Security Reference Update

Security > Carbon

2007-07-18

Apple Inc.
1 Infinite Loop
Cupertino, CA 95014
408-996-1010

Simultaneously published in the United States and Canada.

Contents

Introduction to Security Reference Update

This document summarizes the symbols that have been added to the Security framework. The full reference documentation notes in what version a symbol was introduced, but sometimes it's useful to see only the new symbols for a given release.

If you are not familiar with this framework you should refer to the complete framework reference documentation.

Organization of This Document

Symbols are grouped by class or protocol for Objective-C and by header file for C. For each symbol there is a link to complete documentation, if available, and a brief description, if available.

See Also

For reference documentation on this framework, see *Security Framework Reference*

10.5 Symbol Changes

This article lists the symbols added to `Security.framework` in Mac OS X v10.5.

C Symbols

All of the header files with new symbols are listed alphabetically, with their new symbols described.

AuthorizationPlugin.h

Data Types & Constants

All of the new data types and constants in this header file are listed alphabetically, with links to documentation and abstracts, if available.

`kAuthorizationContextFlagSticky`	This data persists through an interrupted or failed evaluation.

CMSDecoder.h

Functions

All of the new functions in this header file are listed alphabetically, with links to documentation and abstracts, if available.

`CMSDecoderCopyAllCerts`	Obtain an array of all of the certificates in a message.
`CMSDecoderCopyContent`	Obtain the message content, if any.
`CMSDecoderCopyDetachedContent`	Obtains the detached content specified with the CMSDecoderSetDetachedContent function.
`CMSDecoderCopyEncapsulatedContentType`	Obtains the object identifier for the encapsulated data of a signed message.
`CMSDecoderCopySignerCert`	Obtains the certificate of the specified signer of a CMS message.

CMSDecoderCopySignerEmailAddress	Obtains the email address of the specified signer of a CMS message.
CMSDecoderCopySignerStatus	Obtains the status of a CMS message's signature.
CMSDecoderCreate	Creates a CMSDecoder.
CMSDecoderFinalizeMessage	Indicates that there is no more data to decode.
CMSDecoderGetNumSigners	Obtains the number of signers of a message.
CMSDecoderGetTypeID	Returns the type identifier for the CMSDecoder opaque type.
CMSDecoderIsContentEncrypted	Determines whether a CMS message was encrypted.
CMSDecoderSetDetachedContent	Specifies the message's detached content, if any.
CMSDecoderSetSearchKeychain	Specifies the keychains to search for intermediate certificates to be used in verifying a signed message's signer certificates.
CMSDecoderUpdateMessage	Feeds raw bytes of the message to be decoded into the decoder.

Data Types & Constants

All of the new data types and constants in this header file are listed alphabetically, with links to documentation and abstracts, if available.

CMSDecoderRef	Opaque reference to a CMS decoder object.
CMSSignerStatus	Constants that indicate the status of the signature and signer information in a signed message, as obtained by the CMSDecoderCopySignerStatus function.
kCMSSignerInvalidCert	The message was signed but the signer's certificate could not be verified.
kCMSSignerInvalidIndex	The specified value for the signer index (signerIndex parameter) is greater than the number of signers of the message minus one (signerIndex > (numSigners – 1)).
kCMSSignerInvalidSignature	The message was signed but the signature is invalid.
kCMSSignerNeedsDetachedContent	The message was signed but has detached content. You must call the CMSDecoderSetDetachedContent function before ascertaining the signature status.
kCMSSignerUnsigned	The message was not signed.
kCMSSignerValid	The message was signed and both the signature and the signer certificate have been verified.

CMSEncoder.h

Functions

All of the new functions in this header file are listed alphabetically, with links to documentation and abstracts, if available.

`CMSEncode`	Encodes a message and obtains the result in one high-level function call.
`CMSEncoderAddRecipients`	Specifies a message is to be encrypted and specifies the recipients of the message.
`CMSEncoderAddSignedAttributes`	Specifies attributes for a signed message.
`CMSEncoderAddSigners`	Specifies signers of the message.
`CMSEncoderAddSupportingCerts`	Adds certificates to a message.
`CMSEncoderCopyEncapsulatedContentType`	Obtains the object identifier for the encapsulated data of a signed message.
`CMSEncoderCopyEncodedContent`	Finishes encoding the message and obtains the encoded result.
`CMSEncoderCopyRecipients`	Obtains the array of recipients specified with the CMSEncoderAddRecipients function.
`CMSEncoderCopySigners`	Obtains the array of signers specified with the CMSEncoderAddSigners function.
`CMSEncoderCopySupportingCerts`	Obtains the certificates added to a message with CMSEncoderAddSupportingCerts.
`CMSEncoderCreate`	Creates a CMSEncoder reference.
`CMSEncoderGetCertificateChainMode`	Obtains a constant that indicates which certificates are to be included in a signed CMS message.
`CMSEncoderGetHasDetachedContent`	Indicates whether the message is to have detached content.
`CMSEncoderGetTypeID`	Returns the type identifier for the CMSEncoder opaque type.
`CMSEncoderSetCertificateChainMode`	Specifies which certificates to include in a signed CMS message.
`CMSEncoderSetEncapsulatedContentType`	Specifies an object identifier for the encapsulated data of a signed message.
`CMSEncoderSetHasDetachedContent`	Specifies whether the signed data is to be separate from the message.

`CMSEncoderUpdateContent`	Feeds content bytes into the encoder.

Data Types & Constants

All of the new data types and constants in this header file are listed alphabetically, with links to documentation and abstracts, if available.

`CMSCertificateChainMode`	Constants that can be set by the CMSEncoderSetCertificateChainMode function to specify what certificates to include in a signed message.
`CMSEncoderRef`	Opaque reference to a CMS encoder object.
`CMSSignedAttributes`	Optional attributes that can be specified with the CMSEncoderAddSignedAttributes function.
`kCMSAttrNone`	No attributes.
`kCMSAttrSigningTime`	Causes the encoder to include the signing time.
`kCMSAttrSmimeCapabilities`	Adds information to the signed message that identifies signature, encryption, and digest algorithms supported by the encoder. Adding this attribute does not change the way in which the message is encoded. See RFC 2311: S/MIME Version 2 Message Specification (http://www.ietf.org/rfc/rfc2311.txt) section 2.5.2 for more information about the capabilities attribute.
`kCMSAttrSmimeEncryptionKeyPrefs`	Indicates that the signing certificate included with the message is the preferred one for S/MIME encryption.
`kCMSAttrSmimeMSEncryptionKeyPrefs`	Same as kCMSSmimeEncryptionKeyPrefs, using an attribute object identifier (OID) preferred by Microsoft.
`kCMSCertificateChain`	Include the signer certificate chain up to but not including the root certificate.
`kCMSCertificateChainWithRoot`	Include the entire signer certificate chain, including the root certificate.
`kCMSCertificateNone`	Don't include any certificates.
`kCMSCertificateSignerOnly`	Only include signer certificates.

SecAsn1Coder.h

Functions

All of the new functions in this header file are listed alphabetically, with links to documentation and abstracts, if available.

SecAsn1AllocCopy	
SecAsn1AllocCopyItem	
SecAsn1AllocItem	
SecAsn1CoderCreate	
SecAsn1CoderRelease	
SecAsn1Decode	
SecAsn1DecodeData	
SecAsn1EncodeItem	
SecAsn1Malloc	

Data Types & Constants

All of the new data types and constants in this header file are listed alphabetically, with links to documentation and abstracts, if available.

SecAsn1CoderRef	

SecAsn1Templates.h

Data Types & Constants

All of the new data types and constants in this header file are listed alphabetically, with links to documentation and abstracts, if available.

kSecAsn1AnyTemplate	
kSecAsn1BitStringTemplate	
kSecAsn1BMPStringTemplate	
kSecAsn1BooleanTemplate	
kSecAsn1EnumeratedTemplate	
kSecAsn1GeneralizedTimeTemplate	
kSecAsn1IA5StringTemplate	
kSecAsn1IntegerTemplate	
kSecAsn1NullTemplate	

kSecAsn1ObjectIDTemplate	
kSecAsn1OctetStringTemplate	
kSecAsn1PointerToAnyTemplate	
kSecAsn1PointerToBitStringTemplate	
kSecAsn1PointerToBMPStringTemplate	
kSecAsn1PointerToBooleanTemplate	
kSecAsn1PointerToEnumeratedTemplate	
kSecAsn1PointerToGeneralizedTimeTemplate	
kSecAsn1PointerToIA5StringTemplate	
kSecAsn1PointerToIntegerTemplate	
kSecAsn1PointerToNullTemplate	
kSecAsn1PointerToObjectIDTemplate	
kSecAsn1PointerToOctetStringTemplate	
kSecAsn1PointerToPrintableStringTemplate	
kSecAsn1PointerToT61StringTemplate	
kSecAsn1PointerToTeletexStringTemplate	
kSecAsn1PointerToUniversalStringTemplate	
kSecAsn1PointerToUTCTimeTemplate	
kSecAsn1PointerToUTF8StringTemplate	
kSecAsn1PointerToVisibleStringTemplate	
kSecAsn1PrintableStringTemplate	
kSecAsn1SequenceOfAnyTemplate	
kSecAsn1SequenceOfBitStringTemplate	
kSecAsn1SequenceOfBMPStringTemplate	
kSecAsn1SequenceOfBooleanTemplate	
kSecAsn1SequenceOfEnumeratedTemplate	
kSecAsn1SequenceOfGeneralizedTimeTemplate	
kSecAsn1SequenceOfIA5StringTemplate	

kSecAsn1SequenceOfIntegerTemplate	
kSecAsn1SequenceOfNullTemplate	
kSecAsn1SequenceOfObjectIDTemplate	
kSecAsn1SequenceOfOctetStringTemplate	
kSecAsn1SequenceOfPrintableStringTemplate	
kSecAsn1SequenceOfT61StringTemplate	
kSecAsn1SequenceOfTeletexStringTemplate	
kSecAsn1SequenceOfUniversalStringTemplate	
kSecAsn1SequenceOfUTCTimeTemplate	
kSecAsn1SequenceOfUTF8StringTemplate	
kSecAsn1SequenceOfVisibleStringTemplate	
kSecAsn1SetOfAnyTemplate	
kSecAsn1SetOfBitStringTemplate	
kSecAsn1SetOfBMPStringTemplate	
kSecAsn1SetOfBooleanTemplate	
kSecAsn1SetOfEnumeratedTemplate	
kSecAsn1SetOfGeneralizedTimeTemplate	
kSecAsn1SetOfIA5StringTemplate	
kSecAsn1SetOfIntegerTemplate	
kSecAsn1SetOfNullTemplate	
kSecAsn1SetOfObjectIDTemplate	
kSecAsn1SetOfOctetStringTemplate	
kSecAsn1SetOfPrintableStringTemplate	
kSecAsn1SetOfT61StringTemplate	
kSecAsn1SetOfTeletexStringTemplate	
kSecAsn1SetOfUniversalStringTemplate	
kSecAsn1SetOfUTCTimeTemplate	
kSecAsn1SetOfUTF8StringTemplate	

kSecAsn1SetOfVisibleStringTemplate	
kSecAsn1SkipTemplate	
kSecAsn1T61StringTemplate	
kSecAsn1TeletexStringTemplate	
kSecAsn1UniversalStringTemplate	
kSecAsn1UnsignedIntegerTemplate	
kSecAsn1UTCTimeTemplate	
kSecAsn1UTF8StringTemplate	
kSecAsn1VisibleStringTemplate	

SecAsn1Types.h

Data Types & Constants

All of the new data types and constants in this header file are listed alphabetically, with links to documentation and abstracts, if available.

SEC_ASN1_ANY	
SEC_ASN1_ANY_CONTENTS	
SEC_ASN1_APPLICATION	
SEC_ASN1_BIT_STRING	
SEC_ASN1_BMP_STRING	
SEC_ASN1_BOOLEAN	
SEC_ASN1_CHOICE	
SEC_ASN1_CLASS_MASK	
SEC_ASN1_CONSTRUCTED	
SEC_ASN1_CONTEXT_SPECIFIC	
SEC_ASN1_DYNAMIC	
SEC_ASN1_EMBEDDED_PDV	
SEC_ASN1_ENUMERATED	
SEC_ASN1_EXPLICIT	

SEC_ASN1_GENERAL_STRING	
SEC_ASN1_GENERALIZED_TIME	
SEC_ASN1_GRAPHIC_STRING	
SEC_ASN1_GROUP	
SEC_ASN1_HIGH_TAG_NUMBER	
SEC_ASN1_IA5_STRING	
SEC_ASN1_INLINE	
SEC_ASN1_INNER	
SEC_ASN1_INTEGER	
SEC_ASN1_METHOD_MASK	
SEC_ASN1_NULL	
SEC_ASN1_NUMERIC_STRING	
SEC_ASN1_OBJECT_DESCRIPTOR	
SEC_ASN1_OBJECT_ID	
SEC_ASN1_OCTET_STRING	
SEC_ASN1_OPTIONAL	
SEC_ASN1_POINTER	
SEC_ASN1_PRIMITIVE	
SEC_ASN1_PRINTABLE_STRING	
SEC_ASN1_PRIVATE	
SEC_ASN1_REAL	
SEC_ASN1_SAVE	
SEC_ASN1_SEQUENCE	
SEC_ASN1_SEQUENCE_OF	
SEC_ASN1_SET	
SEC_ASN1_SET_OF	
SEC_ASN1_SIGNED_INT	
SEC_ASN1_SKIP	

SEC_ASN1_SKIP_REST	
SEC_ASN1_T61_STRING	
SEC_ASN1_TAG_MASK	
SEC_ASN1_TAGNUM_MASK	
SEC_ASN1_TELETEX_STRING	
SEC_ASN1_UNIVERSAL	
SEC_ASN1_UNIVERSAL_STRING	
SEC_ASN1_UTC_TIME	
SEC_ASN1_UTF8_STRING	
SEC_ASN1_VIDEOTEX_STRING	
SEC_ASN1_VISIBLE_STRING	
SecAsn1Template	
SecAsn1TemplateChooser	
SecAsn1TemplateChooserPtr	

SecBase.h

Functions

All of the new functions in this header file are listed alphabetically, with links to documentation and abstracts, if available.

