

# Tracking Session Data

---



**Sekhar Srinivasan**

@sekharonline4u - [www.sekhartheguru.net](http://www.sekhartheguru.net)

# Module Overview



Hidden Fields

URL Rewriting

Cookies

Sessions

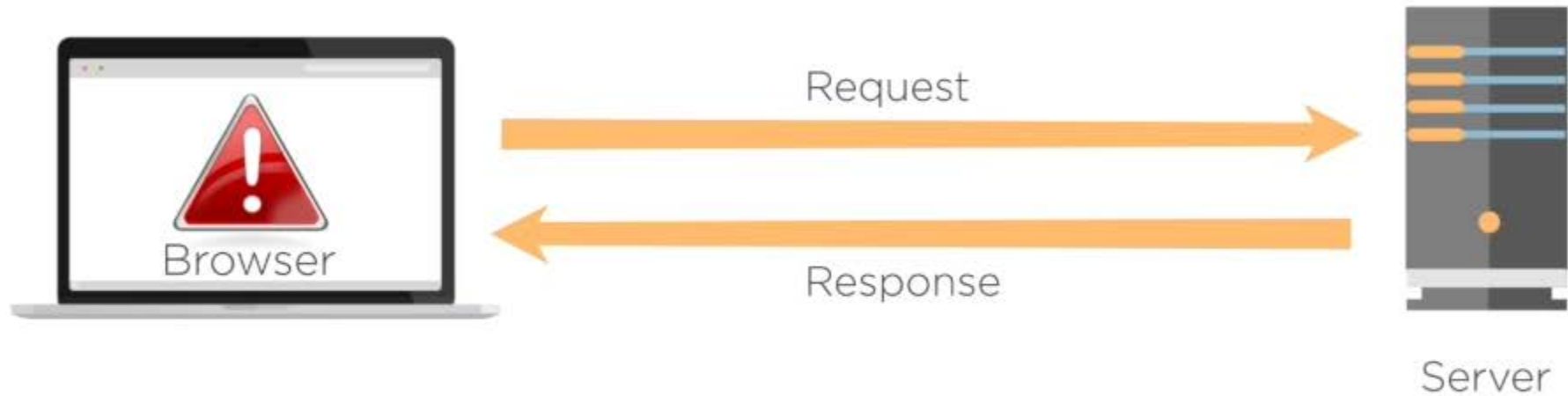


Why we need to track the session data

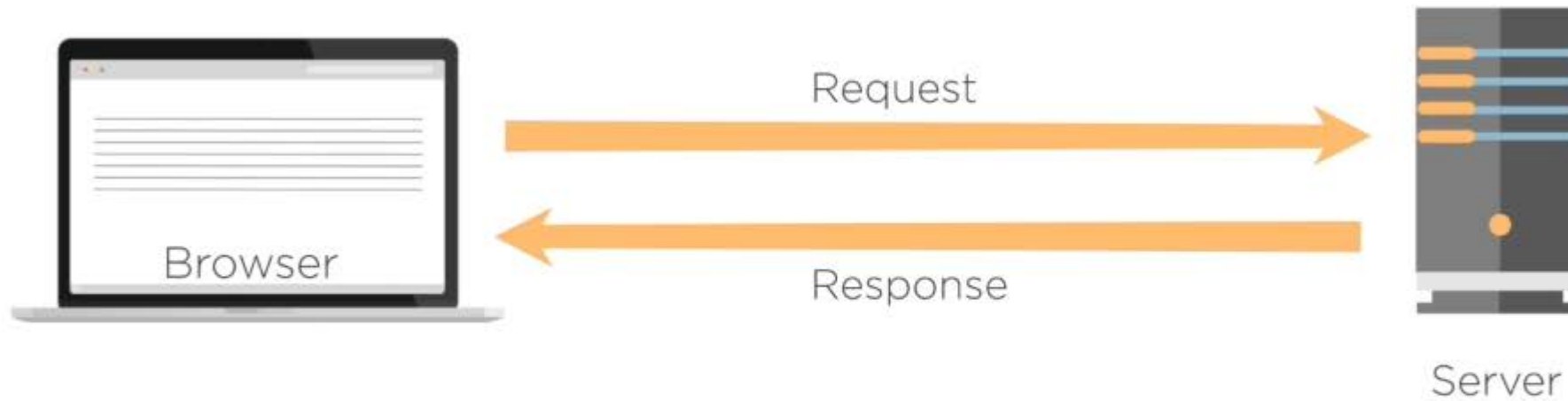
# Requirement: Tracking the Session Data

