Quadratwurzelangriff mit CUDA

Denny Hecht, Silvio Feig

21.01.2015

Übersicht

Diffie-Hellman-Schlüsselaustausch

- öffentliche Werte:
  - n Primzahl, so dass $\frac{n-1}{2}$ auch eine Primzahl ist
  - g Primitivwurzel von n
  - $a = g^x \ mod \ n$
  - $b = g^y \ mod \ n$
- geheime Werte:
  - x
  - y
  - privater Schlüssel $ps = g^{xy} \, mod \ n$

Quadratwurzelangriff

- gesucht ist der diskrete Logarithmus
  - $x = dlog_g(a)$
  - $y = dlog_g(b)$
- der private Schlüssel *ps* kann nun mit $g^{xy} \, mod \, n$ berechnet werden

BABYSTEP-GIANTSTEP-ALGORITHMUS(n, g, a)

```
1   m = √(n−1)
2
3   for j ∈ {0,...,m−1}
4       babyStepTable[j] = g^j
5   end
6
7   for i ∈ {0,...,m−1}
8     giantStepTable[i] = a(g^−m)^i
9   end
10
11  for i ∈ {0,...,m−1}
12      for j ∈ {0,...,m−1}
13          if giantStepTable[i] == babyStepTabel[j]
14              return im+j
15          end
16  end
```

Hardware

- gesucht $n \leq 2^{64}$
- $m = \sqrt{2^{64} - 1} \approx 4.294.967.296$
- 2^{64} ist die größte annehmbar Zahl
- 8 Byte pro Zahl
- 4.294.967.296 * 8 Byte $\approx$ 32 GByte
- Hauptspeicher: 2 GByte
- Grafikspeicher: 1.5 GByte

CUDA

siehe Quelltext