Internet SSL Survey 2010
Black Hat USA 2010

Ivan Ristic
Director of Engineering,
Web Application Firewall and SSL
iristic@qualys.com / @ivanristic

July 19th, 2010 (v1.0)
Agenda

1. Why do we care about SSL?
2. Our SSL assessment engine
3. How does one find SSL-enabled servers to study?
4. Findings of our large-scale study of SSL servers on the Internet
5. Conclusions and future direction
Part I

Why do we care about SSL?
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, actually.

Principal issues:
- Implementation flaws
- MITM
- Usability issues
- Impedance mismatch
- Deployment mistakes
- PKI trust challenges
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
And what is NOT the purpose of the guide?

- The scores are not supposed to be a perfect representation of configuration “quality”
- We don’t know what “secure” means to you
- Besides, security has many enemies:
  - Cost
  - Performance
  - Interoperability
Part II

SSL Assessment Engine
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
Assessment Challenges

Comprehensive assessments are difficult:

- A naïve approach is to open a connection per cipher suite. *But it doesn’t scale.*
- We went to packet level, using partial connections (*with as little crypto as possible*) to extract the information we needed. **Almost no CPU used!**
- Not reliable with multiple servers behind one IP address

Other issues:

- **Complicated topic** – so many RFCs and other documents to read before you can begin to grasp the problem. *It took us ages to just assemble the list of known cipher suites.*
- **Poor programming documentation**; SSL toolkits generally designed to connect (or not), but not for diagnostics.
- **Feature coverage** – toolkits cover only a part of what the protocols can do.
- Bugs, edge cases, and interoperability issues.
Part III
Finding Servers to Scan
Finding servers to assess

We have the assessment engine sizzling, but how do we find servers to assess?

- Scan all IPv4 space
- Crawl the Internet
- Start with domain registrations
- Use a browser toolbar
- Wait for SSL Labs to become popular, recording all site names in the meantime

Are we looking for domain names, servers, or certificates?

- TLS SNI allows multiple certificates per IP address
- One domain name may have many servers / IP addresses
- There may be many servers behind one IP address
- The same certificate (esp. a wildcard one) can be used with many servers
Our approach: domain enumeration

How many domain names and certificates are there?

- 193M domain name registrations in total (VeriSign)
- 207M sites (Netcraft)
- 1.2M valid SSL certificates (Netcraft)

Main data set: domain name registrations

- All .com, .net, .org, .biz, .us, and .info domain names
- 119M domain names (57% of the total)

Bonus data sets:

- Alexa’s top 1m popular sites
- Collect the names in the certificates we find
First pass: lightweight scan

The purpose of the first-pass lightweight scan is to locate the servers we need to examine in depth:

- Those are servers with certificates whose names match the domain names on which they reside.
- Someone made an effort to match the names, therefore the intent is there!

How did we do that?

- Single server with 4 GB RAM (not a particularly powerful one)
- DNS resolution + few packets to probe ports 80 and 443 // Yes, HTTP servers only
- Naturally, incomplete SSL handshakes
- 2,000 concurrent threads
- Resulted in roughly 1,000 probes per second; fast enough
- **A day and a half for the entire scan**
Out of 119m domain names:

- 12.4M (10.37%) failed to resolve
- 14.6M (12.28%) failed to respond
- 92M (77.35%) seemed active

Active means to respond on port 80 or port 443
Port 80 and 443 activity analysis

Domain responses on ports 80 and 443

- Port 80: 91.65M (99.35%)
- Port 443: 33.69M (36.52%)

Protocols on port 443 (in millions)

- SSL: 22.65M (67.27%)
- Other: 11.02M (32.73%)
- Includes 18,222 SSH responses; the rest is mostly plaintext HTTP
- Includes 6,320 SSLv2-only responses
Out of 22.65M domain names with SSL enabled

- Name match: 0.72 (3.17%)
- No match: 21.93 (96.83%)

Alexa’s Top 1M domain names

- Name match: 0.12 (27.86%)
- No match: 0.30 (72.14%)

~720,000 potentially valid SSL certificates
22m invalid certificates! Really!?

Why so many invalid responses?

- 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
- Virtual SSL web hosting practically impossible – the majority of browsers do not support the TLS SNI extension.

We don’t know if a site uses SSL, and end up seeing something else because most don’t
The end result...

Let’s now try to get as many entries as possible

- Add all we have together:
  - 720,000 certificates from the domain name registration data set
  - 120,000 certificates from the Top 1m data set
  - About new 100,000 domains found in certificate names
- Remove duplicates:
  - Unique IP address
  - Unique domain name
  - Unique certificate
- We ended up with 867,361 entries
- Probably 25-50% of all commercial certs
How many certs failed validation and why?

