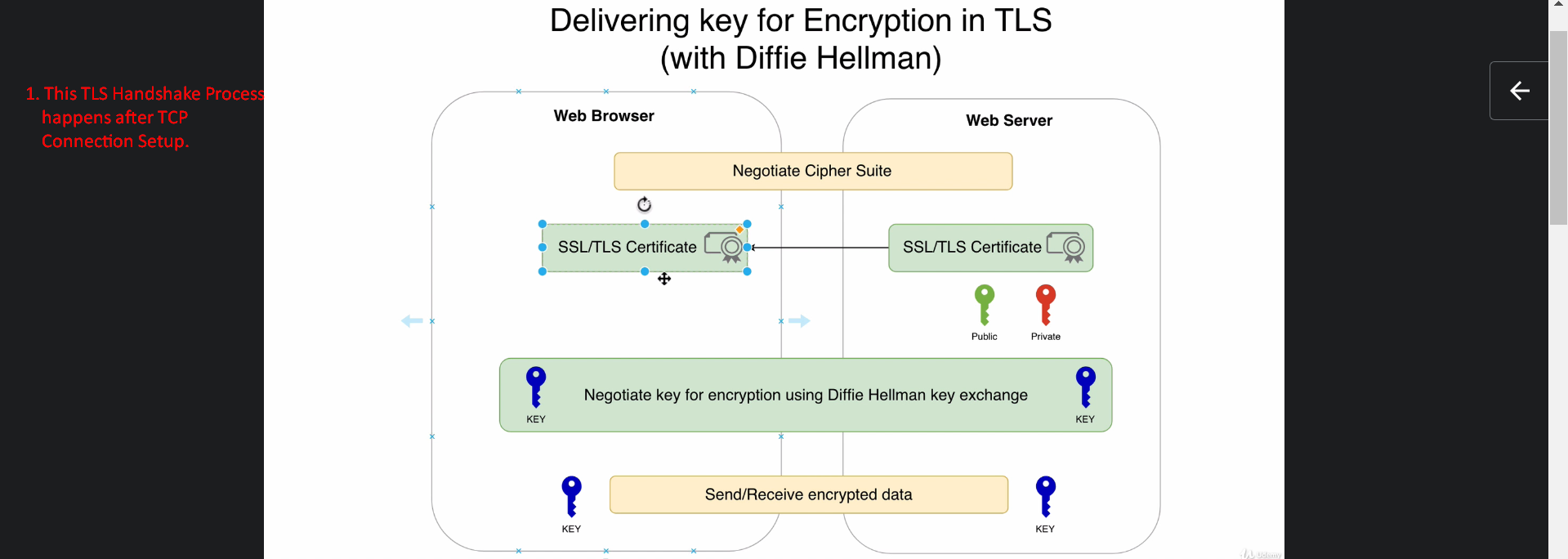
1. **Agenda**:
   1. How Key for encryption is delivered without Diffie-Hellman Algo.
2. 
3. As we know before sending certificate by webserver to browser, Cipher Suite is negotiated.
4. Browser sends a list of supported Cipher Suites and webserver chooses one of them.
5. Then Browers verifies the certificate and if verification successful, it means browser has authenticated the webserver.

After this, generation of key starts. 😊

1. Browser generates a random key which is encrypted using the Public Key from the certificate that was shared by the webserver and the key is sent to the webserver.   
   Webserver has corresponding private key, so it will decrypt the key to get the actual key.  
   Of course anyone can intercept our traffic and get the key but it is encrypted and only private key owner can decrypt.
2. Then this key will be used for both side data transfer.  
   This is delivery was done without Diffie-Hellman Algo.
3. Let’s discuss the drawback of such delivery.
4. As we know that RSA key pair is used for both certificate authentication and the key generation’s encryption, if that RSA’s private is compromised then that processor can decrypt the generated key.
5. 
6.   
   RSA key pair must be used only for certificate authentication (means web server authentication) not for encrypting generated key by the browser nor it should be used for encrypting traffic.
7. Jatin: Why not RSA public key must be used for encryption?
   1. Case 01: The browser uses RSA public key itself to encrypt the data.
      1. A hacker records all the traffic to server which is of course encrypted.
      2. After 6 months, the hacker gets access to the RSA Private key somehow.
      3. Now she can decrypt the data with the private key.   
         So, no Forward-Secrecy.
   2. Case 02: The browser doesn’t use RSA public key but using it, it encrypts some random secrete key and shares with server and now they start sharing data after encrypting it with that secrete key using some symmetric encryption such as AES.
      1. A hacker records all the traffic to server which is of course encrypted using secrete key + AES.
      2. After 6 months, the hacker gets access to the RSA Private key somehow.
      3. Now using that private key, she decrypts the encrypted secrete key.
      4. Now using decrypted secrete key and AES, she can decrypt the encrypted data.
   3. Solution:
      1. Using ECDHE, share ephemeral key both sides even over unsecured network.