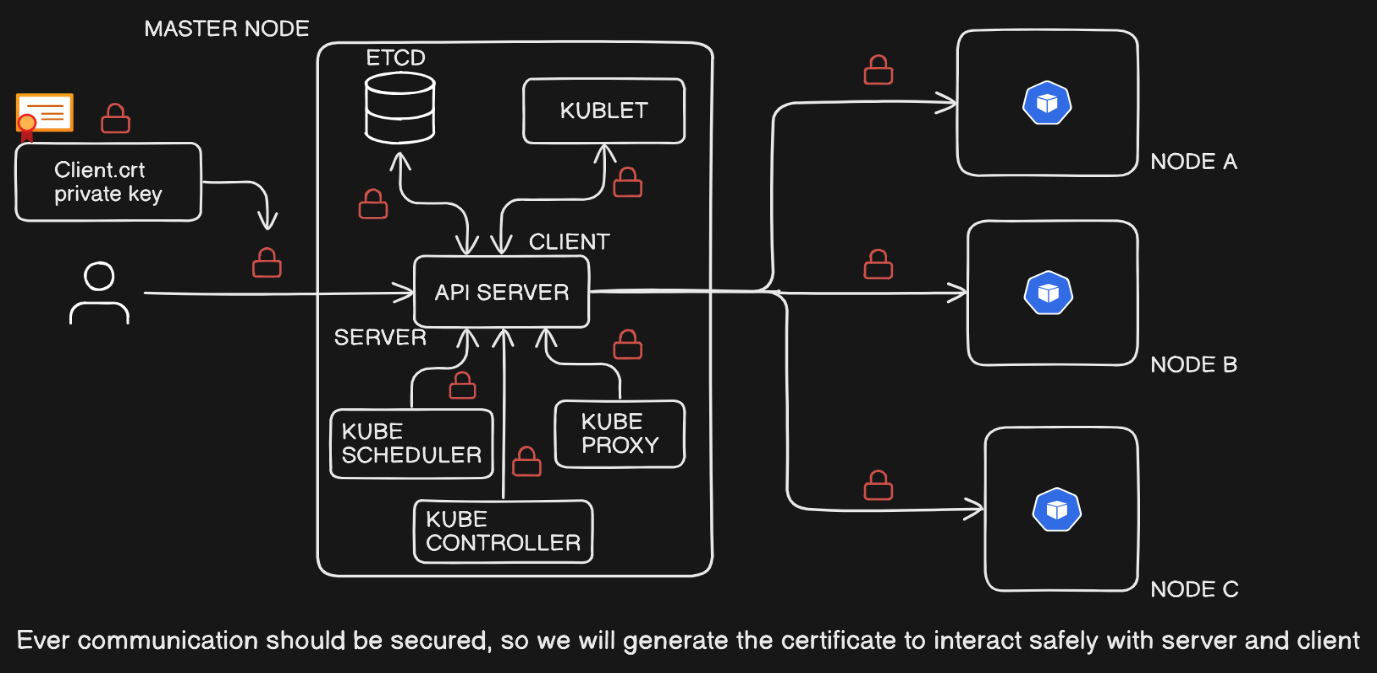
**SSL (Secure Sockets Layer) and TLS (Transport Layer Security)** certificates play a critical role in securing communications between components by encrypting data and verifying identities.

**SSL and TLS Certificates in Kubernetes**

* **SSL/TLS certificates** are digital certificates that provide authentication and encryption for secure data exchange over a network. They ensure the confidentiality and integrity of data transmitted between clients (like a user’s web browser or service) and servers.
* In Kubernetes, TLS is primarily used for securing:
  + **API server** communication: Protects data exchanged between the Kubernetes API server and clients, such as kubectl.
  + **Intra-cluster communication**: Encrypts traffic between pods or services.
  + **Ingress traffic**: Secures external traffic entering the cluster via an Ingress Controller.



