

Web and Security Technologies

Chapter 5: Cryptography and SSL Access





Agenda

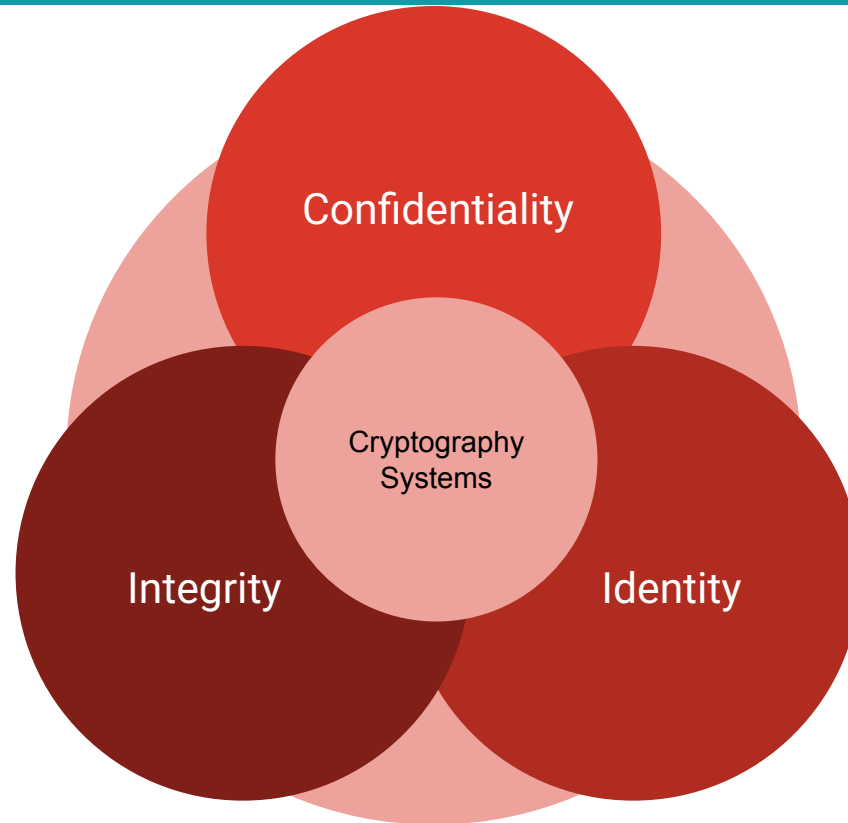
- ❑ Cryptography Basics
- ❑ Securing Website Access with SSL



Cryptography Basics



Introduction to Cryptography Systems



Manual Security

Invoice Detail

Invoice

Invoice Number: 1035
Date: Oct 11, 2001
Salesperson: Sally Jones

Ship To:

Doug's High Tech Services
383 West 600 North
American Fork, UT 84003

Bill To:

Doug's High Tech Services
383 West 600 North
American Fork, UT 84003

PO Number	Terms	Due	Term Discount	Discount Date
7895874	Net 30	Nov 10, 2001	0%	

Item	Description	Quantity	Price	Total
Toner 2	Toner Cartridge	3	\$444.00	\$1,332.00
X191U	XEROX X190 Copier - Used	1	\$653.05	\$653.05
257JH2	Paper	3	\$112.00	\$336.00
1968168	Staples	4	\$46.00	\$184.00
168168	Developer	4	\$200.00	\$800.00

Total

Discount:	(\$0.00)
Sub Total:	\$3,305.05
Tax:	\$206.56
Shipping:	\$0.00
Total:	\$3,511.61
Total Due:	\$3,511.61



CITY OF HOUSTON

Public Works and Engineering
Department

Interoffice

Correspondence

To: Council Member Anne Clutterbuck

From: Director

Date: May 25, 2006

Subject: HIGHLAND VILLAGE NEIGHBORHOOD
TRAFFIC PROJECT

The Department of Public Works and Engineering has completed the evaluation period for partial road closures on Suffolk Drive and Drexel Drive south of Westheimer. Our analysis of data collected prior to the partial closures concluded that the "cut through" traffic traveling between Westheimer and Richmond does not constitute a significant percentage of the overall traffic affecting this area. "Cut through" traffic was found traveling through the neighborhood from Alabama and Wesleyan to Drexel and Westheimer. We found that the partial closures were not effective for the southbound traffic and did not deter the morning traffic coming from Alabama and Wesleyan to the east side of the neighborhood. Field observations found that a significant number of vehicles were ignoring the closures and driving around them. Our survey of neighborhood residents found that the issue of closures divided the neighborhood, with those closest to the closure points generally happy with them and those living away being opposed (89 for the closures and 144 opposed). Based on this analysis, we recommend the removal of these closures.

We agree that several alternative approaches suggest that you have applicability, and could result in benefits outweighing any potential drawbacks. We propose to implement the following:

1. We will closely with the Houston Police Department to identify areas for increased enforcement of speed limits in the area, especially on Suffolk and Drexel Drive.
2. As a temporary step, we will install and place temporary (trailer mounted) radar speed display signs at southbound locations on Suffolk and Drexel Drive. Since we do not currently have such signs in our inventory, we initiate the process, and we request your assistance in identifying appropriate locations for such signs. We have been advised that permanent, powered radar speed display signs, and believe that such signs could be installed for approximately \$1,000 per location. We propose to evaluate the effectiveness of the temporary signs and consider future permanent placements as part of the street improvement projects further discussed below.
3. As part of our efforts under Neighborhood Street Reconstruction Project 458, scheduled for commencement of design work later this year, we will evaluate the feasibility and support for various traffic approaches, including chicanes, roundabouts and medians. It should be recognized that such measures will have noticeable impacts on parking along affected streets, and we will work closely with you during the evaluation of the feasibility and support for such improvements.
4. In an effort to improve pedestrian safety, the Neighborhood Street Reconstruction Project will also provide sidewalks at locations where the neighborhood currently lacks sidewalks.

