CS 6846: Quantum Algorithms and Cryptography

Lec 7: Basics of Cryptography

Shweta Agrawal
IIT Madras

Cryptography

- A mathematical science of controlling access to *information*
- Cryptography deals with methods for protecting the *privacy and integrity* while *preserving functionality* of computer and communication systems.

What would we like to achieve?

Examples

#1 : Secure Elections

Multi-party computation!

VOTES

VOTE COUNTING

Winner ?

CORRECT : Winner determined correctly

SECURITY : individual vote privacy maintained

#2 : Protecting your code

I know a better algorithm to factor numbers!

code → OBFUSCATOR →

Program Obfuscation!

Obfuscated code

```
#include<stdio.h> #include<string.h>
main(){char*O,l[999]="'`acgo\177~|xp .
-\0R^8)NJ6%K4O+A2M(*0ID57$3G1FBL";
while(O=fgets(l+45,954,stdin)){*l=O[
strlen(O)[O-1]=0,strspn(O,l+11)];
while(*O)switch((*l&&isalnum(*O))-!*l)
{case-1:{char*I=(O+=strspn(O,l+12)
+1)-2,O=34;while(*I&3&&(O=(O-16<<1)+
*I---'-')<80);putchar(O&93?*I
&8||!(  I=memchr( l , O , 44 ) ) ?'?':
I-l+47:32); break; case 1: ;}*l=
(*O&31)[l-15+(*O>61)*32];while(putchar
(45+*l%2),(*l=*l+32>>1)>35); case 0:
putchar((++O ,32));}putchar(10);}}
```

- Produces correct output
- Impossible to reverse engineer

#3 : Activism with safety

Probabilistic algorithm

C = Encrypt ("The election was rigged", R)

R, R' : Random bits

Under coercion, reveal R' s.t. C =("Really like to cook", R')

Deniable Encryption!

#4: Computing on encrypted data

- Users access data and infrastructure on-the-go
- Cloud stores data about you, me and many more
- I should only learn information I am authorized to learn

Encrypted Computation
Personalised Medicine

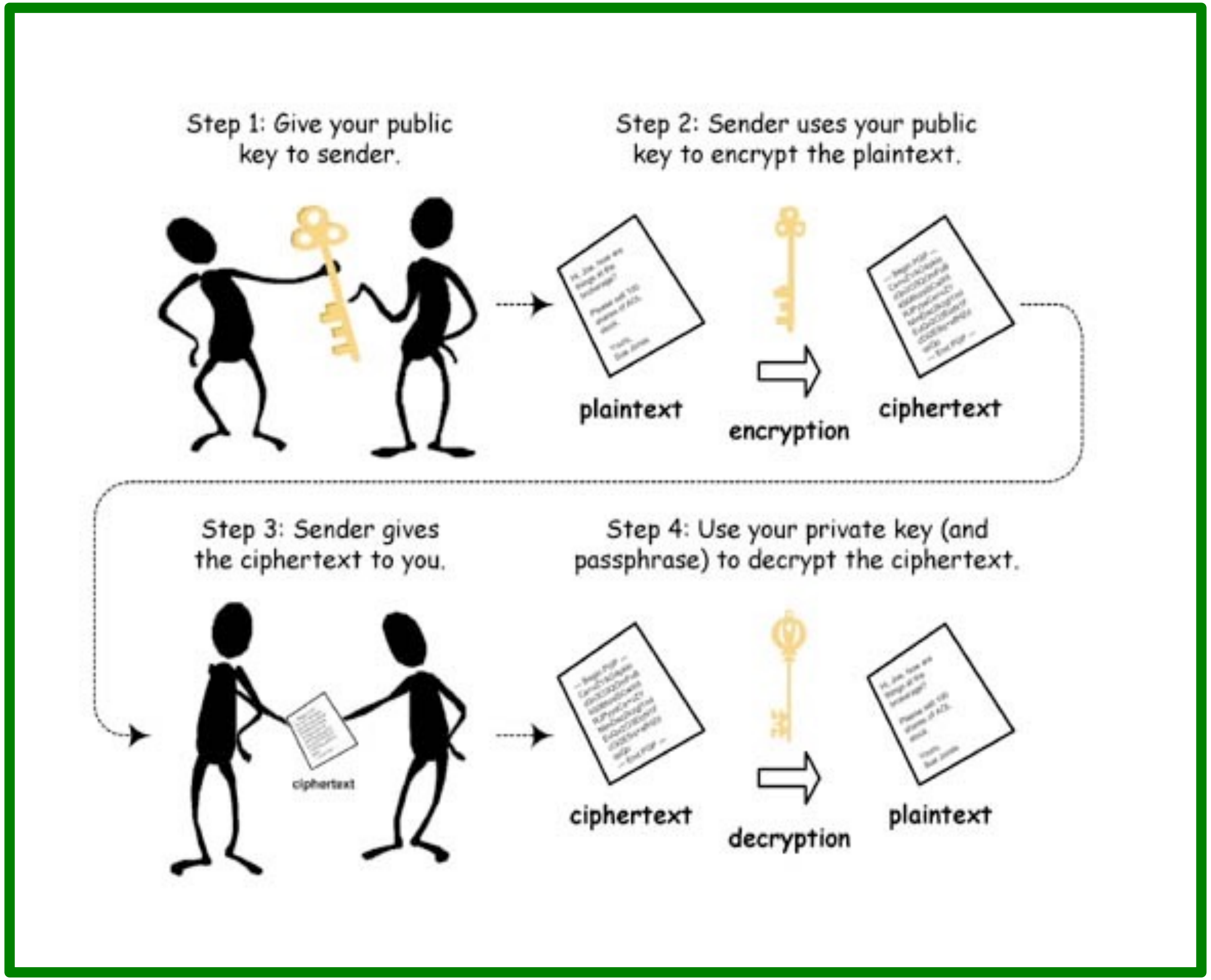
"The dream for tomorrow's medicine is to understand the links between DNA and disease — and to tailor therapies accordingly. But scientists have a problem: how to keep genetic data and medical records secure while still enabling the massive, cloud-based analyses needed to make meaningful associations."

"You don't look anything like the long haired, skinny kid I married 25 years ago. I need a DNA sample to make sure it's still you."

Check Hayden, E. (2015). *Nature*, *519*, 400-401.

Can Cryptography solve this?

Public Key Encryption



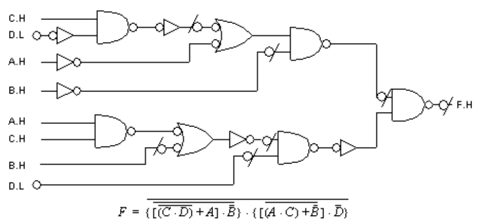
PKE does not suffice!