**We Need SSL/TLS Certificates in Kubernetes**

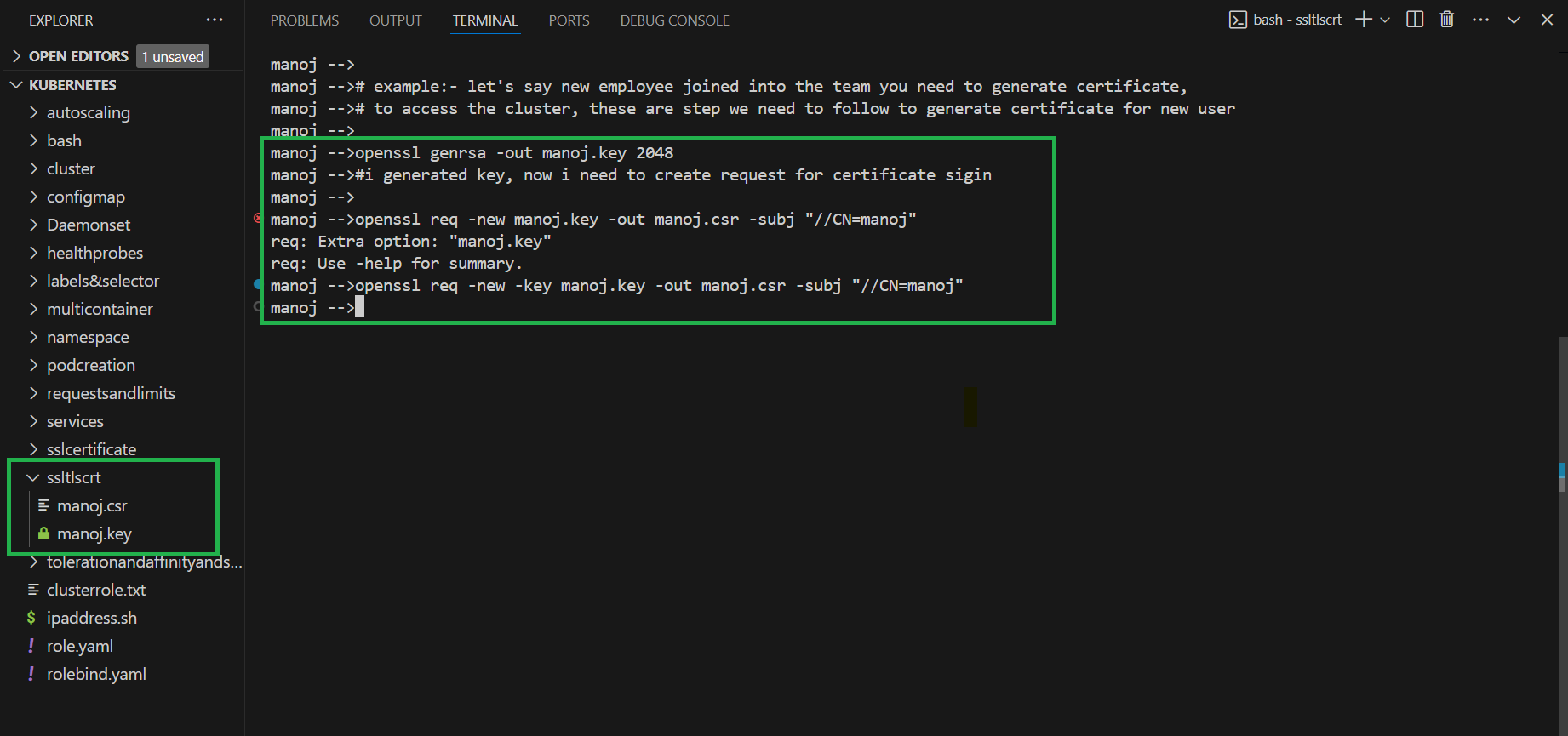
* **Security**: TLS certificates provide end-to-end encryption, preventing data from being intercepted by unauthorized parties.
* **Authentication**: Certificates validate the identity of clients and servers, reducing the risk of man-in-the-middle attacks.
* **Compliance**: Many regulations require data encryption, especially for sensitive or personal information.

