To Label, or Not To Label (In Generic Groups)

Mark Zhandry (NTT Research & Princeton University)

Two very different

~~The~~ generic

group models

Shoup’96: Random labels

L = Random Injection Z_p → $\{0,1\}^n$

Interpret $L(x)$ as g^x

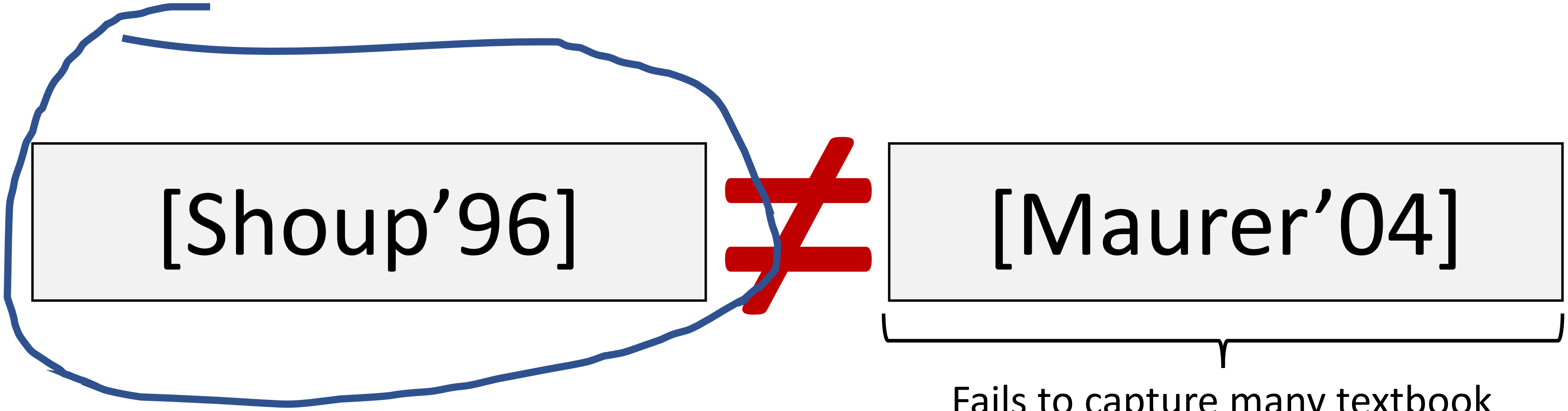
Oracle:
$\mathrm{Mult}(L(x),L(y)) = L(x+y)$

Maurer’04: Pointers

```
Mult(Element h1, Element h2) {
    return new Element(
        h1.value * h2.value);
}
EqualQ(Element h1, Element h2) {
    return h1.value==h2.value;
}
```

No other operations on
`Element` variables allowed

TLDR:



[Shoup’96] ≠ [Maurer’04]

preferred

Fails to capture many textbook generic techniques

An apparent contradiction:

[Jager-Schwenk'08]: "In this paper we prove the equivalence of the models proposed by Shoup and Maurer"

Intuition: can't do anything with random label other than feed it back into oracle

VS

Thm [Chen-Lombardi-Ma-Quach'21]:

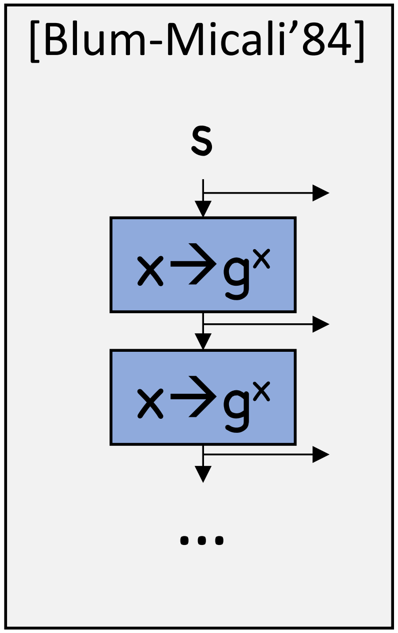
Schnorr secure in Shoup, even with non-cryptographic hash for Fiat-Shamir

