Overview of selected KPN Security Policies

Creation date: Thursday, May 9, 2019 3:52:36 PM

Selected by: Ruud Leurs

Requirement	Use of untrusted certificates
Description	The use of untrusted certificates is not allowed. Known good CAs or alterations are vetted by CISO. Untrusted certificates are: Self-signed certificates, i.e. certificates which have self-vetted and self-validated their own information and key material by signing itself. Certificates signed by an untrusted, unknown or vendor supplied CA, i.e. certificates which have not been vetted and validated by an open or known process. Certificates using key material not generated nor controlled by KPN. Exception: Exclusively for Puppet and VMWare VMCA it can provide and apply their own certificates. These certificates are explicitly scoped to the protocol used for inter-VMWare cluster or Puppet Master-Agent communication. Any other application for these certificates is prohibited.
Supplement	Known good CAs are: For internal trust: All certificates (ultimately) signed by the KPN Private Root CA. via Service Now: "Request private KPN CA SSL certificate (internal domain) - 510.351" All certificates (ultimately) signed by the KPN Workspace Root CA. Per project CA solutions vetted by CISO. For public/external trust: KPN PKI Overheid G2, G3, EV, or other Staat der Nederlanden Root CA signed CA. via Service Now: "New SSL certificate subscription (external domain) - 510.349" GlobalSign CAs

	Comodo CAs DigiCert CAs Know abusive to KPN, due to being used in phishing campagnes and other cyber-crime related activities and therefor blocked for use: Let's Encrypt Amazon CA
ID	KSP-RE-490
Version	1.2
Date	February 1, 2019
Rationale	Cryptography generic

Requirement	**Choosing safe curves for elliptic curve cryptography**
Description	The use of safe elliptic curves is mandatory. Specific elliptic curves are regarded as safe after having passed (cryptographic) peer review. Applications can use the following safe curves: - M-221 (Curve2213) - E-222 - Curve1174 - Curve25519 - E-382 - M-383 - Curve383187 - Curve41417 - Ed448-Goldilocks - M-511 - E-521 Exception: If no safe curves are supported, the following elliptic curves are acceptable for usage: - P-256 - P-384 - P-521
Related info	SafeCurves: http://safecurves.cr.yp.to/
ID	KSP-RE-491
Version	1.0
Date	December 11, 2017
Rationale	Cryptography generic

Requirement	Key Compromise
Description	Compromised keys must be regenerated and rekeyed, not updated. During generation the new key must be generated from a new set of data (no re-use of data used to generate the compromised key) to ensure its full independence from the compromised key. For PKI the CA must be informed of the compromise by means of the contract manager. Example keys involved are: • SSH private keys for hosts or users • Private keys associated to PKI, PGP and other types of certificates • Diffie-Hellman param files • Group keys • Key used for symmetric encryption of e.g. files, databases, file-systems or any other type of arbitrary data
ID	KSP-RE-470
Version	1.0
Date	December 11, 2017
Rationale	Cryptography generic

Requirement	**Password hashing**
Description	Passwords must be hashed and stored using known good salted password hashing methods. The following list of actions must be taken for each password from the service/tooling: • Use a good random salt, see KSP-RE-485 • Use a known good hash algorithm, see KSP-RE-483 • Use a random salt per password • In client/server scenarios, like web applications, always hash on the server side • To make cracking harder use key stretching to protect the passwords Exception for high-volume environments where key stretching is not applicable for performance reasons: use HMAC to protect the passwords with a key per password stored securely in an HSM solution.
ID	KSP-RE-492
Version	1.0
Date	December 11, 2017
Rationale	Cryptography generic
Rationale	Authentication