**When to use SSL/TLS Certificates in Kubernetes**

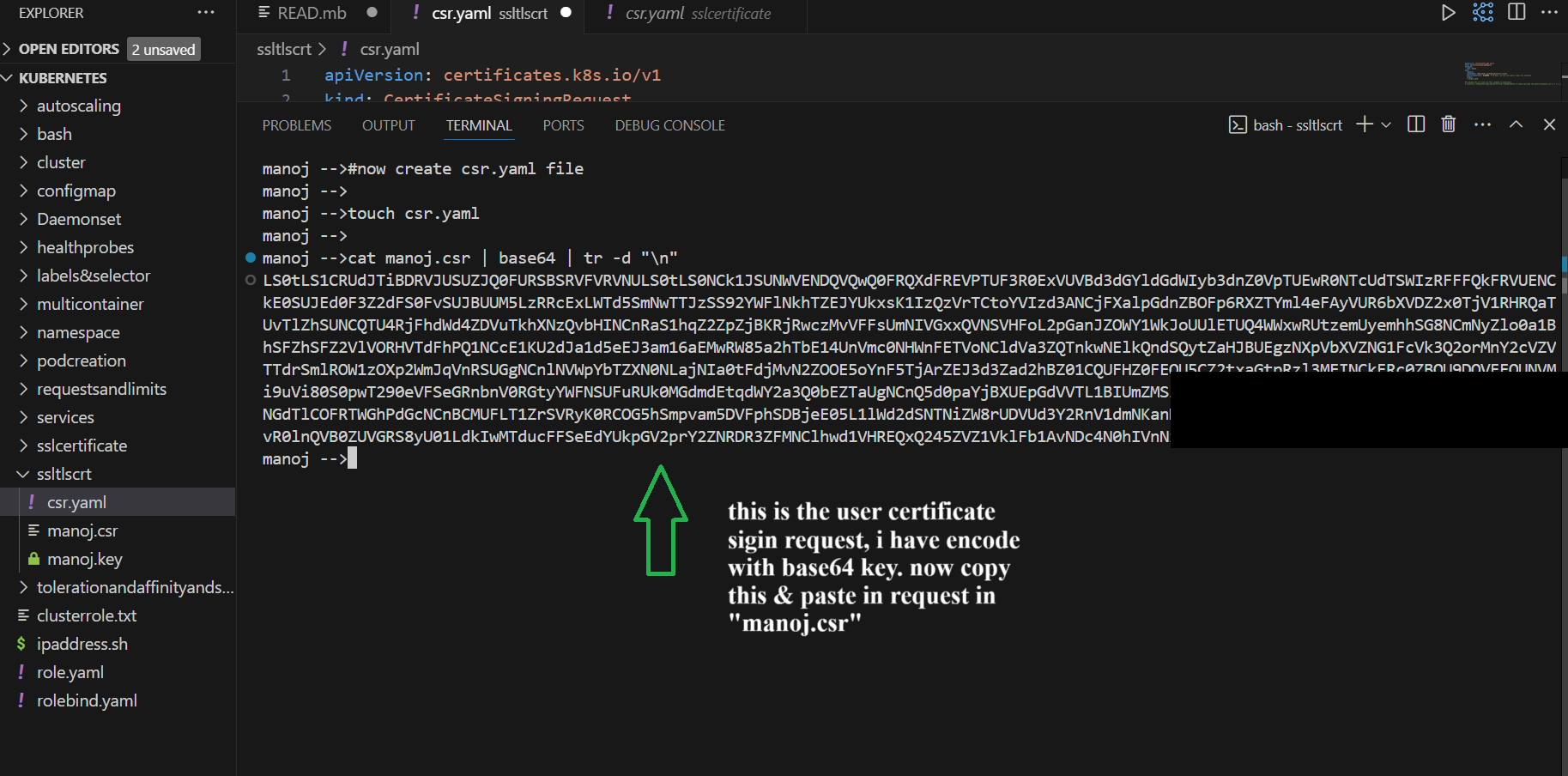
* **API Communication**: Kubernetes by default secures the API server with TLS to ensure secure access for kubectl and other clients.
* **Service Communication**: When sensitive data is transmitted between services or pods, encrypting intra-cluster communication adds another security layer.
