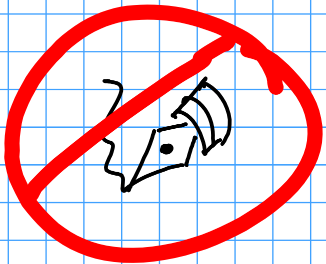
Einführung in die Krypto

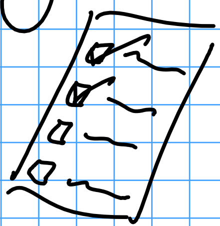
2.2.2011



Letztes Mal: Digitale Signaturen

- El Gamal Signatur
- Existentielle Fälschung
- Digital Signature Algorithm (DSA)

Fragebogen



Dieses Mal: Diskrete Logarithmen (DL)

- DSA Korrektheit
- Diskrete Logarithmen
- Babystep-Giantstep
- Pollard Rho

Ferienübungspunkte

$\varnothing = 7.5$

DSA Korrektheit

Schlüssel:

öffentlich: (A, g, p, q)

geheim: (a)

- $p \approx 2^{1024}$ $q \approx 2^{160}$ prim $p = q \cdot m + 1$
- g hat Ordnung q in $(\mathbb{Z}/p\mathbb{Z})^*$
- a zufällig in $\{1, .., q-1\}$
- $A = g^a \bmod p$

Signieren(m):

Signatur: (r, s)

- k zufällig in $\{1, .., q-1\}$
- $r = (g^k \bmod p) \bmod q$
- $s = k^{-1} (h(m) + ar) \bmod q$

Verifizieren(r, s, m):

- $1 \leq r, s \leq q-1$?
- $\left(g^{s^{-1}h(m) \bmod q} A^{s^{-1}r \bmod q} \bmod p\right) \bmod q$

$== r$?

Korrektheit: $$g^{s^{-1}h(m)} \cdot A^{s^{-1}r} \equiv g^{s^{-1}h(m)+s^{-1}ar}$$
$$\equiv g^{s^{-1}(h(m)+ar)}$$
$$\equiv g^{k s^{-1} k^{-1}(h(m)+ar)}$$
$$\equiv g^{k} \pmod p$$

Wird jede ehrliche Signatur korrekt verifiziert ?

Achtung: Implizite Annahme, dass $s \neq 0$ ist.

Jeder benutzt DSA ... Wie sicher ist das wirklich?

Bester bekannter Angriff: ¿

Diskrete Logarithmen

Sei G endl. zyklische Gruppe erzeugt von γ.

$$G = \{ \gamma^x \mid x \in \mathbb{N}_{\geq 0} \}, \quad |G| = n$$

DL: Gegeben $\alpha \in G$, finde kleinste $x \geq 0$ mit $\alpha = \gamma^x$.

Beispiel: $G = (\mathbb{Z}/p\mathbb{Z})^*$ für $p = 7$, $\gamma = 3$

$\alpha = 2 \qquad x = 2$

$\alpha = 6 \qquad x = 3$

$\alpha = 4 \qquad x = 4$

$$G = \{ 1, 3, 2, 6, 4, 5 \}$$

$$3^0 \; 3^1 \; 3^2 \ldots \quad 3^5$$

$\xleftarrow{\text{easy}}$

$\xrightarrow{\text{hard}}$

Wie finde ich DL ?

(1) abzählen $x = 0, 1, 2, \ldots$ $\qquad \alpha \overset{?}{=} \gamma^{x}$

Analyse: Im schlimmsten Fall ist $x = |G| - 1$, man benötigt also $O(|G|)$ viel Gruppenoperationen.

\+ $O(1)$ Speicher

− $O(|G|)$ Laufzeit

(2) Shanks Babystep-Giantstep 1971

− $O(\sqrt{|G|})$ Speicher

\+ $O(\sqrt{|G|})$ Laufzeit

Shanks Babystep - Giantstep

$n = |G| \quad x = q \cdot m + r$ für $0 \leq q, r < m \quad m = \lceil \sqrt{n} \rceil$

Aufgabe: Finde q, r.

