Comparison of cryptography libraries

The tables below compare cryptography libraries that deal with cryptography algorithms and have API function calls to each of the supported features.

Contents

Cryptography libraries

Implementation	Company	Development Language	Open Source	Software License	FIPS 140 validated[1]	FIPS 140-2 mode	DO-178	Latest Update
Botan	Jack Lloyd	C++	Yes	Simplified BSD	No	No	No	2.12.1 (October 14, 2019[2]) [±] (https://en.wikipedia.org/w/index.php?title=Template:Latest_stable_software_release/Botan&action=edit)
Bouncy Castle	Legion of the Bouncy Castle Inc.	Java, C#	Yes	MIT License	Yes	Yes	No	**Java** 1.64 / October 7, 2019[3] **Java FIPS** BC-FJA 1.0.2 / August 24, 2019[4] **C#** 1.8.5 / January 31, 2019[5] **C# FIPS** BC-FNA 1.0.1 / December 28, 2016[6]
cryptlib	Peter Gutmann	C	Yes	Sleepycat License or commercial license	No[a]	Yes	No	3.4.5 (2019[7]) [±] (https://en.wikipedia.org/w/index.php?title=Template:Latest_stable_software_release/cryptlib&action=edit)
Crypto++	The Crypto++ project	C++	Yes	Boost Software License (all individual files are public domain)	No[b]	No	No	Feb 22, 2019 (8.1.0)
GnuTLS	Nikos Mavrogiannopoulos, Simon Josefsson	C	Yes	GNU LGPL v2.1+	Yes	Yes	No	3.6.8 (May 28, 2019[8]) [±] (https://en.wikipedia.org/w/index.php?title=Template:Latest_stable_software_release/Botan&action=edit)
Libgcrypt	GnuPG community and g10code	C	Yes	GNU LGPL v2.1+	Yes	Yes	No	1.8.5 (August 29, 2019[9]) [±] (https://en.wikipedia.org/w/index.php?title=Template:Latest_stable_software_release/Libgcrypt&action=edit) 1.7.10 (June 13, 2018[10]) [±] (https://en.wikipedia.org/w/index.php?title=Template:Latest_stable_software_release/Libgcrypt&action=edit)
libsodium	Frank Denis	C	Yes	ISC license	No	No	No	December 13, 2017 (1.0.16)
NaCl	Daniel J. Bernstein, Tanja Lange, Peter Schwabe	C	Yes	Public domain	No	No	No	February 21, 2011[11]
Nettle		C	Yes	GNU GPL v2+ or GNU LGPL v3	No	No	No	3.5.1 (June 27, 2019[12]) [±] (https://en.wikipedia.org/w/index.php?title=Template:Latest_stable_software_release/Nettle_(cryptographic_library)&action=edit)
Network Security Services (NSS)	Mozilla	C	Yes	MPL 2.0	Yes[13]	Yes	No	3.46 (August 30, 2019[14]) [±] (https://en.wikipedia.org/w/index.php?title=Template:Latest_stable_software_release/Network_Security_Services&action=edit)
OpenSSL	The OpenSSL Project	C	Yes	Apache Licence 1.0 and 4-Clause BSD Licence	Yes	Yes	No	1.1.1d (September 10, 2019[15]) [±] (https://en.wikipedia.org/w/index.php?title=Template:Latest_stable_software_release/OpenSSL&action=edit)
RSA BSAFE Crypto-C Micro Edition	RSA Security	C	No[c]	Proprietary	Yes	Yes	No	4.1.4 (September 11, 2019[16]) [±] (https://en.wikipedia.org/w/index.php?title=Template:Latest_s

								table_software_release/Comparison_of_cryptography_libraries&action=edit)
RSA BSAFE Crypto-J	RSA Security	Java	No[c]	Proprietary	Yes	Yes	No	6.2.5 (August 15, 2019[17]) [±] (https://en.wikipedia.org/w/index.php?title=Template:Latest_stable_software_release/Comparison_of_cryptography_libraries&action=edit)
wolfCrypt	wolfSSL, Inc.	C	Yes	GPL v2 or commercial license	Yes	Yes	Yes[d]	4.3.0 (December 20, 2019[18]) [±] (https://en.wikipedia.org/w/index.php?title=Template:Latest_stable_software_release/wolfSSL&action=edit)
mbed TLS	ARM Limited	C	Yes	Apache Licence 2.0	No	No	No	2.16.2 (June 11, 2019[19]) [±] (https://en.wikipedia.org/w/index.php?title=Template:Latest_stable_software_release/mbed_TLS&action=edit) 2.7.10 (March 19, 2018[19]) [±] (https://en.wikipedia.org/w/index.php?title=Template:Latest_stable_software_release/mbed_TLS&action=edit)

