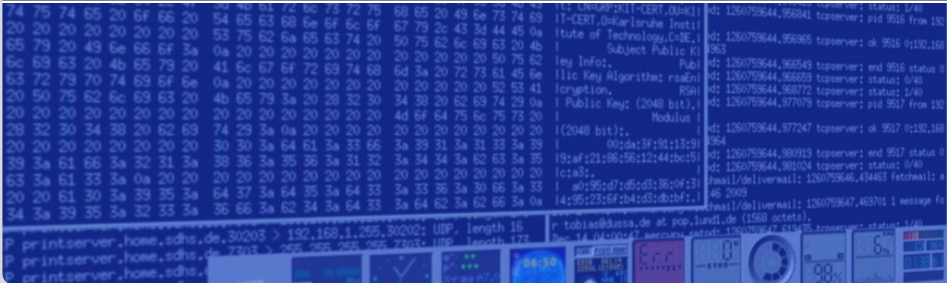


BetterCrypto.org – Applied Crypto Hardening

Tobias Dussa • 60. DFN-Betriebstagung

COMPUTER EMERGENCY RESPONSE TEAM



Einleitung



The NSA
*The only part of government
that actually listens.*

- Sich nicht ohne Not unter Wert verkaufen,
- Angreifern das Leben so schwer wie (sinnvoll) möglich zu machen,
- auch an nicht ganz so gut ausgestattete Angreifer denken!

- **Webseite des Projekts:**

`https://www.bettercrypto.org`

- **Aktueller Draft des Dokuments:**

`https://bettercrypto.org/static/applied-crypto-hardening.pdf`

- **Twitter/App.net:**

`@bettercrypto`

- **Git-Repo:**

`https://git.bettercrypto.org`

oder bei GitHub:

`https://github.com/BetterCrypto/`

- **Mailingliste/-Archiv:**

`https://lists.cert.at/cgi-bin/mailman/listinfo/ach`

- Offene Gruppe von Beitragenden.
- Kein per se gewollter Schwerpunkt; aus »historischen« Gründen derzeit mit Masse europäisch.
- Initial Mitwirkende: Wolfgang Breyha (Uni Wien), David Durveaux (CERT.be), Tobias Dussa (KIT-CERT), L. Aaron Kaplan (CERT.at), Christian Mock (coretec), Daniel Kovacic (A-Trust), Manuel Koschuch (FH Campus Wien), Adi Kriegisch (VRVis), Ramin Sabet (A-Trust), Aaron Zauner (azet.org), Pepi Zawodsky (maclemmon.at).

- Ein bisschen zur Sicherheit im Internet beitragen.
- Kryptographie- und Sicherheitseinstellungen in den am weitesten verbreiteten Diensten prüfen und verifizieren:
 - Webserver (Apache, Nginx, Lighttpd, ...)
 - IMAP-/POP-Server (Dovecot, Cyrus, ...)
 - OpenSSL allgemein
 - was sonst noch sinnvoll erscheint
- Für möglichst viele Dienste sinnvolle Konfigurationsschnipsel bereitstellen, die Administratoren einfach kopieren können.
- Die erarbeiteten Konfigurationsvorschläge von möglichst vielen unabhängig voneinander »begutachten« lassen.

Methodologie

- **Sämtliche Diskussion ist offen.**
 - Grundsätzliche Diskussion findet auf der Mailingliste statt; kann frei subskribiert werden, die Archive sind öffentlich.
 - Von Zeit zu Zeit finden Face-to-face-Besprechungen statt, an denen auch per Telekonferenz teilgenommen werden kann.
- **Die Ergebnisse der Diskussionen werden verteilt in ein Whitepaper eingepflegt.**
 - Das Whitepaper ist von Stunde Null an via Git in Form seiner LaTeX-Sourcen (sowie einer Reihe anderer Ressourcen) öffentlich verfügbar.
 - Insbesondere sind damit sämtliche Änderungen transparent und leicht nachvollziehbar.
- **Das Projekt zielt darauf ab, Empfehlungen und Best Practices nicht auf Steintafeln vom Berg Sinai zu tragen, sondern nachvollziehbar und begründet zu liefern.**

- Je häufiger das Whitepaper beziehungsweise die Empfehlungen reviewed werden, desto besser.
- Das Projekt ist daher auf Mitarbeit insbesondere in Form von Durchsichten angewiesen.

