Compositional problems: In the bilancy, bet G be a first right game generated by g with order g
Theorem by problem: comple
$$\pi \in \mathbb{Z}_{q}$$

gran h g^{π} , compare π
⁻ Compositional Differ Hullman (COM): sample $\pi, g \in \mathbb{Z}_{q}$
gran $g^{\pi}, g^{\pi}, g^{\pi}$

When describing applographic constructions, we will work with an abotract group (easier to work with, less destriks to worry about)

hidely used for key-exchange + signatures on the web

Diffie-Hellman key exchange

But usually, we want a random bit-string as the key, not random group element

L> Element g^{xy} has log p bits of entropy, so should be able to obtain a roundom bit-string with l< log p bits L> Solution is to use a "randomness extractor"

- L> Information-theoretic constructions based on universal hoshing / pairwise-independent hashing good precise to (loses some bits of entropy)
- (1005 some UTS of entropy) L> Use a "random oracle" or an "ideal hash function" [Heuristic: SHA-256 (g, g^x, g^y, g^y, g^{xy})] [binds the key to the entire (very efficient in practice)
 - $\stackrel{\text{L}}{\longrightarrow} \frac{\text{Arguing security}}{\text{Arguing security}} \stackrel{\text{!}}{\text{!}} \text{ Rely on HashDH assumption } (g, g^{x}, g^{y}, H(g, g^{x}, g^{y}, g^{x, y}) \stackrel{\times}{\approx} (g, g^{x}, g^{y}, r)$ where $H : \mathbb{G} \xrightarrow{} \{0, 13^{n}\}$ and $r \stackrel{\text{d}}{\approx} \{0, 13^{n}\}$

Pe for3

2. Model H as ideal hash function $H: \mathbb{G}^{4} \rightarrow \{0,13^{n} \text{ (i.e., random oracle)} and rely on CDH in <math>\mathbb{G}$ [inability to evaluate H on $g^{xy} \Longrightarrow$ output is random string]

Public-key encryption: Encryption scheme where encryption is public (does not require shored secrets)

- \neg Setup $(1^{\lambda}) \rightarrow (pk, sk)$ [generates a public/private key-pair also called KeyGen]
- [−] Encrypt (pk, m) → c
- Decrypt (sk, c) → m. Everyone can publish a public key (in a directory)

-> Can encrypt to anyone without exchanging keys (recipient can be offline)

Security: semantic security from secret-key setting, but adversary also gets public key

adversary

$$(pk, sk) \leftarrow Setup(1^{\lambda})$$

 $(pk, sk) \leftarrow Setup(1^{\lambda})$
 $(pk, sk) \leftarrow Setup(1^{\lambda})$
 $(pk, sk) \leftarrow Setup(1^{\lambda})$
 $(pk, sk) \leftarrow Setup(1^{\lambda})$
 $(pk, sk) \leftarrow Setup(1^{\lambda})$