a. The actual cryptlib is not FIPS 140 validated, although a validation exists for an adapted cryptlib as part of a third party, proprietary, commercial product.
b. Crypto++ received three FIPS 140 validations from 2003 through 2008. In 2016 NIST moved Crypto++ to the Historical Validation List. The move effectively revokes the FIPS validation and federal agencies cannot use the module for validated cryptography.
c. RSA BSAFE source code license was available to purchase when RSA Security was selling BSAFE.
d. wolfCrypt has complete RTCA DO-178C level A certification. In addition, any of the FIPS 140-2 validated crypto algorithms can be used in DO-178 mode for combined FIPS 140-2/DO-178 consumption.

Key operations

Key operations include key generation algorithms, key exchange agreements and public key cryptography standards.

Key generation and exchange

Implementation	ECDH	DH	DSA	RSA	ElGamal	NTRU	DSS
Botan	Yes	Yes	Yes	Yes	Yes	No	Yes
Bouncy Castle	Yes	Yes	Yes	Yes	Yes	Yes	Yes
cryptlib	Yes	Yes	Yes	Yes	Yes	No	Yes
Crypto++	Yes	Yes	Yes	Yes	Yes	No	Yes
Libgcrypt	Yes[a]	Yes	Yes	Yes	Yes	No	Yes
libsodium	Yes	No	No	No	No	No	No
Nettle	No	No	Yes	Yes	No	No	No
OpenSSL	Yes	Yes	Yes	Yes	No	No	No
RSA BSAFE Crypto-C Micro Ediition	Yes	Yes	Yes	Yes	No	No	No
RSA BSAFE Crypto-J	Yes	Yes	Yes	Yes	No	No	No
wolfCrypt	Yes	Yes	Yes	Yes	No	Yes	Yes
mbed TLS	Yes	Yes	Yes	Yes	No	No	No

a. By using the lower level interface.

Elliptic curve cryptography (ECC) support

Implementation	NIST	SECG	ECC Brainpool	ECDSA	ECDH	Curve25519	EdDSA	GOST R 34.10 (https://tools.ietf.org/html/rfc7091)
Botan	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes
Bouncy Castle	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes
cryptlib	Yes	Yes	Yes	Yes	Yes	No	No	No
Crypto++	Yes	Yes	Yes	Yes	Yes	Yes	No	No
Libgcrypt	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes
libsodium	Yes	No	No	No	No	Yes	Yes	No
Nettle	Yes	Partial	No	No	No	Yes	Yes	No
OpenSSL	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes
RSA BSAFE Crypto-C Micro Edition	Yes	Yes	No	Yes	Yes	No	No	No
RSA BSAFE Crypto-J	Yes	Yes	No	Yes	Yes	No	No	No
wolfCrypt	Yes	No	Yes	Yes	Yes	Yes	Yes	No
mbed TLS	Yes	Yes	Yes	Yes	Yes	Yes	No	No

Public key cryptography standards

Implementation	PKCS#1	PKCS#5	PKCS#8	PKCS#12	IEEE P1363	ASN.1
Botan	Yes	Yes	Yes	No	Yes	Yes
Bouncy Castle	Yes	Yes	Yes	Yes	Yes	Yes
cryptlib	Yes	Yes	Yes	Yes	No	Yes
Crypto++	Yes	Yes	Yes[a]	No	Yes	Yes
Libgcrypt	Yes	Yes[b]	Yes[b]	Yes[b]	Yes[b]	Yes[b]
libsodium	No	No	No	No	No	No
Nettle	Yes	Yes	No	No	No	No
OpenSSL	Yes	Yes	Yes	Yes	No	Yes
RSA BSAFE Crypto-C Micro Edition	Yes	Yes	Yes	Yes	Yes	Yes
RSA BSAFE Crypto-J	Yes	Yes	Yes	Yes	No	Yes
wolfCrypt	Yes	Yes	Yes	Yes	No	Yes
mbed TLS	Yes	No	Yes	Yes	No	Yes

a. The library offers X.509 and PKCS #8 encoding without PEM by default. For PEM encoding of public and private keys the PEM Pack (https://www.cryptopp.com/wiki/PEM_Pack) is needed.
b. These Public Key Cryptographic Standards (PKCS) are supported by accompanying libraries and tools, which are also part of the GnuPG framework (https://gnupg.org/download/), although not by the actual libgcrypt library.