- Das Whitepaper enthält derzeit Diskussionen beziehungsweise Empfehlungen zu den folgenden Bereichen:
 - allgemeine kryptographische Aspekte,
 - zu verwendende/zu meidende Chiffren,
 - Schlüssellängen,
 - (Pseudo-)Zufallszahlgeneratoren.
- Grundsätzlich sind im Zweifel zwei Varianten von Empfehlungen enthalten:
 - Variante A: Stärkere Verfahren, dafür aber weniger unterstützte Clientsysteme.
 - Variante B: Etwas schwächere Verfahren, dafür aber umfassenderer Clientsupport.

Kryptographie

- Aktuell andauernde Debatte.
- Vertrauenswürdigkeit ist unklar.
- Das grundsätzliche Verfahren (die Mathematik) scheint sicher zu sein,

- ABER der NIST-Standard schreibt ohne weitere Begründung als Hash-Seed

c49d3608 86e70493 6a6678e1 139d26b7 819f7e90

vor. Begründung: Optimierung von Rechenzeit.

- Es gibt ernstzunehmende Hinweise darauf, dass dieser Seed nicht zufällig vom Himmel gefallen ist.
- Prinzipiell sind auch andere Hash-Seeds möglich, aber die meisten Implementierungen verwenden den NIST-Standard.

- Schlüssellängen müssen sinnvoll aufeinander und auf die verwendeten Algorithmen abgestimmt sein.
- RSA mit 4096-Bit-Schlüsseln ist vergebene Liebesmühe, wenn mit DES und 56-Bit-Schlüsseln kombiniert.
- Schlüssellängen können schön auf dieser Webseite verglichen werden: <http://www.keylength.com>
- Aktuell sinnvoll erscheinend:
 - RSA: ≥ 3248 Bits (ECRYPT II)
 - ECC: ≥ 256 Bits
 - AES: ≥ 128 Bits
 - SHA: SHA2+ (SHA256 und besser)

Choose a Method

Lenstra and Verheul Equations (2000)
 Lenstra Updated Equations (2004)
 ECRYPT II Recommendations (2012)
 NIST Recommendations (2012)
 ANSSI Recommendations (2010)
 Fact Sheet NSA Suite B Cryptography (2013)
 Network Working Group RFC3766 (2004)
 BSI Recommendations (2014)

Compare all Methods

1 Reference for the comparison

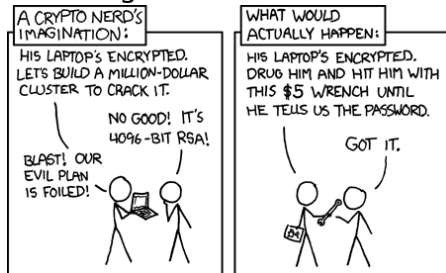
You can enter the year until when your system should be protected and see the corresponding key sizes or you can enter a key/hash/group size and see until when you would be protected.

bits

2 Compare

Method	Date	Symmetric	Asymmetric	Discrete Key	Logarithm Group	Elliptic Curve	Hash
[1] Lenstra / Verheul	2004	135	7813	9815	241	7813	269
[2] Lenstra Updated	2090	128	4440	9274	256	4440	256
[3] ECRYPT II	2031 - 2040	128	3248	256	3248	256	256
[4] NIST	> 2030	128	3072	256	3072	256	256
[5] ANSSI	> 2020	128	4096	200	4096	256	256
[6] NSA	-	128	-	-	-	256	256
[7] RFC3766	-	136	3707	272	3707	257	-
[8] BSI (signature only)	> 2020	-	1978	256	2048	250	256

