Générer des nombres aléatoires avec Hasard

Victor Stinner

OSDCfr, octobre 2009

Usage en sécurité

- Confidentialité : SSL/TLS
- Authentification : mot de passe, certificat
- Chiffrement par flot
- Autres : loterie, roulette, machine à sous

Propriétés pour la sécurité

- empêcher de prédire les nombres suivants et l'état interne à partir de la sortie
- empêcher de prédire les nombres précédents (et les nombres suivants s'il y a une source d'entropie) à partir de l'état interne

Simulation

- Jeux vidéos
- Simulation physique
- Monte Carlo (approximation numérique)

Propriétés pour la simulation

- Rapide
- Distribution uniforme (sur plusieurs dimensions)
- Reproductible (déterministe)
- Longue période (2^{128} ou plus)

Bugs courants

- rand() % n # modulo
- rand() & n # masque (et)
- Faible entropie : getpid(), time(NULL)
- Nouvelle graine pour chaque nombre généré, bug PHP et ClamAV

Bugs connus

- 2003 : Python ne génère que des nombres pairs
- 2007 : Générateur ID de BIND9 *cassé*
- 2008 : Bug OpenSSL dans Debian
- 2008 : Biais dans la graîne en PHP

Bibliothèques existantes

- Initialisation manuelle
- Pas de fonction randint(a, b)
- Distribution non uniforme
- Faible période
- Peu ou pas de test

Bibliothèque Hasard

- Initialisation automatique
- API conçue pour éviter les erreurs courantes
- Fonctions réentrantes
- Nombreux tests

Profiles

- @fast : simulation, jeux vidéos
- @secure_blocking : sécurisé, bloquant
- @secure_nonblocking : compromis vitesse / sécurité
- @hardware : générateur matériel

API Hasard

- Entiers : int, unsigned long, (u)int8/16/32
- Autres : bool, bytes, double, uuid
- Mélanger un tableau
- Informations sur le générateur
- Lire/écrire l'état interne, cloner, reseed

Hébergement

- Avant : page Trac hébergée sur un serveur à la maison (ADSL)
- Aujourd'hui : hébergement gratuit chez Bitbucket
- Mercurial : développer dans le train, commits instantanés, intuitif

Générateurs dans Hasard

- Arcfour, ISAAC, KISS, Mersenne Twister
- Réutilise OpenSSL, gcrypt, GSL, GMP, glib, Havege
- /dev/random, /dev/urandom, CryptGen
- Générateurs faibles bannis

Mersenne Twister

- Bons résultats aux tests statistiques
- Par défaut dans Python, Ruby, glib, GSL, GMP, . . .
- Disponible dans PHP, Perl, . . .
- Non cryptographique

Arcfour (RC4)

- Algorithme très simple
- Cryptographique, mais *cassé* (WEP)
- Répandu : wifi (WEP, WPA), SSL, PDF, . . .

Générateurs faibles

- Bibliothèque à part : hasardweak
- Générateurs congruentiels linéaires
- $x_{n+1} \equiv (a \cdot x(n) + c) \mod m$
- Fonctions de la libc (rand(), nrand48())
- Sert aux tests et à la compatibilité

Où trouver de l'entropie

- Utilisateur : clavier, souris
- Matériel : disque dur, interruptions
- Mauvaise idée : getpid(), time()
- Matériel dédié (ex: clé USB)

Langage C

- Facilité d'intégration aux autres langages
- Compilateurs disponibles partout
- Performances
- Joie de la compilation et des erreurs de segmentation !

Options de gcc

- Communes : -Wall -Wextra
- Mode debug : -Werror -OØ -ggdb
- Mode release : -O3
- Soucis avec int, long, size_t : utiliser -Wconversion sur une machine 64 bits

Chargement dynamique des bibliothèques

- Limiter les dépendences
- Linux : dlopen(), dlsym()
- Windows : LoadLibraryW(), GetProcAddress()

cmake (1/2)

- Génère des Makefile UNIX ou un projet Visual Studio
- Un seul langage simple, lisible et concis
- Testé sous Windows (MinGW), Linux, Mac OS X, FreeBSD, OpenBSD

cmake (2/2)

- Détecte endian, bibliothèques, fonctions
- Choix des drapeaux de compilation
- Points faibles : peu d'utilisateurs, peu de documentation, rarement préinstallé

Binding ctypes

- ctypes : binding en pur Python, sans compilation
- Syntaxe triviale et intuitive
- Intégré à Python 2.5

Hello World ctypes

- from ctypes import cdll
- libc = cdll.LoadLibrary('libc.so.6')
- libc.printf("Hello World\n")

Types

- func.argtypes = (c_int, c_char_p, c_double)
- func.restype = c_char
- func.restype = None # procédure

Binding Hasard

- Binding de chaque fonction
- API objet simple
- Conversion dans les types Python

Tests en Python

- Tests écrits plus rapidement en Python qu'en C
- 36 tests différents : valeurs connues, collision, entropie, ...
- Permet le refactoring
- Effet de bord : permet de valider le binding

Options des tests

- Tester un seul générateur
- Filtrage des tests par mot clé
- Répéter le même test
- Nombre de boucles

Fichier texte

- Texte : facile à manipuler
- Entêtes : nombre, min/max, type, ...
- Pipe : générateur | outil
- Compression gzip / bz2

Outils

- Générer un fichier
- Dessiner une image ou graphique
- Outils externes : ENT, Dieharder
- Calcul de la période
- Calcul de l’état interne