ECCCurve Unterstützung Windows 10/2016

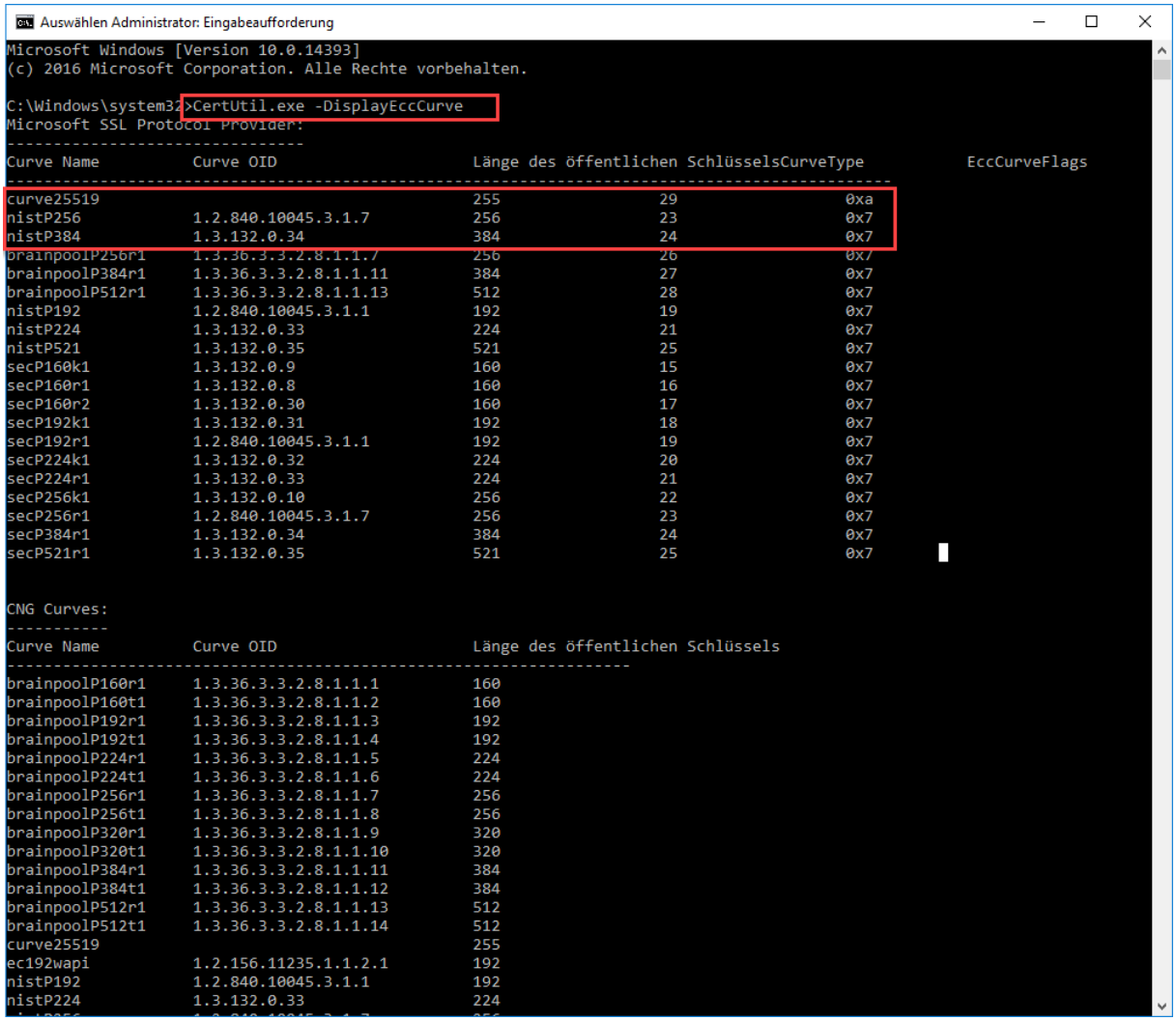
Welche ECCKurven werden von einem System unterstützt und in welcher Reihenfolge.

