IDCFクラウド

活用マニュアル

～バーチャルホストでHTTPSを手軽に使ったWebサーバー構築～

目次

IDCF Cloud

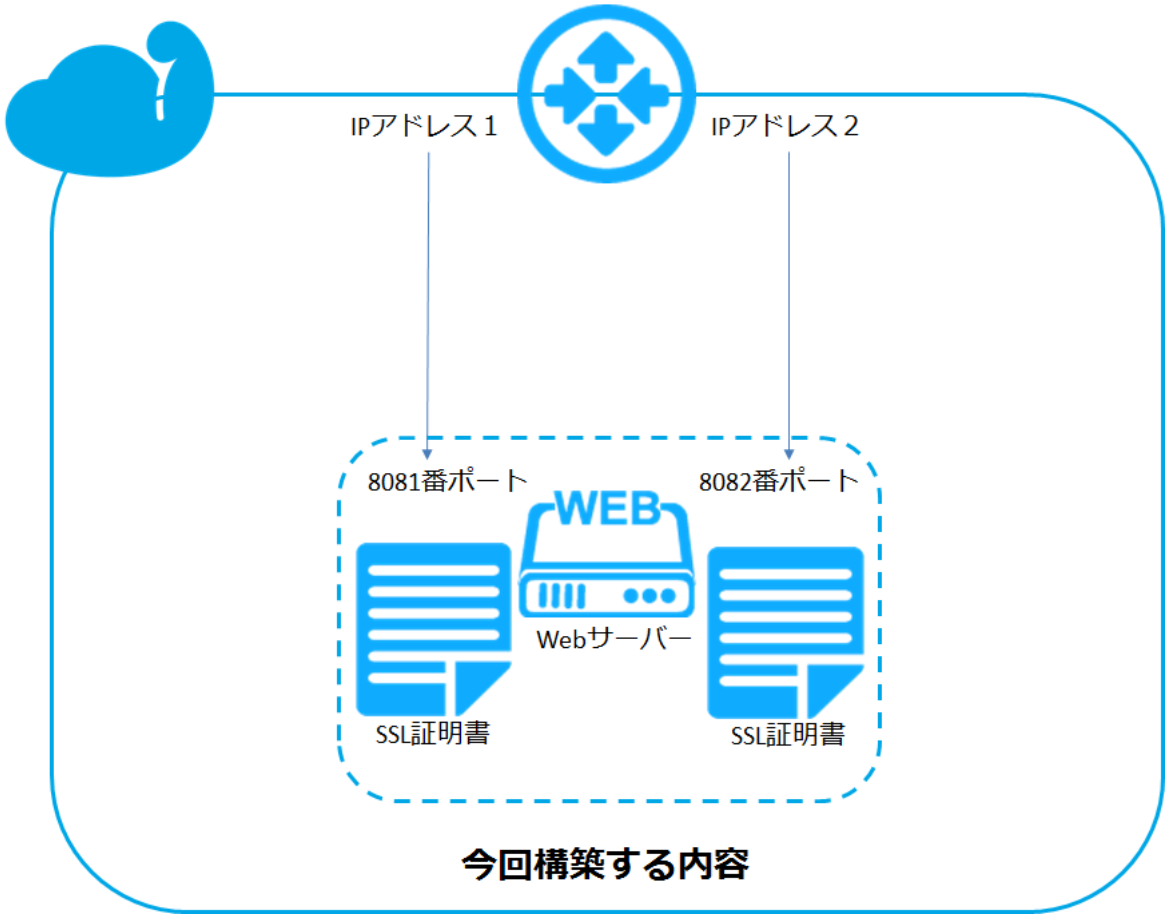
最終更新日：2016/4/28

バーチャルホストでHTTPSを手軽に使ったWebサーバー構築

Webサーバーのリソースを複数のサイトで共用して活用する方法の１つに「バーチャルホスト（VirtualHost）」があります。IDCFクラウドでバーチャルホストを使用すれば、ドメインベースでアクセスを振り分けることができます。