* **Ingress Traffic**: When exposing services to the internet, TLS certificates secure user access, especially when handling personal or sensitive information.

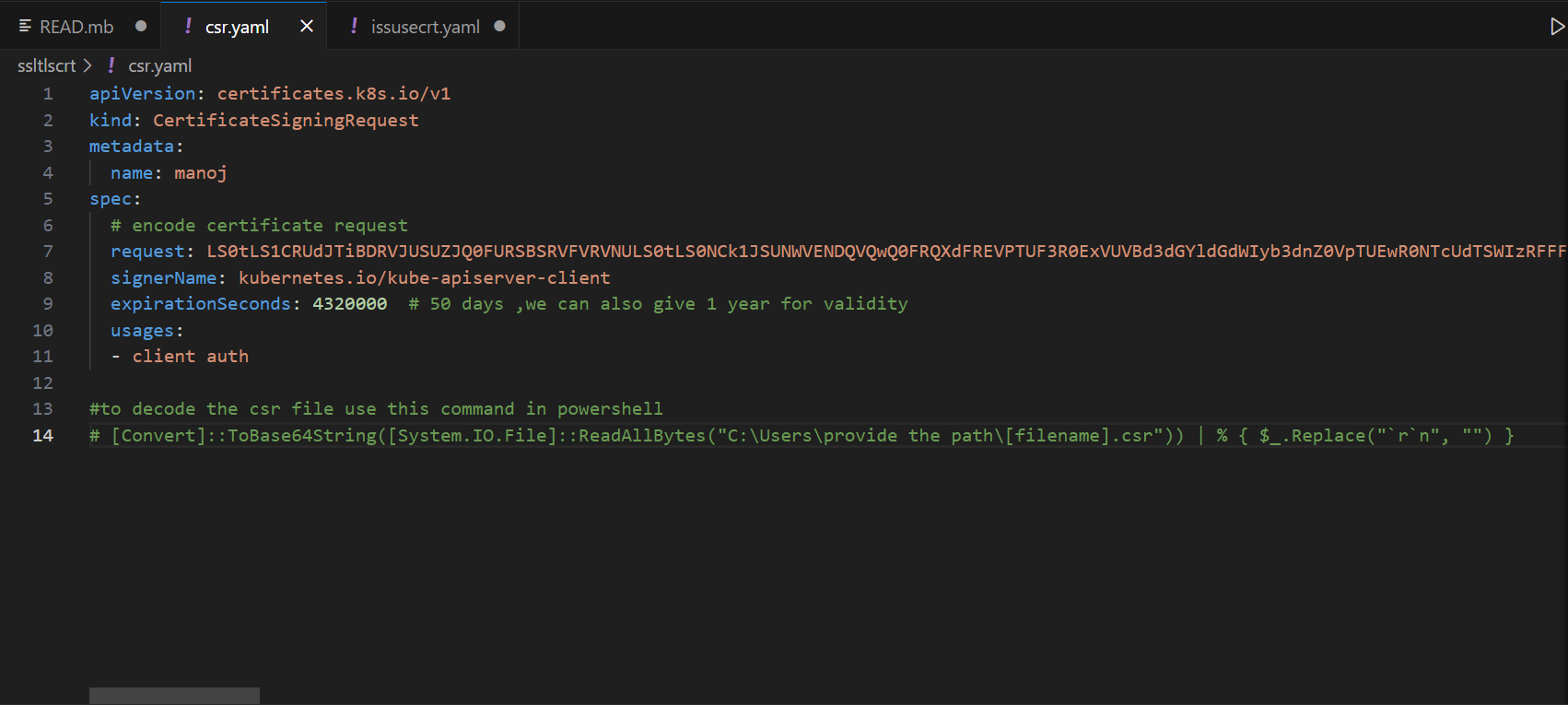
New employee joined inot the team I need to generate certificate to give access to the cluster.

