T^3 - Twórca Tablic Tęczowych

Równoległe wyznaczanie tęczowych tablic ("rainbow tables") w zgadnieniach kryptografii dla haseł zaszyfrowanych algorytmem DES: Scala

Bartosz Pieńkowski, Barnaba Turek

16 maja 2011

1 Opis

T^3 to zestaw programów wyznaczających tablice tęczowe i pozwalających sprawdzić poprawność ich wyznaczenia (przez wyznaczenie funkcji skrótu dla danego ciągu znaków (dalej klucza) i próbę odwrócenia tego procesu).

Tablice tęczowe to sposób przechowywania wcześniej obliczonych danych pozwalających odwracać (analitycznie nieodwracalną) funkcję skrótu[1]. *Tablice tęczowe* pozwalają zmniejszyć wymagania dyskowe (w stosunku do prostego zapisywania wszystkich par klucz-f(klucz)) kosztem wymagań obliczeniowych. Osiągane to jest za pomocą tworzenia tzw. łańcuchów skrótów[2] i zapisywaniu tylko pierwszego i ostatniego elementu łańcucha.

1.1 Funkcja skrótu

T^3 wyznacza *tablice tęczowe* dla funkcji skrótu zgodnej z funkcją crypt (należącą do standardu **POSIX**) działającej w oparciu o standard **DES**.

2 Użycie

2.1 Tworzenie tablic tęczowych

Program generujący tablice tęczowe nazywa się *t3*.

2.1.1 Wywołanie programu

Użytkownik programu t3 podaje trzy argumenty linii poleceń. Pierwszy argument określa długość klucza, dla którego mają być wygenerowane *tablice tęczowe* (od 1 do 8). Drugi argument określa długość obliczanych łańcuchów. Trzeci argument jest opcjonalny i określa alfabet użyty do generowania kluczy. Domyślnie alfabet to małe litery.

2.1.2 Wyjście programu

Tablice tęczowe zostaną zapisane w aktualnym katalogu. Program tworzy plik, składający się z wierszy. Każdy wiersz składa się z początkowego i końcowego elementu łańcucha skrótów, oddzielnoych spacją.

[1] inaczej kryptograficzną funkcję mieszającą, ang. *hash function*
[2] ang. *hash chaining*

2.1.3 Przykładowe wywołania

Wywołanie:

```
$ t3 2 10 abc
```

Wygeneruje tablice tęczowe dla haseł o długości dwóch znaków z alfabetu abc. Łańcuchy skrótów będą miały długość 10.

Aby uprościć podawanie alfabetu można wykorzystać inne, dostępne w systemie narzędzia. Np. w wielu powłokach systemu GNU/Linux można wykorzystać program **perl** w następujący sposób:

```
$ t3 2 10 `perl -e 'print join ``, a..z,A..Z,0..9'``
```

Wywołanie takie jest równoważne wywołaniu:

```
$ t3 2 10 abcdefghijklmnopqrstuvwxyzABCDEFGHIJKLMNOPQRSTUVWXYZ0123456789
```

i generuje tablice tęczowe dla dwuznakowych haseł składających się z kombinacji wielkich liter, małych liter i cyfr.

2.2 Generowanie skrótu

Do generowania i skrótów służy program *t3-hash*.

2.2.1 Wywołanie programu

Użytkownik programu *t3-hash* jako argument podaje klucz (1 do 8 znaków), dla którego ma być wygenerowany skrót. Następnie program wypisuje skrót na standardowe wyjście.

2.3 Odwracanie funkcji skrótu

Do odwracania funkcji skrótu służy program *t3-reverse*.

2.3.1 Wywołanie programu

Użytkownik programu *t3-reverse* jako argument podaje wartość funkcji skrótu, dla której znaleziona ma być wartość klucza.

Jeżeli program został wywołany z katalogu, w którym nie ma wygenerowanych tablic tęczowych, program zakończy działanie wypisując informację o braku tablic.

Jeżeli program nie znajdzie klucza pasującego do zadanej wartości funkcji skrótu, program zakończy działanie wypisując informację o niepowodzeniu.

Jeżeli działanie programu zakończy się sukcesem, program wypisze znaleziony klucz. Poprawność znalezionego klucza będzie można sprawdzić korzystając z programu *t3-hash*.

3 Rozwiązania

Program zostanie wykonany w języku **Scala** na platformę **JVM**.

Programy *t3-hash* i *t3-reverse* nie będą działać współbieżnie - programy te służą głównie do sprawdzania poprawności wygenerowanych tablic i są znacząco mniej wymagające obliczeniowo.

Zrównoleglenie wyznaczania tablic tęczowych zostanie osiągnięte za pomocą mechanizmu Aktorów oferowanego przez język **Scala**. Mechanizm ten jest zrealizowany na wirtualnej maszynie Javy za pomocą wątków.

Zamierzamy wykorzystać komunikację globalną - jeden wątek będzie zarządzał wszystkimi innymi wątkami.

Naszym zdaniem najlepszą dekompozycją problemu przy tak postawionym zadaniu będzie dekompozycja domenowa, tj. równomierny podział początkowych[3] kluczy na wątki.

[3]zaczynających łańcuch skrótów