- Secret keys correspond to users
- Encrypt for each user?
- All or nothing access
  - Genomic data (for instance) is too sensitive to share
  - May be willing to participate in study which reveals output (result of study) without revealing input (personal data)

More Expressive Encryption

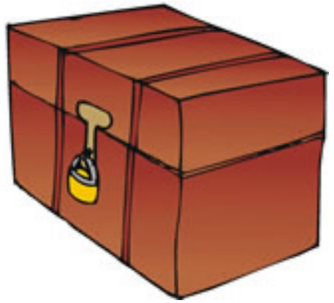
Functional Encryption!

Secret Keys
for functions *F*



+

Ciphertexts
for inputs *x*



Decryption recovers F(x)

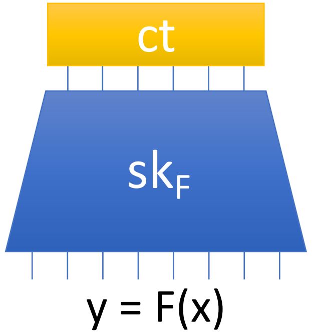
F : Age distribution of people with lung cancer
X : particular user’s disease profile

Encryption with Partial Decryption Keys

Encrypt (x):

Keygen(F):

Decrypt (sk_F, ct):



Security:

Adversary possessing keys for multiple circuits F_i cannot distinguish Enc(x_0) from Enc(x_1) unless $F_i(x_0) \neq F_i(x_1)$

Functional Encryption [SW05,BSW11]

Personalized Medicine?

Encrypt
input = genomic data of users

ct(Anuja)

ct(Simran)

ct(Anshu)

ct(Shweta)

Keygen
input: some medical research algo

Decrypt (sk_F, ct):

ct(Anuja) ct(Simran)

ct(Anshu) ct(Shweta)

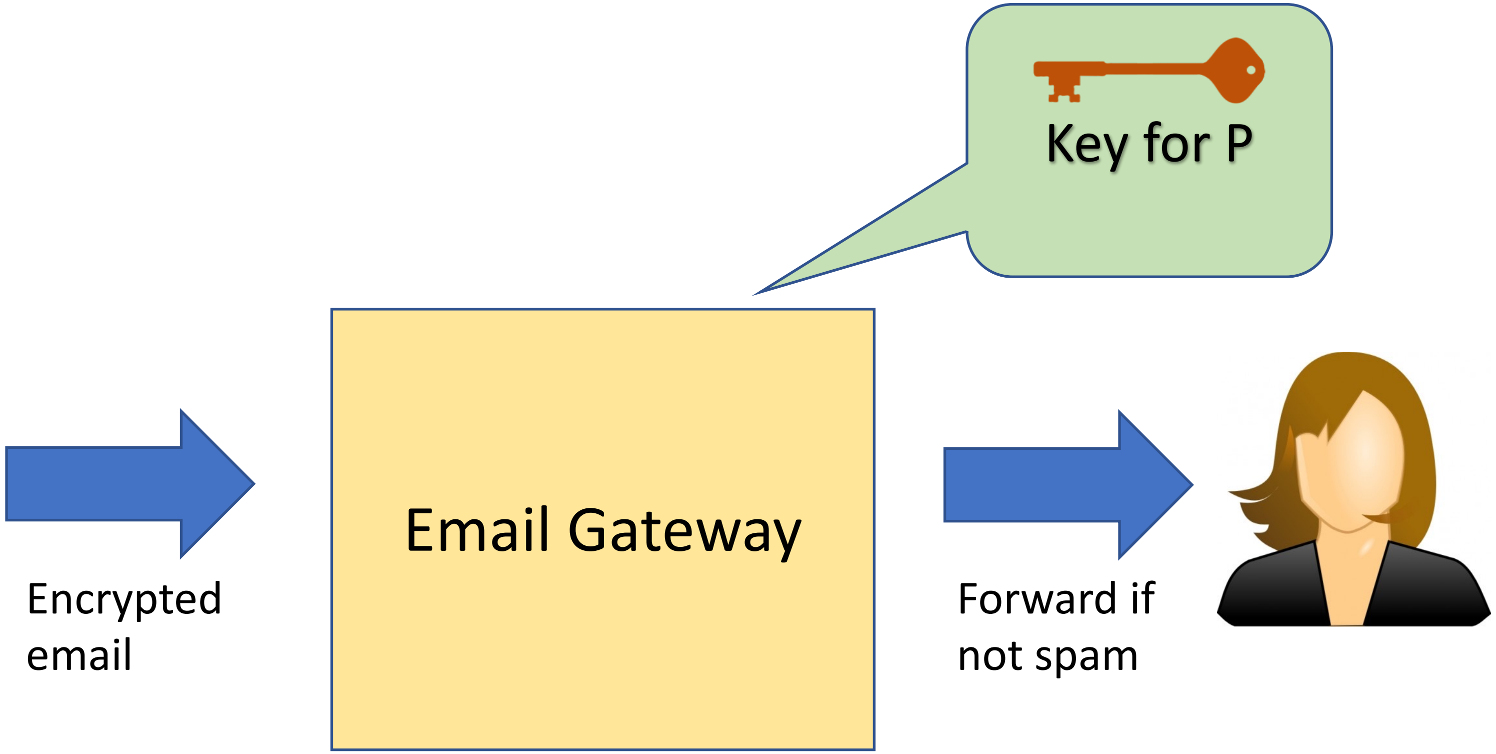
$y = F(x)$

Security: No one's personal genomic data is leaked!

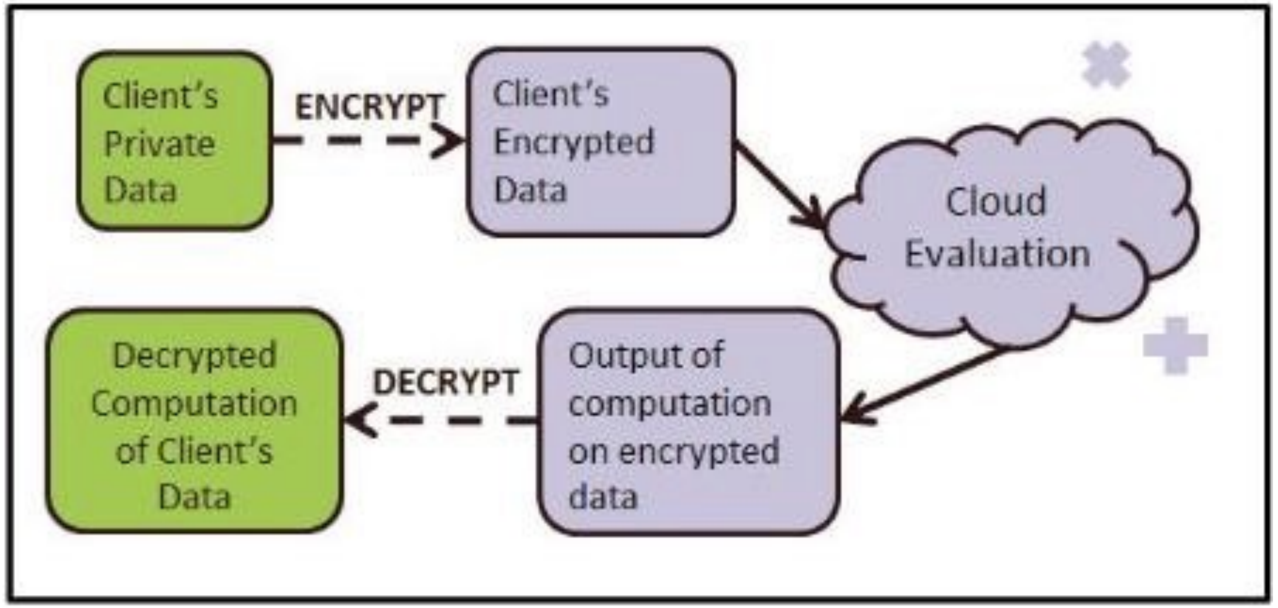
Functional Encryption [SW05,BSW11]

Spam Detection on Encrypted Email

Say we have a program P to detect spam on unencrypted email.