Requirement	**Private key transport and storage**
Description	Private keys are one of the foundations for the security of a service and its data. A private key must be protected during both transport and its storage: Storage: - The private keys should be stored securely in an Hardware Security Module when multiple customer keys could be compromised during a service breach. - Keys stored on a file system must be protected with the most strict possible file system permissions. - Physical security steps must be taken to limit access to the key to authorized personnel. Any form of physical security in addition to building access, that allows verification of access (see point below) will do. - If a stored key is accessed this must be verifiable/detectable. Transport: Before transporting a private key between systems the private key must be encrypted and use message integrity rules to provide tamper resistance. For key encryption and integrity the following rules are mandatory requirements: • KSP-RE-479 (Encryption Algorithms) • KSP-RE-483 (Hash Algorithms) • KSP-RE-228 (Password length) – for static passwords • KSP-RE-229 (Password complexity) In addition: • The transport method must be encrypted itself, e.g. use SSH, HTTPS or FTPS. • Use HMAC or Digital Signatures to authenticate the sender of a private key when the sender and receiver are different entities.
ID	KSP-RE-471
Version	1.1
Date	April 4, 2018
Rationale	Cryptography generic

Requirement	**Key stretching algorithms**
Description	Apply known good key-stretching algorithms: • PBKDF2, when FIPS certification or enterprise support on many platforms is required. Only use in combination with hash algorithms and a salt as mentioned in this document. o On mobile devices § Minimum: 5.000 rounds § Norm: 10.000 rounds o On servers and workstations: § Minimum: 50.000 rounds § Norm: 100.000 rounds • Scrypt, where resisting any/all hardware accelerated attacks is necessary but support isn’t. o On mobile devices § Norm: N = 2^14, r = 8, p = 1 o On servers and workstations: § Norm: N = 2^20,r = 8, p = 1 • Bcrypt, where PBKDF2 or scrypt support is not available o On mobile devices § Norm: cost = 13 o On servers and workstations: § Norm: cost = 16
ID	KSP-RE-493
Version	1.0
Date	December 11, 2017
Rationale	Cryptography generic

Requirement	**Public Key Exchange**
Description	To authenticate a service, host, machine or user a public key must be exchanged to the peer using a key exchange method listed in KSP-GL-514 document. Proper key exchange methods prevent identity spoofing by enabling the peer to verify the authenticity of the public key and challenge the ownership of the private key.
Related info	Wikipedia: Key Exchange (Engels) (http://en.wikipedia.org/wiki/Key_exchange)
ID	KSP-RE-472
Version	1.0
Date	December 11, 2017
Rationale	Cryptography generic

Requirement	**Key destruction**
Description	Cryptographic keys must be destroyed in such a way that restoration is impossible. This procedure must take platform specific properties into account, like removal of a file does not implicitly wipe the key from the disk nor does it implicitly nullify the data in memory.
ID	KSP-RE-494
Version	1.0
Date	December 11, 2017
Rationale	Cryptography generic

Requirement	**Certificate Authority**
Description	Public Key Infrastructure builds trust relationships using trusted third parties, the Certificate Authorities. All used Certificate Authorities: - Must comply with the European Telecommunications Standards Institute (ETSI) standard "ETSI TS 101 456”. - Are FIPS 140-2 level 3 compliant or better. - Have a published CPS (Certification Practice Statement), this also means that our use of the certificate must follow the CPS.
Related info	ETSI TS 101 456: Electronic Signatures and Infrastructures (ESI); Policy requirements for certification authorities issuing qualified certificates (http://www.etsi.org/deliver/etsi_ts/101400_101499/101456/01.04.03_60/ts_101456v010403p.pdf) FIPS 140-2: Security Requirements for Cryptographic Modules (http://csrc.nist.gov/publications/fips/fips140-2/fips1402.pdf)
ID	KSP-RE-473
Version	1.0
Date	December 11, 2017
Rationale	Cryptography generic