Angsatz:

- Berechne alle Babysteps $B = \{(\alpha \gamma^{-r}, r) : 0 \leq r < m\}$
- Berechne Giantsteps $((\gamma^m)^q, q)$ für $0 \leq q < m$

  bis es Kollision gibt

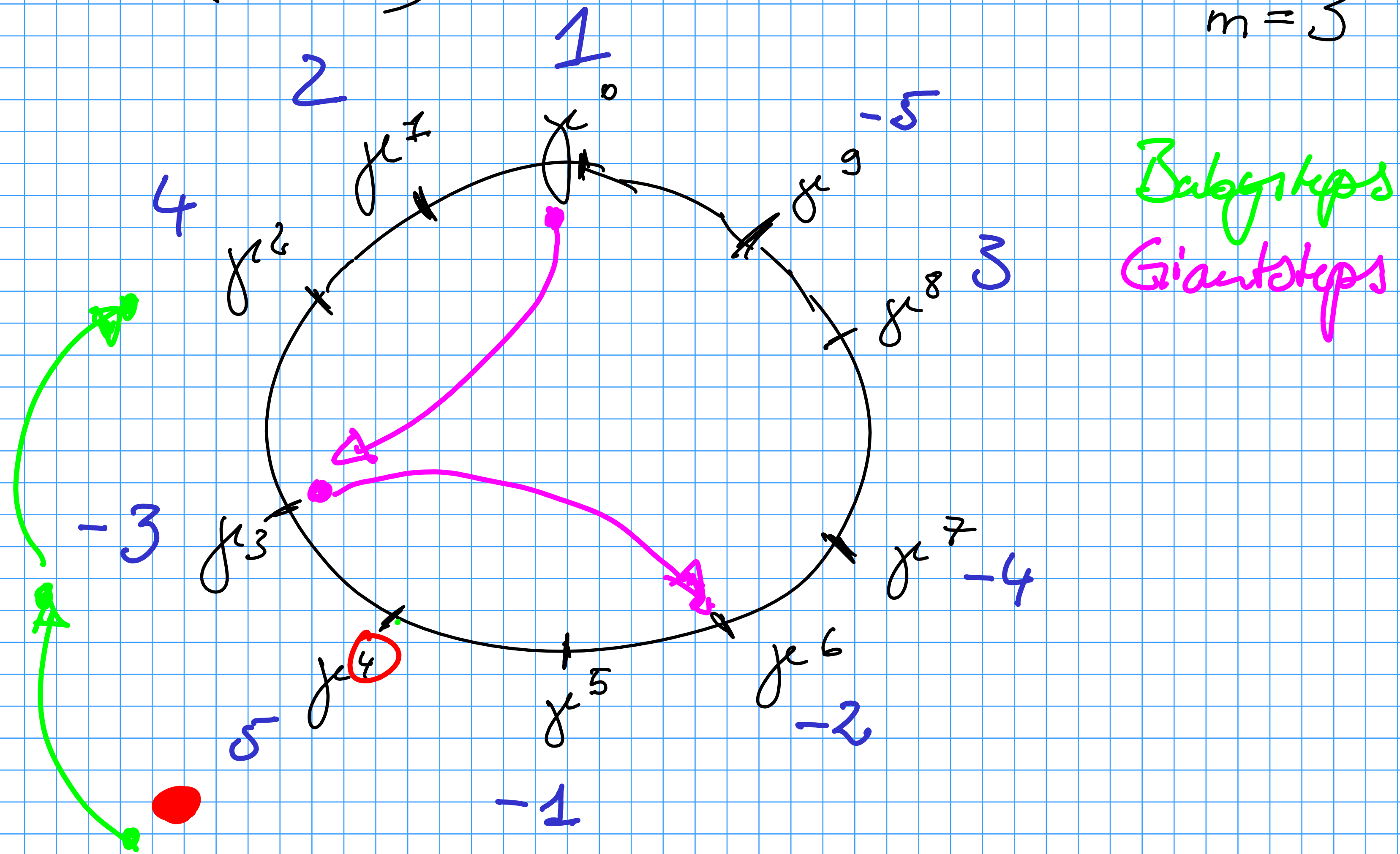
$$\alpha \gamma^{-r} = (\gamma^m)^q$$

$$\alpha = \gamma^{qm+r} \Rightarrow qm + r = x$$

Warum Steps (Schritte)?

Beispiel: $G = (\mathbb{Z}/11\mathbb{Z})^*$ $\quad \alpha = 5 \quad \gamma = 2 \quad n = 10$

$m = 3$



Ist das alles?