SecCopyErrorMessageString	

Data Types & Constants

All of the new data types and constants in this header file are listed alphabetically, with links to documentation and abstracts, if available.

errSecInvalidTrustSettings	
errSecNoTrustSettings	
errSecPkcs12VerifyFailure	

SecCertificate.h

Functions

All of the new functions in this header file are listed alphabetically, with links to documentation and abstracts, if available.

SecCertificateCopyCommonName	
SecCertificateCopyEmailAddresses	
SecCertificateCopyPreference	
SecCertificateCopyPublicKey	
SecCertificateGetAlgorithmID	
SecCertificateSetPreference	

SecIdentity.h

Functions

All of the new functions in this header file are listed alphabetically, with links to documentation and abstracts, if available.

SecIdentityCopyPreference	
SecIdentityCopySystemIdentity	
SecIdentityCreateWithCertificate	
SecIdentitySetPreference	
SecIdentitySetSystemIdentity	

Data Types & Constants

All of the new data types and constants in this header file are listed alphabetically, with links to documentation and abstracts, if available.

kSecIdentityDomainDefault	
kSecIdentityDomainKerberosKDC	

SecImportExport.h

Data Types & Constants

All of the new data types and constants in this header file are listed alphabetically, with links to documentation and abstracts, if available.

kSecFormatSSHv2	

SecKey.h

Functions

All of the new functions in this header file are listed alphabetically, with links to documentation and abstracts, if available.

SecKeyGenerate	
SecKeyGetCredentials	
SecKeyGetCSPHandle	

Data Types & Constants

All of the new data types and constants in this header file are listed alphabetically, with links to documentation and abstracts, if available.

kSecCredentialTypeDefault	
kSecCredentialTypeNoUI	
kSecCredentialTypeWithUI	
SecCredentialType	

SecKeychain.h

Data Types & Constants

All of the new data types and constants in this header file are listed alphabetically, with links to documentation and abstracts, if available.

kSecAuthenticationTypeAny	

kSecProtocolTypeAny	
kSecProtocolTypeCIFS	
kSecProtocolTypeCVSpserver	
kSecProtocolTypeSVN	
kSecTrustSettingsChangedEvent	
kSecTrustSettingsChangedEventMask	

SecKeychainItem.h

Data Types & Constants

All of the new data types and constants in this header file are listed alphabetically, with links to documentation and abstracts, if available.

kSecPrivateKeyItemClass	
kSecPublicKeyItemClass	
kSecSymmetricKeyItemClass	

SecPolicy.h

Functions

All of the new functions in this header file are listed alphabetically, with links to documentation and abstracts, if available.

SecPolicySetValue	

SecTrust.h

Functions

All of the new functions in this header file are listed alphabetically, with links to documentation and abstracts, if available.

SecTrustCopyCustomAnchorCertificates	
SecTrustCopyPolicies	

SecTrustGetCssmResultCode	
SecTrustSetPolicies	

SecTrustSettings.h

Functions

All of the new functions in this header file are listed alphabetically, with links to documentation and abstracts, if available.

SecTrustSettingsCopyCertificates	
SecTrustSettingsCopyModificationDate	
SecTrustSettingsCopyTrustSettings	
SecTrustSettingsCreateExternalRepresentation	
SecTrustSettingsImportExternalRepresentation	
SecTrustSettingsRemoveTrustSettings	
SecTrustSettingsSetTrustSettings	

Data Types & Constants

All of the new data types and constants in this header file are listed alphabetically, with links to documentation and abstracts, if available.

kSecTrustSettingsAllowedError	
kSecTrustSettingsApplication	
kSecTrustSettingsDefaultRootCertSetting	
kSecTrustSettingsDomainAdmin	
kSecTrustSettingsDomainSystem	
kSecTrustSettingsDomainUser	
kSecTrustSettingsKeyUsage	
kSecTrustSettingsKeyUseAny	
kSecTrustSettingsKeyUseEnDecryptData	
kSecTrustSettingsKeyUseEnDecryptKey	

kSecTrustSettingsKeyUseKeyExchange	
kSecTrustSettingsKeyUseSignature	
kSecTrustSettingsKeyUseSignCert	
kSecTrustSettingsKeyUseSignRevocation	
kSecTrustSettingsPolicy	
kSecTrustSettingsPolicyString	
kSecTrustSettingsResult	
kSecTrustSettingsResultDeny	
kSecTrustSettingsResultInvalid	
kSecTrustSettingsResultTrustAsRoot	
kSecTrustSettingsResultTrustRoot	
kSecTrustSettingsResultUnspecified	
SecTrustSettingsDomain	
SecTrustSettingsKeyUsage	
SecTrustSettingsResult	

SecureDownload.h

Functions

All of the new functions in this header file are listed alphabetically, with links to documentation and abstracts, if available.

SecureDownloadCopyCreationDate	
SecureDownloadCopyName	
SecureDownloadCopyTicketLocation	
SecureDownloadCopyURLs	
SecureDownloadCreateWithTicket	
SecureDownloadFinished	
SecureDownloadGetDownloadSize	
SecureDownloadRelease	

SecureDownloadUpdateWithData	

Data Types & Constants

All of the new data types and constants in this header file are listed alphabetically, with links to documentation and abstracts, if available.

errSecureDownloadInvalidDownload	
errSecureDownloadInvalidTicket	
kSecureDownloadDoNotEvaluateSigner	
kSecureDownloadEvaluateSigner	
kSecureDownloadFailEvaluation	
SecureDownloadRef	
SecureDownloadTrustCallbackResult	
SecureDownloadTrustEvaluateCallback	
SecureDownloadTrustSetupCallback	

SecureTransport.h

Functions

All of the new functions in this header file are listed alphabetically, with links to documentation and abstracts, if available.

SSLCopyCertificateAuthorities	
SSLCopyDistinguishedNames	
SSLCopyPeerCertificates	
SSLCopyTrustedRoots	
SSLSetCertificateAuthorities	

certextensions.h

Data Types & Constants

All of the new data types and constants in this header file are listed alphabetically, with links to documentation and abstracts, if available.

CE_NameRegistrationAuthorities	
CE_QC_Statement	
CE_QC_Statements	
CE_SemanticsInformation	
DT_QC_Statements	

cssmapple.h

Data Types & Constants

All of the new data types and constants in this header file are listed alphabetically, with links to documentation and abstracts, if available.

CSSM_ACL_KEYCHAIN_PROMPT_INVALID	
CSSM_ACL_KEYCHAIN_PROMPT_INVALID_ACT	
CSSM_ACL_KEYCHAIN_PROMPT_UNSIGNED	
CSSM_ACL_KEYCHAIN_PROMPT_UNSIGNED_ACT	
CSSM_ACL_SUBJECT_TYPE_ASYMMETRIC_KEY	
CSSM_ALGID_OPENSSH1	
CSSM_CERT_STATUS_TRUST_SETTINGS_DENY	
CSSM_CERT_STATUS_TRUST_SETTINGS_FOUND_ADMIN	
CSSM_CERT_STATUS_TRUST_SETTINGS_FOUND_SYSTEM	
CSSM_CERT_STATUS_TRUST_SETTINGS_FOUND_USER	
CSSM_CERT_STATUS_TRUST_SETTINGS_IGNORED_ERROR	
CSSM_CERT_STATUS_TRUST_SETTINGS_TRUST	
CSSM_DL_DB_RECORD_EXTENDED_ATTRIBUTE	

CSSM_KEYATTR_PUBLIC_KEY_ENCRYPT	
CSSM_KEYBLOB_RAW_FORMAT_OPENSSH2	
CSSM_KEYBLOB_WRAPPED_FORMAT_OPENSSH1	
CSSM_PADDING_APPLE_SSLv2	
CSSM_SAMPLE_TYPE_ASYMMETRIC_KEY	
CSSM_TP_ACTION_IMPLICIT_ANCHORS	
CSSM_TP_ACTION_TRUST_SETTINGS	
CSSM_WORDID_ASYMMETRIC_KEY	
CSSMERR_APPLE_DOTMAC_CSR_VERIFY_FAIL	
CSSMERR_APPLE_DOTMAC_FAILED_CONSISTENCY_CHECK	
CSSMERR_APPLETP_INVALID_EMPTY_SUBJECT	
CSSMERR_APPLETP_RS_BAD_CERT_CHAIN_LENGTH	
CSSMERR_APPLETP_RS_BAD_EXTENDED_KEY_USAGE	
CSSMERR_APPLETP_TRUST_SETTING_DENY	
CSSMERR_APPLETP_UNKNOWN_QUAL_CERT_STATEMENT	
CSSMERR_CSP_APPLE_SSLv2_ROLLBACK	

cssmconfig.h

Data Types & Constants

All of the new data types and constants in this header file are listed alphabetically, with links to documentation and abstracts, if available.

CSSM_INTPTR	
CSSM_SIZE	

oidsalg.h

Data Types & Constants

All of the new data types and constants in this header file are listed alphabetically, with links to documentation and abstracts, if available.

CSSMOID_APPLE_TP_CODE_SIGNING	
CSSMOID_APPLE_TP_PACKAGE_SIGNING	
CSSMOID_APPLE_TP_PKINIT_CLIENT	
CSSMOID_APPLE_TP_PKINIT_SERVER	
CSSMOID_APPLE_TP_RESOURCE_SIGN	
CSSMOID_APPLE_TP_SW_UPDATE_SIGNING	
CSSMOID_DOTMAC_CERT_REQ_SHARED_SERVICES	

oidsattr.h

Data Types & Constants

All of the new data types and constants in this header file are listed alphabetically, with links to documentation and abstracts, if available.

CSSMOID_ETSI_QCS_QC_COMPLIANCE	
CSSMOID_ETSI_QCS_QC_LIMIT_VALUE	
CSSMOID_ETSI_QCS_QC_RETENTION	
CSSMOID_ETSI_QCS_QC_SSCD	
CSSMOID_KERBv5_PKINIT_AUTH_DATA	
CSSMOID_KERBv5_PKINIT_DH_KEY_DATA	
CSSMOID_KERBv5_PKINIT_RKEY_DATA	
CSSMOID_OID_QCS_SYNTAX_V1	
CSSMOID_OID_QCS_SYNTAX_V2	
CSSMOID_PDA_COUNTRY_CITIZEN	
CSSMOID_PDA_COUNTRY_RESIDENCE	
CSSMOID_PDA_DATE_OF_BIRTH	
CSSMOID_PDA_GENDER	
CSSMOID_PDA_PLACE_OF_BIRTH	

oidsbase.h

Data Types & Constants

All of the new data types and constants in this header file are listed alphabetically, with links to documentation and abstracts, if available.

OID_ETSI	
OID_ETSI_LENGTH	
OID_ETSI_QCS	
OID_ETSI_QCS_LENGTH	
OID_KERBv5	
OID_KERBv5_LEN	
OID_KERBv5_PKINIT	
OID_KERBv5_PKINIT_LEN	
OID_OTHER_NAME	
OID_OTHER_NAME_LENGTH	
OID_PDA	
OID_PDA_LENGTH	
OID_QCS	
OID_QCS_LENGTH	

oidscert.h

Data Types & Constants

All of the new data types and constants in this header file are listed alphabetically, with links to documentation and abstracts, if available.

CSSMOID_APPLE_EKU_RESOURCE_SIGNING	
CSSMOID_APPLE_EKU_SYSTEM_IDENTITY	
CSSMOID_BiometricInfo	
CSSMOID_KERBv5_PKINIT_KP_CLIENT_AUTH	

CSSMOID_KERBv5_PKINIT_KP_KDC	
CSSMOID_QC_Statements	

10.4 Symbol Changes

This article lists the symbols added to `Security.framework` in Mac OS X v10.4.

C Symbols

All of the header files with new symbols are listed alphabetically, with their new symbols described.

AuthSession.h

Data Types & Constants

All of the new data types and constants in this header file are listed alphabetically, with links to documentation and abstracts, if available.

`errSessionValueNotSet`	

AuthorizationPlugin.h

Functions

All of the new functions in this header file are listed alphabetically, with links to documentation and abstracts, if available.

`AuthorizationPluginCreate`	Initializes the plug-in and exchanges interfaces with the authorization engine.

Data Types & Constants

All of the new data types and constants in this header file are listed alphabetically, with links to documentation and abstracts, if available.

`AuthorizationCallbacks`	The interface implemented by the Security Server.
`AuthorizationContextFlags`	Defines flags that specify whether authentication data should be made available to the authorization client.

AuthorizationEngineRef	Handle passed from the authorization engine to an instance of a mechanism in a plug-in.
AuthorizationMechanismId	The mechanism ID specified in the authorization policy database is passed to the plug-in to create the appropriate mechanism.
AuthorizationMechanismRef	Handle passed by the plug-in to the authorization engine when creating an instance of a mechanism.
AuthorizationPluginId	
AuthorizationPluginInterface	
AuthorizationPluginRef	Handle passed by the plug-in to the authorization engine when the plug-in is initiated.
AuthorizationResult	The data type for the result of an authorization evaluation.
AuthorizationSessionId	A unique value for an authorization session, provided by the authorization engine.
AuthorizationValue	Used to pass data between the authorization engine and the plug-in mechanism.
AuthorizationValueVector	Used to pass arguments from the authorization policy database to the authorization mechanism.
kAuthorizationCallbacksVersion	
kAuthorizationContextFlagExtractable	It is possible for the authorization client to use the AuthorizationCopyInfo function to obtain the value.
kAuthorizationContextFlagVolatile	The value is not saved for the authorization client.
kAuthorizationPluginInterfaceVersion	
kAuthorizationResultAllow	The authorization operation succeeded and authorization should be granted.
kAuthorizationResultDeny	The authorization operation succeeded and authorization should be denied.
kAuthorizationResultUndefined	The authorization operation failed and should not be retried for this session.
kAuthorizationResultUserCanceled	The user has requested that the authorization evaluation be terminated.

