

Using Cookies to Track User

Qiang Yang, SFU

Thanks: E. Cerami, NYU

1/22/01 Cookies 1

Purpose: Personalization

- Sites may know who you are...
 - www.my.yahoo.com
 - www.amazon.com
 - Cannot be achieved by server logs:

```
127.0.0.1 - - [21/Jan/2001:14:29:02 -0800] "GET /servlet/Hello470Parameters HTTP/1.1" 200 201
127.0.0.1 - - [21/Jan/2001:14:29:17 -0800] "GET /servlet/Hello470HTML HTTP/1.1" 404 299
127.0.0.1 - - [21/Jan/2001:14:29:27 -0800] "GET /servlet/Hello470882HTML HTTP/1.1" 200 109
127.0.0.1 - - [21/Jan/2001:14:30:20 -0800] "GET /servlet/Hello470Parameters.HTML HTTP/1.1" 404 310
127.0.0.1 - - [21/Jan/2001:14:30:27 -0800] "GET /servlet/Hello470Parameters.html HTTP/1.1" 404 310
```

1/22/01 Cookies 2

What is a Cookie?

- Magical cookies: connecting two programs via a text file
- A text file, stored on the client's hard drive.
 - Included within the official HTTP specification.
 - Controversial**, as it enables web sites to track web users and their habits
- Cookie Content
 - Name/value pairs
 - Passed as part of HTTP header
 - Usually unsecure unless encrypted

1/22/01 Cookies 3

Cookies in Action

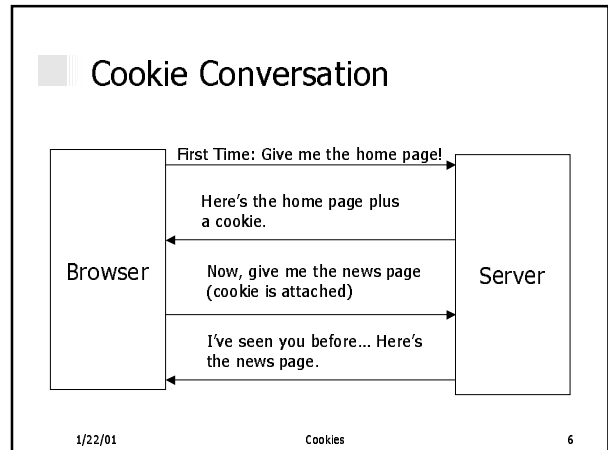
- <http://www.cookiecentral.com>
 - The best web resource for cookie information.
 - Contains a detailed Cookie FAQ
 - Contains lots of example Cookie applications.
 - <http://www.cookiecentral.com/jc2.htm>
 - <http://www.cookiecentral.com/eg1.htm>

1/22/01 Cookies 4

Where are the cookies

- Netscape: a single file:** Program Files \Netscape\Users\qyang\...
- IE: one for each domain:** WinNT\Profiles\Username\Cookies\...
- Created in 1994 for Netscape 1.1
- Cookies cannot be larger than 4K
- No domain (netscape.com, microsoft.com) can have more than 20 cookies.
- Cookies stay on your machine until:
 - they automatically expire
 - they are explicitly deleted
- Cookies work the same on all browsers. No cross-browser problems here!

1/22/01 Cookies 5



Cookie Anatomy: Six Parts in Version 0

- Name: cookie name
 - Example: "Location"
- Value: value of cookie
 - Example: "VancouverBCCanada"
 - No commas, white spaces, etc
- Domain: domain that generated cookie
 - Example: "www.amazon.com"
- Path: the page that created the cookie; restricts cookie within site so no other sites can use the cookie
- Expires: Wdy DD-Mon-YYYY HH:MM:SS GMT
- Secure: flag; (*Not used – don't trust it!*)

1/22/01

Cookies

7

How Ad companies track Users

- AD.com has a database (DB) of users names and purchasing behavior
- AD.com also has a unique ID for each online user
- Question: how to correlate between DB and ID?
- Question: how to know users browsing behavior?

1/22/01

Cookies

8

The DoubleClick Controversy

- DoubleClick: one of the largest providers of on-line ads.
- In November 1999, DoubleClick acquired a data company, Abacus Direct.
 - Abacus keeps a database of names, addresses and offline buying habits of 90 million households - direct mail and larger retailers.
- DoubleClick proposed to combine all
 - All names, addresses, behaviors (purchasing and browsing)
- Huge controversy – finally retreated.

1/22/01

Cookies

9

Creating, Setting and Adding

- Creating cookies
Cookie school = new Cookie ("school", "SFU");
- Set cookie parameters
school.setMaxAge (3600);
- Send cookies back to browser
response.addCookie (school);

1/22/01

Cookies

10

Creating and Reading Cookies

- HttpServletRequest has a getCookies() method: returns an array of cookie objects.

```
Cookie[] cookies = req.getCookies();
for (int i=0; i<cookies.length; i++) {
    cookie = cookies[i];
    name = cookie.getName();
    if (name.equals("counter"))
        value = cookie.getValue();
}
```

1/22/01

Cookies

11

```
import java.io.*;
import javax.servlet.*;
import javax.servlet.http.*;

public class CookieCounter extends HttpServlet {

    public void doGet(HttpServletRequest req,
        HttpServletResponse res)
        throws ServletException, IOException {
        String name, value = null;
        Cookie cookie;
        int counter;
        res.setContentType("text/html");

        // Try to extract the counter cookie (if one exists)
        Cookie[] cookies = req.getCookies();
        for (int i=0; i<cookies.length; i++) {
            cookie = cookies[i];
            name = cookie.getName();
            if (name.equals("counter"))
                value = cookie.getValue();
        }
    }
}
```

1/22/01

Cookies

12

```

// If possible, parse the counter value
// Otherwise, start over at 0.
if (value != null)
    counter = Integer.parseInt (value);
else counter = 0;

// Increment the counter
counter++;

// Create a new counter cookie
// Cookie will exist for one year
cookie = new Cookie ("counter", Integer.toString(counter));
cookie.setMaxAge (60*60*24*365);
res.addCookie (cookie);

// Output number of visits
PrintWriter out = res.getWriter();
out.println ("<HTML><BODY>");
out.println ("<H1>Number of visits:  "+counter);
out.println ("</H1>");
out.println ("</BODY></HTML>");
out.close();
}
1/22/01          Cookies          13

```

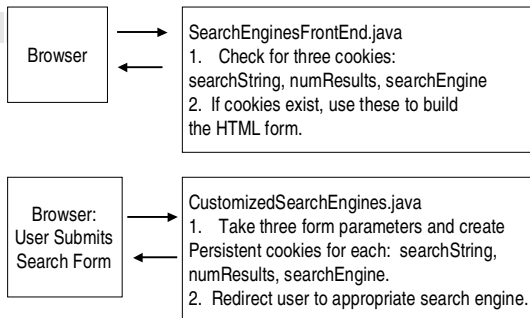
Search Engines that remember

- When the user issues a search request, the servlet remembers the request.
- When you go back to do another search, the form is pre-populated with your previous request.
- It works by using three persistent cookies:
 - searchString: the last search strings
 - numResults: limit on the returned results
 - searchEngine: which engine was used
- We do not have time to discuss the code in class. But, we will go over the high level architecture.

1/22/01

Cookies

14



1/22/01

Cookies

15