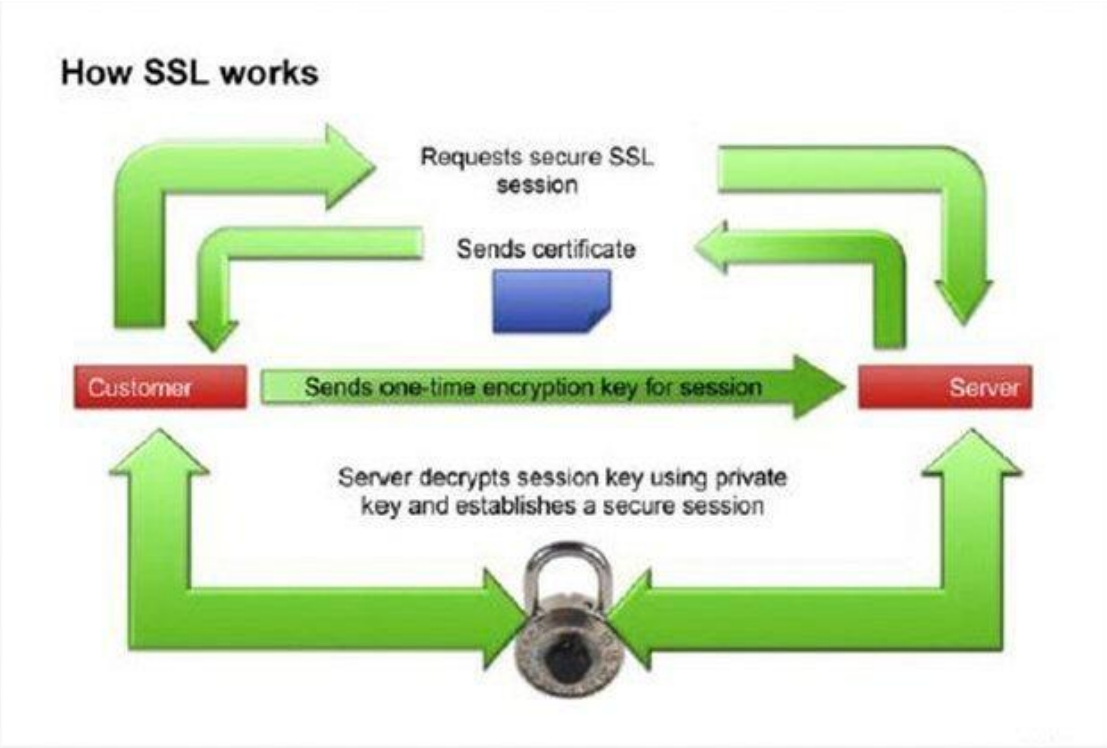
Vatsal Sanchala
Jagjit Singh
Pargat Singh Dhanjal
Vishrut Deshmukh

1. **Design a secure system/protocol to set the question paper in a secure manner.**

1. All authorized faculty members will have their **individual account** on the portal.
2. **IP based Authentication & Authorization** of individual users/teachers managed by the admin user who can authenticate and authorized individual users based on their position and allow/deny edit access to documents which will be IP and individual account based authentication. The IP based authentication allows the admin to restrict access only based on a particular IP.
   For eg: it will allow the teachers to have edit access only if they are connected to the local network of the college and will change to view only based access when viewing the paper from anywhere else. This will allow us to individually identify each and every user accessing and editing the question paper on the database.
3. **Hypertext Transfer Protocol secure (HTTPS) session encryption and session keys.** HTTP over Secure Sockets Layer, or SSL, also known as HTTP Secure (HTTPS), will be used to enable a web protocol that encrypts data sent

between a browser and website. The encryption gets the data to its destination securely so it can't be intercepted and read in transit. This will prevent any Man in the Middle Attacks (MITM) to the server which will add an encryption layer of security to our system. HTTPS ensures a connection is private, using it when a browser sends a request to access a website, it generates a session key. The session key encrypts sent data and decrypts that data when it is received at the server side which will work in conjunction with the next addition to our system i.e Audit login.



4. **Audit logging** to log all the activities of the user on the portal. Which helps us to keep track of all the “Events” of a user on the portal. The Audit logs will track activities like
    1. Login/Logout of an User/Admin.

