1. State of SSL
2. Quick intro to SSL Labs
3. SSL Configuration Surveys
4. Survey of Actual SSL Usage
5. Conclusions
Ivan is a compulsive builder, usually attracted to problems no one else is working on

- **ModSecurity**, open source web application firewall
- **SSL Labs**, SSL, TLS, and PKI research
- **IronBee**, next-generation open source web application firewall
Part I: State of SSL
Protocol goal:

- Turn an insecure communication channel, no matter which protocol it is running, into a secure one
- Designed for HTTP, but can be used for pretty much anything

The original version of the protocol designed at Netscape:

- Version 2 was released 1994
- Found to have many issues, and quickly followed by v3
- Standardized under the name TLS (Transport Layer Security) in 1999
  - TLS v1.1 released in 2006
  - TLS v1.2 released in 2008
The SSL ecosystem includes many players:

- Basic cryptographic algorithms
- SSL and TLS encryption protocols
- IETF TLS Working Group
- Public Key Infrastructure (PKI) standards
- Certificate Authorities and their resellers
- CA/Browser Forum
- SSL Client vendors (esp. major browser vendors)
- SSL library developers
- SSL server vendors
- System administrators
- Consumers
Major Challenges Today

1. Fragility of the trust ecosystem
2. Incorrect or weak configuration
3. Slow adoption of modern standards
4. Lack of support for virtual SSL hosting
5. Mismatch between HTTP and SSL
6. Performance and caching challenges
SSL can fail in many ways, but there are 3 principal attacks:

- Passive MITM
  - Session hijacking (e.g., using Firesheep)
- Active MITM
  - SSL bypass (e.g., using sslstrip)
  - Attacks against renegotiation
  - Rogue certificates
  - User attacks (who reads warnings anyway)
- Third-party compromise

(*) For a complete attack model, visit https://www.ssllabs.com/projects/ssl-threat-model/
It is possible to have a reasonably secure web site (when it comes to communication security):

- Use an EV certificate
- Configure your SSL server properly:
  - Good key size and coverage of desired domain names
  - Good protocols and 128-bit forward-secrecy cipher suites
  - Patches and workarounds applied
- Redirect all port 80 traffic to port 443
- Use HTTP Strict Transport Security
Part II:
SSL Labs
SSL Labs:

- A non-commercial security research effort focused on SSL, TLS, and friends

Projects:

- Assessment tool
- SSL Rating Guide
- Passive SSL client fingerprinting tool
- SSL Threat Model
- SSL Survey
How can SSL fail?

- In about a million and one different ways, some worse than others.

Principal issues:

- Implementation flaws
- MITM
- Usability issues
- Impedance mismatch
- Deployment mistakes
- PKI trust challenges
SSL Rating Guide

What is the purpose of the guide?

- Sum up a server’s SSL configuration, and explain how scores are assigned
- Make it possible for non-experts to understand how serious flaws are
- Enable us to quickly say if one server is better configured than another
- Give configuration guidance
Online SSL Assessment Overview

Main features:
- Free online SSL test
- Comprehensive, yet easy on CPU
- Results easy to understand

What we analyze:
- Configuration
- Certificate chain
- Protocol and cipher suite support
- Enabled Features
- Weaknesses
SSL Assessment Details

Highlights:
- Renegotiation vulnerability
- Cipher suite preference
- TLS version intolerance
- Session resumption
- Firefox 3.6 trust base

Every assessment consists of about:
- 2000 packets
- 200 connections
- 250 KB data
Part IV: SSL Configuration Surveys
Global SSL Surveys

In our first global survey, in 2010:
- We looked at 119 million domain name registrations
- Also examined the Alexa’s top 1m domain names
- Arrived to about 900,000 server to assess
- About 600,000 were valid and were used in the survey

In our second global survey, in 2011:
- We used the data from EFF’s SSL Observatory
- Almost doubled the number of valid certificates, to about 1.2m
In 2010, we looked at 119 million domain names (60% of all registrations):

- 22.66% not operational
- 48.03% does not listen on port 443
- 9.40% runs something else on port 443
- 18.40% certificate name mismatches
- 0.60% certificate name matches (and not even those are all valid)

