuhamama@localhost ~]$ sudo systemctl status httpd.service
■ httpd.service - The Apache HTTP Server
  Loaded: loaded (/usr/lib/systemd/system/httpd.service; disabled; vendor preset: enabled)
  Active: active (running) since Mon 2020-11-09 17:59:10 JST; 1min 6s ago
  Docs: man:httpd(8)
        man:apachectl(8)
  Main PID: 1534 (httpd)
  Status: "Total requests: 0; Current requests/sec: 0; Current traffic: 0"
  CGroup: /system.slice/httpd.service
          └─1534 /usr/sbin/httpd -DFOREGROUND
            └─1535 /usr/sbin/httpd -DFOREGROUND
              └─1536 /usr/sbin/httpd -DFOREGROUND
                └─1537 /usr/sbin/httpd -DFOREGROUND
                  └─1538 /usr/sbin/httpd -DFOREGROUND
                    └─1539 /usr/sbin/httpd -DFOREGROUND

Nov 09 17:59:10 localhost.localdomain systemd[1]: Starting The Apache HTTP Server: httpd.
Nov 09 17:59:10 localhost.localdomain httpd[1534]: AH00558: httpd: could not open error log
Nov 09 17:59:10 localhost.localdomain systemd[1]: Started The Apache HTTP Server: httpd.
Hint: Some lines were ellipsized, use -l to show in full.
```

- psコマンドはauxオプションで実行し、起動中の全プロセスを表示
- パイプを通してgrepコマンドを使うことで、大量に表示される中から、httpdが出現する行のみ抽出
- **\$ ps aux | grep httpd**

```
[furuhamama@localhost ~]$ ps aux | grep httpd
root      9439  0.0  0.4 224080  5032 ?        Ss   19:51   0:01 /usr/sbin/httpd -DFOREGROUND
apache    9440  0.0  0.3 224212  3672 ?        S    19:51   0:00 /usr/sbin/httpd -DFOREGROUND
apache    9441  0.0  0.3 224080  3136 ?        S    19:51   0:00 /usr/sbin/httpd -DFOREGROUND
apache    9442  0.0  0.2 224080  2908 ?        S    19:51   0:00 /usr/sbin/httpd -DFOREGROUND
apache    9443  0.0  0.2 224080  2924 ?        S    19:51   0:00 /usr/sbin/httpd -DFOREGROUND
apache    9444  0.0  0.2 224080  2924 ?        S    19:51   0:00 /usr/sbin/httpd -DFOREGROUND
apache    9445  0.0  0.2 224080  2924 ?        S    19:51   0:00 /usr/sbin/httpd -DFOREGROUND
furuhamama 9468  0.0  0.0 112812   968 tty1    S+   20:52   0:00 grep --color=auto httpd
```

- Apacheを停止
- **\$ sudo systemctl stop httpd.service**
 - 何も表示されなくてOK
- **\$ sudo systemctl status httpd.service**
 - Active: inactive (dead)と表示されていることを確認

```
[furuhamama@localhost ~]$ sudo systemctl status httpd.service
■ httpd.service - The Apache HTTP Server
  Loaded: loaded (/usr/lib/systemd/system/httpd.service; disabled; vendor preset: disabled)
  Active: inactive (dead)
  Docs: man:httpd(8)
        man:apachectl(8)
```

解説

- CentOS7ではデーモンの制御制御対象のことをユニットと呼ぶ
 - デーモンとはUNIX系OSにおける常駐プログラムのこと

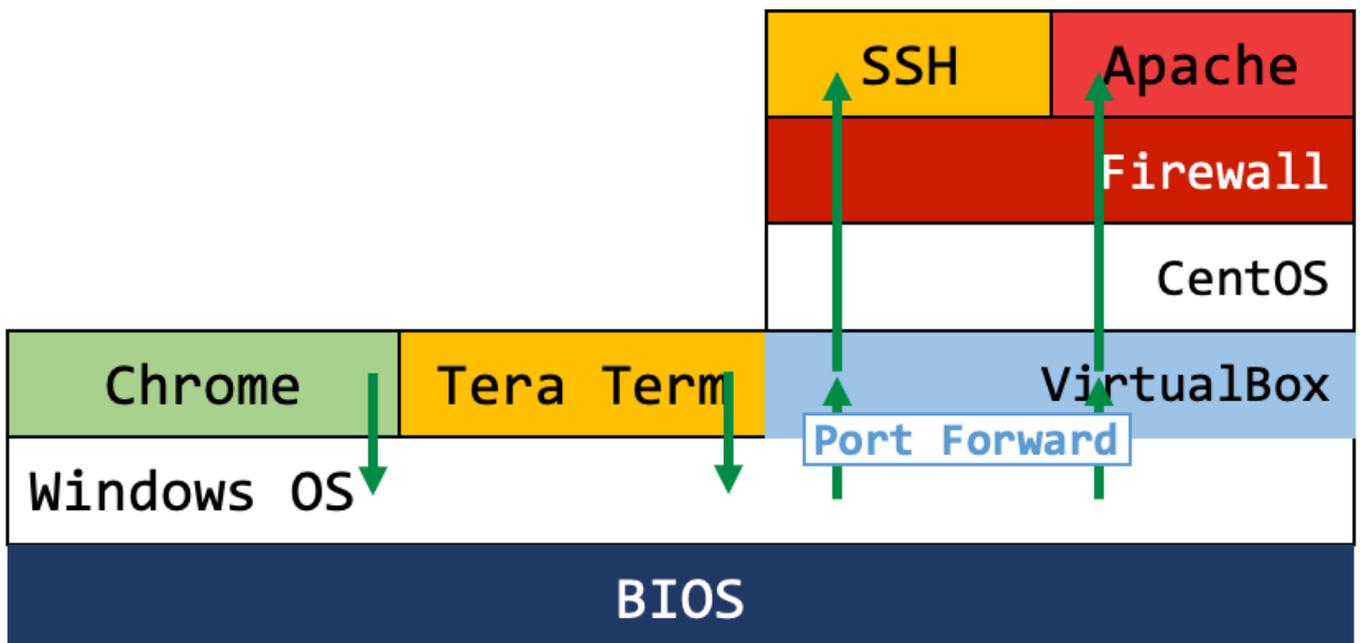
- ユニットにはいくつか種類があり、代表的なものとして、プロセスの起動と停止に関するserviceがある
 - 詳しくは参考文献を参照してほしい
- 各ユニットはsystemctlコマンドで制御できる
- psコマンドで確認すると、PID（プロセスID）が1番で動作していることが確認できる
- **\$ ps aux | head -n 5**

