

Security Tools

What is a CA ?

- ◎ *Certificate Authority* (認證中心)
- ◎ Trusted server which signs certificates
- ◎ One private key and relative public key
- ◎ Tree structure of X.509
 - > Root CA

What is a CA ? (c.2)

- ◎ Root CA (最高層認證中心)
 - > Microsoft 翻譯成「根目錄授權憑證」
 - > 通常 Root CA 不會直接用來簽發憑證，而是授權給一些中間的認證中心，讓這些中間的認證中心來簽發憑證
 - > Root CA 自己幫自己簽名
 - 沒有再上層可以為他簽名
 - > 認可最高層認證中心
 - 經由 secure channel 安裝 Root CA 的憑證
 - > Root CA 只能由一些著名可靠的公司來擔任
 - 無法再向上查驗，所以不可隨便加進系統信任的 Root CA

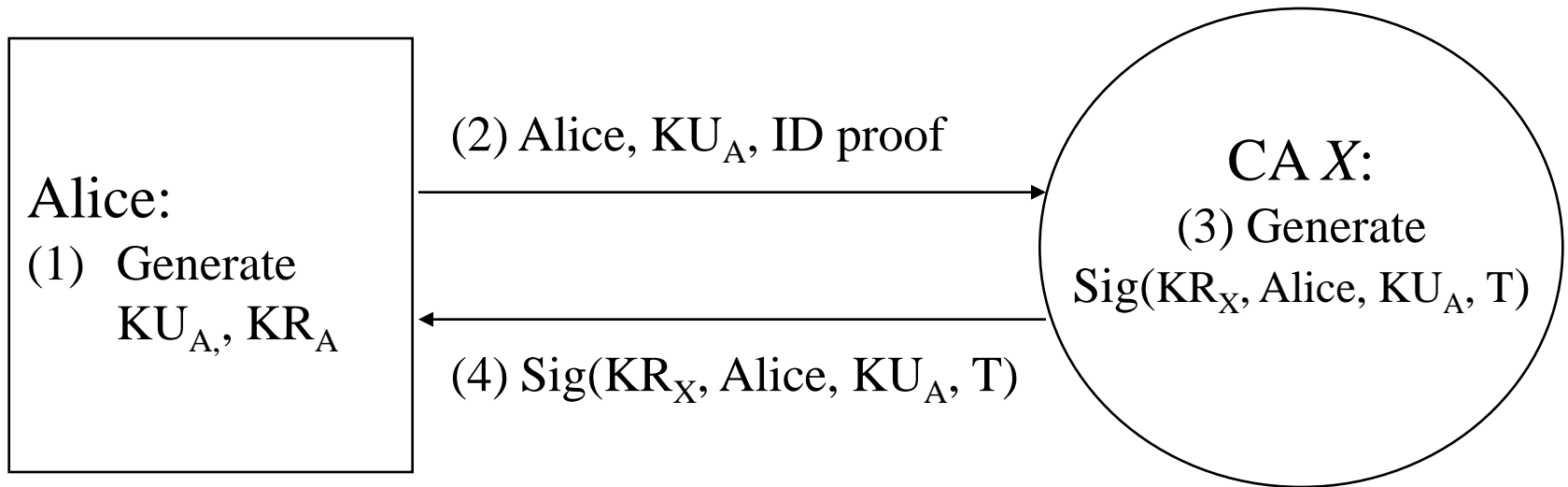
What is a CA ? (c.3)

- ◎ Tree structure of CA
 - > 每個合格的 CA，都會有一個管轄它的最高層 CA 的簽名，表示 Root CA 授權給它，可以簽發別人的憑證
 - > 當程式碰到沒見過的憑證，憑證上簽名的 CA 也沒見過時，只要檢查 Root CA 的簽名無誤，就接受這個憑證
- ◎ Cost of certificate
 - > HiTrust : NT \$30,000 / per year / per host
 - > Myself : NT \$0

Certificate

- ◎ 電子憑證 / 公開金鑰憑證 / 網路身份證
- ◎ A certificate is issued by a CA X
- ◎ A certificate of a user A consists:
 - > The name of the issuer CA X
 - > His/her public key KU_A
 - > The signature $\text{Sig}(KR_X, A, KU_A)$ by the CA X
 - > The expiration date
 - > Applications
 - Encryption / Signature

Certificate (c.1)