Generate ‘key’

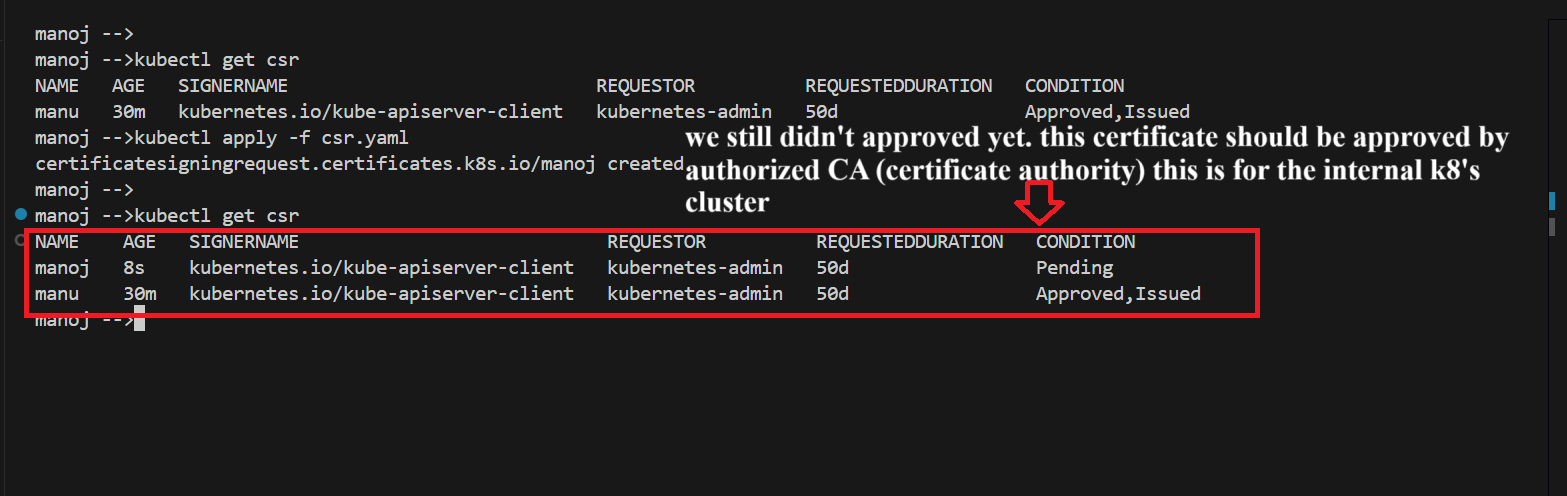


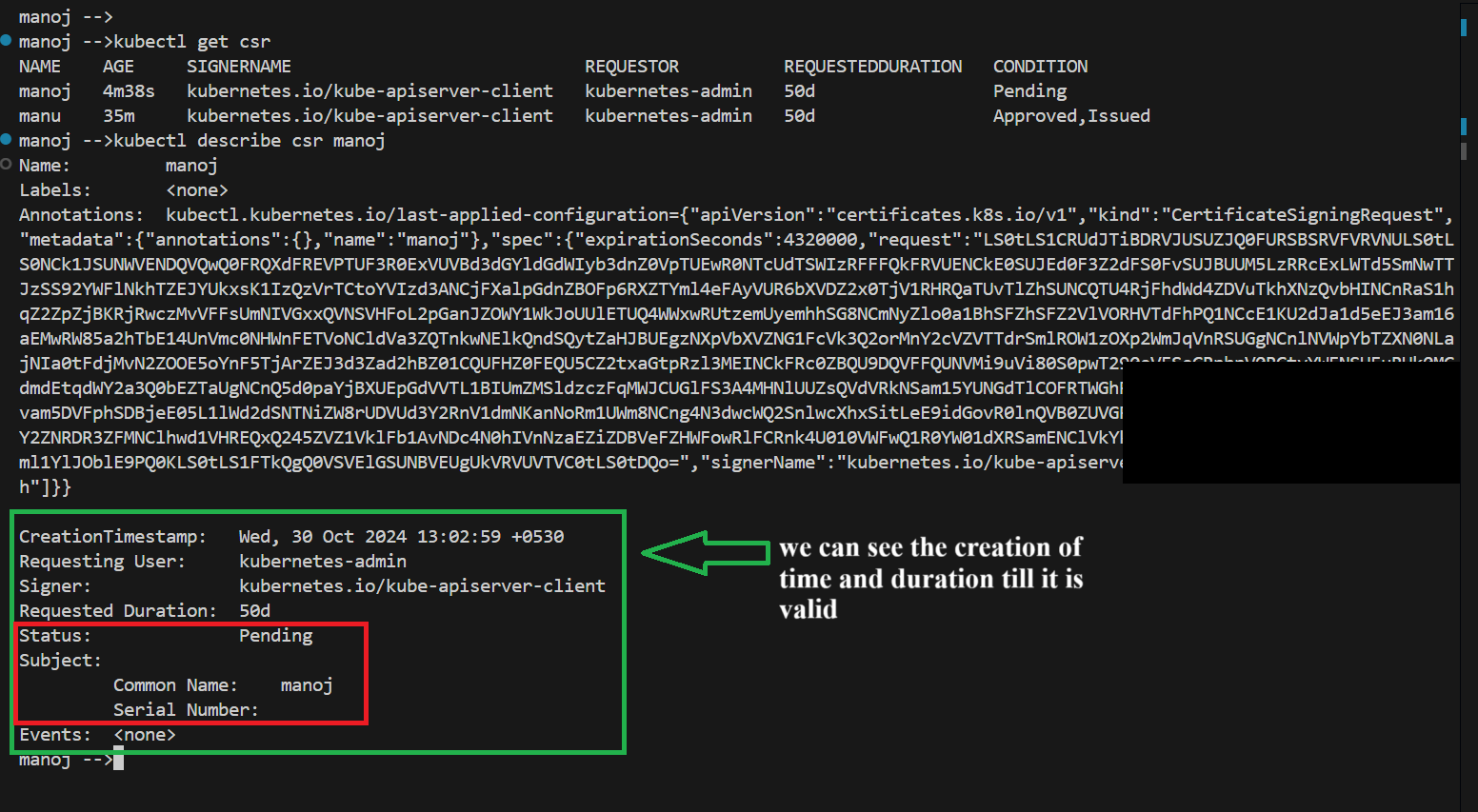
Create the request for certification sigin