```
[furuhaman@localhost ~]$ ps aux | head -n 5
USER      PID %CPU %MEM    VSZ   RSS TTY      STAT START   TIME COMMAND
root         1  0.0  0.6 193680 6796 ?        Ss   Nov09    0:12 /usr/lib/systemd/systemd --switched
-rsroot --system --deserialize 22
root         2  0.0  0.0     0     0 ?        S    Nov09    0:00 [kthreadd]
root         4  0.0  0.0     0     0 ?        S<   Nov09    0:00 [kworker/0:0H]
root         6  0.0  0.0     0     0 ?        S    Nov09    0:02 [ksoftirqd/0]
```

ポート開放

概要

CentOS 7とFirewallのポート開放と転送(ポートフォワード)の設定を行う。



CentOS 7

- CentOS 7はデフォルトでfirewalldというファイアウォールが動作
- firewalld に httpd (80番ポート)への通信を許可
- **\$ sudo firewall-cmd --zone=public --add-service=http --permanent**
- **\$ sudo firewall-cmd --reload**

```
[furuhaman@localhost ~]$ sudo firewall-cmd --zone=public --add-service=http --permanent
success
[furuhaman@localhost ~]$ sudo firewall-cmd --reload
```

- **\$ sudo firewall-cmd --list-all**

- servicesにhttpがあることを確認

