

COMP435: *SECURITY CONCEPTS!*

Lecture 5: Passwords, Biometrics

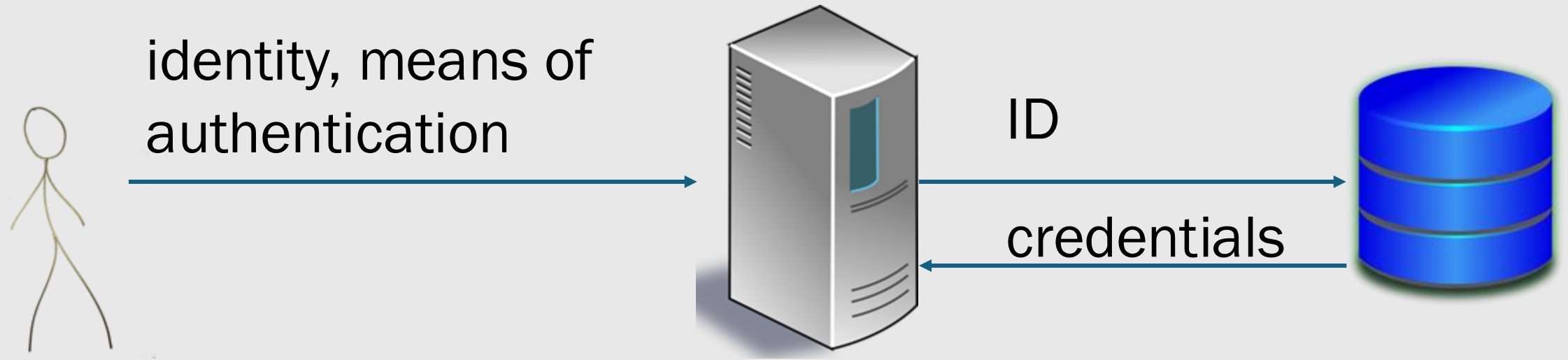
tinyurl.com/comp435-fa25



USER AUTHENTICATION... IN COMPUTER SYSTEMS!



Authentication



1. compute credentials
2. comparison

Means of Authentication

What you
know

What you do
or what you
are

What you
have

Where you
are



PASSWORDS



Means of Authentication

What you
know

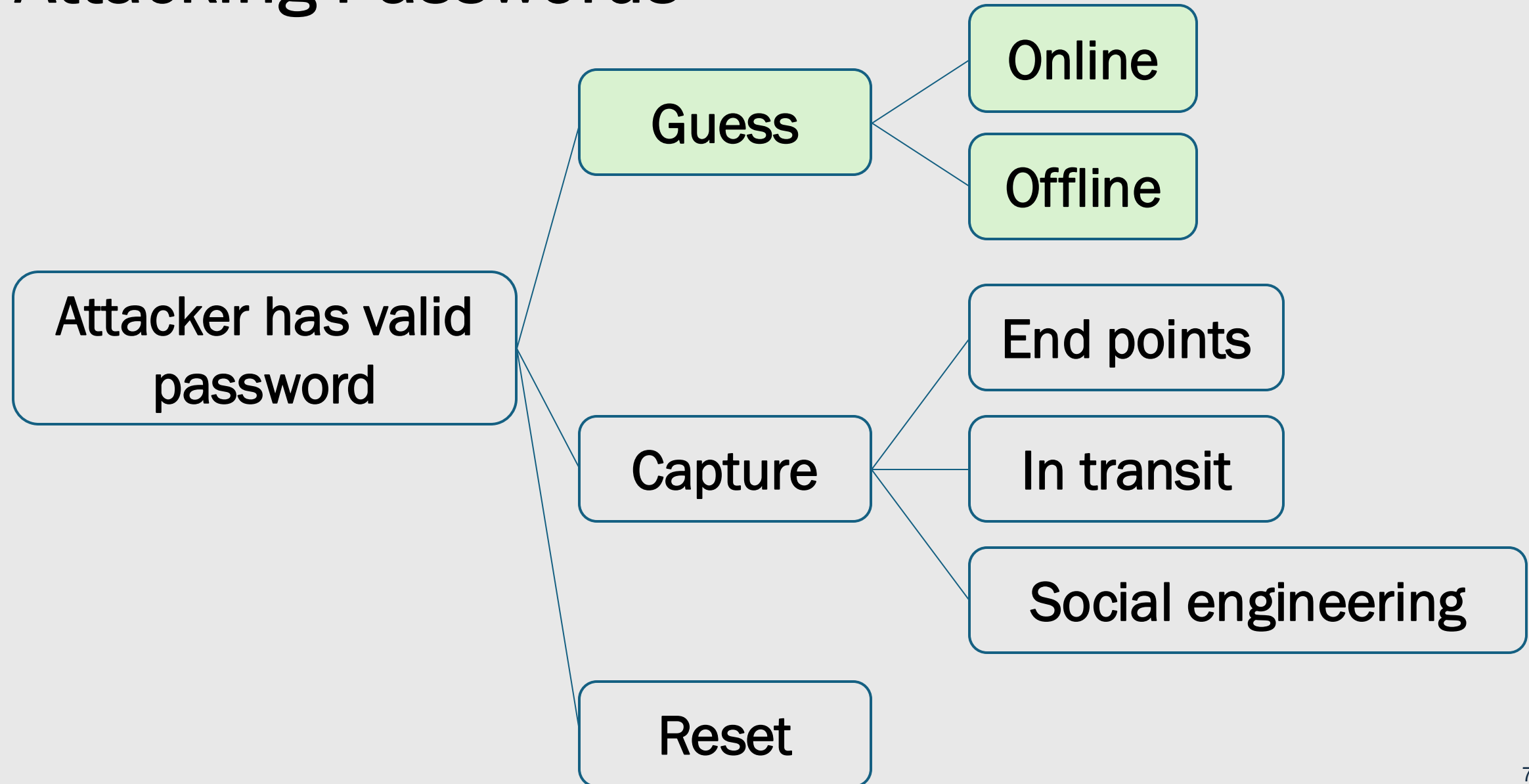
What you do
or what you
are

What you
have

Where you
are

passwords

Attacking Passwords



What is the attacker's chance of guessing correctly?

8-character passwords

printable characters (excluding space)

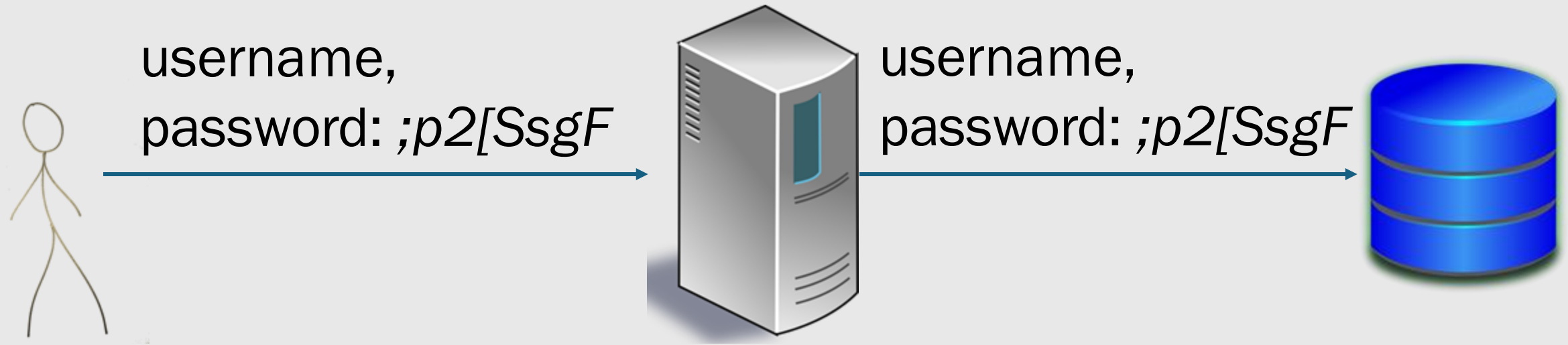
- *52 alphabetic*
- *10 numeric*
- *32 add'l (excluding space)*

WS4 Q1

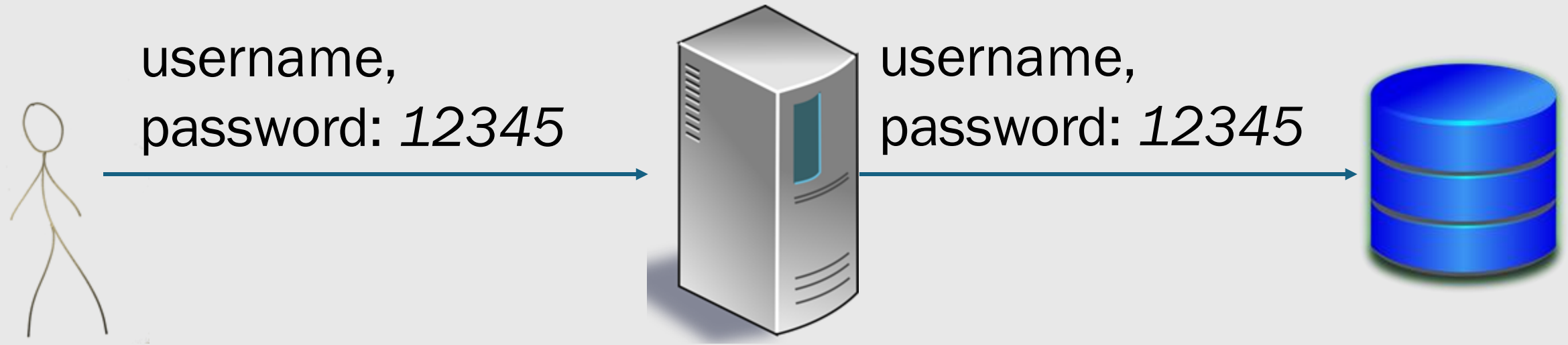
If there are 2^{53} possible passwords, how long will it take, on average, for the attacker to guess the right one assuming the attacker can try **10,000 passwords per second**

