NAME

prng - Kernel pseudo-random number generators

SYNOPSIS

#include <sys/prng.h>

uint32_t
prng32(*void*);

uint32_t
prng32_bounded(*uint32_t bound*);

uint64_t
prng64(*void*);

uint64_t
prng64_bounded(*uint64_t bound*);

DESCRIPTION

GENERIC PRNG ROUTINES

prng is a family of fast, *non-cryptographic* pseudo-random number generators. Unlike random(9), **prng32**(), **prng32_bounded**(), **prng64**(), and **prng64_bounded**() avoid shared global state, removing unnecessary contention on SMP systems. The routines are not explicitly tied to any specific implementation, and may produce different specific sequences on different hosts, reboots, or versions of FreeBSD. Different CPUs in SMP systems are guaranteed to produce different sequences of integers.

For *cryptographically secure* random numbers generated by the random(4) kernel cryptographically secure random number generator subsystem, see arc4random(9).

prng32()
: Generate a 32-bit integer uniformly distributed in [0, 2^32-1].

prng32_bounded(*bound*)
: Generate an integer uniformly in the range [0, bound-1].

prng64()
: Generate a 64-bit integer uniformly distributed in [0, 2^64-1].

prng64_bounded(*bound*)
: Generate an integer uniformly in the range [0, bound-1].

These routines are not reentrant; they are not safe to use in interrupt handlers ("interrupt filters" in bus_setup_intr(9) terminology). They are safe to use in all other kernel contexts, including interrupt threads ("ithreads").

REPRODUCIBLE PRNG APIS

In addition to these per-CPU helpers, the <*sys/prng.h*> header also exposes the entire API of the PCG family of PRNGs as inline functions. The PCG-C API is described in full at **https://www.pcg-random.org/using-pcg-c.html**.

HISTORY

prng was introduced in FreeBSD 13.