CipherSuite.h

Data Types & Constants

All of the new data types and constants in this header file are listed alphabetically, with links to documentation and abstracts, if available.

TLS_DH_anon_WITH_AES_128_CBC_SHA	
TLS_DH_anon_WITH_AES_256_CBC_SHA	
TLS_DH_DSS_WITH_AES_128_CBC_SHA	
TLS_DH_DSS_WITH_AES_256_CBC_SHA	
TLS_DH_RSA_WITH_AES_128_CBC_SHA	
TLS_DH_RSA_WITH_AES_256_CBC_SHA	
TLS_DHE_DSS_WITH_AES_128_CBC_SHA	
TLS_DHE_DSS_WITH_AES_256_CBC_SHA	
TLS_DHE_RSA_WITH_AES_128_CBC_SHA	
TLS_DHE_RSA_WITH_AES_256_CBC_SHA	
TLS_RSA_WITH_AES_128_CBC_SHA	
TLS_RSA_WITH_AES_256_CBC_SHA	

SecBase.h

Data Types & Constants

All of the new data types and constants in this header file are listed alphabetically, with links to documentation and abstracts, if available.

errSecInvalidPasswordRef	
errSecKeyIsSensitive	The key must be wrapped to be exported.
errSecMultiplePrivKeys	An attempt was made to import multiple private keys.
errSecPassphraseRequired	A password is required for import or export.
errSecUnknownFormat	The item you are trying to import has an unknown format.
errSecUnsupportedFormat	The specified import or export format is not supported.

SecPasswordRef	

SecImportExport.h

Functions

All of the new functions in this header file are listed alphabetically, with links to documentation and abstracts, if available.

SecKeychainItemExport	Exports one or more certificates, keys, or identities.
SecKeychainItemImport	Imports one or more certificates, keys, or identities and adds them to a keychain.

Data Types & Constants

All of the new data types and constants in this header file are listed alphabetically, with links to documentation and abstracts, if available.

kSecFormatBSAFE	Format for asymmetric keys. BSAFE is a standard from RSA Security for encryption, digital signatures, and privacy.
kSecFormatNetscapeCertSequence	Set of certificates in the Netscape Certificate Sequence format.
kSecFormatOpenSSL	Format for asymmetric (public/private) keys. OpenSSL is an open source toolkit for Secure Sockets Layer (SSL) and Transport Layer Security (TLS). Also known as X.509 for public keys.
kSecFormatPEMSequence	Sequence of certificates and keys with PEM armour. PEM armour refers to a way of expressing binary data as an ASCII string so that it can be transferred over text-only channels such as email. This is the default format for multiple items.
kSecFormatPKCS12	Set of certificates and private keys. PKCS12 is the Personal Information Exchange Syntax from RSA Security, Inc.
kSecFormatPKCS7	Sequence of certificates, no PEM armour. PKCS7 is the Cryptographic Message Syntax Standard from RSA Security, Inc.
kSecFormatRawKey	Format for symmetric keys. Raw, unformatted key bits. This is the default for symmetric keys.
kSecFormatSSH	Not supported.

kSecFormatUnknown	When importing, indicates the format is unknown.
kSecFormatWrappedLSH	Not supported.
kSecFormatWrappedOpenSSL	Format for wrapped symmetric and private keys. OpenSSL is an open-source toolkit for Secure Sockets Layer (SSL) and Transport Layer Security (TLS).
kSecFormatWrappedPKCS8	Format for wrapped symmetric and private keys. PKCS8 is the Private-Key Information Syntax Standard from RSA Security.
kSecFormatWrappedSSH	Not supported.
kSecFormatX509Cert	Format for certificates. DER (distinguished encoding rules) encoded. X.509 is a standard for digital certificates from the International Telecommunication Union (ITU). This is the default for certificates.
kSecItemPemArmour	The exported data should have PEM armour.
kSecItemTypeAggregate	Indicates a set of certificates or certificates and private keys, such as PKCS7, PKCS12, or kSecFormatPEMSequence formats (see “Keychain Item Import/Export Formats”).
kSecItemTypeCertificate	Indicates a certificate.
kSecItemTypePrivateKey	Indicates a private key.
kSecItemTypePublicKey	Indicates a public key.
kSecItemTypeSessionKey	Indicates a session key.
kSecItemTypeUnknown	
kSecKeyImportOnlyOne	Prevents the importing of more than one private key by the SecKeychainItemImport function.
kSecKeyNoAccessControl	When set, imported private keys have no access object attached to them. In the absence of both this bit and the accessRef field in SecKeyImportExportParameters, imported private keys are given default access controls.
kSecKeySecurePassphrase	When set, the password for import or export is obtained by user prompt.
SEC_KEY_IMPORT_EXPORT_PARAMS_VERSION	
SecExternalFormat	Specifies the format of an item after export from or before import to the keychain.
SecExternalItemType	Specifies the type of keychain item being imported.
SecItemImportExportFlags	Defines values for import and export flags.

SecKeyImportExportFlags	Defines values for the flags field of the import/export parameters.
SecKeyImportExportParameters	Contains input parameters for import and export functions.

SecKey.h

Data Types & Constants

All of the new data types and constants in this header file are listed alphabetically, with links to documentation and abstracts, if available.

kSecKeyAlias	Type blob; currently unused.
kSecKeyAlwaysSensitive	Type uint32; value is nonzero. This key has always been marked sensitive.
kSecKeyApplicationTag	Type blob; currently unused.
kSecKeyDecrypt	Type uint32; value is nonzero. This key can be used in a decrypt operation.
kSecKeyDerive	Type uint32; value is nonzero. This key can be used in a key derivation operation.
kSecKeyEffectiveKeySize	Type uint32; value is the effective number of bits in this key. For example, a DES key has a key size in bits (kSecKeyKeySizeInBits) of 64 but a value for kSecKeyEffectiveKeySize of 56.
kSecKeyEncrypt	Type uint32; value is nonzero. This key can be used in an encrypt operation.
kSecKeyEndDate	Type CSSM_DATE. Latest date at which this key may be used. If the value is all zeros or not present, no restriction applies.
kSecKeyExtractable	Type uint32; value is nonzero. This key can be wrapped.
kSecKeyKeyClass	Type uint32 (CSSM_KEYCLASS); value is one of CSSM_KEYCLASS_PUBLIC_KEY, CSSM_KEYCLASS_PRIVATE_KEY or CSSM_KEYCLASS_SESSION_KEY.
kSecKeyKeyCreator	Type data. The data points to a CSSM_GUID structure representing the module ID of the CSP owning this key.
kSecKeyKeySizeInBits	Type uint32; value is the number of bits in this key.
kSecKeyKeyType	Type uint32; value is a CSSM algorithm (CSSM_ALGORITHMS) representing the algorithm associated with this key.

kSecKeyLabel	Type blob; for private and public keys this contains the hash of the public key.
kSecKeyModifiable	Type uint32; value is nonzero. Attributes of this key can be modified.
kSecKeyNeverExtractable	Type uint32; value is nonzero. This key was never marked extractable.
kSecKeyPermanent	Type uint32; value is nonzero. This key is permanent (stored in some keychain) and is always 1.
kSecKeyPrintName	Type blob; human readable name of the key. Same as kSecLabelItemAttr for normal keychain items.
kSecKeyPrivate	Type uint32; value is nonzero. This key is protected by a user login, a password, or both.
kSecKeySensitive	Type uint32; value is nonzero. This key cannot be wrapped with CSSM_ALGID_NONE.
kSecKeySign	Type uint32, value is nonzero. This key can be used in a sign operation.
kSecKeySignRecover	Type uint32.
kSecKeyStartDate	Type CSSM_DATE. Earliest date at which this key may be used. If the value is all zeros or not present, no restriction applies.
kSecKeyUnwrap	Type uint32; value is nonzero. This key can unwrap other keys.
kSecKeyVerify	Type uint32, value is nonzero. This key can be used in a verify operation.
kSecKeyVerifyRecover	Type uint32. This key can unwrap other keys.
kSecKeyWrap	Type uint32; value is nonzero. This key can wrap other keys.

SecKeychain.h

Data Types & Constants

All of the new data types and constants in this header file are listed alphabetically, with links to documentation and abstracts, if available.

kSecPreferencesDomainDynamic	

SecureTransport.h

Data Types & Constants

All of the new data types and constants in this header file are listed alphabetically, with links to documentation and abstracts, if available.

errSSLConnectionRefused	The peer dropped the connection before responding.
errSSLHostNameMismatch	

certextensions.h

Data Types & Constants

All of the new data types and constants in this header file are listed alphabetically, with links to documentation and abstracts, if available.

CE_AccessDescription	
CE_AuthorityInfoAccess	
DT_AuthorityInfoAccess	

cssmapple.h

Data Types & Constants

All of the new data types and constants in this header file are listed alphabetically, with links to documentation and abstracts, if available.

CSSM_ACL_AUTHORIZATION_IS_PREAUTH	
CSSM_ACL_AUTHORIZATION_PREAUTH	
CSSM_ACL_AUTHORIZATION_PREAUTH_BASE	
CSSM_ACL_AUTHORIZATION_PREAUTH_END	
CSSM_ACL_AUTHORIZATION_PREAUTH_SLOT	
CSSM_ACL_PREAUTH_TRACKING_AUTHORIZED	
CSSM_ACL_PREAUTH_TRACKING_BLOCKED	
CSSM_ACL_PREAUTH_TRACKING_COUNT_MASK	
CSSM_ACL_PREAUTH_TRACKING_STATE	
CSSM_ACL_PREAUTH_TRACKING_UNKNOWN	
CSSM_ACL_SUBJECT_TYPE_PREAUTH	
CSSM_ACL_SUBJECT_TYPE_PREAUTH_SOURCE	

CSSM_ACL_SUBJECT_TYPE_SYMMETRIC_KEY	
CSSM_ALGID_ENTROPY_DEFAULT	
CSSM_ALGID_PBE_OPENSSL_MD5	
CSSM_ALGID_SECURE_PASSPHRASE	
CSSM_ALGID_SHA224	
CSSM_ALGID_SHA224WithRSA	
CSSM_ALGID_SHA256	
CSSM_ALGID_SHA256WithRSA	
CSSM_ALGID_SHA384	
CSSM_ALGID_SHA384WithRSA	
CSSM_ALGID_SHA512	
CSSM_ALGID_SHA512WithRSA	
CSSM_APPLE_PRIVATE_CSPDL_CODE_10	
CSSM_APPLE_PRIVATE_CSPDL_CODE_11	
CSSM_APPLE_PRIVATE_CSPDL_CODE_12	
CSSM_APPLE_PRIVATE_CSPDL_CODE_13	
CSSM_APPLE_PRIVATE_CSPDL_CODE_14	
CSSM_APPLE_PRIVATE_CSPDL_CODE_15	
CSSM_APPLE_PRIVATE_CSPDL_CODE_8	
CSSM_APPLE_PRIVATE_CSPDL_CODE_9	
CSSM_APPLE_TP_SSL_CLIENT	
CSSM_APPLE_UNLOCK_TYPE_KEY_DIRECT	
CSSM_APPLE_UNLOCK_TYPE_WRAPPED_PRIVATE	
CSSM_ATTRIBUTE_ALERT_TITLE	
CSSM_ATTRIBUTE_PROMPT	
CSSM_ATTRIBUTE_VERIFY_PASSPHRASE	
CSSM_DB_ACCESS_RESET	
CSSM_DL_DB_RECORD_UNLOCK_REFERRAL	

CSSM_ERRCODE_DEVICE_FAILED	
CSSM_ERRCODE_DEVICE_RESET	
CSSM_KEYBLOB_WRAPPED_FORMAT_OPENSSL	
CSSM_SAMPLE_TYPE_PREAUTH	
CSSM_SAMPLE_TYPE_SYMMETRIC_KEY	
CSSM_TP_ACTION_CRL_SUFFICIENT	
CSSM_TP_ACTION_REQUIRE_CRL_IF_PRESENT	
CSSM_TP_ACTION_REQUIRE_REV_PER_CERT	
CSSM_WORDID_KEY	
CSSM_WORDID_PIN	
CSSM_WORDID_PREAUTH	
CSSM_WORDID_PREAUTH_SOURCE	
CSSMERR_AC_DEVICE_FAILED	
CSSMERR_AC_DEVICE_RESET	
CSSMERR_APPLE_DOTMAC_NO_REQ_PENDING	
CSSMERR_APPLE_DOTMAC_REQ_IS_PENDING	
CSSMERR_APPLE_DOTMAC_REQ_QUEUED	
CSSMERR_APPLE_DOTMAC_REQ_REDIRECT	
CSSMERR_APPLE_DOTMAC_REQ_SERVER_ALREADY_EXIST	
CSSMERR_APPLE_DOTMAC_REQ_SERVER_AUTH	
CSSMERR_APPLE_DOTMAC_REQ_SERVER_ERR	
CSSMERR_APPLE_DOTMAC_REQ_SERVER_NOT_AVAIL	
CSSMERR_APPLE_DOTMAC_REQ_SERVER_PARAM	
CSSMERR_APPLE_DOTMAC_REQ_SERVER_SERVICE_ERROR	
CSSMERR_APPLE_DOTMAC_REQ_SERVER_UNIMPL	
CSSMERR_APPLETP_CODE_SIGN_DEVELOPMENT	
CSSMERR_APPLETP_CS_BAD_CERT_CHAIN_LENGTH	
CSSMERR_APPLETP_CS_BAD_PATH_LENGTH	

CSSMERR_APPLETP_CS_NO_BASIC_CONSTRAINTS	
CSSMERR_APPLETP_CS_NO_EXTENDED_KEY_USAGE	
CSSMERR_APPLETP_INCOMPLETE_REVOCATION_CHECK	
CSSMERR_APPLETP_NETWORK_FAILURE	
CSSMERR_APPLETP_OCSP_BAD_REQUEST	
CSSMERR_APPLETP_OCSP_BAD_RESPONSE	
CSSMERR_APPLETP_OCSP_INVALID_ANCHOR_CERT	
CSSMERR_APPLETP_OCSP_NO_SIGNER	
CSSMERR_APPLETP_OCSP_NONCE_MISMATCH	
CSSMERR_APPLETP_OCSP_NOT_TRUSTED	
CSSMERR_APPLETP_OCSP_RESP_INTERNAL_ERR	
CSSMERR_APPLETP_OCSP_RESP_MALFORMED_REQ	
CSSMERR_APPLETP_OCSP_RESP_SIG_REQUIRED	
CSSMERR_APPLETP_OCSP_RESP_TRY_LATER	
CSSMERR_APPLETP_OCSP_RESP_UNAUTHORIZED	
CSSMERR_APPLETP_OCSP_SIG_ERROR	
CSSMERR_APPLETP_OCSP_STATUS_UNRECOGNIZED	
CSSMERR_APPLETP_OCSP_UNAVAILABLE	
CSSMERR_APPLETP_SSL_BAD_EXT_KEY_USE	
CSSMERR_CL_DEVICE_FAILED	
CSSMERR_CL_DEVICE_RESET	
CSSMERR_CSP_DEVICE_FAILED	
CSSMERR_CSP_DEVICE_RESET	
CSSMERR_CSPDL_APPLE_DL_CONVERSION_ERROR	
CSSMERR_CSSM_DEVICE_FAILED	
CSSMERR_CSSM_DEVICE_RESET	
CSSMERR_DL_DEVICE_FAILED	
CSSMERR_DL_DEVICE_RESET	

CSSMERR_TP_DEVICE_FAILED	
CSSMERR_TP_DEVICE_RESET	
gGuidAppleDotMacDL	
gGuidAppleDotMacTP	
gGuidAppleLDAPDL	
gGuidAppleSdCSPDL	

oidsalg.h

Data Types & Constants

All of the new data types and constants in this header file are listed alphabetically, with links to documentation and abstracts, if available.