WS4 Q2

Registration



Registration



Common Passwords

- “qwerty”
- “12345678”
- “password”
- “11111111”
- family names
- pet names
- words in the dictionary
- common number substitutions

Common Passwords

- “qwerty”
- “12345678”
- “password”
- “11111111”
- family names
- pet names
- words in the dictionary
- common number substitutions



learnable online,
small set

Common Passwords

- “qwerty”
- “12345678”
- “password”
- “11111111”
- family names
- pet names
- words in the dictionary
- common number substitutions



600,000 English
words

Common Passwords

- “qwerty”
- “12345678”
- “password”
- “11111111”
- family names
- pet names
- words in the dictionary
- common number substitutions



e:3, o:0, t:7,
l:1, for:4



ONLINE GUESSING ATTACKS



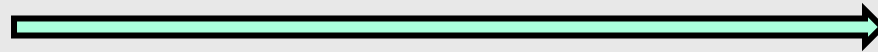
Attacker



Resource



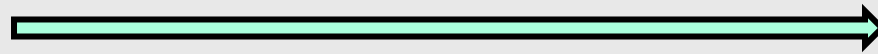
alice, 1234



access denied



alice, qwerty



Targeted Attack

Def'n: an attack aimed at a specific user

Targeted Attack



guess common
passwords



Identity	Password
Jane	p2ssw0rd!
Patrick	qwerty
Philip	ilovefido
Roz	qwerty
Claire	aq3wm\$0To!4

[NOT VISIBLE TO ATTACKER]

Targeted Attack



guess common
passwords



use personal
information



Identity	Password
Jane	p2ssw0rd!
Patrick	qwerty
Philip	ilovefido
Roz	qwerty
Claire	aq3wm\$0To!4

[NOT VISIBLE TO ATTACKER]

Targeted Attack

guess common
passwords

use personal
information

Countermeasure:
rate-limit guesses

Trawling Attack

Def'n: an attack that tries to gain access to a system through any of its users.

Trawling Attack



guess common
passwords against
many users

*try “querty” until
we get a hit!*

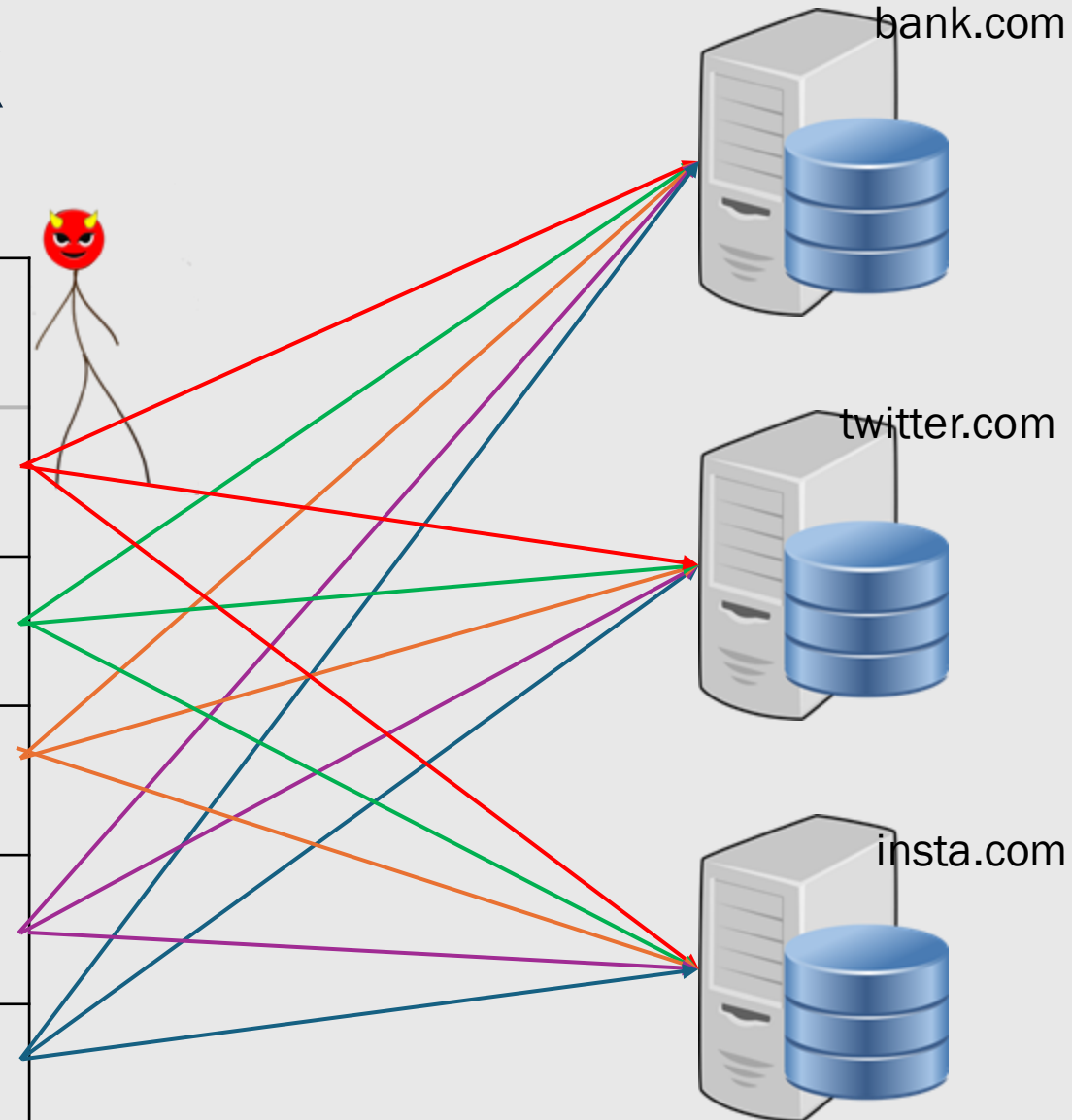
Identity	Password
Jane	p2ssw0rd!
Patrick	qwerty
Philip	ilovefido
Roz	qwerty
Claire	aq3wm\$0To!4



[NOT VISIBLE TO ATTACKER]

Credential Stuffing Attack

Identity	Password
Jane	p2ssw0rd!
Patrick	qwerty
Philip	ilovefido
Roz	qwerty
Claire	aq3wm\$oTo!4



[NOT VISIBLE TO ATTACKER]



OFFLINE GUESSING ATTACKS



Identity	Password
Jane	p2ssw0rd!
Patrick	qwerty
Philip	ilovefido
Roz	qwerty
Claire	aq3wm\$0To!4



Defense: Hashed Passwords

A **hashed value** is a fixed-length string produced by applying a **hash function** to input data.

Easy to compute, but hard to reverse.

Identity	h(Password)
Jane	0x23da09bf
Patrick	0x89bbc234
Philip	0x320cfd9e
Roz	0x89bbc234
Claire	0x2dec9a10



Defense: Hashed Passwords

A **hashed value** is a fixed-length string produced by applying a **hash function** to input data.

Easy to compute, but **hard to reverse**.

Weakness: Hash algorithms are deterministic → same inputs produces the same output.

Identity	h(Password)
Jane	0x23da09bf
Patrick	0x89bbc234
Philip	0x320cfd9e
Roz	0x89bbc234
Claire	0x2dec9a10



Pre-computed Dictionary Attack



Password	h>Password)
qwerty	0x89bbc234
12345678	0xcafe8900
password	0x424523d3
11111111	0xde334211
p2ssw0rd!	0x23da09bf



Identity	h>Password)
Jane	0x23da09bf
Patrick	0x89bbc234
Philip	0x320cfd9e
Roz	0x89bbc234
Claire	0x2dec9a10

Pre-computed Dictionary Attack



Password	h>Password)
qwerty	0x89bbc234
12345678	0xcafe8900
password	0x424523d3
11111111	0xde334211
p2ssw0rd!	0x23da09bf

Identity	h>Password)
Jane	0x23da09bf
Patrick	0x89bbc234
Philip	0x320cfd9e
Roz	0x89bbc234
Claire	0x2dec9a10

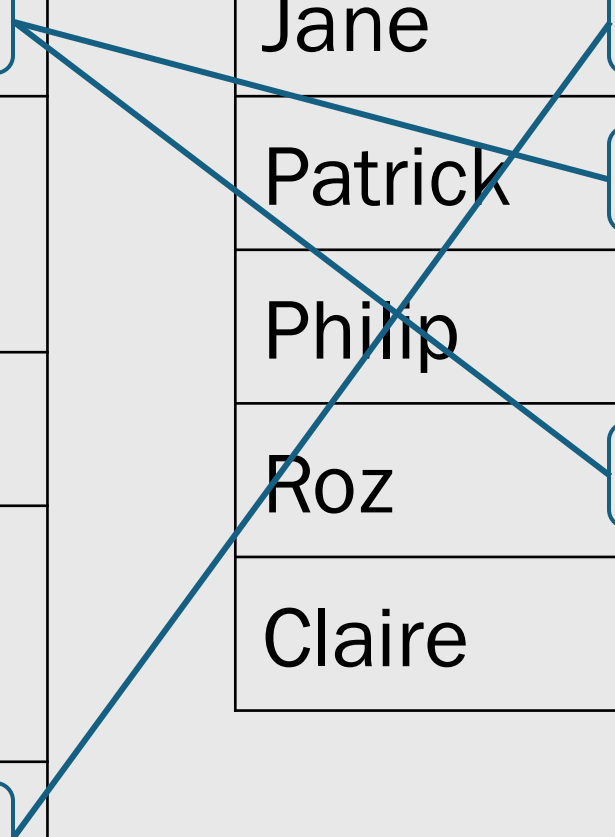


Pre-computed Dictionary Attack



Password	h>Password)
qwerty	0x89bbc234
12345678	0xcafe8900
password	0x424523d3
11111111	0xde334211
p2ssw0rd!	0x23da09bf