Hash functions

Comparison of supported cryptographic hash functions. At the moment this section also includes ciphers that are used for producing a MAC tag for a message. Here hash functions are defined as taking an arbitrary length message and producing a fixed size output that is virtually impossible to use for recreating the original message.

Implementation	MD5	SHA-1	SHA-2	SHA-3	RIPEMD-160	Tiger	Whirlpool	GOST	Stribog	BLAKE2
Botan	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes
Bouncy Castle	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes
cryptlib	Yes	Yes	Yes	Yes	Yes	No	Yes	No	No	No
Crypto++	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes	No	Yes
Libgcrypt	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes
libsodium	No	No	Yes	No	No	No	No	No	No	Yes
Nettle	Yes	Yes	Yes	Yes	Yes	No	No	Yes	No	No
OpenSSL	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes	No	Yes
RSA BSAFE Crypto-C Micro Edition	Yes	Yes	Yes	Yes	No	No	No	Yes	No	No
RSA BSAFE Crypto-J	Yes	Yes	Yes	Yes	Yes	No	No	No	No	No
wolfCrypt	Yes	Yes	Yes	Yes	Yes	No	No	No	No	Yes
mbed TLS	Yes	Yes	Yes	Yes	Yes	No	No	No	No	No

MAC algorithms

Comparison of implementations of message authentication code (MAC) algorithms. A MAC is a short piece of information used to authenticate a message—in other words, to confirm that the message came from the stated sender (its authenticity) and has not been changed in transit (its integrity).

Implementation	HMAC-MD5	HMAC-SHA1	HMAC-SHA2	Poly1305-AES	BLAKE2-MAC
Botan	Yes	Yes	Yes	Yes	Yes
Bouncy Castle	Yes	Yes	Yes	Yes	Yes
cryptlib	Yes	Yes	Yes	No	No
Crypto++	Yes	Yes	Yes	Yes	Yes
Libgcrypt	Yes	Yes	Yes	Yes	Yes
libsodium	No	No	Yes	Yes	Yes
Nettle	Yes	Yes	Yes	Yes	No
OpenSSL	Yes	Yes	Yes	Yes	No
RSA BSAFE Crypto-C Micro Edition	Yes	Yes	Yes	No	No
RSA BSAFE Crypto-J	Yes	Yes	Yes	Yes	No
wolfCrypt	Yes	Yes	Yes	Yes	Yes
mbed TLS	Yes	Yes	Yes	No	No

Block ciphers

Table compares implementations of block ciphers. Block ciphers are defined as being deterministic and operating on a set number of bits (termed a block) using a symmetric key. Each block cipher can be broken up into the possible key sizes and block cipher modes it can be run with.

Block cipher algorithms

Implementation	AES	Camellia	3DES	Blowfish	Twofish	CAST5	IDEA	GOST 28147-89 / GOST R 34.12-2015	ARIA
Botan	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes
Bouncy Castle[27]	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes
cryptlib[28]	Yes	No	Yes	Yes		Yes	Yes		
Crypto++	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes[a]	Yes
Libgcrypt	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes	
libsodium	Yes[b]	No	No	No	No	No	No	No	No
Nettle	Yes	Yes	Yes	Yes					
OpenSSL	Yes	Yes	Yes	Yes	No	Yes	Yes	Yes	Yes
RSA BSAFE Crypto-C Micro Edition	Yes	Yes	Yes	No	No	No	No	Yes[c]	Yes
RSA BSAFE Crypto-J	Yes	No	Yes	No	No	No	No	No	No
wolfCrypt	Yes	Yes	Yes	No	No	No	Yes	No	No
mbed TLS	Yes	Yes	Yes	Yes	No	No	No	No	No

a. Crypto++ provides the 64-bit version of GOST from the 1990s. The library does not provide the 128-bit version of GOST from 2015.
b. libsodium provides AES-256 only. It does not offer AES-128 or AES-192.
c. RSA BSAFE Micro Edition Suite only supports GOST 28147-89