Mit dem Befehl

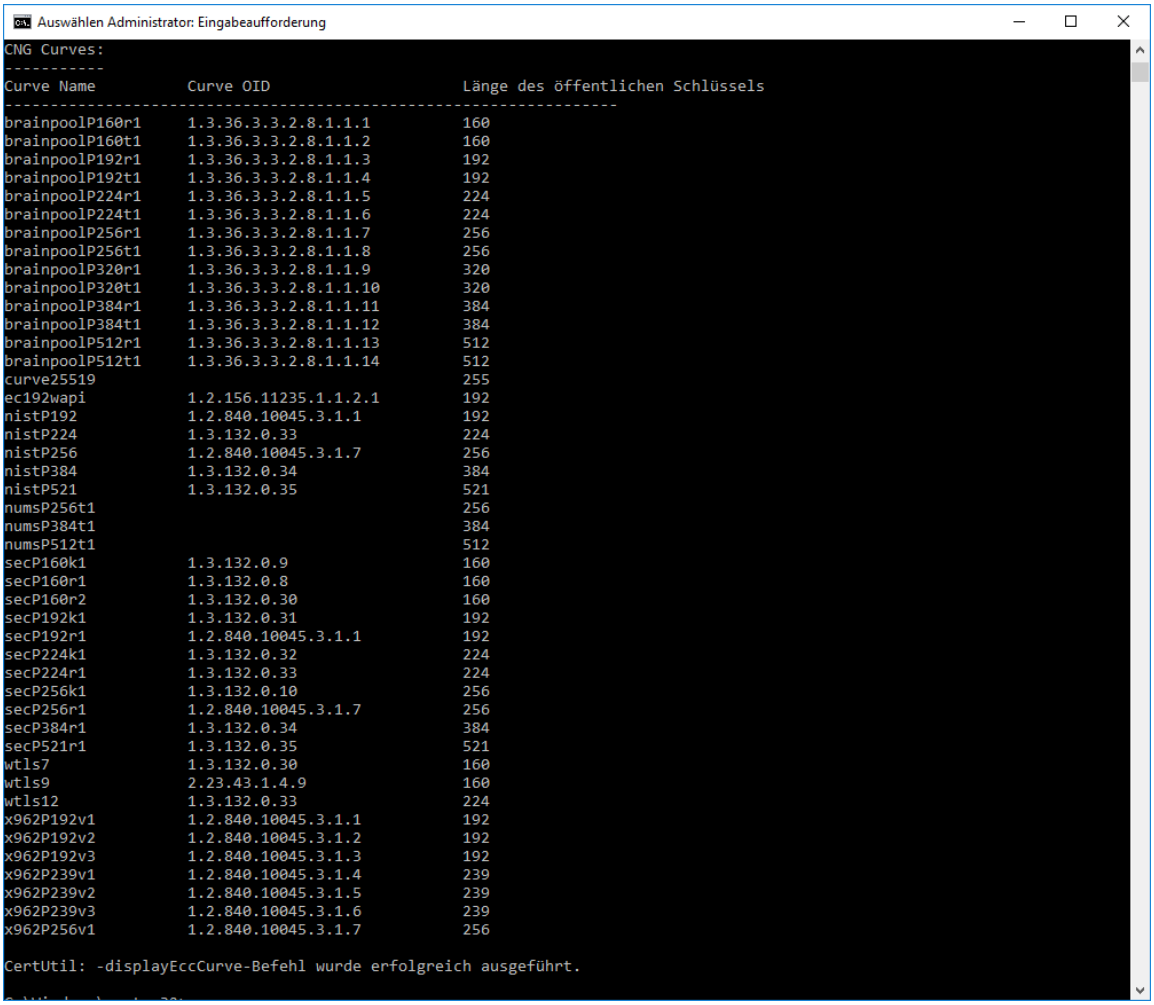
- CertUtil.exe –DisplayEccCurve

lassen wir uns die vom System unterstützen Kurven anzeigen. In diesem Fall handelt es sich um einen Windows Server 2016.

Die ersten 3 rot markierten Kurven sind bevorzugte Kurven. Diese lassen sich über eine Gruppenrichtlinie steuern.



ECCCurve Unterstützung Windows 10/2016



Korrespondierende Kurven und ihre Notationen:

- **P-256 = secp256r1 eine gute Wahl**
- P-384 = secp384r1
- P-521 = secp521r1
- B-283 = sect283r1
- K-283 = sect283k1
- B-409 = sect409r1
- K-409 = sect409k1
- B-571 = sect571r1
- K-571 = sect571k1

Für den Einsatz von PFS (Perfect Forward Secrecy) ist es erforderlich Cipher Suiten bevorzugt einzusetzen die ECDHE/DHE unterstützen, wie in diesem Beispiel.

Handshake Simulation (Experimental)

Chrome 27	TLS 1.1	TLS_ECDHE_ECDSA_WITH_RC4_128_SHA (0xc007) Forward Secrecy	128
Firefox 21	TLS 1.0	TLS_ECDHE_ECDSA_WITH_RC4_128_SHA (0xc007) Forward Secrecy	128
Internet Explorer 10	TLS 1.0	TLS_ECDHE_ECDSA_WITH_AES_128_CBC_SHA (0xc009) Forward Secrecy	128
Safari iOS 6.0.1	TLS 1.2	TLS_ECDHE_RSA_WITH_RC4_128_SHA (0xc011) Forward Secrecy	128
Safari 5.1.9	TLS 1.0	TLS_ECDHE_RSA_WITH_RC4_128_SHA (0xc011) Forward Secrecy	128

ECCCurve Unterstützung Windows 10/2016

Beim roll-out ist darauf zu achten, dass diese dann auch in der Reihenfolge ganz oben stehen. Gegenüber dem RSA Verfahren (welches vermieden werden sollte) haben ECC-Verfahren in puncto Schlüssellänge noch weitere Vorteile als DL-Verfahren ohne elliptische Kurven. Denn eine RSA-Schlüssellänge von 1.024 Bit entspricht eine ECC-Schlüssellänge von 160 und das bei vergleichbarer Sicherheit. Dadurch ist das Signieren oder die Diffie-Hellman-Berechnung eines ECC-Verfahrens bei gleicher Sicherheitsstufe etwa um den Faktor zehn schneller als bei RSA. Das spart Rechenzeit und Bandbreite. Ein weiterer Vorteil ist, dass diese Methode in keiner Abhängig zum privaten Schlüssel des Servers steht. Das bedeutet im Detail, wenn Cipher Suiten eingesetzt werden die kein Perfect Forward Secrecy unterstützen und der Privat-Key in fremde Hände gelangt ist, so kann die aufgezeichnete und verschlüsselte Konversation ohne weitere Probleme und großen Aufwand entschlüsselt werden.

Das Ganze mal als Hintergrundwissen zum Thema Perfect Forward Secrecy und warum PFS Sinn macht.

