Secure Communication

USABILITY AND NECESSITY OF SSL / TLS
We’re going to talk about:

1. **Why** is this important?

2. **What** is SSL?

3. **Where** is SSL being used?

4. **Features**: What to look for in an SSL library?
Why is This Important?

• Number of connected devices is ever increasing

• Frequent Road-blocks:
  – Lack of understanding
  – Insufficient funding
  – Tight deadlines
Why is This Important?

Ivan Ristic: Internet SSL Survey 2010
http://www.ssllabs.com

• Alexa Top 1M Sites
  120,000 Use SSL (12%)
What is SSL?

X509, Encryption, handshakes, and more.
What is SSL?

- Enables secure client / server communication, providing:

  - **Privacy**: Prevent eavesdropping
  - **Authentication**: Prevent impersonation
  - **Integrity**: Prevent modification
Where does SSL fit?

- Layered between Transport and Application layers
SSL: Authentication

- Do you really know who you’re communicating with?

Alice

Bob
SSL: Authentication

- Generate a key pair (private and public key)
SSL: Authentication

- X.509 Certificate == Wrapper around public key
-----BEGIN CERTIFICATE-----
MIIEmDCC4CqAwIBAgIJAIdKdb6R2tg9MA0GCSqGSIb3DQEBBQUAMIGOMQswCQYD
VQQGEwJVUzEPA0GA1UECMBMT3J1Z29uMREwDwYDVQQHEwQb3J0bGFuZDEOMAwG
AlUChMFeWFTU0wxFDASBgNVBAsTC1Byb2dyYW1taW5nMRYwFAYDVQQDEw13d3cu
eWzc2wuY29tM0wGWhcNMA0GCSqGSIb3DQEBCwUAA4IBADY0fCp55q+2/OvE3L1
vX1n4+pC5QaJi5R96ZG99m5Jz0d4T46pY/5sRr2jXomOUySBp2eZ0+bXo5jQu
mzuC0e75IyuczwZD+NAy3vMR+n/xexyZyEi+Xc7Sz5Lz4+oR/OO5s8jL25R7V
-----END CERTIFICATE-----
SSL: X.509 Certificates

Certificate:
Data:
Version: 3 (0x2)
Serial Number: 87:4a:75:be:91:66:d8:3d
Signature Algorithm: sha1WithRSAEncryption
Issuer: C=US, ST=Oregon, L=Portland, O=yaSSL, OU=Programming, CN=www.yassl.com/emailAddress=info@yassl.com

Validity
Not After: Jul 20 18:21:55 2014 GMT

Subject: C=US, ST=Oregon, L=Portland, O=yaSSL, OU=Programming, CN=www.yassl.com/emailAddress=info@yassl.com

Subject Public Key Info:
Public Key Algorithm: rsaEncryption
Public-Key: (2048 bit)
Modulus: 00:c3:03:d1:2b:fe:39:a4 ...
Exponent: 65537 (0x10001)

X509v3 extensions:
X509v3 Subject Key Identifier:

X509v3 Authority Key Identifier:
DirName:/C=US/ST=Oregon/L=Portland/O=yaSSL/OU=Programming/CN=www.yassl.com/emailAddress=info@yassl.com

X509v3 Basic Constraints:
CA:TRUE
Signature Algorithm: sha1WithRSAEncryption
... 1c:7c:42:81:29:9e:21:cf:d0:d8
SSL: Authentication

- Alice and Bob exchange **CA-signed** public keys
SSL: Authentication

• How do you get a CA-signed cert?

**Buy**
- VeriSign, DigiCert, Comodo, etc.
- Costs $$$
- Trusted

**Create**
- Created yourself (self-sign)
- Free!
- Trusted (if you control both sides)
SSL: Encryption

- Uses a variety of encryption algorithms to secure data

**Hashing Functions**  
MD4, MD5, SHA ...

**Block and Stream Ciphers**  
DES, 3DES, AES, ARC4 ...

**Public Key Options**  
RSA, DSS ...

CIPHER SUITE
SSL: Encryption

- A common CIPHER SUITE is negotiated

Protocol_keyexchange_WITH_bulkencryption_mode_messageauth

SSL_RSA_WITH_DES_CBC_SHA
SSL_DHE_RSA_WITH_DES_CBC_SHA
TLS_RSA_WITH_AES_128_CBC_SHA
TLS_DHE_DSS_WITH_AES_128_CBC_SHA
TLS_DHE_RSA_WITH_AES_256_CBC_SHA
SSL: Handshake

1. Client Hello
   Cryptographic Info
   (SSL version, supported ciphers, etc.)

2. Server Hello
   Cipher Suite
   Server Certificate
   Server Key Exchange (public key)
   (Client Certificate Request)
   Server Hello Done

3. Verify server cert, check crypto parameters

4. Client Key Exchange
   (Certificate Verify)
   (Client Certificate)

5. Verify client cert (if required)

6. Change Cipher Spec
   Client Finished

7. Change Cipher Spec
   Server Finished

8. Exchange Messages (Encrypted)
Where is SSL used?

Everywhere!
SSL: Where is it used?

- Energy Monitoring
- Gaming
- Databases
- Sensors
- VoIP
- M2M communication
- And much more...
What to look for?

When shopping for an SSL stack.
1: Protocols

• Support for current protocols?

Notes:

• SSL 2.0 is insecure
• SSL = “Secure Sockets Layer”
• TLS = “Transport Layer Security”
• DTLS = “Datagram TLS”
### 2: Ciphers

- Support for needed cipher suites?

<table>
<thead>
<tr>
<th>Public Key</th>
<th>Block / Stream</th>
<th>Hash</th>
</tr>
</thead>
<tbody>
<tr>
<td>RSA, DSS, DH, NTRU</td>
<td>DES, 3DES, AES, ARC4, RABBIT, HC-128</td>
<td>MD2, MD4, MD5, SHA-128, SHA-256, SHA-256, RIPEMD</td>
</tr>
<tr>
<td></td>
<td></td>
<td>...</td>
</tr>
</tbody>
</table>

Ex: **TLS_RSA_WITH_AES_128_CBC_SHA**
3: Memory Usage

- ROM / RAM usage

![Bar chart showing ROM and RAM usage](chart.png)
4: Simple to Use

- Learning curve?

- Myth: Encryption is too hard to use.
5: Portability

- OS support out-of-the-box?
- Customizable?
6: Hardware Acceleration

• Support for hardware acceleration?

• Assembly code optimizations
7: License

- Flexible license model?
- Does it meet your license needs?
8: Maturity

- Track record?
- Code origin?
- Actively developed?
9: Compatibility

• Is interoperability testing being conducted?

• What browsers is the library actively tested against?
10: Crypto Access

- Direct access to crypto?

**Many reasons:**
- Direct encryption
- Code Signing
- Verifying hashes, etc.
11: Support

• What happens if:
  
  – Something goes wrong
  – You can’t get it to work on your system
  – New vulnerability comes out
  – You need a new cipher/feature

• Is there support available to help you out?
SSL: Shopping List

1. Protocols
2. Ciphers
3. Memory Usage
4. Simple to Use
5. Portability
6. Hardware Acceleration
7. License
8. Maturity
9. Compatibility
10. Crypto Access
11. Support
Thanks!

www.yassl.com

chris@yassl.com
info@yassl.com