#5: Fully Homomorphic Encryption



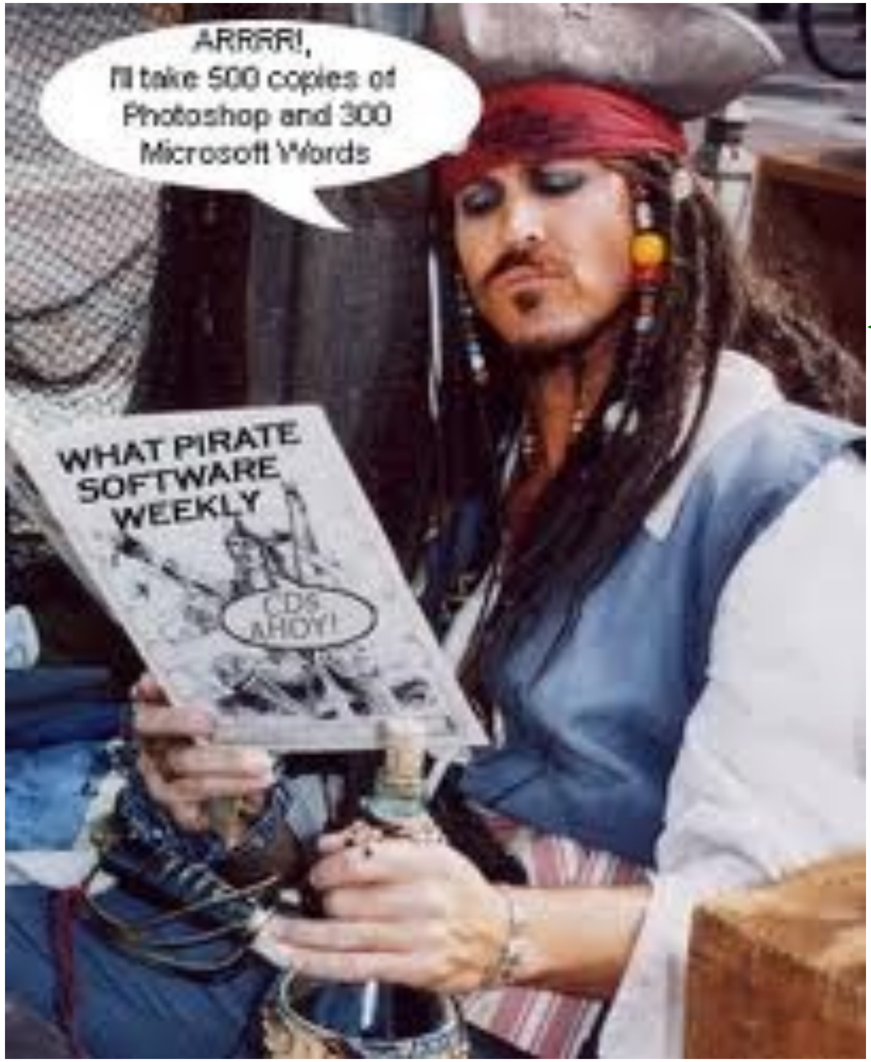
Expressive Functionality: Supports arbitrary circuits

Compact ciphertext, independent of circuit size

Perfect: Encrypted computation with All or Nothing Decryption

* : up to minor variations

#6: Traitor Tracing



I'll buy one license
And use it to forge
and sell new
licenses …

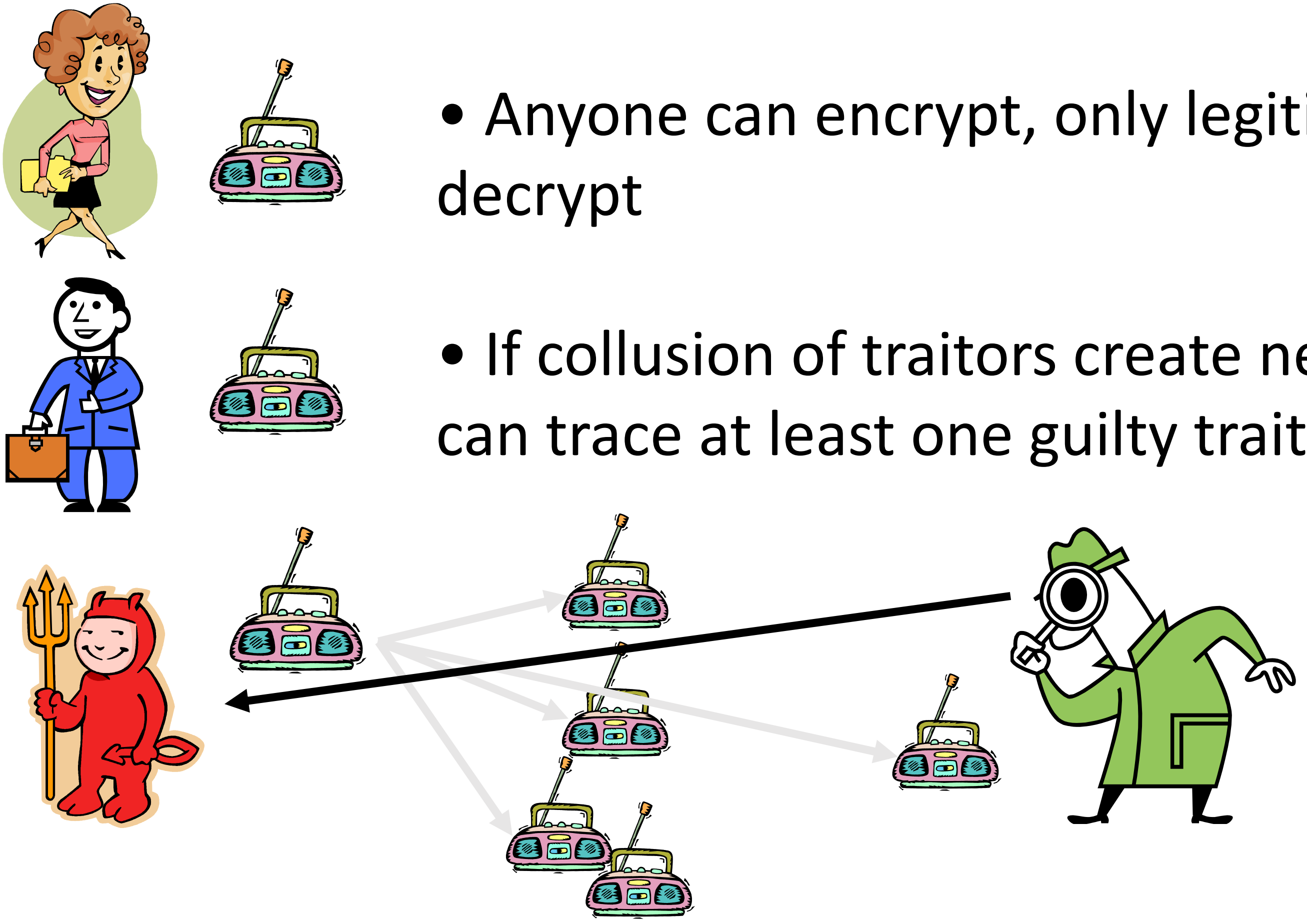
Can we catch him ?