Requirement	Certificates
Description	Certificates in the format X.509 can be used in a Public Key Infrastructure or web-of-trust. In a Public Key Infrastructure (PKI) context the follow applies: - Certificates can identify hosts, servers, users, processes, end-points and (individual) products. - Certificates comply to RFC5280. - Domain validation when used for identification. - Revocation must be implemented using CRLs (RFC5280), OCSP (RFC2560, RFC5019 or RFC6990) or OCSP stapling (RFC6066 and RFC6961), unless the clients are fully enclosed, i.e. no direct or proxied internet access exists. In web-of-trust context the following applies: - The certificates must comply to PGP/GPG standard, i.e. RFC 4880 and additional RFCs. - The certificates are created for a group or individual. - The certificates are digitally signed by others or a master key, after the full fingerprint is compared and matches. PGP/GPG keys which have been used publicly must be invalidated. The invalidation must be published.
Related info	RFC 5280: Internet X.509 Public Key Infrastructure Certificate and Certificate Revocation List (CRL) Profile
ID	KSP-RE-474
Version	1.0
Date	December 11, 2017
Rationale	Cryptography generic

Requirement	**Use of certificates**
Description	The certificate and the applications in which they are used must support the relevant RFC extensions describing use of certificates in combination with application or transport protocols. For instance RFC2818 to bind the identity of a peer to a session.
Related info	RFC 6125: Representation and Verification of Domain-Based Application Service Identity within Internet Public Key Infrastructure Using X.509 (PKIX) Certificates in the Context of Transport Layer Security (TLS)
ID	KSP-RE-475
Version	1.0
Date	December 11, 2017
Rationale	Cryptography generic

Requirement	**Binding Certificates**
Description	Each certificate must be bound to use for an as small as possible set of identities, example one host, virtual machine, one service, person or department. An TLS off-loader or load-balancer MAY hold the certificate and private key to serve/off-load the TLS sessions for one cluster of nodes serving the same service.
ID	KSP-RE-476
Version	1.1
Date	April 4, 2018
Rationale	Cryptography generic

Requirement	**Wildcard certificates**
Description	Wildcard certificates must be scoped as much as possible to the most specific Fully Qualified Domain Name (FQDN). To limit impact on the infrastructure in case of compromise the use of wildcard certificate is: - limited to a single application-type and purpose, i.e. exclusively for a cluster of mail servers or web-servers only, or another specific application-type. - must be scoped to the most specific FQDN possible in a cluster of systems or load-balanced setup, i.e. *.service.domain.tld. For non-KPN infrastructure it is permitted to ommit this restriction. - may use a scheme customer_name.service.domain.tld or customer_name.specific_domain.tld. - must not use on a widely used domain, e.g. *.kpn.com, *.kpnnet.org, *kpn.net, *.kpn.nl, *.kpn, *.kpnmail.nl or *.any_subsidiarymaindomain.tld. - must secure the private key according to KSP-RE-471. - For environments with a high demand on integrity and confidentiality the use of tamperproof solutions must be assessed. E.g. an Hardware Security Module (HSM) with a certification of FIPS140-2 level 2 or better in the context of health sector related services and vital infrastructure.
ID	KSP-RE-477
Version	1.0
Date	December 11, 2017
Rationale	Cryptography generic

Requirement	**Key pair lifetimes**
Description	Keys used must have a maximum lifetime of 36 months. Examples of keys are: - Keys belonging to certificates - Diffie-Hellman keys - Static passwords - pre-shared keys (PSK) - master keys - SSH keys for systems and administrators - PGP keys Exception to this is the key pairs used by a Certificate Authority.
Supplement	Password used with these key pairs need to be compliant with KSP-RE-230. Because of the maximum lifetime of the key pairs these password don't need to change during this lifetime.
Related info	Baseline Requirements Certificate Policy for the Issuance and Management of Publicly-Trusted Certificates (current: 1.3.4)
ID	KSP-RE-478
Version	1.1
Date	April 4, 2018
Rationale	Cryptography generic

Requirement	Encryption Algorithms
Description	One of the following encryption primitives must be used for encryption and decryption: AES-256, AES-192 and AES-128 XSalsa20/20 Salsa20/20 Camellia Twofish IDEA; the key must be generated using a hash algorithm from KSP-RE-483, like SHA2 For AES and Camellia use known good authenticated modes: GCM CCM Use non-authenticated cipher modes only in combination with an authentication method, like HMAC: CTR XTS The following encryption primitives should not be used. Use only for legacy support or explicit compatibility requirements: AES-256-CBC, AES-192-CBC and AES-128-CBC in combination with TLSv1.0 Camellia-CBC in combination with TLSv1.0 Three-key Triple DES Blowfish All not explicitly mentioned encryption algorithms are not allowed. Example are: RC4 All EXPORT ciphers All encryption algorithms resulting in less than 112 security bits The use of a random nonce or initialisation vector (IV) with sufficient length is mandatory with each of these encryption algorithms. To generate a good nonce or IV use a good random bit generator.