- **Trusted**: 607,589 (70.05%)
- **Not trusted**: 239,007 (27.56%)
- Not trusted suspicious: 20,765 (2.39%)

32,642 (3.76%) have incomplete chains.

Validation failures:
- Expired: 136,534
- Self-signed: 96,321
- Unknown CA: 56,864
- Invalid signature: 20,765
- Revoked: 1,072
- Bad CN: 903

Interoperability issues with JSSE?

Remember that the methodology excluded hostname mismatch problems.
Certificate validity and expiry distribution

Certificate period of validity (trusted certificates only)

Expired and other problems 52,190 (38%)

Expired only 83,925 (62%)

How many certificates are only expired, and how many have other problems too?
Unknown issuers

We saw 56,864 unknown issuers

- Great majority of issuers seen only once
- 22 seen in more than 100 certificates
- Manually verified those 22
- Found 4 that one could argue are legitimate, but are not trusted by Mozilla (yet) ([http://www.mozilla.org/projects/security/certs/pending/](http://www.mozilla.org/projects/security/certs/pending/))

<table>
<thead>
<tr>
<th>Issuer</th>
<th>Seen certificates</th>
</tr>
</thead>
<tbody>
<tr>
<td>Firstserver Encryption Services</td>
<td>9486</td>
</tr>
<tr>
<td>CAcert</td>
<td>6117</td>
</tr>
<tr>
<td>ipsCA</td>
<td>462</td>
</tr>
<tr>
<td>KISA Root CA</td>
<td>162</td>
</tr>
</tbody>
</table>
We saw 429 ultimately-trusted certificate issuers

- They led to 78 trust anchors
- That’s only 50% of our trust base, which has 155 trust anchors

This path is 2 levels deep in 44% of cases, and 3 levels deep in 55% of cases.

<table>
<thead>
<tr>
<th>Chain length</th>
<th>Certificates seen</th>
</tr>
</thead>
<tbody>
<tr>
<td>2</td>
<td>270,779</td>
</tr>
<tr>
<td>3</td>
<td>334,248</td>
</tr>
<tr>
<td>4</td>
<td>2368</td>
</tr>
<tr>
<td>5</td>
<td>186</td>
</tr>
<tr>
<td>6</td>
<td>8</td>
</tr>
</tbody>
</table>
Certificate chain correctness

Correct versus incorrect certificate chains

Correct 569,472 93.73%
Incorrect 38,117 6.27%

Potential performance and bandwidth issue
However, some of the extra certificates may be needed by some clients; needs further verification

Issues with certificate chains

Unneeded certificates sent 265,238 43.65%
Incomplete chain 32,642 9.69%
Incorrect order 5,475 1.62%

Could invalidate chains, depending on client
Certificate chain size and length

In **43.65%** of all cases, there’s more certificates sent than needed

- When latency between client and server is high, the unneeded certificates waste the precious initial bandwidth
- Important when you need to want the performance to be as good as possible

### Certs sent | Actual | Should be
--- | --- | ---
1 | 227,520 | 270,779
2 | 181,996 | 334,248
3 | 113,672 | 2,368
4 | 78,931 | 186
5 | 3,320 | 8
6 | 1,491 | 0
7 | 48 | 0
8 | 28 | 0
9 | 49 | 0
10 | 489 | 0
11 | 4 | 0
12 | 10 | 0
13 | 24 | 0
15 | 1 | 0
16 | 1 | 0
17 | 2 | 0
61 | 1 | 0
70 | 1 | 0
116 | 1 | 0
Trust anchors