(3) Pollard Rho 1978

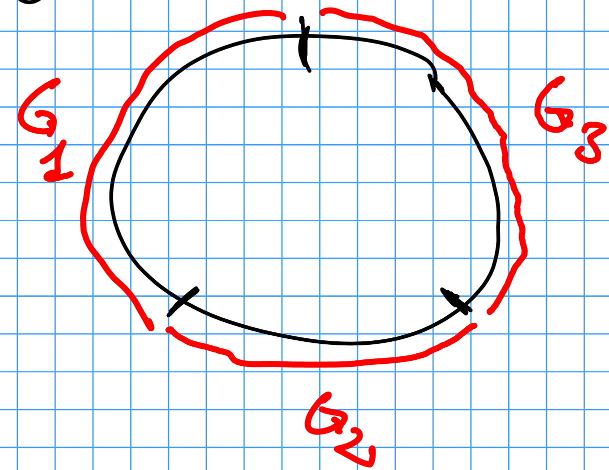
+ $O(\sqrt{|G|})$ Laufzeit

+ $O(1)$ Speicher

Pollard Rho

Ansatz: • Partitioniere G in 3 disjunkte gleichgroße Teile

$$G = G_1 \,\dot\cup\, G_2 \,\dot\cup\, G_3$$



• Sei

$$f : G \to G : \beta \mapsto \begin{cases} \alpha\beta & \beta \in G_1 \\ \beta\beta & \beta \in G_2 \\ \gamma\beta & \beta \in G_3 \end{cases}$$

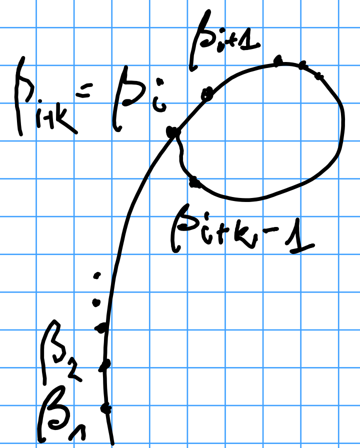
• Berechne Sequenz

$$\beta_{i+1} = f(\beta_i)$$ mit β_1 zufällig "random walk".

Alle β_i haben die Form $\beta_i = \gamma^{x_i}\alpha^{y_i}$ mit ?

$$x_{i+1} = \begin{cases} x_i & \beta_i \in G_1 \\ 2x_i & \beta_i \in G_2 \\ x_i + 1 & \beta_i \in G_3 \end{cases} \qquad y_{i+1} = \begin{cases} y_i + 1 & \beta_i \in G_1 \\ 2y_i & \beta_i \in G_2 \\ y_i & \beta_i \in G_3 \end{cases}$$

$$x_{i+1} = \begin{cases} x_i & \beta_i \in G_1 \\ 2x_i & \beta_i \in G_2 \\ x_i + 1 & \beta_i \in G_3 \end{cases} \qquad y_{i+1} = \begin{cases} y_i + 1 & \beta_i \in G_1 \\ 2y_i & \beta_i \in G_2 \\ y_i & \beta_i \in G_3 \end{cases}$$



- Bei Kollision $\beta_i = \beta_{i+k}$

$$\gamma^{x_i} \alpha^{y_i} = \gamma^{x_{i+k}} \alpha^{y_{i+k}}$$

$$\gamma^{(x_i - x_{i+k})} = \alpha^{(y_{i+k} - y_i)}$$

$$\gamma^{(x_i - x_{i+k})} = \gamma^{x(y_{i+k} - y_i)}$$

$$\Rightarrow \quad (x_i - x_{i+k}) \equiv x\,(y_{i+k} - y_i) \pmod{n}$$

Alles ausser x bekannt → auflösen!

Laut Geburtstagsparadox Koll nach $O(\sqrt{|G|})$ Schritten

Problem: ?

