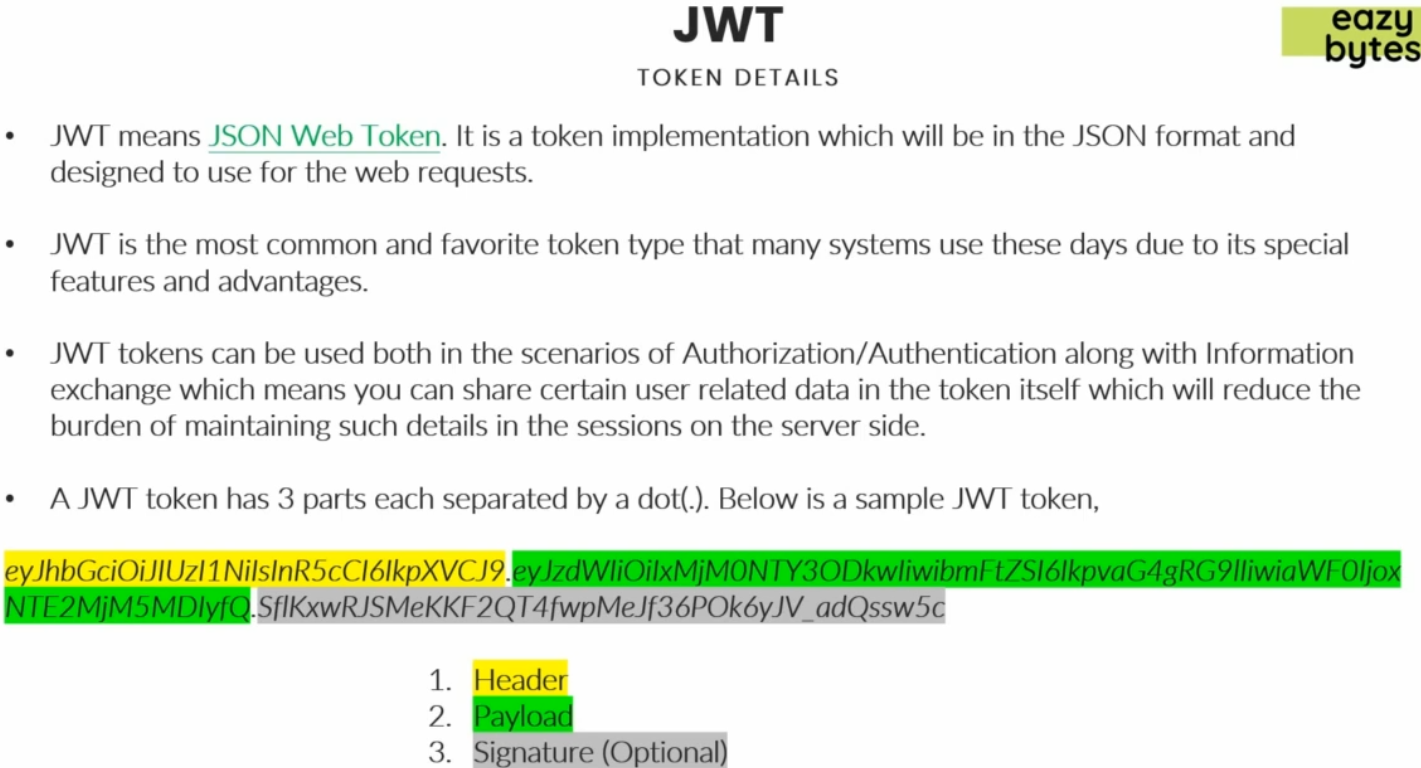
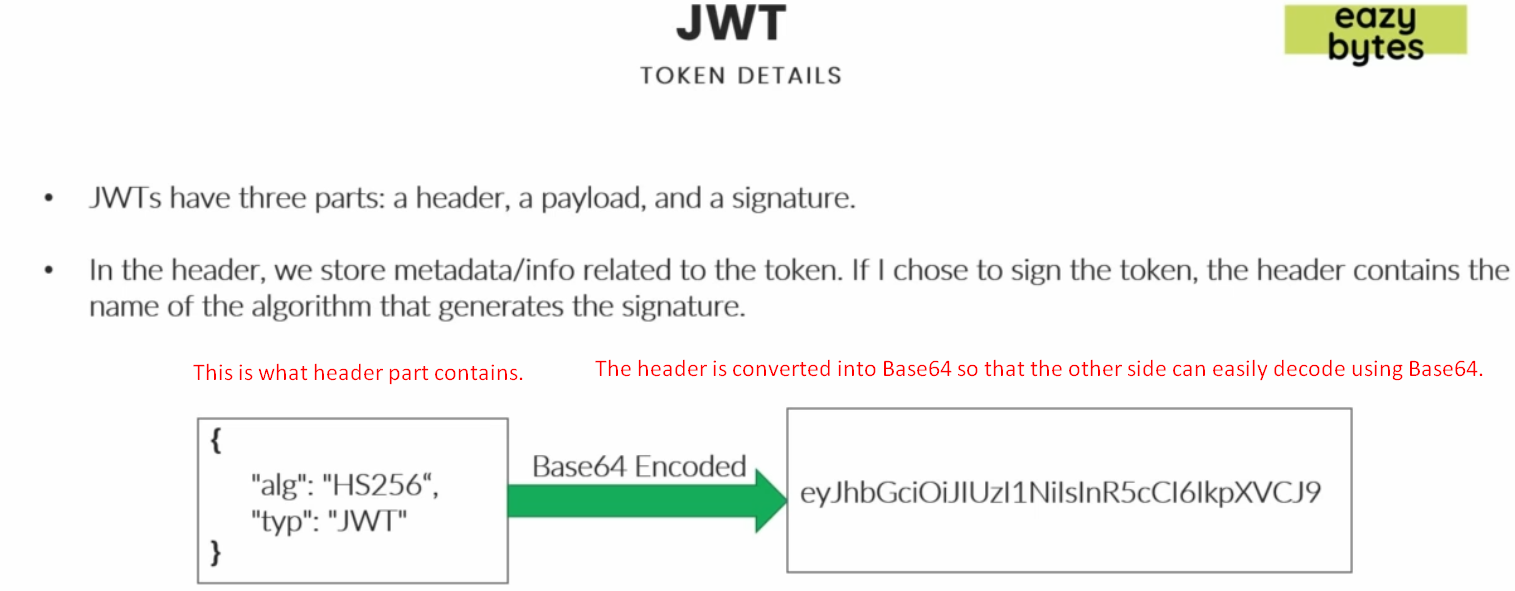
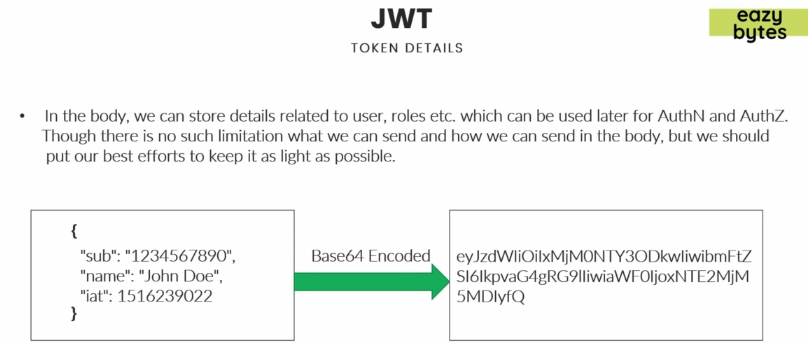
1. **Agenda**:
   1. What is JWT Token?
   2. What are the advanced features it provide related to normal tokens that we discussed in the previous lecture.
2. 
   1. **Advantages**:
      1. Light-Weight.
      2. JSON Format which can be understood by everyone.
      3. Allows sending data in encrypted format.
      4. Can be used beyond authentication and authorization such as if we want to send user related data.  
         Thus maintaining user info on client side so relieving server memory.
      5. Data is shared in encrypted format.
3. Parts of JWT Token:
   1. **Header**:  
      
      1. Base64 encoded value.
      2. **What it contains**:
         1. **Algo**: What kind of Algo was used to signature token?
         2. **Type**: Type of token.
   2. **Payload**:  
      
      1. It is the body part of the token.
      2. Storing user related info or any other info.
      3. **What it contains**:
         1. Info related to User, roles which can be later used for authentication and authorization.
      4. No limitation what can be put inside the body and what is the size of the body.
      5. **Why Putting all these info inside the body**:
         1. So that the other side can use these roles and authorities for their business logic.
         2. Now we don’t need to maintain all this info in the server side.   
            Now all this info is stored in the client side.
   3. **Signature**(optional)
      1. Optional.
      2. It is used when going to encrypt making sure that no one tempered header and body parts of your JWT.
4. **Signature**: The most important part.
   1. We will consider a scenario where we will generate a token and sending the token to other internal application inside my company firewall.
      1. So in this scenario, I’m pretty much sure that a hacker can’t access the token and manipulate it.   
         For such scenarios, I can simply send the header and body of the token since I’m believing the other side of the app and we can communicate any such info b/w my app and the other side app.
      2. But if we have a scenario, where the token will fly into the open world through the web, then we have to make sure there are enough measures to detect that someone tampered that token because you’re not storing that token anywhere inside your backend app like in server memory, db then how will you validate if someone sends where this is the really that you generated or not.
      3. So, for that we use the digital signature.
      4. Start from 06:02