$\text{Cert}_{A,X} = [\text{Alice}, KU_A, \text{Sig}(KR_X, \text{Alice}, KU_A)]$

Note: CA does not know KR_A

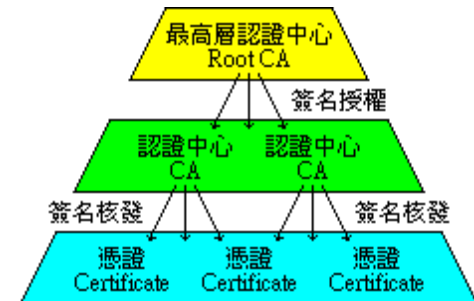
Certificate (c.2)

- ◎ Guarantee of CA and certificate
 - > Guarantee the public key is of *someone*
 - > *Someone* is not guaranteed to be *safe*
- ◎ Security of transmitting DATA
 - > Transmit *session key* first
 - *Public crypto system*
 - > Transmit DATA by session key
 - *Symmetric crypto system*

Certificate Authority (1)

◎ Certificate

- 憑證的原文是 Certificate，是附上所有人 (owner) 的資料 (公司名稱、伺服器名稱、個人真實姓名、連絡 E-mail、通訊地址等資料)，後面加上數位簽名的 Public Key。憑證上會附有幾個數位簽名，代表這些簽名的人，確認過這個 Public Key 的所有人，和憑證上所載的資料相符，沒有假造。
- 在 X.509 中，最下層每一個合格的憑證 (Certificate) 上，會有一個認證中心 (CA) 的簽名，表示這個認證中心 (CA) 檢查過，確認憑證上的所有者資料無誤。當程式碰到沒見過的憑證時，只要檢查憑證上認證中心 (CA) 的簽名無誤，即代表這個認證中心 (CA) 查核過這個憑證 (Certificate)，憑證上的資料無誤。



Certificate Authority (2)

- ◎ Certificate Authority

- > 認證中心的原文是 CA，是 Certificate Authority 的縮寫，在微軟繁體中文 WINDOWS 上翻譯成憑證授權。認證中心是 X.509 的一環。認證中心也是一種憑證，上面附有認證中心本身的資料，但不是用來加解密，而是用來簽發憑證，證明憑證所有人和憑證上所載的資料無誤。
- > 每一個合格的認證中心 (CA) 上，會有一個管轄它的最高層認證中心 (Root CA) 的簽名，表示最高層認證中心授權給它，可以簽發別人的憑證。當程式碰到沒見過的憑證，憑證上簽名的認證中心 (CA) 也沒見過時，只要檢查認證中心上附的最高層認證中心 (Root CA) 的簽名無誤，即代表這個最高層認證中心 (Root CA)，認為這個認證中心 (CA) 的憑證簽發過程很仔細，檢查資料很詳實，所以授權給它，准許它可以簽發憑證 (Certificate)。所以這個認證中心 (CA) 簽發的憑證 (Certificate)，憑證上的資料也沒有問題。
- > Reference: <http://www.imacat.idv.tw/tech/sslcerts.html>

Apache configuration – Certificate Authority (1)

○ Flow

- > Generate random seed
- > Generate RootCA
 - Generate private key of RootCA
 - Fill the Request of Certificate.
 - Sign the certificate itself.
- > Generate certificate of Web Server
 - Generate private key of Web Server
 - Fill the Request of certificate
 - Sign the certificate using RootCA
- > Modify apache configuration → restart apache

Apache configuration – Certificate Authority (2)

- > Generate random seed
 - openssl rand -out rnd-file num
 - Ex. openssl rand -out /etc/ssl/RootCA/private/.rnd 1024
 - chmod go-rwx rnd-file
 - Ex. chmod go-rwx /etc/ssl/RootCA/private/.rnd

Apache configuration – Certificate Authority (3)

> Generate RootCA

- Generate private key of RootCA

- `openssl genrsa -des3 -rand rnd-file -out rootca-key-file num`

```
% openssl genrsa -des3 -rand /etc/ssl/RootCA/private/.rnd \  
-out /etc/ssl/RootCA/private/rootca.key 2048
```

Note: phrase are asked (something like password)

- `chmod go-rwx rootca-key-file`

```
% chmod go-rwx /etc/ssl/RootCA/private/rootca.key
```

Apache configuration – Certificate Authority (4)

> Generate RootCA

- Generate private key of RootCA
- Fill the Request of Certificate.
 - `openssl req -new -key rootca-key-file -out rootca-req-file`
% `openssl req -new -key /etc/ssl/RootCA/private/rootca.key \`
 `-out /etc/ssl/RootCA/private/rootca.req`
 - `chmod go-rwx rootca-req-file`
% `chmod go-rwx /etc/ssl/RootCA/private/rootca.req`