バーチャルホストでHTTPSを設定する場合、ドメインごとに基本的にIPアドレスが必要になります（Server Name Indicationという技術もありますが、古いブラウザに対応してない等の問題があります）。IDCFクラウドでは仮想ルーターのポートフォワーディングの機能を使用して、HTTPSを使用したポートベースのバーチャルホストを設定できます。本マニュアルでは手軽にHTTPSを使って、バーチャルホストされたWebサーバーを構築する方法をご紹介します。

本マニュアルでは、「Let's Encrypt」というサービスの「Certbot」クライアントを使用してSSL証明書を無償で作成します。WebサーバーにはApacheを使用して構築しますまた、IPアドレス1に紐づくバーチャルホストを「YourDomain1」としてポート番号を8081番でapacheに設定し、IPアドレス2に紐づくバーチャルホストを「YourDomain2」としてポート番号を8082番でnginxに設定する形で記載しています。YourDomain1,2の部分は設定するドメイン名と読みかえてください。
※独自ドメインが必要です。本マニュアルではドメインの設定は割愛します。



(1)サーバーの作成

この章ではWebサーバーを作成します。

サーバーの作成は、「Webサイトの本番環境を構築したい（Web1台構成）」の(1)仮想マシンの作成まで完了した状態からの手順となります。本マニュアルでは、ゾーンをradianで設定しています。

① コンピューティングのメニューより「IPアドレス」をクリックし、IPアドレス設定画面を表示します。右側の「IPアドレス追加」をクリックします。

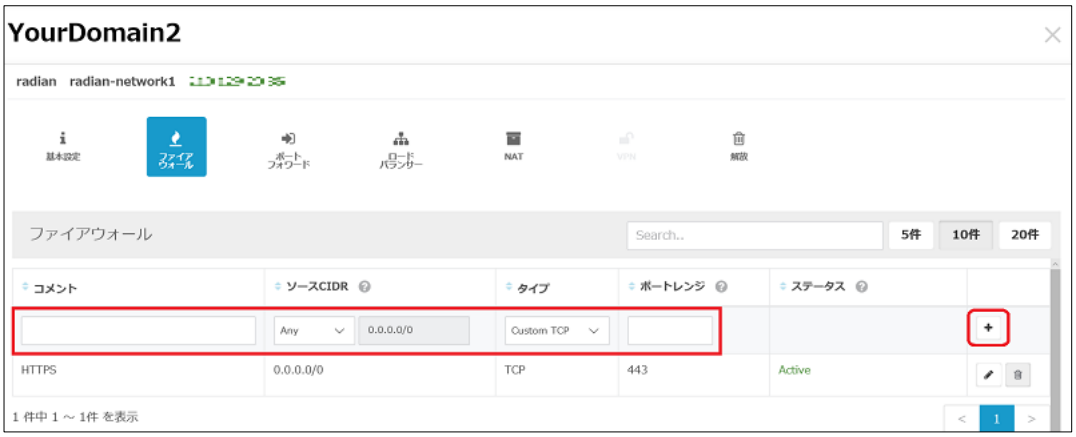
② コンピューティングのメニューより「IPアドレス」をクリックし、IPアドレス設定画面を表示します。右側の「IPアドレス追加」をクリックします。

項目	設定内容
IPアドレス名	YourDomain2（任意）
ゾーン	仮想マシンを作成したゾーン
ネットワーク	仮想マシンを作成したネットワーク

③ IPアドレス取得しました。というメッセージが表示されるので、取得した「IPアドレス」をクリックします。

④ ファイアウォールをクリックし、HTTPSのルールを作成します。入力後、「＋」ボタンで追加します。

項目	設定内容
コメント	HTTPS
ソースCIDR	Any
タイプ	HTTPS
ポートレンジ	443(自動入力)