```
[furuhaman@localhost ~]$ sudo firewall-cmd --list-all
public (active)
  target: default
  icmp-block-inversion: no
  interfaces: enp0s3
  sources:
  services: dhcpv6-client http ssh
  ports:
  protocols:
  masquerade: no
  forward-ports:
  source-ports:
  icmp-blocks:
  rich rules:
```

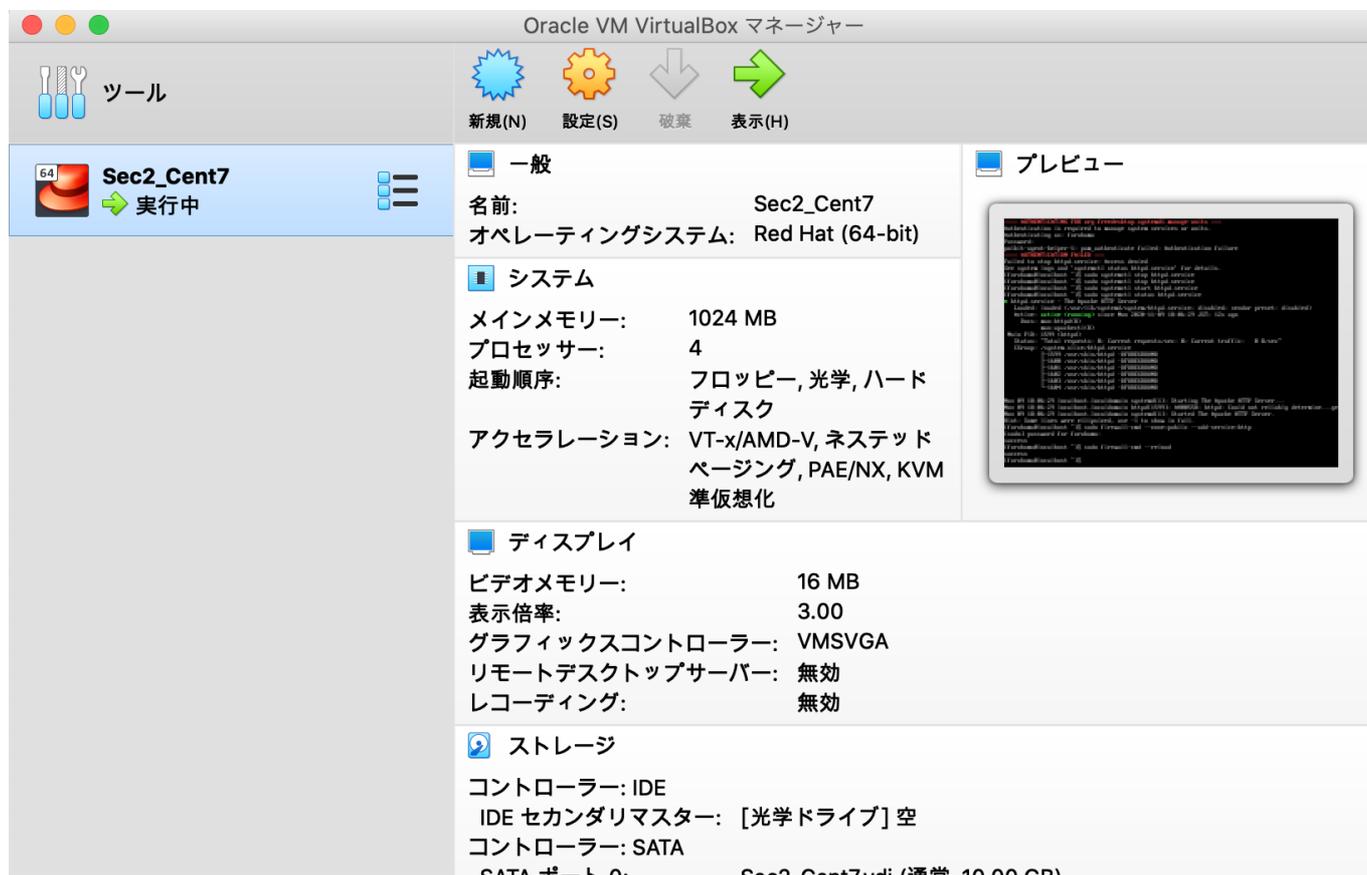
余談であるが、CentOS 6までは、iptablesを直接操作していた。

それを面倒に感じたTA竹原は、CentOS 7が出た当初、早速Firewalldを使ってみたが、Zoneの変更にバグがあり、結局面倒に感じたことを覚えている。

