# Mechanism for per token and per session seal keys

The per token and per session keys are generated internally and automatically upon token initialization and session creation respectively. The keys are NEVER exportable, and may only be used with the CKM\_SEAL\_KEY mechanism. They may have any underlying type that is compatible with the cryptographic mechanism actually used to seal keys.

## Key Identifier for token seal key

* CKA\_GLOBAL = 2
* CKA\_OBJECT\_ID = [xxxx]

This key is persistent across sessions, is generated when the token is initialized, but is zeroized and replaced if the token is re-initialized.

## Key identifier for session seal key

* CKA\_GLOBAL = 2
* CKA\_OBJECT\_ID = [xxxx]

This key is valid only during a specific session and is generated at the beginning of each session zeroized during the conclusion of that session.

# Mechanism Definition

CKM\_SEAL\_KEY

The mechanism is used to wrap and unwrap key material that would normally be unexportable. It may only be used with one of the keys identified above. The actual cryptographic mechanism used to perform the wrap and unwrap operations is implementation dependent, as is the type of key used to do the wrap and unwrap.

There are no parameters for this mechanism.

This mechanism provides a standard method for moving sensitive key material to storage external to the token, but where the material is still cryptographically tied to the token. Since the seal keys are zeroized upon session close or token zeroization, any sealed key material external to the token becomes no better than random bits upon those actions.

When used with one of the seal keys, the mechanism will allow C\_WrapKey to wrap any existing key, whether exportable or not, and will output an opaque byte array. The information wrapped includes all pertinent key attributes in an implementation dependent form. The unwrap operation reverses the wrap operation and MUST produce exactly the same object as was wrapped.

When used with any key other than one of the seal keys, the normal restrictions on wrapping MUST be enforced.

When used with C\_UnwrapKey, ulAttributeCount should be 0. Any specified attributes are silently ignored.

When C\_UnwrapKey is used with one of the seal keys, the CKA\_EXPORTABLE, CKA\_NEVER\_EXTRACTABLE, CKA\_LOCAL, and CKA\_ALWAYS\_SENSITIVE attributes are set from the data included with the wrapped key. If used with any other key, then those attributes are set as described in the C\_UnwrapKey documentation.