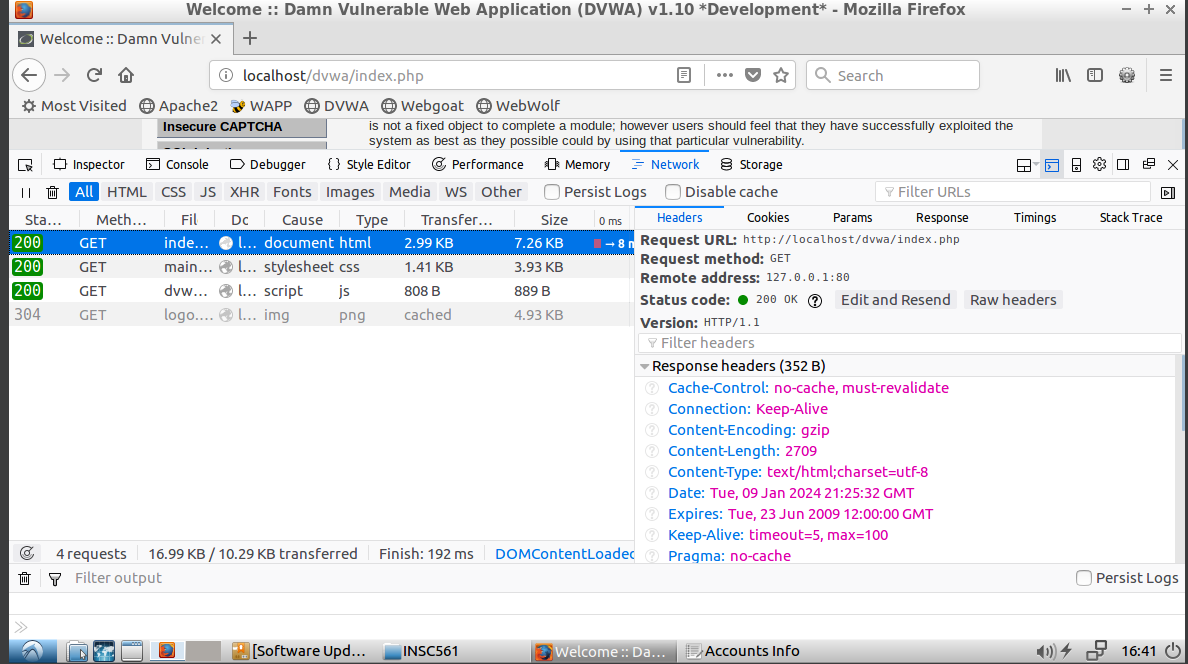
Nick Forleo

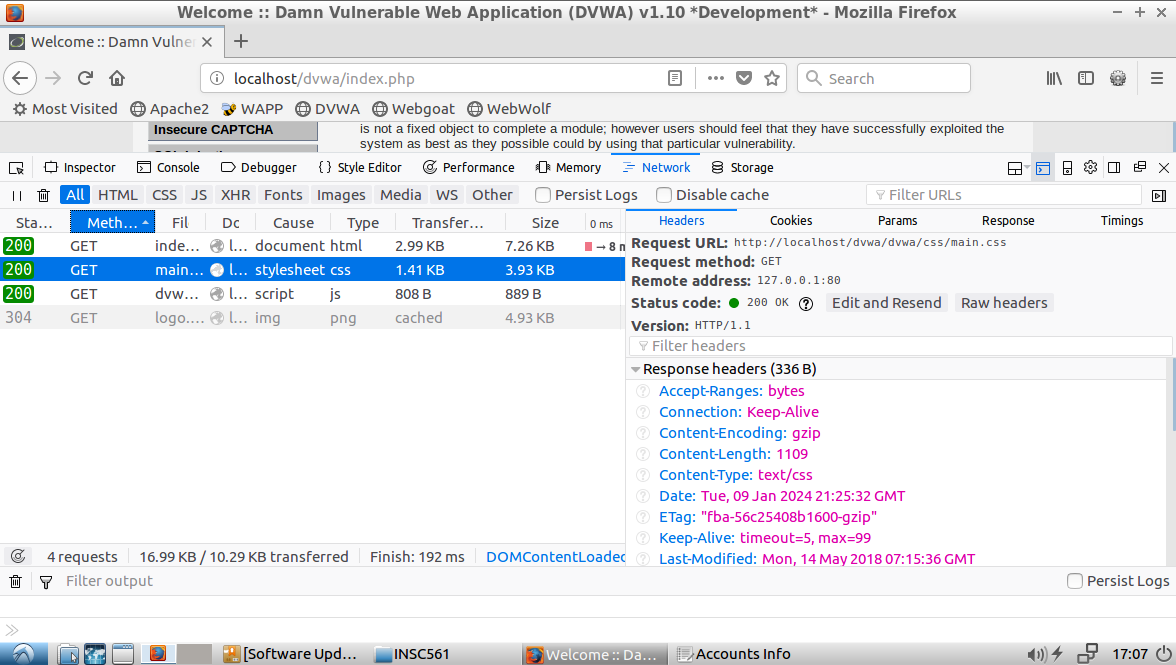
INSC 561

* **Task 1:** HTTP requests and responses (headers) – Find a conversation between the browser and the server; briefly discuss what information is provided in the protocol.



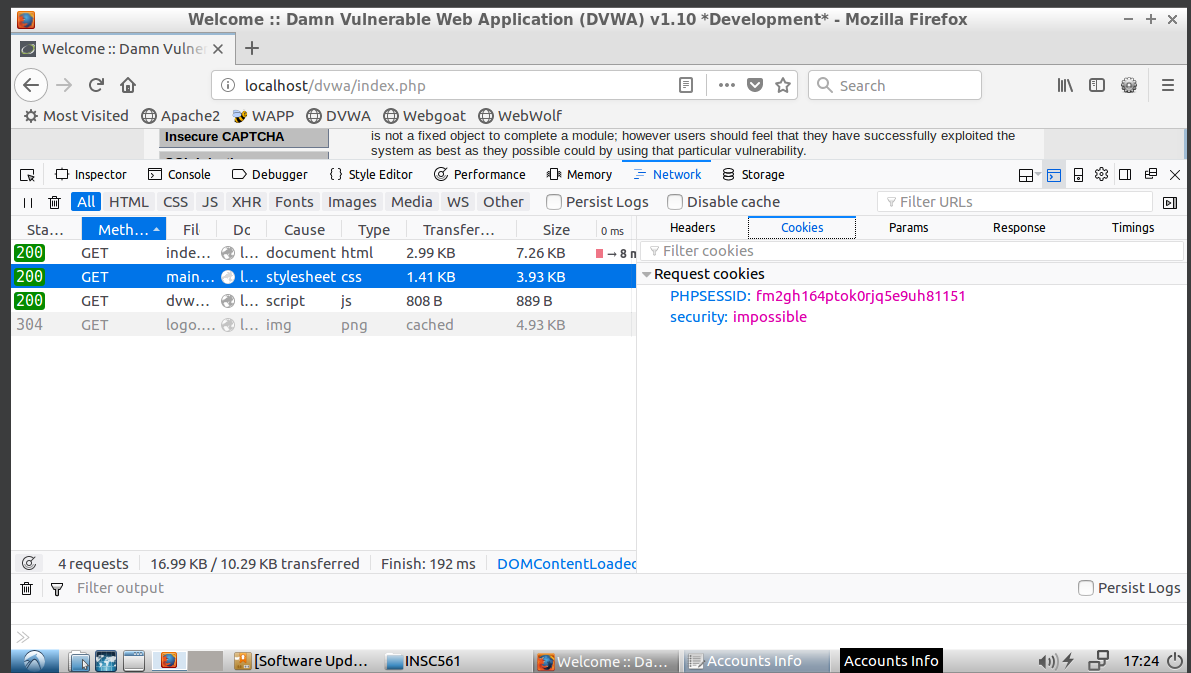
This HTTP request is a GET request. It is retrieving a file of the structure and content of the web page from the server. In this case, the server is hosted locally, as noted by the remote address of 127.0.0.1. The file is called index.php and stored in the /dvwa directory. It is using version 1.1 of HTTP. Various headers are being sent as the response to this request. Some of the notable ones include Cache-Control, Date, and Content-Type. Cache-Control has a value of ‘no-cache’ which means that the response is sending fresh results from the endpoint that it is requesting. The value of Date is when the server sent the response to the request, which is useful metadata to have when trying to debug potential issues or determining the most recent response to a request. Content-Type of value ‘text/html’ is telling the client that the response contains a text file that is formatted in html.

* **Task 2:** HTTP methods – Find at least one HTTP method in a conversation between the browser and the server; briefly discuss what function is provided by the method.