Das ECC-Verfahren wird eingesetzt für die Anwendungen

- Schlüsselaustausch
- Digitale Signatur
- asymmetrische Verschlüsselung

ECDHE = ist die Abkürzung für Elliptic Curve Diffie-Hellmann und bezeichnet das DH-Verfahren auf Basis elliptischer Kurven

ECDSA = ist die Abkürzung für Elliptic Curve DSA (Digital Signature Algorithm) und dieses lässt sich auch mit elliptischen Kurven durchsetzen, ist aber noch langsamer als RSA.

Je nach eingesetzter Cipher Suite ergeben sich folgende Abweichungen:

TLS_RSA_WITH_RC4_128_MD5

- Sicherer Austausch des Schlüssels zwischen Client und Server
- Authentifizierung über RSA
- Verschlüsselung der Daten über RC4
- Die Integrität der Daten mittels MD5

ECDHE-RSA-AES256-GCM-SHA384

- Sicherer Austausch des Schlüssels zwischen Client und Server mittels ECDHE
- Authentifizierung über RSA
- Verschlüsselung der Daten mit AES/GCM
- Die Integrität der Daten mittels SHA2

Die erste Cipher Suite ist unsicher, weil RC4 und MD5 eingesetzt werden. Die zweite Cipher Suite nutzt ECDHE und kann verhindern, das bekanntgewordene Schlüssel zum Entschlüsseln eingesetzt werden.

Also, für TLS Verbindungen sollten nur Cipher Suiten eingesetzt werden die folgende Kriterien erfüllen:

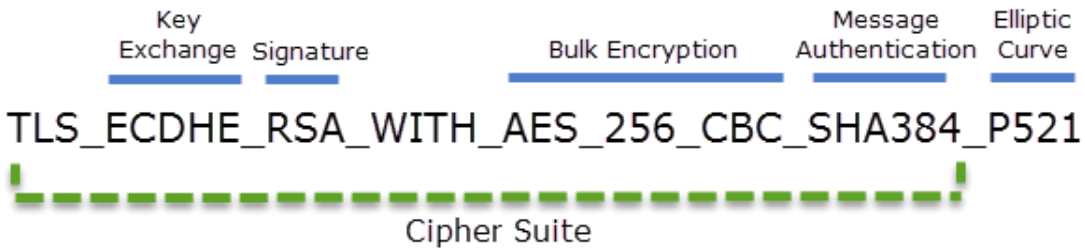
- Schlüsselaustausch über ECDHE
- RSA/ECDSA als Signaturverfahren/Authentizität
- Symmetrische Verschlüsselung der Kommunikation mit AES-GCM ab 128 Bit
- Zur Sicherstellung der Datenintegrität SHA-2 als Hashfunktion und stärker

Erstellt von Jörn Walter **03.03.2018**

Einsetzbar sind somit z.B.

- TLS_DHE_RSA_WITH_AES_128_GCM_SHA256
- ECDHE-ECDSA-AES256-GCM-SHA384
- ECDHE-ECDSA-AES128-GCM-SHA256
- ECDHE-RSA-AES256-GCM-SHA384
- ECDHE-RSA-AES128-GCM-SHA256
- DHE-RSA-AES256-GCM-SHA384
- DHE-RSA-AES128-GCM-SHA256

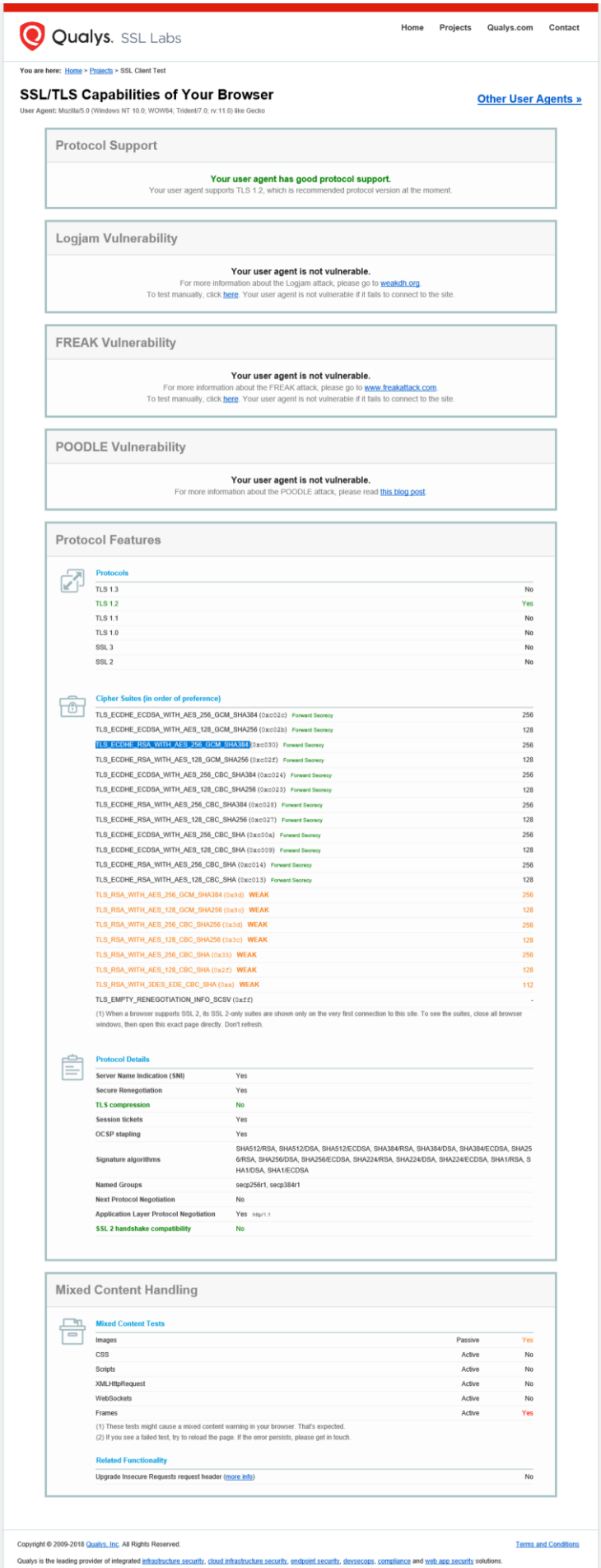
Jede Cipher Suite besteht aus einem Schlüsselaustauschalgorithmus, einem Cipher Algorithmus und einem MAC (Hash) Algorithmus. Message Authentication (MAC) sind Algorithmen die Hashes und Signaturen erzeugen um die Integrität einer Nachricht sicherzustellen.



