Mohammad lotf Al-Hussam

Part 1:-

IoT Device Data Encryption simulation..

steps:

1- Generate random temperature and readings.
2- Encrypt the data using AES or a lightweight cipher.
3- simulate transmission to a server and decrypt on the receiver side.
4- Display the data before and after encryption.

code: (2)

python

```python
from Crypto.Cipher import AES
import os
import base64
import random


def pad(data):
    return data + (16 - len(data) % 16) * chr(16 - len(data) % 16)


def encrypt(plain_text, key):
    cipher = AES.new(key, AES.MODE_CBC, iv=b'1234567890123456')
    return base64.b64encode(cipher.encrypt(pad(plain_text).encode())).decode()


def decrypt(encoded_text, key):
    cipher = AES.new(key, AES.MODE_CBC, iv=b'1234567890123456')
    return cipher.decrypt(base64.b64decode(encoded_text)).rstrip().decode()


# Generating random temperature and humidity data
temperature = random.randint(20, 30)
humidity = random.randint(30, 60)
```

```
# Data to encrypt
data = f"Temperature: {temperature}°C, Humidity: {humidity}%"
print("Original data:", data)

# Key for AES
key = os.urandom(16)

# Encryption
encrypted_data = encrypt(data, key)
print("Encrypted data:", encrypted_data)

# Decryption
decrypted_data = decrypt(encrypted_data, key)
print("Decrypted data:", decrypted_data)
```

Python

plaintext

```
Original data: Temperature: 21°C, Humidity: 43%

Traceback (most recent call last):
  File "/tmp/tmp2wa1jb0s.py", line 33, in <module>
    encrypted_data = encrypt(data, key)
                     ^^^^^^^^^^^^^^^^^^
  File "/tmp/tmp2wa1jb0s.py", line 13, in encrypt
    return base64.b64encode(cipher.encrypt(pad(plain_text).encode())).decode()
                            ^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^
  File "/app/venv/lib/python3.11/site-packages/Crypto/Cipher/_mode_cbc.py", line 183, in encryp
    raise ValueError("Data must be padded to %d byte boundary in CBC mode" % self.block_size)
ValueError: Data must be padded to 16 byte boundary in CBC mode
```

عنوان الدرس _____ اليوم _____ التاريخ / /

Expected output:

- original data, encrypted data and decrypted data will be displayed.

Part II :-

IoT Device lifecycle simulation.

Steps:

1- Create a Python script to simulate the five security lifecycle stages.
2- log each step with timestamps and messages.

code: (1) صورة كود

(1) صورة الناتج

الواجب :

python

```python
import time

def log_security_lifecycle(step):
    print(f"[{time.strftime('%Y-%m-%d %H:%M:%S')}] {step}...")

log_security_lifecycle("Stage 1: Threat model created")
log_security_lifecycle("Stage 2: Secure boot verified")
log_security_lifecycle("Stage 3: Keys injected securely")
log_security_lifecycle("Stage 4: OTA update verified")
log_security_lifecycle("Stage 5: Device decommissioned, secrets wiped")
```

▶ Run

Python

plaintext

```
[2025-11-30 18:10:28] Stage 1: Threat model created...
[2025-11-30 18:10:28] Stage 2: Secure boot verified...
[2025-11-30 18:10:28] Stage 3: Keys injected securely...
[2025-11-30 18:10:28] Stage 4: OTA update verified...
[2025-11-30 18:10:28] Stage 5: Device decommissioned, secrets wiped...
```

9:10 PM

Expected output:

- A console log showing lifecycle events, with timestamps.

Part III:-

(optional) secure Device Boot verification:-

steps:

1- store a hashed firmware signature in Python.

2- verify the hash before executing the main loop.

code:

Expected output:

- A message indicating whether the firmware verification succeeded or failed.

python

```python
import hashlib

# Storing firmware signature
firmware_signature = hashlib.sha256(b'SampleFirmware').hexdigest()

def verify_signature(signature):
    if signature == firmware_signature:
        print("Firmware verified. Booting...")
    else:
        print("Firmware verification failed. Boot aborted.")

# Simulating the boot process
verify_signature(hashlib.sha256(b'SampleFirmware').hexdigest())
```

Run

Python

plaintext

```
Firmware verified. Booting...
```

عنوان الدرس _____ اليوم _____ التاريخ / /

Questions:—

let's answer the questions manually.