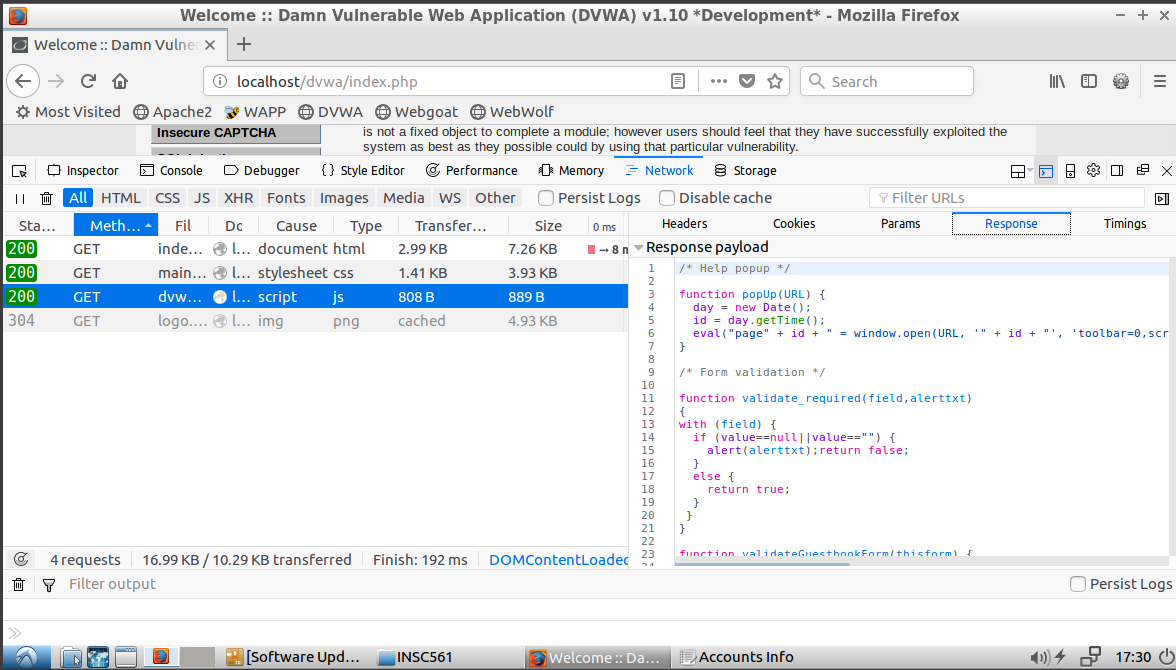
All of the requests sent by this web page are GET requests. GET requests retrieve information from the server. The above GET request returns a style sheet from the server. This file will be loaded locally to style the content that was returned from the first GET request. You can see that the response contains the header Content-Length of value 1109. This means that the CSS file returned from the request contains 1109 bytes.

* **Task 3:** Cookies – Find at least one data in Cookies in a conversation between the browser and the server; briefly discuss what information is delivered in the Cookies.



The Cookies being sent in this request include the keys `PHPSESSID` and `security`. In this case, `PHPSESSID` is most likely a browser session identifier and `security` is some value of an enum that the server understands. The cookies originate from the server and are sent with every request.

* **Task 4:** JavaScript – Find at least one JavaScript module (including Source Code) in a conversation between the browser and the server; briefly discuss what function is enabled by the module.



JS Source Code:

/\* Help popup \*/

function popUp(URL) {

day = new Date();

id = day.getTime();

eval("page" + id + " = window.open(URL, '" + id + "', 'toolbar=0,scrollbars=1,location=0,statusbar=0,menubar=0,resizable=1,width=500,height=300,left = 540,top = 250');");

}

/\* Form validation \*/

function validate\_required(field,alerttxt)

{

with (field) {

if (value==null||value=="") {

alert(alerttxt);return false;

}

else {

return true;

}

}

}

function validateGuestbookForm(thisform) {

with (thisform) {

// Guestbook form

if (validate\_required(txtName,"Name can not be empty.")==false)

{txtName.focus();return false;}

if (validate\_required(mtxMessage,"Message can not be empty.")==false)

{mtxMessage.focus();return false;}

}

}

function confirmClearGuestbook() {

return confirm("Are you sure you want to clear the guestbook?");

}

This request will return JavaScript from the server. The source code includes the following functions (and their signatures): popUp(URL), validate\_required(field, alerttxt), validateGuestbookForm(thisform), and confirmClearGuestBook(). These functions provide some of the functionality to the web page - mostly validation.