CSSMOID_APPLE_TP_CODE_SIGN	
CSSMOID_APPLE_TP_ICHAT	
CSSMOID_APPLE_TP_IP_SEC	
CSSMOID_DES_CBC	
CSSMOID_DOTMAC_CERT	
CSSMOID_DOTMAC_CERT_REQ	
CSSMOID_DOTMAC_CERT_REQ_ARCHIVE_FETCH	
CSSMOID_DOTMAC_CERT_REQ_ARCHIVE_LIST	
CSSMOID_DOTMAC_CERT_REQ_ARCHIVE_REMOVE	
CSSMOID_DOTMAC_CERT_REQ_ARCHIVE_STORE	
CSSMOID_DOTMAC_CERT_REQ_EMAIL_ENCRYPT	
CSSMOID_DOTMAC_CERT_REQ_EMAIL_SIGN	
CSSMOID_DOTMAC_CERT_REQ_IDENTITY	
CSSMOID_DOTMAC_CERT_REQ_VALUE_ASYNC	
CSSMOID_DOTMAC_CERT_REQ_VALUE_HOSTNAME	
CSSMOID_DOTMAC_CERT_REQ_VALUE_IS_PENDING	
CSSMOID_DOTMAC_CERT_REQ_VALUE_PASSWORD	

CSSMOID_DOTMAC_CERT_REQ_VALUE_RENEW	
CSSMOID_DOTMAC_CERT_REQ_VALUE_USERNAME	
CSSMOID_OAEP_ID_PSPECIFIED	
CSSMOID_OAEP_MGF1	
CSSMOID_PKCS5_DES_EDE3_CBC	
CSSMOID_PKCS5_DIGEST_ALG	
CSSMOID_PKCS5_ENCRYPT_ALG	
CSSMOID_PKCS5_HMAC_SHA1	
CSSMOID_PKCS5_PBES2	
CSSMOID_PKCS5_pbeWithMD2AndDES	
CSSMOID_PKCS5_pbeWithMD2AndRC2	
CSSMOID_PKCS5_pbeWithMD5AndDES	
CSSMOID_PKCS5_pbeWithMD5AndRC2	
CSSMOID_PKCS5_pbeWithSHA1AndDES	
CSSMOID_PKCS5_pbeWithSHA1AndRC2	
CSSMOID_PKCS5_PBKDF2	
CSSMOID_PKCS5_PBMAC1	
CSSMOID_PKCS5_RC2_CBC	
CSSMOID_PKCS5_RC5_CBC	
CSSMOID_RSAWithOAEP	
CSSMOID_SHA224	
CSSMOID_SHA224WithRSA	
CSSMOID_SHA256	
CSSMOID_SHA256WithRSA	
CSSMOID_SHA384	
CSSMOID_SHA384WithRSA	
CSSMOID_SHA512	
CSSMOID_SHA512WithRSA	

oidsattr.h

Data Types & Constants

All of the new data types and constants in this header file are listed alphabetically, with links to documentation and abstracts, if available.

CSSMOID_AD_CA_ISSUERS	
CSSMOID_AD_CA_REPOSITORY	
CSSMOID_AD_OCSP	
CSSMOID_AD_TIME_STAMPING	

oidsbase.h

Data Types & Constants

All of the new data types and constants in this header file are listed alphabetically, with links to documentation and abstracts, if available.

APPLE_CERT_POLICIES	
APPLE_CERT_POLICIES_LENGTH	
APPLE_DOTMAC_CERT_EXTEN_OID	
APPLE_DOTMAC_CERT_EXTEN_OID_LENGTH	
APPLE_DOTMAC_CERT_OID	
APPLE_DOTMAC_CERT_OID_LENGTH	
APPLE_DOTMAC_CERT_REQ_OID	
APPLE_DOTMAC_CERT_REQ_OID_LENGTH	
APPLE_DOTMAC_CERT_REQ_VALUE_OID	
APPLE_DOTMAC_CERT_REQ_VALUE_OID_LENGTH	
APPLE_EKU_CODE_SIGNING	
APPLE_EKU_CODE_SIGNING_LENGTH	
APPLE_EKU_OID	
APPLE_EKU_OID_LENGTH	

NETSCAPE_BASE_OID	
NETSCAPE_BASE_OID_LEN	
NETSCAPE_CERT_POLICY	
NETSCAPE_CERT_POLICY_LENGTH	
OID_AD	
OID_AD_LENGTH	
OID_AD_OCSP	
OID_AD_OCSP_LENGTH	
OID_NIST_HASHALG	
OID_NIST_HASHALG_LENGTH	
OID_PE	
OID_PE_LENGTH	

oidscert.h

Data Types & Constants

All of the new data types and constants in this header file are listed alphabetically, with links to documentation and abstracts, if available.

CSSMOID_APPLE_CERT_POLICY	
CSSMOID_APPLE_EKU_CODE_SIGNING	
CSSMOID_APPLE_EKU_CODE_SIGNING_DEV	
CSSMOID_APPLE_EKU_ICHAT_ENCRYPTION	
CSSMOID_APPLE_EKU_ICHAT_SIGNING	
CSSMOID_AuthorityInfoAccess	
CSSMOID_DOTMAC_CERT_EMAIL_ENCRYPT	
CSSMOID_DOTMAC_CERT_EMAIL_SIGN	
CSSMOID_DOTMAC_CERT_EXTENSION	
CSSMOID_DOTMAC_CERT_IDENTITY	
CSSMOID_DOTMAC_CERT_POLICY	

CSSMOID_EKU_IPSec	
CSSMOID_MicrosoftSGC	
CSSMOID_NetscapeCertSequence	
CSSMOID_NetscapeSGC	
CSSMOID_SubjectInfoAccess	

oidscrl.h

Data Types & Constants

All of the new data types and constants in this header file are listed alphabetically, with links to documentation and abstracts, if available.

CSSMOID_PKIX_OCSP	
CSSMOID_PKIX_OCSP_ARCHIVE_CUTOFF	
CSSMOID_PKIX_OCSP_BASIC	
CSSMOID_PKIX_OCSP_CRL	
CSSMOID_PKIX_OCSP_NOCHECK	
CSSMOID_PKIX_OCSP_NONCE	
CSSMOID_PKIX_OCSP_RESPONSE	
CSSMOID_PKIX_OCSP_SERVICE_LOCATOR	

10.3 Symbol Changes

This article lists the symbols added to `Security.framework` in Mac OS X v10.3.

C Symbols

All of the header files with new symbols are listed alphabetically, with their new symbols described.

AuthSession.h

Functions

All of the new functions in this header file are listed alphabetically, with links to documentation and abstracts, if available.

`SessionCreate`	
`SessionGetInfo`	

Data Types & Constants

All of the new data types and constants in this header file are listed alphabetically, with links to documentation and abstracts, if available.

`callerSecuritySession`	
`errSessionAuthorizationDenied`	
`errSessionInternal`	
`errSessionInvalidAttributes`	
`errSessionInvalidFlags`	
`errSessionInvalidId`	
`errSessionSuccess`	
`noSecuritySession`	
`SecuritySessionId`	

SessionAttributeBits	
SessionCreationFlags	
sessionHasGraphicAccess	
sessionHasTTY	
sessionIsRemote	
sessionIsRoot	
sessionKeepCurrentBootstrap	
sessionWasInitialized	

Authorization.h

Data Types & Constants

All of the new data types and constants in this header file are listed alphabetically, with links to documentation and abstracts, if available.

errAuthorizationBadAddress	

AuthorizationDB.h

Functions

All of the new functions in this header file are listed alphabetically, with links to documentation and abstracts, if available.

AuthorizationRightGet	Retrieves a right definition as a dictionary.
AuthorizationRightRemove	Removes a right from the policy database.
AuthorizationRightSet	Creates or updates a right entry in the policy database.

Data Types & Constants

All of the new data types and constants in this header file are listed alphabetically, with links to documentation and abstracts, if available.

`kAuthorizationComment`	Indicates comments for a rule. The comments appear in the policy database for the administrator to understand what the rule is for. Rule comments are not the same as localized descriptions which are presented to the user.
`kAuthorizationRightRule`	Indicates a rule delegation key. Instead of specifying exact behavior, some rules are shipped with the system and may be used as delegate rules. Use this with any of the delegate rule definition constants.
`kAuthorizationRuleAuthenticateAsAdmin`	Indicates a delegate rule definition constant specifying that the user must authenticate as an administrator.
`kAuthorizationRuleAuthenticateAsSessionUser`	Indicates a delegate rule definition constant specifying that the user must authenticate as the session owner (logged-in user).
`kAuthorizationRuleClassAllow`	Indicates a delegate rule definition constant that always allows the specified right.
`kAuthorizationRuleClassDeny`	Indicates a deleage rule definition constant that always denies the specified right.
`kAuthorizationRuleIsAdmin`	Indicates a delegate rule definition constant specifying that the user must be an administrator.

AuthorizationTags.h

Data Types & Constants

All of the new data types and constants in this header file are listed alphabetically, with links to documentation and abstracts, if available.

`kAuthorizationEnvironmentIcon`	Specifies the name of the authorization item that should be passed into the environment when specifying an alternate icon. The value should be a full path to an image compatible with the NSImage class.
`kAuthorizationEnvironmentPrompt`	Specifies the name of the authorization item that should be passed into the environment when specifying invocation-specific additional text. The value should be a localized UTF8 string.

SecACL.h

Functions

All of the new functions in this header file are listed alphabetically, with links to documentation and abstracts, if available.

`SecACLCopySimpleContents`	Returns the application list, description, and CSSM prompt selector for a given access control list entry.
`SecACLCreateFromSimpleContents`	Creates a new access control list entry from the application list, description, and prompt selector provided and adds it to an item's access object.
`SecACLGetAuthorizations`	Retrieves the CSSM authorization tags of a given access control list entry.
`SecACLGetTypeID`	Returns the unique identifier of the opaque type to which a SecACLRef object belongs.
`SecACLRemove`	Removes the specified access control list entry.
`SecACLSetAuthorizations`	Sets the CSSM authorization tags for a given access control list entry.
`SecACLSetSimpleContents`	Sets the application list, description, and prompt selector for a given access control list entry.

SecBase.h

Data Types & Constants

All of the new data types and constants in this header file are listed alphabetically, with links to documentation and abstracts, if available.

`errSecInvalidPrefsDomain`	The preference domain specified is invalid. This error is available in Mac OS X v10.3 and later.
`errSecTrustNotAvailable`	No trust results are available.

SecKeychain.h

Functions

All of the new functions in this header file are listed alphabetically, with links to documentation and abstracts, if available.

SecKeychainCopyDomainDefault	Retrieves the default keychain from a specified preference domain.
SecKeychainCopyDomainSearchList	Retrieves the keychain search list for a specified preference domain.
SecKeychainGetPreferenceDomain	Gets the current keychain preference domain.
SecKeychainSetDomainDefault	Sets the default keychain for a specified preference domain.
SecKeychainSetDomainSearchList	Sets the keychain search list for a specified preference domain.
SecKeychainSetPreferenceDomain	Sets the keychain preference domain.