### Certificates per issuer

<table>
<thead>
<tr>
<th>Trust Anchor</th>
<th>Certificates</th>
</tr>
</thead>
<tbody>
<tr>
<td>Go Daddy Class 2 Certification Authority</td>
<td>146,173</td>
</tr>
<tr>
<td>Equifax Secure Certificate Authority</td>
<td>141,210</td>
</tr>
<tr>
<td>UTN-USERFirst-Hardware</td>
<td>86,868</td>
</tr>
<tr>
<td>Thawte Premium Server CA</td>
<td>27,976</td>
</tr>
<tr>
<td>Thawte Server CA</td>
<td>26,972</td>
</tr>
<tr>
<td>Class 3 Public Primary Certification Authority</td>
<td>26,765</td>
</tr>
<tr>
<td>VeriSign Trust Network</td>
<td>26,163</td>
</tr>
<tr>
<td>GlobalSign Root CA</td>
<td>20,290</td>
</tr>
<tr>
<td>Starfield Class 2 Certification Authority</td>
<td>17,824</td>
</tr>
<tr>
<td>Equifax Secure Global eBusiness CA-1</td>
<td>15,662</td>
</tr>
<tr>
<td>COMODO Certification Authority</td>
<td>14,296</td>
</tr>
<tr>
<td>SecureTrust CA</td>
<td>8,793</td>
</tr>
<tr>
<td>VeriSign Class 3 Public Primary Certification Authority - G5</td>
<td>7,619</td>
</tr>
<tr>
<td>DigiCert High Assurance EV Root CA</td>
<td>6,769</td>
</tr>
<tr>
<td>StartCom Certification Authority</td>
<td>6,197</td>
</tr>
<tr>
<td>Entrust.net Secure Server Certification Authority</td>
<td>5,068</td>
</tr>
<tr>
<td>GTE CyberTrust Global Root</td>
<td>4,659</td>
</tr>
</tbody>
</table>

17 trust anchors on this page account for 589,304 (97%) certificates
On average, there will be **5.5** issuers for every trust anchor.

- Top 6 anchors have more than 10 issuers each
- They account for a total of 286 issuers, or 67% of all
- Deutsche Telekom alone accounts for 39% of all issuers we saw
How many trust anchors do we need?

Let’s try to figure the minimum number of trust anchors!

- Of course, this is very subjective
- Our data set is biased and contains predominantly U.S. web sites
- Your browsing habits are probably different
- Still, it’s interesting to see that you probably need only between 10 and 20 trust anchors.
- But your selection may be different from mine!
Virtually all trusted certificates use **RSA** keys; only 3 **DSA** keys

- 127 DSA keys across all certificates (i.e., including those certs we could not validate)
- SHA1 with RSA is the most popular choice for the signature algorithm
- A very small number of stronger hash functions seen across all certificates:
  - SHA256 with RSA: 190
  - SHA385 with RSA: 1
  - SHA512 with RSA: 75
- Virtually all keys 1024 or 2048 bits long
- Only 99 weak RNG keys from Debian (but 3,938 more among the untrusted)
- Only 8% servers support server-gated crypto

### Signature algorithm

- **SHA1** with RSA: 597,404 (98.32%)
- **MD5** with RSA: 10,185 (1.68%)

### Key length

<table>
<thead>
<tr>
<th>Key length</th>
<th>Certificates seen</th>
</tr>
</thead>
<tbody>
<tr>
<td>512</td>
<td>3,005</td>
</tr>
<tr>
<td>1024</td>
<td>386,694</td>
</tr>
<tr>
<td>2048</td>
<td>211,155</td>
</tr>
<tr>
<td>4096</td>
<td>6,315</td>
</tr>
<tr>
<td>8192</td>
<td>14</td>
</tr>
<tr>
<td>Other</td>
<td>406</td>
</tr>
</tbody>
</table>
Support for multiple domain names

Most sites support 0, 1, or 2 alternative domain names

- Some CAs will automatically add 2 alternative domain names (“example.com” and www.example.com)
- Untrusted 3o.hu has 354 (8.2 KB cert)!
- Untrusted www.epi.es has 287 and they are all wildcards (7.5 KB cert)!

About 4.44% certificates use wildcards

- 2.72% as the common name
- 1.72% in the alternative name

About 35.59% certificates support access with and without the “www” part.

- 88% of the domains tested are under a TLD

<table>
<thead>
<tr>
<th>Alternative names</th>
<th>Name</th>
</tr>
</thead>
<tbody>
<tr>
<td>252</td>
<td><a href="http://www.hu-berlin.de">www.hu-berlin.de</a></td>
</tr>
<tr>
<td>191</td>
<td><a href="http://www.tu-berlin.de">www.tu-berlin.de</a></td>
</tr>
<tr>
<td>153</td>
<td>*.abyx.com</td>
</tr>
<tr>
<td>150</td>
<td><a href="http://www.newcreditera.com">www.newcreditera.com</a></td>
</tr>
<tr>
<td>116</td>
<td>edgecastcdn.net</td>
</tr>
<tr>
<td>101</td>
<td>jpbsecurehostingservice.com</td>
</tr>
<tr>
<td></td>
<td><a href="http://www.indiebound.org">www.indiebound.org</a></td>
</tr>
<tr>
<td>100</td>
<td>quotes.usinsuranceonline.com</td>
</tr>
</tbody>
</table>
Half of all trusted servers support the insecure SSL v2 protocol

- Modern browsers won’t talk 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)
- At least 10,462 servers will accept SSLv2 but only deliver a user-friendly error message over HTTP