Related info	NIST Special Publication 800-131Ar1: Transitions: Recommendation for Transitioning the Use of Cryptographic Algorithms and Key Lengths
ID	KSP-RE-479
Version	1.2
Date	February 1, 2019
Rationale	Cryptography generic

Requirement	**Digital Signatures Algorithms**
Description	One of the following digital signature algorithms must be used: - CECPQ1-ECDSA - EdDSA - ECDSA - RSA - DSA For Post Quantum resistance, the following algorithms must be used: - CECPQ1-ECDSA (New Hope) - NTRU-6130 (Lattice-based) - McEliece or Goppa-based McEliece - SPHINCS-256 (hash based signatures) These algorithms can be used in authentication phases or integrity checks.
Related info	NIST Special Publication 800-131Ar1: Transitions: Recommendation for Transitioning the Use of Cryptographic Algorithms and Key Lengths
ID	KSP-RE-480
Version	1.0
Date	December 11, 2017
Rationale	Cryptography generic

Requirement	**Digital Signature Generation and Verification**
Description	Digital signatures must have at least 112 bits of security strength. This means: - For EC: key length $\geq$ 224 - For RSA: key length $\geq$ 2048 - For DSA: key length 3072/256 or 4096/256.
Related info	NIST Special Publication 800-131Ar1: Transitions: Recommendation for Transitioning the Use of Cryptographic Algorithms and Key Lengths
ID	KSP-RE-481
Version	1.0
Date	December 11, 2017
Rationale	Cryptography generic

Requirement	**Key exchange**
Description	For key exchange one of the following must be used: ECDH (Diffie-Hellman) with a minimal key length of 256 bits; DH (Diffie-Hellman) with a minimal key length of 2048 bits. All ECDH and DH parameters must be newly generated before use to assure unicity and avoid default parameters reuse between various installations.
Related info	NIST Special Publication 800-131Ar1: Transitions: Recommendation for Transitioning the Use of Cryptographic Algorithms and Key Lengths
ID	KSP-RE-482
Version	1.1
Date	August 16, 2018
Rationale	Cryptography generic

Requirement	Hash Algorithms
Description	One of the following hash algorithms must be used: SHA-2: SHA-512, SHA-384 or SHA-256 SHA-3 GOST R 34.11-94 (256 bit hash) Skein JH Grøstl BLAKE and BLAKE2 RIPEMD160 The following hash algorithms should not be used. Use only for legacy support or explicit compatibility requirements: SHA-1: for Non-digital signature generation applications only, not for Digital signature verification nor Digital signature generation after 2013 SHA-224: for Non-digital signature generation applications only, not for Digital signature verification nor Digital signature generation after 2014 WIRLPOOL-T HAVAL, using >= 160 bit with 3 rounds The following hash algorithms must not be used: SHA-0 HAVAL, using 128 bit with 3 rounds RIPEMD MD5 MD4 MD2 Exception: the use of MS-CHAPv2 is allowed, when transported as payload within an encrypted protocol, like TLS or various EAP protocols using a TLS based outer layer.
Related info	NIST Special Publication 800-131Ar1: Transitions: Recommendation for Transitioning the Use of Cryptographic Algorithms and Key Lengths
ID	KSP-RE-483