Cipher modes

Implementation	ECB	CBC	OFB	CFB	CTR	CCM	GCM	OCB	XTS	AES-Wrap (https://tools.ietf.org/html/rfc3394)	Stream
Botan	No	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes
Bouncy Castle	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes		Yes	Yes
cryptlib	Yes	Yes	Yes	Yes		No	Yes				
Crypto++	Yes	Yes	Yes	Yes	Yes	Yes	Yes	No	No	No	Yes
Libgcrypt	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes
libsodium	No	No	No	No	Yes	No	Yes	No	No	No	No
Nettle	Yes	Yes	No	No	Yes	Yes	Yes	No	No	No	No
OpenSSL	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes
RSA BSAFE Crypto-C Micro Edition	Yes	Yes	Yes	Yes	Yes	Yes	Yes	No	Yes	Yes	Yes
RSA BSAFE Crypto-J	Yes	Yes	Yes	Yes	Yes	Yes	Yes	No	Yes	Yes	Yes
wolfCrypt	Yes	Yes	No	Yes	Yes	Yes	Yes	No	Yes	No	Yes
mbed TLS	Yes	Yes	No	Yes	Yes	Yes	Yes	No	No	No	No

Stream ciphers

Table compares implementations of the various stream ciphers. Stream ciphers are defined as using plain text digits that are combined with a pseudorandom cipher digit stream. Stream ciphers are typically faster than block ciphers and may have lower hardware complexity, but may be more susceptible to attacks.

Implementation	RC4	HC-256	Rabbit	Salsa20	ChaCha	SEAL	Panama	WAKE	Grain	VMPC	ISAAC
Botan	Yes	No	No	Yes	Yes	No	No	No	No	No	No
Bouncy Castle	Yes	Yes	No	Yes	Yes	No	No	No	Yes	Yes	Yes
cryptlib	Yes	No	No	No	No	No	No	No	No	No	No
Crypto++	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes	No	No	No
Libgcrypt	Yes	No	No	Yes	Yes	No	No	No	No	No	No
libsodium	No	No	No	Yes	Yes	No	No	No	No	No	No
Nettle	Yes	No	No	Yes	Yes	No	No	No	No	No	No
OpenSSL	Yes	No	No	No	Yes	No	No	No	No	No	No
RSA BSAFE Crypto-C Micro Edition	Yes	No	No	No	No	No	No	No	No	No	No
RSA BSAFE Crypto-J	Yes	No	No	No	Yes	No	No	No	No	No	No
wolfCrypt	Yes	Yes	Yes	Yes	Yes	No	No	No	No	No	No
mbed TLS	Yes	No	No	No	No	No	No	No	No	No	No

Hardware-assisted support

Table compares the ability to utilize hardware enhanced cryptography. With using the assistance of specific hardware the library can achieve greater speeds and / or improved security than otherwise.

Smartcard, SIM and HSM protocol support

Implementation	PKCS #11	PC/SC	CCID
Botan	Yes	No	No
Bouncy Castle	Yes [a]	No	No
cryptlib	Yes	No	No
Crypto++	No	No	No
Libgcrypt	Yes [32]	Yes [33]	Yes [34]
libsodium	No	No	No
OpenSSL	Yes [35]	No	No
RSA BSAFE Crypto-C Micro Edition	Yes	No	No
RSA BSAFE Crypto-J	Yes[b]	No	No
wolfCrypt	Yes	No	No
mbed TLS	Yes [36]	No	No

a. In conjunction with the PKCS#11 provider, or through the implementation of operator interfaces providing access to basic operations.
b. When using RSA BSAFE Crypto-J in native mode using RSA BSAFE Crypto-C Micro Edition.

General purpose CPU / platform acceleration support

Implementation	AES-NI	SSSE3 / SSE4.1	AVX / AVX2	RDRAND	VIA PadLock	Intel QuickAssist (http://www.intel.com/content/www/us/en/embedded/technology/quickassist/overview.html)	AltiVec[a]	ARMv7-A NEON	ARMv8-A
Botan	Yes	Yes	Yes	Yes	No	No	Yes	Yes	Yes
cryptlib	Yes	Yes	Yes	Yes	Yes	No	No	No	No
Crypto++	Yes	Yes	Yes	Yes	Yes[b]	No	Yes	Yes	Yes
Libgcrypt[39]	Yes	Yes	Yes	Yes	Yes	No	No	Yes	Yes
libsodium	Yes	Yes	Yes	No	No	No	No	No	No
OpenSSL	Yes	Yes	Yes	Yes[c]	Yes	No	Yes	Yes	Yes
RSA BSAFE Crypto-C Micro Edition	Yes	Yes	Yes	Yes	No	No	No	No	Yes
RSA BSAFE Crypto-J	Yes[d]	Yes[d]	Yes[d]	Yes[d]	No	No	No	No	Yes[d]
wolfCrypt	Yes	No	Yes	Yes	No	Yes[40]	No	No	Yes[41]

a. AltiVec includes POWER4 through POWER8 SIMD processing. POWER8 added in-core crypto, which provides accelerated AES, SHA and PMUL similar to SSE and ARMv8.1.