<table>
<thead>
<tr>
<th>Protocol</th>
<th>Support</th>
<th>Best protocol</th>
</tr>
</thead>
<tbody>
<tr>
<td>SSL v2.0</td>
<td>302,886</td>
<td>-</td>
</tr>
<tr>
<td>SSL v3.0</td>
<td>607,249</td>
<td>3,249</td>
</tr>
<tr>
<td>TLS v1.0</td>
<td>604,242</td>
<td>603,404</td>
</tr>
<tr>
<td>TLS v1.1</td>
<td>838</td>
<td>827</td>
</tr>
<tr>
<td>TLS v1.2</td>
<td>11</td>
<td>11</td>
</tr>
</tbody>
</table>
Ciphers, key exchange and hash functions

**Triple DES and RC4 rule in the cipher space**

- There is also good support for AES, DES and RC2

<table>
<thead>
<tr>
<th>Key exchange</th>
<th>Servers</th>
<th>Percentage</th>
</tr>
</thead>
<tbody>
<tr>
<td>RSA</td>
<td>607,582</td>
<td>99.99%</td>
</tr>
<tr>
<td>DHE_RSA</td>
<td>348,557</td>
<td>57.36%</td>
</tr>
<tr>
<td>RSA_EXPORT</td>
<td>319,826</td>
<td>52.63%</td>
</tr>
<tr>
<td>RSA_EXPORT_1024</td>
<td>193,793</td>
<td>31.89%</td>
</tr>
<tr>
<td>DHE_RSA_EXPORT</td>
<td>176,258</td>
<td>29.00%</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Cipher</th>
<th>Servers</th>
<th>Percentage</th>
</tr>
</thead>
<tbody>
<tr>
<td>3DES_EDE_CBC</td>
<td>603,888</td>
<td>99.39%</td>
</tr>
<tr>
<td>RC4_128</td>
<td>596,363</td>
<td>98.15%</td>
</tr>
<tr>
<td>AES_128_CBC</td>
<td>418,095</td>
<td>68.81%</td>
</tr>
<tr>
<td>AES_256_CBC</td>
<td>415,585</td>
<td>68.39%</td>
</tr>
<tr>
<td>DES_CBC</td>
<td>341,145</td>
<td>56.14%</td>
</tr>
<tr>
<td>RC4_40</td>
<td>320,689</td>
<td>52.78%</td>
</tr>
<tr>
<td>RC2_CBC_40</td>
<td>314,689</td>
<td>51.79%</td>
</tr>
<tr>
<td>RC2_128_CBC</td>
<td>283,416</td>
<td>46.64%</td>
</tr>
<tr>
<td>DES_CBC_40</td>
<td>192,558</td>
<td>31.69%</td>
</tr>
<tr>
<td>RC4_56</td>
<td>192,192</td>
<td>31.63%</td>
</tr>
<tr>
<td>IDEA_CBC</td>
<td>52,762</td>
<td>8.68%</td>
</tr>
<tr>
<td>RC2_CBC_56</td>
<td>50,897</td>
<td>8.37%</td>
</tr>
<tr>
<td>CAMELLIA_256_CBC</td>
<td>29,709</td>
<td>4.88%</td>
</tr>
<tr>
<td>CAMELLIA_128_CBC</td>
<td>29,708</td>
<td>4.88%</td>
</tr>
<tr>
<td>SEED_CBC</td>
<td>14,796</td>
<td>2.43%</td>
</tr>
<tr>
<td>NULL</td>
<td>2,185</td>
<td>0.35%</td>
</tr>
<tr>
<td>AES_128_GCM</td>
<td>2</td>
<td>-</td>
</tr>
<tr>
<td>AES_256_GCM</td>
<td>1</td>
<td>-</td>
</tr>
<tr>
<td>FORTEZZA_CBC</td>
<td>1</td>
<td>-</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Hash</th>
<th>Servers</th>
<th>Percentage</th>
</tr>
</thead>
<tbody>
<tr>
<td>SHA</td>
<td>606,489</td>
<td>99.81%</td>
</tr>
<tr>
<td>MD5</td>
<td>591,433</td>
<td>97.34%</td>
</tr>
<tr>
<td>SHA256</td>
<td>4</td>
<td>-</td>
</tr>
<tr>
<td>SHA384</td>
<td>156</td>
<td>-</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

![Cipher strength support chart]