Data Types & Constants

All of the new data types and constants in this header file are listed alphabetically, with links to documentation and abstracts, if available.

kSecAuthenticationTypeHTMLForm	Specifies HTML form based authentication. This constant is available in Mac OS X v10.3 and later.
kSecAuthenticationTypeHTTPBasic	Specifies HTTP Basic authentication. This constant is available in Mac OS X v10.3 and later.
kSecPreferencesDomainAlternate	Indicates an alternate preference domain preferences.
kSecPreferencesDomainCommon	Indicates the preferences are common to everyone.
kSecPreferencesDomainSystem	Indicates the system or daemon preference domain preferences.
kSecPreferencesDomainUser	Indicates the user preference domain preferences.
kSecProtocolTypeDAAP	Indicates DAAP. This constant is available in Mac OS X v10.3 and later.
kSecProtocolTypeEPPC	Indicates Remote Apple Events. This constant is available in Mac OS X v10.3 and later.
kSecProtocolTypeFTPProxy	Indicates FTP proxy. This constant is available in Mac OS X v10.3 and later.
kSecProtocolTypeFTPS	Indicates FTP over TLS/SSL. This constant is available in Mac OS X v10.3 and later.
kSecProtocolTypeHTTPProxy	Indicates HTTP proxy. This constant is available in Mac OS X v10.3 and later.
kSecProtocolTypeHTTPS	Indicates HTTP over TLS/SSL. This constant is available in Mac OS X v10.3 and later.
kSecProtocolTypeHTTPSProxy	Indicates HTTPS proxy. This constant is available in Mac OS X v10.3 and later.

kSecProtocolTypeIMAPS	Indicates IMAP4 over TLS/SSL. This constant is available in Mac OS X v10.3 and later.
kSecProtocolTypeIPP	Indicates IPP. This constant is available in Mac OS X v10.3 and later.
kSecProtocolTypeIRCS	Indicates IRC over TLS/SSL. This constant is available in Mac OS X v10.3 and later.
kSecProtocolTypeLDAPS	Indicates LDAP over TLS/SSL. This constant is available in Mac OS X v10.3 and later.
kSecProtocolTypeNNTPS	Indicates NNTP over TLS/SSL. This constant is available in Mac OS X v10.3 and later.
kSecProtocolTypePOP3S	Indicates POP3 over TLS/SSL. This constant is available in Mac OS X v10.3 and later.
kSecProtocolTypeRTSP	Indicates RTSP. This constant is available in Mac OS X v10.3 and later.
kSecProtocolTypeRTSPProxy	Indicates RTSP proxy. This constant is available in Mac OS X v10.3 and later.
kSecProtocolTypeSMB	Indicates SMB. This constant is available in Mac OS X v10.3 and later.
kSecProtocolTypeTelnetS	Indicates Telnet over TLS/SSL. This constant is available in Mac OS X v10.3 and later.
SecPreferencesDomain	Defines constants for the keychain preference domains.

SecureTransport.h

Functions

All of the new functions in this header file are listed alphabetically, with links to documentation and abstracts, if available.

SSLGetConnection	Retrieves an I/O connection—such as a socket or endpoint—for a specific session.
SSLGetDiffieHellmanParams	Retrieves the Diffie-Hellman parameters specified earlier.
SSLGetProtocolVersionEnabled	Retrieves the enabled status of a given protocol.
SSLGetRsaBlinding	Obtains a value indicating whether RSA blinding is enabled.
SSLSetDiffieHellmanParams	Specifies Diffie-Hellman parameters.
SSLSetProtocolVersionEnabled	Sets the allowed SSL protocol versions.

SSLSetRsaBlinding	Enables or disables RSA blinding.

Data Types & Constants

All of the new data types and constants in this header file are listed alphabetically, with links to documentation and abstracts, if available.

errSSLBadConfiguration	A configuration error occurred.
errSSLBadRecordMac	A bad record MAC was encountered.
errSSLDecryptionFail	Decryption failed.
errSSLIllegalParam	An illegal parameter was encountered.
errSSLPeerAccessDenied	Access was denied.
errSSLPeerBadCert	A bad certificate was encountered.
errSSLPeerBadRecordMac	A bad record MAC was encountered.
errSSLPeerCertExpired	The certificate expired.
errSSLPeerCertRevoked	The certificate was revoked.
errSSLPeerCertUnknown	The certificate is unknown.
errSSLPeerDecodeError	A decoding error occurred.
errSSLPeerDecompressFail	Decompression failed.
errSSLPeerDecryptError	A decryption error occurred.
errSSLPeerDecryptionFail	Decryption failed.
errSSLPeerExportRestriction	An export restriction occurred.
errSSLPeerHandshakeFail	The handshake failed.
errSSLPeerInsufficientSecurity	There is insufficient security for this operation.
errSSLPeerInternalError	An internal error occurred.
errSSLPeerNoRenegotiation	No renegotiation is allowed.
errSSLPeerProtocolVersion	A bad protocol version was encountered.
errSSLPeerRecordOverflow	A record overflow occurred.
errSSLPeerUnexpectedMsg	An unexpected message was received.
errSSLPeerUnknownCA	An unknown certificate authority was encountered.
errSSLPeerUnsupportedCert	An unsupported certificate format was encountered.

errSSLPeerUserCancelled	The user canceled the operation.
errSSLRecordOverflow	A record overflow occurred.
kSSLProtocolAll	Specifies all supported versions.

certextensions.h

Data Types & Constants

All of the new data types and constants in this header file are listed alphabetically, with links to documentation and abstracts, if available.

CE_CD_AffiliationChanged	
CE_CD_CACompromise	
CE_CD_CertificateHold	
CE_CD_CessationOfOperation	
CE_CD_KeyCompromise	
CE_CD_Superseded	
CE_CD_Unspecified	
CE_CDNT_FullName	
CE_CDNT_NameRelativeToCrlIssuer	
CE_CR_AffiliationChanged	
CE_CR_CACompromise	
CE_CR_CertificateHold	
CE_CR_CessationOfOperation	
CE_CR_KeyCompromise	
CE_CR_RemoveFromCRL	
CE_CR_Superseded	
CE_CR_Unspecified	
CE_CRLDistPointsSyntax	
CE_CrlDistReasonFlags	
CE_CRLDistributionPoint	

CE_CrlDistributionPointNameType	
CE_CrlNumber	
CE_CrlReason	
CE_DeltaCrl	
CE_DistributionPointName	
CE_IssuingDistributionPoint	
CE_OtherName	
DT_CrlDistributionPoints	
DT_CrlNumber	
DT_CrlReason	
DT_DeltaCrl	
DT_IssuerAltName	
DT_IssuingDistributionPoint	

cssmapple.h

Functions

All of the new functions in this header file are listed alphabetically, with links to documentation and abstracts, if available.

cssmAlgToOid	
cssmOidToAlg	

Data Types & Constants

All of the new data types and constants in this header file are listed alphabetically, with links to documentation and abstracts, if available.

CSSM_ALGID_KEYCHAIN_KEY	
CSSM_ALGID_PKCS12_PBE_ENCR	
CSSM_ALGID_PKCS12_PBE_MAC	
CSSM_APPLE_CRL_END_OF_TIME	

CSSM_APPLE_TP_CRL_OPT_FLAGS	
CSSM_APPLE_TP_CRL_OPTIONS	
CSSM_APPLE_TP_CRL_OPTS_VERSION	
CSSM_APPLE_TP_SMIME_OPTIONS	
CSSM_APPLE_TP_SMIME_OPTS_VERSION	
CSSM_APPLECSPDL_DB_GET_HANDLE	
cssm_appledl_open_parameters_mask	
CSSM_APPLESCPDL_CSP_GET_KEYHANDLE	
CSSM_ATTRIBUTE_PARAM_KEY	
CSSM_ATTRIBUTE_RSA_BLINDING	
CSSM_CERT_STATUS_IS_FROM_NET	The certificate was obtained through some mechanism other than the certificates stored by the operating system and those passed into the SecTrustCreateWithCertificates function. For example, the certificate might have been fetched over a network.
CSSM_DL_DB_RECORD_X509_CRL	
CSSM_ERRCODE_INSUFFICIENT_CLIENT_IDENTIFICATION	
CSSM_KEYATTR_PARTIAL	
CSSM_KEYBLOB_RAW_FORMAT_OPENSSH	
CSSM_KEYBLOB_RAW_FORMAT_OPENSSL	
CSSM_KEYBLOB_RAW_FORMAT_VENDOR_DEFINED	
CSSM_KEYBLOB_RAW_FORMAT_X509	
CSSM_TP_ACTION_FETCH_CERT_FROM_NET	Enable fetching intermediate certificates over the network using http or LDAP.
CSSM_TP_ACTION_FETCH_CRL_FROM_NET	
CSSM_TP_ACTION_LEAF_IS_CA	First certificate is that of a certification authority (CA).
CSSM_TP_ACTION_REQUIRE_CRL_PER_CERT	
CSSM_WORDID_SYMMETRIC_KEY	

CSSM_WORDID_SYSTEM	
CSSMERR_AC_INSUFFICIENT_CLIENT_IDENTIFICATION	
CSSMERR_APPLETP_BAD_CERT_FROM_ISSUER	
CSSMERR_APPLETP_CERT_NOT_FOUND_FROM_ISSUER	
CSSMERR_APPLETP_CRL_BAD_URI	
CSSMERR_APPLETP_CRL_EXPIRED	
CSSMERR_APPLETP_CRL_INVALID_ANCHOR_CERT	
CSSMERR_APPLETP_CRL_NOT_FOUND	
CSSMERR_APPLETP_CRL_NOT_TRUSTED	
CSSMERR_APPLETP_CRL_NOT_VALID_YET	
CSSMERR_APPLETP_CRL_POLICY_FAIL	
CSSMERR_APPLETP_CRL_SERVER_DOWN	
CSSMERR_APPLETP_IDP_FAIL	
CSSMERR_APPLETP_INVALID_ROOT	
CSSMERR_APPLETP_SMIME_BAD_EXT_KEY_USE	
CSSMERR_APPLETP_SMIME_BAD_KEY_USE	
CSSMERR_APPLETP_SMIME_EMAIL_ADDRS_NOT_FOUND	
CSSMERR_APPLETP_SMIME_KEYUSAGE_NOT_CRITICAL	
CSSMERR_APPLETP_SMIME_NO_EMAIL_ADDRS	
CSSMERR_APPLETP_SMIME_SUBJ_ALT_NAME_NOT_CRIT	
CSSMERR_APPLETP_UNKNOWN_CERT_EXTEN	
CSSMERR_APPLETP_UNKNOWN_CRL_EXTEN	
CSSMERR_CL_INSUFFICIENT_CLIENT_IDENTIFICATION	
CSSMERR_CSP_APPLE_INVALID_KEY_END_DATE	
CSSMERR_CSP_APPLE_INVALID_KEY_START_DATE	
CSSMERR_CSP_APPLE_PUBLIC_KEY_INCOMPLETE	
CSSMERR_CSP_APPLE_SIGNATURE_MISMATCH	
CSSMERR_CSP_INSUFFICIENT_CLIENT_IDENTIFICATION	

CSSMERR_CSSM_INSUFFICIENT_CLIENT_IDENTIFICATION	
CSSMERR_DL_INSUFFICIENT_CLIENT_IDENTIFICATION	
CSSMERR_TP_INSUFFICIENT_CLIENT_IDENTIFICATION	
errSecErrnoBase	
errSecErrnoLimit	
kCSSM_APPLEDL_MASK_MODE	
kKeychainSuffix	
kSystemKeychainDir	
kSystemKeychainName	
kSystemUnlockFile	

oidsalg.h

Data Types & Constants

All of the new data types and constants in this header file are listed alphabetically, with links to documentation and abstracts, if available.

CSSMOID_ANSI_DH_EPHEM	
CSSMOID_ANSI_DH_EPHEM_SHA1	
CSSMOID_ANSI_DH_HYBRID1	
CSSMOID_ANSI_DH_HYBRID1_SHA1	
CSSMOID_ANSI_DH_HYBRID2	
CSSMOID_ANSI_DH_HYBRID2_SHA1	
CSSMOID_ANSI_DH_HYBRID_ONEFLOW	
CSSMOID_ANSI_DH_ONE_FLOW	
CSSMOID_ANSI_DH_ONE_FLOW_SHA1	
CSSMOID_ANSI_DH_PUB_NUMBER	
CSSMOID_ANSI_DH_STATIC	
CSSMOID_ANSI_DH_STATIC_SHA1	
CSSMOID_ANSI_MQV1	

CSSMOID_ANSI_MQV1_SHA1	
CSSMOID_ANSI_MQV2	
CSSMOID_ANSI_MQV2_SHA1	
CSSMOID_APPLE_TP_EAP	
CSSMOID_APPLE_TP_REVOCATION_CRL	
CSSMOID_APPLE_TP_REVOCATION_OCSP	
CSSMOID_APPLE_TP_SMIME	
CSSMOID_DSA_CMS	
CSSMOID_DSA_JDK	
CSSMOID_PKCS12_pbeWithSHAAnd128BitRC2CBC	
CSSMOID_PKCS12_pbeWithSHAAnd128BitRC4	
CSSMOID_PKCS12_pbeWithSHAAnd2Key3DESCBC	
CSSMOID_PKCS12_pbeWithSHAAnd3Key3DESCBC	
CSSMOID_PKCS12_pbewithSHAAnd40BitRC2CBC	
CSSMOID_PKCS12_pbeWithSHAAnd40BitRC4	
CSSMOID_PKCS3	
CSSMOID_SHA1WithDSA_CMS	
CSSMOID_SHA1WithDSA_JDK	
CSSMOID_SHA1WithRSA_OIW	

oidsattr.h

Data Types & Constants

All of the new data types and constants in this header file are listed alphabetically, with links to documentation and abstracts, if available.

CSSMOID_PKCS12_certBag	
CSSMOID_PKCS12_crlBag	
CSSMOID_PKCS12_keyBag	
CSSMOID_PKCS12_safeContentsBag	

CSSMOID_PKCS12_secretBag	
CSSMOID_PKCS12_shroudedKeyBag	
CSSMOID_PKCS7_Data	
CSSMOID_PKCS7_DataWithAttributes	
CSSMOID_PKCS7_DigestedData	
CSSMOID_PKCS7_EncryptedData	
CSSMOID_PKCS7_EncryptedPrivateKeyInfo	
CSSMOID_PKCS7_EnvelopedData	
CSSMOID_PKCS7_SignedAndEnvelopedData	
CSSMOID_PKCS7_SignedData	
CSSMOID_PKCS9_CertTypes	
CSSMOID_PKCS9_CrlTypes	
CSSMOID_PKCS9_FriendlyName	
CSSMOID_PKCS9_LocalKeyId	
CSSMOID_PKCS9_SdsiCertificate	
CSSMOID_PKCS9_X509Certificate	
CSSMOID_PKCS9_X509Crl	

oidsbase.h

Data Types & Constants

All of the new data types and constants in this header file are listed alphabetically, with links to documentation and abstracts, if available.