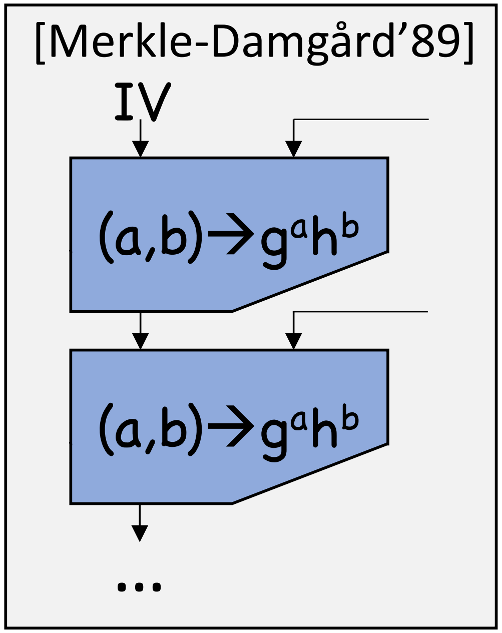
+

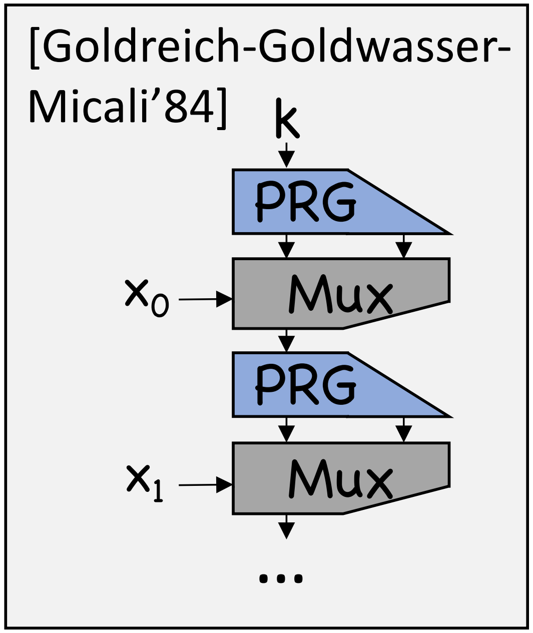
Thm [Döttling-Hartmann-Hofheinz-Kiltz-Schäge-Ursu'21]:

Signatures impossible in Maurer

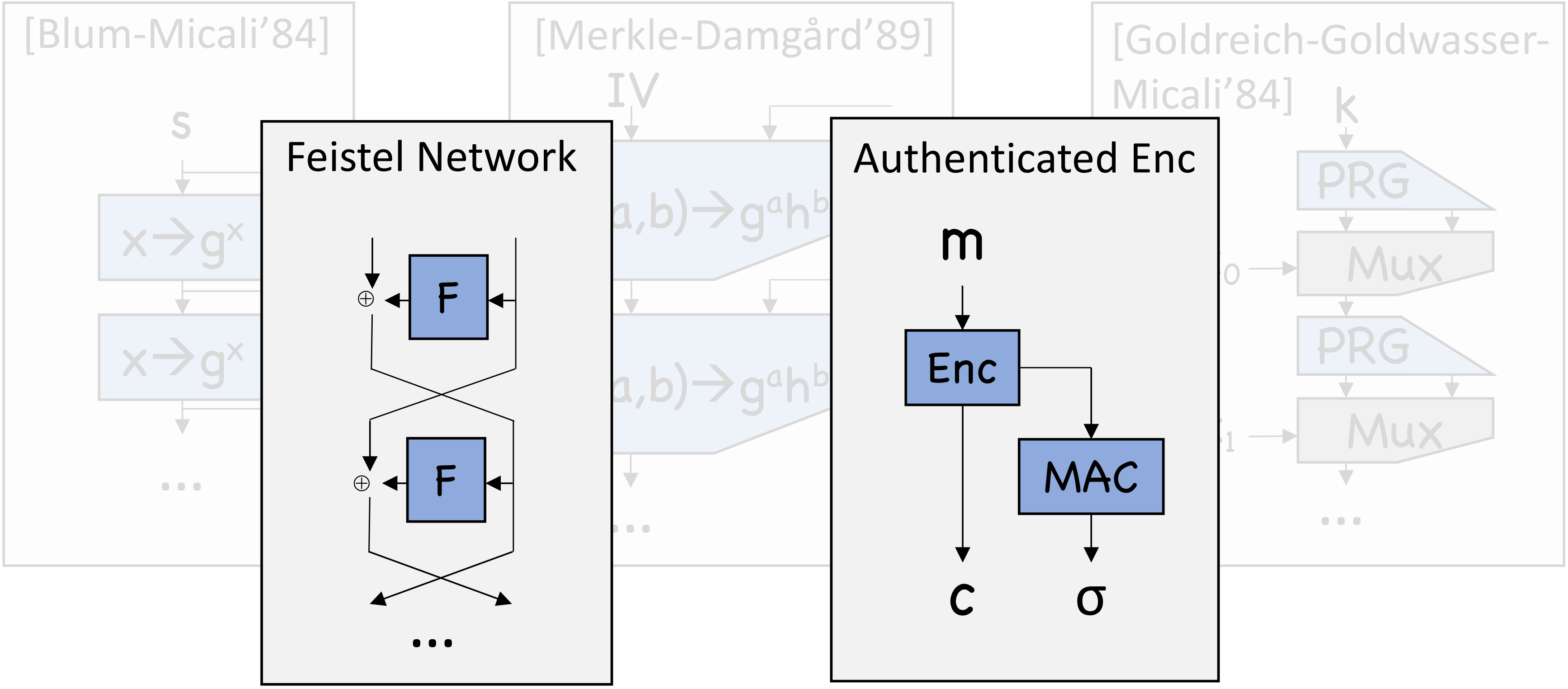
Starting observation: textbook techniques that fail in Maurer


[Blum-Micali’84]

