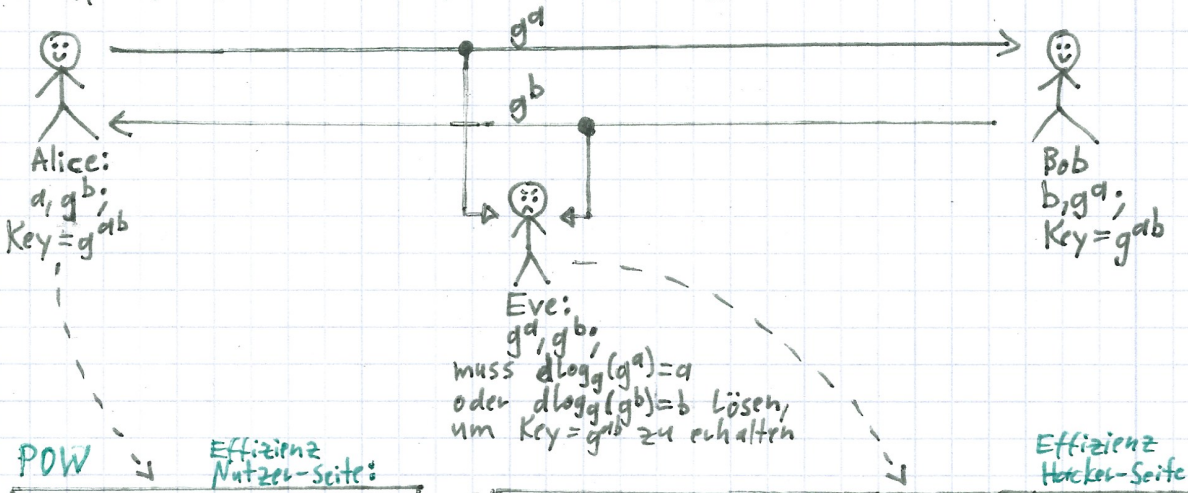
Eigenes „Gesamtbild" CRYPTO 1 – Diskr. Log. (I)

~~4.5.22~~ 5.5.22 & 7.5.22

Diffie-Hellman-Schlüsselaustausch zw. Alice & Bob:



POW → Effizienz Nutzer-Seite:

Wie berechnet Alice g^a und $(g^b)^a$ effizient?:

Square and Multiply:

$g^1 \xrightarrow{\wedge 2} g^{10} \xrightarrow{\wedge 2} g^{100} \to \dots$

z.B. $g^{18} = g^{10010} = g^{10000} \cdot g^{10}$

Register 1	Register 2
1	$\wedge 2 \wedge 2 \wedge \dots$
x	wenn ne 1

$(\Rightarrow \sigma(\log(g^a))$ Platz)(?)
$\Rightarrow \sigma(\log(a))$ Zeit

Effizienz Hacker-Seite: DLP

Wie berechnet Eve $x = dlog_g(g^x)$, gegeben $h = g^x$, effizient?:

1.) Silver-Pohlig-Hellman-Reduktion: $G = \{g^0, \dots, g^{n-1}\}$ zyklisch.

[1] $ord(G) = a \cdot b$ mit $ggT(a,b) = 1$ (Chinesischer Restsatz)
und DLP ist „leicht" in Ordnung a und b:
- Finde „leicht" x_a, x_b: $(g^b)^{x_a} = h^b$ (Ord. a); $(g^a)^{x_b} = h^a$ (Ord. b)
- Finde mit Euklid u, v: $ua + vb = ggT(a,b) = 1$
- Ergebnis: $x = u \cdot a \cdot x_b + v \cdot b \cdot x_a$
- Beweis: $g^x = (g^{ax_b})^u \cdot (g^{bx_a})^v = h^{au+bv} = h$ ■

[2] $ord(G) = p^e$ mit $p \in \mathbb{P}$ und $e > 1$
und DLP ist „leicht" in Ordnung p:
Idee: Reduziere rekursiv von Ord. p^e auf p^{e-1}:
- $h = g^x$
- $h = g^{x_0 + x_1 p + x_2 p^2 + \dots}$
- $h^{p^{e-1}} = (g^{x_0 + x_1 p + \dots})^{p^{e-1}}$ (schreibe x zur Basis p)
- $h^{p^{e-1}} = (g^{p^{e-1} x_0}) \cdot g^{p^{e-1} \cdot p \cdot (x_1 + x_2 p + \dots)}$
- $h^{p^{e-1}} = (g^{p^{e-1}})^{x_0} \cdot 1$ (da Ordnung p^e)
- $ord(\langle g^{p^{e-1}} \rangle) = p$, also x_0 „leicht" bestimmbar
- Neues Problem: $h = g^{x_0} \cdot (g^p)^{x_1 + x_2 p + \dots}$
  $h \cdot g^{-x_0} = (g^p)^{x_1 + x_2 p + \dots}$
  → DLP in $\langle g^p \rangle$ der Ordnung p^{e-1} ■

$\Rightarrow$ Bleibt das DLP in (zykl.) Gruppen mit Primzahlordnung:

2a.) Baby-step-Giant-step-Algo / Shanks' Algo: (beliebige zykl. Grp.)
($|\langle g \rangle| = n$)

Giant Steps: $g^0, g^N, g^{2N}, \dots, g^{(N-1)N}$ $(N = \lceil \sqrt{n} \rceil)$
Baby Steps: $h, hg, hg^2, \dots$
Irgendwann muss Kollision: $g^{uN} = hg^{-v}$
$\Rightarrow x = dlog_g(h) = uN + v$ ■

2b.) Pollard-ρ („Rho"): $f(x) = \begin{cases} xh, & x \in G_1 \\ x^2, & x \in G_2 \\ xg, & x \in G_3 \end{cases} \Rightarrow x_i = g^{u_i} \cdot h^{v_i}$

$x_L = x_{L+T} \Rightarrow g^{u_L} \cdot h^{v_L} = g^{u_{L+T}} \cdot h^{v_{L+T}}$
$\Rightarrow g^{u_L - u_{L+T}} = h^{v_{L+T} - v_L} = (g^x)^{v_{L+T} - v_L} \Rightarrow x = \frac{u_L - u_{L+T}}{v_{L+T} - v_L}$ ■

$\Rightarrow$ Beide basieren auf dem Geburtstagsparadoxon ($\sqrt{|G|}$) und sind besser als naives Brute-Force, aber dennoch für große Gruppen n. durchf.bar!