Identity	h>Password)
Jane	0x23da09bf
Patrick	0x89bbc234
Philip	0x320cfd9e
Roz	0x89bbc234
Claire	0x2dec9a10



Defense: Iterated Hashing

d iterations

Identity	$h(h(h(\dots(\text{Password}))))$
Jane	0x356323da
Patrick	0x094f89bb
Philip	0x987cca21
Roz	0x094f89bb
Claire	0x23c29585



Defense: Salting

Identity	Salt	h>Password,Salt)
Jane	0xdef	0x93762d21
Patrick	0x32c	0x2874efa2
Philip	0xfef	0x954eabbc
Roz	0xcaf	0x5609ab1c
Claire	0x835	0xfe3dc550



Password Requirements

- Chosen uniformly at random
- Easy to remember



BIOMETRICS



Means of Authentication

What you
know

What you do
or what you
are

What you
have

Where you
are

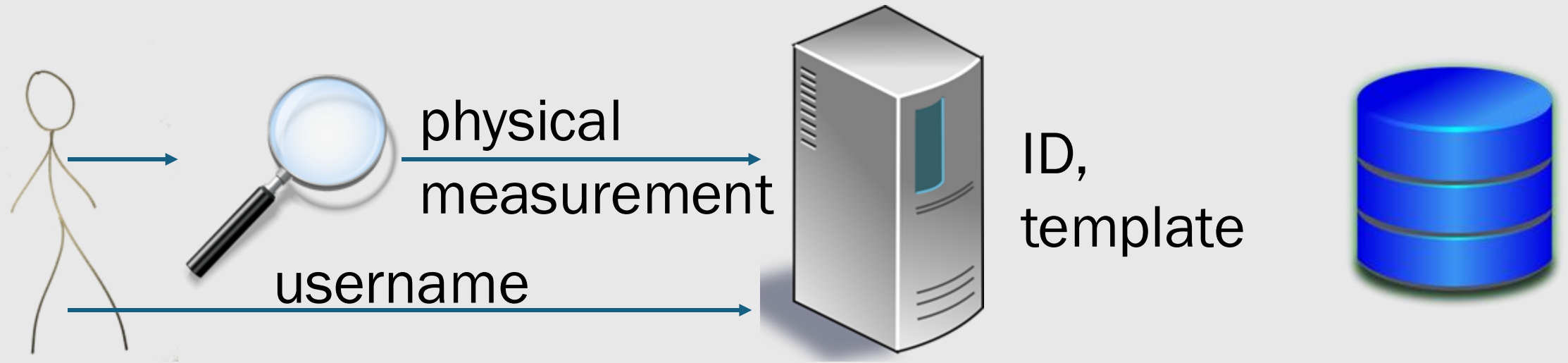
biometrics

Biometrics

Def'n: Measurement of a physical characteristic

E.g., fingerprints, hand geometry, retina measurement, iris scan, voice, facial features

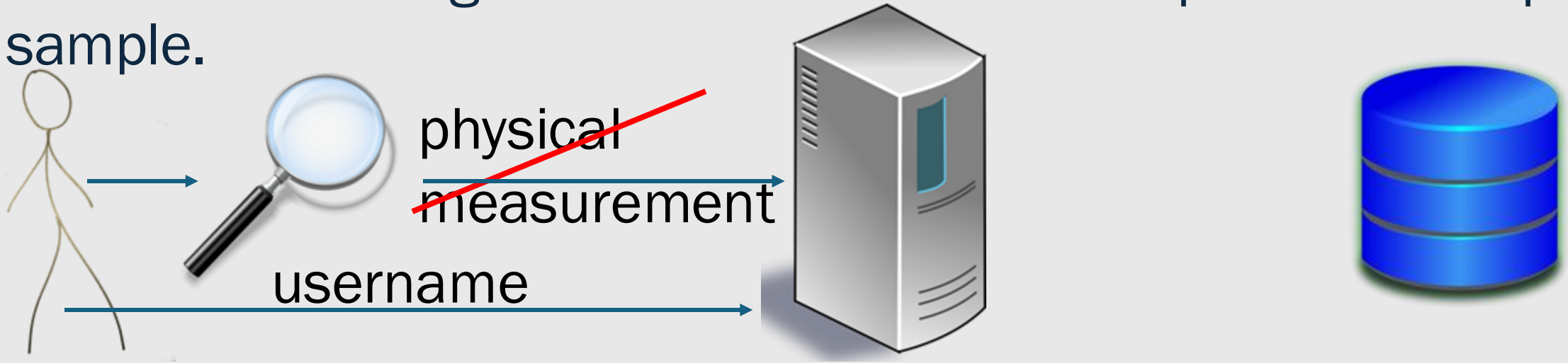
Registration



1. digitization
2. feature (template) extraction

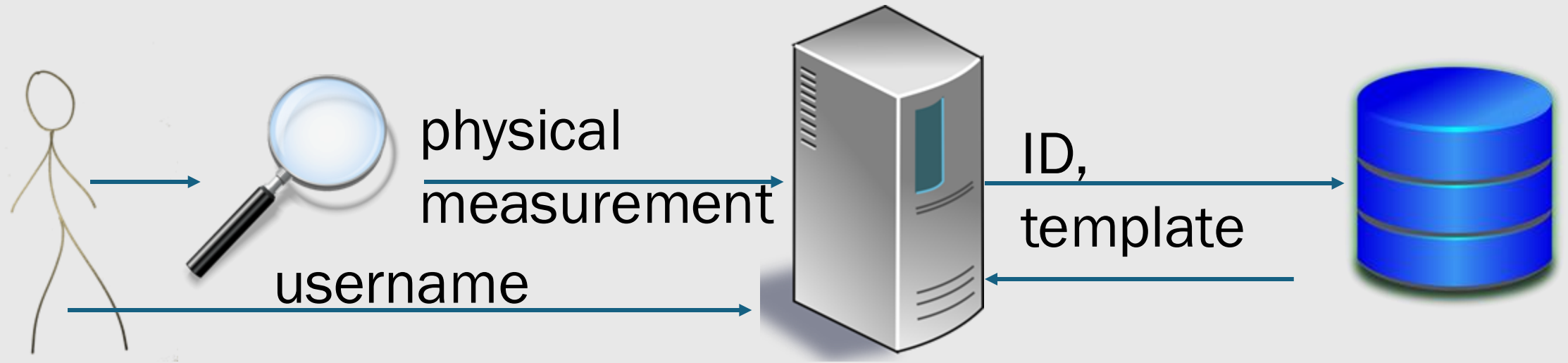
Failure to Enroll (FTE)

Def'n: The sensing mechanism is unable to capture an adequate sample.



1. ~~digitization~~
2. ~~feature (template)~~
extraction

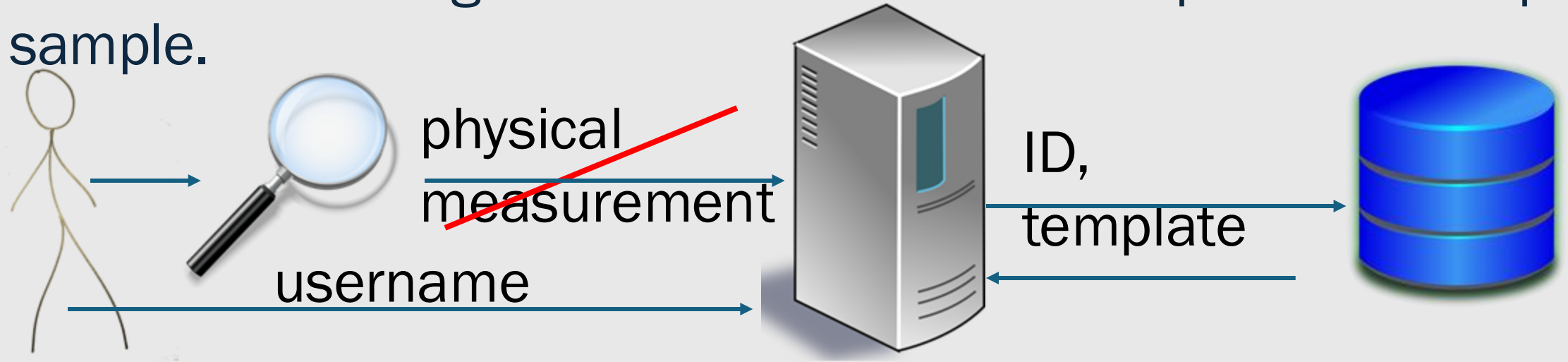
Authentication



1. digitization
2. feature (template) extraction
3. comparison

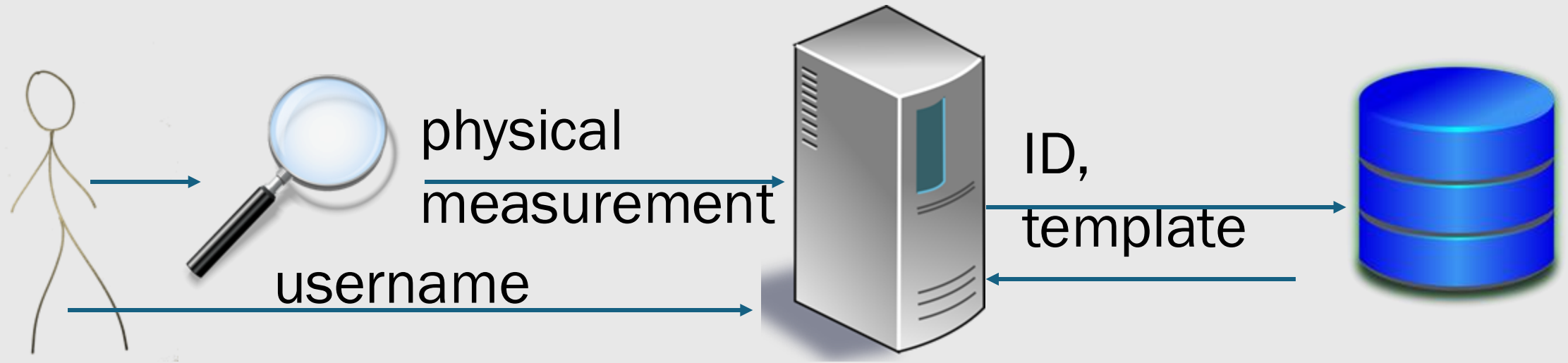
Failure to Capture (FTC)

Def'n: The sensing mechanism is unable to capture an adequate sample.