Der nachfolgende Test zeigt, dass mein Windows 10 auch schwache Cipher Suiten anbietet. Über den Gruppenrichtlinieneditor werde ich die schwachen Cipher Suiten entfernen und nur noch diese systemseitig hinterlegen:

TLS_AES_256_GCM_SHA384,TLS_AES_128_GCM_SHA256,TLS_ECDHE_ECDSA_WITH_AES_256_GCM_SHA384,TLS_ECDHE_ECDSA_WITH_AES_128_GCM_SHA256,TLS_ECDHE_RSA_WITH_AES_256_GCM_SHA384,TLS_ECDHE_RSA_WITH_AES_128_GCM_SHA256,TLS_DHE_RSA_WITH_AES_256_GCM_SHA384,TLS_DHE_RSA_WITH_AES_128_GCM_SHA256,TLS_ECDHE_ECDSA_WITH_AES_256_CBC_SHA384,TLS_ECDHE_ECDSA_WITH_AES_128_CBC_SHA256,TLS_ECDHE_RSA_WITH_AES_256_CBC_SHA384,TLS_ECDHE_RSA_WITH_AES_128_CBC_SHA256,TLS_ECDHE_ECDSA_WITH_AES_256_CBC_SHA,TLS_ECDHE_ECDSA_WITH_AES_128_CBC_SHA,TLS_ECDHE_RSA_WITH_AES_256_CBC_SHA,TLS_ECDHE_RSA_WITH_AES_128_CBC_SHA,TLS_RSA_WITH_NULL_SHA256,TLS_RSA_WITH_NULL_SHA,TLS_PSK_WITH_AES_256_GCM_SHA384,TLS_PSK_WITH_AES_128_GCM_SHA256,TLS_PSK_WITH_AES_256_CBC_SHA384,TLS_PSK_WITH_AES_128_CBC_SHA256,TLS_PSK_WITH_NULL_SHA384,TLS_PSK_WITH_NULL_SHA256

ECCCurve Unterstützung Windows 10/2016



Qualys. SSL Labs

Home Projects Qualys.com Contact

You are here: Home > Projects > SSL Client Test

SSL/TLS Capabilities of Your Browser

User Agent: Mozilla/5.0 (Windows NT 10.0; WOW64; Trident/7.0; rv:11.0) like Gecko

Other User Agents »

Protocol Support

Your user agent has good protocol support.
Your user agent supports TLS 1.2, which is recommended protocol version at the moment.

Logjam Vulnerability

Your user agent is not vulnerable.
For more information about the Logjam attack, please go to weakdh.org.
To test manually, click here. Your user agent is not vulnerable if it fails to connect to the site.

FREAK Vulnerability

Your user agent is not vulnerable.
For more information about the FREAK attack, please go to www.freakattack.com.
To test manually, click here. Your user agent is not vulnerable if it fails to connect to the site.

POODLE Vulnerability

Your user agent is not vulnerable.
For more information about the POODLE attack, please read this blog post.

Protocol Features

Protocols

TLS 1.3	No
TLS 1.2	Yes
TLS 1.1	No
TLS 1.0	No
SSL 3	No
SSL 2	No

Cipher Suites (in order of preference)