MSM:JSW-JW/jr

cc: Waynette Chan
John Whaley

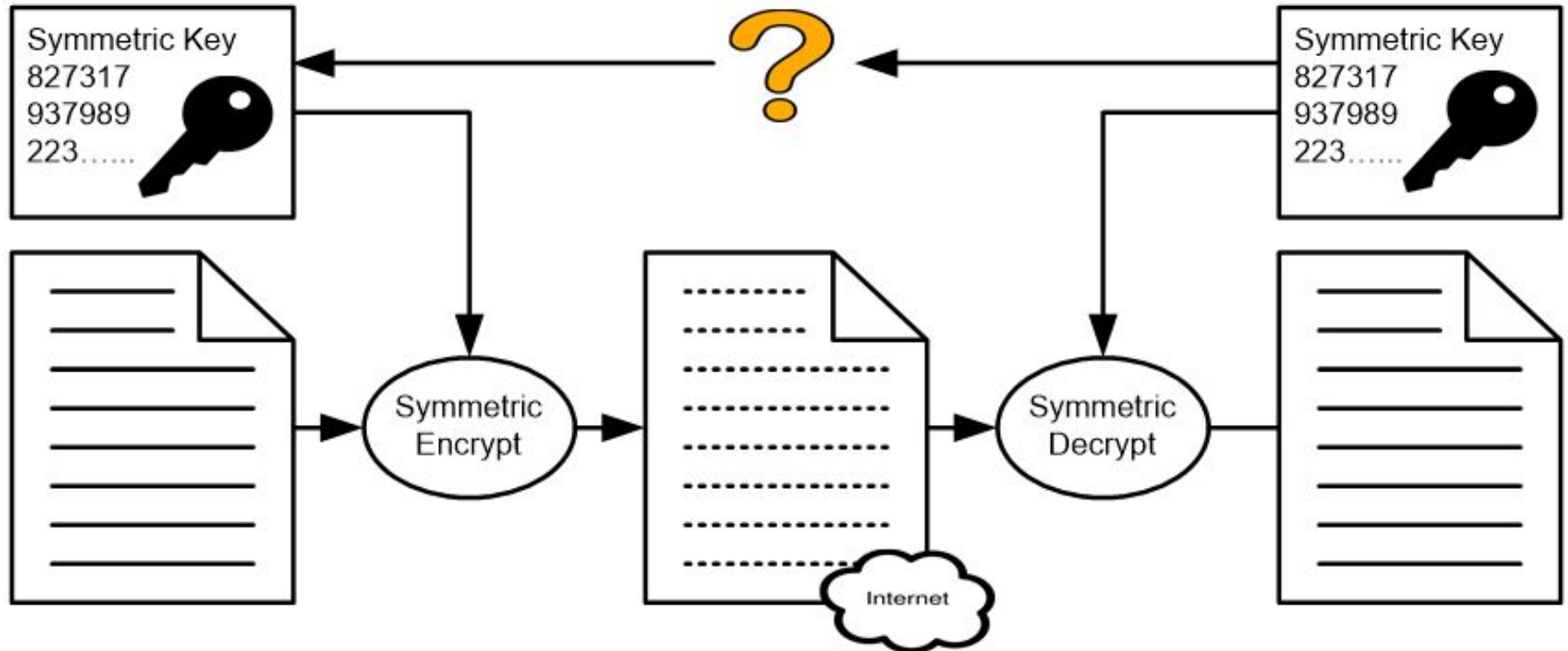
Katherine P.
Jeff Weatherford

Michael J. Deane, P.E., DEE

- Identity?
- Integrity?
- Confidentiality?



Confidentiality



Confidentiality

- Convert data to unreadable form → **Encryption Operation**
- Recover encrypted data → **Decryption Operation**
- Conversion and recovery method → **Encryption Technique/Algorithm/Cipher**
- Secret value used for encryption/decryption → **Key**
- Owners should share key with others to give them access to data → **Key Exchange**
- Because the same key is used for Encryption/Decryption → **Symmetric Encryption, Symmetric Decryption, Symmetric Cipher, Symmetric Key, Symmetric Key Exchange**



Known Symmetric Ciphers

- DES (Key Length = 8 Byte) - **Legacy**
- 3DES (Key Length = 16/24 Byte) - **Legacy**
- AES/Rijndael (Key Length = 16/24/32 Byte) - **Standard**
- RC4 (Stream Cipher)
- RC5, RC6
- Blowfish, Twofish, Cast
- More ...



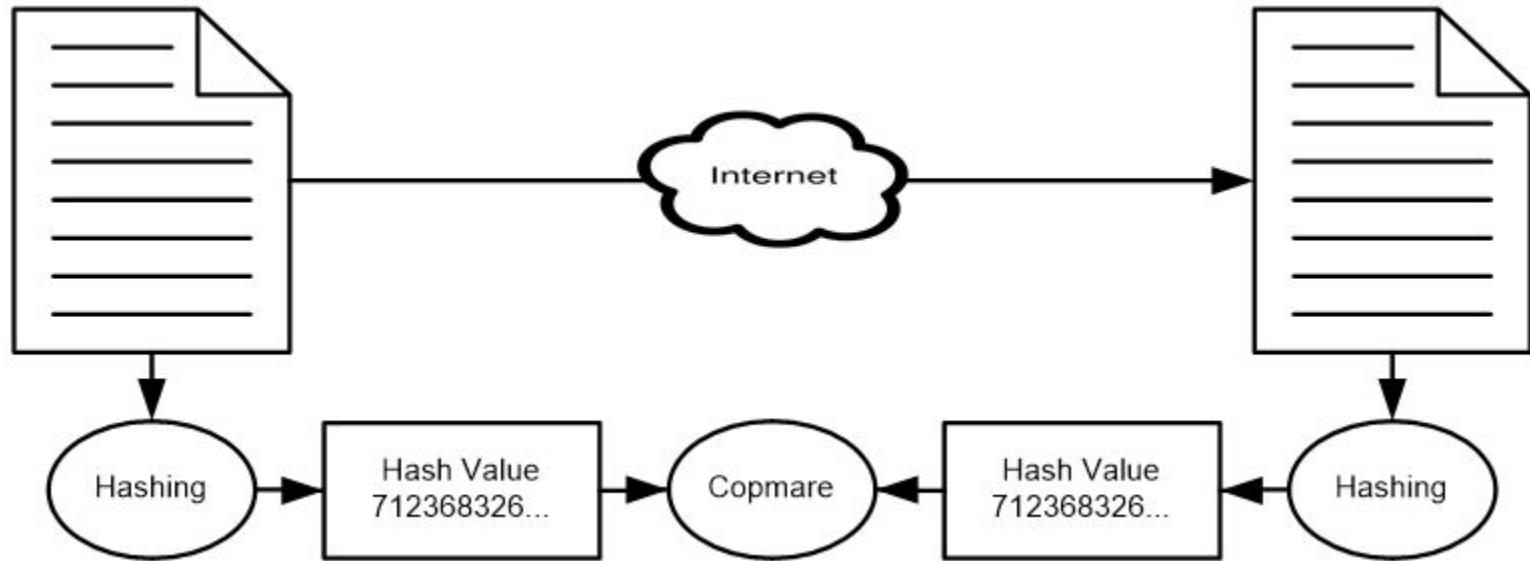
Symmetric Ciphers Strength



Super Computers requires more
than 10^{50} year to break *Symmetric*
Technique like *AES/256*



Integrity



Integrity

- Generating numeric record of fixed length of any given data → **Hashing Technique/Methods/Algorithm**
- The generated numeric record → **Hash Value**
- Regenerating the numeric record and comparing it with original one → **Integrity Check**



Known Integrity Algorithms / Hashing Techniques

- CRC16, CRC32, CRC64 (Checksum) → **Legacy**
- MD2 (128), MD4 (128), MD5 (128), MD6 (512) → **Legacy**
- SHA1 (196) → **Legacy**
- SHA2 (224/256/384/512) → **Standard**
- Whirlpool (512)
- More ...