1. ~~digitization~~
2. ~~feature (template)~~
extraction
3. comparison

Authentication



1. digitization
2. feature (template) extraction
3. comparison

Authentication via Biometrics

	Is the identified user	Is not the identified user
Considered a match	True Positive	False Positive
Considered not a match	False Negative	True Negative

Biometrics + Templates

- **Reference template:** feature-based, digital representation of a person's biometric trait

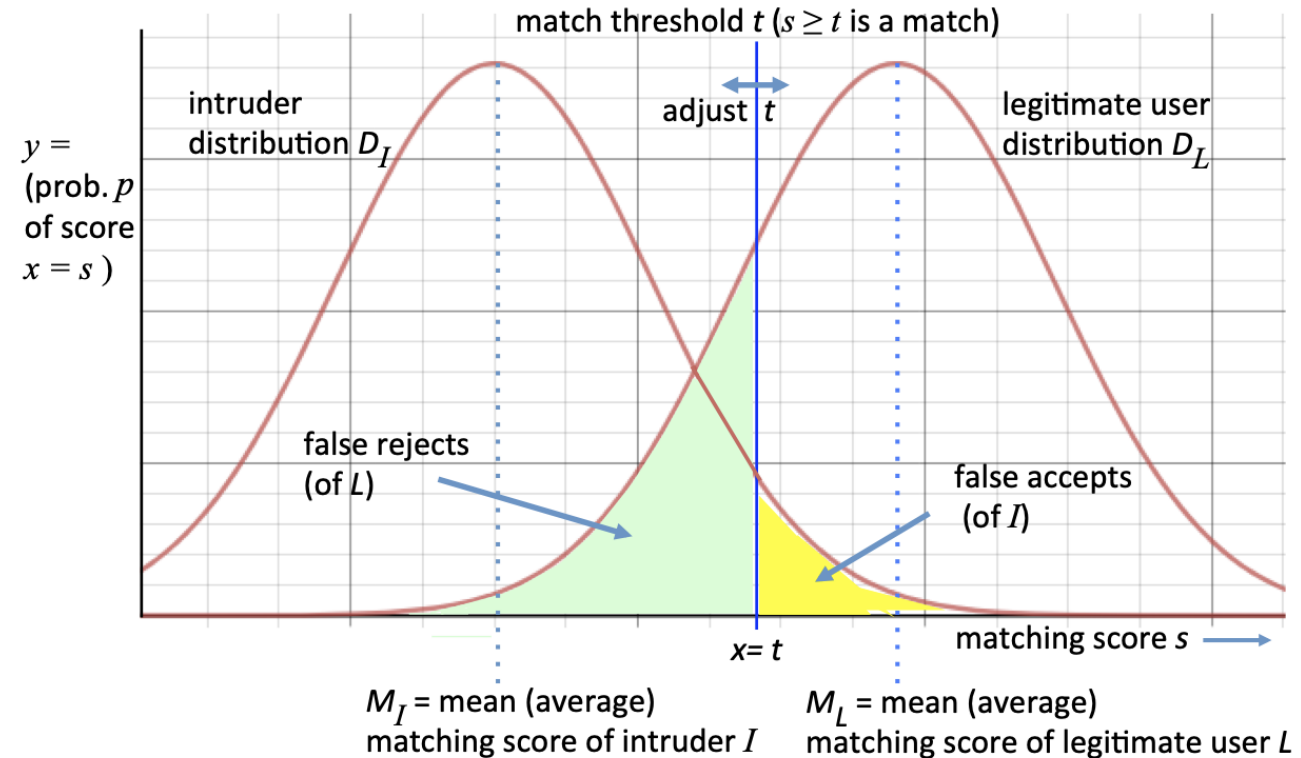
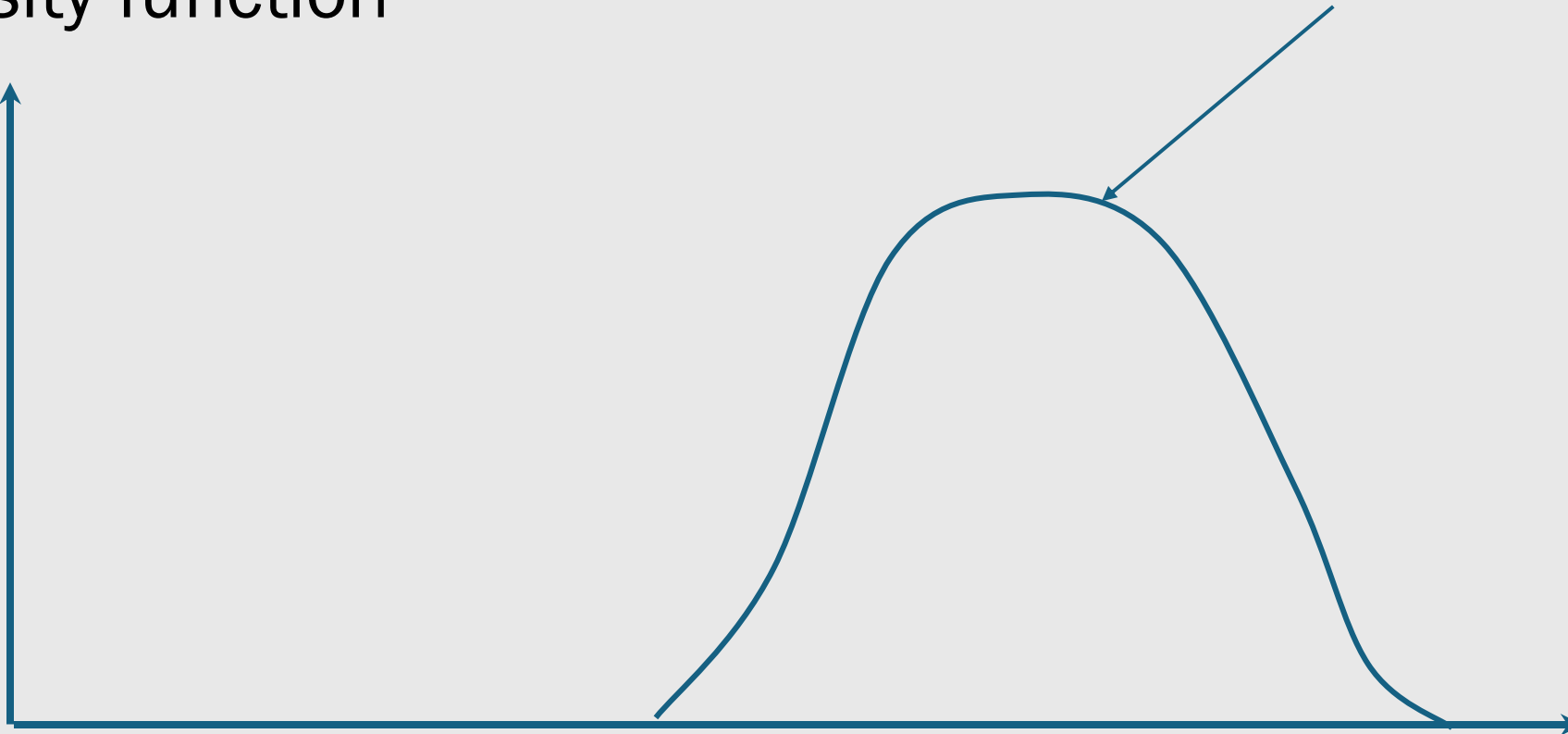


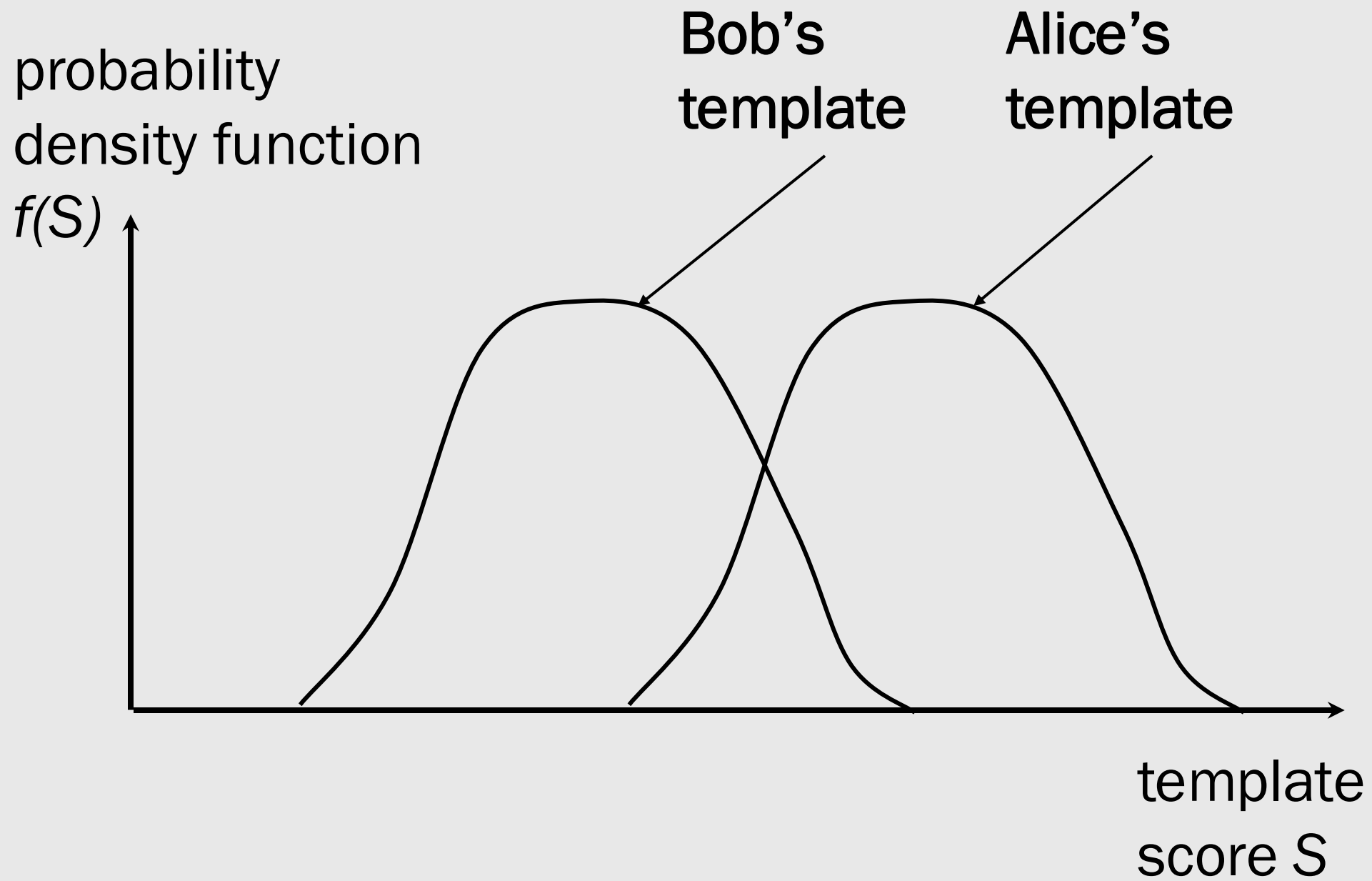
Figure 3.6: Biometric system tradeoffs. Curves model probability distributions for an intruder and legitimate user's matching scores; higher scores match the user's biometric template better. The y axis reflects how many biometric samples get matching score $x = s$.