- Ohne PFS kann bei Brechen eines Schlüssels sämtlicher Chiffretext auch früherer und zukünftiger Sessions mitgelesen werden.
- Schlecht, weil ein Angreifer (bekanntermaßen mindestens einschlägige TLAs) allen Verkehr mitschneiden.
- Nur eine Frage der Zeit, bis ein Schlüssel gebrochen wird:



- PFS wird erreicht beispielsweise mit DHE (Diffie-Hellman **Ephemeral**).
- Damit wird pro Schlüsselaustausch ein neuer Schlüssel gewürfelt.

Pseudozufallszahlengeneratoren

- PRNGs sind ein kritischer Punkt und werden nicht selten schlecht umgesetzt:

	Our TLS Scan		Our SSH Scans	
Number of live hosts	12,828,613	(100.00%)	10,216,363	(100.00%)
...using repeated keys	7,770,232	(60.50%)	6,642,222	(65.00%)
...using vulnerable repeated keys	714,243	(5.57%)	981,166	(9.60%)
...using default certificates or default keys	670,391	(5.23%)		
...using low-entropy repeated keys	43,852	(0.34%)		
...using RSA keys we could factor	64,081	(0.50%)	2,459	(0.03%)
...using DSA keys we could compromise			105,728	(1.03%)
...using Debian weak keys	4,147	(0.03%)	53,141	(0.52%)
...using 512-bit RSA keys	123,038	(0.96%)	8,459	(0.08%)
...identified as a vulnerable device model	985,031	(7.68%)	1,070,522	(10.48%)
...model using low-entropy repeated keys	314,640	(2.45%)		

- Hardware-RNG (Intel) vertrauenswürdig? Im Zweifel Systementropie »addieren« (kann nur besser werden).
- Außerdem klassisches Problem: Woher Entropie nehmen?
 - Embedded Devices direkt nach dem Systemstart?
 - Virtuelle Maschinen?

Cipher Suites

- SSL 2.0 ist eine GANZ schlechte Idee.
- SSL 3.0 ist auch keine besonders gute Idee.
- TLS 1.0 und besser ist akzeptabel.
- TLS Compression ist angreifbar und sollte ausgeschaltet werden.
- HTTP Strict Transport Security (HSTS) sollte verwendet werden.
- Erinnerung: Das Whitepaper unterscheidet starke (A) und schwache (B) Varianten.
- Das Zusammenstellen von Cipher Suites ist keine triviale Aufgabe, sondern ein multidimensionales Optimierungsproblem mit (mindestens) den folgenden Parametern:
 - Kompatibilität von Clients und Servern,
 - bekannte mehr oder weniger kritische Schwachstellen von Algorithmen,
 - Verfügbarkeit von (hinreichend neuen) SSL-Bibliotheken (ist Selberbauen von Serversoftware eine akzeptable Alternative?).

EECDH+aRSA+AES256:EDH+aRSA+AES256:!SSLv3

ID	OpenSSL Name	Version	KeyEx	Auth	Cipher	Hash
0xC030	ECDHE-RSA-AES256-GCM-SHA384	TLSv1.2	ECDH	RSA	AESGCM(256)	AEAD
0xC028	ECDHE-RSA-AES256-SHA384	TLSv1.2	ECDH	RSA	AES(256)	SHA384
0x009F	DHE-RSA-AES256-GCM-SHA384	TLSv1.2	DH	RSA	AESGCM(256)	AEAD
0x006B	DHE-RSA-AES256-SHA256	TLSv1.2	DH	RSA	AES(256)	SHA256