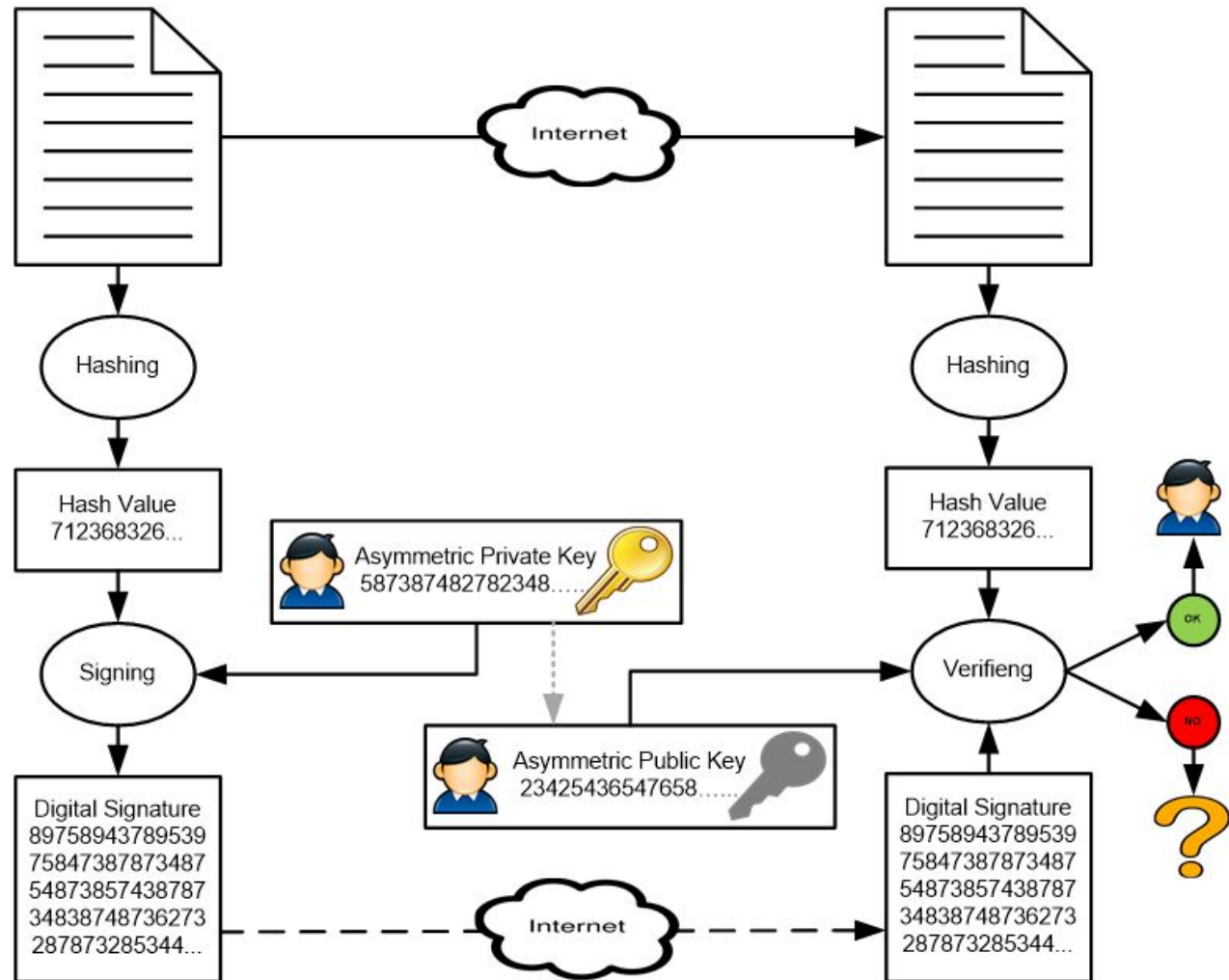
Hashing Techniques Strength



Super Computers requires more
than 10^{50} year to break *Hash*
Technique like *SHA/256*



Identity



Identity

- Generate numeric record with user private key and document → **Signing Operation**
- The generated numeric record → **Signature**
- Verifying the correctness of signature with user public key and document in addition to integrity check → **Verification Operation**
- Method of generating and verifying the signature → **Signing Technique/Algorithm/Cipher**
- Because there were two keys public and private → **Asymmetric Cipher, Asymmetric Key**
- User public key must be announced for others → **Public Key Distribution**



Known Asymmetric Ciphers

- RSA (1024, **2048**, **4069**, **8192**) → **Standard**
- EC (223, 255, 383, **521**) → **Standard**
- DSA
- ELGamal
- More ...



Asymmetric Ciphers Strength



Super Computers requires more than 10^{55} year to break **Asymmetric Technique** like **RSA/2048** or **EC/256**

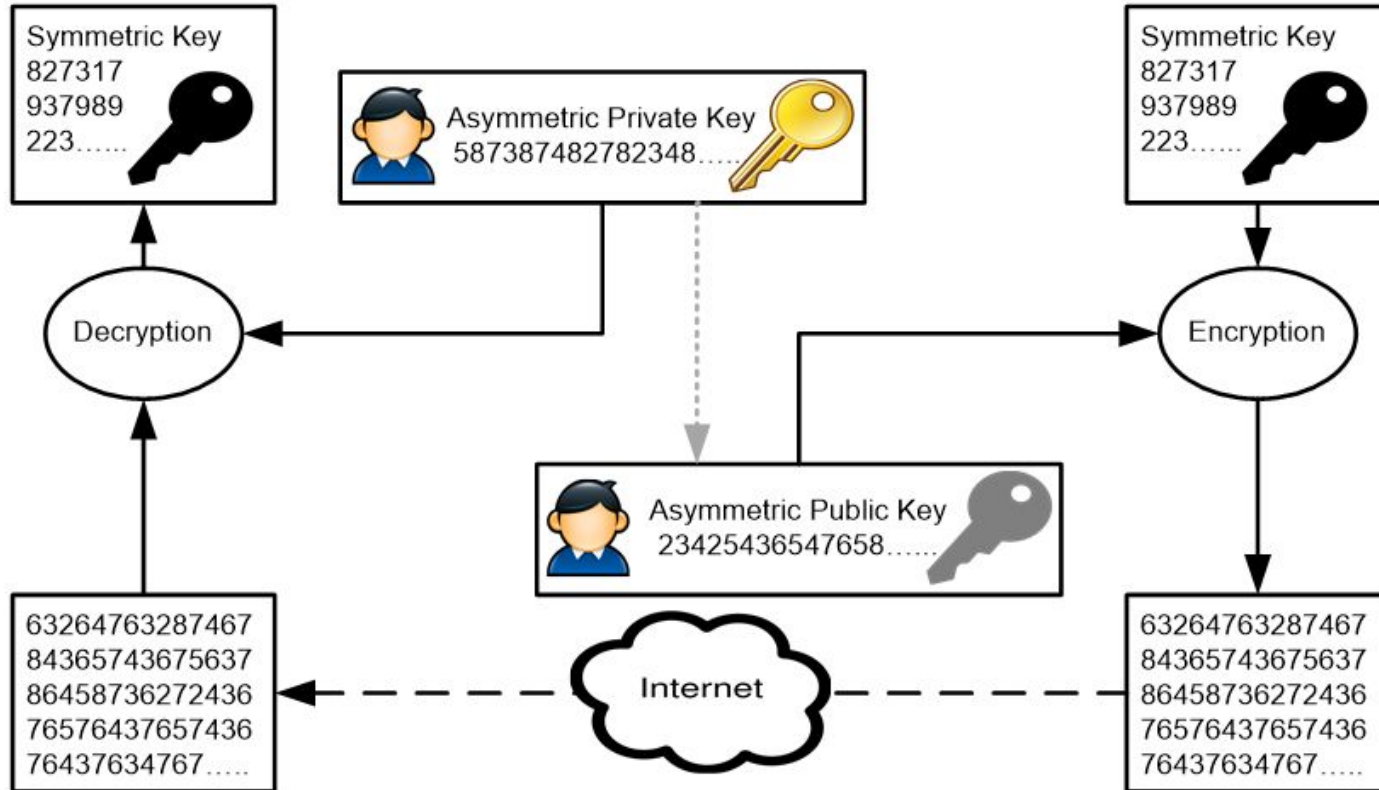


Additional Cryptographic Topics

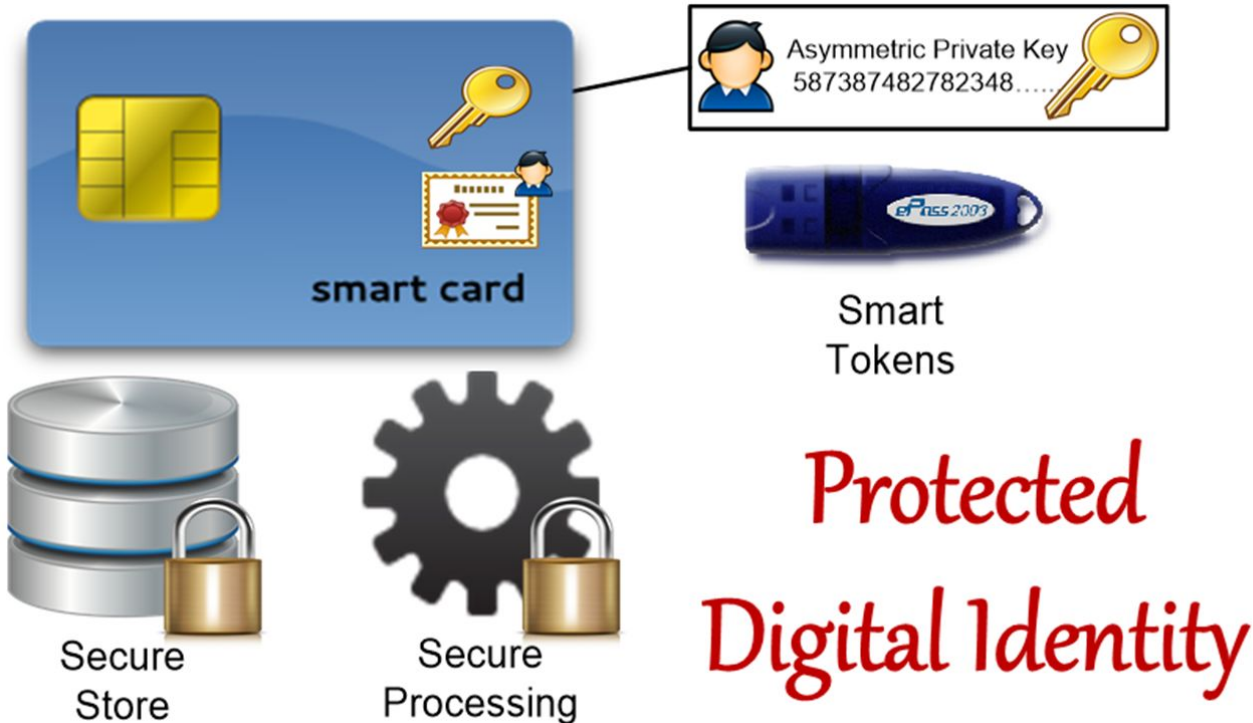
- Secure Symmetric Key Exchange
- Private Key Generation, Processing and Storage
- Trusted Public Key Distribution