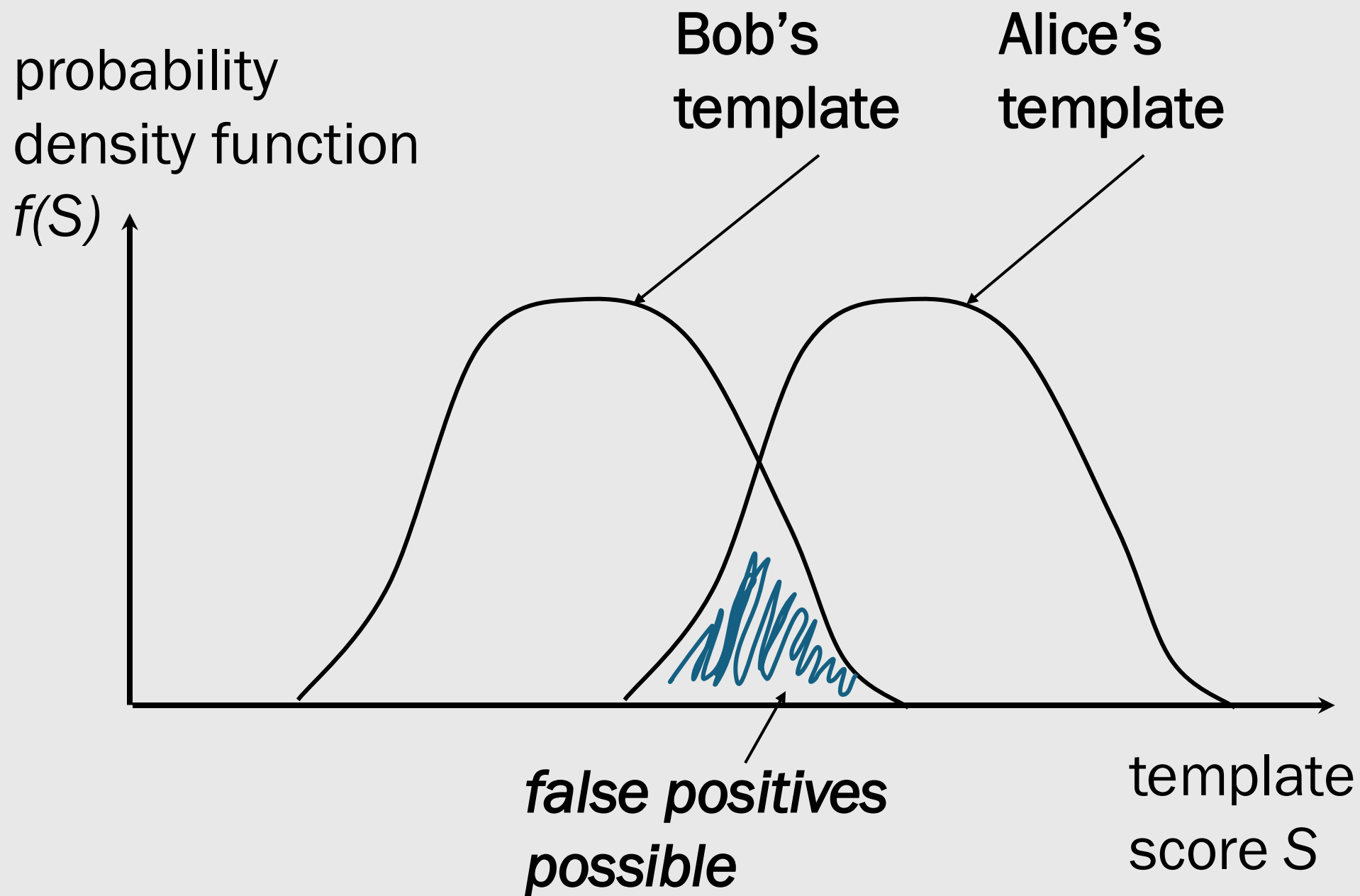
probability
density function
 $f(S)$

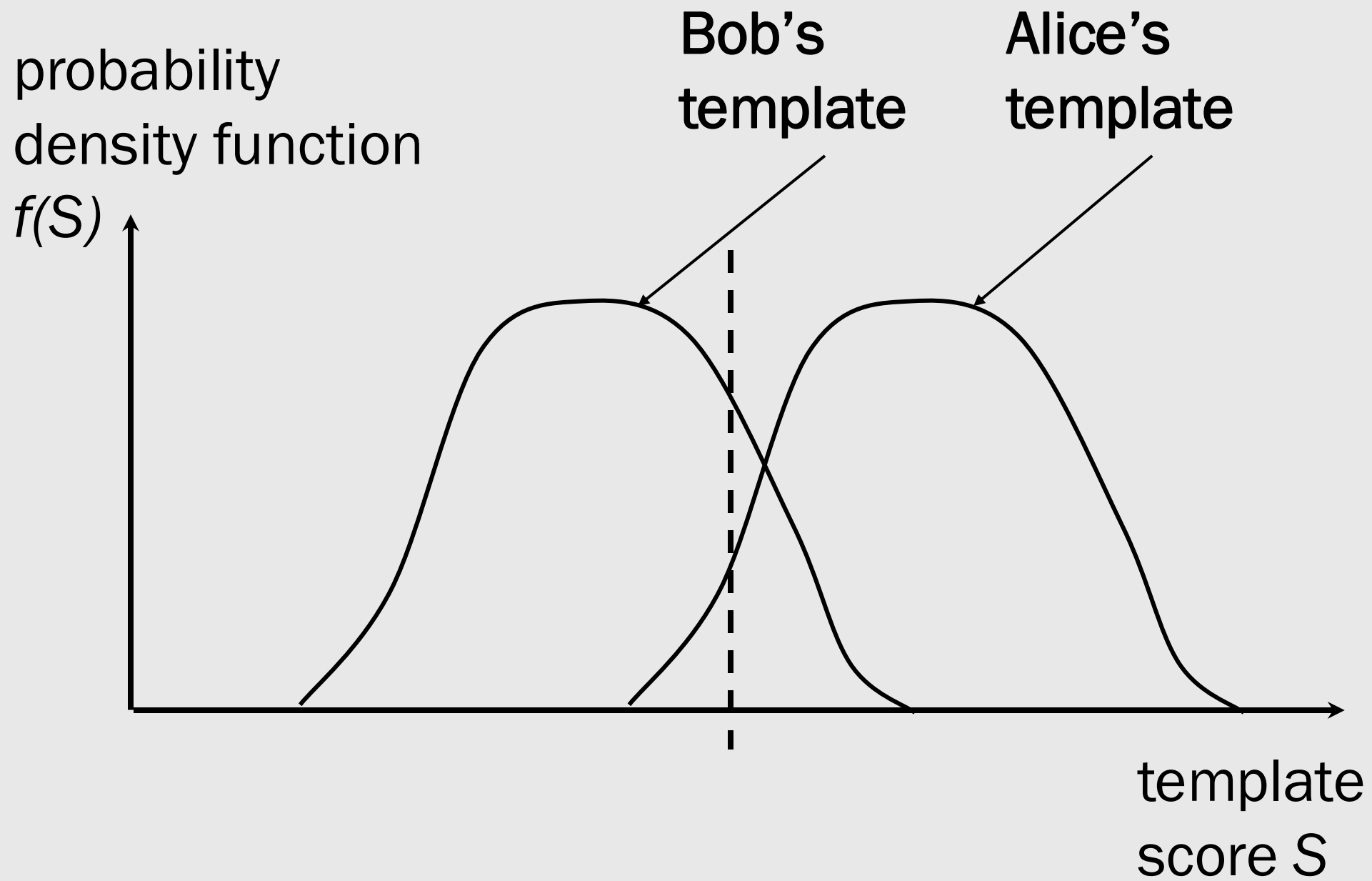
Alice's
template

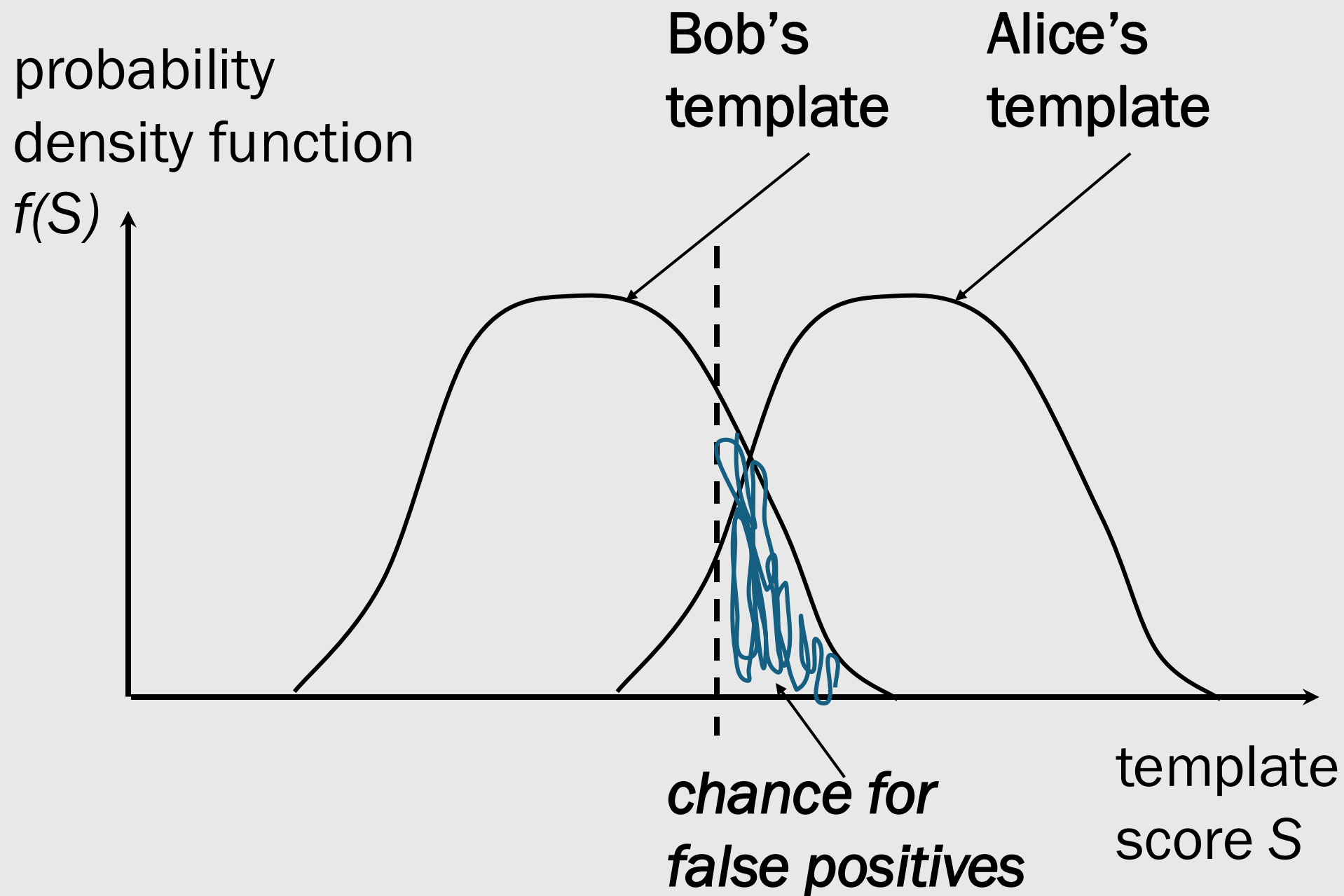


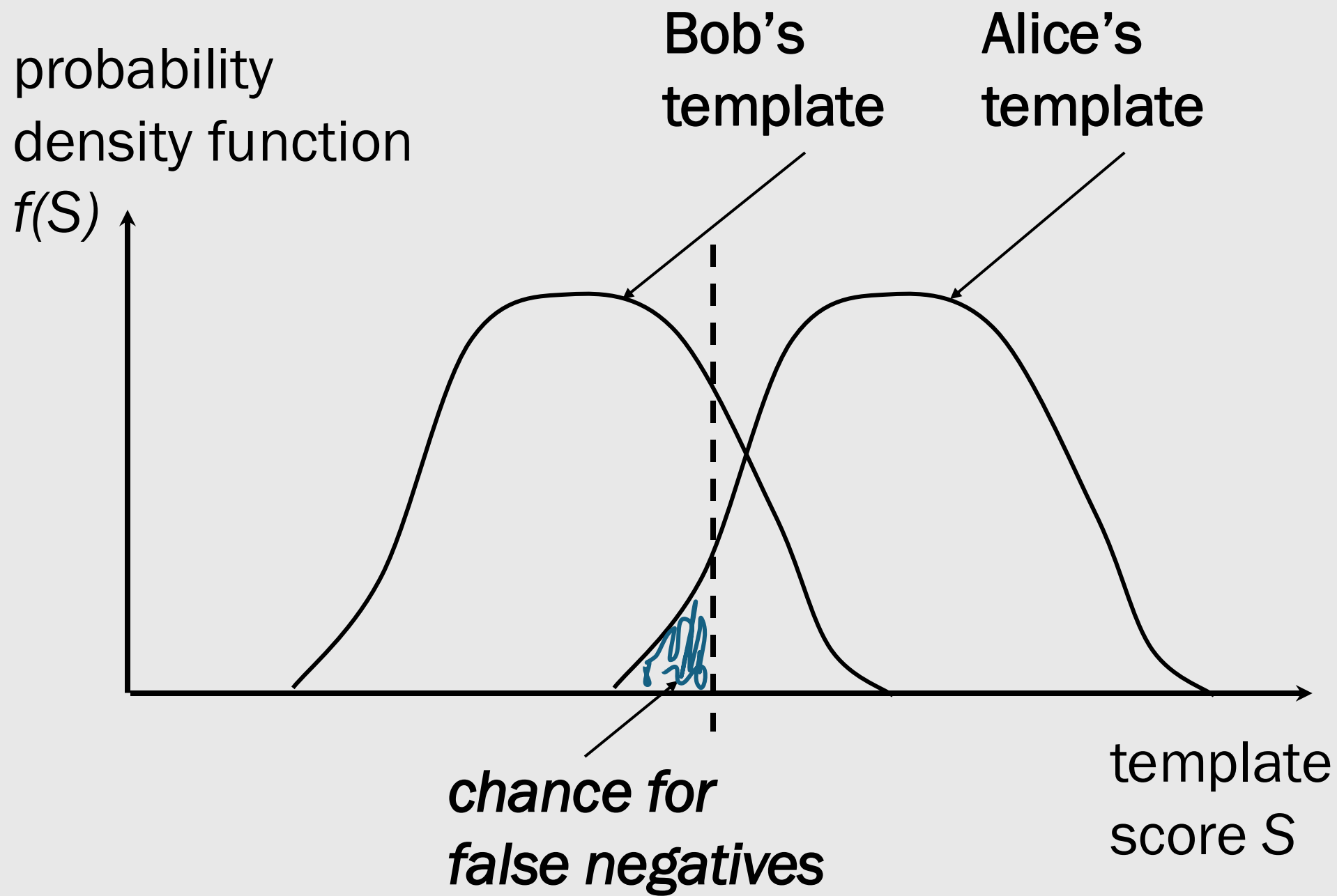
template
score S











Usability of Biometrics



- low cognitive load
- nothing extra to carry
- scales well for users

Pros



- failure to enroll
- failure to capture
- false negatives
- privacy concerns

Cons

Security of Biometrics for User Authentication