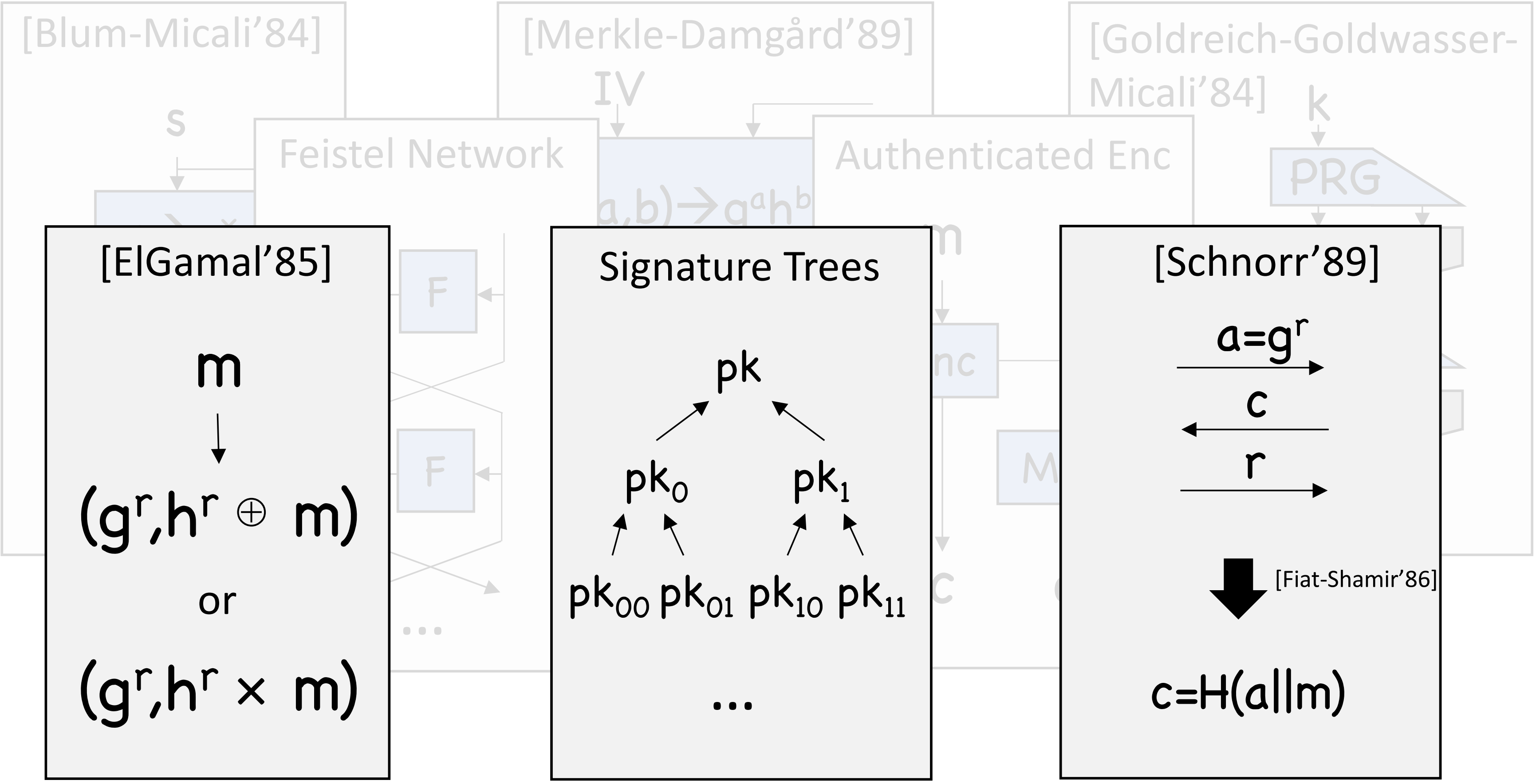

[Merkle-Damgård’89]


[Goldreich-Goldwasser-Micali’84]

Starting observation: textbook techniques that fail in Maurer



Starting observation: textbook techniques that fail in Maurer



[ElGamal'85]

$$m \downarrow$$

$$(g^r, h^r \oplus m)$$

or

$$(g^r, h^r \times m)$$

Signature Trees

pk

pk_0 pk_1

pk_{00} pk_{01} pk_{10} pk_{11}

...

[Schnorr'89]