しかし、Debian系でよく使われるufwよりは好みであり、わざわざDebianにもFirewalldを導入している。これらは、結局のところ、内部的にiptablesを操作しており、フロントエンドが異なるだけである。

VirtualBox

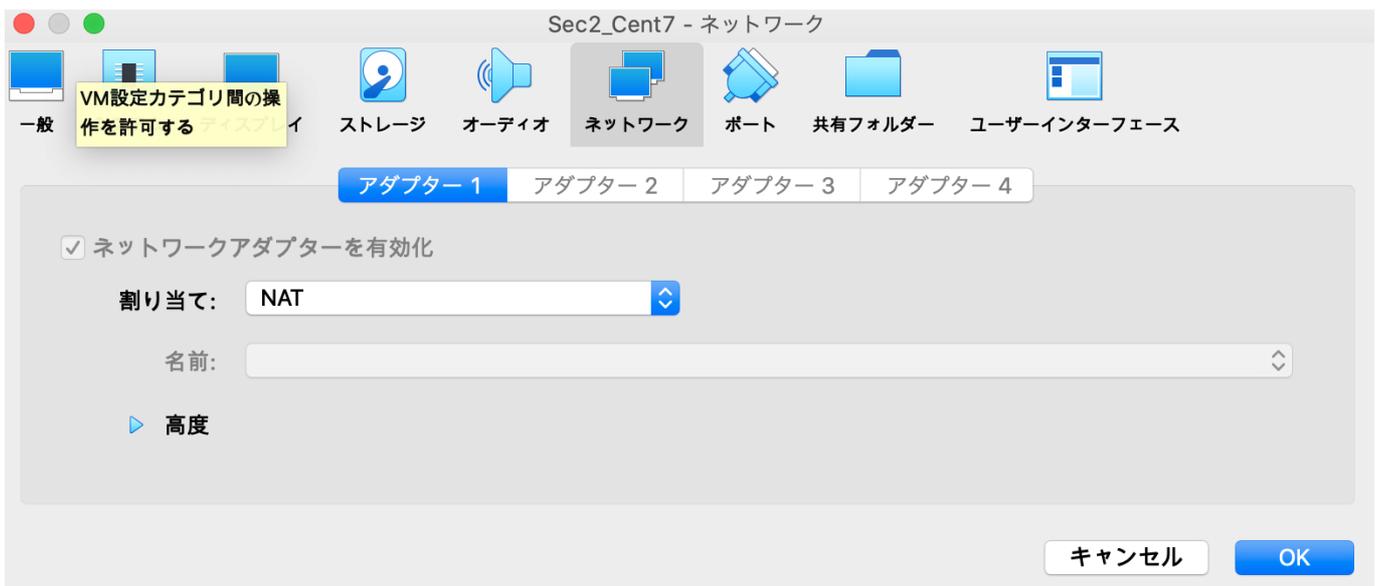
- 同様にVirtualBoxでも設定を行う
- VirtualBox マネージャー上部の設定を押す



- ネットワークを押す



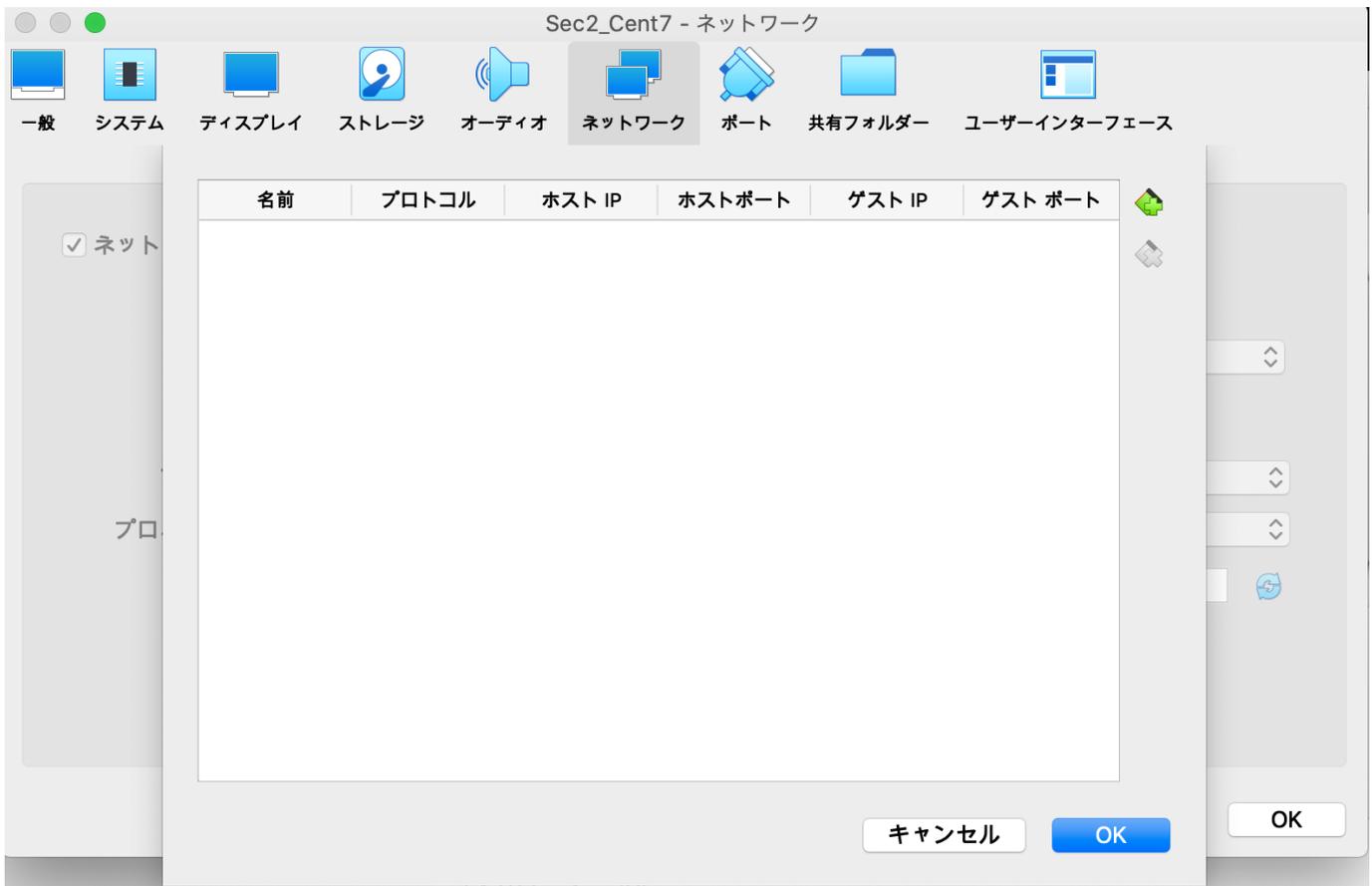
- 高度を押す



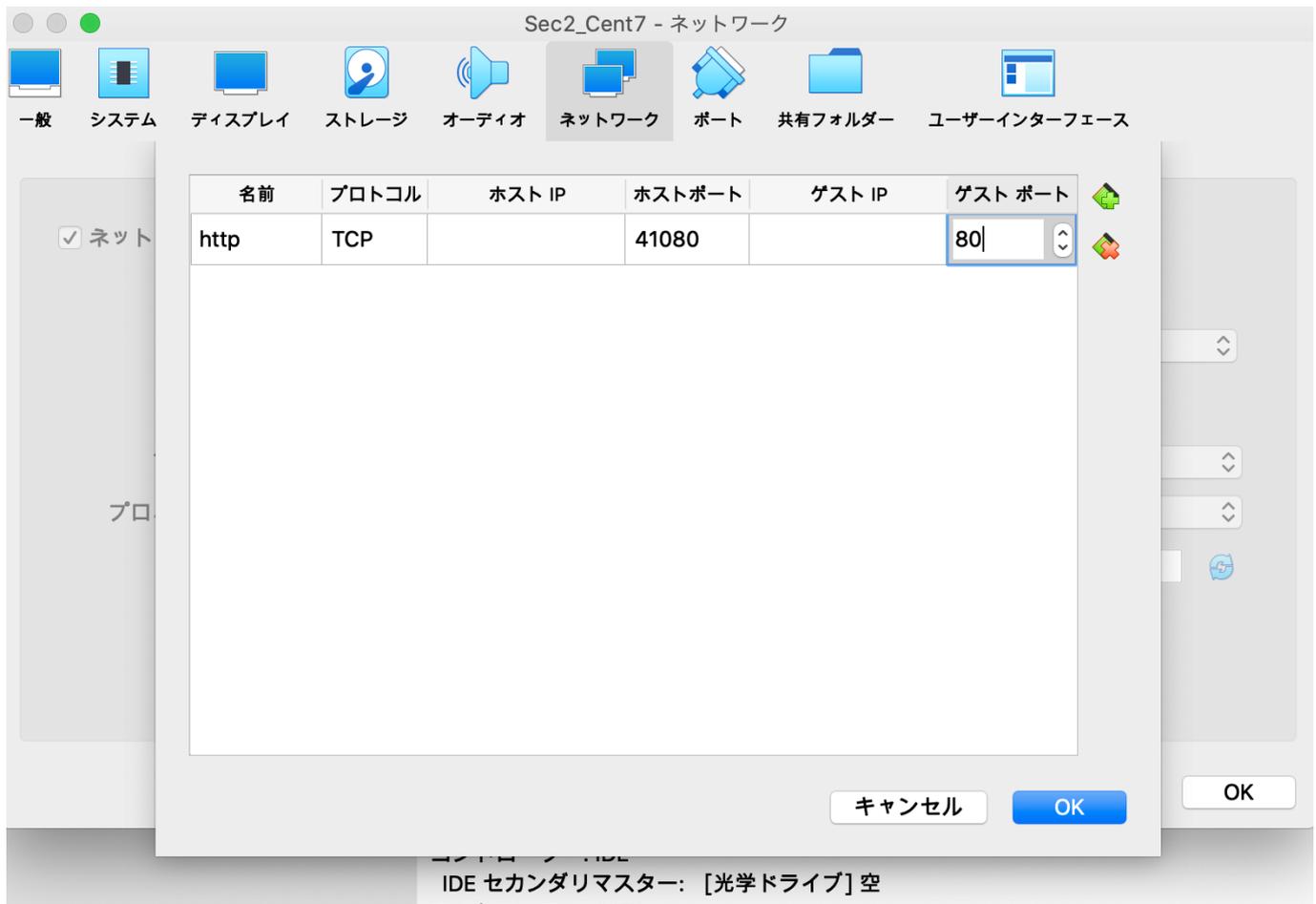
- ポートフォワーディングを押す



- 右側の緑色の+を押す

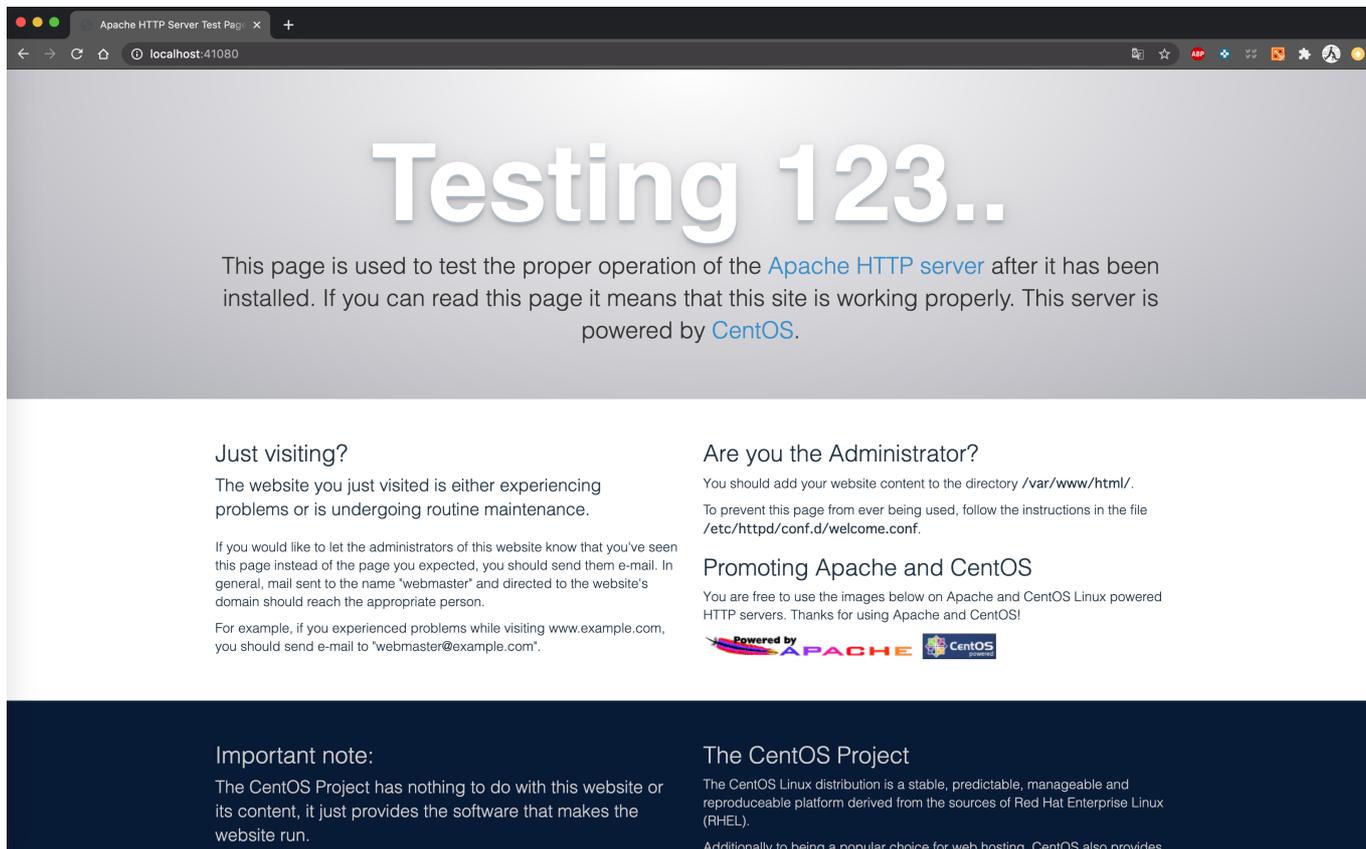


- 名前は適当に（例はhttp）
- ホストポート:41080、ゲストポート:80を入力



ブラウザでテストページ確認

- **\$ sudo systemctl start httpd.service**
- ブラウザで **http://localhost:41080** にアクセス
- 下記に画像が画面に表示されればOK



手順2 : index.html、test.php表示

設定ファイルの確認とバックアップの作成

- WebサーバApacheの設定は、`/etc/httpd/conf`と`/etc/httpd/conf.d`の下にある.confファイルを編集
- `httpd.conf`のバックアップ作成
 - `$ cd /etc/httpd/conf/`
 - `$ ls -l`
 - `$ sudo cp httpd.conf httpd.conf.orig`
 - `$ ls -l`
- `conf.d`のバックアップ作成
 - `$ cd /etc/httpd/`
 - `$ ls -l`
 - `$ sudo cp -r conf.d conf.d.orig`
 - `$ ls -l`

余談であるが、以下のように引数を短縮して記述することができる。