- physical characteristics are unique

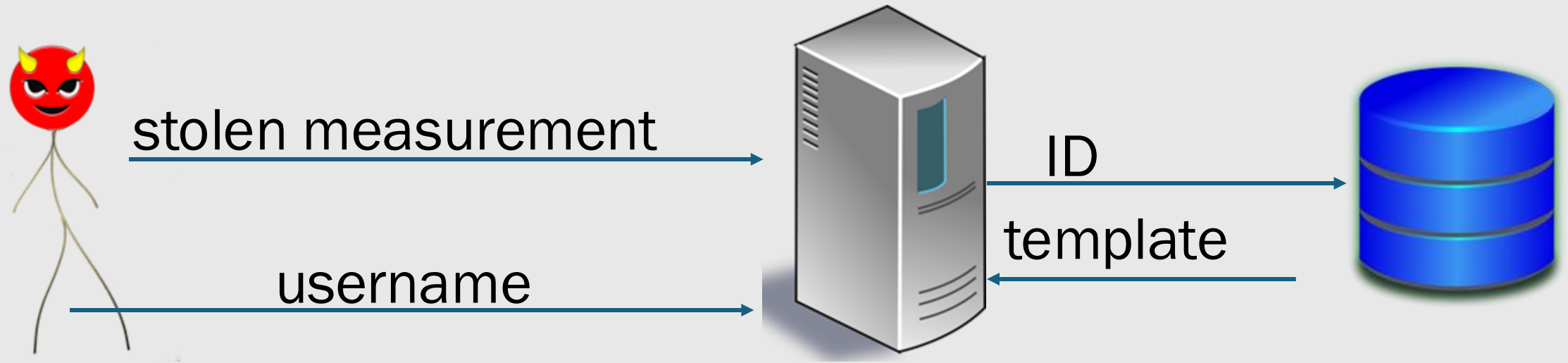
Pros



- insecure in remote or unsupervised setting

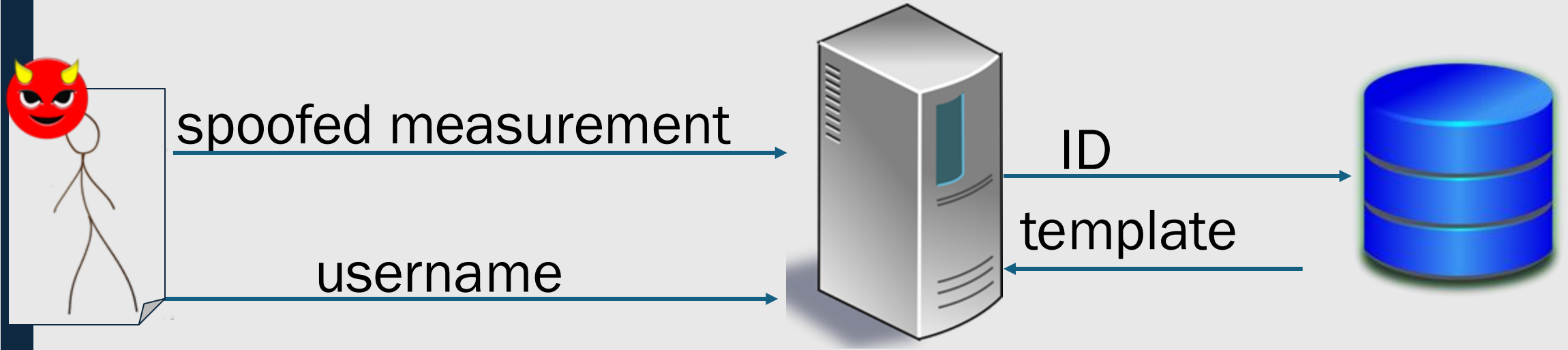
Cons

Remote Authentication



1. feature extraction
2. comparison

Unsupervised Authentication



1. digitization
2. feature extraction
3. comparison

Security of Biometrics for User Authentication



- physical characteristics are unique

Pros



- insecure in remote or unsupervised setting
- cannot be revoked
- false positives