$\xrightarrow{a=g^r}$

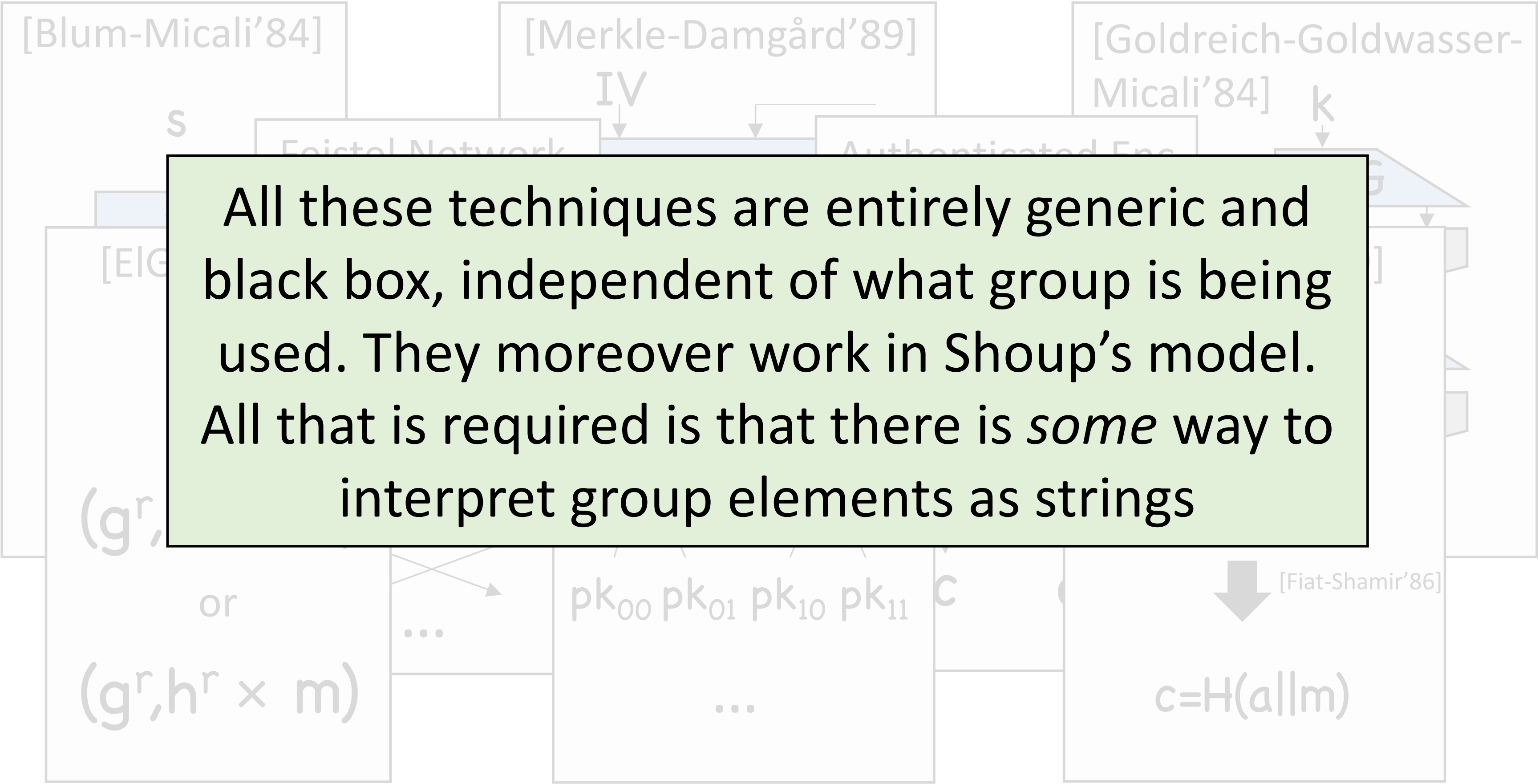
$\xleftarrow{c}$

$\xrightarrow{r}$

[Fiat-Shamir'86]

$c=H(a||m)$

Starting observation: textbook techniques that fail in Maurer



All these techniques are entirely generic and black box, independent of what group is being used. They moreover work in Shoup's model. All that is required is that there is *some* way to interpret group elements as strings

Our Results, Part I

Thm: $\nexists$ CRHFs with unbounded domain in Maurer

Thm: $\nexists$ PRPs in Maurer

Thm: $\nexists$ rate-1 encryption in Maurer

Black box separations in Maurer must be taken with grain of salt

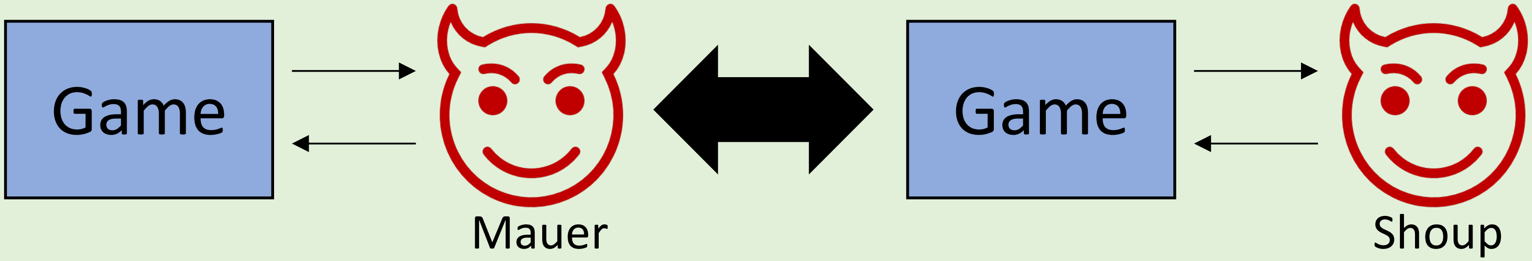
So what’s the deal with Jager-Schwenk?