#6: Traitor Tracing

- N users in system, One PK, N SKs

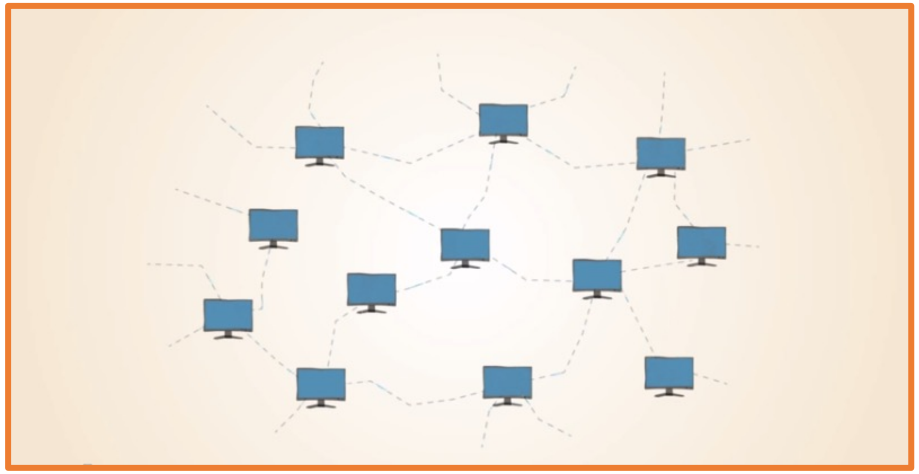
- Anyone can encrypt, only legitimate user should decrypt

- If collusion of traitors create new secret key SK^*, can trace at least one guilty traitor.

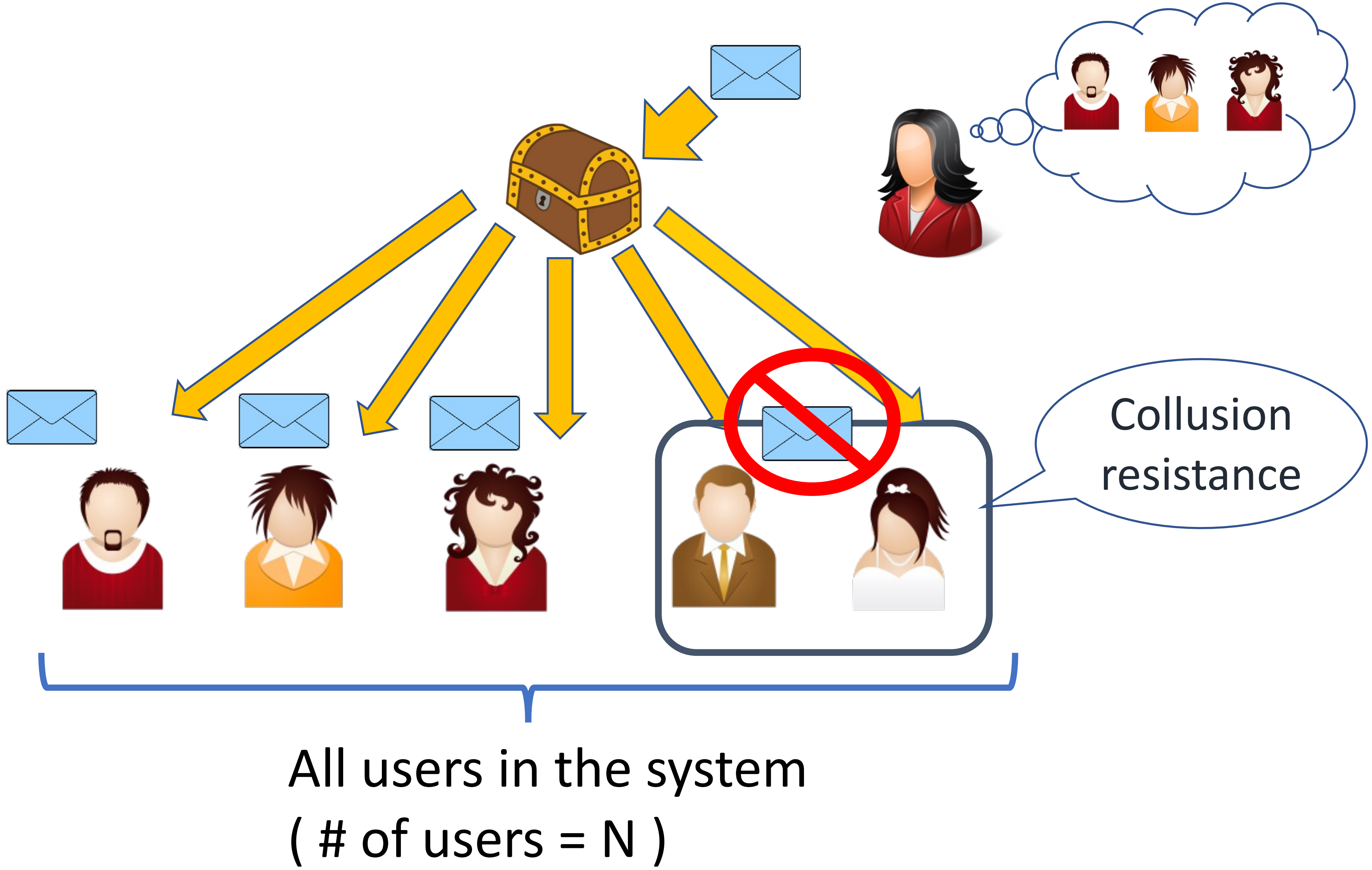


#7: Bitcoins

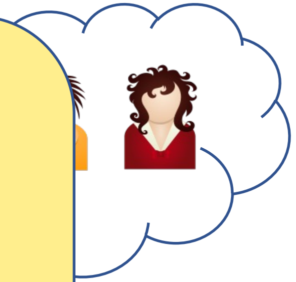
- Bitcoins uses cryptography to authenticate transactions, prevent theft and double spending, incentivize honest behaviour (we'll see how).
- It is underpinned by a peer to peer network made up of its users machines, akin to the networks that underpin BitTorrent and Skype.



