HTTP Overview

Computer Science
NC STATE UNIVERSITY

URL: The Starting Point

scheme://user:password@domain:port/path?query_string#fragment_id

scheme: name of the protocol

user:password: credentials

domain: destination/server

port: which network port to communicate

path: hierarchical path. May be case-sensitive

query_string: pass arbitrary parameters to the server

fragment: part/position within the resource / document

Source: RFC 3896

HTTP Requests / Responses

Both types have:

- an initial line
- zero or more header lines
- a blank line
- an optional message body

RFC 2616 (obsolete), RFCs: 7230-7237

HTTP Request

Request Line

```
GET /path/to/file/index.html HTTP/1.1
```

Header Lines

```
HeaderName: value1{, value2}
```

HTTP 1.1 defines 46 headers. Only "Host:" required

HTTP Response

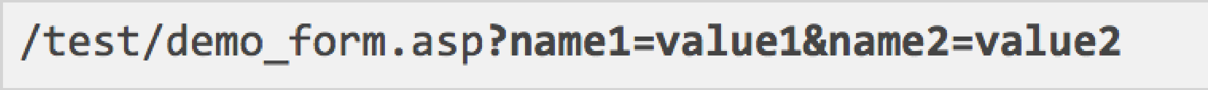
Initial Response (Status) Line

```
HTTP/1.1 200 OK
```

- The HTTP version is in the same format as in the request line, "**HTTP/x.x**".