Enter pass phrase for rootca-key-file:

Country Name (2 letter code) [AU]:TW
State or Province Name (full name) [Some-State]:Taiwan
Locality Name (eg, city) []:HsinChu
Organization Name (eg, company) [Internet Widgits Pty
Ltd]:NCTU
Organizational Unit Name (eg, section) []:CS
Common Name (eg, YOUR name) []:sysadm.cs.nctu.edu.tw
Email Address []:ta@sysadm.nctu.edu.tw

A challenge password []: (set if you want to have a password)
An optional company name []: (depends on above setting)

Apache configuration – Certificate Authority (5)

> Generate RootCA

- Generate private key of RootCA
- Fill the Request of Certificate.
- Sign the certificate itself.
 - `openssl x509 -req -days number_of_days -sha1 \`
`-extfile path_of_openssl.cnf -extensions v3_ca \`
`-signkey rootca-key-file -in rootca-req-file -out rootca-crt-`
`file`
`% openssl x509 -req -days 5109 -sha1 -extfile /etc/ssl/openssl.cnf -extensions v3_ca \`
`-signkey /etc/ssl/RootCA/private/rootca.key \`
`-in /etc/ssl/RootCA/private/rootca.req \`
`-out /etc/ssl/RootCA/private/rootca.crt \`
 - `rm -f rootca-req-file`
`% rm -f /etc/ssl/RootCA/private/rootca.req`
 - `chmod go-rwx rootca-crt-file`
`% chmod go-rwx /etc/ssl/RootCA/private/rootca.crt`

Apache configuration – Certificate Authority (6)

- > Generate certificate of Web Server
 - Generate private key of Web Server
 - openssl genrsa -out host-key-file num
%openssl genrsa -out /etc/ssl/key/sysadm.key 1024
 - chmod go-rwx host-key-file
%chmod go-rwx /etc/ssl/key/sysadm.key
 - Fill the Request of certificate
 - openssl req -new -key host-key-file -out host-req-file
% openssl req -new -key /etc/ssl/key/sysadm.key \
-out /etc/ssl/cert/sysadm.req
 - chmod go-rwx host-req-file
% chmod go-rwx /etc/ssl/cert/sysadm.req

Apache configuration – Certificate Authority (7)

- > Generate certificate of Web Server
 - Generate private key of Web Server
 - Fill the Request of certificate
 - Sign the certificate using RootCA
 - Transmit host-req-file to Root CA, and do following steps in RootCA
 - `openssl x509 -req -days number_of_days -sha1 -extfile path_of_openssl.cnf \`
`-extensions v3_ca -CA rootca-crt-file -CAkey rootca-key-file \`
`-CAserial rootca-srl-file -CAcreateserial -in host-req-file -out host-crt-file`
`% openssl x509 -req -days 361 -sha1 -extfile /etc/ssl/openssl.cnf -extensions v3_ca`
`-CA /etc/ssl/RootCA/private/rootca.crt \`
`-CAkey /etc/ssl/RootCA/private/rootca.key \`
`-CAserial /etc/ssl/RootCA/private/rootca.srl \`
`-CAcreateserial \`
`-in /etc/ssl/cert/sysadm.req \`
`-out /etc/ssl/cert/sysadm.crt`
 - `rm -f host-req-file` (in both RootCA and Web Server)
`% rm -f /etc/ssl/cert/sysadm.req`
 - Transmit host-crt-file back to Web Server

Apache configuration – Certificate Authority (8)

```
##
## SSL Virtual Host Context
##
<VirtualHost _default_:443>
# General setup for the virtual host
DocumentRoot /www/data
<Directory "/www/data">
    Options Indexes FollowSymLinks
    AllowOverride All
    Order allow,deny
    Allow from all
</Directory>
ServerName sysadm.cs.nctu.edu.tw:443
ServerAdmin ta@sabsd.cs.nctu.edu.tw
ErrorLog /var/log/httpd/sabsd.cs-error.log
CustomLog /var/log/httpd/sabsd.cs-access.log common

SSLEngine on
SSLCipherSuite
ALL:!ADH:!EXPORT56:RC4+RSA:+HIGH:+MEDIUM:+LOW:!SSLv2:+EXP:+eNULL
SSLCertificateFile /etc/ssl/cert/sysadm.crt
SSLCertificateKeyFile /etc/ssl/key/sysadm.key
```

PGP

- ◎ Pretty Good Privacy
- ◎ Public key system
 - > Encryption
 - > Signature
- ◎ security/gnupg

- ◎ Will talk more in Network Administration