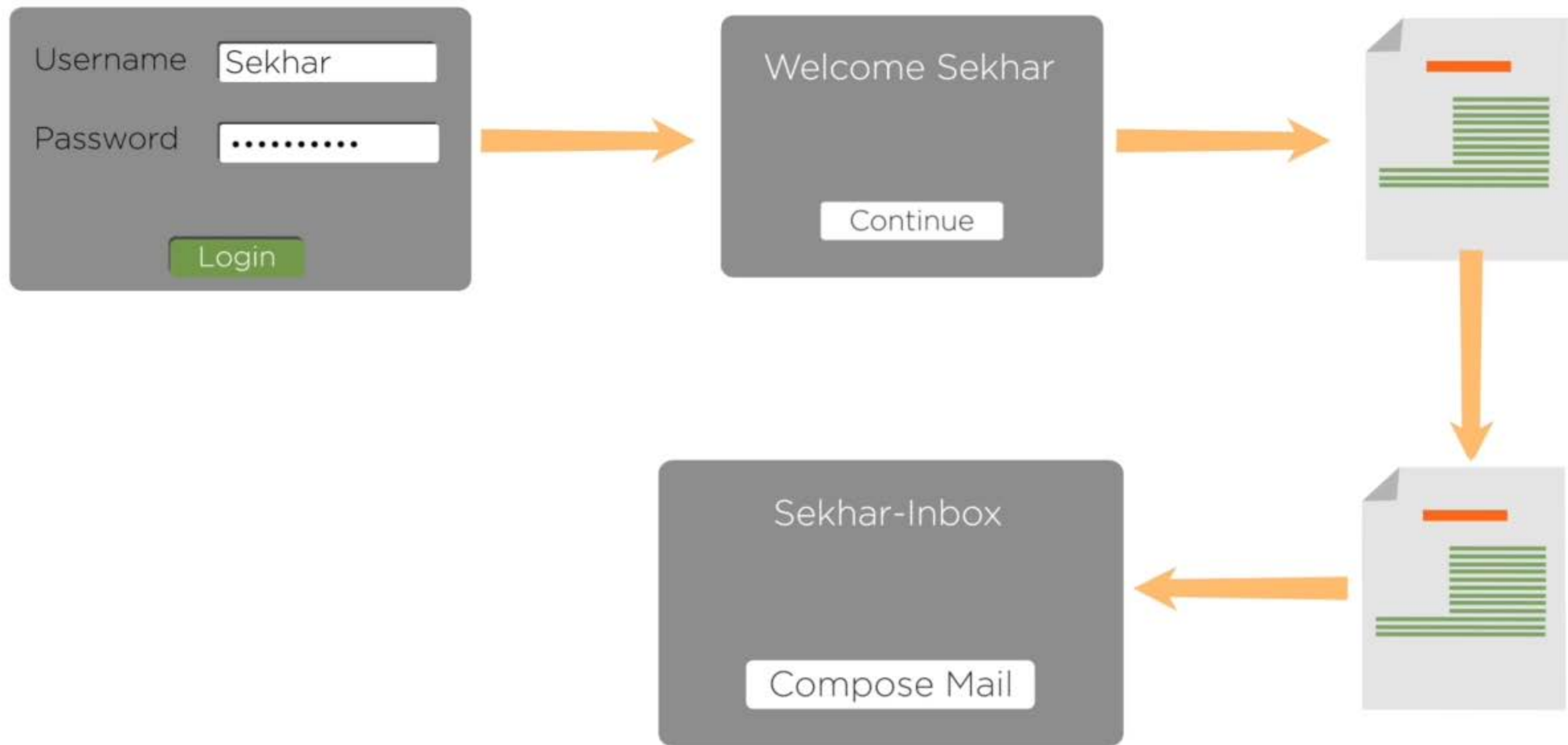


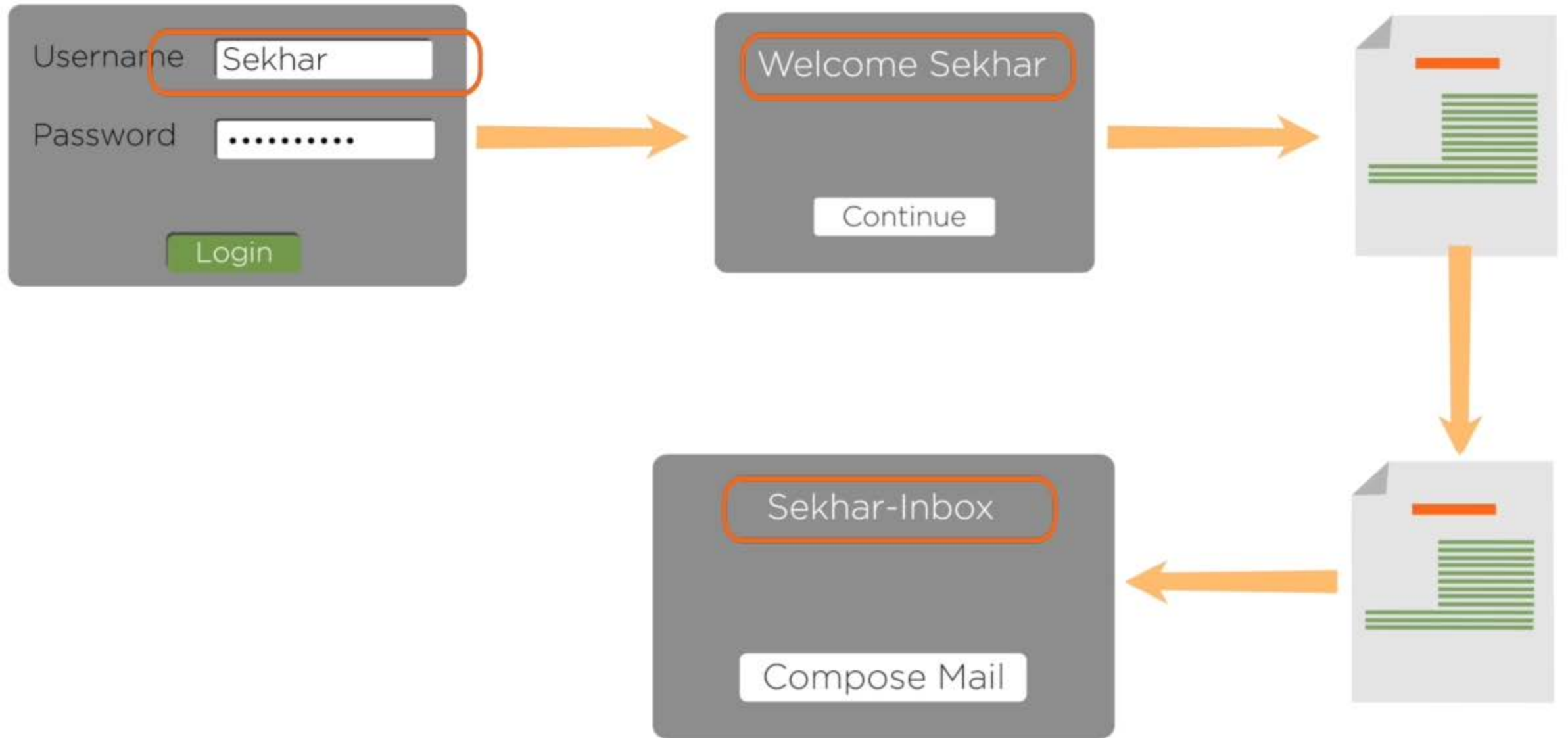
# Requirement: Tracking the Session Data



