

NAME

pollinate – an Entropy-as-a-Service client

SYNOPSIS

pollinate [-c|--curl-opts CURL_OPTS] [-d|--device DEVICE] [-i|--insecure] [-n|--no-challenge] [-r|--reseed] [-s|--server SERVER1 [-s|--server SERVER2] ...]

DESCRIPTION

pollinate is an entropy as a service client. It will fetch a random seed from one or more remote servers, and feed those into a (pseudo) random number generator. This is often useful in cloud and virtual machine environments with limited sources of entropy.

OPTIONS**-b|--binary**

rather than producing hex text sha512sum(1) output, convert each two hex digits into a byte of binary data

-c|--curl-opts CURL_OPTS

options to pass through to **curl**(1)

-d|--device DEVICE

device to seed (default is */dev/urandom*); note, you can use - to print to stdout

-n|--no-challenge

disable the challenge/response portion of the protocol; this enables a compatibility mode to speak to any entropy server, like random.org or news.google.com

-i|--insecure

ignore SSL/TLS certificate errors (use of this option is highly discouraged)

-p|--pool SERVER

remote **pollen**(8) server pool to interact with; this option can be specified multiple times to interact with multiple servers in a pool

-r|--reseed

by default, **pollinate** technically only needs to run successfully once ever; this option should be used if someone wants run more than once, and re-seed the PRNG

-s|--server SERVER

remote **pollen**(8) server to interact with; if this option is used, the pool is ignored, and only a single server is used

--strict handle network errors strictly and exit non-zero for any network error encountered, otherwise default is to warn

-t|--testing

test communications with the **pollen**(8) server; note that this will NOT actually seed your PRNG, but only verify that the local client can communicate with the remote servers; note that this option forces DEVICE to stdout

-q|--quiet

silence any log messages

-w|--wait WAIT

time in seconds you're willing to wait for a response from the server, uses **timeout**(1)

--print-user-agent

Print the user-agent that is used and exit immediately.

SEE ALSO**pollen(8), timeout(1)****AUTHOR**

This manpage and the utility were written by Dustin Kirkland <kirkland@ubuntu.com> for Ubuntu systems (but may be used by others). Permission is granted to copy, distribute and/or modify this document under the terms of the GNU General Public License, Version 3 published by the Free Software Foundation.

On Debian systems, the complete text of the GNU General Public License can be found in `/usr/share/common-licenses/GPL`, or on the web at <http://www.gnu.org/licenses/gpl.txt>.