- The status code is meant to be computer-readable; the reason phrase is meant to be human-readable, and may vary.
- Status Codes
  - **1xx** indicates an informational message only
  - **2xx** indicates success of some kind
  - **3xx** redirects the client to another URL
  - **4xx** indicates an error on the client's part
  - **5xx** indicates an error on the server's part

HTTP Requests: GET vs POST

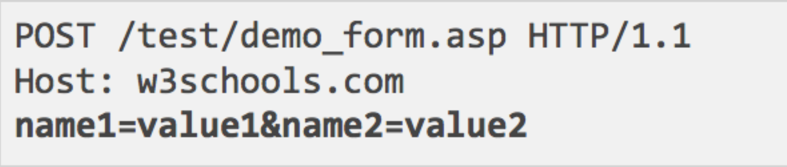
- GET:



```
/test/demo_form.asp?name1=value1&name2=value2
```

  - Requests data from a specified resource; form data sent as part of the URL
    - GET requests can be cached
    - GET requests remain in the browser history
    - GET requests can be bookmarked
    - GET requests should never be used when dealing with sensitive data
    - GET requests have length restrictions
    - GET requests should be used only to retrieve data

- POST:



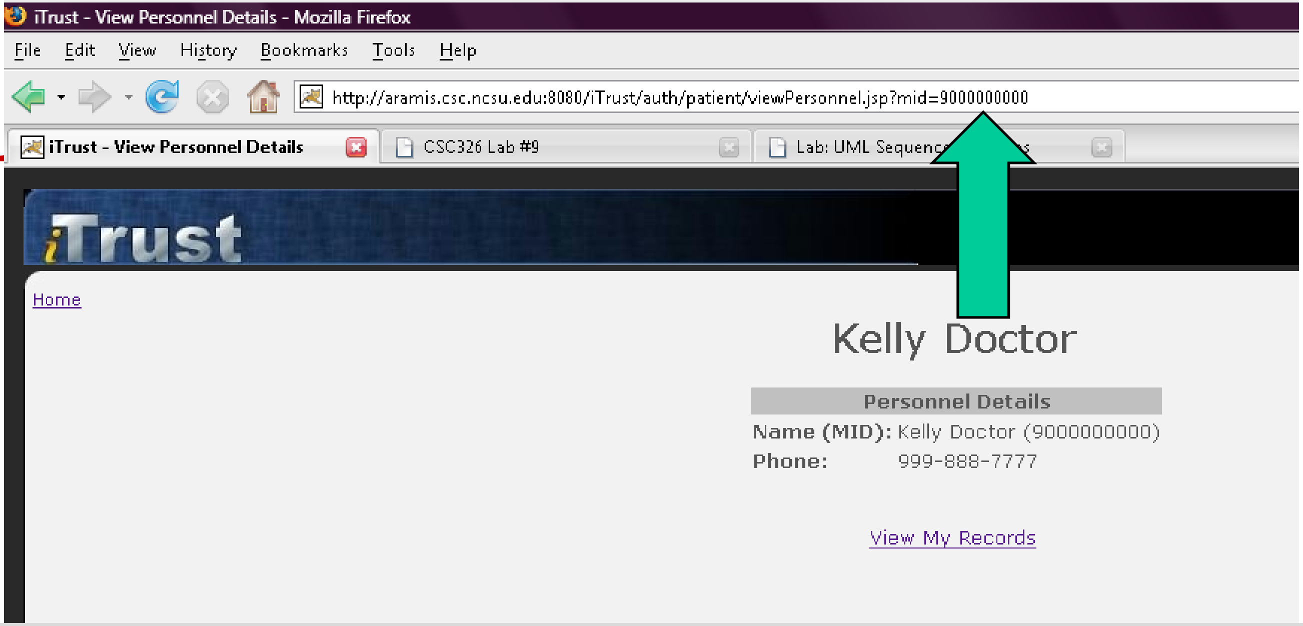
```
POST /test/demo_form.asp HTTP/1.1
Host: w3schools.com
name1=value1&name2=value2
```