# Requirement: Tracking the Session Data











Email



Shopping Cart



Chat



Forums

# Tracking Session Data

Hidden Fields

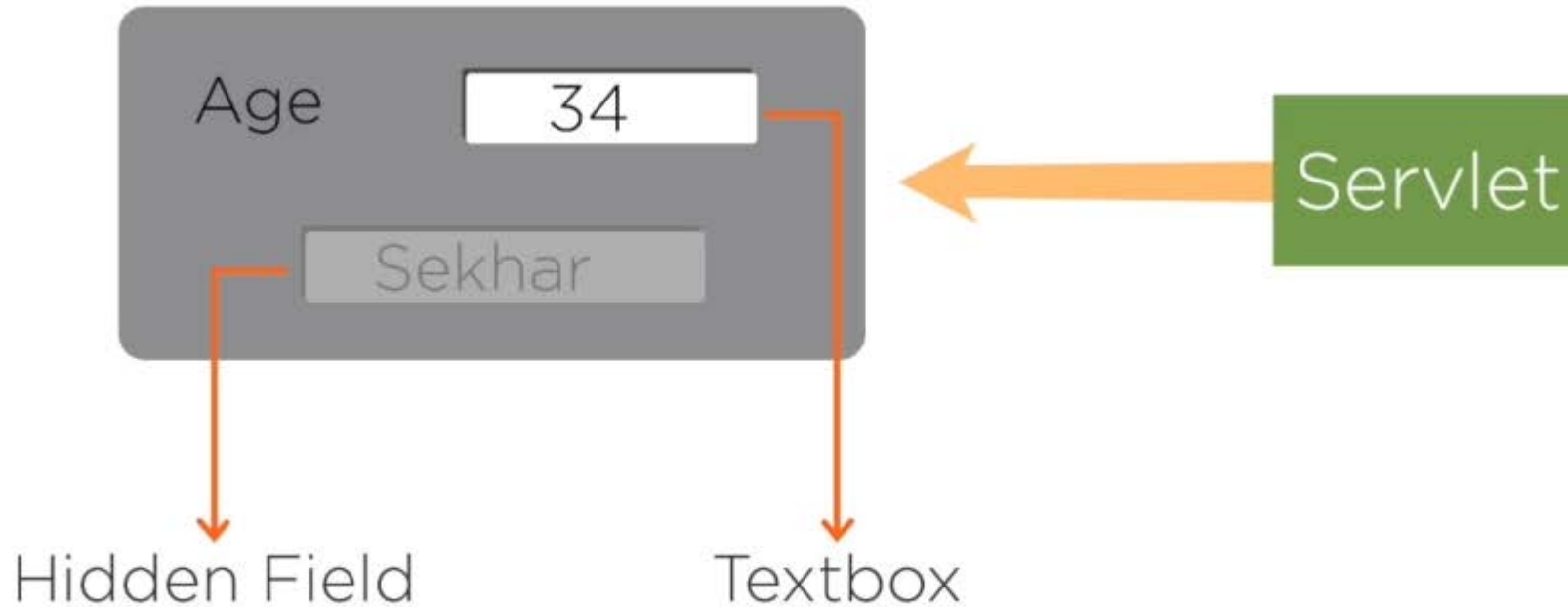
URL Rewriting

Cookies

Sessions

# Hidden Form Fields

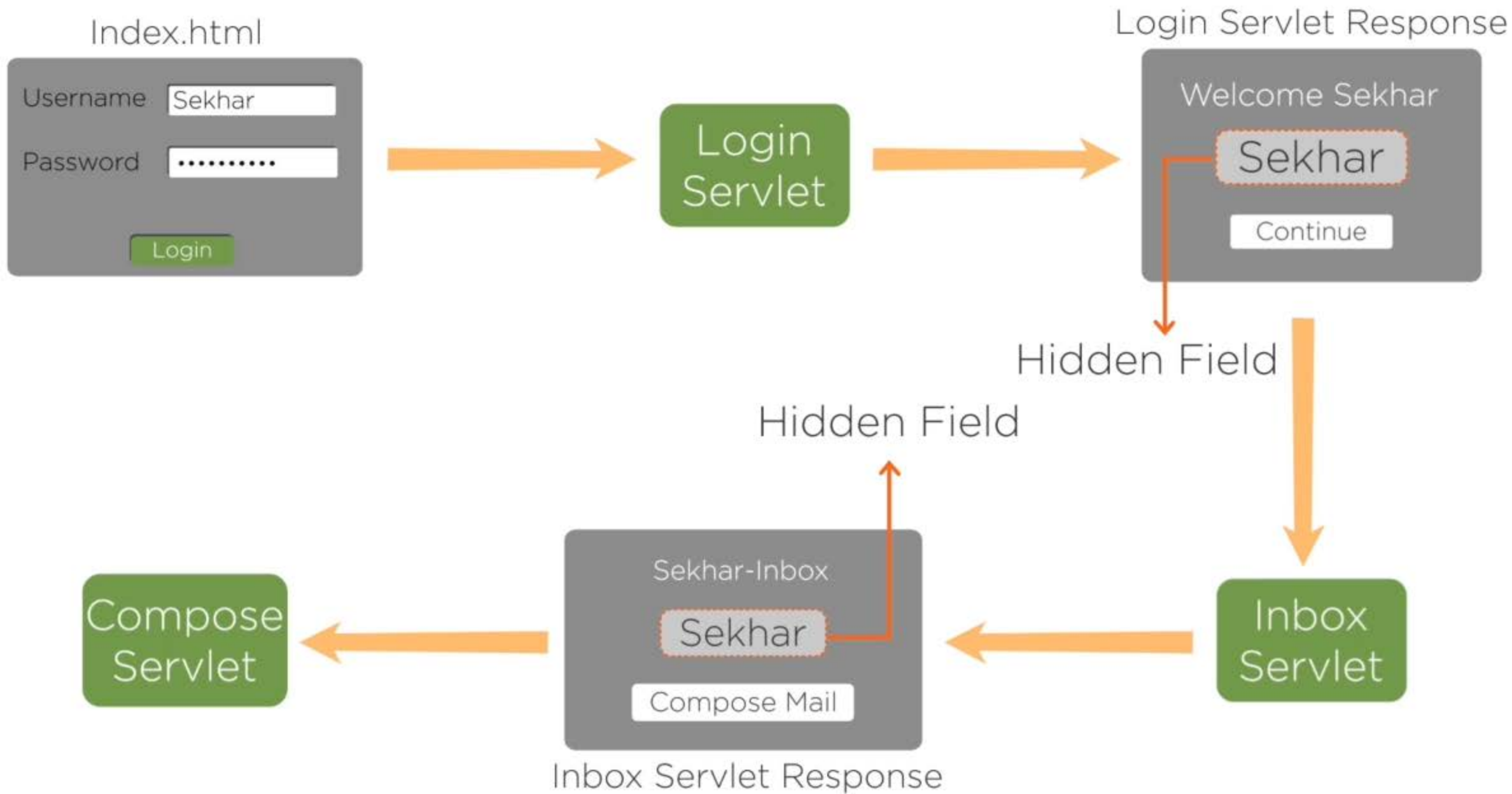
An HTML Form element used to store the value similar to Textbox