TLS_ECDHE_ECDSA_WITH_AES_256_GCM_SHA384 (0xc02c) Forward Secrecy	256
TLS_ECDHE_ECDSA_WITH_AES_128_GCM_SHA256 (0xc02b) Forward Secrecy	128
TLS_ECDHE_RSA_WITH_AES_256_GCM_SHA384 (0xc030) Forward Secrecy	256
TLS_ECDHE_RSA_WITH_AES_128_GCM_SHA256 (0xc02f) Forward Secrecy	128
TLS_ECDHE_ECDSA_WITH_AES_256_CBC_SHA384 (0xc024) Forward Secrecy	256
TLS_ECDHE_ECDSA_WITH_AES_128_CBC_SHA256 (0xc023) Forward Secrecy	128
TLS_ECDHE_RSA_WITH_AES_256_CBC_SHA384 (0xc028) Forward Secrecy	256
TLS_ECDHE_RSA_WITH_AES_128_CBC_SHA256 (0xc027) Forward Secrecy	128
TLS_ECDHE_ECDSA_WITH_AES_256_CBC_SHA (0xc00a) Forward Secrecy	256
TLS_ECDHE_ECDSA_WITH_AES_128_CBC_SHA (0xc009) Forward Secrecy	128
TLS_ECDHE_RSA_WITH_AES_256_CBC_SHA (0xc014) Forward Secrecy	256
TLS_ECDHE_RSA_WITH_AES_128_CBC_SHA (0xc013) Forward Secrecy	128
TLS_RSA_WITH_AES_256_GCM_SHA384 (0x9d) WEAK	256
TLS_RSA_WITH_AES_128_GCM_SHA256 (0x9c) WEAK	128
TLS_RSA_WITH_AES_256_CBC_SHA256 (0x3d) WEAK	256
TLS_RSA_WITH_AES_128_CBC_SHA256 (0x3c) WEAK	128
TLS_RSA_WITH_AES_256_CBC_SHA (0x35) WEAK	256
TLS_RSA_WITH_AES_128_CBC_SHA (0x2f) WEAK	128
TLS_RSA_WITH_3DES_EDE_CBC_SHA (0xa) WEAK	112
TLS_EMPTY_RENEGOTIATION_INFO_SCSV (0xff)	-

(1) When a browser supports SSL 2, its SSL 2-only suites are shown only on the very first connection to this site. To see the suites, close all browser windows, then open this exact page directly. Don't refresh.

Protocol Details

Server Name Indication (SNI)	Yes
Secure Renegotiation	Yes
TLS compression	No
Session tickets	Yes
OCSP stapling	Yes
Signature algorithms	SHA512/RSA, SHA512/DSA, SHA512/ECDSA, SHA384/RSA, SHA384/DSA, SHA384/ECDSA, SHA256/RSA, SHA256/DSA, SHA256/ECDSA, SHA224/RSA, SHA224/DSA, SHA224/ECDSA, SHA1/RSA, SHA1/DSA, SHA1/ECDSA
Named Groups	secp256r1, secp384r1
Next Protocol Negotiation	No
Application Layer Protocol Negotiation	Yes http/1.1
SSL 2 handshake compatibility	No

Mixed Content Handling

Mixed Content Tests

Images	Passive	Yes
CSS	Active	No
Scripts	Active	No
XMLHttpRequest	Active	No
WebSockets	Active	No
Frames	Active	Yes

(1) These tests might cause a mixed content warning in your browser. That's expected.
(2) If you see a failed test, try to reload the page. If the error persists, please get in touch.

Related Functionality

Upgrade Insecure Requests request header (more info)	No

Copyright © 2009-2018 Qualys, Inc. All Rights Reserved. Terms and Conditions

Qualys is the leading provider of integrated infrastructure security, cloud infrastructure security, endpoint security, devsecops, compliance and web app security solutions.

ECCCurve Unterstützung Windows 10/2016

Qualys. SSL Labs

Home Projects Qualys.com Contact

You are here: Home > Projects > SSL Client Test

SSL/TLS Capabilities of Your Browser

User Agent: Mozilla/5.0 (Windows NT 10.0; WOW64; Trident/7.0; rv:11.0) like Gecko

Other User Agents »

Protocol Support

Your user agent has good protocol support.
Your user agent supports TLS 1.2, which is recommended protocol version at the moment.

Logjam Vulnerability

Your user agent is not vulnerable.
For more information about the Logjam attack, please go to weakdh.org.
To test manually, click here. Your user agent is not vulnerable if it fails to connect to the site.

FREAK Vulnerability

Your user agent is not vulnerable.
For more information about the FREAK attack, please go to www.freakattack.com.
To test manually, click here. Your user agent is not vulnerable if it fails to connect to the site.

POODLE Vulnerability

Your user agent is not vulnerable.
For more information about the POODLE attack, please read this blog post.

Protocol Features

Protocols

TLS 1.3	No
TLS 1.2	Yes
TLS 1.1	No
TLS 1.0	No
SSL 3	No
SSL 2	No

Cipher Suites (in order of preference)

TLS_ECDHE_ECDSA_WITH_AES_256_GCM_SHA384 (0xc02c) Forward Secrecy	256
TLS_ECDHE_ECDSA_WITH_AES_128_GCM_SHA256 (0xc02b) Forward Secrecy	128
TLS_ECDHE_RSA_WITH_AES_256_GCM_SHA384 (0xc030) Forward Secrecy	256
TLS_ECDHE_RSA_WITH_AES_128_GCM_SHA256 (0xc02f) Forward Secrecy	128
TLS_ECDHE_ECDSA_WITH_AES_256_CBC_SHA384 (0xc024) Forward Secrecy	256
TLS_ECDHE_ECDSA_WITH_AES_128_CBC_SHA256 (0xc023) Forward Secrecy	128
TLS_ECDHE_RSA_WITH_AES_256_CBC_SHA384 (0xc028) Forward Secrecy	256
TLS_ECDHE_RSA_WITH_AES_128_CBC_SHA256 (0xc027) Forward Secrecy	128
TLS_ECDHE_ECDSA_WITH_AES_256_CBC_SHA (0xc00a) Forward Secrecy	256
TLS_ECDHE_ECDSA_WITH_AES_128_CBC_SHA (0xc009) Forward Secrecy	128
TLS_ECDHE_RSA_WITH_AES_256_CBC_SHA (0xc014) Forward Secrecy	256
TLS_ECDHE_RSA_WITH_AES_128_CBC_SHA (0xc013) Forward Secrecy	128
TLS_EMPTY_RENEGOTIATION_INFO_SCSV (0xff)	-

(1) When a browser supports SSL 2, its SSL 2-only suites are shown only on the very first connection to this site. To see the suites, close all browser windows, then open this exact page directly. Don't refresh.