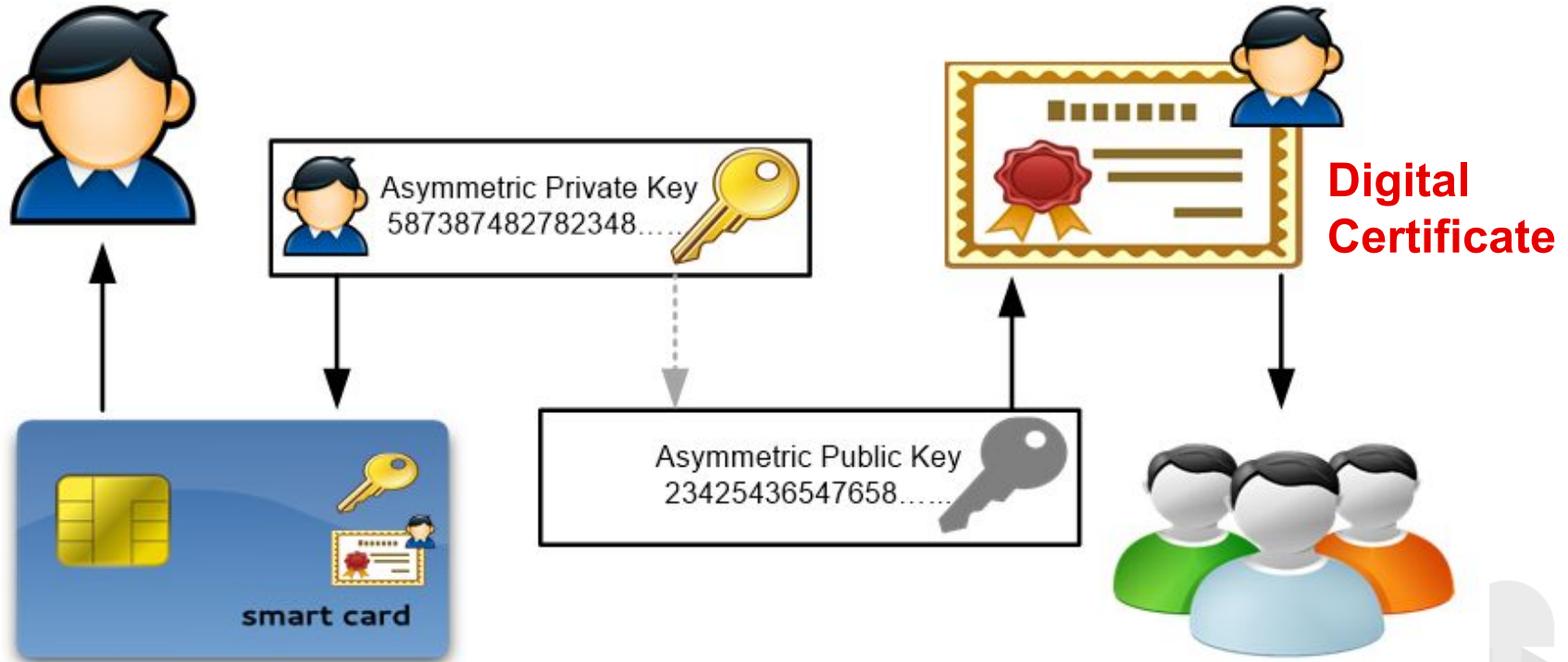
Symmetric Key Exchange



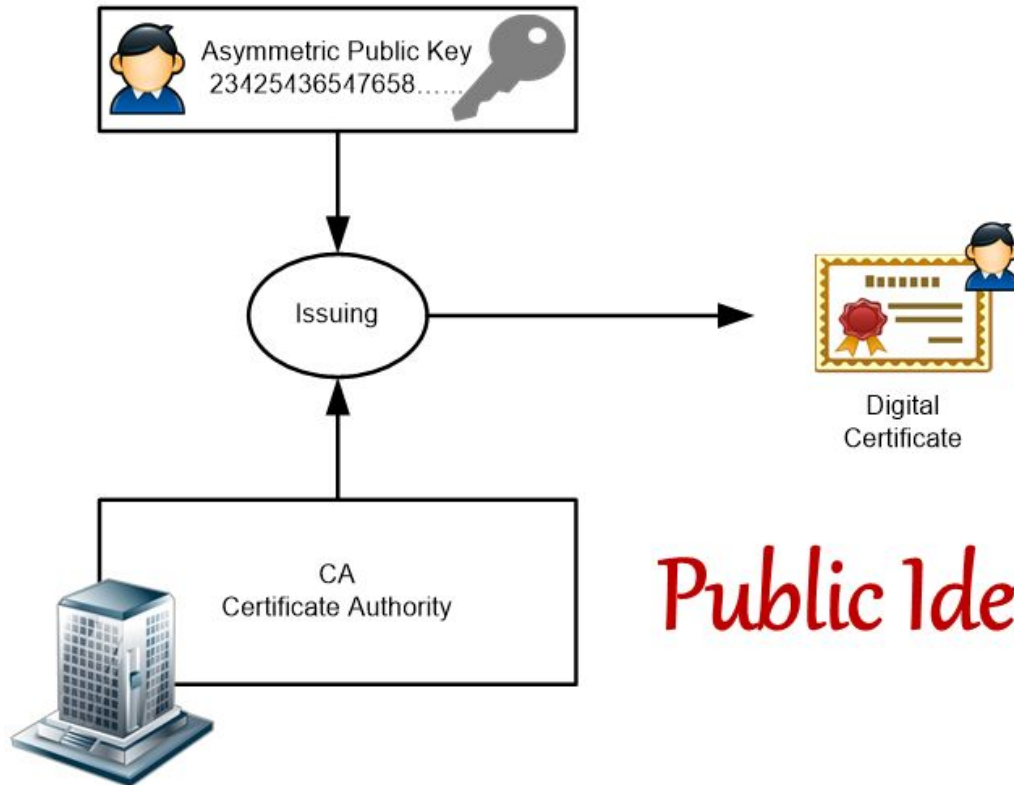
Private Key Generation, Processing and Storage → Digital Identity



Digital Certificate: Trusted Public Key Distribution



Digital Certificate Issuing



Public Identity



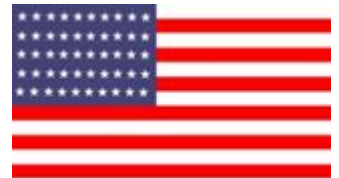
Digital Signature Law - USA 1997

Electronic Record

means a record created, generated, sent, communicated, received, or stored by electronic means

Electronic signature

means an electronic sound, symbol, or process attached to or logically associated with a record and executed or adopted by a person with the intent to sign the record



Digital Signature Law - Egypt 2004

المحرر الالكتروني

رسالة تتضمن معلومات تنشأ أو تدمج ، أو تخزن ، أو ترسل أو تستقبل كلياً أو جزئياً بوسيلة الكترونية أو رقمية أو ضوئية أو بأية وسيلة أخرى مشابهة

التوقيع الالكتروني

ما يوضع على محرر الكتروني ويتخذ شكل حروف أو أرقام أو رموز أو إشارات أو غيرها ويكون له طابع متفرد يسمح بتحديد شخص الموقع ويميزه عن غيره