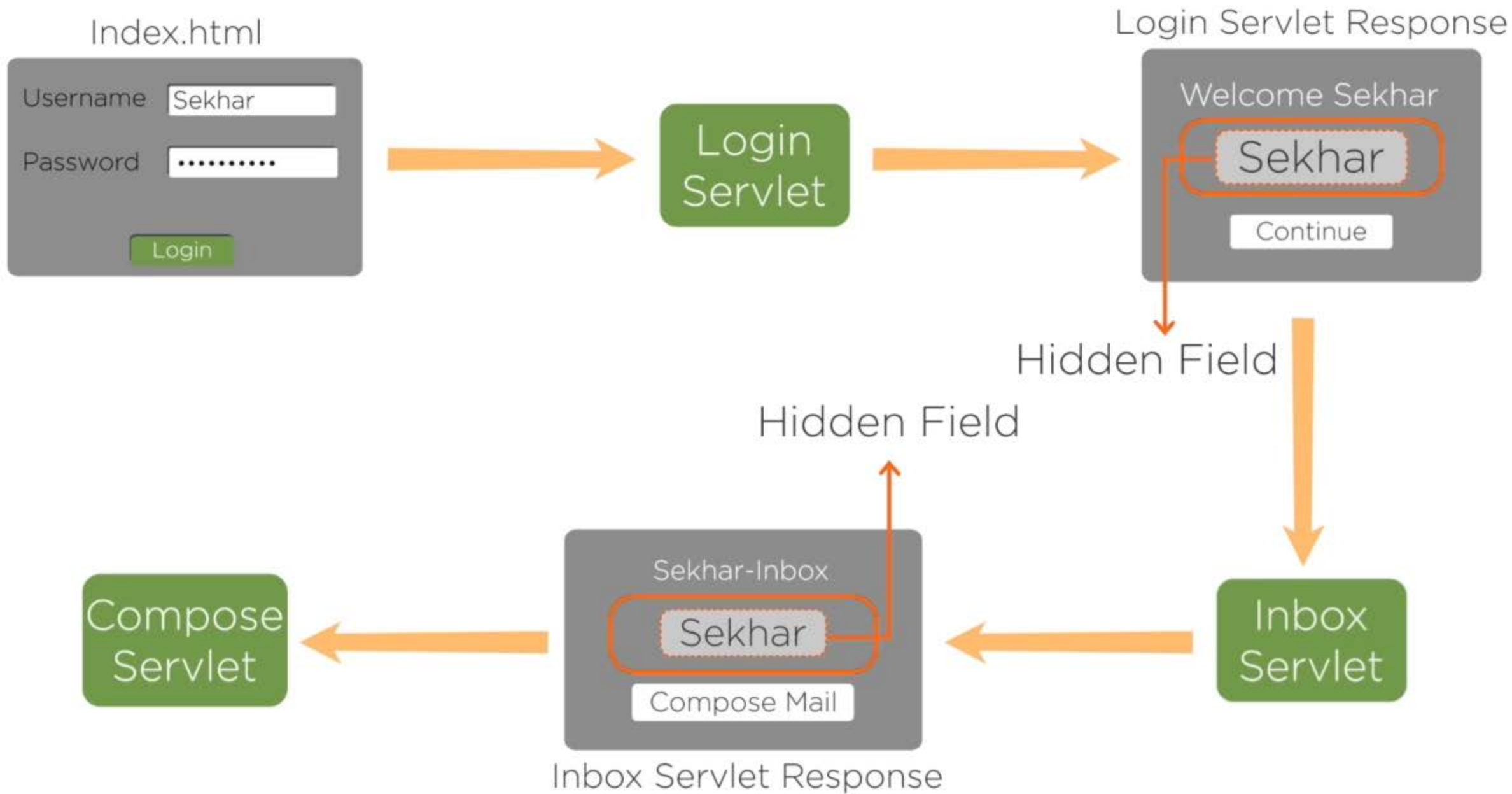


```
<input type="hidden" name="FormElementName" Value="Value  
to be maintained across the pages" />
```

Syntax:

To define an Hidden Form Field

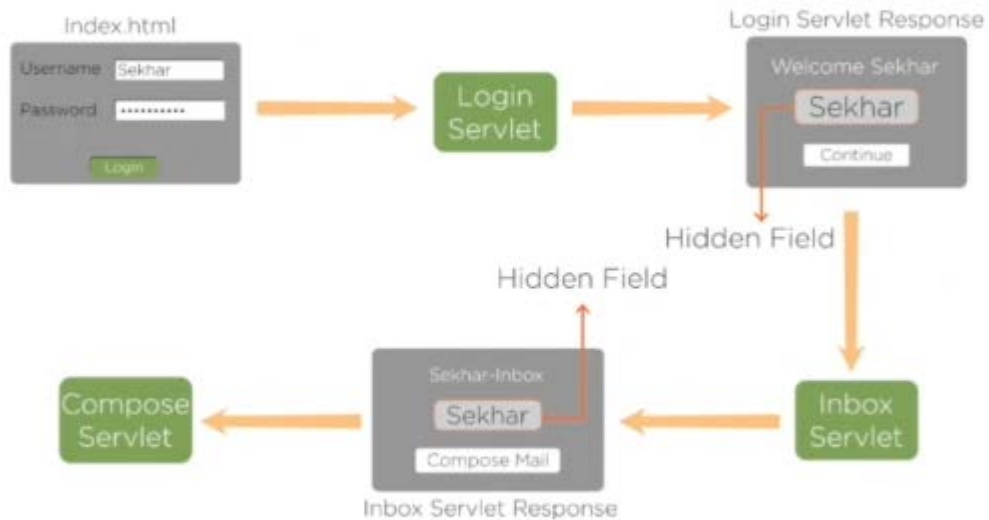




# Demo: Hidden Form Fields



# Hidden Form Fields



## Pros:

Supported by all browsers



## Cons:

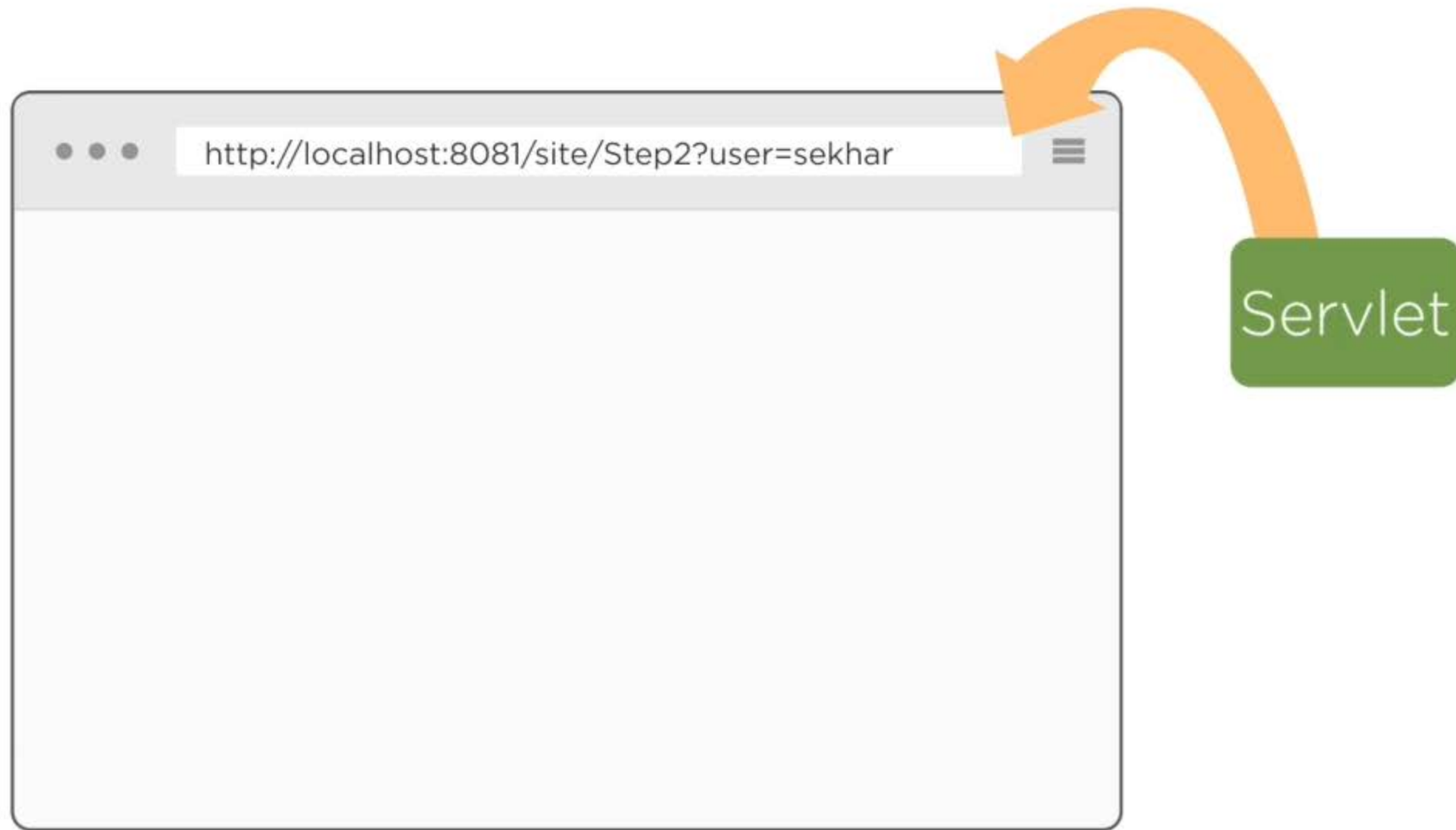
Complex to implement

Mandatory to submit the data

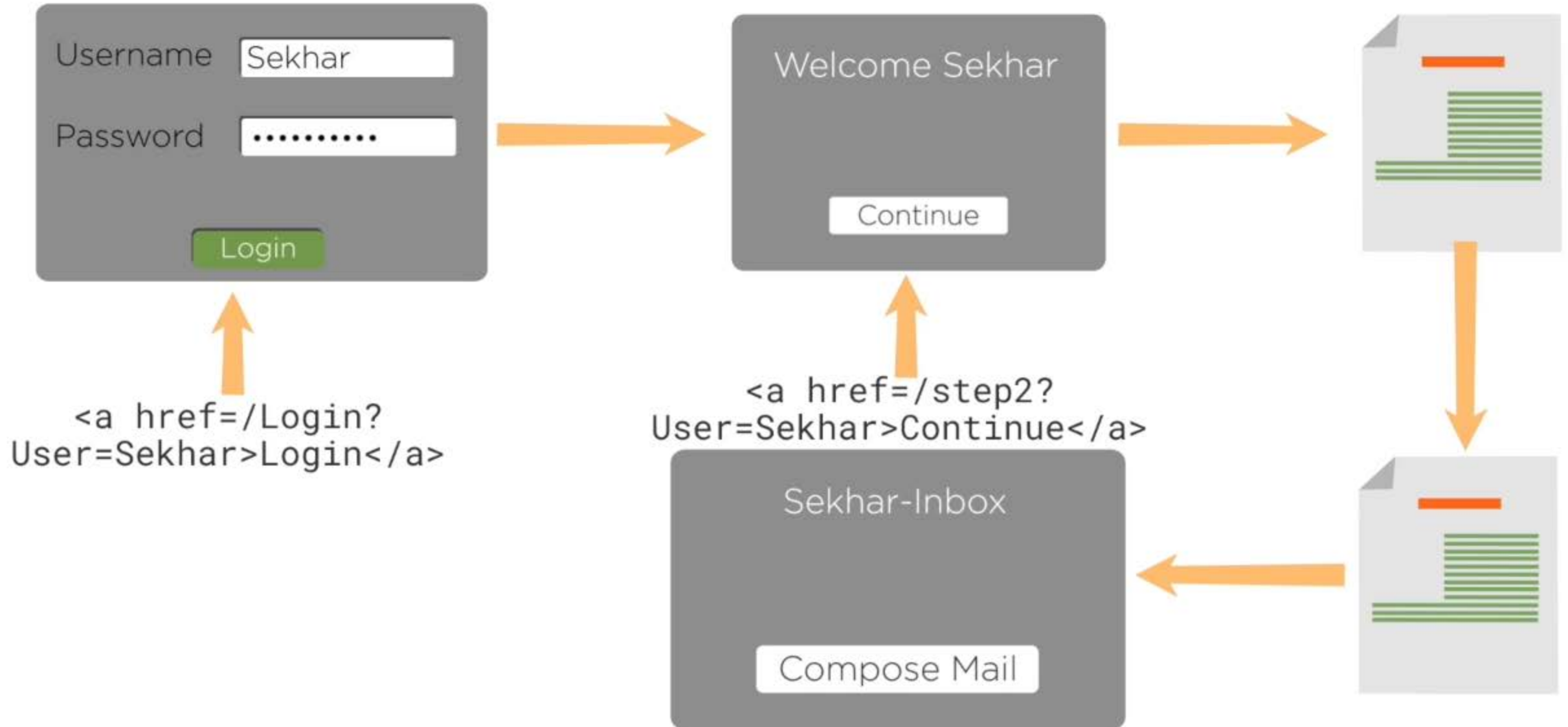


# URL Rewriting

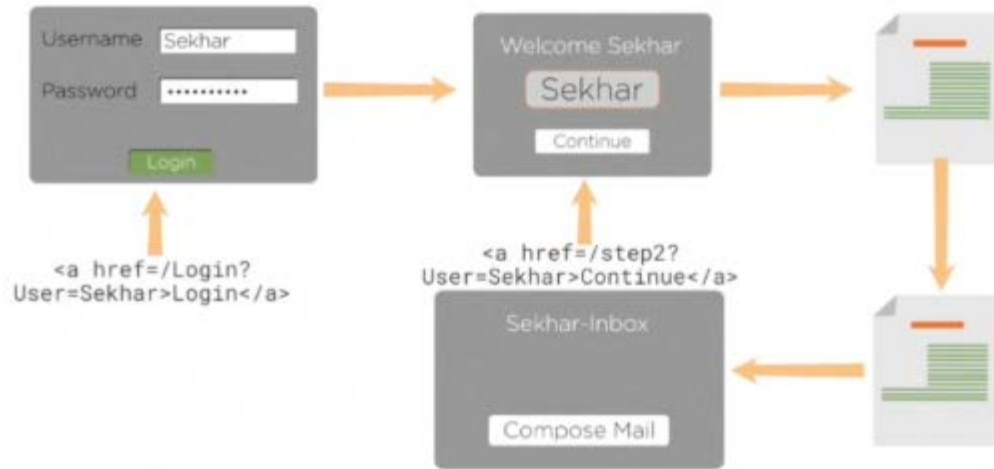
---



# URL Rewriting



# URL Rewriting



## Pros:

Supported by all browsers



Doesn't depend on browser capabilities

## Cons:

Works only with Hyperlinks

Can submit only text information

Complex to implement