Cons

Summary: Biometrics



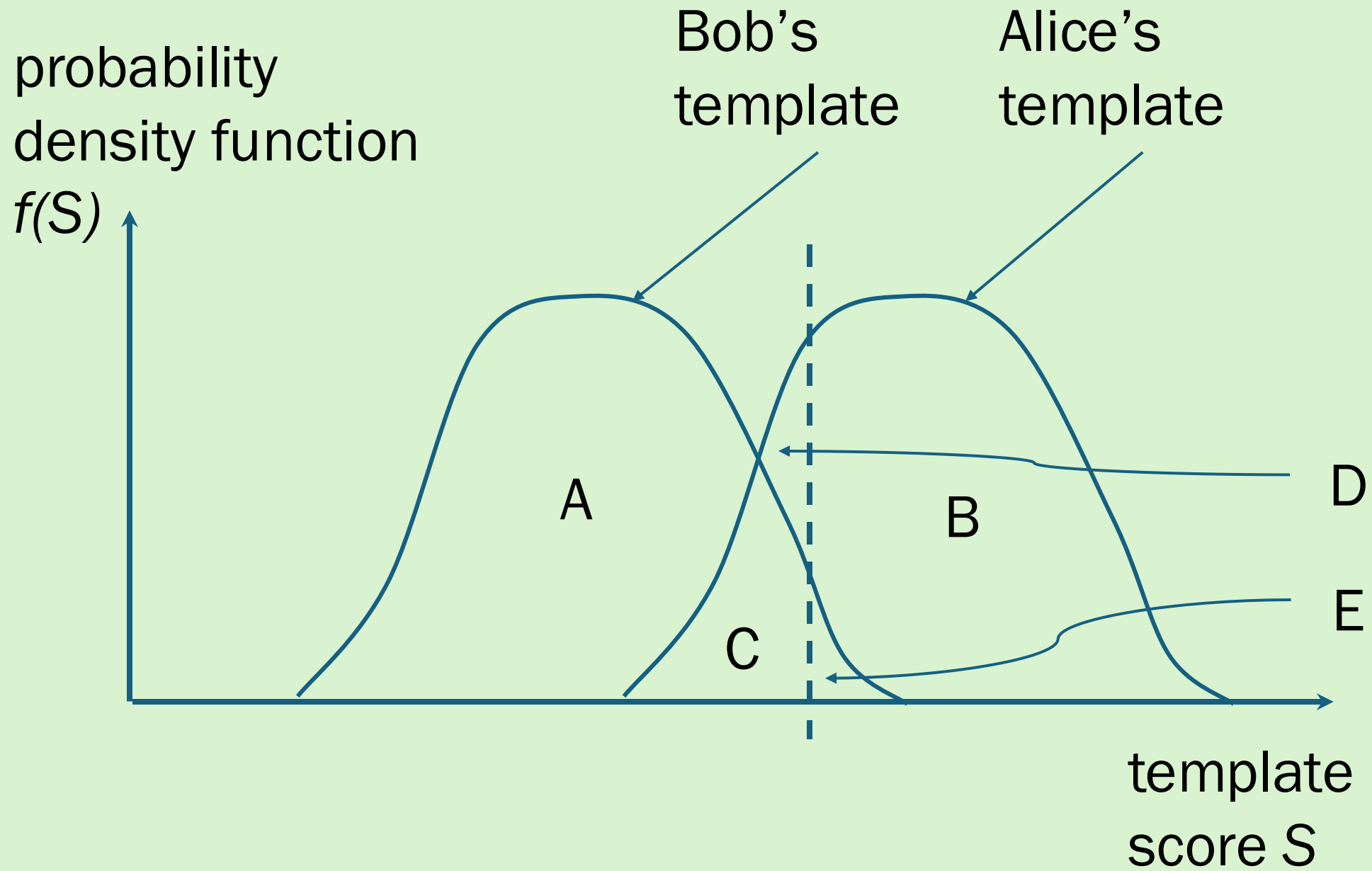
- Low cognitive burden
- Easy to use

Pros



- Security is lower than expected
- Privacy concerns

Cons

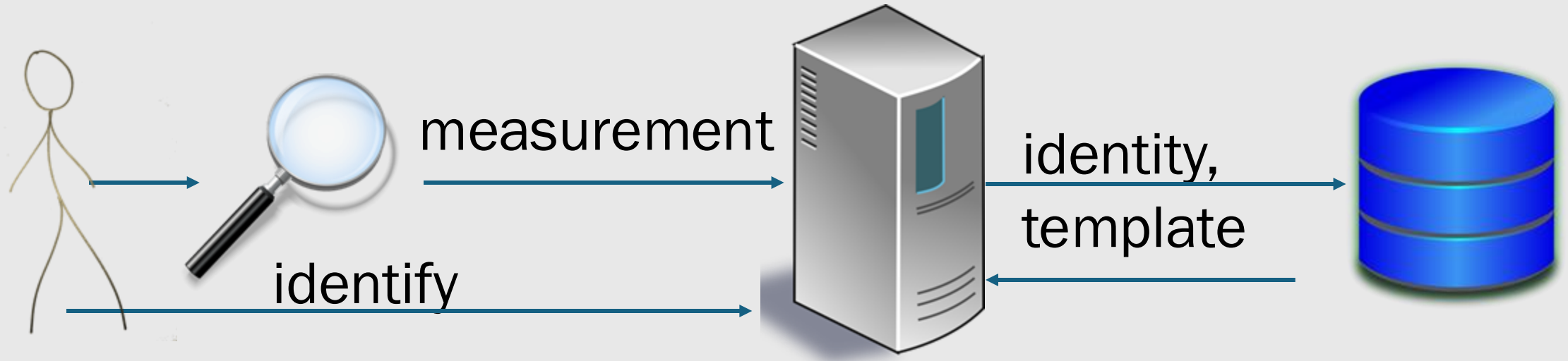




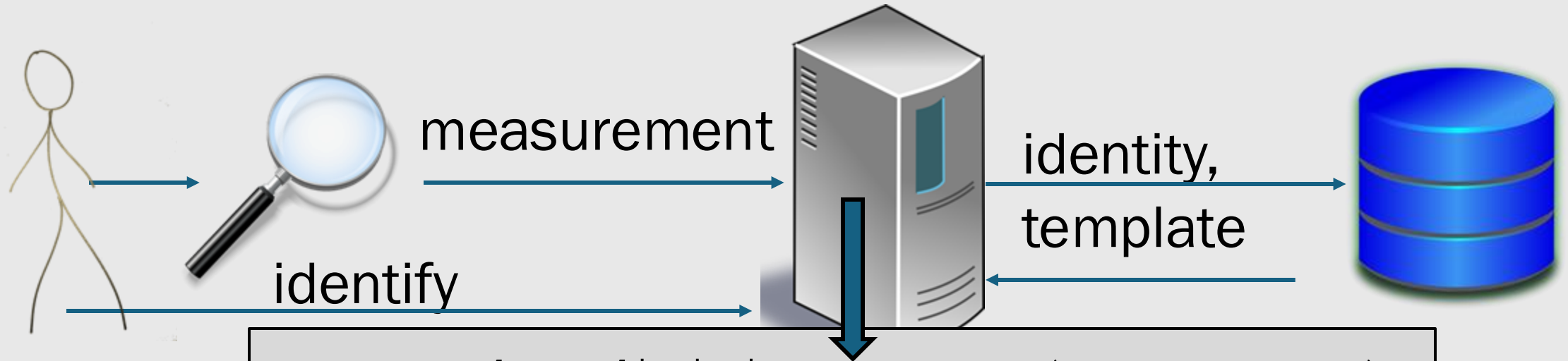
BIOMETRICS: AUTHENTICATION VS. IDENTIFICATION



Authentication using Biometrics



Authentication using Biometrics

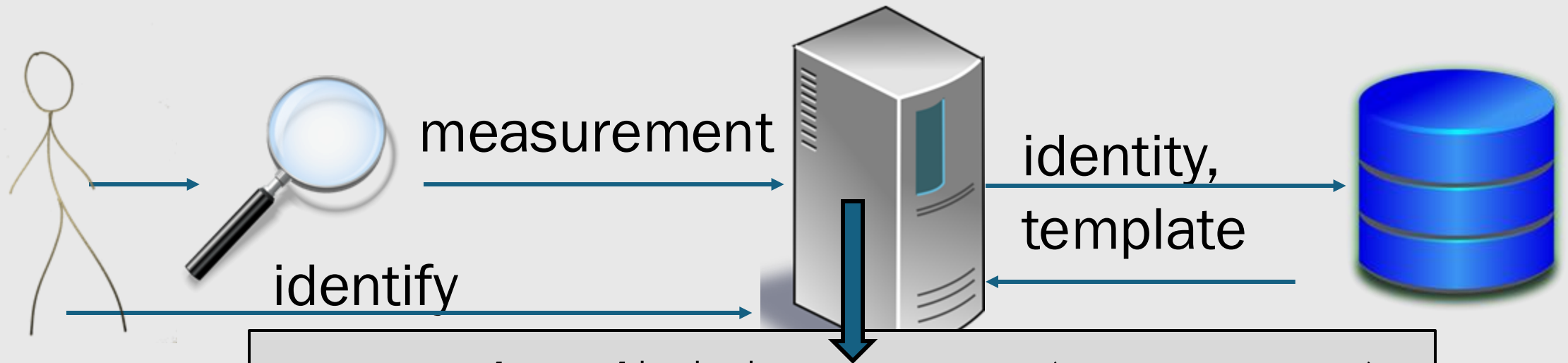


```
1. t_auth = digitize_extract(measurement)
2. t_stored = query_db(identity)

3. if is_match(t_auth, t_stored):
4.     allow_access()

5. else:
6.     deny_access()
```

Identification & Authentication (one user)

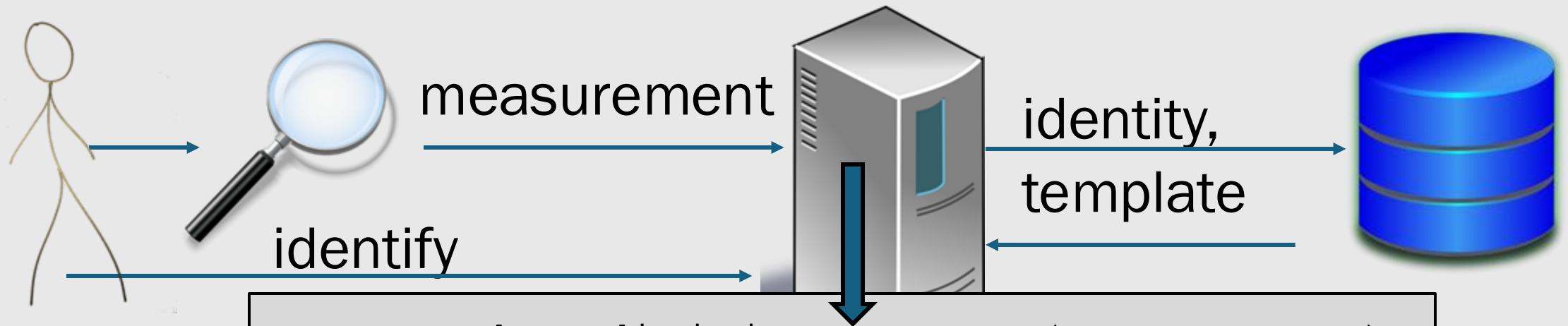


```
1. t_auth = digitize_extract(measurement)
2. t_stored = query_db()

3. if is_match(t_auth, t_stored):
4.     allow_access()

5. else:
6.     deny_access()
```

Identification & Authentication (multiple users)

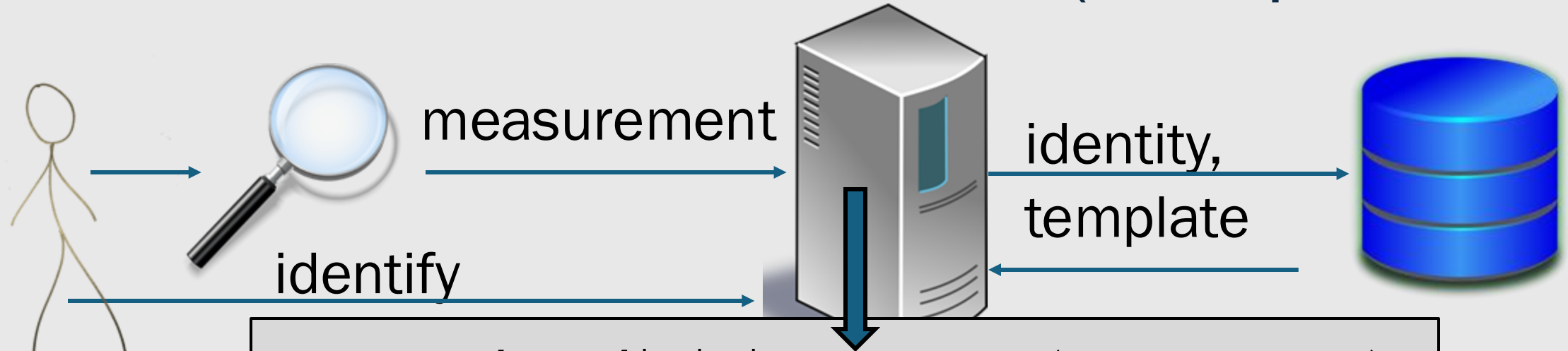


```
1. t_auth = digitize_extract(measurement)
2. t_stored_list = query_db()