1-The Internet of Things (IoT) refers to a network of interconnected physical devices, sensors, and machines that communicate and exchange data over the internt without human intervention, enabling applications in domains like healthcare and smart cities.
Examples include RfID tags of inventory tracking and wearable sensors for health monitoring.

2-A cyber-physical system (CPS) integrates computational algorithms physical processes, where embedded systems monitor and control real-world entities through feedback loops, often in industrial or infrastructural settings.

الواجب:

عنوان الدرس _____ اليوم _____ التاريخ / /

- CPS extends beyond IoT by emphasizing tight coupling between cyber and Physical components for real-time operations.

3- IoT focuses on device interconnectivity and data exchange for everyday applications, often with heterogeneous, resource-constrained nodes.
while CPS emphasizes embedded control of physical systems with high reliability and real-time constraints, such as in smart grids or autonomous vehicles.
IoT is a subset of CPS, but CPS requires stricter determinism and safety.

4- Botnets are networks of compromised devices controlled by attackers to perform coordinated malicious activities, such as DDoS attacks.
A well-known example is the mirai botnet, which infected unsecured IoT devices to

الواجب: _____

launch massive DDoS attacks, taking down websites in 2016.

5- secure boot is a process where an embedded device verifies the integrity and authenticity of its firmware or OS during statup using cryptographic hashes or signatures, preventing execution of tampered code.
This establishes a chain of trust from hardware root.

6-
*Design/Threat modeling:
example: Assessing botnet vulnerabilities in sensor ntworks.
*Development/secure Boot:
example: Hash verification during initialization
*Deployment/key management:
example: ~~system~~ symmetric key distribution for encryption.

7

عنوان الدرس _____ اليوم _____ التاريخ / /

*operation/ oTA updates:
example: Remote updates to fix exploits.
*Decommissioning:
example: key deletion to prevent data leaks post-use.

7- main vulnerabilities include resource constraints leading to weak encryption unsecured protocols, and physical access risks, exposing devices to eavesdropping, DoS, and fabrication attacks
Examples:-
1- mirai botnet exploited default credentials on cameras ofr DDos.
2- jeep Hack 2015 allowed remote vehicle control via cellular vulnerabilities

8- oTA (over-The-Air) updates refer to remote delivery and installation of firmware or software patches to IoT devices, ensuring security fixes without physical access, but requiring signature verification to prevent malicious injections.

الواجب:

9- lightweight cryptography encompasses efficient algorithms designed for resource-constrained devices, optimizing for low energy, memory, and computation while maintaining security against attacks. It is essential in IoT due to devices' limitations (e.g., tiny sensors with <32kB RAM), where conventional methods like AES drain resources and hinder performance.

10- mirai scans for vulnerable IoT devices (e.g., via telnet with weak credentials), infects them to form botnets for DDos, exploiting unsecured firmware and protocols. Prevention includes secure boot, regular OTA updates, lightweight encryption for communications, and network segmentation.

11- Secure firmware updates use cryptographic fixing vulnerabilities without compromising integrity.

عنوان الدرس _____ اليوم _____ التاريخ / /

Hardware root of trust provides a tamper-resistant foundation (e.g., secure elements) for boot and key storage enabling chain-of-trust in constrained devices.