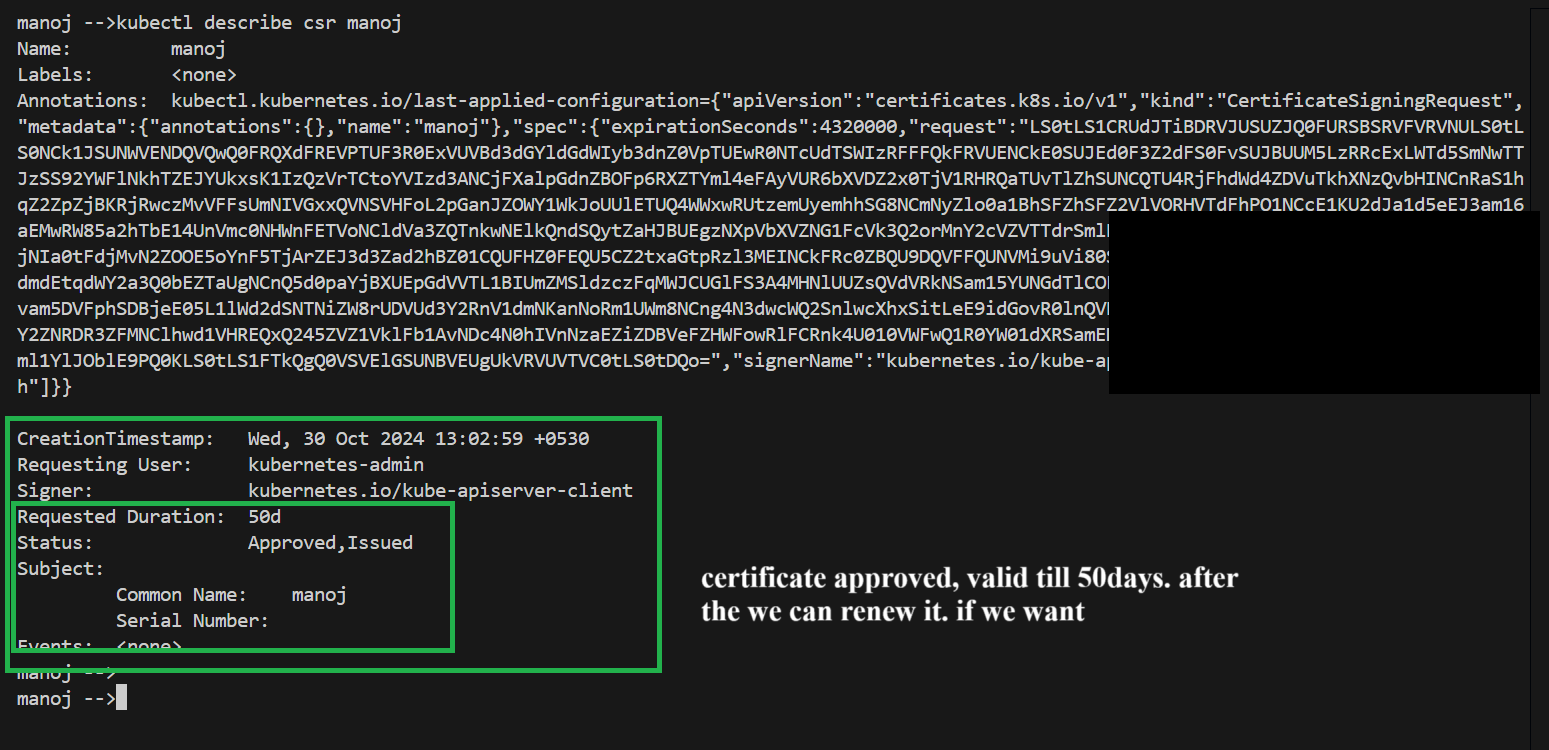


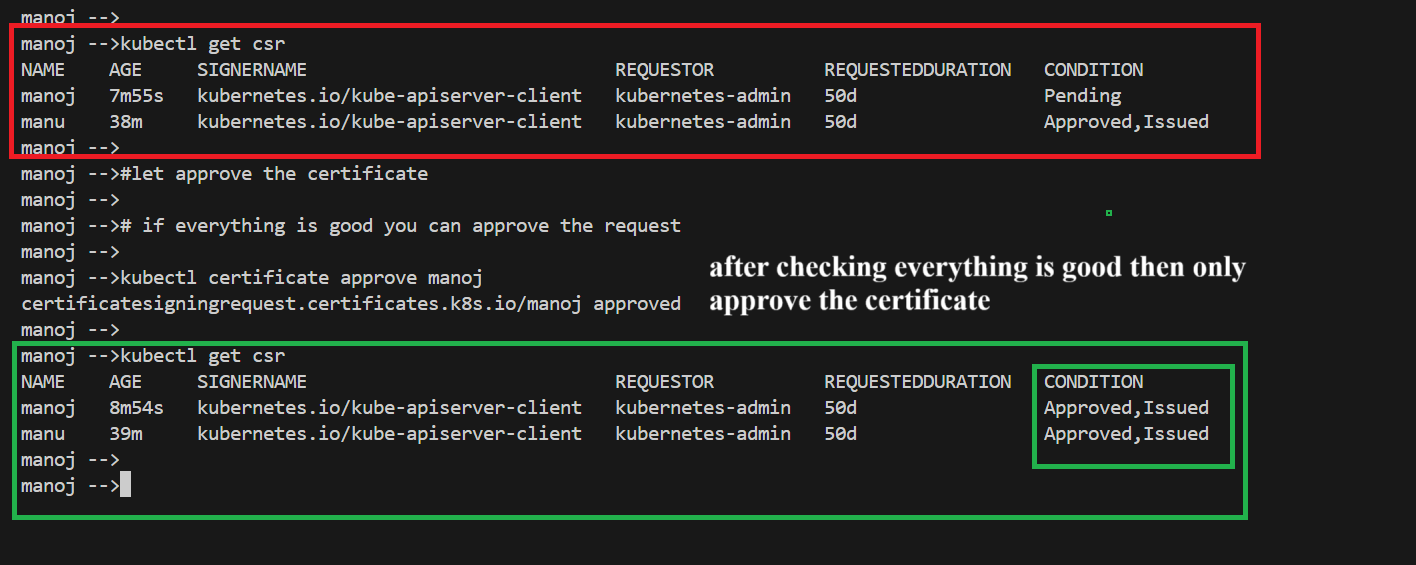
Created certificate signing request still in pending state