  - form data sent within message body
    - POST requests are never cached
    - POST requests do not remain in the browser history
    - POST requests cannot be bookmarked
    - POST requests have no restrictions on data length

http://www.w3schools.com/tags/ref_httpmethods.asp

Computer Science
NC STATE UNIVERSITY

URL



iTrust - View Personnel Details - Mozilla Firefox

http://aramis.csc.ncsu.edu:8080/iTrust/auth/patient/viewPersonnel.jsp?mid=9000000000

Kelly Doctor

Personnel Details

Name (MID): Kelly Doctor (9000000000)

Phone: 999-888-7777

View My Records

- There are two reasons why a parameter should not in the URL
  - The parameter is one the user should not be able to set the value of.
  - The parameter is one the user should not be able to see the value of.

Comments?

Computer Science NC STATE UNIVERSITY

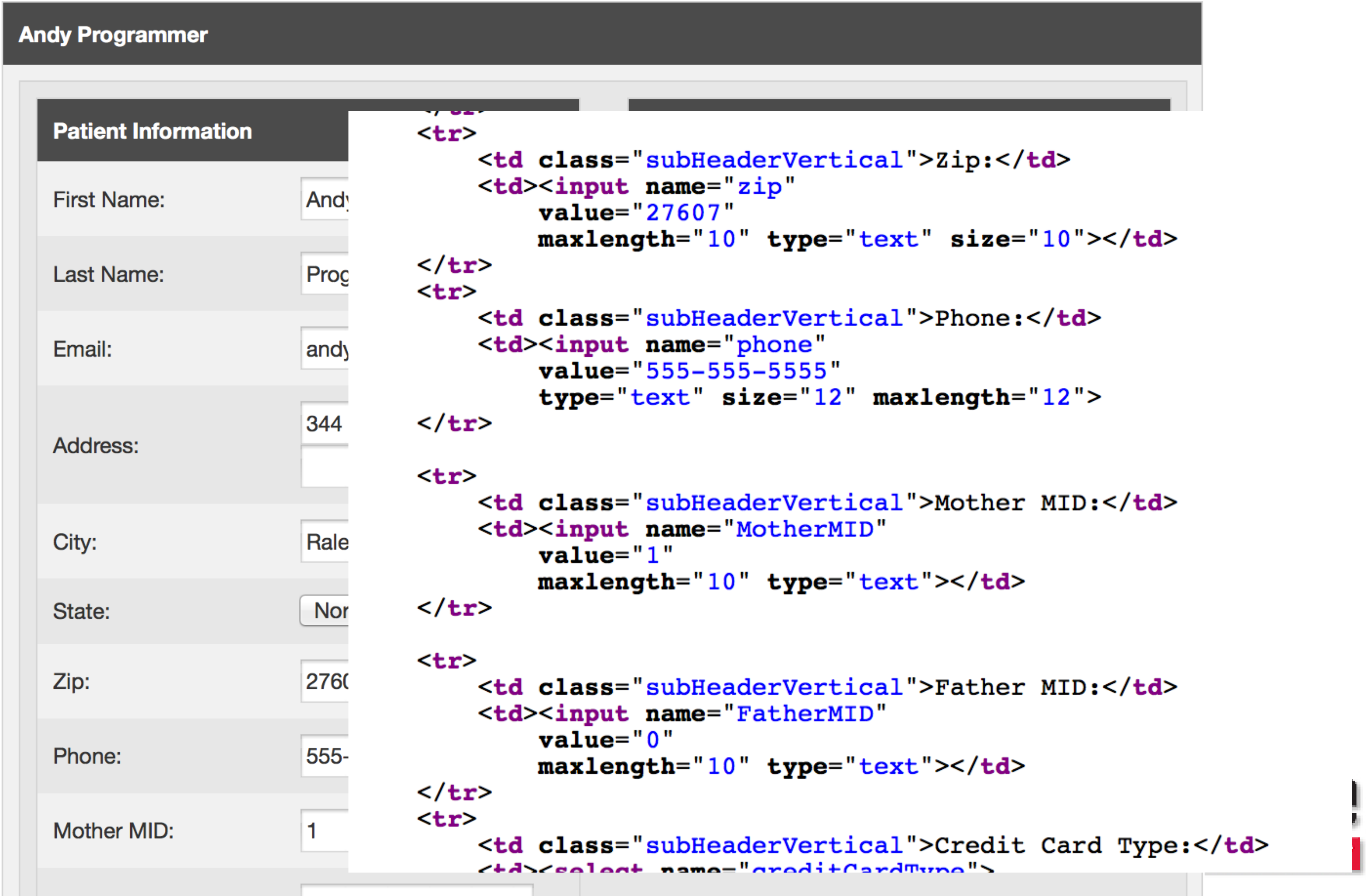
See URL manipulation: http://www.cgisecurity.com/owasp/html/ch11s04.html#id2863856