```
$ sudo cp httpd.conf{,.orig}
```

これは、`$ sudo cp httpd.conf httpd.conf.orig` と展開される。
また、

```
$ cp hoge{1,2}.txt
```

と入力すると、

```
$ cp hoge1.txt hoge2.txt
```

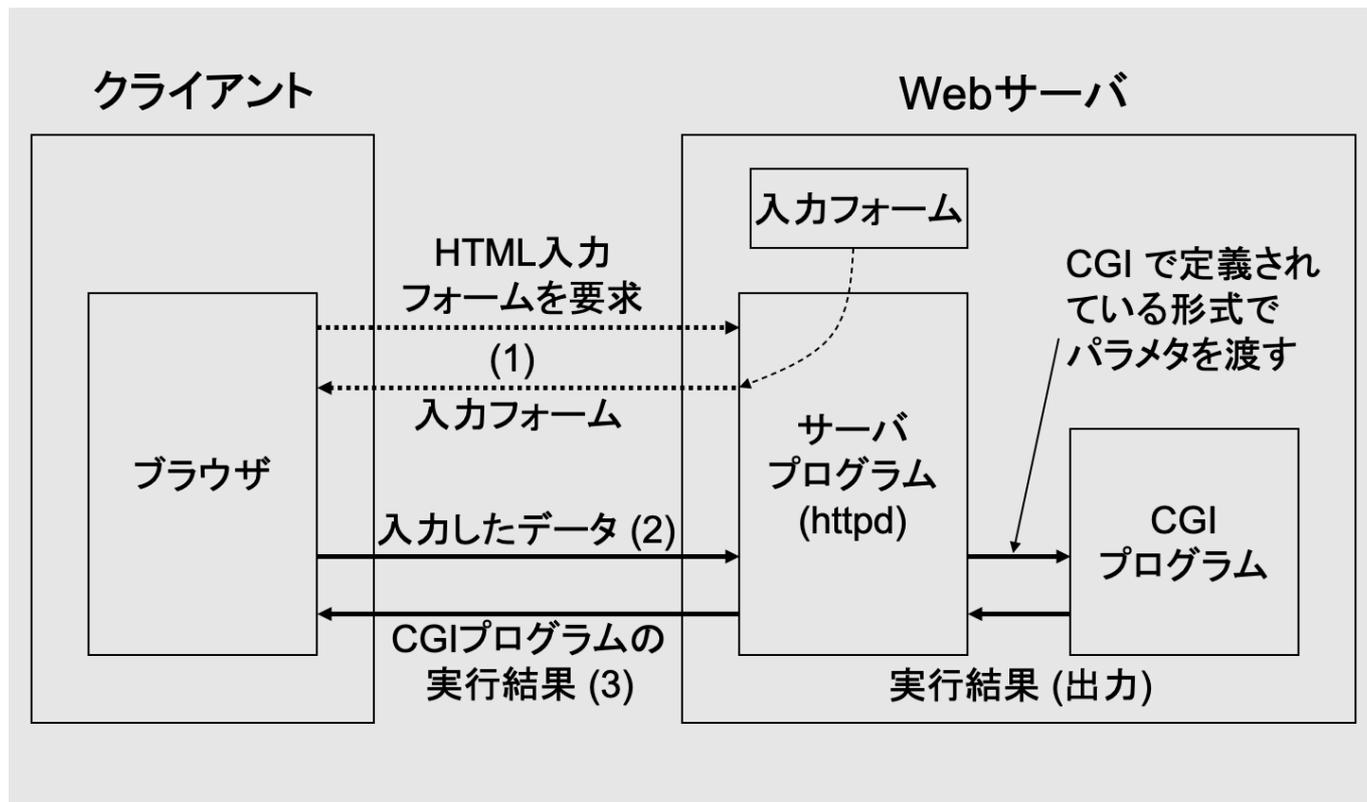
と展開される。

サイトのホームディレクトリの変更の仕方

- httpd.confではApacheの設定を行う
- httpd.confの中のDocumentRootではHTMLファイルを置くドキュメントルートを設定
 - 外部から見たサイトのルート
- 今回ドキュメントルートはデフォルトの/var/www/html/のまま作業をすすめるため、確認だけ行う
- **\$ cd /etc/httpd/conf**
- **\$ sudo vi httpd.conf**
- viで起動したら、/を入力し、DocumentRoot "/var/www/html"を検索
- DocumentRootのパスを変更を変更したい場合ここを編集（今回はしない）

AddHandlerの追加によるCGIファイルの認識

- CGIとはWebサーバ上のプログラムに問い合わせやデータをHTTPで送信するための仕組み
 - この仕様に基づいて作成されたプログラムをCGIプログラムと言い、CGIプログラムの内、コンパイル作業が不要なスクリプト言語で記述されたものをCGIスクリプトという（今回はPHP）
- 下図はブラウザ、サーバープログラム、CGIプログラムの概略を示したものである
- CGIを用いることで、あらかじめ用意されたHTMLを表示するだけでなく、アクセスカウンターやアンケートホームなど、Webページの内容を動的に変化させることができる



- 同様に/を入力し、AddHandlerで検索
- #AddHandler cgi-script .cgiの行をアンコメントアウトし、.phpを追加
 - コメントアウトとアンコメントアウトは#を先頭につけるかつけないかでできる
- さらに、Options +ExecCGIを追加
 - TAはここを書いていなかったためphpファイルが表示されず発狂してました (余談)
- :wqで保存

```
#
# AddHandler allows you to map certain file extensions to "handlers":
# actions unrelated to filetype. These can be either built into the server
# or added with the Action directive (see below)
#
# To use CGI scripts outside of ScriptAliased directories:
# (You will also need to add "ExecCGI" to the "Options" directive.)
#
AddHandler cgi-script .cgi .php
Options +ExecCGI
```

解説

- AddHandler cgi-script .cgi .phpでは、.cgiや.phpの拡張子がついたファイルをCGIとして実行した結果を返すように設定している
- Options +ExecCGIでは、CGIの実行を有効にしている

=====
 余談であるが、旧来は、CGIといえばPerlであった。

しかし、PHPやRubyの台頭によりその場を奪われつつある。

また、CGIはその特性上、近年では、CGI自体が減りつつある。

=====

autoindex.confを編集

- iconsディレクトリはデフォルトコンテンツで使う画像などが設置されているディレクトリ
- このディレクトリが有効な場合、そこに含まれるREADMEファイルなども表示されてしまうため、セキュリティ上、無効化することが好ましい
- CentOS 7ではautoindex.confで無効化できる
- **\$ cd /etc/httpd/conf.d**
- **\$ sudo vi autoindex.conf**
- viで起動したら、/を入力し、Alias /icons/ "/usr/share/httpd/icons/"を検索
- 下記をコメントアウトし、:wqで保存

