

NAME

kinfo_getvmmap - function for getting per-process memory map information

LIBRARY

System Utilities Library (libutil, -lutil)

SYNOPSIS

```
#include <sys/types.h>
```

```
#include <sys/user.h>
```

```
#include <libutil.h>
```

```
struct kinfo_vmentry *
```

```
kinfo_getvmmap(pid_t pid, int *cntp);
```

DESCRIPTION

This function is used for obtaining virtual memory mapping information of a particular process.

The *pid* field contains the process identifier. This should be the a process that you have privilege to access. The *cntp* field is allows the caller to know how many records are returned.

This function is a wrapper around sysctl(3) with the KERN_PROC_VMMAP mib. While the kernel returns a packed structure, this function expands the data into a fixed record format.

RETURN VALUES

On success the **kinfo_getvmmap()** function returns a pointer to an array of *struct kinfo_vmentry* structures as defined by *<sys/user.h>*. The array was obtained by an internal call to malloc(3) and must be freed by the caller with a call to free(3). On failure the **kinfo_getvmmap()** function returns NULL.

SEE ALSO

free(3), kinfo_getfile(3), malloc(3)

HISTORY

The **kinfo_getvmmap()** function first appeared in FreeBSD 7.0.

CAVEATS

kinfo_getvmmap() calls can cause significant CPU consumption because calculating the residency counts requires iterating over all memory pages of the process. Setting the sysctl variable *kern.proc_vmmap_skip_resident_count* to 1 prevents this calculation.