Kompatibilität: Nur Clients, die TLS 1.2 unterstützen, können mit dieser Cipher Suite umgehen. Aktuell sind das etwa Chrome 30, Windows 7 und Windows 8.1, Opera 17, OpenSSL 1.0.1e, Safari 6/iOS 6.0.1, Safari 7/OS X 10.9.

```
' ECDH+aRSA+AESGCM : ECDH+aRSA+SHA384 : ECDH+aRSA+SHA256 : EDH+CAMELLIA256 : ECDH :  
  EDH+aRSA :+SSLv3 :!aNULL :!eNULL :!LOW :!3DES :!MD5 :!EXP :!PSK :!SRP :!DSS :!RC4 :!SEED  
  :!AES128 :!CAMELLIA128 :!ECDSA :AES256-SHA '
```

ID	OpenSSL Name	Version	KeyEx	Auth	Cipher	Hash
0xC030	ECDHE-RSA-AES256-GCM-SHA384	TLSv1.2	ECDH	RSA	AESGCM(256)	AEAD
0xC028	ECDHE-RSA-AES256-SHA384	TLSv1.2	ECDH	RSA	AES(256)	SHA384
0x009F	DHE-RSA-AES256-GCM-SHA384	TLSv1.2	DH	RSA	AESGCM(256)	AEAD
0x006B	DHE-RSA-AES256-SHA256	TLSv1.2	DH	RSA	AES(256)	SHA256
0x0088	DHE-RSA-CAMELLIA256-SHA	SSLv3	DH	RSA	Camellia(256)	SHA1
0xC014	ECDHE-RSA-AES256-SHA	SSLv3	ECDH	RSA	AES(256)	SHA1
0x0039	DHE-RSA-AES256-SHA	SSLv3	DH	RSA	AES(256)	SHA1
0x0035	AES256-SHA	SSLv3	RSA	RSA	AES(256)	SHA1

Viel breitere Kompatibilitätsbasis, enthält aber schwächere Algorithmen.

Beispiel für Variante B – Kompatibilität



Handshake Simulation

Bing Oct 2013	TLS 1.0	TLS_DHE_RSA_WITH_AES_256_CBC_SHA (0xc39)	FS	256
Chrome 31 / Win 7	TLS 1.2	TLS_ECDHE_RSA_WITH_AES_256_CBC_SHA (0xc014)	FS	256
Firefox 10.0.12 ESR / Win 7	TLS 1.0	TLS_DHE_RSA_WITH_CAMELLIA_256_CBC_SHA (0xc88)	FS	256
Firefox 17.0.7 ESR / Win 7	TLS 1.0	TLS_DHE_RSA_WITH_CAMELLIA_256_CBC_SHA (0xc88)	FS	256
Firefox 21 / Fedora 19	TLS 1.0	TLS_DHE_RSA_WITH_CAMELLIA_256_CBC_SHA (0xc88)	FS	256
Firefox 24 / Win 7	TLS 1.0	TLS_DHE_RSA_WITH_CAMELLIA_256_CBC_SHA (0xc88)	FS	256
Googlebot Oct 2013	TLS 1.0	TLS_ECDHE_RSA_WITH_AES_256_CBC_SHA (0xc014)	FS	256
IE 6 / XP	No FS ¹	No SN ²		Fail ³
IE 7 / Vista	TLS 1.0	TLS_ECDHE_RSA_WITH_AES_256_CBC_SHA (0xc014)	FS	256
IE 8 / XP	No FS ¹	No SN ²		Fail ³
IE 8-10 / Win 7	TLS 1.0	TLS_ECDHE_RSA_WITH_AES_256_CBC_SHA (0xc014)	FS	256
IE 11 / Win 7	TLS 1.2	TLS_ECDHE_RSA_WITH_AES_256_CBC_SHA (0xc014)	FS	256
IE 11 / Win 8.1	TLS 1.2	TLS_ECDHE_RSA_WITH_AES_256_CBC_SHA (0xc014)	FS	256
Java 6u45	No SN ²			Fail ³
Java 7u25				Fail ³
OpenSSL 0.9.8y	TLS 1.0	TLS_DHE_RSA_WITH_AES_256_CBC_SHA (0xc39)	FS	256
OpenSSL 1.0.1e	TLS 1.2	TLS_ECDHE_RSA_WITH_AES_256_GCM_SHA384 (0xc030)	FS	256
Opera 17 / Win 7	TLS 1.2	TLS_DHE_RSA_WITH_AES_256_CBC_SHA256 (0xc6b)	FS	256
Safari 5.1.9 / OS X 10.6.8	TLS 1.0	TLS_ECDHE_RSA_WITH_AES_256_CBC_SHA (0xc014)	FS	256
Safari 6 / iOS 6.0.1	TLS 1.2	TLS_ECDHE_RSA_WITH_AES_256_CBC_SHA384 (0xc028)	FS	256
Safari 6.0.4 / OS X 10.8.4	TLS 1.0	TLS_ECDHE_RSA_WITH_AES_256_CBC_SHA (0xc014)	FS	256
Safari 7 / OS X 10.9	TLS 1.2	TLS_ECDHE_RSA_WITH_AES_256_CBC_SHA384 (0xc028)	FS	256
Tor 17.0.9 / Win 7	TLS 1.0	TLS_DHE_RSA_WITH_CAMELLIA_256_CBC_SHA (0xc88)	FS	256
Yahoo Slurp Oct 2013	TLS 1.0	TLS_DHE_RSA_WITH_CAMELLIA_256_CBC_SHA (0xc88)	FS	256