```
# Alias /icons/ "/usr/share/httpd/icons/"
#
# <Directory "/usr/share/httpd/icons">
#   Options Indexes MultiViews FollowSymlinks
#   AllowOverride None
#   Require all granted
# </Directory>
```

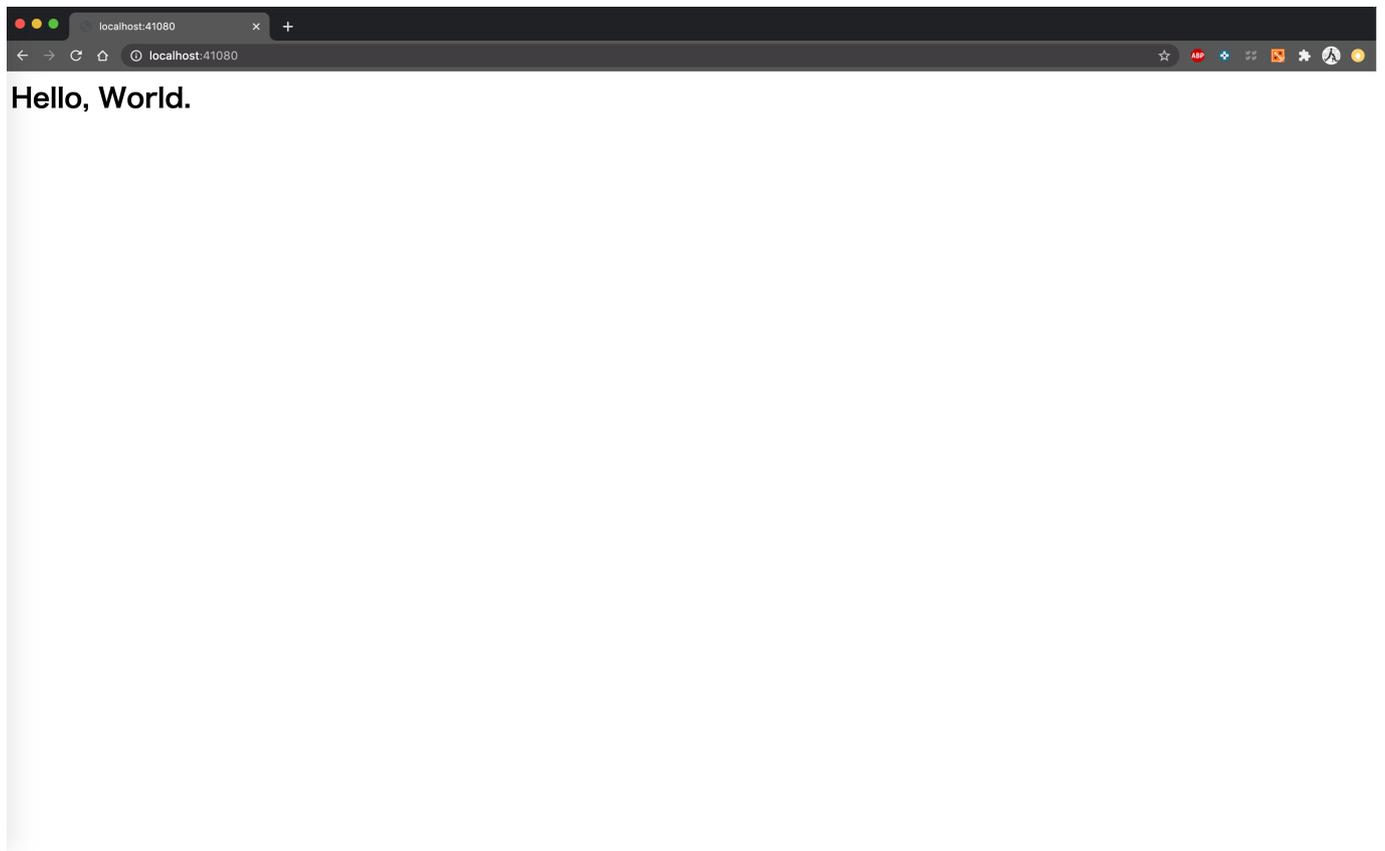
- Apacheの設定をリスタート
- **\$ sudo systemctl restart httpd.service**

index.htmlをブラウザで表示

- **\$ cd /var/www/html**
- **\$ sudo vi index.html**
- index.htmlに下記を記述

```
<HTML><BODY>
<H1>Hello, World.</H1>
</BODY></HTML>
```

- ブラウザからhttp://localhost:41080にアクセス
- 下図が表示されればOK

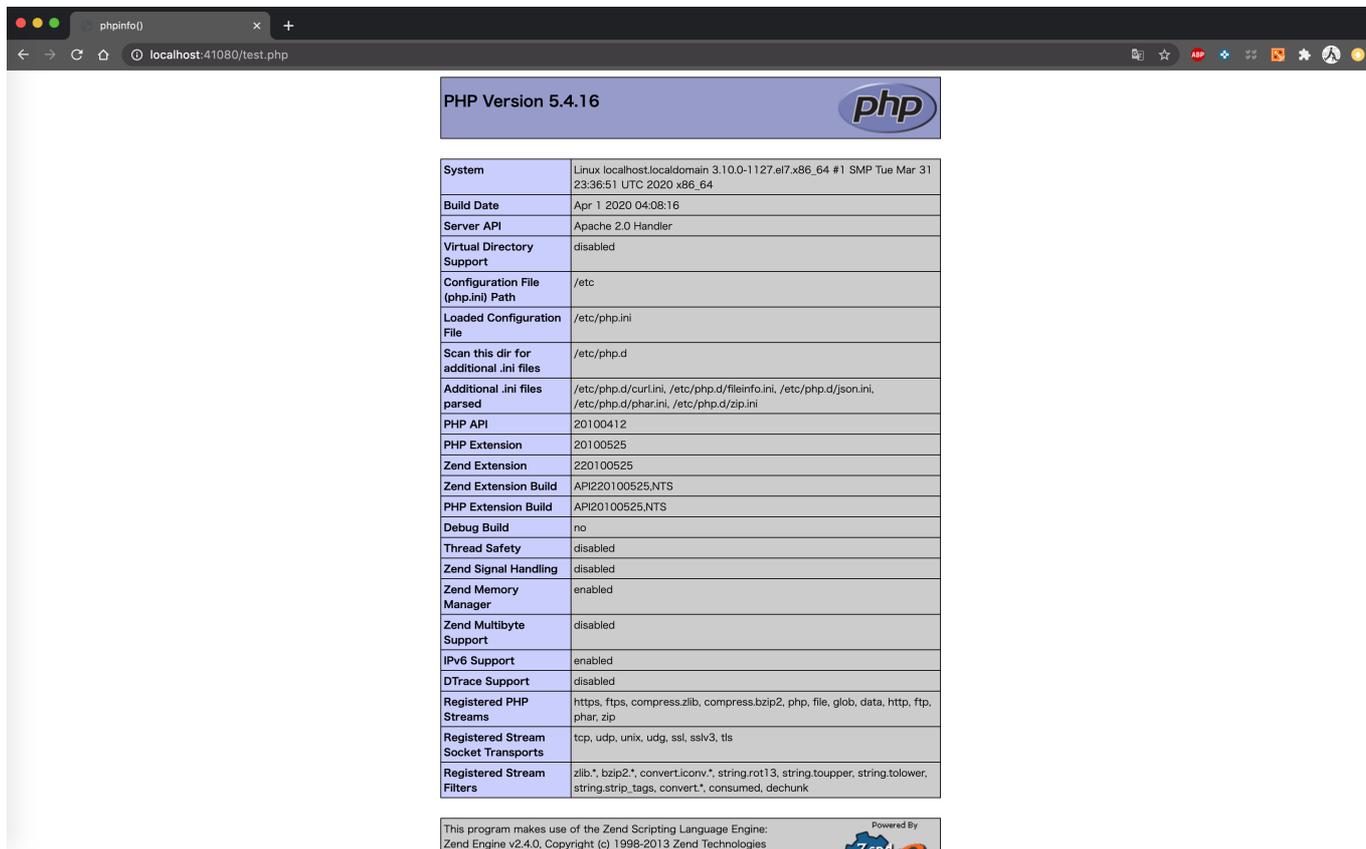


test.phpを表示

- **\$ sudo yum install -y php**
- **\$ cd var/www/html**
- **\$ sudo vi test.php**
- test.phpに下記を記述

