#### **Unprotected functionality**

Administrative functions might be linked from an administrator's welcome page but not from a user's welcome page. However, a user might simply be able to access the administrative functions by browsing directly to the relevant admin URL.

* **Find administrative URLs**: Burp Intruder (interesting files and directories), **/robots.txt**

The **administrative URL** may be **hidden in the source code** => **Discover content / Find references / Find scripts**

#### **Parameter-based access control methods**

Try changing **access role in hidden fields, cookies, query string parameters**

e.g.,

https://insecure-website.com/login/home.jsp?admin=true

https://insecure-website.com/login/home.jsp?role=1

**Try upgrading access role with POST-Requests** when changing sth (e.g., email) by **adding this parameter** to the request.

#### **Broken access control resulting from platform misconfiguration**

If a web site uses front-end controls to restrict access based on URL, but the application allows the URL to be overridden via a request header (**X-Original-URL**, **X-Rewrite-URL)**, then it might be possible to bypass the access controls using a request like the following:

POST **/** HTTP/1.1

**X-Original-URL: /admin**

...

Some web sites are tolerant of alternate HTTP request methods when performing an action. If an attacker can use the GET (or another) method to perform actions on a restricted URL, then they can circumvent the access control that is implemented at the platform layer.

* Try changing HTTP method GET <=> POST

### **Horizontal privilege escalation**

- Try accessing information of other users/objects:https://insecure-website.com/myaccount?id=123

In some cases, an application does detect when the user is not permitted to access the resource, and returns a redirect to the login page. The **response containing the redirect** **might still include some sensitive data belonging to the targeted user**

### **IDOR vulnerability with direct reference to static files**

Websites might save chat message transcripts to disk using an incrementing filename, and allow users to retrieve these by visiting a URL like the following:https://insecure-website.com/static/12144.txt

* We can simply modify the filename to retrieve a transcript created by another user.

### **Referer-based access control**

Some websites base access controls on the Referer header (last page visited) submitted in the HTTP request. For example, suppose an application enforces access control over the main administrative page at /admin, but for sub-pages such as /admin/deleteUser only inspects the Referer header. If the Referer header contains the main /admin URL, then the request is allowed. We can forge the required Referer header, and so gain unauthorized access.