Konfigurationsschnipsel

BetterCrypto hat bereits fertige Konfigurationsschnipsel für die folgenden Softwarepakete:

- Webserver: Apache, nginx, lighttpd, MS IIS
- Mailserver: Dovecot, cyrus, Postfix, Exim
- DB-Systeme: MySQL, Oracle, PostgreSQL, DB2
- VPN-Lösungen: OpenVPN, IPSec, Checkpoint
- Proxy-Server: Squid, Pound
- IM-Server: Jabber, IRC
- GnuPG
- SSH

Auf der Wunschliste stehen insbesondere noch Schnipsel für:

- Exchange
- SIP
- RDP

Außerdem auf der Liste:

- Schnipsel nicht nur im PDF, sondern auch als HTML verfügbar machen – leichter zu kopieren!
- Ein webbasierter Konfigurator zum individuellen Erstellen von Konfigurationsschnipseln.

■ Cipher Suites konfigurieren:

```
SSLProtocol All -SSLv2 -SSLv3
SSLHonorCipherOrder On
SSLCompression off
# Add six earth month HSTS header for all users...
Header add Strict-Transport-Security "max-age=15768000"
# If you want to protect all subdomains, use the following header
# ALL subdomains HAVE TO support https if you use this!
# Strict-Transport-Security: max-age=15768000 ; includeSubDomains

SSLCipherSuite 'EECDH+aRSA+AESGCM:EECDH+aRSA+SHA384:EECDH+aRSA+SHA256:EDH
+CAMELLIA256:EECDH:EDH+aRSA:+SSLv3:!aNULL:!eNULL:!LOW:!3DES:!MD5:!EXP
:!PSK:!SRP:!DSS:!RC4:!SEED:!AES128:!CAMELLIA128:!ECDH:AES256-SHA'
```

■ Redirect von HTTP auf HTTPS konfigurieren:

```
<VirtualHost *:80>
#...
RewriteEngine On
RewriteRule ^.*$ https://%{SERVER_NAME}%{REQUEST_URI} [L,R=
permanent]
#...
</VirtualHost>
```