OID_ANSI_X9_42	
OID_ANSI_X9_42_LEN	
OID_ANSI_X9_42_NAMED_SCHEME	
OID_ANSI_X9_42_NAMED_SCHEME_LEN	
OID_ANSI_X9_42_SCHEME	
OID_ANSI_X9_42_SCHEME_LEN	

OID_KP	
OID_KP_LENGTH	
OID_PKCS_11	
OID_PKCS_11_LENGTH	
OID_PKCS_12	
OID_PKCS_12_LENGTH	

oidscert.h

Data Types & Constants

All of the new data types and constants in this header file are listed alphabetically, with links to documentation and abstracts, if available.

CSSMOID_CertIssuer	
CSSMOID_ClientAuth	
CSSMOID_EmailProtection	
CSSMOID_ExtendedKeyUsageAny	
CSSMOID_IssuingDistributionPoint	
CSSMOID_OCSPSigning	
CSSMOID_ServerAuth	
CSSMOID_TimeStamping	
CSSMOID_X509V1IssuerNameStd	
CSSMOID_X509V1SubjectNameStd	

10.2 Symbol Changes

This article lists the symbols added to `Security.framework` in Mac OS X v10.2.

C Symbols

All of the header files with new symbols are listed alphabetically, with their new symbols described.

CipherSuite.h

Data Types & Constants

All of the new data types and constants in this header file are listed alphabetically, with links to documentation and abstracts, if available.

`SSL_DH_anon_EXPORT_WITH_DES40_CBC_SHA`	Session key size conforms to pre-1998 US export restrictions.
`SSL_DH_anon_EXPORT_WITH_RC4_40_MD5`	Session key size conforms to pre-1998 US export restrictions.
`SSL_DH_anon_WITH_3DES_EDE_CBC_SHA`	
`SSL_DH_anon_WITH_DES_CBC_SHA`	
`SSL_DH_anon_WITH_RC4_128_MD5`	
`SSL_DH_DSS_EXPORT_WITH_DES40_CBC_SHA`	
`SSL_DH_DSS_WITH_3DES_EDE_CBC_SHA`	
`SSL_DH_DSS_WITH_DES_CBC_SHA`	
`SSL_DH_RSA_EXPORT_WITH_DES40_CBC_SHA`	Session key size conforms to pre-1998 US export restrictions.
`SSL_DH_RSA_WITH_3DES_EDE_CBC_SHA`	
`SSL_DH_RSA_WITH_DES_CBC_SHA`	
`SSL_DHE_DSS_EXPORT_WITH_DES40_CBC_SHA`	Session key size conforms to pre-1998 US export restrictions.
`SSL_DHE_DSS_WITH_3DES_EDE_CBC_SHA`	

SSL_DHE_DSS_WITH_DES_CBC_SHA	
SSL_DHE_RSA_EXPORT_WITH_DES40_CBC_SHA	Session key size conforms to pre-1998 US export restrictions.
SSL_DHE_RSA_WITH_3DES_EDE_CBC_SHA	
SSL_DHE_RSA_WITH_DES_CBC_SHA	
SSL_FORTEZZA_DMS_WITH_FORTEZZA_CBC_SHA	
SSL_FORTEZZA_DMS_WITH_NULL_SHA	
SSL_NO_SUCH_CIPHERSUITE	
SSL_NULL_WITH_NULL_NULL	
SSL_RSA_EXPORT_WITH_DES40_CBC_SHA	Session key size conforms to pre-1998 US export restrictions.
SSL_RSA_EXPORT_WITH_RC2_CBC_40_MD5	Session key size conforms to pre-1998 US export restrictions.
SSL_RSA_EXPORT_WITH_RC4_40_MD5	Session key size conforms to pre-1998 US export restrictions.
SSL_RSA_WITH_3DES_EDE_CBC_MD5	This value can be specified for SSL 2 but not SSL 3.
SSL_RSA_WITH_3DES_EDE_CBC_SHA	
SSL_RSA_WITH_DES_CBC_MD5	This value can be specified for SSL 2 but not SSL 3.
SSL_RSA_WITH_DES_CBC_SHA	
SSL_RSA_WITH_IDEA_CBC_MD5	This value can be specified for SSL 2 but not SSL 3.
SSL_RSA_WITH_IDEA_CBC_SHA	
SSL_RSA_WITH_NULL_MD5	
SSL_RSA_WITH_NULL_SHA	
SSL_RSA_WITH_RC2_CBC_MD5	This value can be specified for SSL 2 but not SSL 3.
SSL_RSA_WITH_RC4_128_MD5	
SSL_RSA_WITH_RC4_128_SHA	
SSLCipherSuite	Represents the cipher suites available.

SecAccess.h

Functions

All of the new functions in this header file are listed alphabetically, with links to documentation and abstracts, if available.

`SecAccessCopyACLList`	Retrieves all the access control list entries of a given access object.
`SecAccessCopySelectedACLList`	Retrieves selected access control lists from a given access object.
`SecAccessCreate`	Creates a new access object.
`SecAccessCreateFromOwnerAndACL`	Creates a new access object using the owner and access control list you provide.
`SecAccessGetOwnerAndACL`	Retrieves the owner and the access control list of a given access object.
`SecAccessGetTypeID`	Returns the unique identifier of the opaque type to which a SecAccessRef object belongs.

SecBase.h

Data Types & Constants

All of the new data types and constants in this header file are listed alphabetically, with links to documentation and abstracts, if available.

`errSecACLNotSimple`	The access control list is not in standard simple form.
`errSecAuthFailed`	Authorization/Authentication failed.
`errSecBufferTooSmall`	The buffer is too small.
`errSecCreateChainFailed`	The attempt to create a certificate chain failed.
`errSecDataNotAvailable`	The data is not available.
`errSecDataNotModifiable`	The data is not modifiable.
`errSecDataTooLarge`	The data is too large for the particular data type.
`errSecDuplicateCallback`	More than one callback of the same name exists.
`errSecDuplicateItem`	The item already exists.
`errSecDuplicateKeychain`	A keychain with the same name already exists.

errSecInteractionNotAllowed	Interaction with the Security Server is not allowed.
errSecInteractionRequired	User interaction is required.
errSecInvalidCallback	The callback is not valid.
errSecInvalidItemRef	The item reference is invalid.
errSecInvalidKeychain	The keychain is not valid.
errSecInvalidOwnerEdit	An invalid attempt to change the owner of an item.
errSecInvalidSearchRef	The search reference is invalid.
errSecInvalidTrustSetting	The trust setting is invalid.
errSecItemNotFound	The item cannot be found.
errSecKeySizeNotAllowed	The key size is not allowed.
errSecNoAccessForItem	The specified item has no access control.
errSecNoCertificateModule	There is no certificate module available.
errSecNoDefaultKeychain	A default keychain does not exist.
errSecNoPolicyModule	There is no policy module available.
errSecNoStorageModule	There is no storage module available.
errSecNoSuchAttr	The attribute does not exist.
errSecNoSuchClass	The keychain item class does not exist.
errSecNoSuchKeychain	The keychain does not exist.
errSecNotAvailable	No trust results are available.
errSecPolicyNotFound	The policy specified cannot be found.
errSecReadOnly	Read only error.
errSecReadOnlyAttr	The attribute is read only.
errSecWrongSecVersion	The version is incorrect.
SecAccessRef	Identifies a keychain or keychain item's access information.
SecACLRef	Represents information about an access control list entry.
SecCertificateRef	Contains information about a certificate.
SecIdentityRef	Contains information about an identity.
SecKeychainAttribute	Contains keychain attributes.

SecKeychainAttributeInfo	Represents an attribute.
SecKeychainAttributeList	Represents a list of keychain attributes.
SecKeychainAttributePtr	
SecKeychainAttrType	Represents a keychain attribute type.
SecKeychainItemRef	Contains information about a keychain item.
SecKeychainRef	Contains information about a keychain.
SecKeychainSearchRef	Contains information about a keychain search.
SecKeychainStatus	Defines the current status of a keychain.
SecKeyRef	Contains information about a key.
SecPolicyRef	Contains information about a policy.
SecTrustedApplicationRef	Contains information about a trusted application.

SecCertificate.h

Functions

All of the new functions in this header file are listed alphabetically, with links to documentation and abstracts, if available.

SecCertificateAddToKeychain	Adds a certificate to a keychain.
SecCertificateCreateFromData	Creates a certificate object based on the specified data, type, and encoding.
SecCertificateGetCLHandle	Retrieves the certificate library handle from a certificate object.
SecCertificateGetData	Retrieves the data for a certificate.
SecCertificateGetIssuer	Unsupported.
SecCertificateGetItem	Unsupported.
SecCertificateGetSubject	Unsupported.
SecCertificateGetType	Retrieves the type of a specified certificate.
SecCertificateGetTypeID	Returns the unique identifier of the opaque type to which a SecCertificate object belongs.

Data Types & Constants

All of the new data types and constants in this header file are listed alphabetically, with links to documentation and abstracts, if available.

`kSecCertEncodingItemAttr`	Certificate encoding.
`kSecCertTypeItemAttr`	Certificate type.
`kSecIssuerItemAttr`	DER-encoded issuer distinguished name.
`kSecPublicKeyHashItemAttr`	Public key hash.
`kSecSerialNumberItemAttr`	DER-encoded certificate serial number.
`kSecSubjectItemAttr`	DER-encoded subject distinguished name.
`kSecSubjectKeyIdentifierItemAttr`	Subject key identifier.

SecIdentity.h

Functions

All of the new functions in this header file are listed alphabetically, with links to documentation and abstracts, if available.

`SecIdentityCopyCertificate`	Retrieves a certificate associated with an identity.
`SecIdentityCopyPrivateKey`	Retrieves the private key associated with an identity.
`SecIdentityGetTypeID`	Returns the unique identifier of the opaque type to which a SecIdentity object belongs.

SecIdentitySearch.h

Functions

All of the new functions in this header file are listed alphabetically, with links to documentation and abstracts, if available.

`SecIdentitySearchCopyNext`	Finds the next identity matching specified search criteria
`SecIdentitySearchCreate`	Creates a search object for finding identities.
`SecIdentitySearchGetTypeID`	Returns the unique identifier of the opaque type to which a SecIdentitySearch object belongs.

Data Types & Constants

All of the new data types and constants in this header file are listed alphabetically, with links to documentation and abstracts, if available.

`SecIdentitySearchRef`	Contains information about an identity search.

SecKey.h

Functions

All of the new functions in this header file are listed alphabetically, with links to documentation and abstracts, if available.

`SecKeyCreatePair`	Creates an asymmetric key pair and stores it in a keychain.
`SecKeyGetCSSMKey`	Retrieves a pointer to the CSSM_KEY structure containing the key stored in a keychain item.
`SecKeyGetTypeID`	Returns the unique identifier of the opaque type to which a SecKey object belongs.

SecKeychain.h

Functions

All of the new functions in this header file are listed alphabetically, with links to documentation and abstracts, if available.

`SecKeychainAddCallback`	Registers your keychain event callback function
`SecKeychainAddGenericPassword`	Adds a new generic password to a keychain.
`SecKeychainAddInternetPassword`	Adds a new Internet password to a keychain.
`SecKeychainAttributeInfoForItemID`	Obtains tags for all possible attributes of a given item class.
`SecKeychainCopyAccess`	Retrieves the application access of a keychain.
`SecKeychainCopyDefault`	Retrieves a pointer to the default keychain.
`SecKeychainCopySearchList`	Retrieves a keychain search list.
`SecKeychainCopySettings`	Obtains a keychain's settings.
`SecKeychainCreate`	Creates an empty keychain.

SecKeychainDelete	Deletes one or more keychains from the default keychain search list, and removes the keychain itself if it is a file.
SecKeychainFindGenericPassword	Finds the first generic password based on the attributes passed.
SecKeychainFindInternetPassword	Finds the first Internet password based on the attributes passed.
SecKeychainFreeAttributeInfo	Releases the memory acquired by calling the SecKeychainAttributeInfoForItemID function.
SecKeychainGetCSPHandle	Returns the CSSM CSP handle for the given keychain object.
SecKeychainGetDLDBHandle	Returns the CSSM database handle for a given keychain object.
SecKeychainGetPath	Determines the path of a keychain.
SecKeychainGetStatus	Retrieves status information of a keychain.
SecKeychainGetTypeID	Returns the unique identifier of the opaque type to which a SecKeychainRef object belongs.
SecKeychainGetUserInteractionAllowed	Indicates whether Keychain Services functions that normally display a user interaction are allowed to do so.
SecKeychainGetVersion	Determines the version of Keychain Services installed on the user's system.
SecKeychainLock	Locks a keychain.
SecKeychainLockAll	Locks all keychains belonging to the current user.
SecKeychainOpen	Opens a keychain.
SecKeychainRemoveCallback	Unregisters your keychain event callback function.
SecKeychainSetAccess	Sets the application access for a keychain.
SecKeychainSetDefault	Sets the default keychain.
SecKeychainSetSearchList	Specifies the list of keychains to use in the default keychain search list.
SecKeychainSetSettings	Changes the settings of a keychain.
SecKeychainSetUserInteractionAllowed	Enables or disables the user interface for Keychain Services functions that automatically display a user interface.
SecKeychainUnlock	Unlocks a keychain.

Data Types & Constants

All of the new data types and constants in this header file are listed alphabetically, with links to documentation and abstracts, if available.

