EI-9

Montag, 23. November 2020 13:26

Lukas Ingold 20-123-998

Institut für Informatik, Universität Bern
Einführung in die Informatik, HS 2020

Prof. Christian Cachin

Kryptographie – Übungen

1. Eine Hashfunktion?

Gegeben sei eine grosse Primzahl p und eine Zahl $g \in \mathbb{Z}_p^*$, so dass das Problem des diskreten Logarithmus modulo p zur Basis g schwierig ist. Dies bedeutet, kein effizienter Algorithmus findet für ein gegebenes, zufälliges $y \in \mathbb{Z}_p^*$ ein x für welches $g^x = y \textbf{ mod } p$ gilt.

Die Funktion $H : \mathbb{Z} \to \mathbb{Z}_p^*$ sei definiert durch $H(a) = g^a \textbf{ mod } p$. Beachte, dass H Argumente erlaubt, welche viel grösser sein können als ihre Ausgabe (d.h., $a \gg p$). Ist H eine kryptographisch sichere Hashfunktion? Begründen Sie die Antwort.

2. Modulare Exponentiation

In der Vorlesung die Exponentiation modulo eine Primzahl gezeigt für die Rechnung $7^8 \textbf{ mod } 11$. Da $8 = 2^3$ funktionierte dies durch wiederholtes Quadrieren modulo 11.

Wie kann man effizient eine modulare Exponentiation $a^b \textbf{ mod } m$ berechnen, wenn der Exponent b und der Modulus m beliebige Zahlen sind? Also auch insbesondere dann, wenn p prim ist und wenn m *nicht* prim ist. Repräsentieren Sie dazu den Exponenten im Binärsystem als $b = (b_k b_{k-1} \ldots b_1 b_0)_2$.

a) Beschreiben Sie einen Algorithmus dafür in Pseudocode oder in einer Programmiersprache Ihrer Wahl.

b) Berechnen Sie nach dieser Methode $x = 7^{151} \textbf{ mod } 15$. Zeigen Sie alle Zwischenergebnisse, indem Sie entweder von Hand rechnen oder die Zwischenschritte durch Ihr Programm ausgeben lassen.

1. Eine Hashfunktion?

Gegeben sei eine grosse Primzahl p und eine Zahl $g \in \mathbb{Z}_p^*$, so dass das Problem des diskreten Logarithmus modulo p zur Basis g schwierig ist. Dies bedeutet, kein effizienter Algorithmus findet für ein gegebenes, zufälliges $y \in \mathbb{Z}_p^*$ ein x für welches $g^x = y \textbf{ mod } p$ gilt.

Die Funktion $H : \mathbb{Z} \to \mathbb{Z}_p^*$ sei definiert durch $H(a) = g^a \textbf{ mod } p$. Beachte, dass H Argumente erlaubt, welche viel grösser sein können als ihre Ausgabe (d.h., $a \gg p$). Ist H eine kryptographisch sichere Hashfunktion? Begründen Sie die Antwort.

Primzahl p

Zahl g

diskreter log mod p basis g

$y \in \mathbb{Z}$

$g^x = y$ mod p gilt

$H: \mathbb{Z} \longrightarrow \mathbb{Z}_p^*$ $\quad H(a) = g^a$ mod p

$a \gg p$

ist H Kryptographisch sichere Hashfunktion?

<u>Sicher:</u> bedeutet für y (kennen wir] fast unmöglich ein x zu finden welches $H(x) = y$ erfüllt

Diese FUNKTION IST SICHER DA DIE UMKEHRFUNKTION $\mathrm{DLOG}_g(n)$ mod p Fast unmöglich ist. Das bedeutet $H(x) = y$ ist nahe zu nicht lösbar (ein x zu finden ist sehr schwierig)

2. Modulare Exponentiation

In der Vorlesung die Exponentiation modulo eine Primzahl gezeigt für die Rechnung $7^8 \textbf{ mod } 11$. Da $8 = 2^3$ funktionierte dies durch wiederholtes Quadrieren modulo 11.

Wie kann man effizient eine modulare Exponentiation $a^b \textbf{ mod } m$ berechnen, wenn der Exponent b und der Modulus m beliebige Zahlen sind? Also auch insbesondere dann, wenn p prim ist und wenn m *nicht* prim ist. Repräsentieren Sie dazu den Exponenten im Binärsystem als $b = (b_k b_{k-1} \ldots b_1 b_0)_2$.

a) Beschreiben Sie einen Algorithmus dafür in Pseudocode oder in einer Programmiersprache Ihrer Wahl.

b) Berechnen Sie nach dieser Methode $x = 7^{151} \textbf{ mod } 15$. Zeigen Sie alle Zwischenergebnisse, indem Sie entweder von Hand rechnen oder die Zwischenschritte durch Ihr Programm ausgeben lassen.

a)

```
int result = 1;
a = a%m;
if ( a == 0)
{
  return 0;
}
while(b > 0)
{
  b = b/2;
  a = (a * a)%m;

  if(b % 2 == 1)
  {
    result = (result * a) % m;
    b = b-1;
  }
  return result;
}
```

b) $x = 7^{151}$ Mod 15

$a = 7$ mod 15
result = $(1 \cdot 7)$ mod 15
$b = 151 - 1$
$b = 150 : 2$
result = $(7 \cdot 4)$ mod 15
$b = 75 - 1$
$b = 74 : 2$
$a = (7 \cdot 7)$ mod 15
result = $(4 \cdot 7)$ mod 15
$b = 37 - 1$
$b = 36 : 2$
result = $(4 \cdot 7)$ mod 15
$b = 18 : 2$
$a = (7 \cdot 7)$ mod 15
result = $(4 \cdot 7)$ mod 15
$b = 9 - 1$
$b = 8 : 2$
result = $(4 \cdot 7)$ mod 15
$b = 2 : 2$
$b = 1 - 1$
$b = 0 : 2$
$a = (7 \cdot 7)$ mod 15
return result