Авторизация в административный интерфейс по сертификату

vim /etc/ssl/openssl.cnf

```
[ ca ]
default_ca = CA # The default ca section
[ CA ]
dir = /usr/abills/Certs/ # Where everything is kept
certs = $dir/certs # Where the issued certs are kept
crl_dir = $dir/crl # Where the issued crl are kept
database = $dir/index.txt # database index file.
new_certs_dir = $dir/newcerts # default place for new certs.
certificate = $dir/CA/ca.crt # The CA certificate
serial = $dir/serial # The current serial number
#crlnumber = $dir/crlnumber # the current crl number must be
# commented out to leave a V1 CRL
crl = $dir/CA/ca.crl # The current CRL
private_key = $dir/CA/ca.key # The private key
x509_extensions = usr_cert
default_days    = 365                   # how long to certify for
default_crl_days= 30                    # how long before next CRL
default_md      = sha1                  # which md to use.
preserve        = no                    # keep passed DN ordering
policy          = policy_match

# For the CA policy
[ policy_match ]
countryName             = match
stateOrProvinceName     = match
organizationName        = match
organizationalUnitName  = optional
commonName              = supplied
emailAddress            = optional

# For the 'anything' policy
[ policy_anything ]
countryName             = optional
stateOrProvinceName     = optional
localityName            = optional
organizationName        = optional
organizationalUnitName  = optional
commonName              = supplied
emailAddress            = optional

[ req ]
default_md = sha1
distinguished_name = req_distinguished_name

[ req_distinguished_name ]
countryName = Country Name (2 letter code)
countryName_default = UA
countryName_min = 2
countryName_max = 2
stateOrProvinceName = State or Province Name (full name)
stateOrProvinceName_default = IF
localityName = Locality Name (eg, city)
localityName_default = city
0.organizationName = Organization Name (eg, company)
0.organizationName_default = Org_name
organizationalUnitName = Organizational Unit Name (eg, section)
commonName = Common Name (eg, hostname)
commonName_max = 64
emailAddress = Email Address
emailAddress_default = test
emailAddress_max = 64
```

```
[ server ]
basicConstraints=CA:FALSE
nsCertType                      = server
keyUsage = nonRepudiation, digitalSignature, keyEncipherment
subjectKeyIdentifier=hash
authorityKeyIdentifier=keyid,issuer

[ client ]
basicConstraints=CA:FALSE
nsCertType = client
keyUsage = nonRepudiation, digitalSignature, keyEncipherment
subjectKeyIdentifier=hash
authorityKeyIdentifier=keyid,issuer

[ v3_req ]

basicConstraints = CA:FALSE
keyUsage = nonRepudiation, digitalSignature, keyEncipherment

[ v3_ca ]

subjectKeyIdentifier=hash
authorityKeyIdentifier=keyid:always,issuer:always
basicConstraints = CA:true
nsCertType = sslCA

[ crl_ext ]
authorityKeyIdentifier=keyid:always,issuer:always
```

```
cd /usr/abills/Certs/ && mkdir {CA,server,user}
```

echo «01» > serial

```
touch index.txt
```

Создаем сертификат организации Создаем ключ и самподписанный CA сертификат

```
openssl genrsa – rand /var/log/messages -out ./CA/ca.key -camellia256 2048

openssl req -new -key ./CA/ca.key -out ./CA/ca.csr

openssl x509 -req -signkey ./CA/ca.key -in ./CA/ca.csr -extfile /etc/ssl/openssl.cnf -out ./CA/ca.crt
```

Проверка

```
openssl x509 -in ca.crt -text
```

Теперь создаем сертификат сервера

```
openssl genrsa -rand /var/log/messages -out ./server/freebsd.local.com.key -camellia256 2048
```

COMMON NAME вводим **ДНС имя серверу** Пароль тот же что и для частного ключа freebsd.local.com.key

```
openssl req -new -key ./server/freebsd.local.com.key -config /etc/ssl/openssl.cnf  -out ./server
freebsd.local.com.csr
```

и подписываем его сертификатом организации

```
openssl ca -in ./server/freebsd.local.com.csr -cert ./CA/ca.crt -keyfile ./CA/ca.key -out
./server/freebsd.local.com.crt -config /etc/ssl/openssl.cnf -days 365
```