⑤ ポートフォワードをクリックし、HTTPSのポートフォワード設定を行います。本設定も同様に「＋」ボタンで追加します。追加できたら右上の「x」で設定画面を閉じます。

項目	設定内容
コメント	HTTPS
パブリックポート	HTTPS (プルダウンより選択)
プライベートポート	443
プロトコル	TCP
仮想マシン	Web01

⑥ ゾーンに払い出されている既存のIPアドレスをクリックします。（既存のIPアドレスに対して、名前を設定していなければ、IPアドレス名に(no name)と記載されています）

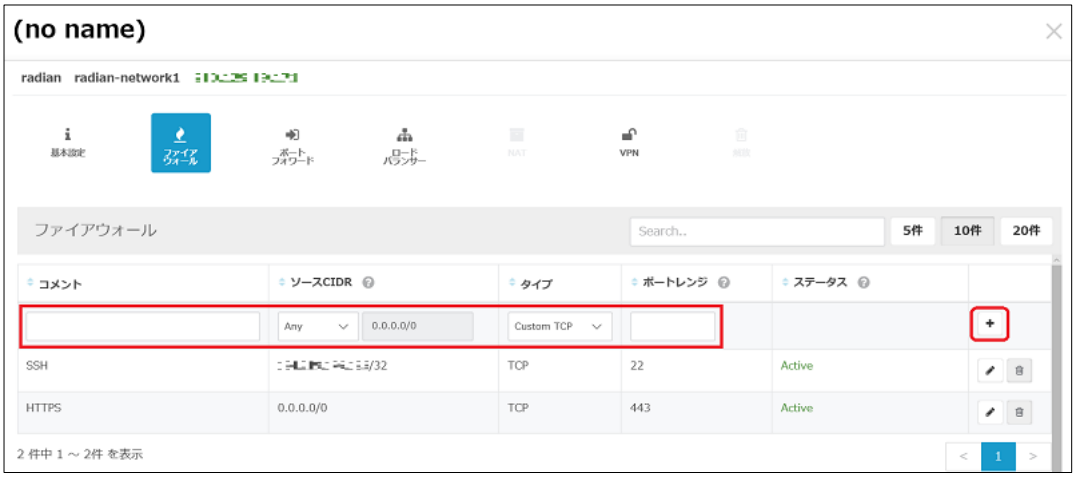
⑦ ファイアウォールをクリックし、ルールを図のように2つ作成します。作成は1つずつ行い、「＋」ボタンで追加します。

・HTTPS

項目	設定内容
コメント	HTTPS
ソースCIDR	Any
タイプ	HTTPS
ポートレンジ	443 (自動入力)

・SSH

項目	設定内容
コメント	SSH
ソースCIDR	My IP
タイプ	SSH
ポートレンジ	22 (自動入力)



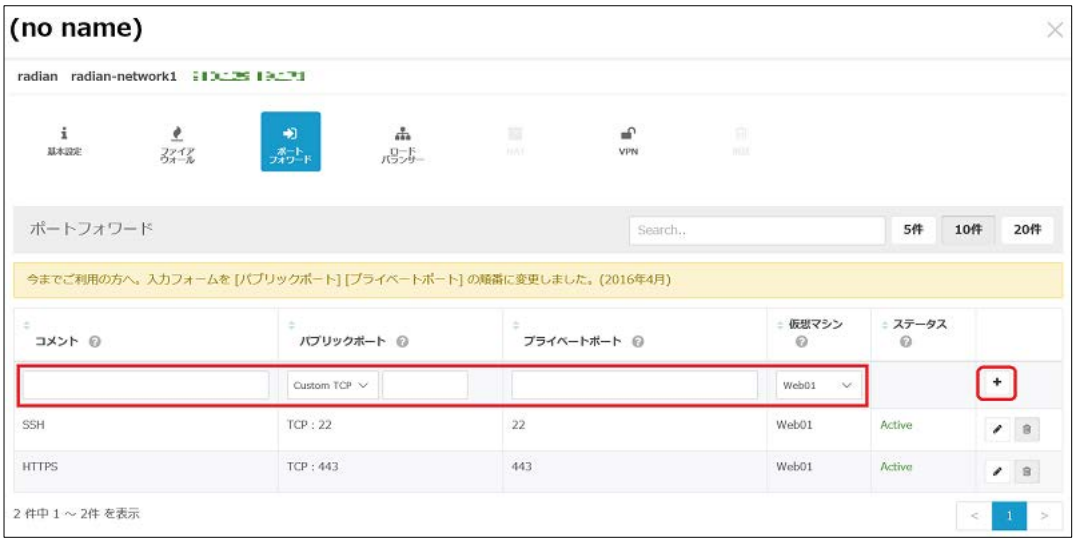
⑧ ポートフォワードをクリックし、ルールを図のように2つ作成します。作成は1つずつ行い、「＋」ボタンで追加します。