2. Creating/Deletion of an account.
3. All the Views/Edits/Downloads of data.
4. Creation/Deletion of data.

To Summarize : There is a portal on which every teacher who is responsible for setting the question paper has an account provided by the administrator. The teacher , based on his/her IP address will have access to the portal which can only be granted by the administrator.
Every time a teacher gains access or logs in an unique session key is generated for the teacher. This session key to transmit data over HTTPS protocol. It is used along with the user id and password to identify the teacher.
Each and every activity on the portal is logged using audit logs to detect a breach in the security. If an authorized log is found in the system, the administrators will be alerted and necessary action will be taken.

2. **Design a secure system/protocol to store the question papers separately.**

1. **Full disk encryption (FDE)** On devices where confidential data is stored or transmitted we must enable full disk encryption (FDE). Encryption protects the data in case the device falls into the wrong hands. In Windows, the FDE tool is called BitLocker. The macOS equivalent is FileVault. They both will utilize **XTS-AES-128 encryption** with a 256-bit key to help prevent unauthorized access to the information to be accessed without user authentication. This will prevent

anyone from physically decrypting the stored data on the database. If in a scenario the attacker gains access to the database it will be very difficult to decrypt the data as it is encrypted using one of the most secure algorithms.

2. **Encrypting Backups :** Backups are very important in a system like this. The portal will take frequent backups as well as encrypt them using **XTS-AES-128 encryption** and stored on a cloud based server**.** This will ensure that in case of an attack , the availability of the system is not lost and the data is still available.

3. **Delete Sensitive data :** All the unnecessary data will be deleted using the **DoD 5220.22-M (ECE)** designed by the **US National Industrial Security Program (NISP)**.This protocol overwrites the data to be deleted 7 times with pseudo random values

   The data to be deleted includes:

   - Deleting login data of previous unauthorized users.
   - Deleting log files on a regular basis.

4. **User details and passwords.** The details of the users along with their respective passwords are also stored on the database with an additional layer of security on top of FDE. Passwords are first Salted and Peppered then encrypted using a strong hash function such as SHA256 which will ensure us that the passwords are strongly encrypted and stored on the database.

3. **Design protocols for results to be communicated only to the authenticated user.**

Students Answer Paper -> Central System ----------

Let the students be x.

Center

Node

Node sends a request to center to send them a session key via Assymetric Encrytpion with request body with is digitaly signed. Now centre identifies the node and then processes the data by following method.

Centre has 2 keys – public and private

All Node has 2 keys – public and private

Centre generates a hashed sessionKey and then encrypts it with firstly by its own private key and then the public key of the node and sends it to the node.

Now node decrypts it via its own private key and then the public key of the center. Node can thus confirm if the sessionKey is truly generated by centre.

This sessionKey is only valid for few minutes let's suppose to be 5 or 10 at max and up to 1 wrong attempt to use it (whichever takes place first).

Now node again sends a request with the decrypted sessionKey and number of accessors available (let's suppose y) there to the centre (with a unique elements which defines if it demands papers for checking or supplies checked papers) to start a new session. Now centre can securely communicate with the server at the node. Now centre calculates appropriate number of answer sheets that are to be sent RANDOMLY using provided data (x and y) without disclosing the identity of students and only providing the paper serial number.

After checking the node again requests for sessionKey and uses the same procotol. After sessionKey decryption, node sends the corrected answer sheets to centre and results encrypted with the sessionKey.

Now centre has every student's school email ID. It send the appropriate result comparing the roll number and paper serial number data with its result and sends the locked result via an email. The mail has secure handshake procotol at the backend and hence, no manipulation can be done in the communication as various middlewares and api keys are used to send the mail.

The students can open the document with only their result with their password of lms portal account. Hence, it is now fully secured.

4. Discuss attacks and control on the designed protocol.

- If someone's result is leaked, then its by his fault and not due to any system fault or error. We can control this by warning to keep their password secured and keep checking logged in devices regularly.

- A separate API is made at backend at centre to kill the sessionKey once the required task is completed. Now if node admin forgets to log out or make an request to end session, then for the time it is valid, anyone with physical access to the computer or even via remote controlling can send authenticated request to the server and misuse the data thus exploiting the vulnerability in the software.

- Social engineering attacks are possible. Example, HoneyPot attacks and Web based attacks. Users can use VPN for further security.

- Node admins can be trained to only use their personal devices for personal use and not misuse company's

hardware and resources which could create vulnerabilities that could be exploited and harm the company’s records.

- If the number of requests to centre goes beyond 50, then it will be rejected automatically without any bit of processing. This would prevent DDOS attacks.