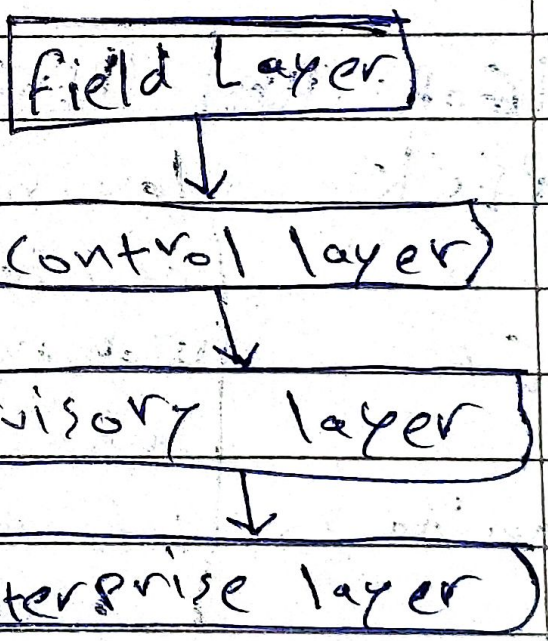
12- PKI manges digital certificates and keys for authentication, encryption, and integrity in IoT, variants are needed for constraints

13- Edge computing processes data near the source (e.g., gateways) rather than central clouds, reducing latency and exposure.
In IoT security, it enhances privacy by local encryption and threat detection, minimizing data transmission risk.

14- ICS/SCADA

الواجب: _____

architecture includes field devices (sensors/actuators), PLCs, RTUs, historians, and HMIs connected via networks for monitoring/control.
vulnerabilities: unsecured protocols (e.g., modbus), legacy systems open to remote exploits, and physical access risk.



15- Threat modeling systematically identifies, assesses, and prioritizes risks (e.g., using STRIDE) during design to build security in. In IoT, it applies by mapping devices layers (Physical to cloud).

to anticipat attacks like botnets, informing lightweight crypto choices.

16- TEE is isolated hardware/software environment protecting sensitive operations (e.g, key handling) from the main os.
It enhances embedded security by ensuring confidentiality and integrity in IoT resisting side-channel attacks on constraind devices.

17- Blockchain provides decentralized, immutable ledgers for secure data sharing, device authentication.

18- MPC allows devices to jointly compute function on private data without revealing inputs, using cryptographic protocols.

19- lightweight cryptography includes block ciphers like SPECK, Simon, LEA (low RAM/energy) and stream ciphers like chacha20-Poly1305, optimized for IoT's constraints.

ALGORITHM	STRENGTHS [SOURCES]	SUITABILITY
SPECK	low cycles, hardware-efficient.	sensors/RFID.
LEA	High throughput energy-saving.	constrained IoT.
chacha20	software-optimized, secure.	mobile devies.

20- The Mirai botnet (2016) scanned unsecured IoT devices (e.g., cameras) for weak passwords, infecting millions to launch record DDoS attacks, disrupting internet services.

الواجب: _____

21-

a/ How can you ensure data authenticity between nodes and the central server?

Use digital signatures or message authentication codes (MACs).

b/ which cryptographic techniques would you choose and why?

lightweight symmetric ciphers like SPECK for efficiency on sensors, combined with PKI for initial key exchange.

22-

a/ Scan for open ports, exploit weak authentication in PLCs, or inject malware via unsecured protocols to manipulate controls (e.g., Stuxnet-like).

b/ Implement firewalls, intrusion detection (e.g., snort rules).

عنوان الدرس _____ اليوم _____ التاريخ / /

23-

a/ Eavesdropping on wifi, DOS via botnet infection, and unauthorized access altering controls.

b/

1]-confidentiality: End-to-end lightweight encryption (e.g., SPECK)

2]-Integrity: firmware signatures and hash checks.

3]-Availability: Rate limiting and secure OTA updates to patch DOS vectors.

24-

ASPECT	Embedded system security	TRADITIONAL COMPUTER
resource constraints	lightweight crypto minimal overhead	Full-strength algorithms (e.g. AES-256)
Attack surface	physical access, real time failurer	primarily network/ software exploits
lifecycle	secure boot, OTA from design.	Patch management post
Examples	IoT sensors with TEE	Enterprise/PKI

Embedded security prioritizes efficiency and hardware trust due to constraints, unlike traditional focus on scalability.

25-challenges includ balancing security with ultra-low resources (e.g., <128kB flash).

- DeepContext trends show ongoing research ofr hybrid approaches [1-6].
- master summary table of key findings.
- This synthesis reveals consensus on lightweight cryptography's role in mitigating IoT risks, with dedates on standardization.
- for further exploration, consult sources on emerging standards like SUIT for updates.