**Best cipher strength support**

**Cipher strength support**
Cipher suite support

Most supported cipher suites

<table>
<thead>
<tr>
<th>Cipher suites</th>
<th>Servers</th>
<th>Percentage</th>
</tr>
</thead>
<tbody>
<tr>
<td>TLS_RSA_WITH_3DES_EDE_CBC_SHA</td>
<td>603,545</td>
<td>99.33%</td>
</tr>
<tr>
<td>TLS_RSA_WITH_RC4_128_SHA</td>
<td>593,884</td>
<td>97.74%</td>
</tr>
<tr>
<td>TLS_RSA_WITH_RC4_128_MD5</td>
<td>590,901</td>
<td>97.25%</td>
</tr>
<tr>
<td>TLS_RSA_WITH_AES_128_CBC_SHA</td>
<td>417,866</td>
<td>68.77%</td>
</tr>
<tr>
<td>TLS_RSA_WITH_AES_256_CBC_SHA</td>
<td>415,348</td>
<td>68.36%</td>
</tr>
<tr>
<td>TLS_DHE_RSA_WITH_3DES_EDE_CBC_SHA</td>
<td>347,729</td>
<td>57.23%</td>
</tr>
</tbody>
</table>

Most preferred cipher suites

<table>
<thead>
<tr>
<th>Cipher suite</th>
</tr>
</thead>
<tbody>
<tr>
<td>TLS_RSA_WITH_RC4_128_MD5</td>
</tr>
<tr>
<td>TLS_RSA_WITH_RC4_128_SHA</td>
</tr>
<tr>
<td>TLS_RSA_EXPORT1024_WITH_RC4_56_SHA</td>
</tr>
<tr>
<td>TLS_RSA_EXPORT1024_WITH_DES_CBC_SHA</td>
</tr>
<tr>
<td>TLS_RSA_EXPORT1024_WITH_AES_128_CBC_SHA</td>
</tr>
<tr>
<td>TLS_RSA_EXPORT1024_WITH_AES_256_CBC_SHA</td>
</tr>
</tbody>
</table>

Cipher suite server preference

- No preference: 367,758 (60.53%)
- Server preference: 239,831 (39.47%)
SSL Labs grade distribution

<table>
<thead>
<tr>
<th>Key length</th>
<th>Score</th>
</tr>
</thead>
<tbody>
<tr>
<td>A</td>
<td>&gt;= 80</td>
</tr>
<tr>
<td>B</td>
<td>&gt;= 65</td>
</tr>
<tr>
<td>C</td>
<td>&gt;= 50</td>
</tr>
<tr>
<td>D</td>
<td>&gt;= 35</td>
</tr>
<tr>
<td>E</td>
<td>&gt;= 20</td>
</tr>
<tr>
<td>F</td>
<td>&lt; 20</td>
</tr>
</tbody>
</table>

- A: 234,201
- B: 205,444
- C: 117,225
- D: 2
- E: 5,274
- F: 45,443

Thousands of grades by score. The chart shows the distribution of grades by key length, with different color bars representing each grade category.
Strict Transport Security (STS)

Only 12 trusted sites seem to support Strict Transport Security (STS)

- Supported by further 3 untrusted sites
- STS allows sites to say that they do not want plain-text traffic
- Just send a Strict-Transport-Security response header from the SSL portion of the site
- Supported in Chrome and Firefox with NoScript
- Internet draft

Sites that support STS

- secure.grepular.com
- secure.informaction.com
- www.acdet.com
- www.datamerica.com
- www.defcon.org
- www.elanex.biz
- www.feistyduck.com
- www.paypal.com
- www.squareup.com
- www.sslabs.com
- www.strongspace.com
- www.voips scanner.com
Secure and insecure renegotiation

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

- Became public in November 2009
- Initial response was to disable renegotiation
- But not all sites can do that
- Some vendors have started to support it
- We are seeing servers patched at about 4% per month
- There are 68 sites that support insecure and secure renegotiation at the same time
Part V
What Next?
Possible future improvements, part 1

Fix small assessment engine issues:
- JSSE interoperability issue
- Inability to assess SSLv2-only servers and some other edge cases

Improve process:
- Automate assessment
- Automate report generation

Assessment improvements:
- Deeper look into protocols (e.g., SNI, compression, exotic extensions)
- Deeper look into chain failures (e.g., expired intermediate certificates)
- Improve detection of error pages that are used with weak protocols and suites
- SSL server fingerprinting
Possible future improvements, part 2

Should we try to find all servers and certificates?
- It’s very time consuming
- Would finding all of them substantially add to our knowledge?

Or, should we scale down and add more depth instead?
- Expand into protocols other than HTTP
- Insecure cookie usage
- Same-page mixed content
- Sites that mix HTTP and HTTPS
That’s all for today

Thank you for being here today

Do you have any questions?