

# cockpit用サーバー証明書の自動更新

サーバー管理に便利なWebコンソールの[cockpit](#)だが、サーバー証明書がない環境だと**0-self-signed.cert**という自己署名証明書を生成して終わってしまう。Apacheには[Let's Encrypt](#)による証明書を導入しているので、これをcockpit用に転用する。

## cockpit用サーバー証明書のフォーマット

**cockpit-ws**(Cockpit web service)に設定する証明書は以下の形式である必要がある。

- ファイル名の拡張子は.cert
- OpenSSLスタイルPEMブロックとして下記の2つ以上を含むこと
  - 先頭からにサーバー証明書もしくは中間証明書のBEGIN CERTIFICATEブロック
  - 最後にBEGIN PRIVATE KEYのブロック

最小構成
--BEGIN CERTIFICATE--
サーバー証明書ブロック
--END CERTIFICATE--
--BEGIN PRIVATE KEY--
PRIVATE KEYブロック
--END PRIVATE KEY--
中間証明書を含む構成
--BEGIN CERTIFICATE--
中間証明書ブロック
--END CERTIFICATE--
--BEGIN CERTIFICATE--
サーバー証明書ブロック
--END CERTIFICATE--
--BEGIN PRIVATE KEY--
PRIVATE KEYブロック
--END PRIVATE KEY--

## 証明書更新スクリプト

Apacheに設定した証明書はcronで起動されるcertbotで更新されるので、その更新を検知してcockpit用証明書を複製するスクリプトを作成した。

### [cockpit-cert](#)

```
#!/bin/sh
CERTFILE=/etc/letsencrypt/archive/domain.jp/cert1.pem
KEYFILE=/etc/letsencrypt/archive/domain.jp/privkey1.pem
NEWCERT=/etc/cockpit/ws-certs.d/cockpit.cert

update=0
if [ -e ${NEWCERT} ]
```

```
then
    fp1=`openssl x509 -fingerprint -noout -in ${CERTFILE}`
    fp2=`openssl x509 -fingerprint -noout -in ${NEWCERT}`
    if [ "${fp1}" != "${fp2}" ]
    then
        update=1
    fi
else
    update=1
fi

if [ ${update} -eq 1 ]
then
    cat ${CERTFILE} ${KEYFILE} > ${NEWCERT}
    chmod 640 ${NEWCERT}
    chown root:cockpit-ws ${NEWCERT}
fi
```

From: <https://kijima.mydns.jp/dokuwiki/> - Kijima's private wiki

Permanent link:

<https://kijima.mydns.jp/dokuwiki/doku.php?id=cockpit%E7%94%A8%E3%82%B5%E3%83%BC%E3%83%90%E3%83%BC%E8%A8%BC%E6%98%8E%E6%9B%BB%E3%81%AE%E8%87%AA%E5%8B%95%E6%9B%4%E6%96%80&rev=1693100615>

Last update: 2023/08/27 01:43

