NAME
kernacc, **useracc** - check memory regions for accessibility

SYNOPSIS
#include <sys/param.h>
#include <sys/proc.h>
#include <vm/vm.h>
#include <vm/vm_extern.h>

int
kernacc(*void *addr*, *int len*, *int rw*);

int
useracc(*void *addr*, *int len*, *int rw*);

DESCRIPTION
The **kernacc**() and **useracc**() functions check whether operations of the type specified in *rw* are permitted in the range of virtual addresses given by *addr* and *len*. The possible values of *rw* are any bitwise combination of VM_PROT_READ, VM_PROT_WRITE and VM_PROT_EXECUTE. **kernacc**() checks addresses in the kernel address space, while **useracc**() considers *addr* to represent an user space address. The process context to use for this operation is taken from the global variable *curproc*.

RETURN VALUES
Both functions return boolean true if the type of access specified by *rw* is permitted. Otherwise boolean false is returned.

BUGS
The process pointer should be passed in as an argument to **useracc**().