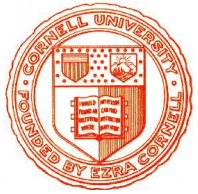
Kryptographie und Komplexität



Einheit 5

Kryptosysteme auf der Basis diskreter Logarithmen

1. Diffie Hellman Schlüsselaustausch
2. El Gamal Systeme
3. Angriffe auf Diskrete Logarithmen
4. Elliptische Kurven

Schwächen des RSA

- **Schlüssel müssen sehr groß werden**
  - Faktorisierungsalgorithmen können Schlüssel bis zu 1024 Bit angreifen
  - Blockgröße muß auf 2048 Bit oder größer anwachsen
  - Wachsende Blockgröße macht Verschlüsselung infeffizient
    - Potenzierung modulo n benötigt $\mathcal{O}(\|n\|^3)$ Schritte
    - Zeit für Verschlüsselung langer Nachrichten wächst quadratisch
  - Verschlüsselung braucht neue algebraische Probleme als Fundament
    Schwer zu brechende kleine Schlüssel oder effizientere Verschlüsselung
- **Semantische Sicherheit nicht sichergestellt**
  - Zahlentheoretisches Verfahren enthält keine Randomisierung
  - Gleiche Nachrichten werden imm auf gleiche Art verschlüsselt
  - Verschlüsselungsprotokoll sollte Zufall mit einbauen

Diskrete Logarithmen

- **Umstellung der RSA Ver-/Entschlüsselung**
  - RSA: Gegeben $y = x^e \bmod n$ bestimme $x = \sqrt[e]{y} \bmod n$
  - DL: Gegeben $y = e^x \bmod n$ bestimme $x = \log_e y \bmod n$
  - Formulierbar für beliebige zyklische Gruppen anstelle von $\mathbb{Z}_n$
    e muß keine Zahl sein sondern nur Gruppenelement der Ordnung n
- **Algebraische Formulierung des Problems**
  - Sei $(G, \cdot)$ multiplikative Gruppe, g Element der Ordnung n
    Für $y \in \langle g \rangle$ ist der **diskrete Logarithmus von y zur Basis g**
    (bezeichnet als $\boldsymbol{x = \log_g y}$) die eindeutige Zahl $x<n$ mit $y = g^x$
- **Welche Gruppen sind geeignet?**
  - Prime Restklassen modulo einer Primzahl $(\mathbb{Z}_p^*, \cdot)$
  - Punktgruppe einer elliptischen Kurve über endlichen Körpern
  - Hyperelliptische Kurven, Gruppen imaginär-quadratischer Ordnungen ...

  Verzicht auf numerische Struktur macht Logarithmen z.T. erheblich schwerer zu berechnen als Wurzeln über $\mathbb{Z}_n$

Diskrete Logarithmen am Beispiel

- **Logarithmen zur Basis 2 modulo 13**
  - Berechne Potenzen von 2 zur Basis 13 mit Elementen aus $\mathbb{Z}_{13}^*$

x	1	2	3	4	5	6	7	8	9	10	11	12
2^x	2	4	8	3	6	12	11	9	5	10	7	1

  - Umstellung nach Logarithmen (Ordnung von 2 ist $n = 12$)

y	1	2	3	4	5	6	7	8	9	10	11	12
$\log_2 y$	0	1	4	2	9	5	11	3	8	10	7	6

- **Logarithmen zur Basis 5 modulo 19**
  - Berechne Potenzen von 5 zur Basis 19 mit Elementen aus $\mathbb{Z}_{19}^*$

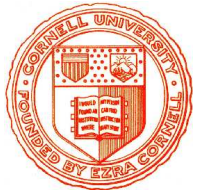
x	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18
5^x	5	6	11	17	9	7	16	4	1	5	6	11	17	9	7	16	4	1