b. Crypto++ provides access to the Padlock random number generator. Other functions, like AES acceleration, is not provided.

c. OpenSSL RDRAND support is provided through the ENGINE interface. The RDRAND generator is not used by default.

d. When using RSA BSAFE Crypto-J in native mode using BSAFE Crypto-C Micro Edition

Microcontrollers' cryptographic accelerator support

Implementation	STM32F2 (http://www.st.com/web/en/catalog/mmc/FM141/SC1169/SS1575)	STM32F4 (http://www.st.com/en/microcontrollers/stm32f4-series.html?querycriteria=productId=SS1577)	Cavium NITROX (http://www.cavium.com/processor_security_nitroxPX.html)	Freescale CAU/mmCAU (http://www.freescale.com/webapp/sps/site/prod_summary.jsp?code=CAUAP)	Microchip PIC32MZ (https://www.microchip.com/pagehandler/en-us/technology/embeddedsecurity/technology/hardwarecryptoengine.html)	Atmel ATECC508A (http://www.microchip.com/wwwproducts/en/ATECC508A)	TI TivaC Series (https://web.archive.org/web/20170521001315/http://processors.wiki.ti.com/index.php/Using_wolfSSL_with_TI-RTOS)	CubeMX	Nordic nRF51
wolfCrypt	Yes	Yes	Yes	Yes	Yes	Yes[46]	Yes[47]	Yes	Yes

Code size and code to comment ratio

Implementation	Source Code Size (kSLOC = 1000 lines of source code)	Code Lines to Comment Lines Ratio
Botan	133[48]	4.55[48]
Bouncy Castle	1359[49]	5.26[49]
cryptlib	241	2.66
Crypto++	115[50]	5.74[50]
Libgcrypt	216[51]	6.27[51]
libsodium	44[52]	21.92[52]
Nettle	111[53]	4.08[53]
OpenSSL	472[54]	4.41[54]
RSA BSAFE Crypto-C Micro Edition	1117[a]	4.04[a]
RSA BSAFE Crypto-J	271[b]	1.3[b]
wolfCrypt	39	5.69
mbed TLS	105[55]	33.9[55]

a. Based on CCME 4.1.4, including tests source. Generated using https://github.com/XAMPPRocky/tokei
b. Based on Crypto-J 6.2.5, excluding tests source. Generated using https://github.com/XAMPPRocky/tokei