・HTTPS

項目	設定内容
コメント	HTTPS
パブリックポート	HTTPS (プルダウンより選択)
プライベートポート	443
プロトコル	TCP
仮想マシン	Web01

・SSH

項目	設定内容
コメント	SSH
パブリックポート	SSH (プルダウンより選択)
プライベートポート	22
プロトコル	TCP
仮想マシン	Web01



⑨ 活用マニュアルの「Webサイトの本番環境を構築したい（Web1台構成）」を参照して行います。この例では、イメージはCentOS 7.3 64-bit、マシンタイプは1コア4GBのstandard.S4を使用しています。

次の章では、2つのVirtualServer用にドメインが2つ必要になります。
以下はそれぞれをYourDomain1、YourDomain2とします。DNSにて既存のIPアドレス1（YourDomain1）、新規追加したIPアドレス2（YourDomain2）に対してDNSでAレコードを設定してください。DNSの設定についての説明は割愛します。

(2)サーバー証明書の作成

Let's EncryptのCertbotクライアントを使用してサーバー証明書を作成します。Certbotはドメイン名でサーバー認証を行うため、対象ドメインが名前解決できることを確認してから進めてください。

① CertbotのクライアントとApacheをダウンロードします。

```
[root@Web01 ~]# yum -y install epel-release
読み込んだプラグイン:fastestmirror, remove-with-leaves, show-leaves
Loading mirror speeds from cached hostfile
～～～～～～～～～～～～～～～～
インストール:
  epel-release.noarch 0:7-9
完了しました!
New leaves:
  epel-release.noarch
[root@Web01 ~]# yum -y install certbot python-certbot-apache
読み込んだプラグイン:fastestmirror, remove-with-leaves, show-leaves
～～～～～～～～～～～～～～～～
完了しました!
New leaves:
  python2-certbot-apache.noarch
```

② CLIでYourDomain1の証明書を作成します。-mオプションの後にはご自身のメールアドレスを入力してください。-dオプションのあとにはYourDomain1のFQDNを入力します。

```
[root@Web01 ~]# certbot certonly --standalone --agree-tos -m YourEmailAddress -d YourDomain1
```

③ 上記実行後、以下のメッセージが表示されます。これで作成は成功です。

```
Generating key (2048 bits): /etc/letsencrypt/keys/0000_key-certbot.pem
Creating CSR: /etc/letsencrypt/csr/0000_csr-certbot.pem
```

```
IMPORTANT NOTES:
 - Congratulations! Your certificate and chain have been saved at
   /etc/letsencrypt/live/YourDomain1/fullchain.pem. Your
   cert will expire on 2017-07-11. To obtain a new or tweaked version
   of this certificate in the future, simply run certbot again. To
   non-interactively renew *all* of your certificates, run "certbot
   renew"
 - Your account credentials have been saved in your Certbot
   configuration directory at /etc/letsencrypt. You should make a
   secure backup of this folder now. This configuration directory will
   also contain certificates and private keys obtained by Certbot so
   making regular backups of this folder is ideal.
 - If you like Certbot, please consider supporting our work by:

   Donating to ISRG / Let's Encrypt:   https://letsencrypt.org/donate
   Donating to EFF:                    https://eff.org/donate-le
```