Testen und Verifizieren

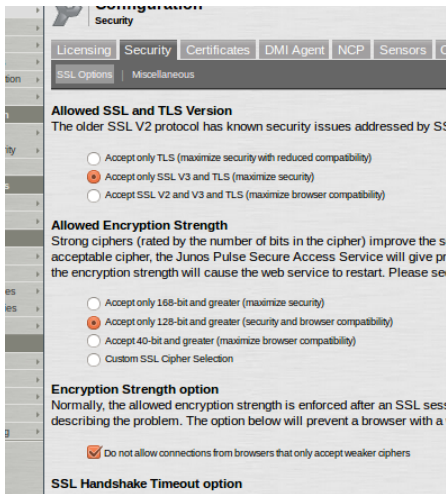
»... aber unsere Einstellungen sind doch super! Die haben wir von BetterCrypto.org kopiert!«

Ja, aber:

- Auch bei Copy-and-Paste passieren Fehler.
- Konfigurationen müssen auch an der richtigen Stelle stehen.
- Gelegentlich sind Konfigurationsmöglichkeiten auch sehr unklar dokumentiert.

Darüber hinaus:

- Ist es möglicherweise eine gute Idee, den Ist-Zustand der eigenen Infrastruktur genauer kennenzulernen, um gezielte Verbesserungen durchzuführen.
- Ist es immer gut zu verifizieren, dass die Konfiguration, die man glaubt ausgerollt zu haben, auch tatsächlich gezogen wird.



The screenshot shows the 'Configuration' page for 'Security' in the Junos Pulse Secure management console. The 'SSL Options' tab is selected. The page is divided into three main sections: 'Allowed SSL and TLS Version', 'Allowed Encryption Strength', and 'Encryption Strength option'. Each section contains radio button options for different security levels. The 'Allowed SSL and TLS Version' section has three options: 'Accept only TLS (maximize security with reduced compatibility)', 'Accept only SSL V3 and TLS (maximize security)' (which is selected), and 'Accept SSL V2 and V3 and TLS (maximize browser compatibility)'. The 'Allowed Encryption Strength' section has four options: 'Accept only 168-bit and greater (maximize security)', 'Accept only 128-bit and greater (security and browser compatibility)' (which is selected), 'Accept 40-bit and greater (maximize browser compatibility)', and 'Custom SSL Cipher Selection'. The 'Encryption Strength option' section has a checked checkbox for 'Do not allow connections from browsers that only accept weaker ciphers'. The 'SSL Handshake Timeout option' section is partially visible at the bottom.

Configuration
Security

Licensing | Security | Certificates | DMI Agent | NCP | Sensors | C

SSL Options | Miscellaneous

Allowed SSL and TLS Version
The older SSL V2 protocol has known security issues addressed by SS

☐ Accept only TLS (maximize security with reduced compatibility)
☒ Accept only SSL V3 and TLS (maximize security)
☐ Accept SSL V2 and V3 and TLS (maximize browser compatibility)

Allowed Encryption Strength
Strong ciphers (rated by the number of bits in the cipher) improve the s
acceptable cipher, the Junos Pulse Secure Access Service will give pr
the encryption strength will cause the web service to restart. Please see

☐ Accept only 168-bit and greater (maximize security)
☒ Accept only 128-bit and greater (security and browser compatibility)
☐ Accept 40-bit and greater (maximize browser compatibility)
☐ Custom SSL Cipher Selection

Encryption Strength option
Normally, the allowed encryption strength is enforced after an SSL sess
describing the problem. The option below will prevent a browser with a

