

4.3.1 Mutexes

MRAPI mutexes provide exclusive locking functionality through the use of a flag (just like a binary semaphore). MRAPI mutexes support recursive locking. Recursive locking means that once a mutex is locked, lock may be called again before unlock is called. For each call to lock, a unique lock key is returned. This lock key must be passed in to the call to unlock. The implementation uses the keys to match the order of the lock/unlock calls. Recursive locking is disabled by default and can be enabled by setting the `MRAPI_MUTEX_RECURSIVE` attribute when the mutex is created. When the mutex is not recursive, the lock_keys are ignored.

If `mrapi_mutex_lock()` is called and the lock is currently locked and recursive locking is disabled, then the function will block until the lock is available. It is safer to use `mrapi_mutex_trylock()` unless you are certain that the lock will eventually succeed. Otherwise, a thread of execution can block forever waiting for the lock.