Portability

Implementation	Supported Operating System	Thread safe
Botan	Linux, Windows, macOS, Android, iOS, FreeBSD, NetBSD, OpenBSD, DragonflyBSD, AIX, QNX, Haiku, IncludeOS	Yes
Bouncy Castle	General Java API: J2ME, Java Runtime Environment 1.1+, Android. Java FIPS API: Java Runtime 1.5+, Android. C# API (General & FIPS): CLR 4.	
cryptlib	AMX, ARINC 653, BeOS, ChorusOS, CMSIS-RTOS/mbed-rtos, DOS, DOS32, eCOS, embOS, FreeRTOS/OpenRTOS, uItron, MQX, MVS, Nucleus, OS/2, Palm OS, QNX Neutrino, RTEMS, SMX, Tandem NonStop, Telit, ThreadX, uC/OS II, Unix (AIX, FreeBSD, HP-UX, Linux, macOS, Solaris, etc.), VDK, VM/CMS, VxWorks, Win16, Win32, Win64, WinCE/PocketPC/etc, XMK	Yes
Crypto++	Unix (AIX, OpenBSD, Linux, MacOS, Solaris, etc.), Win32, Win64, Android, iOS, ARM	Yes[a]
Libgcrypt	All 32 and 64 bit Unix Systems (GNU/Linux, FreeBSD, NetBSD, macOS etc.), Win32, Win64, WinCE and more	Yes[58]
libsodium	macOS, Linux, OpenBSD, NetBSD, FreeBSD, DragonflyBSD, Android, iOS, 32 and 64-bit Windows (Visual Studio, MinGW, C++ Builder), NativeClient, QNX, JavaScript, AIX, MINIX, Solaris	Yes
OpenSSL	Solaris, IRIX, HP-UX, MPE/iX, Tru64, Linux, Android, BSD (OpenBSD, NetBSD, FreeBSD, DragonflyBSD), NextSTEP, QNX, UnixWare, SCO, AIX, 32 and 64-bit Windows (Visual Studio, MinGW, UWIN, CygWin), UEFI, macOS (Darwin), iOS, HURD, VxWorks, uClinux, VMS, DJGPP (DOS), Haiku	Yes
RSA BSAFE Crypto-C Micro Edition	Solaris, HP-UX, Tru64, Linux, Android, FreeBSD, AIX, 32 and 64-bit Windows (Visual Studio), macOS (Darwin), iOS, VxWorks	Yes
RSA BSAFE Crypto-J	Solaris, Linux, Android, FreeBSD, AIX, 32 and 64-bit Windows, macOS (Darwin)	Yes
wolfCrypt	Win32/64, Linux, macOS, Solaris, ThreadX, VxWorks, FreeBSD, NetBSD, OpenBSD, embedded Linux, WinCE, Haiku, OpenWRT, iPhone (iOS), Android, Nintendo Wii and Gamecube through DevKitPro, QNX, MontaVista, NonStop, TRON/ITRON/µITRON, Micrium's µC/OS, FreeRTOS, SafeRTOS, Freescale MQX, Nucleus, TinyOS, HP-UX	Yes
mbed TLS	Win32/64, Unix Systems, embedded Linux, Micrium's µC/OS, FreeRTOS	?

a. Crypto++ is thread safe at the object level, i.e. there is no shared data among instances. If two different threads access the same object then the user is responsible for locking.

References

1. Validated FIPS 140 Cryptographic Modules (http://csrc.nist.gov/groups/STM/cmvp/documents/140-1/140val-all.htm) Archived (https://web.archive.org/web/20141226152243/http://csrc.nist.gov/groups/STM/cmvp/documents/140-1/140val-all.htm) 2014-12-26 at the Wayback Machine, NIST.gov, retrieved 2015-12-22