☒ Do not allow connections from browsers that only accept weaker ciphers

SSL Handshake Timeout option

- openssl s_client bzw. GNUTLS-CLI
- sslabs.com
- xmpp.net
- sslscan
- SSLyze


```
openssl s_client -connect git.bettercrypto.org:443
```

```
New, TLSv1/SSLv3, Cipher is ECDHE-RSA-AES256-GCM-SHA384
Server public key is 4096 bit
Secure Renegotiation IS supported
Compression: NONE
Expansion: NONE
SSL-Session:
  Protocol : TLSv1.2
  Cipher   : ECDHE-RSA-AES256-GCM-SHA384
  Session-ID: 53D90B7D9D1FFC7EA98C105A2FC27F752B9CE9026CDAB57F4A7D4491C3C5ECC6
  Session-ID-ctx:
  Master-Key: 8F06DE9669BD6BF9628A38DF4F92C2CEBA6B7EA91F465164440CF31F7E8F55F2A67E7320B388D6E7AC4BC141C2FF3F68
  Key-Arg : None
  PSK identity: None
  PSK identity hint: None
  SRP username: None
  TLS session ticket lifetime hint: 300 (seconds)
  TLS session ticket:
0000 - fe 5b 93 84 a8 c6 ab 4a-74 b8 59 81 dc 3e 52 40  .[.....Jt.Y..>R@
0010 - 0e dd f6 59 b4 a1 d2 54-65 df 9a 1b c9 fb 0d 2e    ...Y...Te.....
0020 - 64 9c 65 cf 1c 0d d9 19-57 a6 cd 50 a5 d9 16 a4    d.e.....W..P....
0030 - 17 b6 e8 38 ac e5 76 15-a4 9d d5 62 ee 51 55 09    ...8..v....b.QU.
0040 - 52 36 58 84 04 0f 93 94-7b a9 dc e3 6f 8e 2f 7a    R6X.....{...o./z
0050 - 9f bf 3d 4f a1 e1 bb 83-21 0f 7d f2 bd 02 48 a6    ..=0.....!.)...H.
0060 - 5a 96 82 fd dc a6 5a 55-77 b3 9f fb 60 0d 86 66    Z.....ZUw...`..f
0070 - f1 68 42 e2 90 93 8b f6-25 aa 85 cf 08 07 c6 76    .hB.....%.....v
0080 - 06 62 37 32 09 4f ac 23-28 9c db b9 29 c0 23 1b    .b72.0.#{...).#.
0090 - e4 c3 d2 a3 a4 b4 87 b5-0e 5c 68 16 73 07 96 90    .....h.s...

Start Time: 1385118946
Timeout : 300 (sec)
Verify return code: 21 (unable to verify the first certificate)
---
```

Command-line-Tool zum Testen von Cipher-Parametern von Webservern.

```

  ssllscan

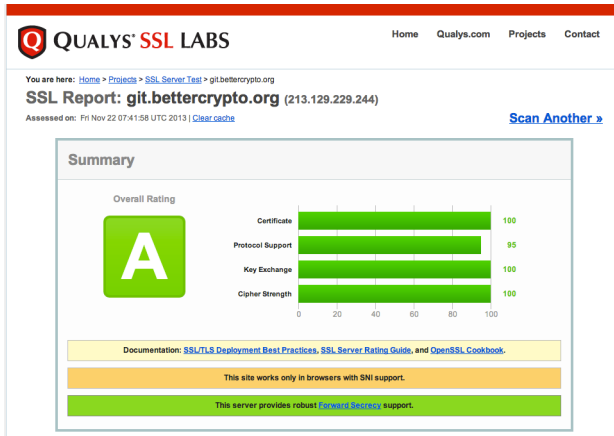
      Version 1.8.2
      http://www.titania.co.uk
      Copyright Ian Ventura-Whiting 2009

Testing SSL server git.bettercrypto.org on port 443

Supported Server Cipher(s):
Failed    SSLv2    168 bits  DES-CBC3-MD5
Failed    SSLv2    128 bits  IDEA-CBC-MD5
Failed    SSLv2    128 bits  RC2-CBC-MD5
Failed    SSLv2    128 bits  RC4-MD5
Failed    SSLv2    56 bits   DES-CBC-MD5
Failed    SSLv2    40 bits   EXP-RC2-CBC-MD5
Failed    SSLv2    40 bits   EXP-RC4-MD5
Failed    SSLv3    256 bits  ECDHE-RSA-AES256-GCM-SHA384
Failed    SSLv3    256 bits  ECDHE-ECDSA-AES256-GCM-SHA384
Failed    SSLv3    256 bits  ECDHE-RSA-AES256-SHA384
Failed    SSLv3    256 bits  ECDHE-ECDSA-AES256-SHA384
Rejected  SSLv3    256 bits  ECDHE-RSA-AES256-SHA
Rejected  SSLv3    256 bits  ECDHE-ECDSA-AES256-SHA
Rejected  SSLv3    256 bits  SRP-DSS-AES-256-CBC-SHA
Rejected  SSLv3    256 bits  SRP-RSA-AES-256-CBC-SHA
Failed    SSLv3    256 bits  DHE-DSS-AES256-GCM-SHA384
Failed    SSLv3    256 bits  DHE-RSA-AES256-GCM-SHA384
Failed    SSLv3    256 bits  DHE-RSA-AES256-SHA256