```
<?php phpinfo(); ?>
```

- Apacheの設定をリスタートして、下図が表示されればOK



PHP Version 5.4.16

| | |
|---|--|
| System | Linux localhost.localdomain 3.10.0-1127.el7.x86_64 #1 SMP Tue Mar 31 23:36:51 UTC 2020 x86_64 |
| Build Date | Apr 1 2020 04:08:16 |
| Server API | Apache 2.0 Handler |
| Virtual Directory Support | disabled |
| Configuration File (php.ini) Path | /etc |
| Loaded Configuration File | /etc/php.ini |
| Scan this dir for additional .ini files | /etc/php.d |
| Additional .ini files parsed | /etc/php.d/curl.ini, /etc/php.d/fileinfo.ini, /etc/php.d/json.ini, /etc/php.d/phar.ini, /etc/php.d/zip.ini |
| PHP API | 20100412 |
| PHP Extension | 20100525 |
| Zend Extension | 220100525 |
| Zend Extension Build | API220100525.NTS |
| PHP Extension Build | API20100525.NTS |
| Debug Build | no |
| Thread Safety | disabled |
| Zend Signal Handling | disabled |
| Zend Memory Manager | enabled |
| Zend Multibyte Support | disabled |
| IPv6 Support | enabled |
| DTrace Support | disabled |
| Registered PHP Streams | https, ftps, compress.zlib, compress.bzip2, php, file, glob, data, http, ftp, phar, zip |
| Registered Stream Socket Transports | tcp, udp, unix, udg, ssl, sslv3, tls |
| Registered Stream Filters | zlib*, bzip2*, convert.iconv*, string.rot13, string.toupper, string.tolower, string.strip_tags, convert.*, consumed, dechunk |

This program makes use of the Zend Scripting Language Engine:
Zend Engine v2.4.0, Copyright (c) 1998-2013 Zend Technologies

手順3：ログの確認

Webサーバのログ

- Webサーバにどこからアクセスがあったか、どんなエラーがあったかの確認は `/var/log/httpd/` のファイルに記録されている
- アクセスログ： `access_log`、エラーログ： `error_log` のファイルを参照

アクセスログ

- `$ sudo tail /var/log/httpd/access_log`
- `tail` コマンドはファイルの末尾を表示
- 下図のようなフォーマットで出力されることを確認

```
10.0.2.2 - - [10/Nov/2020:01:57:47 +0900] "GET / HTTP/1.1" 200 50 "-" "Mozilla/5.0 (Macintosh; Intel Mac OS X 10_15_6) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/86.0.4240.111 Safari/537.36"
```

フォーマット解説

- 10.0.2.2 - -
 - 10.0.2.2 どこからアクセスしたかのIPアドレス。VirtualBoxの仕様より、10.0.2.2になる
- [10/Nov/2020:01:57:47 +0900]
 - アクセス日時
- "GET / HTTP/1.1" 200 50 "-"
 - WebサーバにHTTPとしてどんなリクエストがあったかを示す
- "Mozilla/5.0 (Macintosh; Intel Mac OS X 10_15_6) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/86.0.4240.111 Safari/537.36"
 - どのOSでどのブラウザからアクセスしたか

- 詳細は参考文献を参照

WEBサーバへのGETとPOSTでのリクエスト

- GETは指定されたWebページの閲覧を要求
- POSTはクライアント側からサーバ側へ情報を送信

エラーログ

- **\$ sudo tail /var/log/httpd/error_log**
- 下図のようなフォーマットで出力されることを確認

```
[Tue Nov 10 03:06:45.734371 2020] [error] [pid 10001] [client 10.0.2.2:53952] script '/var/www/html/default.php' not found or unable to stat
```

フォーマット解説

- [Tue Nov 10 03:06:45.734371 2020]
 - アクセス日時
- [error]
 - errorやwarnなどを記述
- [pid 10001]
 - プロセスID
- [Client 10.0.2.2:53952]
 - WebクライアントのIPアドレス
- script '/var/www/html/default.php' not found or unable to stat
 - エラーの内容。この例では/var/www/htmlの中にdefault.phpが存在していないためエラーになっている
- 詳細は参考文献を参照

参考文献

1. Apacheのテストページ表示まで
 - <https://weblabo.oscasierra.net/apache-installing-apache24-yum-CentOS-7-1/>, 2020/11/10
 - <https://www.karakaram.com/how-to-use-systemctl-journalctl/>, 2020/11/10
 - <http://equj65.net/tech/systemd-boot/>, 2020/11/18
2. index.html、test.php表示まで
 - <http://www.tohoho-web.com/wwwcgi1.htm>, 2020/11/10
 - <https://www.ritolab.com/entry/42>, 2020/11/10
 - <https://www.adminweb.jp/apache/cgi/index2.html>, 2020/11/10
3. ログの確認
 - <https://www.tweeety.blog/entry/2018/06/21/235130>, 2020/11/10
 - <https://beyondjapan.com/blog/2020/02/access-log/>, 2020/11/10
 - <https://www.itmedia.co.jp/help/tips/linux/l0466.html>, 2020/11/10
 - <https://www.atmarkit.co.jp/ait/articles/0202/16/news001.html>, 2020/11/10

奥付

名前 古濱 尚樹 (Naoki FURUHAMA), 竹原 一駿 (Ichitoshi TAKEHARA)
所属 香川大学大学院 工学研究科 信頼性情報システム工学専攻 喜田研究室 M1

2020/12/03 初版

2021/05/25 2版