  - Ordnung von 5 ist nur $n = 9$: $\langle 5 \rangle = \{1; 4; 5; 6; 7; 11; 16; 17\}$
  - Umstellung nach Logarithmen für die Elemente von $\langle 5 \rangle$

y	1	4	5	6	7	9	11	16	17
$\log_5 y$	9	17	1	11	6	5	3	7	4

  - Logarithums ist eine Zahl $x<n$, kein Gruppenelement

Kryptographie und Komplexität



Einheit 5.1

Diffie Hellman Schlüsselaustausch

1. Protokoll für sicheren Schlüsselaustausch
2. Sicherheit des Verfahrens
3. Verallgemeinerung auf beliebige Gruppen

Diffie Hellman Schlüsselaustausch

Sicherer Austausch von Schlüsseln

- **Protokoll mit Diskreten Logarithmen über $\mathbb{Z}_p$**
  - Wähle Primzahl p und Erzeuger g von $\mathbb{Z}_p$ mit $2 \leq g \leq p-2$
    p und g werden nicht geheim gehalten
  - Alice wählt zufällige Zahl $a \in \{0, .., p-2\}$ und berechnet $A = g^a \bmod p$
    Alice hält a geheim und schickt A an Bob
  - Bob wählt zufällige Zahl $b \in \{0, .., p-2\}$ und berechnet $B = g^b \bmod p$
    Bob hält b geheim und schickt B an Alice
  - Alice berechnet $B^a \bmod p = g^{a \cdot b} \bmod p$
    Bob berechnet $A^b \bmod p = g^{a \cdot b} \bmod p$
  - Gemeinsamer Schlüssel $K = g^{a \cdot b} \bmod p$ ist nur Alice und Bob bekannt
- **Beispiel für n=17 und g=3**
  - Alice wählt a=7 und berechnet $A = g^a \bmod 17 = 2187 \bmod 17 = 11$
  - Bob wählt b=4 und berechnet $B = g^b \bmod 17 = 81 \bmod 17 = 13$
  - Der gemeinsame Schlüssel ist $K = A^b \bmod 17 = 14641 \bmod 17 = 4$

Sicherheit des Diffie Hellman Schlüssels

- **Angreifer kennt p, g, A und B**
  - p, g, A und B wurden über unsichere Kanäle ausgetauscht
  - Methode zur Bestimmung des gemeinsamen Schlüssels K ist bekannt
  - Um $K = A^b \bmod p = B^a \bmod p$ zu berechnen, müsste Angreifer entweder a oder b bestimmen können
- **Angreifer muß diskreten Logarithmus lösen**
  - Um K zu bestimmen muß Angreifer entweder $a = \log_g A$ oder $b = \log_g B$ ausrechnen können
  - Andere Methode, gemeinsamen Schlüssel zu brechen ist nicht bekannt
  - Äquivalenz des Diffie-Hellman Problems (bestimme $g^{a \cdot b}$ aus g^a und g^b) zum Problem des diskreten Logarithmus (berechne $\log_g A$) nicht bewiesen
- **Berechnung diskreter Logarithmen ist schwer**
  - Beste bekannte Verfahren für $\mathbb{Z}_p$ liegen in $L_n[1/3, 1.92]$
    Effizienteste Verfahren sind auf andere Gruppen nicht anwendbar

Alternativen zu $\mathbb{Z}_p^*$

- **Verfahren möglich auf beliebigen Gruppen**
  - Gruppen müssen zyklisch sein und erzeugende Elemente haben
  - Multiplikation/ und Potenzierung muß effizient implementierbar sein
  - Diffie-Hellman Problem muß schwer zu lösen sein
    $(\mathbb{Z}_p, +)$ ist ungeeignet, da $\log_g A = A \cdot g^{-1}$ leicht zu berechnen
- **Protokoll nahezu identisch**
  - Wähle Erzeuger g der Gruppe G mit Ordnung n $2 \leq g \leq p-2$
  - Alice wählt zufällige Zahl $a \in \{1, .., n-1\}$ und berechnet $A = g^a \in G$
  - Bob wählt zufällige Zahl $b \in \{1, .., p-1\}$ und berechnet $B = g^b \in G$
  - Alice berechnet $B^a = g^{a \cdot b}$ – Bob berechnet $A^b g^{a \cdot b}$
  - Gemeinsamer Schlüssel $K = g^{a \cdot b}$ ist nur Alice und Bob bekannt
- **Vorteil alternativer Gruppen**
  - Gruppenoperationen sind komplexer und schwerer zu invertieren
  - Lösung des Diffie-Hellman Problems kann sich nicht (nur) auf zahlentheoretische Zusammenhänge stützen
  - Größere Sicherheit bei geringerer Schlüssellänge möglich