

# إدارة المفاتيح والشهادة

## توليد مفتاح خاص وطلب شهادة لعامل

### باستخدام خوارزمية RSA 2048

التعليمية التالية ستقوم بتوليد ملفين أحدهما هو الملف الخاص Private key والآخر هو طلب الشهادة (ذو اللامقة csr) يمكن تنفيذ التعليمية بأي نظام تشغيل يحوي الحزمة Openssl يجب التأكيد وضع القيم الصحيحة لمكان العمل والصفة الوظيفية والرقم الوطني قبل تنفيذ الأمر

```
openssl req -new -utf8 -nameopt multiline,utf8 -nodes -out my_name.csr -  
newkey rsa:2048 -keyout my_name.key -config <(  
    cat <<-EOF  
        [req]  
        prompt = no  
        default_md = sha256  
        distinguished_name = dn  
  
        [ dn ]  
        C=SY  
        Change Me #             title=الصفة الوظيفية  
        Change Me #             جهتي=0  
        Change Me #             OU=المديرية  
        Change Me #             CN=اسمي  
        emailAddress=my@email.sy      # Change Me  
        serialNumber=NID-01010101010 # Change the Number  
    EOF  
(
```

### باستخدام خوارزمية ECDSA

التعليمية التالية ستقوم بتوليد ملفين أحدهما هو الملف الخاص Private key والآخر هو طلب الشهادة (ذو اللامقة csr) يمكن تنفيذ التعليمية بأي نظام تشغيل يحوي الحزمة Openssl يجب التأكيد وضع القيم الصحيحة لمكان العمل والصفة الوظيفية والرقم الوطني قبل تنفيذ الأمر

```
openssl ecparam -name prime256v1 -genkey -out my_name.key  
openssl req -new -utf8 -nameopt multiline,utf8 -key my_name.key -out  
    my_name.csr -config <(  
    cat <<-EOF
```

```
[req]
prompt = no
default_md = sha256
distinguished_name = dn

[ dn ]
C=SY

Change Me #      الصفة الوظيفية=title
Change Me #      جهتي=0
Change Me #      المديرية=OU
Change Me #      اسمي=CN
emailAddress=my@email.sy    # Change Me
serialNumber=NID-01010101010 # Change the Number
EOF
()
```

## تشكيل ملف pfx/pkcs12

يمكن دمج ملف الشهادة الرقمية مع ملف المفتاح الخاص لتشكيل ملف pfx (الحماية الملف الخاص بكلمة مرور وكذلك للتمكن من تنصيب الشهادة على نظام التشغيل أو وتحميلها على حامل إلكتروني) باستخدام الأمر التالي الذي يمكن تنفيذه من أي نظام تشغيل يحوي الحزمة Openssl كما يلي:

```
openssl pkcs12 -export -out my_name.pfx -inkey my_name.key -in my_name.pem
```

سيتم طلب رمز لقفل الملف به، ندخل الرمز مرتين

ملاحظة: الأمر السابق يحتاج لأن تكون الشهادة بتنسيق pem (base64)، فإذا كانت الشهادة بتنسيق (der/cer/binary) فيمكن تحويلها باستخدام الأمر التالي:

```
openssl x509 -inform der -in my_name.cer -out my_name.pem
```

## توليد مفتاح خاص وطلب شهادة Code Signing

الأمر التالي سيقوم بتوليد مفتاح خاص Private key وملف طلب شهادة CSR للنطاق example.sy مع ملاحظة ضرورة تعديل قيم الحقول التالية بالطلب حسب الحاجة، 0 و CN وكذلك يمكن إضافة حقول أخرى عند الحاجة

```
openssl req -new -nodes -out example.sy.csr -newkey rsa:2048 -keyout
example.sy.key -config <(
```

```

cat <<-EOF
[req]
prompt = no
default_md = sha256
distinguished_name = dn

[ dn ]
C=SY
O=Organization Name
CN=Organization Name
EOF
(

```

## توليد مفتاح خاص وطلب شهادة ssl

الأمر التالي سيقوم بتوليد مفتاح خاص Private key وملف طلب شهادة CSR للنطاق example.sy مع ملاحظة ضرورة تعديل قيم الحقول التالية بالطلب حسب الحاجة، o و CN و 1. DNS و 2. DNS وكذلك يمكن إضافة حقول أخرى عند الحاجة

```

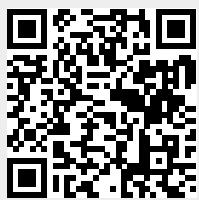
openssl req -new -nodes -out example.sy.csr -newkey rsa:2048 -keyout
example.sy.key -config <(
cat <<-EOF
[req]
prompt = no
default_md = sha256
distinguished_name = dn
req_extensions = ssl_reqext

[ dn ]
C=SY
O=Organization Name
CN=example.sy

[ ssl_reqext ]
subjectAltName=@alt_names

[ alt_names ]
DNS.1 = example.sy
DNS.2 = *.example.sy
EOF
(

```



From:  
- مركز معلومات مركز التصديق /<https://info.ecc.sy>

Permanent link:  
<https://info.ecc.sy/doku.php?id=howto:keymgmt>

Last update: **2025/05/05 09:02**