Protocol Details

Server Name Indication (SNI)	Yes
Secure Renegotiation	Yes
TLS compression	No
Session tickets	Yes
OCSP stapling	Yes
Signature algorithms	SHA512/RSA, SHA512/DSA, SHA512/ECDSA, SHA384/RSA, SHA384/DSA, SHA384/ECDSA, SHA256/RSA, SHA256/DSA, SHA256/ECDSA, SHA224/RSA, SHA224/DSA, SHA224/ECDSA, SHA1/RSA, SHA1/DSA, SHA1/ECDSA
Named Groups	secp256r1, secp384r1
Next Protocol Negotiation	No
Application Layer Protocol Negotiation	Yes http/1.1
SSL 2 handshake compatibility	No

Mixed Content Handling

Mixed Content Tests

Images	Passive	Yes
CSS	Active	Yes
Scripts	Active	Yes
XMLHttpRequest	Active	No
WebSockets	Active	No
Frames	Active	Yes

(1) These tests might cause a mixed content warning in your browser. That's expected.
(2) If you see a failed test, try to reload the page. If the error persists, please get in touch.

Related Functionality

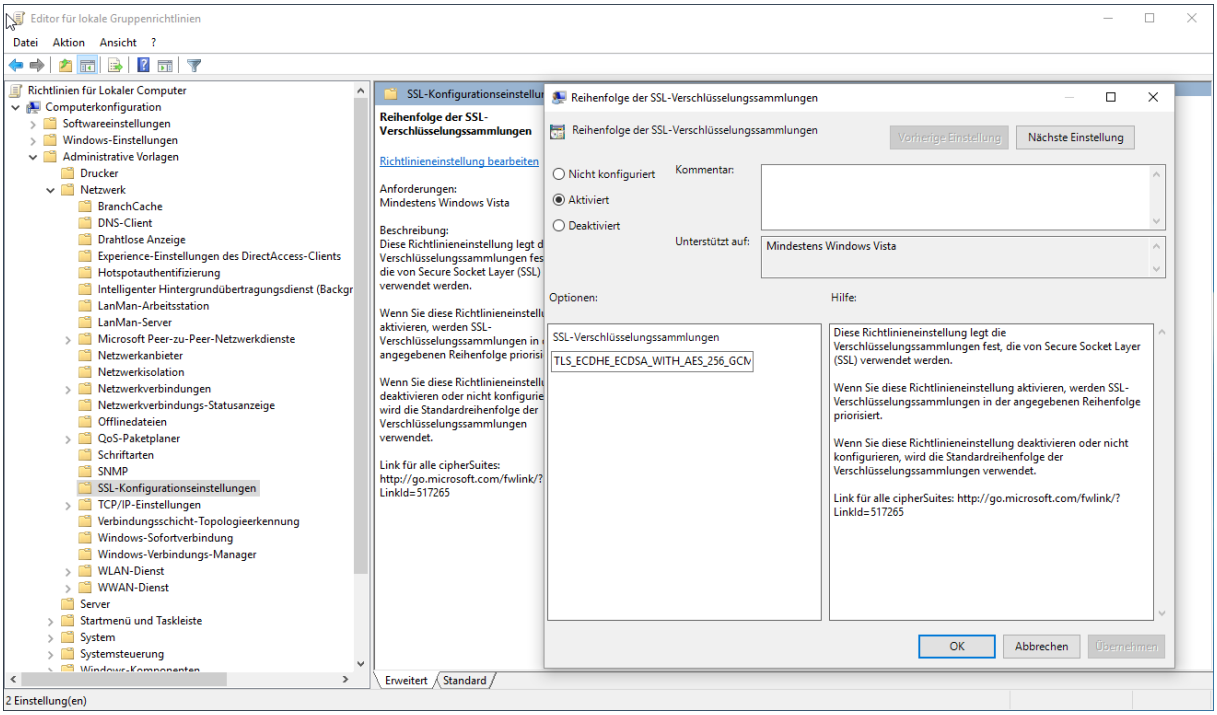
Upgrade Insecure Requests request header (more info)	No