```

Online-Variante von sslscan, auch zum Testen des eigenen Browsers geeignet.



Configuration



Protocols

TLS 1.2	Yes
TLS 1.1	Yes
TLS 1.0	Yes
SSL 3	No
SSL 2	No



Cipher Suites (SSL 3+ suites in server-preferred order, then SSL 2 suites where used)

TLS_ECDHE_RSA_WITH_AES_256_GCM_SHA384 (0xc030)	ECDH 256 bits (eq. 3072 bits RSA) FS	256
TLS_ECDHE_RSA_WITH_AES_256_CBC_SHA384 (0xc028)	ECDH 256 bits (eq. 3072 bits RSA) FS	256
TLS_DHE_RSA_WITH_AES_256_GCM_SHA384 (0xc09f)	DH 4096 bits (p: 512, g: 1, Ys: 512) FS	256
TLS_DHE_RSA_WITH_AES_256_CBC_SHA256 (0xc06b)	DH 4096 bits (p: 512, g: 1, Ys: 512) FS	256
TLS_DHE_RSA_WITH_CAMELLIA_256_CBC_SHA (0xc088)	DH 4096 bits (p: 512, g: 1, Ys: 512) FS	256
TLS_ECDHE_RSA_WITH_AES_256_CBC_SHA (0xc014)	ECDH 256 bits (eq. 3072 bits RSA) FS	256
TLS_DHE_RSA_WITH_AES_256_CBC_SHA (0xc039)	DH 4096 bits (p: 512, g: 1, Ys: 512) FS	256
TLS_RSA_WITH_AES_256_CBC_SHA (0xc035)		256

Summary

Overall Rating



Documentation: [SSL/TLS Deployment Best Practices](#), [SSL Server Rating Guide](#), and [OpenSSL Cookbook](#).

This server supports SSL 2, which is obsolete and insecure. Grade set to F.

The server supports only older protocols, but not the current best TLS 1.2. Grade capped to B.

The server does not support Forward Secrecy with the reference browsers. [MORE INFO »](#)

Authentication

Wrap-Up

- Erster Public Draft ist soweit fertig.
- Präsentationen unter anderem auf dem 30C3 und dem TF-CSIRT-Treffen in Zürich.
- Verbindung mit der IETF aufgenommen.
- Debian-Entwicklerteam hat Interesse bekundet, die Empfehlungen grundsätzlich als Standard zu übernehmen.
- HTML-Variante des Whitepapers ist noch zu erarbeiten.
- ... und natürlich immer Aufräum- und Housekeeping-Arbeiten.

- Ja, bitte! Wir brauchen Kryptographen und Admins, die das Whitepaper reviewen.
- Der LaTeX-Source des Whitepapers ist komplett frei als Git-Repo verfügbar.
- Die Mailingliste kann ebenfalls frei subskribiert werden.
- Neue Konfigurationsschnipsel bitte zunächst auf Variante B basieren und als Diffs verfügbar machen.

Fragen?

Vielen Dank für die Aufmerksamkeit!

tobias.dussa@kit.edu; Telefon 0721-608-42479

PGP-Fingerprint:

0D29 63BE DB07 1264 DD1C
EFE0 **34E7 F72A 2366 36AE**