Historical note: Generic groups originally only used for analyzing hardness of computational problems. Use for *impossibilities* came later

[Dodis-Haitner-Tentes’12, Cramer-Damgård-Kiltz-Zakarias-Zottarel’12, Papakonstantinou-Rackoff-Vahlis’12]

So what’s the deal with Jager-Schwenk?

Thm [Jager-Schwenk’08]:



Game

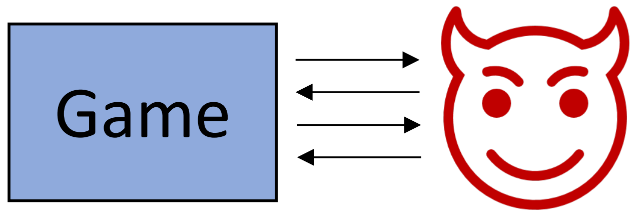
Mauer

Game

Shoup

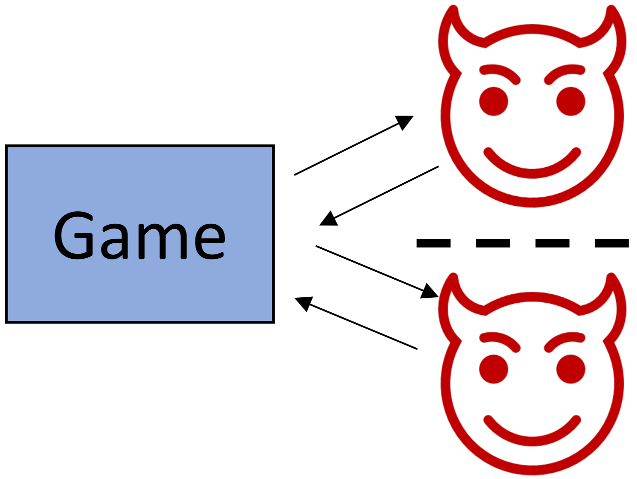
Important: Only sensible if game works in both models!

Single stage



Examples: essentially all of the “basic” security games

Multi-stage



Examples: deterministic encryption, leakage resilience, auxiliary input one-wayness, etc

Our Results, Part II

Thm: Any cryptosystem/game in Maurer also works in Shoup

Part I and prior work already disproved converse

Thm: Amongst Maurer games, Shoup security → Maurer security

Thm: Amongst **single-stage** Maurer games, Maurer security → Shoup security

Re-interpretation of Jager-Schwenk

Thm: $\exists$ multi-stage Maurer game secure in Maurer but not in Shoup

(Also insecure in any standard-model group)

Def: Uninstantiability result = secure in generic group model + insecure in any actual group

Observation: All existing single-stage generic group uninstantiability results only work in Shoup

Typical technique: break scheme by finding code <H> such that H(x)=L(x)

Could Maurer single-stage games avoid uninstantiability results?

Our Results, Part III

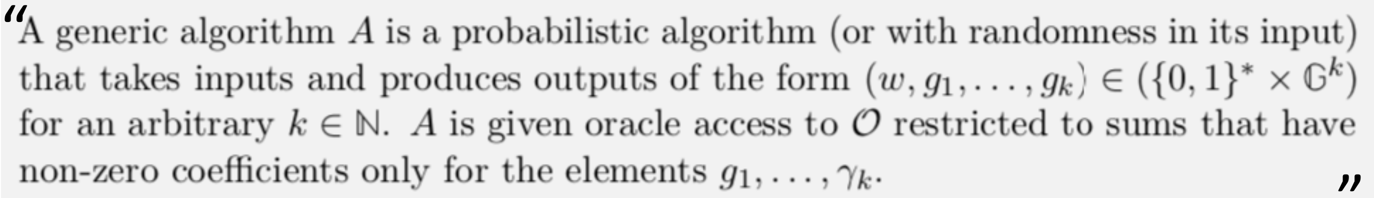
Thm: ∃ single-stage Maurer game secure in Maurer but not in real world

Bitwise ElGamal + one extra (contrived) bit

$c = (g^{r1} , h^{r1+m1} , \ldots, g^{rn} , h^{rn+mn}, L(m,r_1,\ldots,r_n))$

Thm [Papakonstantinou-Rackoff-Vahlis’12]:
No IBE in *some* generic group model

Claim Shoup, but…



“A generic algorithm A is a probabilistic algorithm (or with randomness in its input) that takes inputs and produces outputs of the form $(w, g_1, \ldots, g_k) \in (\{0,1\}^* \times \mathbb{G}^k)$ for an arbitrary $k \in \mathbb{N}$. A is given oracle access to $\mathcal{O}$ restricted to sums that have non-zero coefficients only for the elements $g_1, \ldots, \gamma_k$. ”

This is a Maurer-style restriction!