#8: Broadcast Encryption



Broadcast Encryption

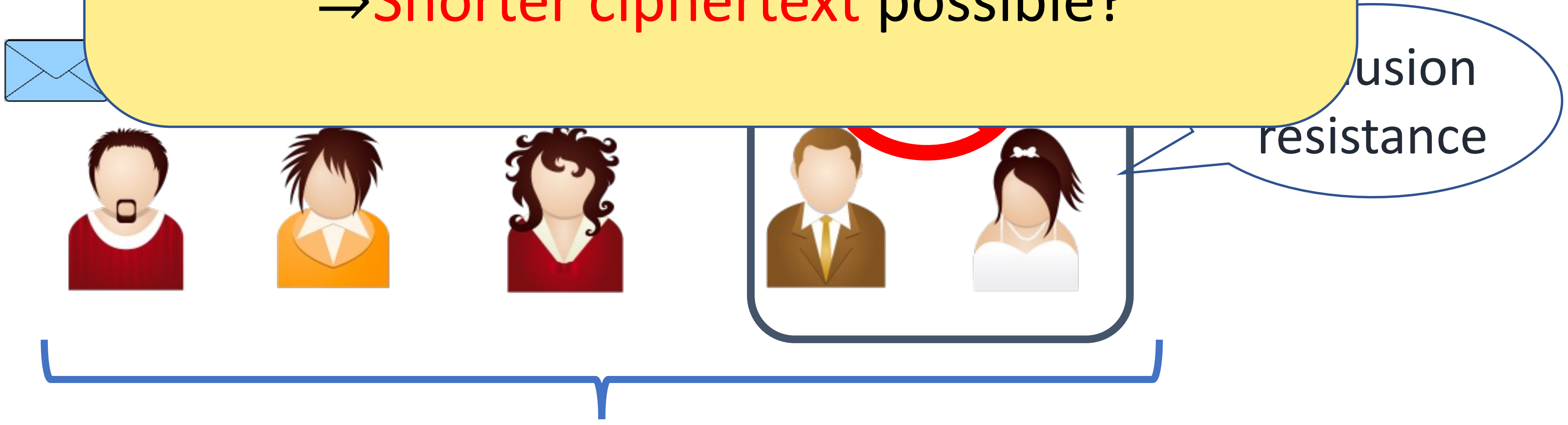


<u>Trivial solution:</u>

Encrypt message to each user using PKE.

O(N) ciphertext!

⇒Shorter ciphertext possible?



All users in the system
(# of users = N)

#9: Zero Knowledge

- How do you prove you know something without revealing what you know?
- When you write exams, you prove to the instructor that you know a solution by writing down the solution
- Suppose you want to sell an idea but want to reveal the idea *after* you get the money?

Sudoku

8			4		6			7
						4		
	1					6	5	
5		9		3		7	8	
				7				
	4	8		2		1		3
	5	2					9	
		1						
3			9		2			5

Fill in the grid so that :

- Every row has digits 1-9
- Every column has digits 1-9
- Every cube has digits 1-9

Zero Knowledge proof

- I want to convince my friend that I know the solution to the Sudoku puzzle without telling him the solution

- Lets be a bit formal. We have two players : a prover and a verifier

- Our tool will be packs of cards

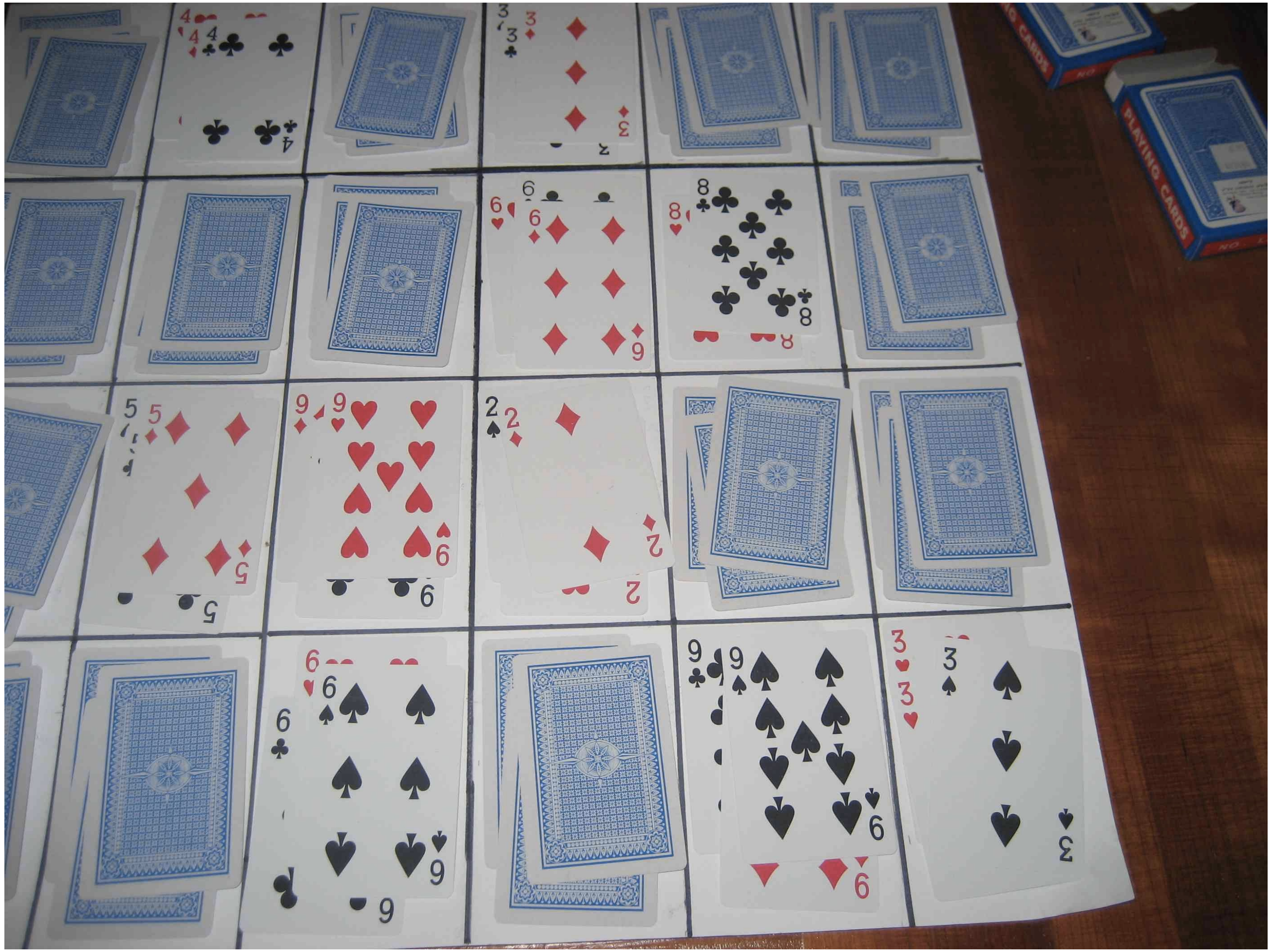
Step 1

- Draw the 9 by 9 grid and place cards face-up on it

Step 1

Step 2

- Prover places cards corresponding to her solution face down.
- She places three identical valued cards face down