# Handling Cookies in Servlets

---

Cookies are used to store the server side information at the client system

## Types Of Cookies

---

Session or Temporary Cookies

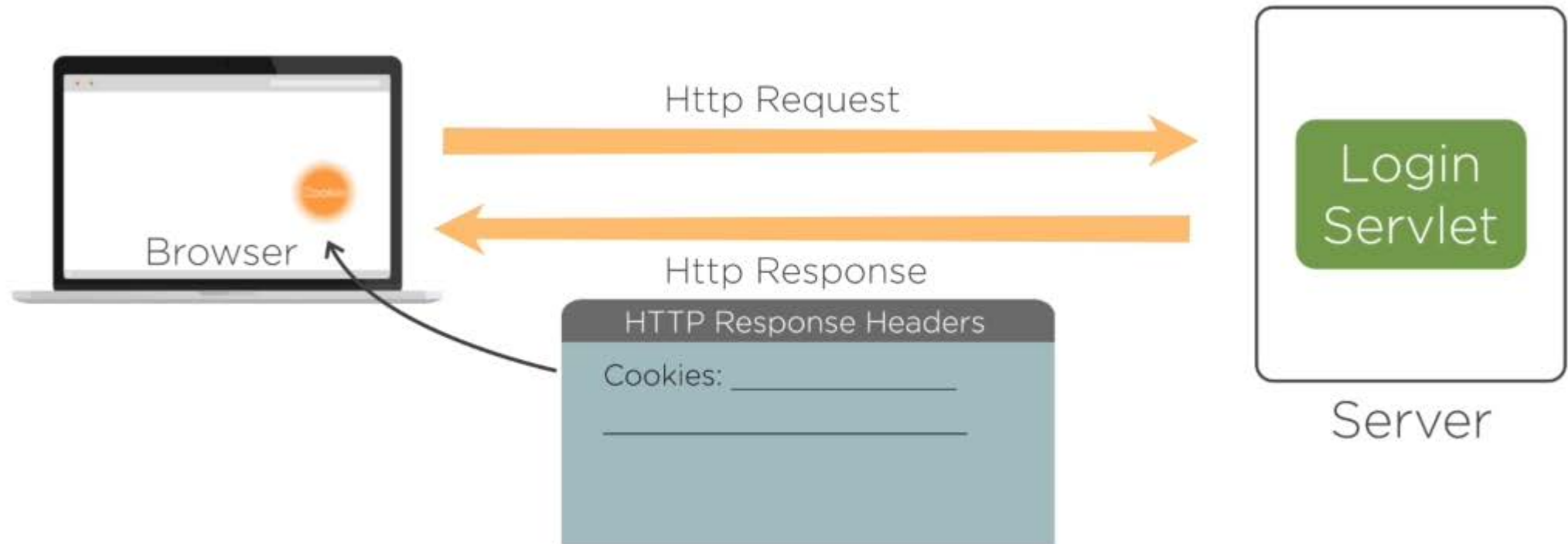
Persistent or Permanent Cookies

# Handling Cookies in Servlets

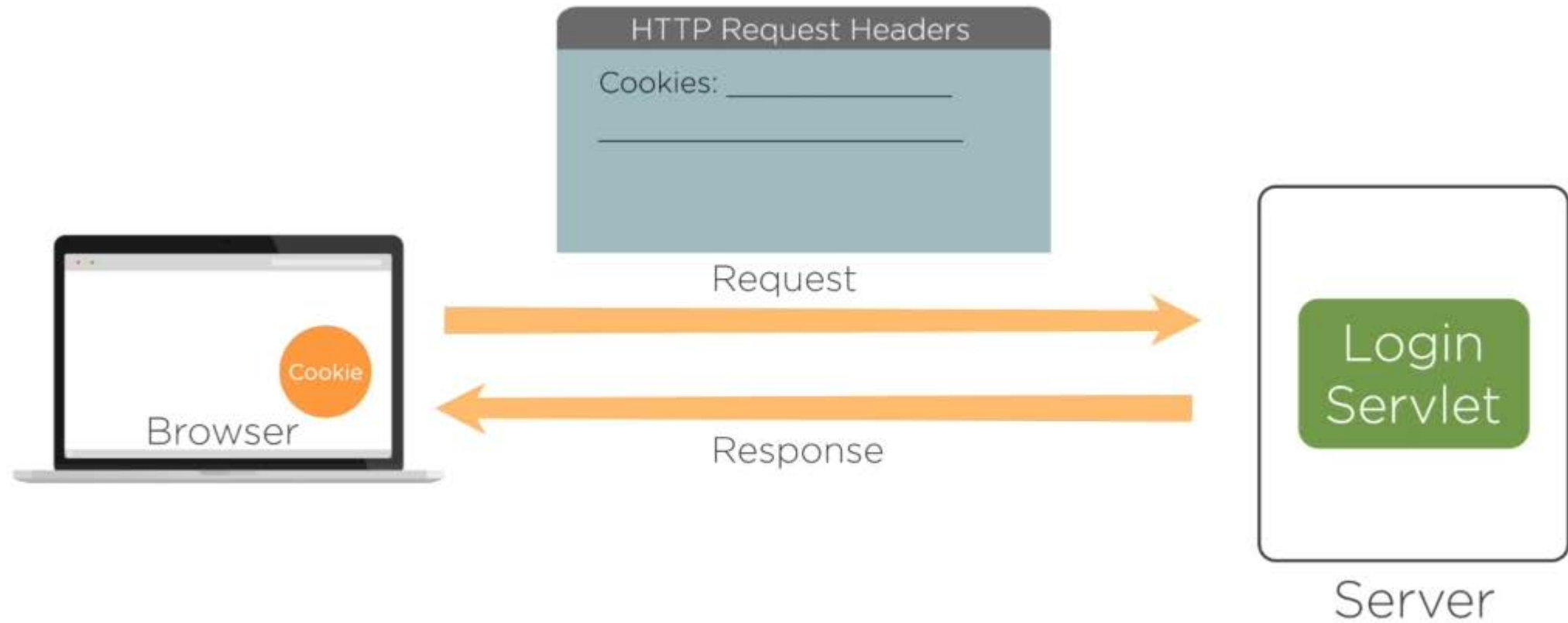




# Handling Cookies in Servlets



# Handling Cookies in Servlets



```
Cookie cookieObject = new Cookie  
("key", "value");
```

◀ To Create an object for the Cookie

```
cookie.setMaxAge(3000);
```

◀ To set the expiration period for the Cookie

```
response.addCookie(cookieObject)
```