Hidden Variables

- `<input name=“MID" type="hidden" value=“90000000001">`
- `<input name="masteraccess" type="hidden" value=“Y">`

- However, a malicious user can save the page; obtain and/or change the `MID` or `masteraccess;` and reload the page in his/her browser.

HTML is always editable!



Andy Programmer

Patient Information

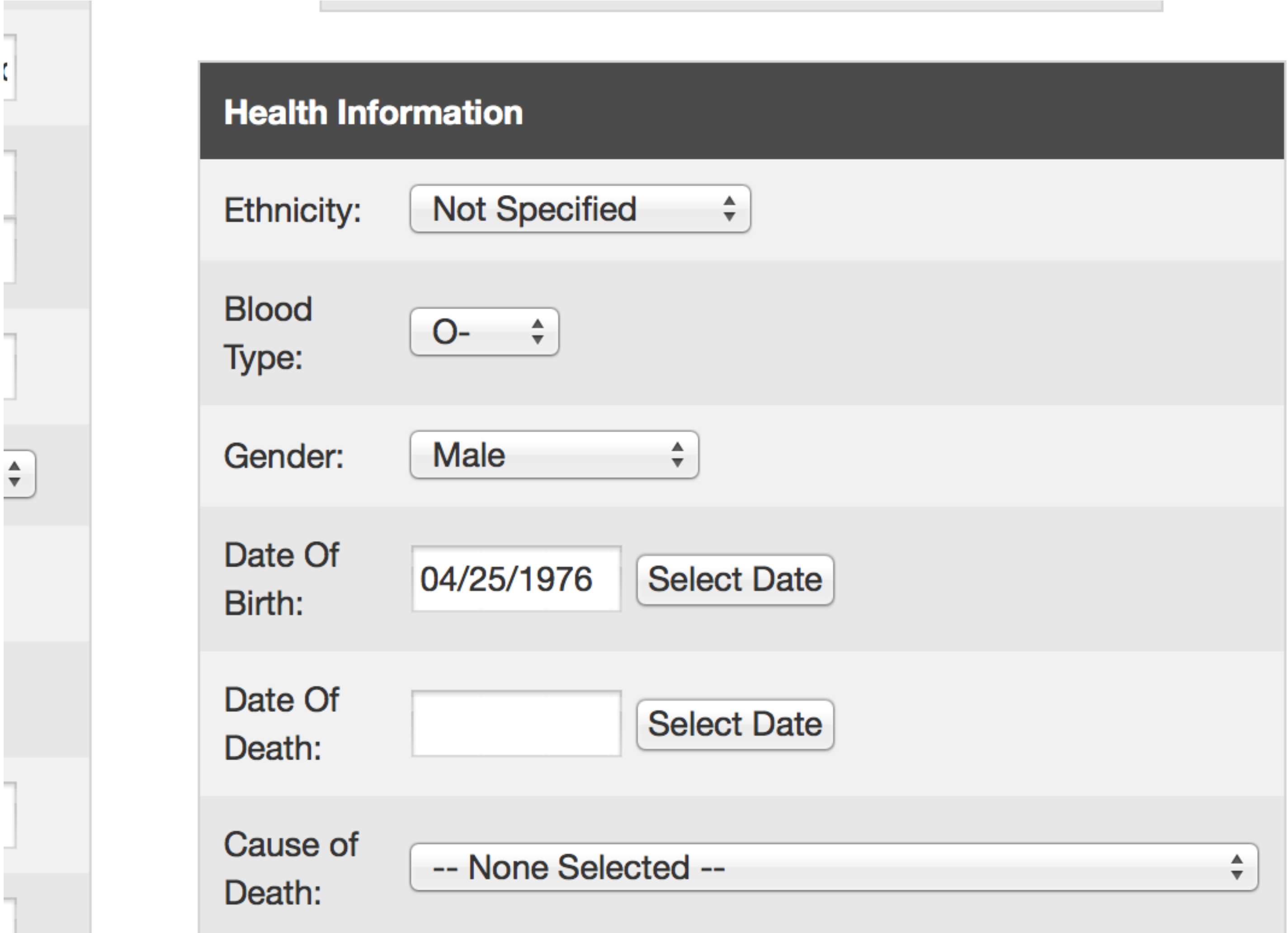
First Name:	Andy
Last Name:	Prog
Email:	andy
Address:	344
City:	Rale
State:	Nor
Zip:	2760
Phone:	555-
Mother MID:	1

```
    </tr>
    <tr>
        <td class="subHeaderVertical">Zip:</td>
        <td><input name="zip"
            value="27607"
            maxlength="10" type="text" size="10"></td>
    </tr>
    <tr>
        <td class="subHeaderVertical">Phone:</td>
        <td><input name="phone"
            value="555-555-5555"
            type="text" size="12" maxlength="12">
    </tr>

    <tr>
        <td class="subHeaderVertical">Mother MID:</td>
        <td><input name="MotherMID"
            value="1"
            maxlength="10" type="text"></td>
    </tr>

    <tr>
        <td class="subHeaderVertical">Father MID:</td>
        <td><input name="FatherMID"
            value="0"
            maxlength="10" type="text"></td>
    </tr>
    <tr>
        <td class="subHeaderVertical">Credit Card Type:</td>
        <td><select name="creditCardType">
```