Step 3

- Verifier starts making packets one for each column
- From each cell in the column one of three cards is chosen at random
- Similarly verifier makes packets for rows
- Lastly, verifier makes packets for subgrids

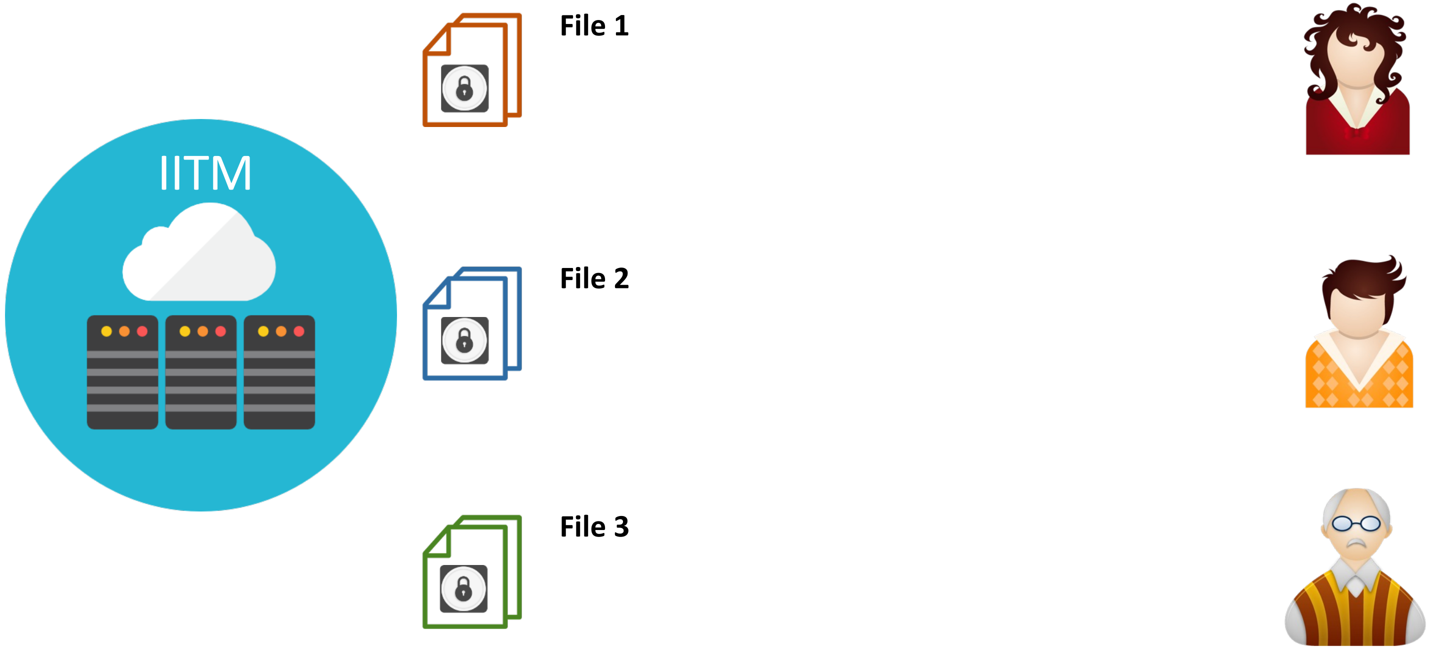
Step 4

- Prover turns over the cards which are facing up in each packet and shuffles them.
- Verifier opens each packet to see that all 9 values occur in each packet

- Can the prover cheat?
- If so, with what probability of success?
- Does the verifier learn anything about the solution?

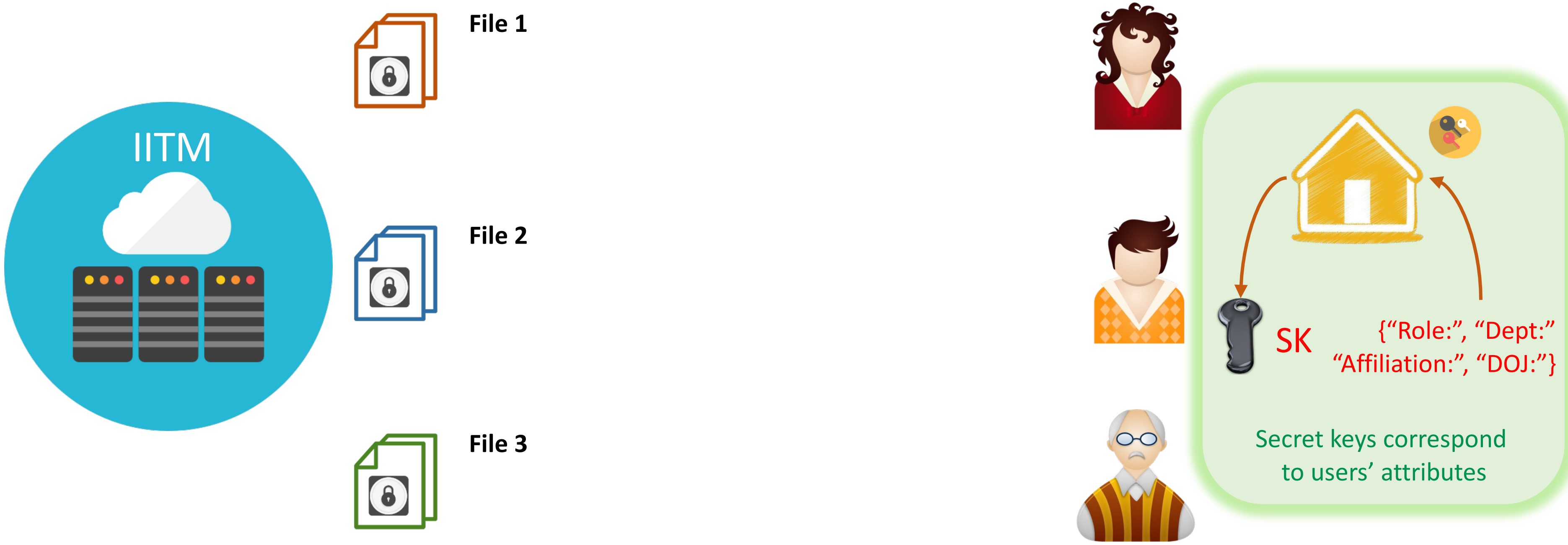
#10: Attribute based Encryption

[SW05, GPSW06]



