SPA-API Module Secure Communication Architecture Review

1. Encryption of Data in Transit

The data is encrypted using AES-GCM which is the industry standard when it comes to symmetric encryption. The key is transmitted using RSA. Thus hybrid encryption is implemented to ensure encryption of data in transit.

What is Hybrid Encryption?
It is a form of data encryption that utilizes a combination of symmetric and asymmetric encryption:

- 256 bit AES-GCM is implemented for encrypting the actual message payloads, thus providing confidentiality and integrity.
- RSA is used for securely exchanging the AES session key between client and server.

2. Integrity Protection

Message integrity is ensured in the following manner:

- Each encrypted payload includes a SHA-256 hash of the message along with a timestamp and nonce.
- The backend validates the hash after decryption and rejects messages with mismatched hashes.
- Timestamp and nonce mechanisms are in place to defend against tampering and message replay.

3. Replay Protection

Protection against replay attacks is implemented in the following manner:

- Messages contain a unique nonce and timestamp.
- The backend keeps a cache of recent nonces and timestamps with expiry controls.
- If a nonce is reused or the timestamp is outside the acceptable window, the message is rejected.

4. Session Key Management

A new session key is generated by the client with every new message and transmitted using RSA to the server.

- The server generates new RSA key pairs for each session thus strengthening security even further.
- The client generates new AES session keys per communication session which prevents reuse of any key.
- AES keys are never sent in plaintext, only encrypted using RSA thus preventing people from accessing it during transmission.

5. Transport Security

The project doesn’t use any protocols or certificates during message transmission.

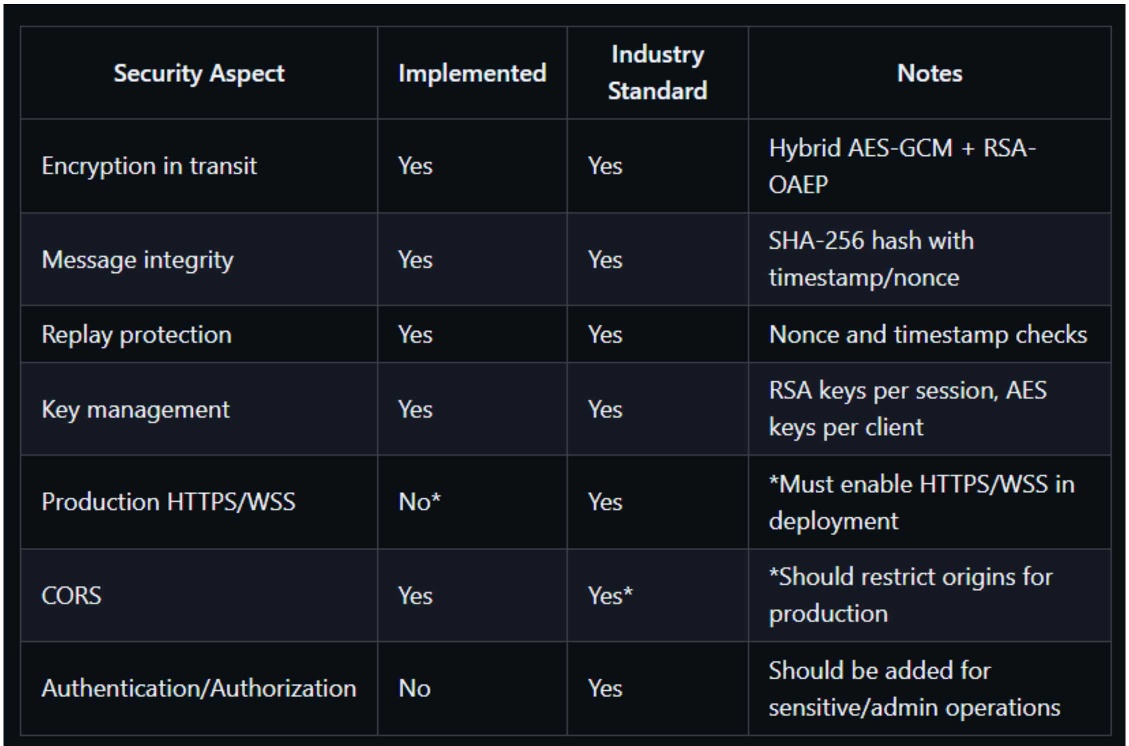
- The project currently uses:
  - http:// and ws:// (WebSocket over plaintext)
- Industry best practice mandates using https:// and wss:// in production to prevent man-in-the-middle (MITM) attacks.

6. Conclusion

The core encryption and integrity mechanisms in this system follow the best modern cryptographic practices. However, to ensure full security in a production environment, the following improvements may be implemented:

- Restrict CORS to trusted origins only.
- Enforce HTTPS/WSS for all endpoints.
- Implement authentication and authorization.
- Add rate limiting, input validation, and security logging for a complete security posture.

Summary Table



Security Aspect	Implemented	Industry Standard	Notes
Encryption in transit	Yes	Yes	Hybrid AES-GCM + RSA-OAEP
Message integrity	Yes	Yes	SHA-256 hash with timestamp/nonce
Replay protection	Yes	Yes	Nonce and timestamp checks
Key management	Yes	Yes	RSA keys per session, AES keys per client
Production HTTPS/WSS	No*	Yes	*Must enable HTTPS/WSS in deployment
CORS	Yes	Yes*	*Should restrict origins for production
Authentication/Authorization	No	Yes	Should be added for sensitive/admin operations