3. for t_stored in t_stored_list:
4.     if is_match(t_auth, t_stored):
5.         allow_access()
6.         return

7. deny_access()
```

Identification & Authentication (multiple users)



```
1. t_auth = digitize_extract(measurement)
```

```
2. t_stored_list = query_db()
```

```
3. for t_stored in t_stored_list:
```

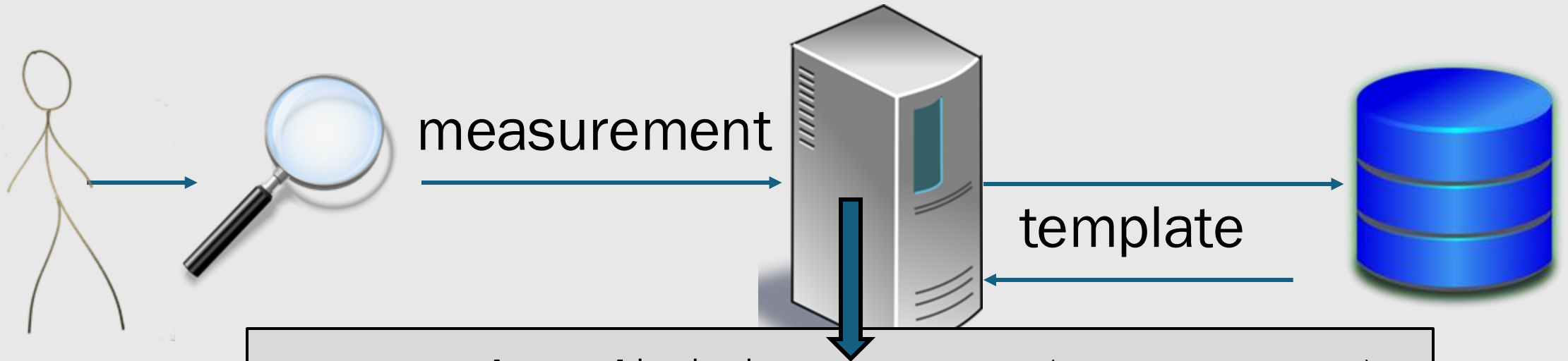
```
4.     if is_match(t_auth, t_stored):
```

```
5.         allow_access()
```

```
6.         return
```

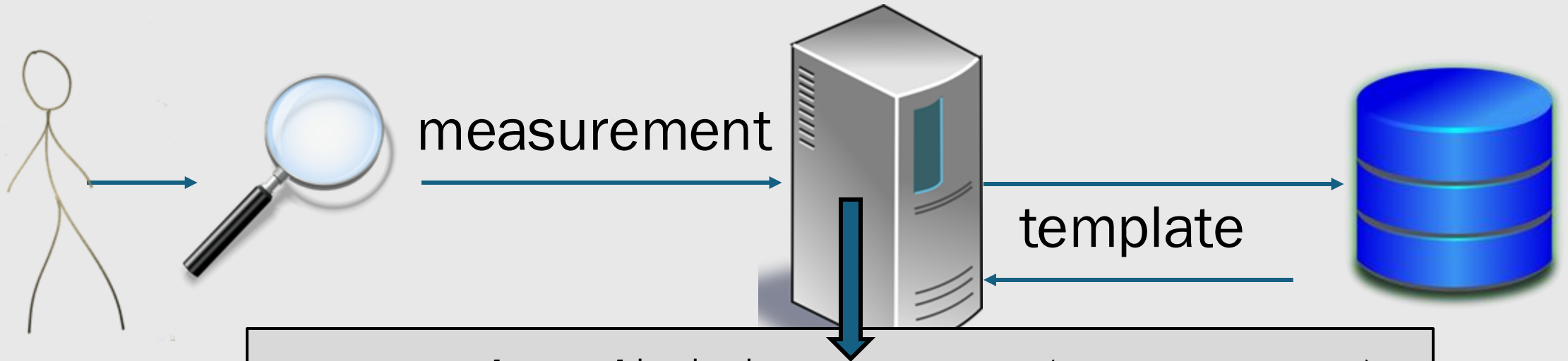
```
7. deny_access()
```

Identification



```
1. t_auth = digitize_extract(measurement)
2. t_stored_list = query_db()
3. matches = set()
4. for t_stored, id in t_stored_list:
5.     if is_match(t_auth, t_stored):
6.         matches.add(id)
7. return matches
```

Identification



```
1. t_auth = digitize_extract(measurement)
2. t_stored_list = query_db()
3. matches = set()

4. for t_stored, id in t_stored_list:
5.     if is_match(t_auth, t_stored):
6.         matches.add(id)

7. return matches
```