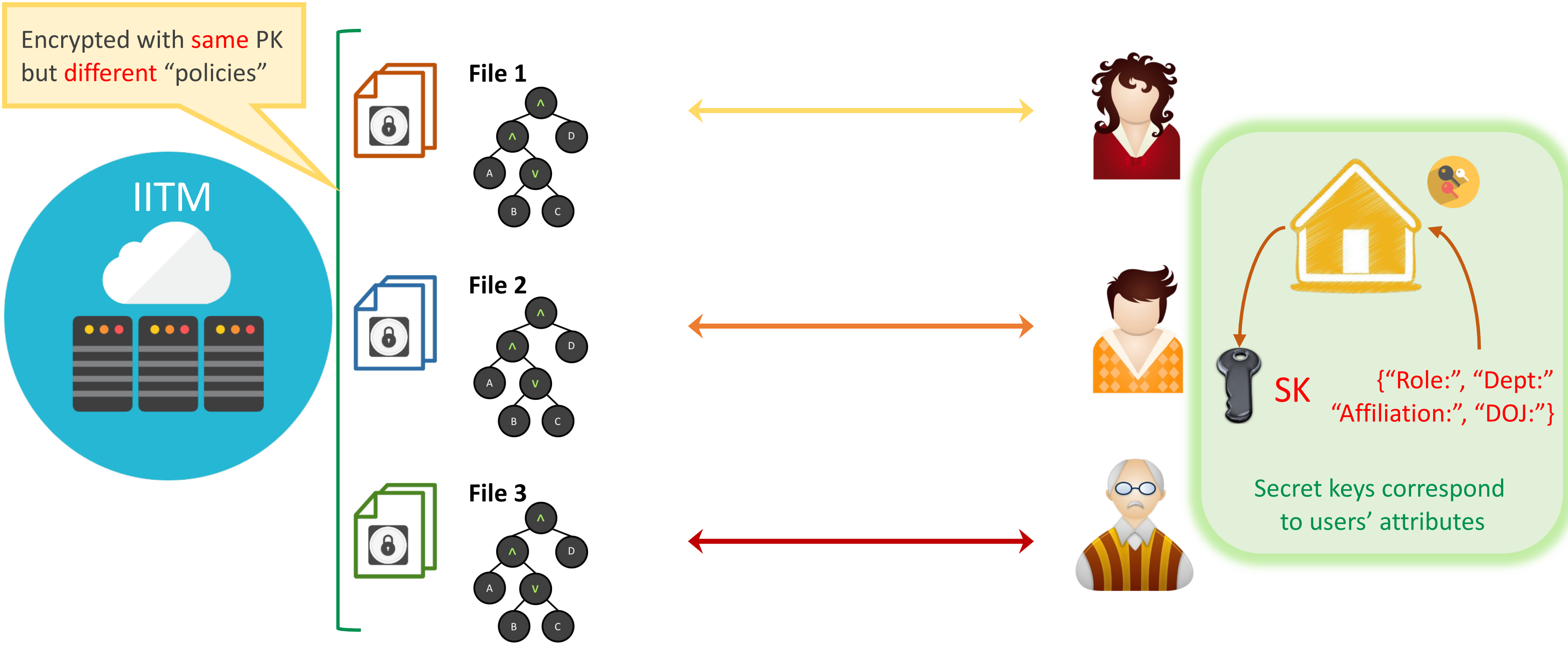
Attribute based Encryption

[SW05, GPSW06]



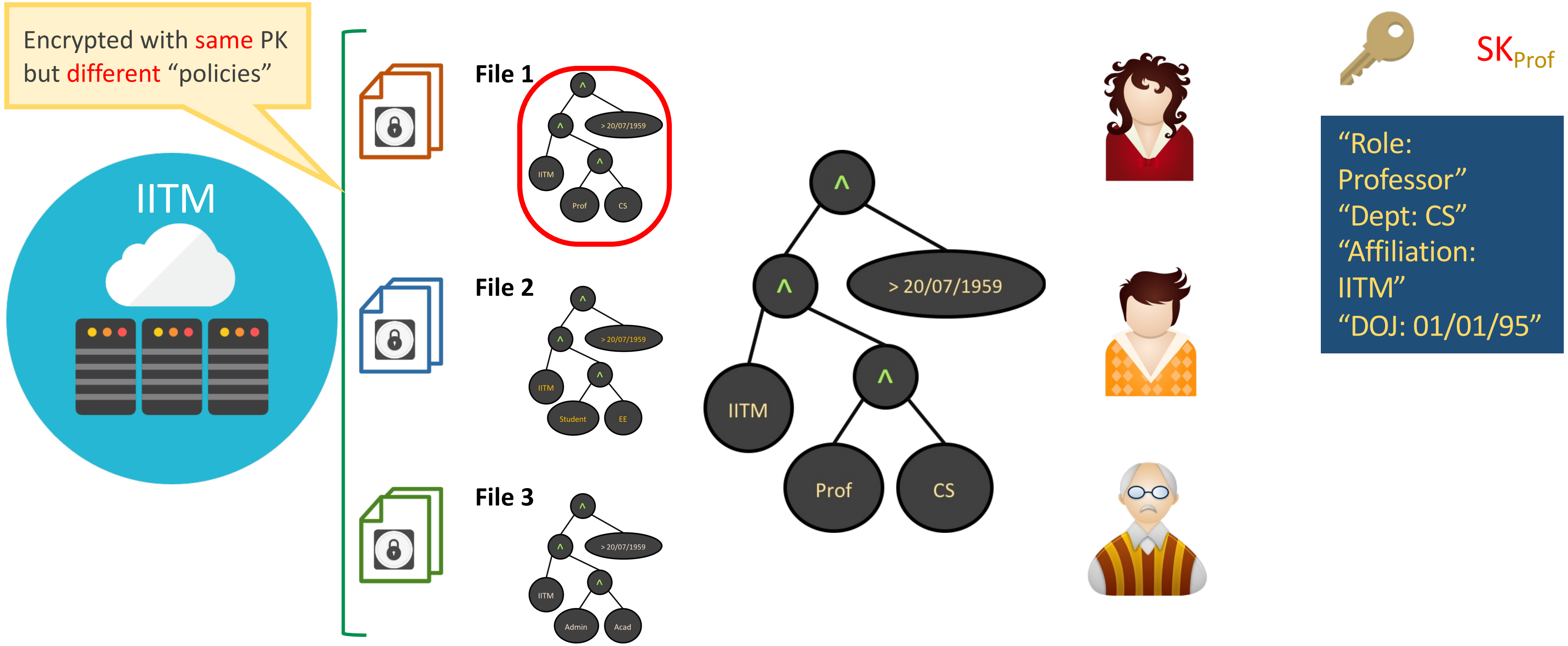
Attribute based Encryption

[SW05, GPSW06]