Создаем сертификат клиента

```
openssl genrsa -rand /var/log/messages -out ./user/client.key -camellia256 2048
```

openssl req -new -key /user/client.key -out ./user/client.csr

и подписываем его сертификатом организации

```
openssl ca -in ./user/client.csr -cert ./CA/ca.crt -keyfile ./CA/ca.key -out ./user/client.crt -
extensions client -config /etc/ssl/openssl.cnf
```

Экспортируем в формат читаємий IE, Mozilla...

```
openssl pkcs12 -export -clcerts -in ./user/client.crt -inkey ./user/client.key -out ./user/client.p12
```

или

```
openssl pkcs12 -export -clcerts -in ./user/certificates/client.crt -inkey ./user/keys/client.key -out
./user/certificates/client.p12 -name "test certificate"
```

Копируем файл client.p12 на компьютер клиента

Редактируем конфиг /usr/abills/misc/apache/abills_httpd.conf

Вставляем между <IfModule ssl_module> </IfModule ssl_module>

```
<Location /admin>
SSLVerifyClient require
SSLVerifyDepth 10
</Location>
```

Изменяем путь к сертификатам

```
SSLCertificateFile     /usr/abills/Certs/server/freebsd.local.com.crt
SSLCertificateKeyFile  /usr/abills/Certs/server/freebsd.local.com.key
SSLCACertificateFile    /usr/abills/Certs/CA/ca.crt
```

Перезапускаем веб сервер

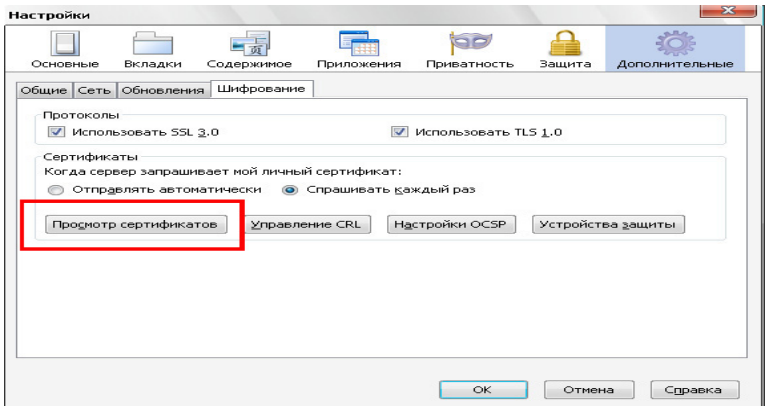
```
/usr/local/etc/rc.d/apache22 restart
```

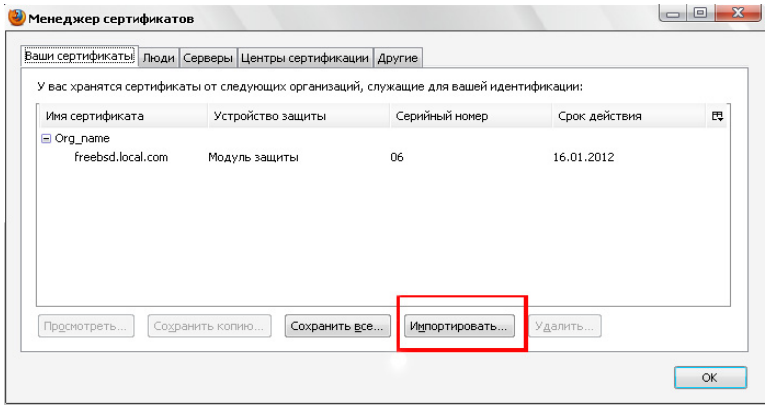
смотрим лог ошибок

Добавляем сертификат на стороне клиента

Для для работы в Internet Explorer, GoogleChrome запускаем freebsd.local.com.p12

Для Mozilla FireFox





Дополнительные ссылки по авторизации с помощью сертификатов

http://www.linuxconfig.org/apache-web-server-ssl-authentication
http://www.cafesoft.com/products/cams/ps/docs30/admin/ConfiguringApache2ForSSLTLSMutualAuthentication.html
http://www.symantec.com/connect/articles/apache-2-ssltls-step-step-part-3