الوسيط الالكتروني

أداة أو أدوات أو أنظمة إنشاء التوقيع الالكتروني



Securing Website Access with SSL



Preparation

- Install Apache + Composer + PHP 8.x + Laravel 11.x + MySQL/MariaDB (For Windows use XAMPP and Composer)
- Clone the WebSec project from following Github repository at (<https://github.com/sutengtech/websec>) to “xampp/htdocs/websec”
- Restart the Apache Server



Preparation

```
<VirtualHost *:80>
    DocumentRoot "D:\xampp\htdocs\websec\WebSecService"
    ServerName websecservice.localhost.com
</VirtualHost>
```

```
127.0.0.1    websecservice.localhost.com
```

```
<IfModule mod_rewrite.c>
    RewriteEngine on
    RewriteCond %{REQUEST_URI} !^(public)
    RewriteRule ^(.*)$ public/$1 [L]
</IfModule>
<IfModule mime_module>
    AddHandler application/x-httpd-ea-php82 .php .php8 .phtml
</IfModule>
```

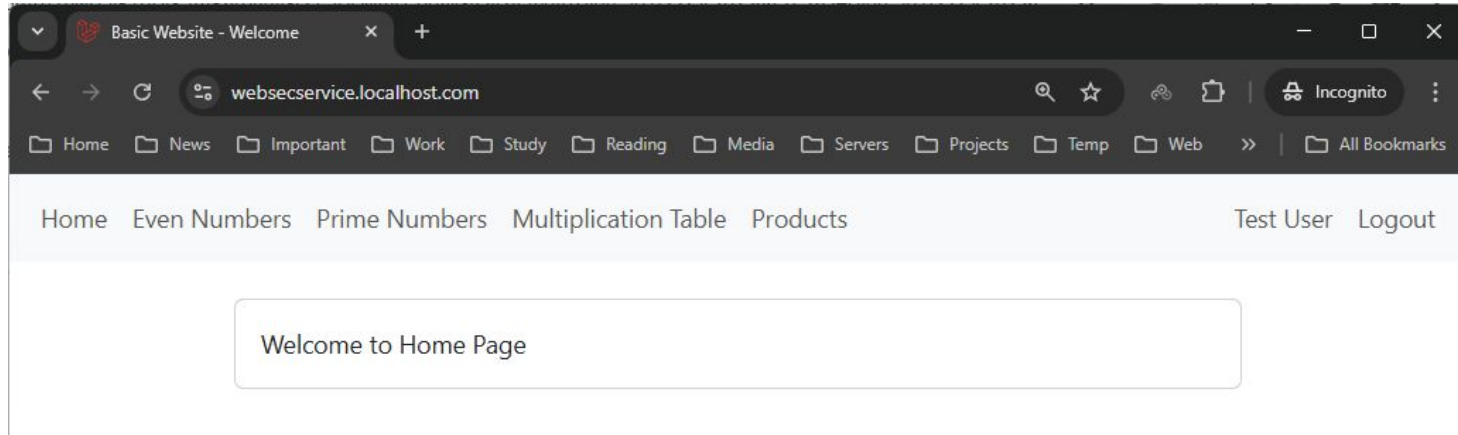
Modify the virtual Hosts File at
"xampp\apache\conf\extra"

Modify System Host File at
"Windows\System32\drivers\etc"

Copy .htaccess to the root of
"xampp\htdocs\websec\WebSecS
ervice"

Preparation

- Restart the Apache Server
- Open the WebSecService URL: <http://websecservice.localhost.com>
- Register user with your email “useremail@domain.com” for example.



OpenSSL

<https://openssl-library.org>

OpenSSL is a robust, open-source toolkit for general-purpose cryptography and secure communication. It provides a comprehensive set of cryptographic functions and is widely used to implement the SSL and TLS protocols.

SSL/TLS Implementation: OpenSSL provides the core functionality for implementing the Secure Sockets Layer (SSL) and Transport Layer Security (TLS) protocols, which are essential for securing internet communications.

Cryptographic Library: It includes a general-purpose cryptographic library (libcrypto) that offers a wide range of cryptographic algorithms for: Encryption and decryption, Key generation and management, Digital signatures and Certificate handling.



OpenSSL

Command-Line Tool: OpenSSL also provides a command-line tool (openssl) for various cryptographic tasks, such as: generating private keys, creating and managing X.509 certificates and Performing cryptographic operations.

Wide Applicability: OpenSSL is used in a variety of applications and systems, including: Web servers (e.g., Apache, Nginx), Email servers, Virtual Private Networks (VPNs), Other network applications that require secure communication



How OpenSSL Works:

OpenSSL works by providing a set of functions and tools that developers can use to implement cryptographic operations within their applications. It supports a wide range of cryptographic algorithms, including:

- **Symmetric encryption:** AES, Blowfish, etc.
- **Asymmetric encryption:** RSA, Elliptic Curve Cryptography (ECC)
- **Hashing algorithms:** SHA-256, SHA-3, etc.



OpenSSL Advantages

- **Open Source:** Being open source, OpenSSL is freely available and can be inspected, modified, and distributed by anyone.
- **Wide Adoption:** OpenSSL is one of the most widely used cryptographic libraries, making it a standard in many industries.
- **Cross-Platform Compatibility:** It supports various operating systems, including Linux, macOS, Windows, and others.
- **Rich Functionality:** OpenSSL provides a comprehensive set of cryptographic tools and algorithms, making it suitable for diverse security needs.



OpenSSL Alternatives

Open Source:

- LibreSSL
- BoringSSL
- GnuTLS.
- mbed TLS
- Libsodium
- EJBCA PrimeKey

Commercial:

- Entrust
- Thales
- Verisign
- Comodo
- wolfSSL
- AWS Certificate Manager



Installation

- Download and install openssl.
- For windows find openssl bin path (EX: C:\Programs\OpenSSL\Bin)
- Make sure to set environmental variables to openssl bin path
- Make sure that XAMPP with Apache is installed and running
- Make sure that WebSecService example is up and running with following virtual host: <http://websecservice.localhost.com>



Generate Certificates

- **Root Certificate**
 - O: WebSec, OU: WebSec Service, CN: WebSec Root CA
- **Website Certificate**
 - O: WebSec, OU: WebSec Service, CN: websecservice.localhost.com
 - Subject Alternative Name: DNS:websecservice.localhost.com
- **User Certificate**
 - O: WebSec User, OU: WebSec Service, CN: useremail@domain.com



Generate CA Certificate

```
openssl genrsa -out ca.key 2048
```

```
openssl req -x509 -new -key ca.key -days 3650 -out  
ca.crt -subj "/C=EG/ST=Cairo/L=Cairo/O=WebSec  
Course/OU=WebSec Service/CN=WebSec Root CA"
```