④ YourDomain2の証明書も作成します。先ほどと同じメッセージが表示されれば作成は完了です。

```
[root@Web01 ~]# certbot certonly --standalone -d YourDomain2

～～～～～～～～～～～～～～～～

Generating key (2048 bits): /etc/letsencrypt/keys/0001_key-certbot.pem
Creating CSR: /etc/letsencrypt/csr/0001_csr-certbot.pem

IMPORTANT NOTES:
 - Congratulations! Your certificate and chain have been saved at
   /etc/letsencrypt/live/YourDomain2/fullchain.pem. Your
   cert will expire on 2017-07-11. To obtain a new or tweaked version
   of this certificate in the future, simply run certbot again. To
   non-interactively renew *all* of your certificates, run "certbot
```

```
   renew"
 - If you like Certbot, please consider supporting our work by:


   Donating to ISRG / Let's Encrypt:   https://letsencrypt.org/donate
   Donating to EFF:                    https://eff.org/donate-le
```

⑤ 証明書が作成されていることを確認します。また、証明書に対してシンボリックリンクが張られていることも確認します。

```
[root@Web01 ~]# ls /etc/letsencrypt/archive/YourDomain1/
cert1.pem  chain1.pem  fullchain1.pem  privkey1.pem


[root@Web01 ~]# ls /etc/letsencrypt/archive/YourDomain2/
cert1.pem  chain1.pem  fullchain1.pem  privkey1.pem


[root@Web01 ~]# ls -l /etc/letsencrypt/live/YourDomain1/
合計 4

-rw-r--r-- 1 root root 543  4月 26 10:16 README

lrwxrwxrwx 1 root root  48  4月 26 10:16 cert.pem ->


../../archive/YourDomain1/cert1.pem

lrwxrwxrwx 1 root root  49  4月 26 10:16 chain.pem ->


../../archive/YourDomain1/chain1.pem

lrwxrwxrwx 1 root root  53  4月 26 10:16 fullchain.pem ->


../../archive/YourDomain1/fullchain1.pem

lrwxrwxrwx 1 root root  51  4月 26 10:16 privkey.pem ->


../../archive/YourDomain1/privkey1.pem


[root@Web01 ~]# ls -l /etc/letsencrypt/live/YourDomain2/
合計 4

-rw-r--r-- 1 root root 543  4月 26 10:21 README

lrwxrwxrwx 1 root root  48  4月 26 10:21 cert.pem ->


../../archive/YourDomain2/cert1.pem

lrwxrwxrwx 1 root root  49  4月 26 10:21 chain.pem ->
```

```
../../archive/YourDomain2/chain1.pem
lrwxrwxrwx 1 root root  53  4月 26 10:21 fullchain.pem ->

../../archive/YourDomain2/fullchain1.pem
lrwxrwxrwx 1 root root  51  4月 26 10:21 privkey.pem ->

../../archive/YourDomain2/privkey1.pem
```

以上でサーバ証明書の作成が完了しました。それぞれのファイルの役割は以下に記載された通りとなります。

サーバー証明書（公開鍵）：cert.pem （apacheで使用）
中間証明書：chain.pem （apacheで使用）
サーバー証明書と中間証明書の結合ファイル：fullchain.pem （nginxで使用）
秘密鍵：privkey.pem （両方で使用）