Copyright © 2009-2018 Qualys, Inc. All Rights Reserved. Terms and Conditions

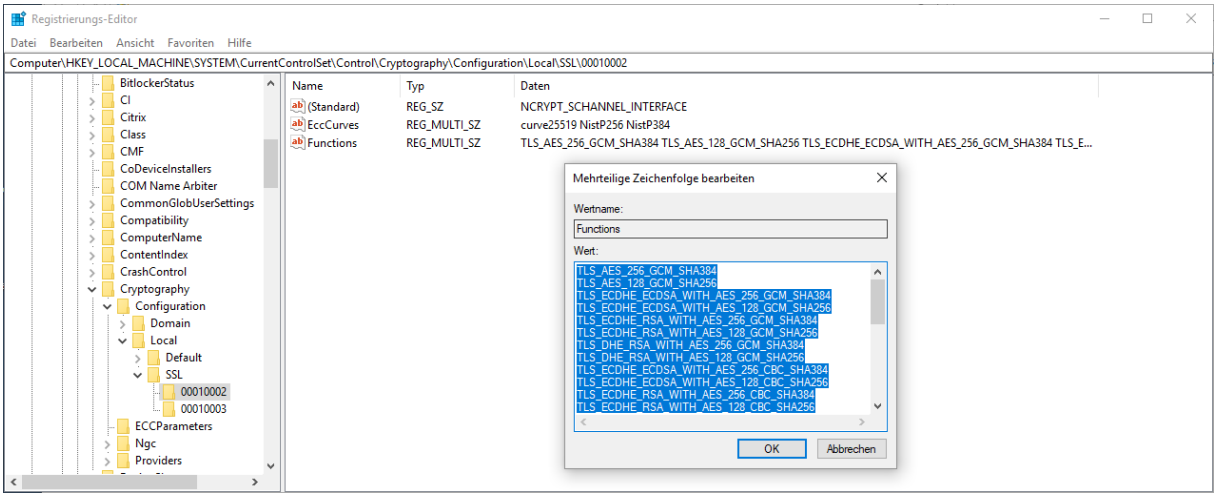
Qualys is the leading provider of integrated infrastructure security, cloud infrastructure security, endpoint security, devsecops, compliance and web app security solutions.

ECCCurve Unterstützung Windows 10/2016

Mittels dieser Richtlinie lassen sich die einzusetzenden Cipher Suiten bestimmen, maximal 1024 Zeichen möglich.

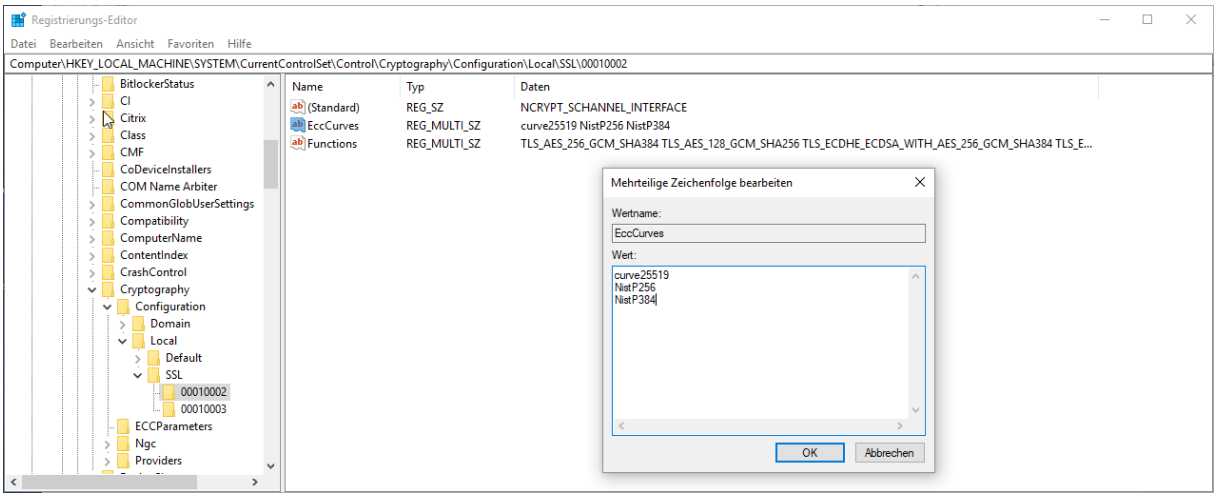


In der Registry finden wir die Cipher Suiten unter diesem Schlüssel.



ECCCurve Unterstützung Windows 10/2016

Die bevorzugten ECCCurves:



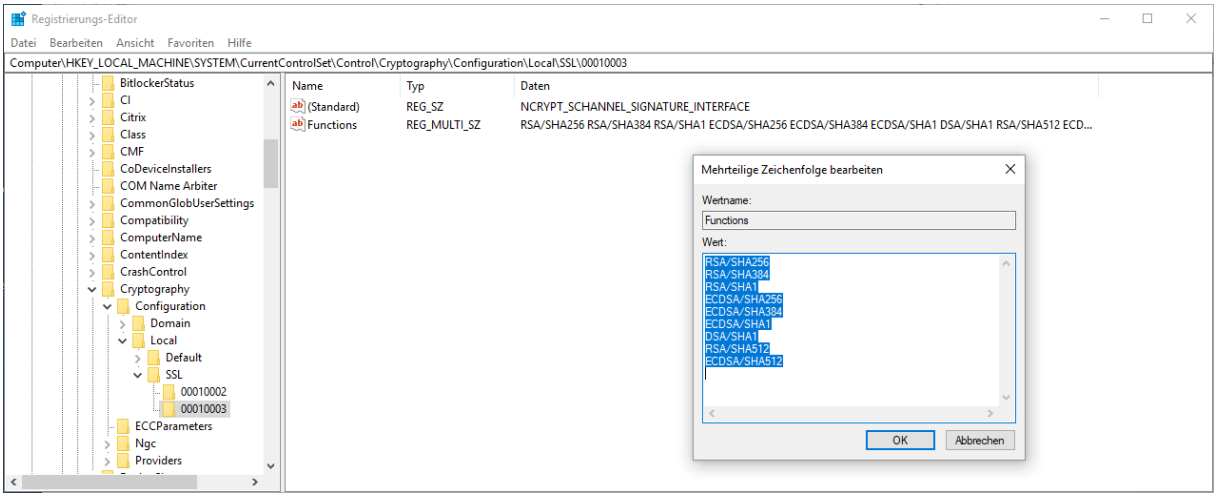
TLS1.3 bevorzugte Cipher Suiten unterstützt von OpenSSL:

TLS13-AES128-GCM-SHA256
TLS13-AES256-GCM-SHA384
TLS13-CHACHA20-POLY1305-SHA256
TLS13-AES128-CCM-SHA256
TLS13-AES128-CCM-8-SHA256

Hinweis am Rande:

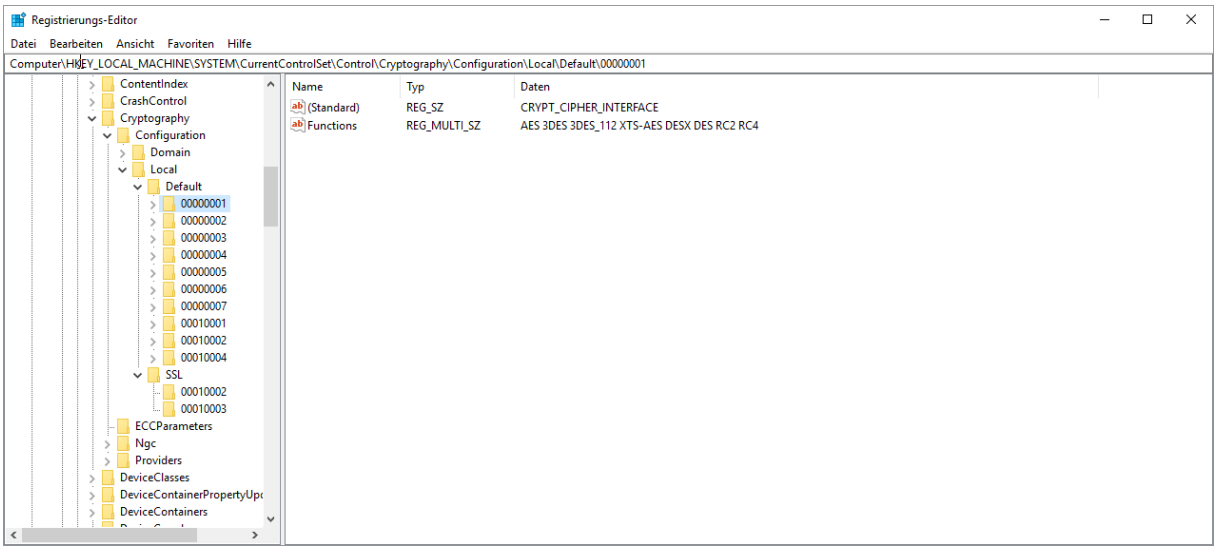
DSA Zertifikate sind in TLS1.3 nicht mehr unterstützt. Jeder Versuch einer Verbindung würde scheitern.

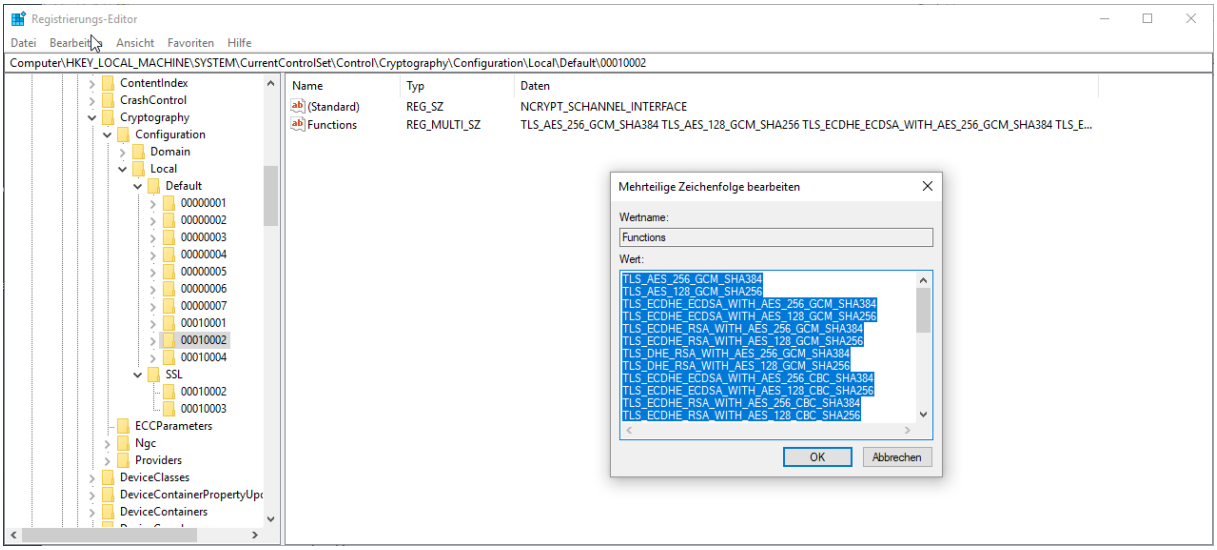
https://tools.ietf.org/html/draft-ietf-tls-tls13-20



ECCCurve Unterstützung Windows 10/2016

Windows 10 Default Werte:





ECCCurves löschen:

Certutil.exe –deleteEccCurve nistP256

ECCCurves Kontrolle:

CertUtil.exe –DisplayEccCurve

ECCCurves hinzufügen:

Certutil —addEccCurue nistP256 curveParameters 1.2.840.10045.3.1.7 23

Erstellt von Jörn Walter **03.03.2018**