ca.crt

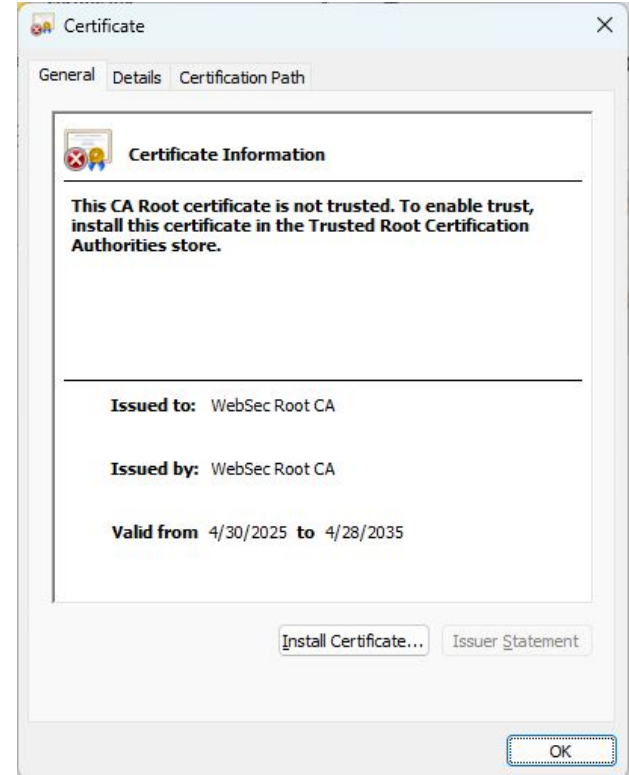


ca.key

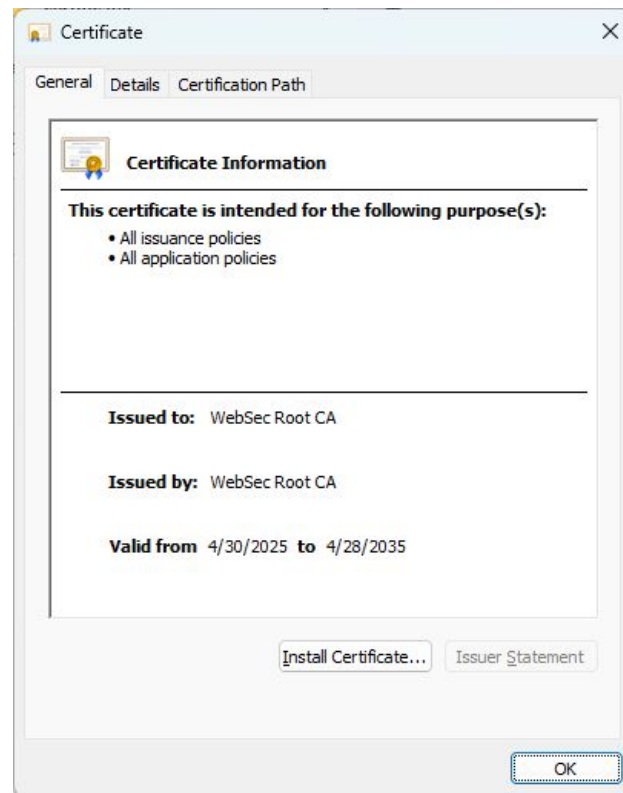
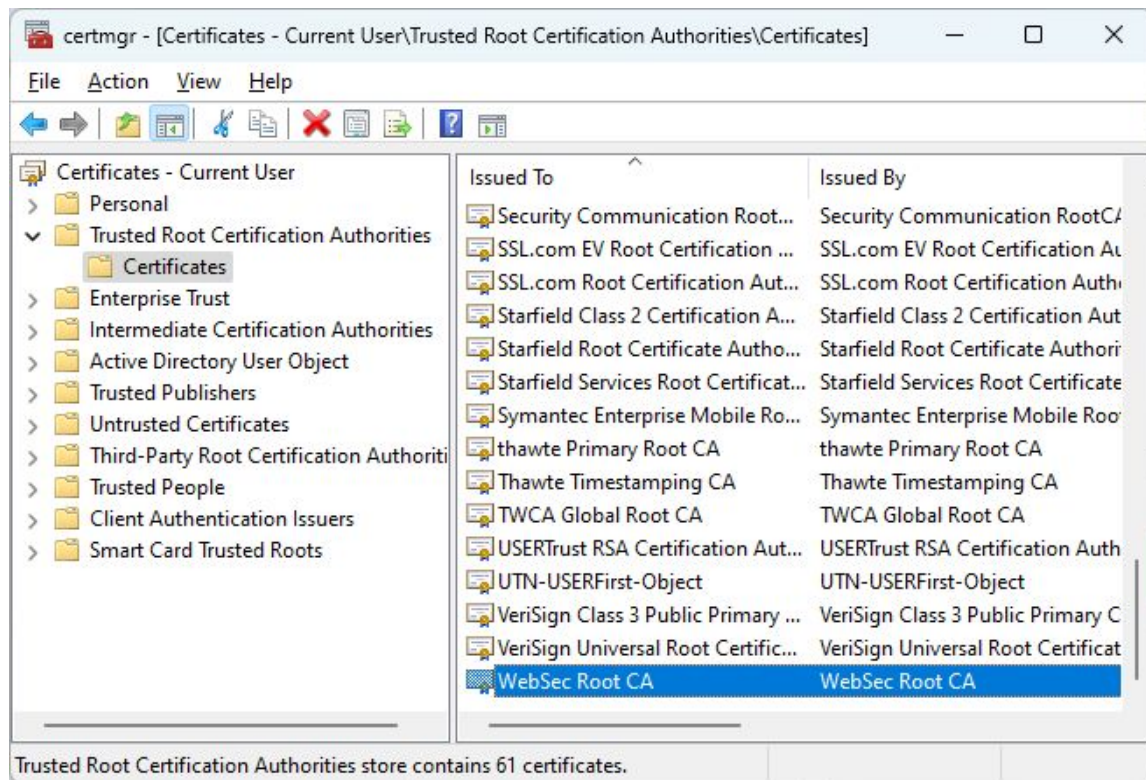


Install Root CA

- Double click on CA.CRT file
- Click Install
- Select Local Machine
- Select Trusted Root Certification Authorities
- Run “certmgr” from windows “start” menu to check that the certificate is added successfully.
- Also use “certmgr” to remove unused or old certificates.



Install Root CA



Generate Website Certificate

```
[req]
prompt = no
distinguished_name = dn
req_extensions = v3_req
```





```
[dn]
C=EG
ST=Cairo
L=Cairo
O=WebSec Course
OU=WebSec Service
CN=websecdservice.localhost.com
```

```
[v3_req]
basicConstraints = CA:FALSE
keyUsage = digitalSignature, keyEncipherment
subjectAltName = DNS:websecdservice.localhost.com
```

Generate Website Certificate

```
basicConstraints = CA:FALSE  
keyUsage = digitalSignature, keyEncipherment  
subjectAltName = DNS:websecservice.localhost.com
```

websecservice.localhost.com.ext

Name	
 ca.crt	ca.crt
 ca.key	ca.key
 websecservice.localhost.com.cnf	websecservice.localhost.com.cnf
 websecservice.localhost.com.ext	websecservice.localhost.com.ext

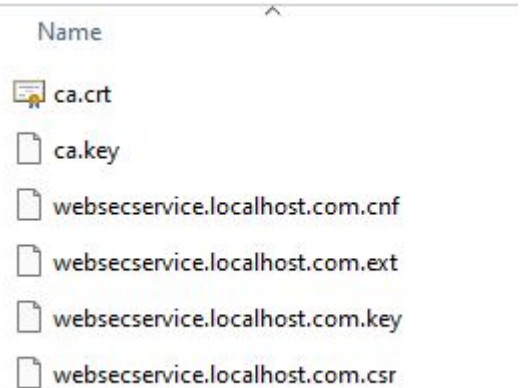
Generate Website Certificate

```
openssl genrsa -out websecdservice.localhost.com.key 2048
```

```
openssl req -new -key websecdservice.localhost.com.key -out  
websecdservice.localhost.com.csr -config  
websecdservice.localhost.com.cnf
```

```
openssl x509 -req -in websecdservice.localhost.com.csr -CA  
ca.crt -CAkey ca.key -CAcreateserial -out websecdservice.
```

```
localhost.com.crt -days 730 -extfile  
websecdservice.localhost.com.ext
```



Configure XAMPP/Apache

xampp\apache\conf\extra\httpd-ssl.conf

```
...
<VirtualHost *:443>
    ServerName websecservice.localhost.com:443
    DocumentRoot "D:/xampp/htdocs/websec/WebSecService"
    ServerAdmin admin@websecservice.localhost.com
    ErrorLog "D:/xampp/apache/logs/error.log"
    TransferLog "D:/xampp/apache/logs/access.log"

    SSLEngine on
    SSLCertificateFile "conf/ssl.crt/websecservice.localhost.com.crt"
    SSLCertificateKeyFile "conf/ssl.key/websecservice.localhost.com.key"
</VirtualHost>
```