Problem: Wieder $O(\sqrt{|G|})$ Speicher / Laufzeit :-

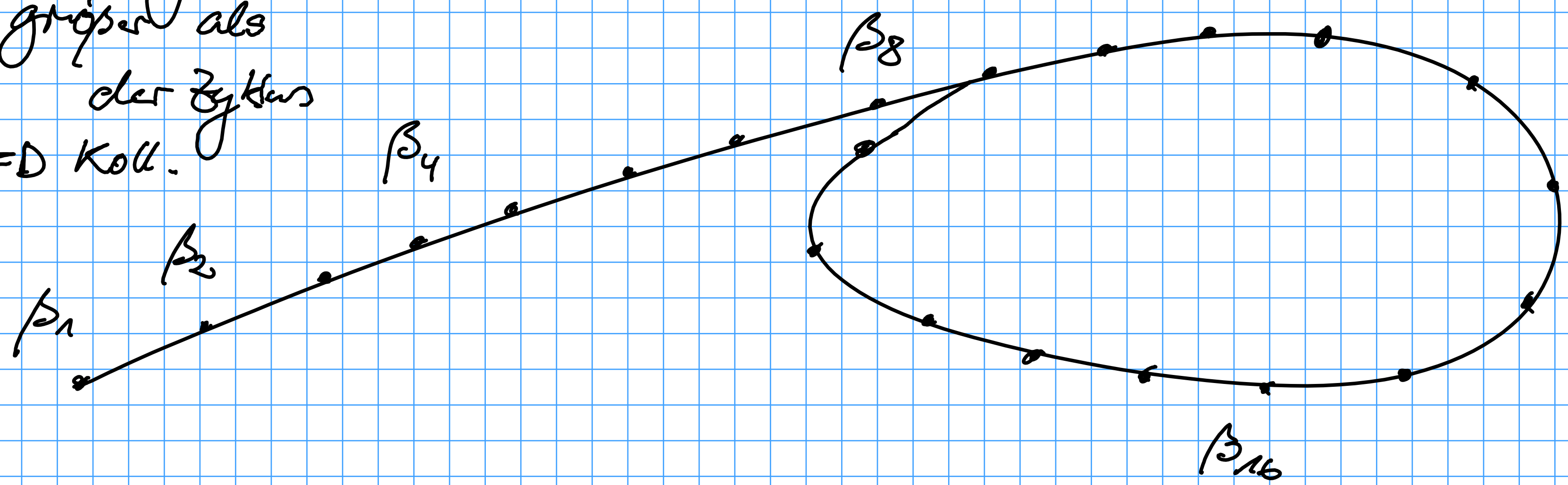
Lösung: „Spähen" eng verwandt mit Floyds cycle finding

1. Speichere β_1

2. Ist β_i im Speicher, suche Koll mit $\beta_{i+1}, \dots, \beta_{2i}$

3. Bei Misserfolg, speichere β_{2i} und gehe zu 2.

Wird die „Spählänge" größer als der Zyklus
$\Rightarrow$ Koll.



Beispiel: $G = (\mathbb{Z}/_{11\mathbb{Z}})^*$ $\alpha = 5$ $\gamma = 2$ $n = 10$

$G_1 = \{1, 2, 3\}$ $G_2 = \{4, 5, 6\}$ $G_3 = \{7, 8, 9, 10\}$

i	x_i	y_i	$\beta_i = \gamma^{x_i}\alpha^{y_i}$	saved
1	2	0	4	✓
2	4	0	5	✓
3	8	0	3	
4	8	1	4	✓
5	16	2	5	
6	32	4	3	
7	32	5	4	

$8 - 32 \equiv x(5-1) \pmod{10}$

$6 \equiv x\,4 \pmod{10}$

$x = 4$ ✓

Beispiel: $G = (\mathbb{Z}/11\mathbb{Z})^*$ $\quad \alpha = 5 \quad \gamma = 2 \quad n = 10$

$G_1 = \{1, 2, 3\} \quad G_2 = \{4, 5, 6\} \quad G_3 = \{7, 8, 9, 10\}$

i	x_i	y_i	$\beta_i = \gamma^{x_i}\alpha^{y_i}$	saved	
1	2	0	4	✓	
2	4	0	5	✓	
3	8	0	3		
4	8	1	4	✓	$\Leftarrow$
5	16	2	5		
6	32	4	3		
7	32	5	4		$\Leftarrow$

$$\left.\begin{aligned} (x_i - x_{i+k}) &\equiv x\,(y_{i+k} - y_i) \pmod{n} \\ 8 - 32 &\equiv x\,(5-1) \pmod{16} \\ 6 &\equiv x\,4 \end{aligned}\right\} \Rightarrow x = 4 .$$

Danke für die Aufmerksamkeit!

Noch Fragen

?