Version	1.1
Date	February 1, 2019
Rationale	Cryptography generic

Requirement	**HMAC (Hash-based Message Authentication Code)**
Description	HMAC is a keyed-hash message authentication code and must use: - A hash algorithm as defined in KSP-RE-483 - A key with a length $\geq$ 128 bits - The key should be generated using a known good random bit generator Known good examples are: - HMAC-SHA1 with a key length of 160 bit. - HMAC-SHA2 with a key length of 256 bit. - HMAC-SHA3 with a key length of 256 bit. HMAC-MD5 must not be used.
Related info	http://csrc.nist.gov/publications/nistpubs/800-107-rev1/sp800-107-rev1.pdf
ID	KSP-RE-484
Version	1.0
Date	December 11, 2017
Rationale	Cryptography generic

Requirement	SALT use
Description	The length of the randomly-generated portion of the salt must be at least 128 bits. The salt must be generated using a known good random bit generator. Example uses for a salt: • KSP-RE-236 (Password storage) • KSP-RE-479 (Encryption Algorithms) • KSP-RE-492 (Password hashing) • KSP-RE-493 (Key stretching algorithms)
ID	KSP-RE-485
Version	1.0
Date	December 11, 2017
Rationale	Cryptography generic

Requirement	**Maximum token lifetime**
Description	Authentication tickets/tokens, e.g. Kerberos, AFS and Windows logon, must have a maximum lifetime of 6 hours. During their period of validity tokens may be refreshed automatically.
ID	KSP-RE-486
Version	1.0
Date	December 11, 2017
Rationale	Cryptography generic

Requirement	**Web application data encryption**
Description	For encryption of transported application data applications: The highest available TLS version must be enabled. Protection against downgrade attacks must be activated. When this feature is absent: TLSv1.0 must be de-activated. TLSv1.3 must be enabled, when available. TLSv1.2 must be enabled. TLSv1.1 may only be enabled when there is a need to be able to communicate with legacy systems. When this need is absent, it must be disabled. TLSv1.0 may only be enabled when there is a need to be able to communicate with legacy systems. When this need is absent, it must be disabled. SSLv3 is not allowed to be enabled and must be completely disabled. SSLv2 is not allowed to be enabled and must be completely disabled. Enterprise TLS, as standardised by ETSI, is not allowed and must be completely disabled.
Related info	TLS 1.2 standard TLS 1.3 standard Enterprise TLS standard
ID	KSP-RE-487
Version	1.2
Date	May 3, 2019
Rationale	Cryptography generic

Requirement	**Cryptographic Key Generation, Random Bit Generator**
Description	Use a known good entropy source to generate cryptographic keys, identifiers or random seeds. Known good entropy sources for an application combine several random sources and use a cryptographic secure hash algorithm over the values of the sources. Known good sources are: On Apple iOS use SecRandomCopyBytes On Android use java.security.SecureRandom and must not be combined with setSeed(). On Unix and Linux systems use /dev/urandom On Windows use CryptGenRandom or RtlGenRandom In .Net use System.Security.Cryptography.RNGCryptoServiceProvider In Java use java.security.SecureRandom In Perl use Math::Random::Secure In PHP use random_bytes, or openssl_random_pseudo_bytes In Python use os.urandom In Ruby use SecureRandom At (re)boot time of a system, when there isn't sufficient entropy gathered yet, all processes generating new key material must block until sufficient entropy has been gathered. Random bit generators must be compliant with one of the following standards: SP 800-90A, revision 1. ANSI X9.62:2005, Annex D. The use of the following methods or entropy sources are forbidden: EC_Dual_DRBG
Related info	NIST Special Publication 800-90A (revision 1): Recommendation for Random Number Generation Using Deterministic Random Bit Generators
ID	KSP-RE-466
Version	1.1
Date	May 3, 2019
Rationale	Cryptography generic

Requirement	**Use Perfect Forward Secrecy**
Description	Perfect Forward Secrecy must be used when setting up encrypted connections with any of the following protocols: - IPSEC (Internet Protocol Security) met Group 14 (or better) - SSH (Secure Shell) - TLS (Transport Layer Security) - OTR (Off-The-Record messaging for instant messaging) Non-perfect forward secrecy protocols are allowed for legacy support and compatibility only. TLS cipher suite configuration should explicitly prefer ECDHE and DHE/EDH cipher suites above other cipher suites.
Related info	http://nl.wikipedia.org/wiki/Perfect_forward_secrecy
ID	KSP-RE-488
Version	1.1
Date	April 4, 2018
Rationale	Cryptography generic