(3) Webサーバーの設定をする

Apacheのコンフィグにバーチャルホストと作成したサーバー証明書を使用するための設定を記述します。

① Apacheの設定を行います。エディタを使用してコンフィグファイルにバーチャルホストの設定を追記します。（以下、例ではviを使用します。）
※ドメイン名の部分はそれぞれYourDomain1、YourDomain2と読みかえてください。

```
[root@Web01 ~]# vi /etc/httpd/conf/httpd.conf
#追記内容
#ここから------------------------------
Listen 8081

Listen 8082



<VirtualHost *:8081>

ServerName YourDomain1

DocumentRoot /var/www/html/domain1

SSLEngine on

SSLCertificateFile /etc/letsencrypt/live/YourDomain1/cert.pem

SSLCertificateKeyFile /etc/letsencrypt/live/YourDomain1/privkey.pem

</VirtualHost>



<VirtualHost *:8082>

ServerName YourDomain2

DocumentRoot /var/www/html/domain2

SSLEngine on

SSLCertificateFile /etc/letsencrypt/live/YourDomain2/cert.pem

SSLCertificateKeyFile /etc/letsencrypt/live/YourDomain2/privkey.pem

</VirtualHost>
#ここまで------------------------------
```

② 同じく上記ファイルに対して、80番ポートをListenするコンフィグをコメントアウトします。

```
[root@Web01 ~]# vi /etc/httpd/conf/httpd.conf
#元々の設定

Listen 80
```

```
#新しい設定（冒頭にシャープを入れます）
#Listen 80
```

③ サーバー証明書の更新作業をラクにするため、デフォルトで設定されているSSLのコンフィグを退避させます。（証明書の更新については章(5)で詳しく説明します。）

```
[root@Web01 ~]# mv /etc/httpd/conf.d/ssl.conf /etc/httpd/conf.d/ssl.conf.backup
```

④ それぞれのドメインのドキュメントルートディレクトリを作成します。

```
[root@Web01 ~]# mkdir /var/www/html/domain1
[root@Web01 ~]# mkdir /var/www/html/domain2
```

⑤ Apacheのコンフィグの書式を確認します。エラーが出たらもう一度設定を確認しましょう。

```
[root@Web01 ~]# apachectl configtest
Syntax OK
```

⑥ Apacheを起動します。

```
[root@Web01 ~]# systemctl start httpd
```

⑦ 8081番と8082番ポートがListenしているか確認します。下記のように2行返ってくればListenしている状態となります。

```
[root@Web01 ~]# netstat -lnep | grep httpd
tcp6       0      0 :::8081                 :::*                    LISTEN      0
102008     2781/httpd
tcp6       0      0 :::8082                 :::*                    LISTEN      0
102012     2781/httpd
```

以上でWebサーバーの設定が完了しました。

コラム：nginxの場合

nginxの場合も、同様にエディタを使用してコンフィグファイルに記述することで、バーチャルホストの設定が可能です。（以下、例ではviを使用します。）

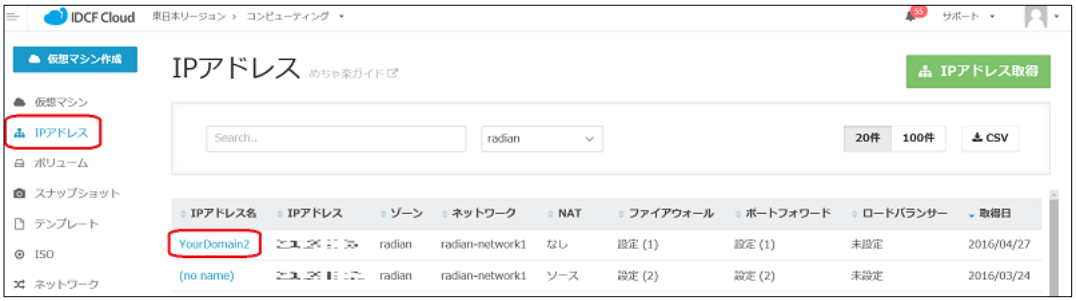
※ドメイン名の部分はそれぞれYourDomain1、YourDomain2と読み替えてください。

```
[root@Web01 ~]# vi /etc/nginx/conf.d/default.conf

#ここから------------------------------
server {
    listen       8081;
    ssl on;
    server_name  YourDomain1;
    ssl_certificate             /etc/letsencrypt/live/YourDomain1/cert.pem;
    ssl_certificate_key         /etc/letsencrypt/live/YourDomain1/privkey.pem;

   location / {
        root   /usr/share/nginx/html;
        index  index.html index.htm;
    }
}

server {
    listen       8082;
    ssl on;
    server_name  YourDomain2;
    ssl_certificate             /etc/letsencrypt/live/YourDomain2/cert.pem;
    ssl_certificate_key         /etc/letsencrypt/live/YourDomain2/privkey.pem;

   location / {
        root   /usr/share/nginx/html;
        index  index.html index.htm;
    }
}
#ここまで------------------------------
```