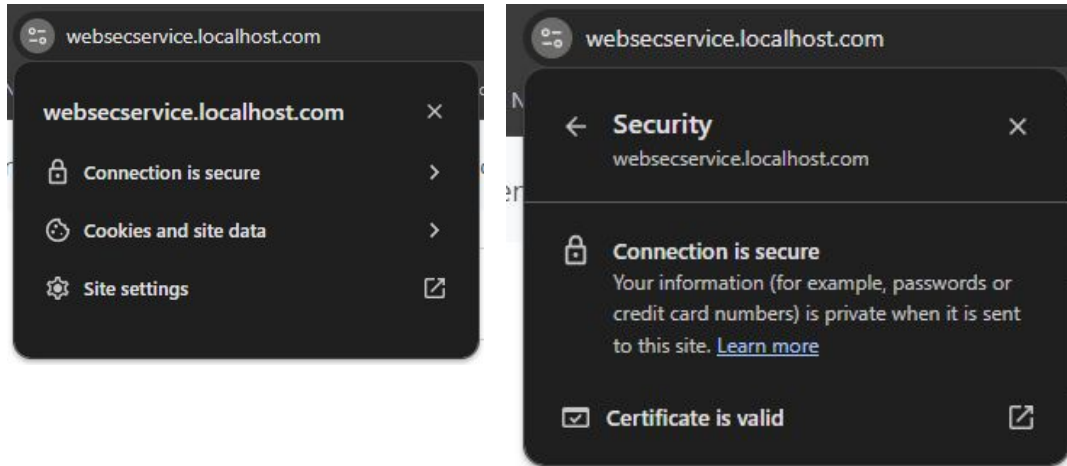
Configure XAMPP/Apache

- Copy “websecdservice.localhost.com.key” to “xampp\apache\conf\ssl.key”
- Copy “websecdservice.localhost.com.crt” to “xampp\apache\conf\ssl.crt”
- Restart the Apache from XAMPP
- Hints:
 - Make sure that windows firewall does not block 443 port
 - Make sure that you do not use 443 port for other application
 - Make sure the SSL is enabled in “xampp\apache\conf\httpd.conf”



Secure Website Access using SSL is Ready

- Open any browser and visit <https://websecservice.localhost.com>
- Check security icon and preview website certificate.



Block Non Secure Access

xampp\apache\conf\extra\httpd-vhosts.conf

```
<VirtualHost *:80>  
    ServerName websecservice.localhost.com  
    Redirect permanent / https://websecservice.localhost.com/  
</VirtualHost>
```

Try now access the non secure website <http://websecservice.localhost.com> it redirect to the secure one.



Login with User Certificate

- It is required now to login with user certificate.
- This means we need to generate a certificate for every user.
- The user is identified by his/her email.
- So we will generate a certificate based on user email.
- Alternatively, you can use supply other values to certificate like National ID, Social Security Number, Unique Name, Student ID ...



Generate User Certificate

```
[req]
prompt = no
distinguished_name = dn
req_extensions = v3_req

[dn]
C=EG
ST=Cairo
L=Cairo
O=WebSec Course
OU=Users
CN=useremail@domain.com
emailAddress=useremail@domain.com

[v3_req]
basicConstraints = CA:FALSE
keyUsage = digitalSignature
extendedKeyUsage = clientAuth
subjectAltName = email:useremail@domain.com
```

Generate User Certificate

```
basicConstraints = CA:FALSE  
keyUsage = digitalSignature  
extendedKeyUsage = clientAuth  
subjectAltName = email:useremail@domain.com
```

useremail@domain.com.ext



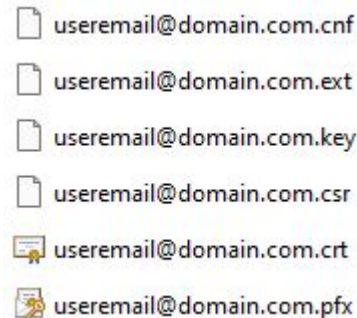
Generate User Certificate

`openssl genrsa -out useremail@domain.com.key 2048`

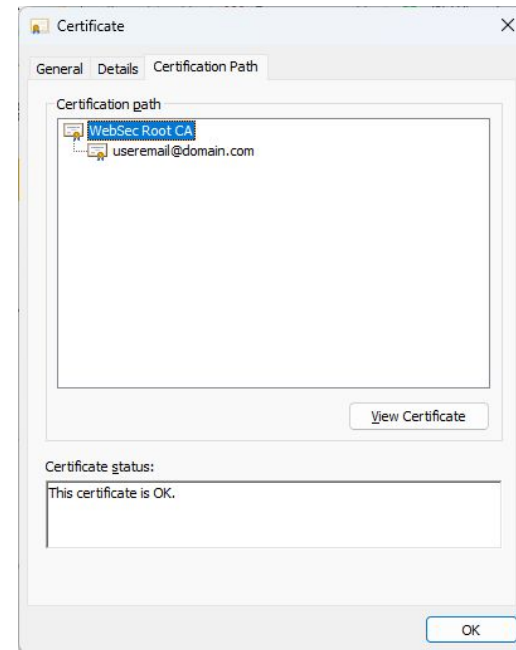
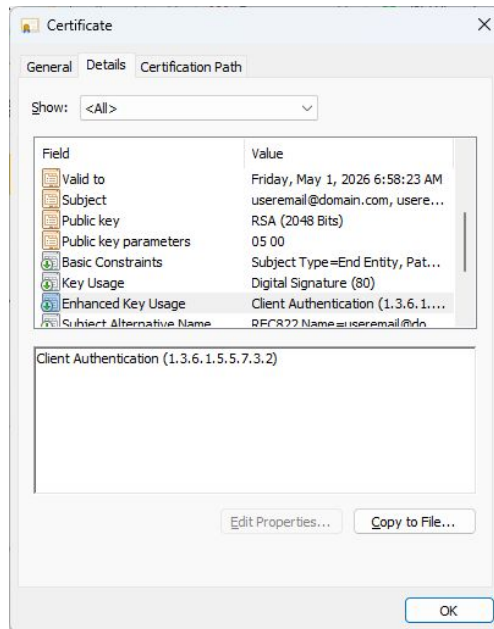
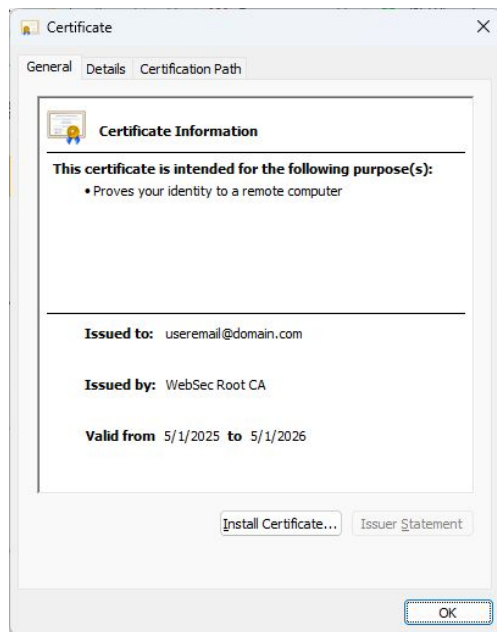
`openssl req -new -key useremail@domain.com.key -out
useremail@domain.com.csr -config useremail@domain.com.cnf`

`openssl x509 -req -in useremail@domain.com.csr -CA ca.crt
-CAkey ca.key -CAcreateserial -out useremail@domain.com.crt
-days 365 -extfile useremail@domain.com.ext`

`openssl pkcs12 -export -out useremail@domain.com.pfx -inkey
useremail@domain.com.key -in useremail@domain.com.crt
-certfile ca.crt`



Generate User Certificate



Install User Key + Certificate in the Secure Certificate Store (SW)

- Double click on the [useremail@domain.com.pfx](#) file
- Select Current User
- Supply the Password
- Install the certificate into “Personal” certificate store.
- The key is also stored in internal key store.
- The certificate used to verify your membership to “WebSec Root CA”.
- The key will be used to sign the login request to the website.