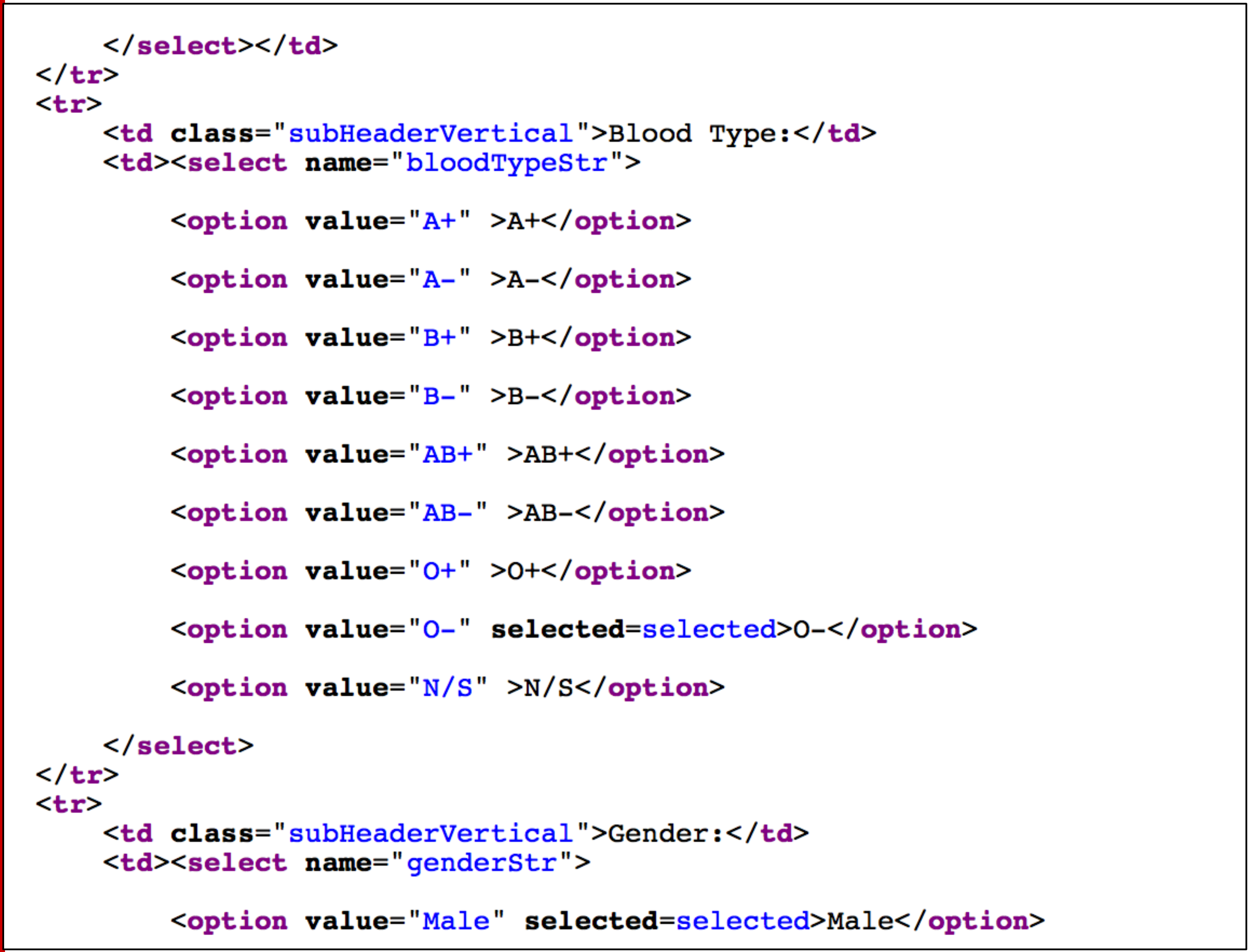
Limited value of GUI controls on user input …



Health Information

Ethnicity:	Not Specified	
Blood Type:	O-	
Gender:	Male	
Date Of Birth:	04/25/1976	Select Date
Date Of Death:		Select Date
Cause of Death:	-- None Selected --	

With client side validation only, you can do this …



```
    </select></td>
</tr>
<tr>
    <td class="subHeaderVertical">Blood Type:</td>
    <td><select name="bloodTypeStr">

        <option value="A+" >A+</option>

        <option value="A-" >A-</option>

        <option value="B+" >B+</option>

        <option value="B-" >B-</option>

        <option value="AB+" >AB+</option>

        <option value="AB-" >AB-</option>

        <option value="O+" >O+</option>

        <option value="O-" selected=selected>O-</option>

        <option value="N/S" >N/S</option>

    </select>
</tr>
<tr>
    <td class="subHeaderVertical">Gender:</td>
    <td><select name="genderStr">

        <option value="Male" selected=selected>Male</option>
```

Bypass Client-side Validation

- Check on server side, even if checks done on client side (such as with JavaScript)
  - Can be bypassed
  - Can be modified in transit
    - Paros: http://www.parosproxy.org/index.shtml
    - Webscarab: http://www.owasp.org
    - ZAP: https://www.owasp.org/index.php/OWASP_Zed_Attack_Proxy_Project

HTTP: Use Developer Tools