WebCryptoAPI Overview

Daniel Roesler - HTML5DevConf - October 2015

What is WebCryptoAPI?

Web Cryptography API is a standard that allows access to cryptographic primitives in the browser.

http://www.w3.org/TR/WebCryptoAPI/

```
==Random number generation==
window.crypto.getRandomValues(...)

==Key handling==
window.crypto.subtle.generateKey(...)
window.crypto.subtle.importKey(...)
window.crypto.subtle.exportKey(...)
window.crypto.subtle.wrapKey(...)
window.crypto.subtle.unwrapKey(...)
window.crypto.subtle.deriveKey(...)
window.crypto.subtle.deriveBits(...)
```

```
==Signatures==
window.crypto.subtle.sign(...)
window.crypto.subtle.verify(...)

==Encryption==
window.crypto.subtle.encrypt(...)
window.crypto.subtle.decrypt(...)

==Hashing==
window.crypto.subtle.digest(...)
```

random number generation

crypto.getRandomValues(...)

```
//Generate some random data

window.crypto.getRandomValues(
array //e.g. new Uint8Array(16)
)
```

key handling

crypto.subtle.generateKey(...)

```
//Generate a new CryptoKey

window.crypto.subtle.generateKey(
{name: 'AES-GCM', …}, //key algorithm
false, //extractable
["encrypt", "decrypt"] //usage settings
)
```

crypto.subtle.importKey(...)

```
//Create a new CryptoKey from raw data
window.crypto.subtle.importKey(
"jwk", //import format
rawData, //ArrayBuffer or {}
{name: 'ECDSA', …}, //key algorithm
false, //extractable
["encrypt", "decrypt"] //usage settings
)
```

crypto.subtle.exportKey(...)

```
//Dump a CryptoKey (must be extractable)

window.crypto.subtle.exportKey(
"jwk", //export format ("jwk", "raw", …)
key, //CryptoKey you want to export
)
```

crypto.subtle.wrapKey(...)

```
//Encrypt a CryptoKey with another CryptoKey
window.crypto.subtle.wrapKey(
"raw", //export format
key, //CrytoKey to wrap
wrappingKey, //wrapping CryptoKey
{name: 'AES-KW', …}, //wrapping algorithm
)
//(used to export a key for storage)
```

crypto.subtle.unwrapKey(...)

```
//Decrypt a previously wrapped CryptoKey
window.crypto.subtle.unwrapKey(
"raw", //wrapped format
buffer, //wrapped key ArrayBuffer
wrappingKey, //wrapping CryptoKey
{name: 'AES-KW', …}, //wrapping algorithm
{name: 'AES-GCM', …}, //wrapped key algorithm
false, //extractable
["encrypt", "decrypt"] //usage settings
)
```

crypto.subtle.deriveKey(...)

```
//Create a new CryptoKey from key exchange
window.crypto.subtle.deriveKey(
{name: 'ECDH', …}, //public CryptoKey
baseKey, //private CryptoKey
{name: 'AES-GCM', …}, //key algorithm
false, //extractable
["encrypt", "decrypt"] //usage settings
)
```

crypto.subtle.deriveBits(...)

```
//Generates shared secret from key exchange

window.crypto.subtle.deriveBits(
{name: 'ECDH', …}, //public CryptoKey
baseKey, //private CryptoKey
256, //length of bits
)
//(Used to generate symmetric keys from asymmetric or
// password-based key derivation functions)
```

signatures

crypto.subtle.sign(...)

```
//Cryptographically sign some data

window.crypto.subtle.sign(
{name: 'ECDSA', …}, //key algorithm
signingKey, //private CryptoKey
arrayBuffer, //the data to sign
)
```

crypto.subtle.verify(...)

```
//Verify a signature on some data

window.crypto.subtle.verify(
{name: 'ECDSA', …}, //key algorithm
signingKey, //public CryptoKey
signature, //ArrayBuffer signature
data, //ArrayBuffer data
)
```

encryption

crypto.subtle.encrypt(...)

```
//Encrypt some data

window.crypto.subtle.encrypt(
{name: 'AES-GCM', …}, //key algorithm
key, //CryptoKey
arrayBuffer, //data to encrypt
)
```

crypto.subtle.decrypt(...)

```
//Decrypt some previously encrypted data

window.crypto.subtle.decrypt(
{name: 'AES-GCM', …}, //key algorithm
key, //CryptoKey
arrayBuffer, //encrypted data
)
```

hashing

crypto.subtle.digest(...)

```
//Cryptographically hash some data

window.crypto.subtle.digest(
{name: 'SHA-256'}, //digest algorithm
arrayBuffer, //data to hash
)
```

algorithms

Algorithm Overview

=sign()/verify()=

ECDSA HMAC

RSASSA-PKCS1-v1_5

~~RSA-PSS~~ ~~AES-CMAC~~

=encrypt()/decrypt()=

AES-GCM

RSA-OAEP AES-CTR

AES-CBC ~~AES-CFB~~

=digest()=

SHA-256 SHA-384

SHA-512 SHA-1

=deriveKey/Bits()=

ECDH PBKDF2

~~DH~~ ~~CONCAT~~

~~HKDF-CTR~~

=wrapKey()/unwrapKey()=

AES-GCM AES-KW

RSA-OAEP AES-CTR AES-CBC ~~AES-CFB~~

examples

https://github.com/diafygi/webcrypto-examples

browser test

https://diafygi.github.io/webcrypto-examples/

Questions?

Daniel Roesler, Co-founder & CTO, UtilityAPI

My side projects: https://daylightpirates.org