Requirement	**Cryptographic Key Generation, Cryptographic Module**
Description	For high-security services where the entropy source needs to be protected from tampering a cryptographic hardware module must be used. The Cryptography Module of the product used must be compliant with the FIPS-140-2 level 2 standard.
Related info	FIPS PUB 140-2: Security Requirements for Cryptographic Modules
ID	KSP-RE-467
Version	1.0
Date	December 11, 2017
Rationale	Cryptography generic

Requirement	**Use of multi-domain certificates**
Description	Certificates must be scoped to only one application. The application may use multiple FQDNs (Fully Qualified Domain Names) to be identified. The FQDNs must share the same domain name. Example: • "www.kpn.com" and "kpn.com" can be combined • "www.kpn.com" and "kpninternational.com" cannot be combined • "reporting.kpn.com" and "www.kpn.com" and "kpn.com" may be combined in one certificate when the "reporting" hostname is explicitly part of the overall application.
ID	KSP-RE-489
Version	1.0
Date	December 11, 2017
Rationale	Cryptography generic

Requirement	**Registration of Key Pair properties**
Description	For each public/private key pair the following must be registered: - The owner - The intended use (infrastructure on which deployed) - Key length - Key Algorithm (including curve if Elliptic Curve is used) - Hash function - CA used for signing - Serial number (if applicable, like for certificates) - Validity from and to dates Registration may be omitted when the certificates are ordered through the central certificate application process. This rule is implicitly satisfied when the certificates are ordered via internal processes.
ID	KSP-RE-468
Version	1.0
Date	December 11, 2017
Rationale	Cryptography generic

Requirement	**Key Pair privacy**
Description	The private part of the key pair should be generated on the device on which it will be used. To support this: - Certificate signing request must be submitted by CSR (Certificate Signing Request). - Alternatively, key pairs must be generated locally by the key-pair owner or a delegated party within KPN. - For load balancing purposes, it is allowed to copy the private key into multiple devices.
Related info	Wikipedia: Certificate Signing Request (http://en.wikipedia.org/wiki/Certificate_signing_request)
ID	KSP-RE-469
Version	1.0
Date	December 11, 2017
Rationale	Cryptography generic

Requirement	**Certificate pinning and mutual authentication**
Description	Mobile apps which form a cryptographic foundation by implementing key provisioning for other services must apply certificate pinning or implement mutual authentication based on PKI with mutual validation. All other applications must comply to specifically KSP-RE-410. Note: The pinning method must guarantee business continuity, e.g. to allow for multiple public keys to be pinned. Also allowed is to pin on the intermediate CA.
Supplement	By following the specifications for TLS and PKI as specified in the KSP attackers will be limited to: 1. Compromising the endpoint, e.g. mobile device or computer system; 2. Attack the digitale signature scheme as the foundation to PKI; 3. Compromising a CA, i.e. DigiNotar; 4. Create certificate at another CA. The first vector already compromised the endpoint severely. The second requires a mistake in digital signature techniques or a quantum computer with sufficient qubits and the third requires misconduct or infiltration into a CA, whereafter the expectation is that the business for this CA will stop. The fourth problem is mitigated/lowered in risk by applying CAA records on all KPN domains. See KSP-RE-435 for requirements on CAA records.
ID	KSP-RE-344
Version	1.3
Date	February 1, 2019
Rationale	Web-based and other application software

Requirement	**Safe key lengths for several algorithms**
Description	For every cryptographic methods is a safe key length. These can be found in the tab "Safe Key Lengths" in the guideline KSP-GL-514 - Cryptographic algorithms and cipher suites.
ID	KSP-RE-697
Version	1.0
Date	April 4, 2018
Rationale	Cryptography generic