Used in crucial step of proof to compile out group elements in secret keys

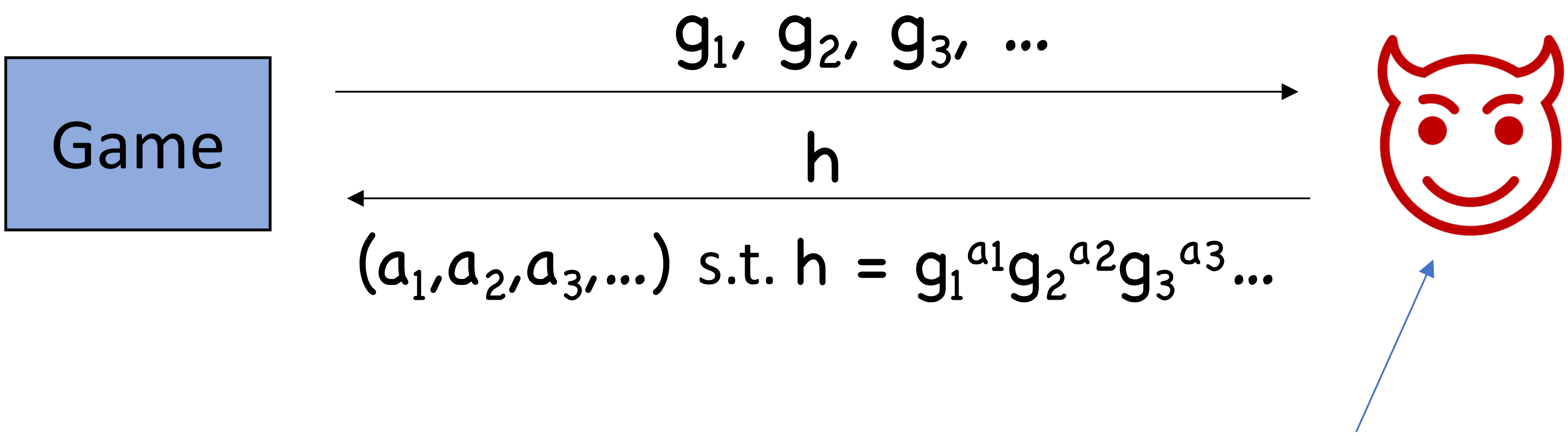
Our Results, Part IV

Thm: IBE impossible in Shoup's model

Adapt existing techniques, but make sure
every step makes sense in Shoup

Algebraic Group Model (AGM)

[Fuchsbauer-Kiltz-Loss’18], building on [Paillier-Vergnaud’05]



Game

$g_1, g_2, g_3, \dots$

h

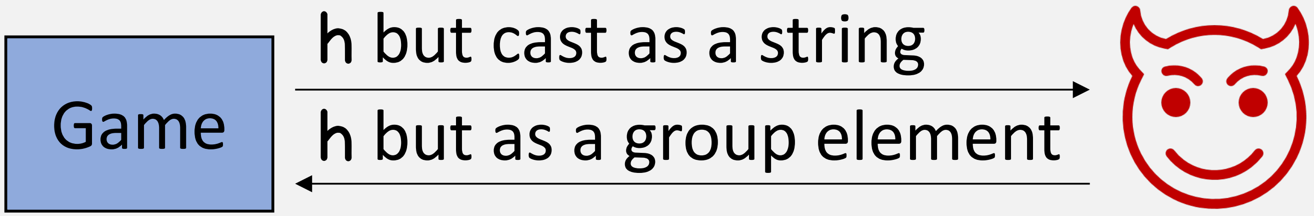
$(a_1,a_2,a_3,\dots)$ s.t. $h = g_1^{a_1} g_2^{a_2} g_3^{a_3} \dots$

Non-black box access to group

Often claimed to be “between” generic groups and standard model

Observation: AGM not fully defined by FKL

Trivial uninstantiability:



Game

h but cast as a string

h but as a group element

Finding representation impossible!