(4)ポートフォワーディングと設定確認

現在、443番ポートに転送しているポートフォワーディング設定を8081番、8082番ポートに振り直し、Webブラウザで設定に問題がないか確認をします。

① コントロールパネルよりIPアドレスをクリックし、IPアドレス一覧画面を表示します。今回追加したIPアドレス名をクリックし、IP設定画面を表示します。



② ポートフォワードをクリックし、ごみ箱ボタンでHTTPSの設定を削除します。確認のポップアップが表示されるので「はい」をクリックします。

③ 既存のHTTPSの設定が削除されたことを確認し、8082番ポートにポートフォワードする設定を追加します。追加後、設定画面を「×」で閉じます。

項目	設定内容
コメント	HTTPS
パブリックポート	HTTPS (プルダウンより選択)
プライベートポート	8082
プロトコル	TCP
仮想マシン	Web01

④ ゾーンに払い出されている既存のIPをクリックします。（既存のIPに対して、何も設定していなければ、IPアドレス名に(no name)と記載されています。）

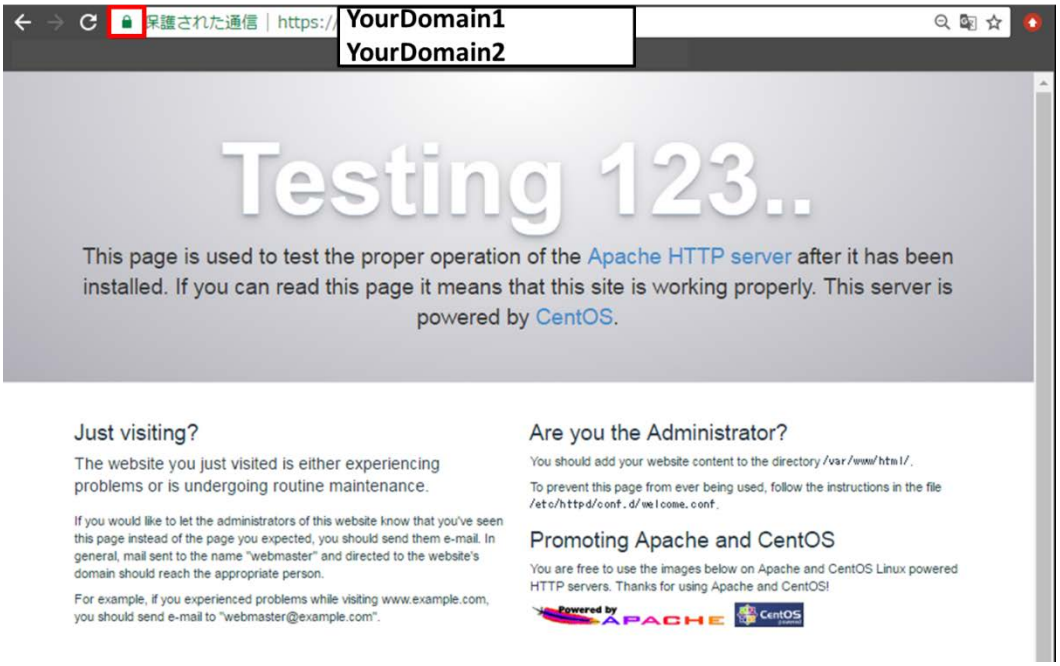
⑤ ポートフォワードをクリックし、ごみ箱ボタンでHTTPSの設定を削除します。確認のポップアップが表示されるので「はい」をクリックします。※SSHは削除しません。