2. "Botan: Newslog" (https://botan.randombit.net/news.html). Retrieved 2019-08-04.
3. "Latest Java Releases - bouncycastle.org" (https://www.bouncycastle.org/latest_releases.html). 2019-10-07. Retrieved 2019-10-08.
4. "Java FIPS Resources - bouncycastle.org" (https://www.bouncycastle.org/fips-java/). 2019-08-24. Retrieved 2019-08-29.
5. "The Legion of the Bouncy Castle C# Cryptography APIs" (https://www.bouncycastle.org/csharp/). 2019-01-31. Retrieved 2019-03-17.
6. "C# .NET FIPS Resources - bouncycastle.org" (https://www.bouncycastle.org/fips-csharp/). 2016-11-11. Retrieved 2017-08-28.
7. Gutmann, Peter (2019). "Downloading" (https://www.cs.auckland.ac.nz/~pgut001/cryptlib/download.html). cryptlib. *University of Auckland School of Computer Science*. Retrieved 2019-08-07.
8. "The GnuTLS Transport Layer Security Library" (https://www.gnutls.org/news.html). Retrieved 4 June 2019.
9. "Release 1.8.5" (https://lists.gnupg.org/pipermail/gnupg-announce/2019q3/000440.html). *dev.gnupg.org*. 2019-08-29. Retrieved 2019-08-29.
10. "Release 1.7.10" (https://lists.gnupg.org/pipermail/gnupg-announce/2018q2/000426.html). *dev.gnupg.org*. 2018-06-13. Retrieved 2018-06-13.
11. Downloading and installing NaCl (https://nacl.cr.yp.to/install.html), Bernstein, Lange, Schwabe, retrieved 2017-05-22
12. "Nettle ChangeLog file @ git tag nettle_3.5.1_release_20190627" (https://git.lysator.liu.se/nettle/nettle/blob/nettle_3.5.1_release_20190627/ChangeLog).
13. "FIPS" (https://web.archive.org/web/20130502054951/https://www.mozilla.org/projects/security/pki/nss/fips/). Mozilla Foundation. 2012-02-01. Archived from the original (https://www.mozilla.org/projects/security/pki/nss/fips/) on 2013-05-02. Retrieved 2013-05-17.
14. "NSS Changelog" (https://hg.mozilla.org/projects/nss/log). Retrieved 2019-09-04.
15. "OpenSSL: Newslog" (https://www.openssl.org/news/newslog.html). Retrieved 2019-09-11.
16. "RSA announces the release of RSA BSAFE® Crypto-C Micro Edition 4.1.4" (https://community.rsa.com/docs/DOC-107001).
17. "RSA announces the release of RSA BSAFE® Crypto-J 6.2.5" (https://community.rsa.com/docs/DOC-106557).
18. "wolfSSL ChangeLog" (https://www.wolfssl.com/docs/wolfssl-changelog/). 2019-12-20. Retrieved 2019-12-20.
19. "Mbed TLS 2.16.0, 2.7.9 and 2.1.18 released" (https://tls.mbed.org/tech-updates/releases/mbedtls-2.16.0-2.7.9-and-2.1.18-released). 2018-12-21. Retrieved 2018-03-24.
20. Bouncy Castle Specifications (https://www.bouncycastle.org/specifications.html), bouncycastle.org, retrieved 2018-04-10
21. cryptlib Encryption Toolkit (https://www.cs.auckland.ac.nz/~pgut001/cryptlib/), Peter Gutmann, retrieved 2015-11-28
22. With Scute (http://www.scute.org/scute.html/Overview.html), scute.org
23. With GnuPG's SCdaemon (https://www.gnupg.org/documentation/manuals/gnupg/Scdaemon-Options.html) & gpg-agent, gnupg.org
24. With GnuPG's SCdaemon (https://www.gnupg.org/documentation/manuals/gnupg/Scdaemon-Options.html) & gpg-agent, gnupg.org
25. With an libp11 (https://github.com/OpenSC/libp11) engine
26. With an libp11 (https://github.com/OpenSC/libp11) engine
27. hwfeatures.c (https://dev.gnupg.org/source/libgcrypt/browse/master/src/hwfeatures.c), dev.gnupg.org
28. https://www.wolfssl.com/wolfSSL/Blog/Entries/2017/1/18_wolfSSL_Asynchronous_Intel_QuickAssist_Support.html
29. https://www.wolfssl.com/wolfSSL/Blog/Entries/2016/10/13_wolfSSL_ARMv8_Support.html
30. https://www.wolfssl.com/wolfSSL/wolfssl-atmel.html
31. "Archived copy" (https://web.archive.org/web/20170521001315/http://processors.wiki.ti.com/index.php/Using_wolfSSL_with_TI-RTOS). Archived from the original (http://processors.wiki.ti.com/index.php/Using_wolfSSL_with_TI-RTOS) on 2017-05-21. Retrieved 2017-05-01.
32. Language Analysis of Botan (https://www.openhub.net/p/botan/analyses/latest/languages_summary), OpenHub.net, retrieved 2018-07-18
33. Language Analysis of Bouncy Castle (https://www.openhub.net/p/5523/analyses/latest/languages_summary), OpenHub.net, retrieved 2015-12-23
34. Language Analysis of Crypto++ (https://www.openhub.net/p/3522/analyses/latest/languages_summary), OpenHub.net, retrieved 2018-07-18
35. Language Analysis of Libgcrypt (https://www.openhub.net/p/libgcrypt/analyses/latest/languages_summary), OpenHub.net, retrieved 2015-12-23
36. Language Analysis of libsodium (https://www.openhub.net/p/libsodium/analyses/latest/languages_summary), OpenHub.net, retrieved 2017-05-07
37. Language Analysis of Nettle (https://www.openhub.net/p/nettle/analyses/latest/languages_summary), OpenHub.net, retrieved 2015-12-23
38. Language Analysis of OpenSSL (https://www.openhub.net/p/openssl/analyses/latest/languages_summary), OpenHub.net, retrieved 2017-05-07

39. Language Analysis of mbed-tls (https://www.openhub.net/p/mbed-tls/analyses/latest/languages_summary), OpenHub.net, retrieved 2019-09-15
40. GnuPG documentation: Libgcrypt overview - thread safety (https://gnupg.org/documentation/manuals/gcrypt-devel/Overview.html), GnuPG.org, retrieved 2016-04-16