Modify SSL Configuration in XAMPP/Apache

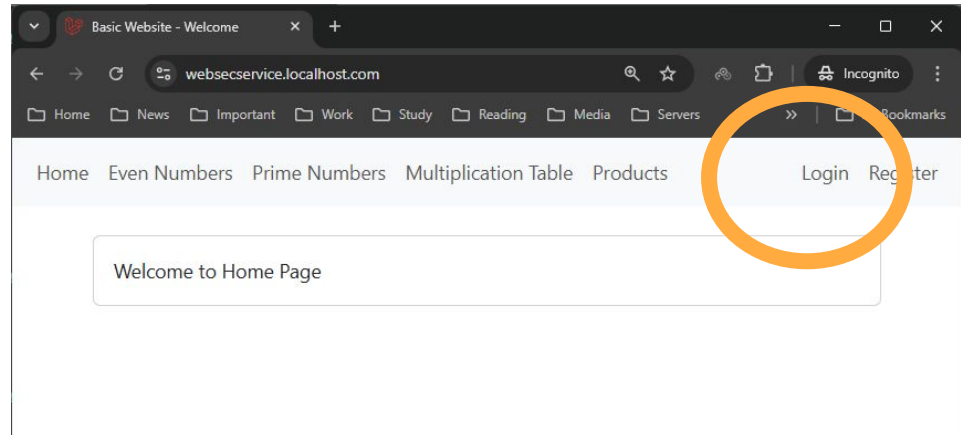
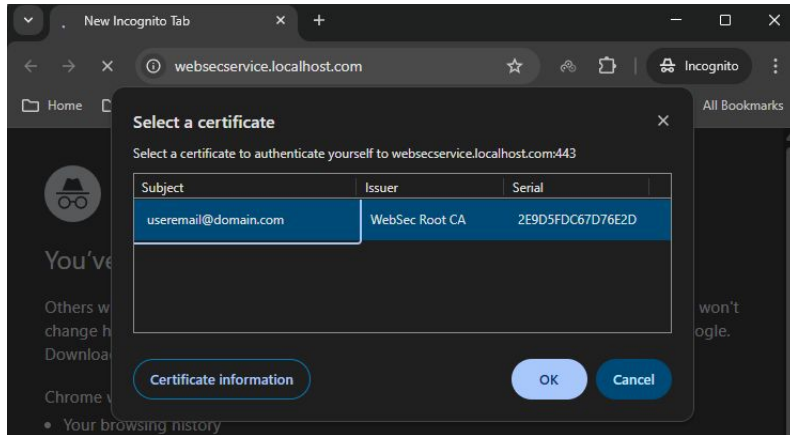
xampp\apache\conf\extra\httpd-ssl.conf

```
...  
<VirtualHost *:443>  
    ...  
    SSLCACertificateFile "conf/ssl.crt/ca.crt"  
    SSLVerifyClient require  
    SSLOptions +ExportCertData  
</VirtualHost>
```



Secure Website Access using SSL is Ready

- Make sure that user with email “useremail@domain.com” already have active account.
- Restart Apache Server
- Open any browser and visit <https://websecdservice.localhost.com>



Recognize User Certificate and Do Login

- In previous scenario only user with valid certificate can access the website.
- Now we need to recognize the user and perform automatic login.
- Simply when the user certificate is accepted by the server.
- The certificate will be available at backend inside the `$_SERVER` variable.
- However the certificate is available in encoded format.
- So we need to use PHP openssl library to decode the certificate and find out all required certificate information [Common Name, Email, Organization, Organization Unit, Usage, Extended Usage, Expiration ... and more].
- We will write a simple helper function to extract only the email.



```
if (!function_exists('emailFromLoginCertificate')) {  
    function emailFromLoginCertificate()  
    {  
        if (!isset($_SERVER['SSL_CLIENT_CERT'])) return null;  
  
        $clientCertPEM = $_SERVER['SSL_CLIENT_CERT'];  
  
        $certResource = openssl_x509_read($clientCertPEM);  
        if (!$certResource) return null;  
  
        $subject = openssl_x509_parse($certResource, false);  
  
        if (!isset($subject['subject']['emailAddress'])) return null;  
  
        return $subject['subject']['emailAddress'];  
    }  
}
```



Modify Home Page Route

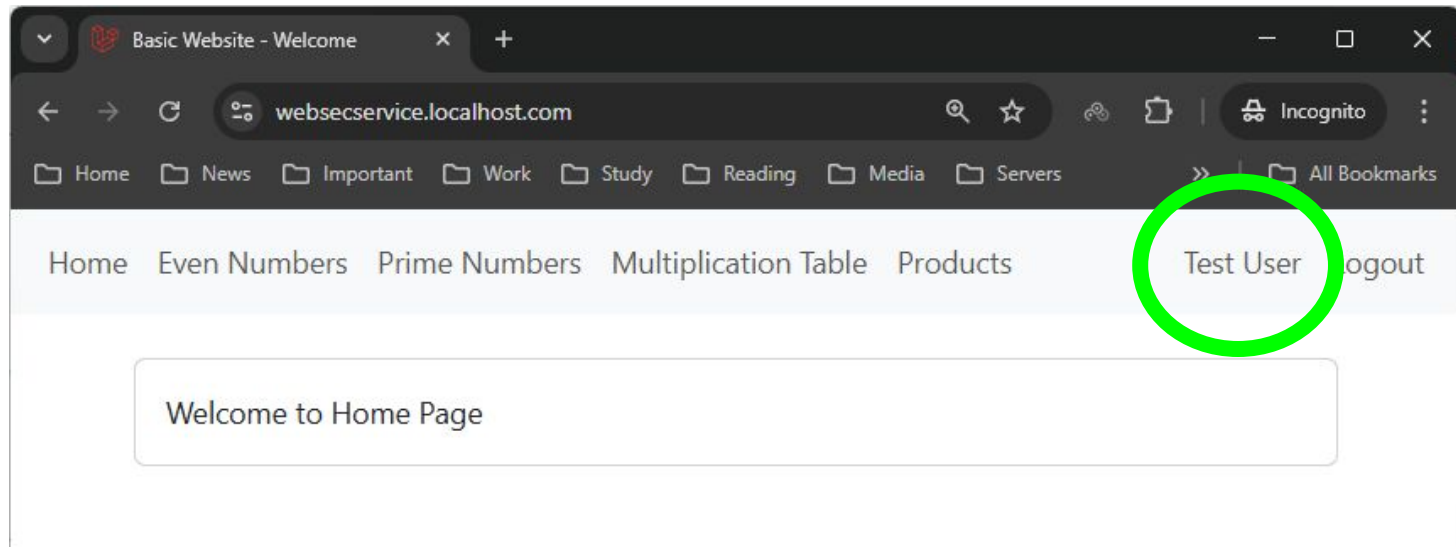
routes\web.php

```
Route::get('/', function () {  
  
    $email = emailFromLoginCertificate();  
    if($email && !auth()->user()) {  
  
        $user = User::where('email', $email)->first();  
        if($user) Auth::login($user);  
    }  
  
    return view('welcome');  
});
```



Try Access the Secure WebSecService with Login Certificate

- Try Access the Secure WebSecService with Login Certificate Again
- Open any browser and visit <https://websecservice.localhost.com>



Smart Card / Token / SIM / eSIM

- Operating System Secure Certificate Store at the end is a software store.
- Legally software store is not certified. However it is accepted for business purposes. Business is caring about security only.
- In Order to full legality you must use certified secure certificate store.
- Certified secure certificate store is a hardware element certified with Common Criteria EAL5+ or FIPS 140-2.
- Hardware element may take several forms: USB Token, Smart Card, SIM Card
- Many processor comes with integrated certified secure hardware store like eSIM



Smart Card/Token/SIM/eSIM



Using Smart Card / Token

- Bring Smart Card then connect it to your computer/mobile using card reader or NFC reader if the card support wireless access. Alternatively; Bring Smart Token and connect it to your computer via USB port.
- Install the related token/card software make sure it is running and the token is recognised.
- Install the [useremail@domain.com.pfx](#) using the Smart Token/Card tool.
- Most of the hardware token/card has has seamless integration with operating system certificate store. So you can view it from within system certificates viewer.



Using Smart Card / Token

- Finally, you can access <https://websecservice.localhost.com> after selecting the certificate and additionally authenticate with your token with PIN/Fingerprint/...
- For eSIM you have to authenticate yourself using available mobile authentication Face/PIN/Pattern/Fingerprint.