[FKL]: Syntactically distinguish group elements from non-group elements, non-group elements must not “depend” on group elements

What does “depend” mean?

Our position:

AGM only applies to Maurer games

[Katz-Zhang-Zhou’22]:
Different interpretation

Our Results, Part V

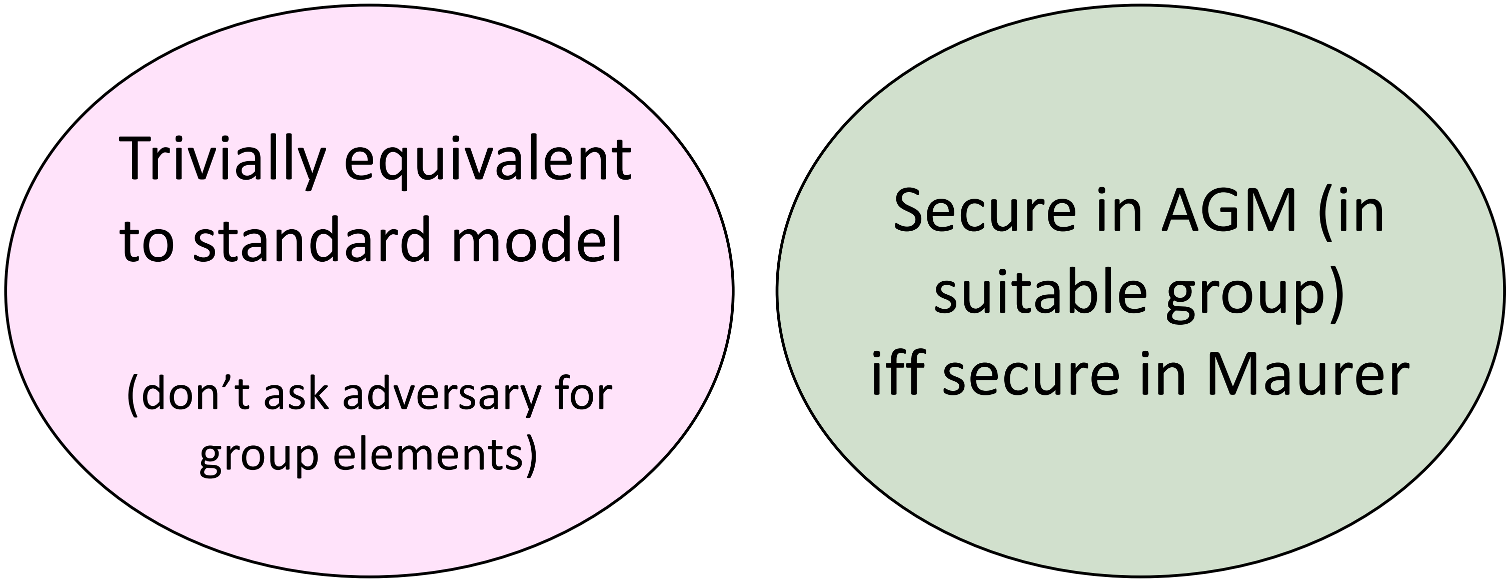
Under our interpretation:

Cor: AGM *incomparable* to Shoup

Thm: $\exists$ single-stage Maurer game secure in AGM but not in real world

Open question

Existing games in AGM:



Trivially equivalent to standard model

(don’t ask adversary for group elements)

Secure in AGM (in suitable group) iff secure in Maurer

Q: Are there any games that don’t fit into these two buckets?

Summary

[Shoup’96] ≠ [Maurer’04]

Black box separations:

Shoup preferred, Maurer may provide useful guidance

Security proofs:

Shoup = Maurer for “single-stage” games

Maurer seems unsuitable for “multi-stage” games

Thanks!