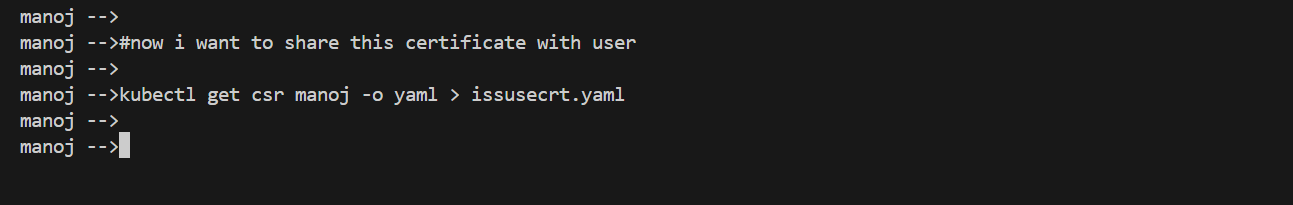




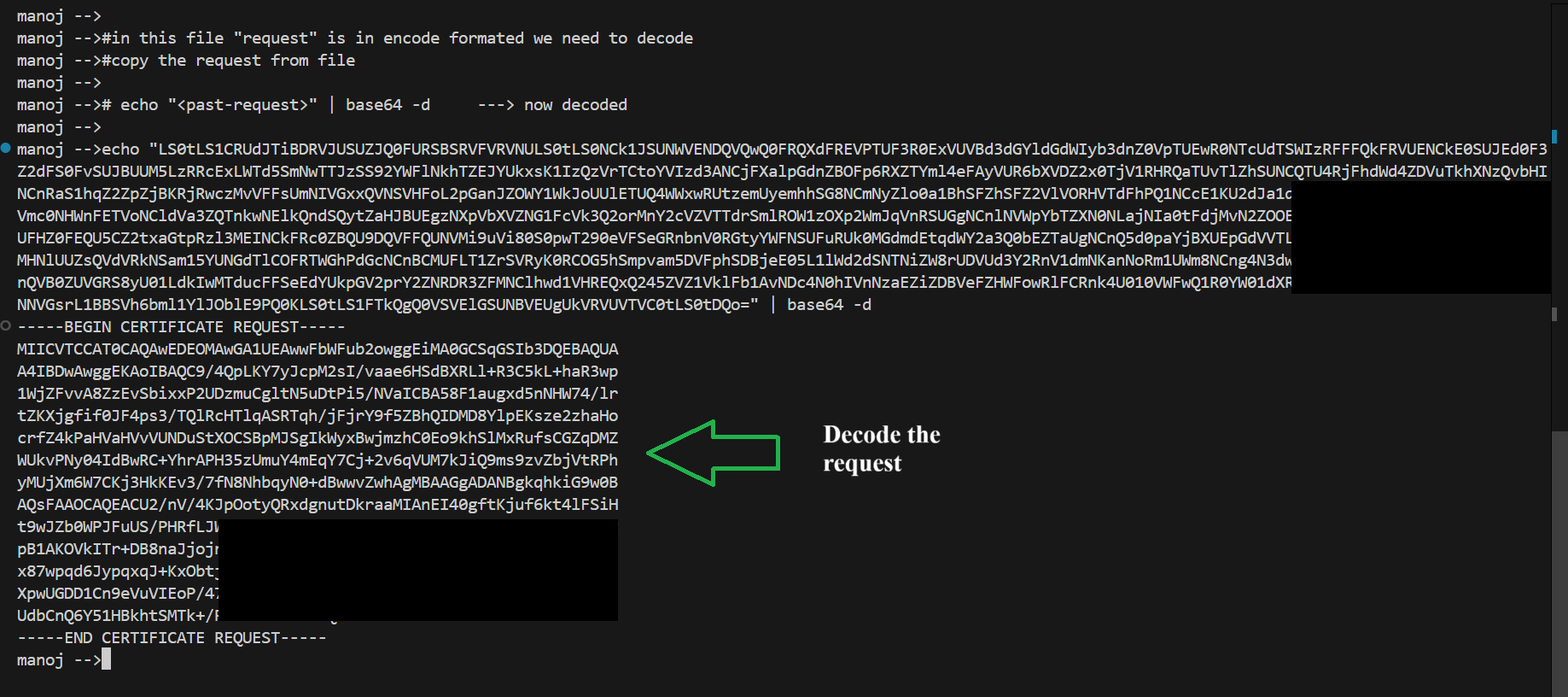
After checking very details then I need to approve the request



