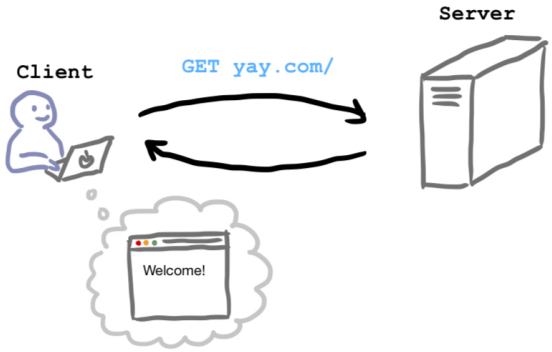
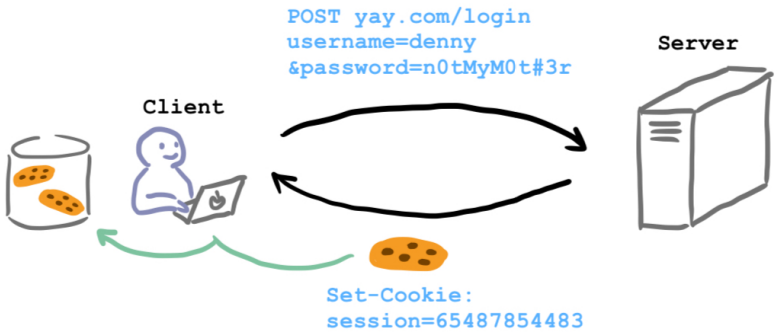
The different players involved when someone visits a website are:

- The **user**
- The user's **computer**
- The **browser**, which is a process running on their computer
- The **web server**, which serves data constituting a website




Communication between your browser and the web server

- A browser communicates with a web server by sending **HTTP requests**.
- When you click on a link, your browser makes a **GET request** (a type of HTTP request).
- When you submit a web form, your browser often makes a **POST request** (another type).
- The web server sends back **HTTP responses**.
  - These can contain HTML, CSS, Javascript, image files, or any other kind of data.
  - When your browser receives data, it renders it and displays it to you as a web page.
- **Forms** are the elements you interact with in your browser to submit data to a website.
  - Submitting the form triggers your browser to send a POST request containing your data.
  - Some forms look like what you expect, such as a form for creating an account on a website.
  - Some forms might look like like/dislike buttons, comment fields or search bars.

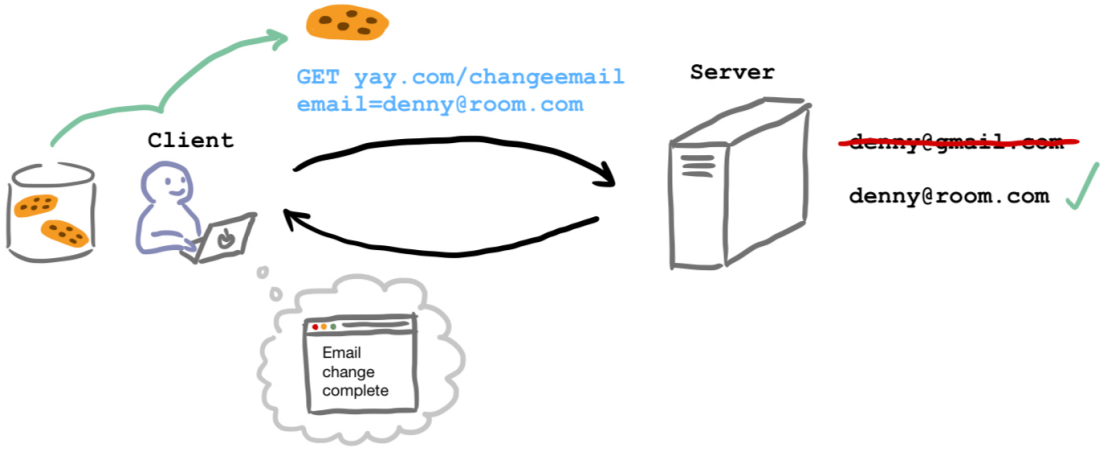


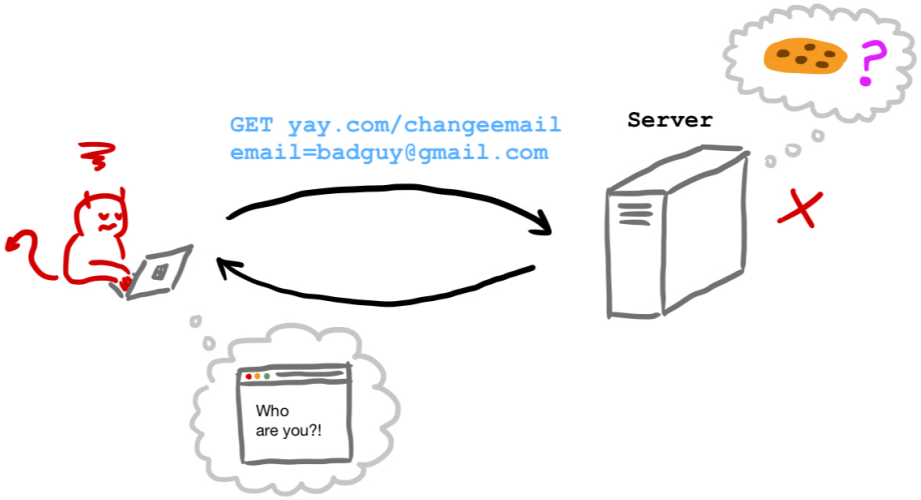

Security and session management in web apps

- Some websites have **access control**, allowing some users to take actions that others can't.
- These websites have to use **authentication** to confirm a user's identity.
  - A common form of authentication is to ask users to login with a username and password.
- Users are given **sessions** so that they don't have to re-login each time they take an action.

- After a successful login, the web server issues a **session cookie** to the user.
- Browsers automatically attach your cookies to any request to the same domain.
- A session cookie can authenticate your requests for a short time.
- Once it expires, you have to log in again.


- If an attacker can guess your username and password, then they can control your account.
- If they can guess your session cookie, then they can control your account for a short time.
- Otherwise, only requests from your browser will be authenticated as coming from you.



Cross-Site Request Forgery (CSRF)

- Sometimes an attacker can cause you to take an undesired action without this information. They can do this by causing your browser to make a request on your behalf.
- If you click on a link, your browser will make a GET request to that link.
- If you visit a webpage, it can redirect you (GET) or submit a form from your browser (POST).
- An attacker might trick you into clicking on a link or visiting a website that they control.
- Remember that your cookies are automatically attached to your requests.
- When your browser sends your cookies to the web server, it authenticates the request as coming from you.
- This kind of attack is called **Cross-Site Request Forgery (CSRF)**.