- Virtual web hosting hugely popular
  - 119m domain names represented by about 5.3m IP addresses
  - 22.65m domain names with SSL represented by about 2m IP addresses

- Issues:
  - No virtual SSL web hosting
  - No way for a browser to know if a site uses SSL
In order to understand impedance mismatch issues, we undertook a deep survey of most popular SSL web sites:

- Start with the top 1M popular sites from Alexa
- And with 1.4m valid SSL sites globally from SSL Observatory
- Cross-reference to arrive at **327,476** SSL sites
- Accept **248,161** sites into the survey

Then:

- Build a custom crawler to visit each site from the list, and examine things such as:
  - Mixed content
  - Insecure cookies
  - Use of third-party resources (delegation of trust)
  - Response header usage
Countries Overview

Countries with over 1,000 certificates:
SSL Labs Grade Distribution

Most servers not configured well

- Only 32.37% got an A
- 67.63% got a B or worse
- Most probably just use the default settings of their web server

Key length | Score
---|---
A | >= 80
B | >= 65
C | >= 50
D | >= 35
E | >= 20
F | < 20

Score distribution

Grade distribution

- A: 96,664 (32.37%)
- B: 28,293 (9.47%)
- C: 100,387 (33.62%)
- D: 67,456 (22.59%)
- E: 7,801 (2.61%)
Certificates

Virtually all trusted certificates use **RSA** keys; **only 9 DSA** keys

- SHA1 with RSA is the most popular choice for the signature algorithm
- We are starting to see SHA256, but only on 18 certificate
- Virtually all keys 1024 or 2048 bits long
- Still 43 weak RNG keys from Debian
- About 10% incorrect certificate chains

### Key length vs. Certificates seen

<table>
<thead>
<tr>
<th>Key length</th>
<th>Certificates seen</th>
</tr>
</thead>
<tbody>
<tr>
<td>512</td>
<td>559</td>
</tr>
<tr>
<td>1024</td>
<td>170,423</td>
</tr>
<tr>
<td>2048</td>
<td>125,333</td>
</tr>
<tr>
<td>4096</td>
<td>2,108</td>
</tr>
<tr>
<td>8192</td>
<td>3</td>
</tr>
</tbody>
</table>
Half of all trusted servers support the insecure SSL v2 protocol

- Modern browsers won’t use it, but wide support for SSL v2 demonstrates how we neglect to give any attention to SSL configuration
- Virtually all servers support SSLv3 and TLS v1.0
- Virtually no support for TLS v1.1 (released in 2006) or TLS v1.2 (released in 2008)

<table>
<thead>
<tr>
<th>Protocol</th>
<th>Support</th>
<th>Best protocol</th>
</tr>
</thead>
<tbody>
<tr>
<td>SSL v2.0</td>
<td>143,591</td>
<td>110</td>
</tr>
<tr>
<td>SSL v3.0</td>
<td>298,078</td>
<td>5,205</td>
</tr>
<tr>
<td>TLS v1.0</td>
<td>293,286</td>
<td>292,366</td>
</tr>
<tr>
<td>TLS v1.1</td>
<td>916</td>
<td>854</td>
</tr>
<tr>
<td>TLS v1.2</td>
<td>69</td>
<td>69</td>
</tr>
</tbody>
</table>
Cipher Strength

All servers support **strong** and most support **very strong** ciphers

- But there is also wide support for weak ciphers
Secure and Insecure Renegotiation

Insecure renegotiation is the closest thing to a serious TLS protocol flaw so far:

- Published in November 2009
- Last major vendor patched in January 2011
- Globally:

Support for secure and insecure client-initiated renegotiation

- Secure renegotiation
  - 122,585
  - 41.05%
- Insecure renegotiation
  - 104,441
  - 34.98%
- Not supported
  - 65,881
  - 22.06%
- Both
  - 5,699
  - 1.91%

- Secure renegotiation
  - 606,456
  - 52.39%
- Insecure renegotiation
  - 298,909
  - 25.82%
- Not supported
  - 229,252
  - 19.81%
- Both
  - 22,866
  - 1.98%
Part V: Survey of Actual SSL Usage
First we wanted to know how many sites make exclusive use of SSL:

- Out of 248,161 sites tested (remember, all support SSL)
- **20.61% (51,160)** redirect to SSL

The rest, **79.29% sites**, may or may not (most likely not) redirect to SSL for authentication.:

- Sites without redirection are easily exploitable via *sslstrip* or *Firesheep*
Strict Transport Security

Next we looked at HTTP Strict Transport Security:

- Out of 248,161 sites tested
- **Only 80 use HSTS**
  - 162 globally (out of 1.2m SSL servers)

We saw 142 different HSTS responses, and looked at the *max-age* and *includeSubdomains* settings:

- Varied approaches to max-age, from short term to long term
- **13 out of 142 use HSTS to include subdomains**
  - These are safe from cookie forcing attacks

![Bar chart showing max-age settings for HSTS](chart.png)
Proper deployment of HSTS requires a redirection, so we cross-references the list of sites that support HSTS with the list of sites that have redirection in place:

- Out of 51,160 sites with redirection
  - **Only 55 use HSTS**

The final piece here is the EV certificate:

- Out of 55 sites with HSTS and redirection
  - **Only 9 have an EV certificate**

Thus:

- Out of 248,161 sites tested
  - **Only 9 have state of the art protection**

- Actually, it’s 0 if you consider `includeSubdomains` important
In most web applications, cookies are used for authentication for the duration of the session:

- Out of 248,161 sites tested
- We saw **36.80% (91,335)** sites with session cookies
- **16,530 HttpOnly**
- **14,506 Secure**
- **1,706 HttpOnly and Secure**
Mixed Content

When it comes to mixed content, we wanted an indication of how many sites are suffering from this problem:

- Out of 248,161 sites tested
- **22.41% (55,628)** use mixed content
- **18.71% (46,434)** use mixed content, excluding images
27.4% (68,020) include services of other web sites, and thus rely on other sites’ security:

- Most of these have one or two links
- A small number uses many (up to 22)
- The usual suspects:
  - Google Analytics
  - Google Ads
  - Quantcast
  - Twitter
  - Google jQuery hosting
  - Facebook
  - And a long tail…
Authentication

You would expect that most sites understand the need to protect user credentials:

- **25.91%** (64,321) sites have a login form
- But **68.96%** (44,361) over HTTP
- And **54.39%** (34,990) submit over HTTP too
- Less than half of forms is protected using SSL
Declarative protection measures are very effective because they can often be implemented in configuration, and after the fact:

- Out of **248,161** sites tested

<table>
<thead>
<tr>
<th>Measure</th>
<th>Sites</th>
<th>Popularity</th>
</tr>
</thead>
<tbody>
<tr>
<td>HttpOnly</td>
<td>16,530</td>
<td>6.66%</td>
</tr>
<tr>
<td>Secure</td>
<td>14,506</td>
<td>5.84%</td>
</tr>
<tr>
<td>X-Frame-Options</td>
<td>686</td>
<td>0.27%</td>
</tr>
<tr>
<td>X-XSS-Protection</td>
<td>200</td>
<td>0.080%</td>
</tr>
<tr>
<td>Strict-Transport-Security</td>
<td>80</td>
<td>0.032%</td>
</tr>
<tr>
<td>X-Content-Type-Options</td>
<td>67</td>
<td>0.027%</td>
</tr>
<tr>
<td>Access-Control-Allow-Origin</td>
<td>47</td>
<td>0.019%</td>
</tr>
<tr>
<td>X-Content-Security-Policy</td>
<td>12</td>
<td>0.005%</td>
</tr>
</tbody>
</table>
Part VI: Conclusions
Conclusions

We conclude:

1. Systemic issues are hotly debated by the community and the press
2. In real life, however, it’s deployment and implementation issues that break SSL
3. It’s possible to achieve reasonable security, but most sites choose not to do it
4. Among the popular sites, only a handful have decent SSL deployments, when all is taken into account
Thank You

Ivan Ristic
iristic@qualys.com
@ivanristic