`kSecAddEvent`	Indicates an item was added to a keychain.
`kSecAddEventMask`	If the bit specified by this mask is set, your callback function is invoked when an item is added to a keychain.
`kSecAuthenticationTypeDefault`	Specifies the default authentication type.
`kSecAuthenticationTypeDPA`	Specifies Distributed Password authentication.
`kSecAuthenticationTypeHTTPDigest`	Specifies HTTP Digest Access authentication.
`kSecAuthenticationTypeMSN`	Specifies Microsoft Network default authentication.
`kSecAuthenticationTypeNTLM`	Specifies Windows NT LAN Manager authentication.
`kSecAuthenticationTypeRPA`	Specifies Remote Password authentication.
`kSecDataAccessEvent`	Indicates a process has accessed a keychain item's data.
`kSecDataAccessEventMask`	If the bit specified by this mask is set, your callback function is invoked when a process accesses a keychain item's data.
`kSecDefaultChangedEvent`	Indicates that a different keychain was specified as the default.
`kSecDefaultChangedEventMask`	If the bit specified by this mask is set, your callback function is invoked when a different keychain is specified as the default.
`kSecDeleteEvent`	Indicates an item was deleted from a keychain.
`kSecDeleteEventMask`	If the bit specified by this mask is set, your callback function is invoked when an item is deleted from a keychain.
`kSecEveryEventMask`	If all the bits are set, your callback function is invoked whenever any event occurs.
`kSecKeychainListChangedEvent`	Indicates the list of keychains has changed.
`kSecKeychainListChangedMask`	If the bit specified by this mask is set, your callback function is invoked when a keychain list is changed.
`kSecLockEvent`	Indicates a keychain was locked.
`kSecLockEventMask`	If the bit specified by this mask is set, your callback function is invoked when a keychain is locked.
`kSecPasswordChangedEvent`	Indicates the keychain password was changed.
`kSecPasswordChangedEventMask`	If the bit specified by this mask is set, your callback function is invoked when the keychain password is changed.

kSecProtocolTypeAFP	Indicates AFP over TCP.
kSecProtocolTypeAppleTalk	Indicates AFP over AppleTalk.
kSecProtocolTypeFTP	Indicates FTP.
kSecProtocolTypeFTPAccount	Indicates a client side FTP account. The usage of this constant is deprecated as of Mac OS X v10.3.
kSecProtocolTypeHTTP	Indicates HTTP.
kSecProtocolTypeIMAP	Indicates IMAP.
kSecProtocolTypeIRC	Indicates IRC.
kSecProtocolTypeLDAP	Indicates LDAP.
kSecProtocolTypeNNTP	Indicates NNTP.
kSecProtocolTypePOP3	Indicates POP3.
kSecProtocolTypeSMTP	Indicates SMTP.
kSecProtocolTypeSOCKS	Indicates SOCKS.
kSecProtocolTypeSSH	Indicates SSH.
kSecProtocolTypeTelnet	Indicates Telnet.
kSecReadPermStatus	Indicates the keychain is readable.
kSecUnlockEvent	Indicates a keychain was successfully unlocked.
kSecUnlockEventMask	If the bit specified by this mask is set, your callback function is invoked when a keychain is unlocked.
kSecUnlockStateStatus	Indicates the keychain is unlocked.
kSecUpdateEvent	Indicates a keychain item was updated.
kSecUpdateEventMask	If the bit specified by this mask is set, your callback function is invoked when a keychain item is updated.
kSecWritePermStatus	Indicates the keychain is writable.
SEC_KEYCHAIN_SETTINGS_VERS1	
SecAuthenticationType	Defines constants you can use to identify the type of authentication to use for an Internet password.
SecKeychainCallback	Defines a pointer to a customized callback function that Keychain Services calls when a keychain event has occurred.
SecKeychainCallbackInfo	Contains information about a keychain event.
SecKeychainEvent	Defines the keychain-related event.

SecKeychainEventMask	Defines bit masks for keychain event constants
SecKeychainSettings	Contains information about keychain settings.
SecProtocolType	Defines the protocol type associated with an AppleShare or Internet password.

SecKeychainItem.h

Functions

All of the new functions in this header file are listed alphabetically, with links to documentation and abstracts, if available.

SecKeychainItemCopyAccess	Copies the access of a given keychain item.
SecKeychainItemCopyAttributesAndData	Retrieves the data and/or attributes stored in the given keychain item.
SecKeychainItemCopyContent	Copies the data and attributes stored in the given keychain item.
SecKeychainItemCopyKeychain	Returns the keychain object of a given keychain item.
SecKeychainItemCreateCopy	Copies a keychain item from one keychain to another.
SecKeychainItemCreateFromContent	Creates a new keychain item from the supplied parameters.
SecKeychainItemDelete	Deletes a keychain item from the default keychain's permanent data store.
SecKeychainItemFreeAttributesAndData	Releases the memory used by the keychain attribute list and/or the keychain data retrieved in a call to SecKeychainItemCopyAttributesAndData.
SecKeychainItemFreeContent	Releases the memory used by the keychain attribute list and/or the keychain data retrieved in a call to the SecKeychainItemCopyContent function.
SecKeychainItemGetDLDBHandle	Returns the CSSM database handle for a given keychain item object.
SecKeychainItemGetTypeID	Returns the unique identifier of the opaque type to which a SecKeychainItemRef object belongs.
SecKeychainItemGetUniqueRecordID	Returns a CSSM unique record for the given keychain item object.
SecKeychainItemModifyAttributesAndData	Updates an existing keychain item after changing its attributes or data.

SecKeychainItemModifyContent	Updates an existing keychain item after changing its attributes and/or data.
SecKeychainItemSetAccess	Sets the access of a given keychain item.

Data Types & Constants

All of the new data types and constants in this header file are listed alphabetically, with links to documentation and abstracts, if available.

kSecAccountItemAttr	Identifies the account attribute. You use this tag to set or get a string that represents the user account. It also applies to generic and AppleShare passwords.
kSecAddressItemAttr	Identifies the address attribute. You use this tag to set or get a value of type string that represents the AppleTalk zone name, or the IP or domain name that represents the server address. This is unique to AppleShare password attributes.
kSecAlias	Indicates an alias.
kSecAppleSharePasswordItemClass	Indicates that the item is an AppleShare password.
kSecAuthenticationTypeItemAttr	Identifies the authentication type attribute.
kSecCertificateEncoding	Indicates a CSSM_CERT_ENCODING type.
kSecCertificateItemClass	Indicates that the item is an X509 certificate.
kSecCertificateType	Indicates a CSSM_CERT_TYPE type.
kSecCommentItemAttr	Identifies the comment attribute. You use this tag to set or get a value of type string that represents a user-editable string containing comments for this item.
kSecCreationDateItemAttr	Identifies the creation date attribute. You use this tag to set or get a value of type UInt32 that indicates the date the item was created.
kSecCreatorItemAttr	Identifies the creator attribute. You use this tag to set or get a value that represents the item's creator.
kSecCrlEncoding	Indicates a CSSM_CRL_ENCODING type.
kSecCrlType	Indicates a CSSM_CRL_TYPE type.
kSecCustomIconItemAttr	Identifies the custom icon attribute.
kSecDescriptionItemAttr	Identifies the description attribute. You use this tag to set or get a value of type string that represents a user-visible string describing this particular kind of item, for example "disk image password".

`kSecGenericItemAttr`	Identifies the generic attribute. You use this tag to set or get a value of untyped bytes that represents a user-defined attribute. This is unique to generic password attributes.
`kSecGenericPasswordItemClass`	Indicates that the item is a generic password.
`kSecInternetPasswordItemClass`	Indicates that the item is an Internet password.
`kSecInvisibleItemAttr`	Identifies the invisible attribute. You use this tag to set or get a value of type Boolean that indicates whether the item is invisible.
`kSecLabelItemAttr`	Identifies the label attribute. You use this tag to set or get a value of type string that represents a user-editable string containing the label for this item.
`kSecModDateItemAttr`	Identifies the modification date attribute. You use this tag to set or get a value of type UInt32 that indicates the last time the item was updated.
`kSecNegativeItemAttr`	Identifies the negative attribute.
`kSecPathItemAttr`	Identifies the path attribute. You use this tag to set or get a value that represents the path. This is unique to Internet password attributes.
`kSecPortItemAttr`	Identifies the port attribute. You use this tag to set or get a value of type UInt32 that represents the Internet port number. This is unique to Internet password attributes.
`kSecProtocolItemAttr`	Identifies the protocol attribute.
`kSecScriptCodeItemAttr`	Identifies the script code attribute. You use this tag to set or get a value of type ScriptCode that represents the script code for all strings. Use of this attribute is deprecated; string attributes should be stored in UTF-8 encoding.
`kSecSecurityDomainItemAttr`	Identifies the security domain attribute. You use this tag to set or get a value that represents the Internet security domain. This is unique to Internet password attributes.
`kSecServerItemAttr`	Identifies the server attribute. You use this tag to set or get a string that represents the Internet server's domain name or IP address. This is unique to Internet password attributes.
`kSecServiceItemAttr`	Identifies the service attribute. You use this tag to set or get a string that represents the service associated with this item, for example, "iTools". This is unique to generic password attributes.
`kSecSignatureItemAttr`	Identifies the server signature attribute. You use this tag to set or get a value of type SecAFPServerSignature that represents the server signature block. This is unique to AppleShare password attributes.
`kSecTypeItemAttr`	Identifies the type attribute. You use this tag to set or get a value that represents the item's type.

`kSecVolumeItemAttr`	Identifies the volume attribute. You use this tag to set or get a value that represents the AppleShare volume. This is unique to AppleShare password attributes.
`SecAFPServerSignature`	Represents a 16-byte Apple File Protocol server signature block.
`SecItemAttr`	Specifies a keychain item's attributes.
`SecItemClass`	Specifies a keychain item's class code.
`SecPublicKeyHash`	Represents a 20-byte public key hash.

SecKeychainSearch.h

Functions

All of the new functions in this header file are listed alphabetically, with links to documentation and abstracts, if available.

`SecKeychainSearchCopyNext`	Finds the next keychain item matching the given search criteria.
`SecKeychainSearchCreateFromAttributes`	Creates a search object matching a list of zero or more attributes.
`SecKeychainSearchGetTypeID`	Returns the unique identifier of the opaque type to which a SecKeychainSearchRef object belongs.

SecPolicy.h

Functions

All of the new functions in this header file are listed alphabetically, with links to documentation and abstracts, if available.

`SecPolicyGetOID`	Retrieves a policy's object identifier.
`SecPolicyGetTPHandle`	Retrieves the trust policy handle for a policy object.
`SecPolicyGetTypeID`	Returns the unique identifier of the opaque type to which a SecPolicy object belongs.
`SecPolicyGetValue`	Retrieves a policy's value.

SecPolicySearch.h

Functions

All of the new functions in this header file are listed alphabetically, with links to documentation and abstracts, if available.

`SecPolicySearchCopyNext`	Retrieves a policy object for the next policy matching specified search criteria.
`SecPolicySearchCreate`	Creates a search object for finding policies.
`SecPolicySearchGetTypeID`	Returns the unique identifier of the opaque type to which a SecPolicySearch object belongs.

Data Types & Constants

All of the new data types and constants in this header file are listed alphabetically, with links to documentation and abstracts, if available.

`SecPolicySearchRef`	Contains information about a policy search.

SecTrust.h

Functions

All of the new functions in this header file are listed alphabetically, with links to documentation and abstracts, if available.

`SecTrustCopyAnchorCertificates`	Retrieves the anchor (root) certificates stored by Mac OS X.
`SecTrustCreateWithCertificates`	Creates a trust management object based on certificates and policies.
`SecTrustEvaluate`	Evaluates trust for the specified certificate and policies.
`SecTrustGetCSSMAnchorCertificates`	Retrieves the CSSM anchor certificates.
`SecTrustGetCssmResult`	Retrieves the CSSM trust result.
`SecTrustGetResult`	Retrieves details on the outcome of a call to the function SecTrustEvaluate.
`SecTrustGetTPHandle`	Retrieves the trust policy handle.
`SecTrustGetTypeID`	Returns the unique identifier of the opaque type to which a SecTrust object belongs.

SecTrustGetUserTrust	Retrieves the user-specified trust setting for a certificate and policy.
SecTrustSetAnchorCertificates	Sets the anchor certificates used when evaluating a trust management object.
SecTrustSetKeychains	Sets the keychains searched for intermediate certificates when evaluating a trust management object.
SecTrustSetParameters	Sets the action and action data for a trust management object.
SecTrustSetUserTrust	Sets the user-specified trust settings of a certificate and policy.
SecTrustSetVerifyDate	Sets the date and time against which the certificates in a trust management object are verified.

Data Types & Constants

All of the new data types and constants in this header file are listed alphabetically, with links to documentation and abstracts, if available.

kSecTrustResultConfirm	Confirmation from the user is required before proceeding. This value may be returned by the SecTrustEvaluate function or stored as part of the user trust settings. In the Keychain Access utility, this value is termed “Ask Permission.”
kSecTrustResultDeny	The user specified that the certificate should not be trusted. This value may be returned by the SecTrustEvaluate function or stored as part of the user trust settings. In the Keychain Access utility, this value is termed “Never Trust.”
kSecTrustResultFatalTrustFailure	Trust denied; no simple fix is available.
kSecTrustResultInvalid	Invalid setting or result. Usually, this result indicates that the SecTrustEvaluate function did not complete successfully.
kSecTrustResultOtherError	A failure other than that of trust evaluation; for example, an internal failure of the SecTrustEvaluate function. This value may be returned by the SecTrustEvaluate function but not stored as part of the user trust settings.
kSecTrustResultProceed	The user indicated that you may trust the certificate for the purposes designated in the specified policies.
kSecTrustResultRecoverableTrustFailure	Trust denied; retry after changing settings.
kSecTrustResultUnspecified	The user did not specify a trust setting.

SecTrustRef	Contains information about trust management.
SecTrustResultType	Specifies the trust result type.
SecTrustUserSetting	Represents user-specified trust settings.

SecTrustedApplication.h

Functions

All of the new functions in this header file are listed alphabetically, with links to documentation and abstracts, if available.

SecTrustedApplicationCopyData	Retrieves the data of a trusted application object.
SecTrustedApplicationCreateFromPath	Creates a trusted application object based on the application specified by path.
SecTrustedApplicationGetTypeID	Returns the unique identifier of the opaque type to which a SecTrustedApplication object belongs.
SecTrustedApplicationSetData	Sets the data of a given trusted application object.

SecureTransport.h

Functions

All of the new functions in this header file are listed alphabetically, with links to documentation and abstracts, if available.