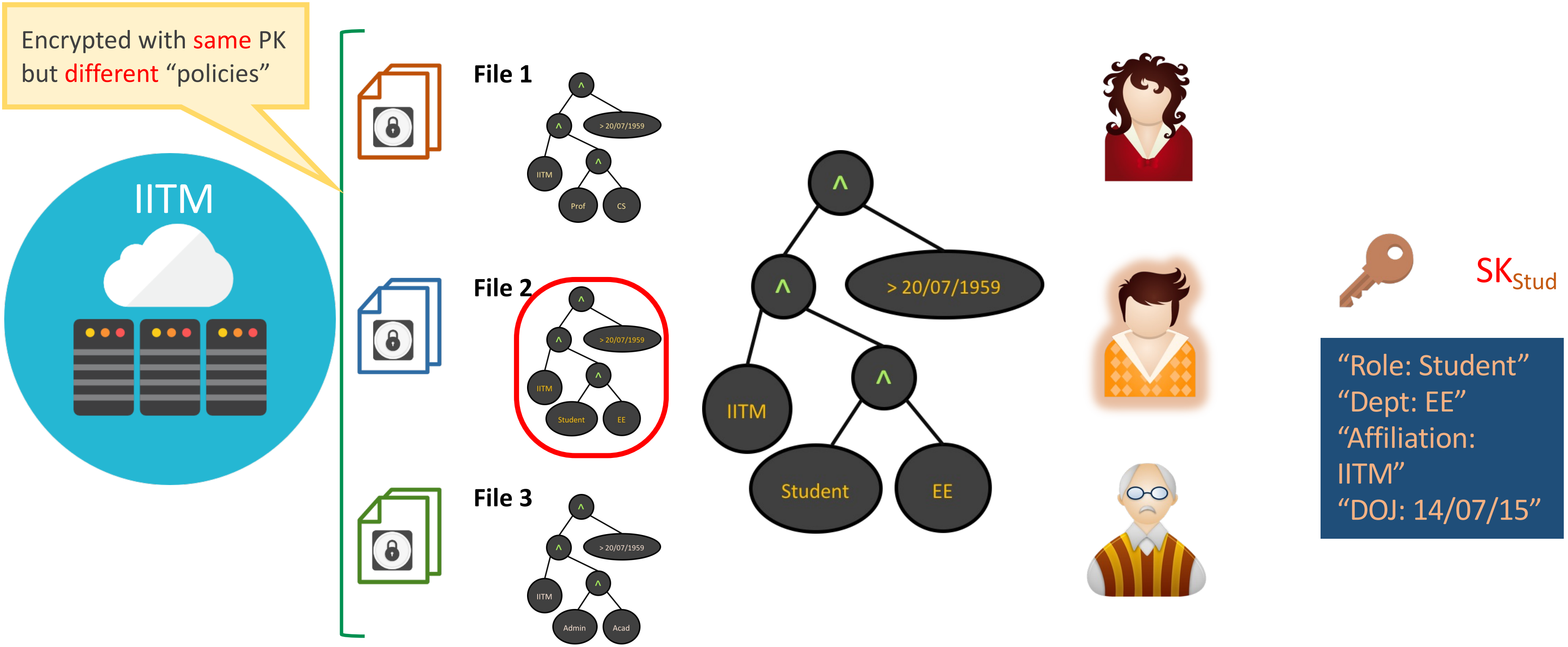
Attribute based Encryption

[SW05, GPSW06]



Attribute based Encryption

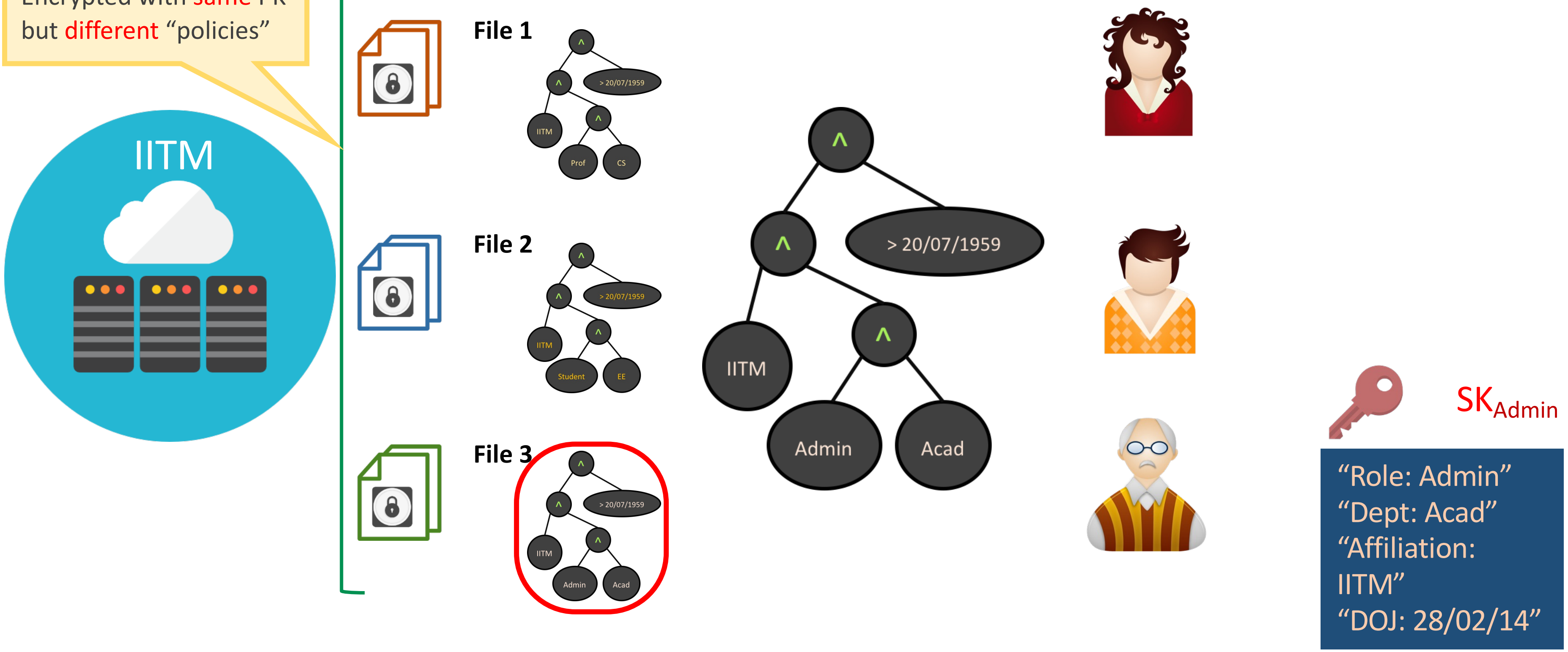
[SW05, GPSW06]



Attribute based Encryption

[SW05, GPSW06]

Encrypted with same PK but different "policies"



Attribute based Encryption

[SW05, GPSW06]