◀ To add the Cookie object

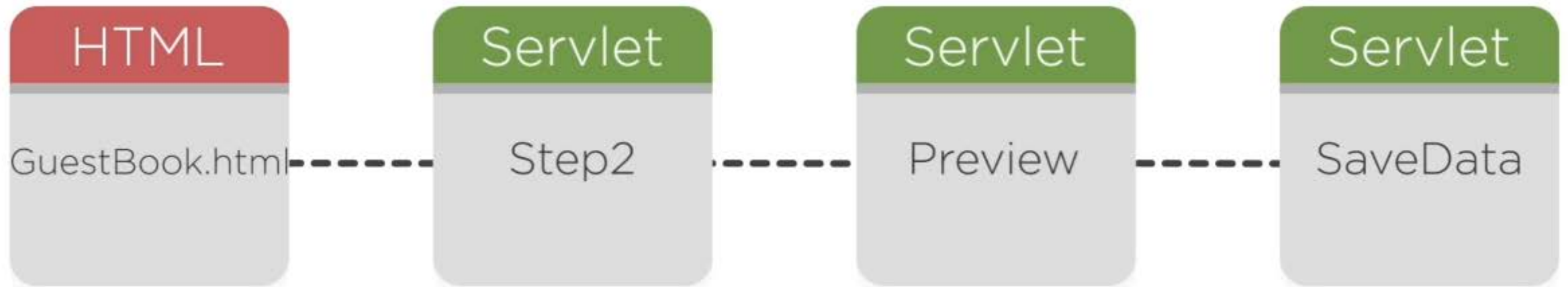
```
request.getCookies()
```

◀ To retrieve the Cookie information

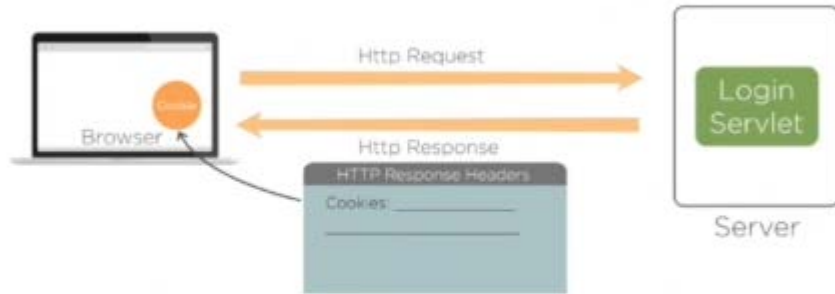
## Methods to Manipulate Cookies

```
setDomain(String pattern)
getDomain()
setMaxAge(int expiry)
getMaxAge()
getName()
setValue(String newValue)
getValue()
setPath(String uri)
getPath()
setSecure(boolean flag)
setComment(String purpose)
getComment()
```

# Demo: Handling Cookies in Servlets



# Handling Cookies in Servlets



## Pros:

Easy to implement

Performance will be fast

Can store huge information for a longer period

## Cons:

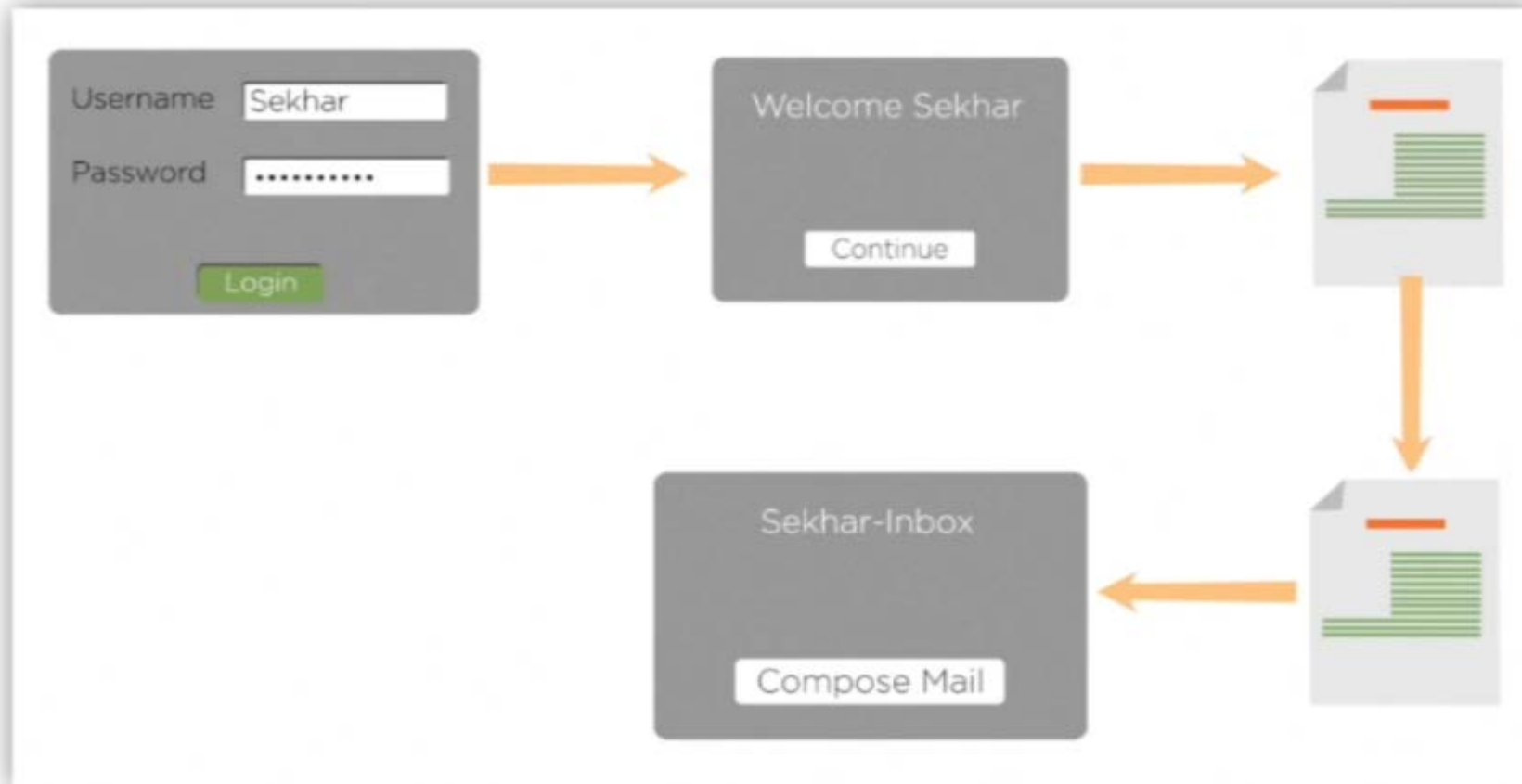
Cookies depends on browser capabilities

Not advisable to store secure data

Can maintain only String information

# Handling Sessions in Servlets

Sessions are used to maintain the user information at the server side temporarily such that the values can be accessed across the Servlets within the web applications



# Handling Sessions in Servlets

```
HttpSession session = request.getSession();
```

Syntax:

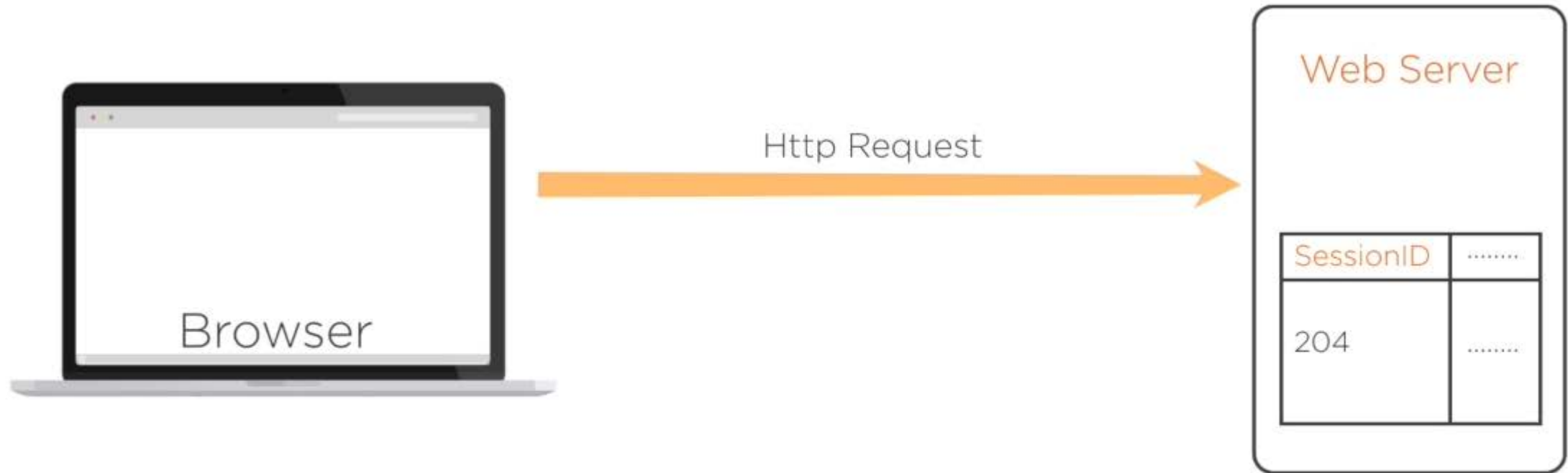
To Create an HttpSession Object



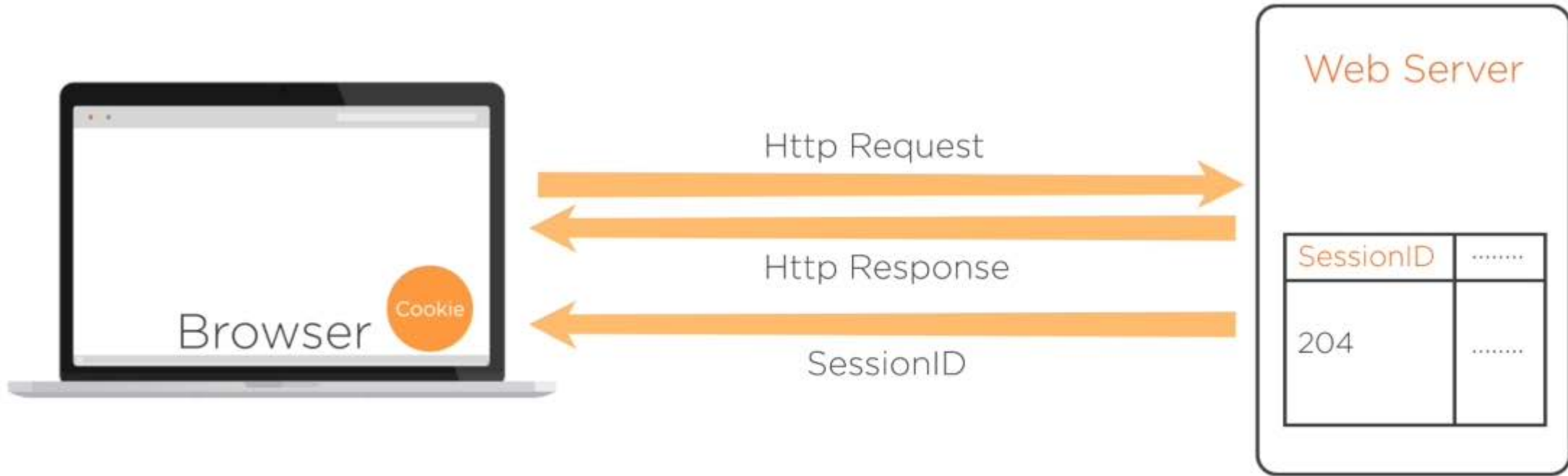
Call `request.getSession()` before we send any content to the client



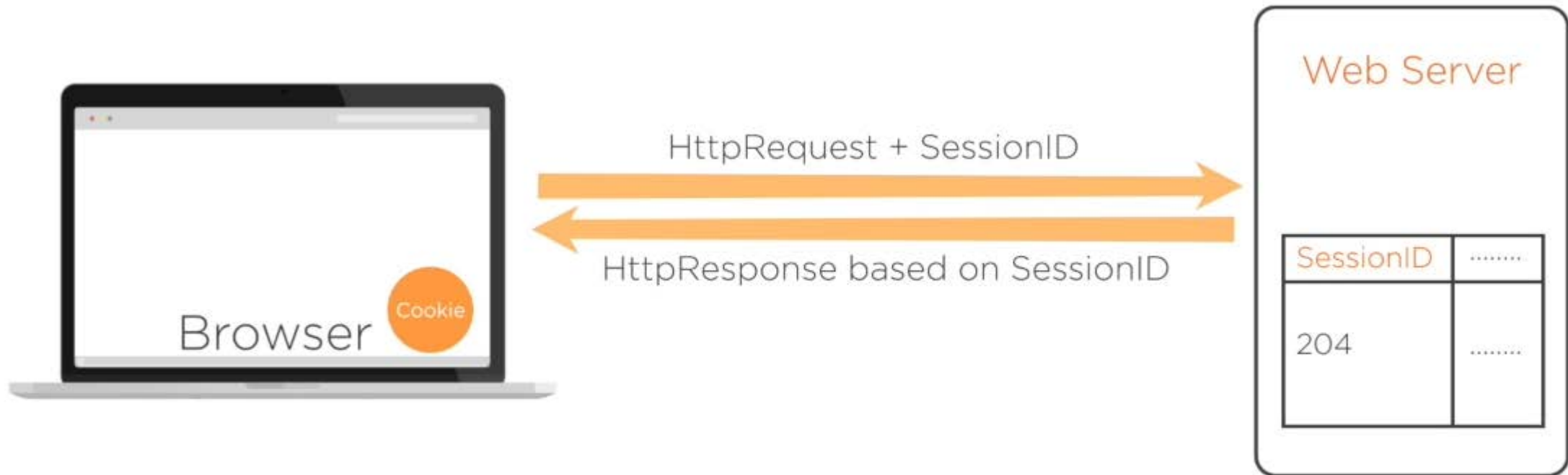
# Handling Sessions in Servlets



# Handling Sessions in Servlets



# Handling Sessions in Servlets



## Methods to Track the Session Data

- getAttribute(String name)
- getAttributeNames()
- getCreationTime()
- getId()
- getLastAccessedTime()
- getMaxInactiveInterval()
- invalidate()
- isNew()
- removeAttribute(String name)
- setAttribute(String name,object value)
- setMaxInactiveInterval(int interval)

Demo



Handling Sessions in Servlets

# Handling Sessions in Servlets



## Pros:

Easy to implement

Can store String and Object type of data as value

Data is secured

## Cons:

Not advisable to store data for long period

Not advisable to store huge data

Sessions depends on browser capabilities

# Summary



Hidden Fields

URL Rewriting

Cookies

Sessions