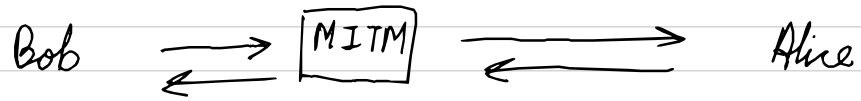
(Q1) Explain MITM attack

→ Attacker secretly relays & alters the communication between to parties



examples → i) IP spoof (ARP spoof)
ii) DNS spoof
iii) Session hijack
iv) Page in Middle

(Q2) Explain types of Harm

→ B I M F

B Block → attack on availability eg DOS
I Intercept → Attack on confidentiality
M Modify → Attack on Integrity
F Fabricate → Attack on availability eg fake mail

(Q3) Explain CIA Triad

→

C → Confidentiality
I → Integrity
A → Availability

(Q4) Explain Method Objective Motive

→ Deny any one & attack wont occur
eg Bank robery

Method → open lock

Objective → gain access to bank vault

Motive → Get rich / revenge on bank?

(Q5) Method of defense

→

i) Prevent attack → i) Remove motive

ii) Deter attack → Make attack Harder

iii) Deflect attack → Make another target more attractive

iv) Mitigate attack → Make impact less severe

eg access rights

v) Detect attack → Raise alarm

vi) Recovery → Recover from damage

P D D | M | D R

Before attack | During attack | after attack

Q6 Explain Buffer overflow / stack smashing

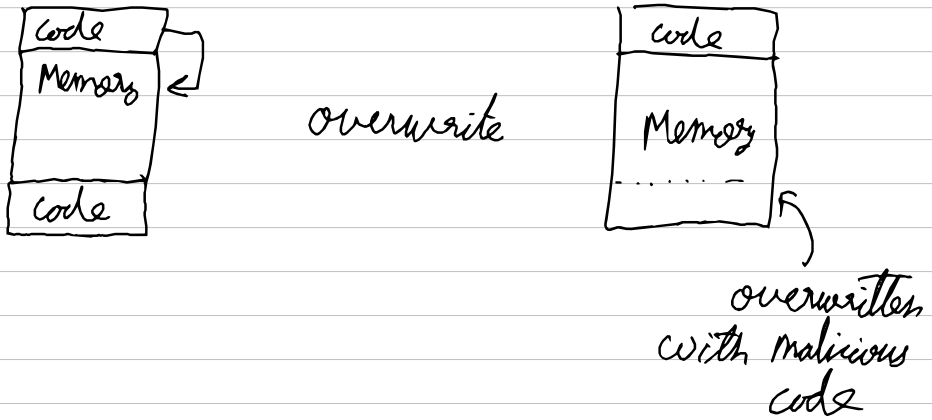
→ i) data is written beyond space allocated to it

ii) overides memory reserved for executable code

iii) Simplest form → code fails → DOS

iv) found in old programming languages

v) C → gets (char)
strcpy (A, B)
etc
} unsafe routines



vi) Testing may not identify

(Q7) How to prevent overflow / overflow countermeasures

→

i) check length of variable eg int > 2^{128}

ii) Confirm array subscripts are in limits

eg
```
a = int[10]
b = a[10]
```

iii) Double check boundary conditions

```
a = input( )
if (a ≤ 10) ←
  b = c[9]
```

iv) limit input to no. of acceptable characters

v) Limit program privileges

vi) Code analyzers

vii) Testing & Pentesting

viii) Use safe procedures

(Q8) Explain Incomplete Mediation

→ i) when validation is not proper eg client side validation

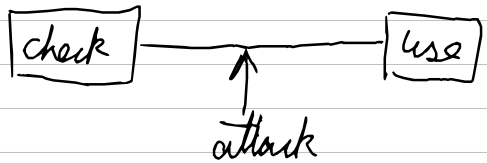
ii) eg changing price of product

iii) solution → Validate in server side

(Q9) Explain 0 day exploit in context of TOCTTOU

→ 0 day exploit → vulnerability is unknown to developers

TOCTTOU → Time of check to time of use
→ data is checked correctly but malicious activity is performed after it is checked
→ exploits time between check & use.



→ actions must be atomic to prevent TOCTTOU

(Q10) Explain Non malicious programming errors.

→ i) Buffer overflow
ii) Incomplete mediation
iii) TOCTTOU
iv) Unterminated null terminated string
  ↳ string ends by \00
  ↳ someone overides \00
  ↳ system continues to read everything till next \00 found
v) Access point off by one
  ↳ $< n$ or $\leq h$
  ↳ fails at boundary condition
vii) Undocumented access point → exploit backdoor entries (cheat codes)

Q11) Types of Malwares

Virus	Propagates copies of itself to others
Trojan	Piggybacks on a useful service
Worm	Propatages through network
Rabbit	Replicates itself to exaust resources
Logic Bomb	Triggers action when a condition occurs
Time bomb	Triggers action at a time
spyware	spies on user eg keylogger (attack on confidentiality)
adware	display ads (Objective → show ads)
ransomware	encrypts data & demands ransom (Availability)

(Q12) Malware detection techniques

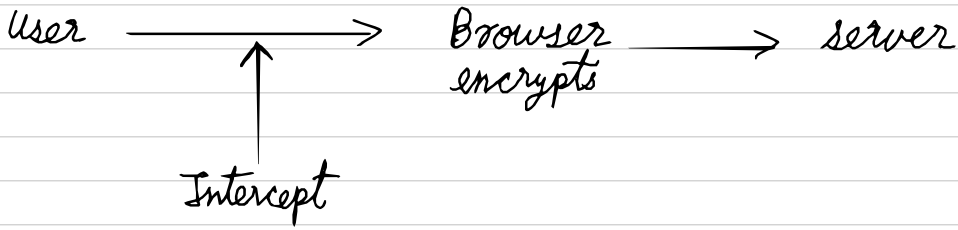
→ 1) Signature detection
- Match against code DB
- Not very accurate as code can be updated
- useful for wellknown malware

2) Change detection
- If file has changed, it may be infected
- Compare hash values
- Many false positives

3) Anamoly detection
- Monitor for unusual virus like behaviours
- Unusual network activity
- Unusual file access

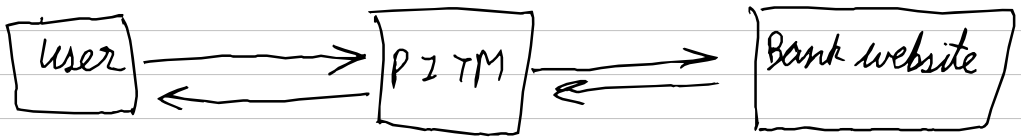
(913) Types of browser attacks

→ i) Man in the browser



ii) Keylogger → i) web based keylogger
ii) inject malicious JS in a legitimate website & capture keystrokes

iii) Page in the middle → i) User directed to a different page & credentials stolen
ii) eg pshing websites



iv) Program download substitution → i) Install malware when user wants to install real app
ii) Piggyback Trojan

v) User in the middle → i) Make user solve captcha on the bots behalf