⑥ 既存のHTTPSの設定が削除されたことを確認し、8081番ポートにポートフォワードする設定を追加します。追加後、設定画面を「×」で閉じます。

項目	設定内容
コメント	HTTPS
パブリックポート	HTTPS (プルダウンより選択)
プライベートポート	8081
プロトコル	TCP
仮想マシン	Web01

⑦ WebブラウザにてYourDomain1とYourDomain2へHTTPSでアクセスできることを確認します。

HTTPSでアクセスし、Webページを表示します。証明書エラーが出ずに、テストページが表示されれば問題ありません。鍵マークをクリックし、証明書を確認しエラーがないか確認します。（次の例ではChromeを使用しています。ブラウザにより挙動が異なる可能性があります）



以上でバーチャルホストを使用してHTTPSを有効にする作業が完了しました。

(5)サーバー証明書の更新

Certbotクライアントで作成する証明書の有効期限は90日なので、定常的に運用する際は失効する前に証明書更新をする必要があります。通常のサーバー証明書更新は'certbot renew'コマンドでできるのですが、今回はバーチャルホストを使用しているため、ポートフォワードの設定を変更する手順がともないます。

① 上記(4)-②と(4)-③の手順を参考に、YourDomain2のIPアドレスのポートフォワードの画面に遷移します。8082番ポートに対してのポートフォワーディングの設定を一度削除し、以下設定を入れます。

項目	設定内容
コメント	HTTPS
パブリックポート	HTTPS (プルダウンより選択)
プライベートポート	443
プロトコル	TCP
仮想マシン	Web01

② 上記設定をゾーンに払い出されている既存のIPに対しても実行します。8081番ポートに対してのポートフォワーディングの設定を一度削除し、以下設定を入れます。

項目	設定内容
コメント	HTTPS
パブリックポート	HTTPS (プルダウンより選択)
プライベートポート	443
プロトコル	TCP
仮想マシン	Web01

③ 下記コマンドを入力して、証明書を更新します。成功したとメッセージが出るのを確認します。

```
[root@Web01 ~]# certbot renew
～～～～～～～～～～
Congratulations, all renewals succeeded. The following certs have been renewed:
/etc/letsencrypt/live/YourDomain1/fullchain.pem (success)
/etc/letsencrypt/live/YourDomain2/fullchain.pem (success)
```

④ 最後に①と②で変更したポートフォワードの設定を元に戻します。手順は章(4)と同じです。

今回はポートフォワードを上手に活用することによって、バーチャルホストの設定を行いました。バーチャルホスト環境でも気軽にHTTPSを使用することができます。
本マニュアルでは説明のためにApacheを利用しましたが、Nginxでも設定可能です。証明書を発行したLet's Encrypt（Certbotクライアント）はLinux Foundationのコラボレートプロジェクトで、ISRGが中心となり複数企業で運営を行っています。2016年4月12日に正式サービスとしてリリースされました。手軽に暗号化して通信を行いたい場合や今回のようにHTTPSのテストを行いたい場合等、さまざまなシーンで有用なサービスです。
しかしながら、身元審査等は行わないため、信頼されたサイトの証明として使用する場合は、OV証明書、EV証明書を使用することをおすすめします。

Let's Encrypt公式ページ（英語）
https://letsencrypt.org/