SSLAddDistinguishedName	Unsupported.
SSLClose	Terminates the current SSL session.
SSLDisposeContext	Disposes of an SSL session context.
SSLGetAllowsAnyRoot	Obtains a value specifying whether an unknown root is allowed.
SSLGetAllowsExpiredCerts	Retrieves the value specifying whether expired certificates are allowed.
SSLGetAllowsExpiredRoots	Retrieves the value indicating whether expired roots are allowed.
SSLGetBufferedReadSize	Determines how much data is available to be read.
SSLGetClientCertificateState	Retrieves the exchange status of the client certificate.

SSLGetEnableCertVerify	Determines whether peer certificate chain validation is currently enabled.
SSLGetEnabledCiphers	Determines which SSL cipher suites are currently enabled.
SSLGetNegotiatedCipher	Retrieves the cipher suite negotiated for this session.
SSLGetNegotiatedProtocolVersion	Obtains the negotiated protocol version of the active session.
SSLGetNumberEnabledCiphers	Determines the number of cipher suites currently enabled.
SSLGetNumberSupportedCiphers	Determines the number of cipher suites supported.
SSLGetPeerCertificates	Retrieves a peer certificate.
SSLGetPeerDomainName	Retrieves the peer domain name specified previously.
SSLGetPeerDomainNameLength	Determines the length of a previously set peer domain name.
SSLGetPeerID	Retrieves the current peer ID data.
SSLGetProtocolVersion	Gets the SSL protocol version. This function is deprecated.
SSLGetSessionState	Retrieves the state of an SSL session.
SSLGetSupportedCiphers	Determines the values of the supported cipher suites.
SSLGetTrustedRoots	Retrieves the current list of trusted root certificates.
SSLHandshake	Performs the SSL handshake.
SSLNewContext	Creates a new SSL session context.
SSLRead	Performs a normal application-level read operation.
SSLSetAllowsAnyRoot	Specifies whether root certificates from unrecognized certification authorities are allowed.
SSLSetAllowsExpiredCerts	Specifies whether certificate expiration times are ignored.
SSLSetAllowsExpiredRoots	Specifies whether expired root certificates are allowed.
SSLSetCertificate	Specifies this connection's certificate or certificates.
SSLSetClientSideAuthenticate	Specifies the requirements for client-side authentication.
SSLSetConnection	Specifies an I/O connection for a specific session.
SSLSetEnableCertVerify	Enables or disables peer certificate chain validation.
SSLSetEnabledCiphers	Specifies a restricted set of SSL cipher suites to be enabled by the current SSL session context.
SSLSetEncryptionCertificate	Specifies the encryption certificates used for this connection.

SSLSetIOFuncs	Specifies callback functions that perform the network I/O operations.
SSLSetPeerDomainName	Specifies the fully qualified domain name of the peer.
SSLSetPeerID	Specifies data that is sufficient to uniquely identify the peer of the current session.
SSLSetProtocolVersion	Sets the SSL protocol version. This function is deprecated.
SSLSetTrustedRoots	Augments or replaces the default set of trusted root certificates for this session.
SSLWrite	Performs a normal application-level write operation.

Data Types & Constants

All of the new data types and constants in this header file are listed alphabetically, with links to documentation and abstracts, if available.

errSSLBadCert	Bad certificate format.
errSSLBadCipherSuite	A bad SSL cipher suite was encountered.
errSSLBufferOverflow	An insufficient buffer was provided.
errSSLCertExpired	The certificate chain had an expired certificate.
errSSLCertNotYetValid	The certificate chain had a certificate that is not yet valid.
errSSLClosedAbort	The connection closed due to an error.
errSSLClosedGraceful	The connection closed gracefully.
errSSLClosedNoNotify	The server closed the session with no notification.
errSSLCrypto	An underlying cryptographic error was encountered.
errSSLFatalAlert	A fatal alert was encountered.
errSSLInternal	Internal error.
errSSLLast	
errSSLModuleAttach	Module attach failure.
errSSLNegotiation	The cipher suite negotiation failed.
errSSLNoRootCert	No root certificate for the certificate chain.
errSSLProtocol	SSL protocol error.
errSSLSessionNotFound	An attempt to restore an unknown session failed.

errSSLUnknownRootCert	Certificate chain is valid, but root is not trusted.
errSSLWouldBlock	Function is blocked; waiting for I/O. This is not fatal.
errSSLXCertChainInvalid	Invalid certificate chain.
kAlwaysAuthenticate	Indicates that client-side authentication is required.
kNeverAuthenticate	Indicates that client-side authentication is not required. (Default.)
kSSLAborted	The connection aborted.
kSSLClientCertNone	Indicates that the server hasn't asked for a certificate and that the client hasn't sent one.
kSSLClientCertRejected	Indicates that the client sent a certificate but the certificate failed validation. This value is seen only on the server side. The server application can inspect the certificate using the function SSLGetPeerCertificates.
kSSLClientCertRequested	Indicates that the server has asked for a certificat, but the client has not sent it.
kSSLClientCertSent	Indicates that the server asked for a certificate, the client sent one, and the server validated it. The application can inspect the certificate using the function SSLGetPeerCertificates.
kSSLClosed	The connection closed normally.
kSSLConnected	The SSL handshake is complete; the connection is ready for normal I/O.
kSSLHandshake	The SSL handshake is in progress.
kSSLIdle	No I/O has been performed yet.
kSSLProtocol2	Specifies that only the SSL 2.0 protocol may be negotiated.
kSSLProtocol3	Specifies that the SSL 3.0 protocol is preferred; the SSL 2.0 protocol may be negotiated if the peer cannot use the SSL 3.0 protocol.
kSSLProtocol3Only	Specifies that only the SSL 3.0 protocol may be negotiated; fails if the peer tries to negotiate the SSL 2.0 protocol.
kSSLProtocolUnknown	Specifies that no protocol has been or should be negotiated or specified; use default.
kTLSProtocol1	Specifies that the TLS 1.0 protocol is preferred but lower versions may be negotiated.
kTLSProtocol1Only	Specifies that only the TLS 1.0 protocol may be negotiated.
kTryAuthenticate	Indicates that client-side authentication should be attempted. There is no error if the client doesn't have a certificate.

SSLAuthenticate	Represents the requirements for client-side authentication.
SSLClientCertificateState	Represents the status of client certificate exchange.
SSLConnectionRef	Represents a pointer to an opaque I/O connection object.
SSLContextRef	Represents a pointer to an opaque SSL session context object.
SSLProtocol	Represents the SSL protocol version.
SSLReadFunc	Defines a pointer to a customized read function that Secure Transport calls to read data from the connection.
SSLSessionState	Represents the state of an SSL session.
SSLWriteFunc	Defines a pointer to a customized write function that Secure Transport calls to write data to the connection.

certextensions.h

Data Types & Constants

All of the new data types and constants in this header file are listed alphabetically, with links to documentation and abstracts, if available.

CE_DataAndType	

cssmapple.h

Functions

All of the new functions in this header file are listed alphabetically, with links to documentation and abstracts, if available.

cssmPerror	

Data Types & Constants

All of the new data types and constants in this header file are listed alphabetically, with links to documentation and abstracts, if available.

CSSM_ACL_KEYCHAIN_PROMPT_CURRENT_VERSION	
CSSM_ACL_KEYCHAIN_PROMPT_REQUIRE_PASSPHRASE	
CSSM_ACL_KEYCHAIN_PROMPT_SELECTOR	

CSSM_APPLE_CL_CSR_REQUEST	
CSSM_APPLE_TP_ACTION_DATA	
CSSM_APPLE_TP_ACTION_FLAGS	Specifies options for the AppleX509TP trust policy module's default action.
CSSM_APPLE_TP_ACTION_VERSION	
CSSM_APPLE_TP_CERT_REQUEST	
CSSM_APPLE_TP_NAME_OID	
CSSM_APPLE_TP_SSL_OPTIONS	
CSSM_APPLE_TP_SSL_OPTS_VERSION	
CSSM_APPLEX509CL_OBTAIN_CSR	
CSSM_APPLEX509CL_VERIFY_CSR	
CSSM_ASC_OPTIMIZE_ASCII	
CSSM_ASC_OPTIMIZE_DEFAULT	
CSSM_ASC_OPTIMIZE_SECURITY	
CSSM_ASC_OPTIMIZE_SIZE	
CSSM_ASC_OPTIMIZE_TIME	
CSSM_ASC_OPTIMIZE_TIME_SIZE	
CSSM_ATTRIBUTE_ASC_OPTIMIZATION	
CSSM_CERT_STATUS_EXPIRED	The certificate has expired.
CSSM_CERT_STATUS_IS_IN_ANCHORS	This certificate was found in the system's store of anchor certificates (see SecTrustSetAnchorCertificates).
CSSM_CERT_STATUS_IS_IN_INPUT_CERTS	This is one of the certificates included in the array of certificates passed to the SecTrustCreateWithCertificates function.
CSSM_CERT_STATUS_IS_ROOT	The certificate is a root certificate. If this bit is set but the CSSM_CERT_STATUS_IS_IN_ANCHORS bit is not, then this is an untrusted anchor.
CSSM_CERT_STATUS_NOT_VALID_YET	The certificate is not yet valid. In addition to the expiration, or "Not Valid After," date and time, each certificate has a "Not Valid Before" date and time.
CSSM_DL_DB_RECORD_METADATA	

CSSM_DL_DB_RECORD_USER_TRUST	
CSSM_DL_DB_RECORD_X509_CERTIFICATE	
CSSM_EVIDENCE_FORM_APPLE_CERT_INFO	
CSSM_EVIDENCE_FORM_APPLE_CERTGROUP	
CSSM_EVIDENCE_FORM_APPLE_CUSTOM	
CSSM_EVIDENCE_FORM_APPLE_HEADER	
CSSM_TP_ACTION_ALLOW_EXPIRED	Ignore the expiration date and time for all certificates.
CSSM_TP_ACTION_ALLOW_EXPIRED_ROOT	Ignore the expiration date and time for root certificates only.
CSSM_TP_APPLE_CERT_STATUS	Specifies the status of a certificate.
CSSM_TP_APPLE_EVIDENCE_HEADER	
CSSM_TP_APPLE_EVIDENCE_INFO	Contains information about a certificate evaluation.
CSSM_TP_APPLE_EVIDENCE_VERSION	
CSSM_WORDID__RESERVED_1	
CSSMERR_APPLETP_HOSTNAME_MISMATCH	
CSSMERR_APPLETP_INVALID_AUTHORITY_ID	
CSSMERR_APPLETP_INVALID_CA	
CSSMERR_APPLETP_INVALID_EXTENDED_KEY_USAGE	
CSSMERR_APPLETP_INVALID_ID_LINKAGE	
CSSMERR_APPLETP_INVALID_KEY_USAGE	
CSSMERR_APPLETP_INVALID_SUBJECT_ID	
CSSMERR_APPLETP_NO_BASIC_CONSTRAINTS	
CSSMERR_APPLETP_PATH_LEN_CONSTRAINT	
CSSMERR_APPLETP_UNKNOWN_CRITICAL_EXTEN	

cssmerr.h

Data Types & Constants

All of the new data types and constants in this header file are listed alphabetically, with links to documentation and abstracts, if available.

CSSMERR_CSSM_FUNCTION_NOT_IMPLEMENTED	
CSSMERR_DL_INVALID_CL_HANDLE	
CSSMERR_DL_INVALID_CSP_HANDLE	
CSSMERR_DL_INVALID_DB_LIST_POINTER	
CSSMERR_DL_INVALID_DL_HANDLE	

oidsalg.h

Data Types & Constants

All of the new data types and constants in this header file are listed alphabetically, with links to documentation and abstracts, if available.

CSSMOID_APPLE_TP_CSR_GEN	
CSSMOID_APPLE_TP_LOCAL_CERT_GEN	
CSSMOID_SHA1	

10.1 Symbol Changes

This article lists the symbols added to `Security.framework` in Mac OS X v10.1.

C Symbols

All of the header files with new symbols are listed alphabetically, with their new symbols described.

cssmapple.h

Data Types & Constants

All of the new data types and constants in this header file are listed alphabetically, with links to documentation and abstracts, if available.

`CSSM_ALGID_ASC`	
`CSSM_ALGID_FEE`	
`CSSM_ALGID_FEE_MD5`	
`CSSM_ALGID_FEE_SHA1`	
`CSSM_ALGID_FEED`	
`CSSM_ALGID_FEEDEXP`	
`CSSM_ALGID_SHA1HMAC_LEGACY`	
`CSSM_ATTRIBUTE_FEE_CURVE_TYPE`	
`CSSM_ATTRIBUTE_FEE_PRIME_TYPE`	
`CSSM_ATTRIBUTE_PUBLIC_KEY`	
`CSSM_ATTRIBUTE_VENDOR_DEFINED`	
`CSSM_FEE_CURVE_TYPE_DEFAULT`	
`CSSM_FEE_CURVE_TYPE_MONTGOMERY`	
`CSSM_FEE_CURVE_TYPE_WEIERSTRASS`	
`CSSM_FEE_PRIME_TYPE_DEFAULT`	

CSSM_FEE_PRIME_TYPE_FEE	
CSSM_FEE_PRIME_TYPE_GENERAL	
CSSM_FEE_PRIME_TYPE_MERSENNE	
CSSMERR_APPLEDL_INCOMPATIBLE_DATABASE_BLOB	
CSSMERR_APPLEDL_INCOMPATIBLE_KEY_BLOB	
CSSMERR_APPLEDL_INVALID_DATABASE_BLOB	
CSSMERR_APPLEDL_INVALID_KEY_BLOB	

oidsalg.h

Data Types & Constants

All of the new data types and constants in this header file are listed alphabetically, with links to documentation and abstracts, if available.

CSSMOID_APPLE_ECDSA	
CSSMOID_APPLE_FEE_SHA1	
CSSMOID_APPLE_FEED	
CSSMOID_APPLE_FEEDEXP	

Document Revision History

This table describes the changes to *Security Reference Update*.

Date	Notes
2007-07-18	Updated with the symbols added to the Security framework in Mac OS X v10.5.
2005-04-29	New document that summarizes the symbols added to the Security framework in Mac OS X v10.4.
	New document that summarizes the symbols added to the Security framework in Mac OS X v10.4.