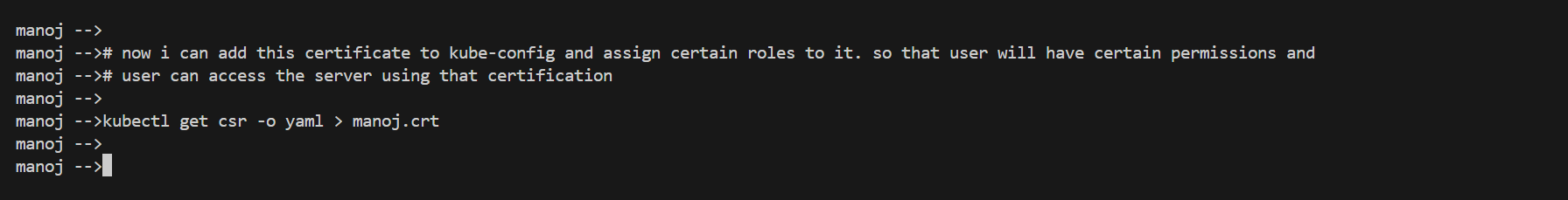


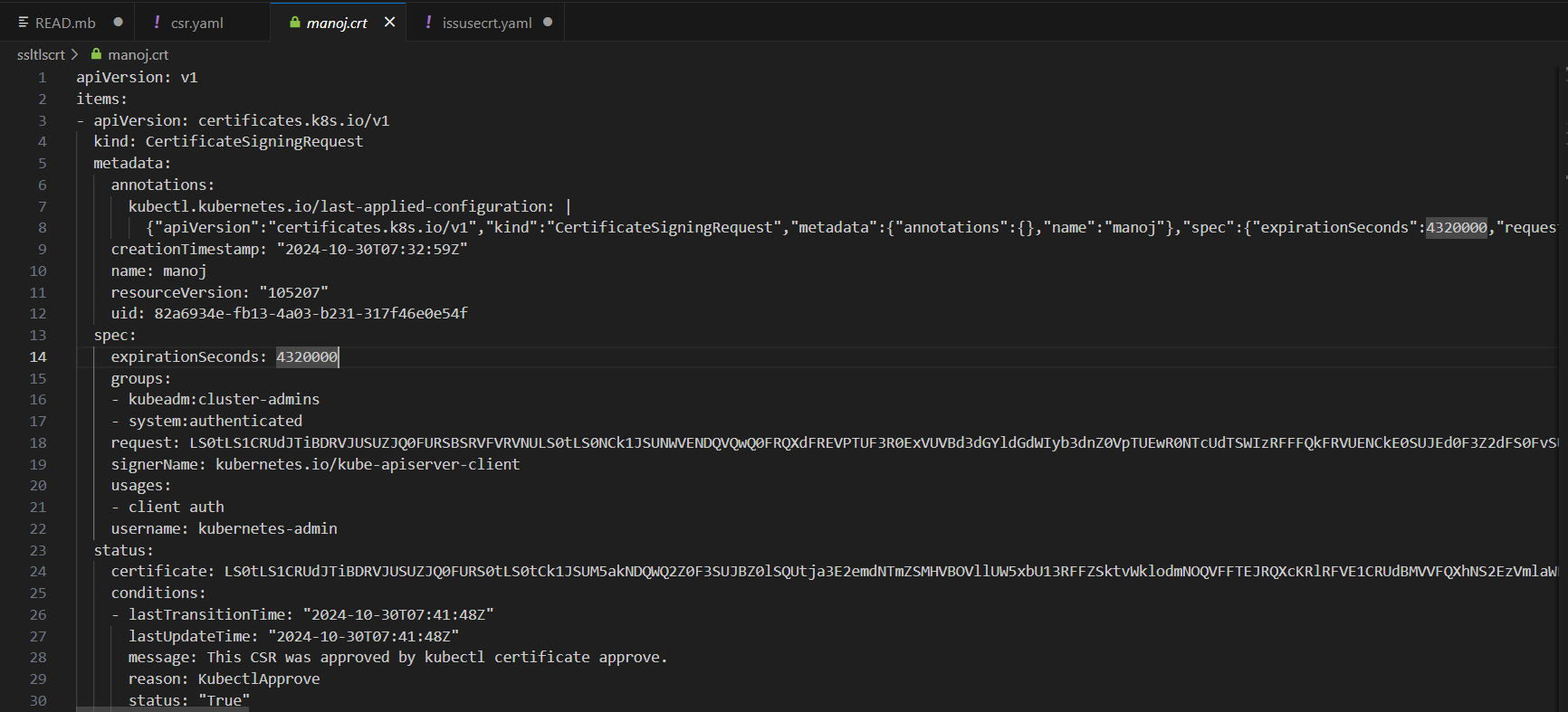






Now I can add this certificate to kube-config and assign certain roles to it. So that user will have certain permissions and user can access the server using that certificate.





Comparison between SSL and TLS in table format:

| **Feature** | **SSL (Secure Sockets Layer)** | **TLS (Transport Layer Security)** |
| --- | --- | --- |
| **Security** | Vulnerable to various attacks, weaker encryption | Stronger encryption, more secure, supports PFS |
| **Handshake Process** | Slower and more complex | Faster, simplified handshake, especially in TLS 1.3 |
| **Cipher Suites** | Limited, lacks modern algorithms | Expanded support for secure, modern cipher suites |
| **Deprecation Status** | Deprecated and unsupported | Current standard for secure communications |
| **Usage in Certificates** | Often referred to as "SSL certificates" due to legacy naming | Technically uses TLS encryption, even in "SSL certificates" |