Implementacja funkcji Blake3 i atak kryptograficzny

inż. Marek Borzyszkowski 184266

18 lipca 2024

Spis treści

1 Wstęp

Cele projektu było zaimplementowanie funckji na bazie `Blake3`, wykorzystanie tej funkcji do stworzenia programu hashującego zadane ciągi znaków ze standardowego wejścia oraz dokonanie ataku kryptograficznego.

2 Opis projektu

Projekt wraz z instrukcją i wynikami znajdują się w repozytorium:
`https://github.com/MarekBorzyszkowski/Blake3ModifiedHash`

Znajdują się tam 2 implementacje, pod CPU i CUDA. Wersja CUDA została zoptymalizowana do ataku kryptograficznego na ciąg składający się do 8 znaków. Język programowania wykorzystany w projekcie to `Python` w wersji `3.12`, a wykorzystane pakiety do optymalizacji kodu to:

- numpy w wersji 2.0.0,
- numba w wersji 0.60.0.

Do łatwego zainstalowania wymaganych pakietów stworzono plik `requirements.txt`, który można wykorzystać w komendzie:
`pip install -r requirements.txt`

W celu ułatwienia tworzenia działającego i łatwego w używaniu kodu stworzono szereg testów jednostkowych i end to end.

Do włączenia programu pozwalającego na korzystanie z funckcji hashującej w formie interaktywnej istnieje skrypt `startHashing.sh`. W przypadku utworzenia wirtualnego środowiska w innym miejscu niż wyspecyfikowanym w `startHashing.sh`, należy dokonać odpowiednich zmian w pliku. Po włączeniu programu, hashuje on ciągi znaków ze standardowego wejścia, do momentu użycia znaku `EoF`, co wyłącza program. Przykładowe użycie programu Lst. 1.

Listing 1: Przykładowy przebieg działanaia programu

```
To end the program insert EOF (Ctrl + D on unix, Ctrl + Z on windows)

Blake3 hash of '' = 898F E038 CC44 AC95 0F78 F84D 8796 98C9
abc
Blake3 hash of 'abc' = 9E0F 5F51 00C1 44A0 9F84 CB56 D23C 9770
AbCxYz
Blake3 hash of 'AbCxYz' = E1C1 3F52 3C78 7589 22FD 11AA 3132 D01C
Thank you for using Blake3 hash!
```

Powyższy program korzysta z implementacji na CPU. Z kodu napisanego pod CUDA można korzystać na komputerze bez CUDA dodając flagę `NUMBA_ENABLE_CUDASIM=1`.

Podczas ataku kryptograficznego korzystano z jednego komputera o specyfikcaji:

- CPU: intel i7-13700K,
- GPU: RTX 4070Ti,
- RAM: 32GB DDR5 5600MHz,

- Dysk: M.2 PCI-e 4.0.

Podczas ataku za pomocą CPU korzystano z 24 wątków, a podczas ataku za pomocą CUDA z 12288 bloków po 512 wątków.

3 Wyniki

Wyniki ataków kryptograficznych dla kolejnych długości ciągów znaków i ich hashów znajdują się w tabeli Tab. 1. Dla każdej długości i znanej dla niej wynikowego hasha zapisany jest wejściowy ciąg znaków oraz czas jego znalezienia korzystając z ataku kryptograficznego pisanego pod CPU i CUDA. Dla ciągów 7 i 8 w wartości CPU znajduje się -, gdyż według szacunków ciąg 7 znakowy liczyłby się 48 dni, a 8 znakowy 12 lat, oba czasy są zbyt długie jak na potrzeby ataku.

Tablica 1: Wyniki ataków kryptograficznych

Dł. ciągu	Wynikowy hash	Wejściowy ciąg znaków	CPU	CUDA
2	29 0D 8E 30 A7 F7 58 DE 02 3C 9C 74 62 33 63 1D	2#	0,00213 s	0,45342 s
3	6C 34 6E 8D 30 67 EF 3B 7B C3 E5 C2 99 CC 75 35	aLV	0,05210 s	0,44876 s
4	E1 4D A6 D5 EB 17 15 BE CD 5D 46 80 D9 9D 6E DC	_)lU	4,86261 s	0,48625 s
5	26 8D DE E3 CD 85 4D 73 80 E5 4F 61 57 12 86 CD	4H/&#	7,49707 min	1,10177 s
6	CF AC 55 48 46 A0 7C F5 54 34 4C 38 7B 8E 48 DC	q?u5G(	12,5012 h	1,02363 min
7	22 AA 75 76 75 8A 39 78 77 BC 3A A0 40 F5 BD 12	dBD4@N&	-	1,60530 h
8	62 07 25 80 37 4C 71 E6 0D 2C 83 5E 33 98 5B E5	njpn7y83	-	6,30219 dni