

Cookies / Content / Security

Cookie Dynamics

Put cookie on HTTP response Before anything else in code

You need to put the cookie on the response BEFORE it is sent back to the client. Therefore, do cookie writing before anything else in your servlet

setMaxAge

-1 (the default) = until the user exits the browser
otherwise # seconds until cookie deleted from client side

Read Cookie From request

Cookie Code

```
// HelloCookie.java
import java.io.*;
import java.util.*;
import javax.servlet.*;
import javax.servlet.http.*;

public class HelloCookie extends HttpServlet
{
    public void doGet(HttpServletRequest request,
                      HttpServletResponse response)
        throws IOException, ServletException
    {
        String param = null;
        String color = null;
        Cookie cookie = null;

        // Do cookie operations BEFORE anything else

        // Check for color form param --> make cookie
        param = request.getParameter("newcolor");
        if (param!=null && !param.equals("")) {
            color = param;
            cookie = new Cookie("color", color);

            String delay = request.getParameter("delay");
            if (delay!=null && !delay.equals("")) {
                cookie.setMaxAge(Integer.parseInt(delay));
            }

            // Store the cookie back to client on the response
```

```

        response.addCookie(cookie);
    }
    else { // see if there's a cookie on the client request

        cookie = findCookie(request, "color");
        if (cookie != null) color = cookie.getValue();
    }

    response.setContentType("text/html");
    PrintWriter out = response.getWriter();

    //System.out.println("Hello world is leaving the building...");

    out.println("<html><body>");
    if (color!=null) out.println(" bgcolor=\"\" + color + "\"");
    out.println(">");
    out.println("<h1>Hello World</h1>");

    if (color!=null) out.println("Color: " + color);

    out.println("<hr><form>Color: <input type=text name=newcolor>");
    out.println("delay:<input type=text name=delay>");
    out.println("<input type=submit value=Submit></form>");

    out.println("</body></html>");
}

public Cookie findCookie(HttpServletRequest request, String name) {
    Cookie[] cookies = request.getCookies();

    for (int i=0; i<cookies.length; i++) {
        if (cookies[i].getName().equals(name)) return(cookies[i]);
    }

    return(null);
}
}

```

Web Publishing

Cheap Publishing

Consumer initiated

unlike TV, newspaper

Results

Higher volume
Higher diversity
Range of quality

••\$ Flow

1. Free, voluntary content

Info you want to give away and other people want to consume. e.g. Caltrain schedule
The Internet works great for this

2. Advertising supported content

TV

TV, newspaper, Bride magazines -- advertisements delivered with content

Inefficient

If you follow the \$'s, this is a pretty inefficient path
Show the ads to the wrong people too much
Such waste ultimately hurts us all

3. Pay-the-artist

Efficient

Potentially very efficient -- the artist would get a lot more of the \$ compared to (2) above

Anti-piracy

Depends on anti-piracy technology for the symbiosis to work

Micro-payment

May depend on "micro-payment" tech (like \$0.01 transfer) -- does not yet exist

Why Piracy and Rent Control Don't work

Symbiosis

2-way relationship

Each provides something the other wants
Sheryl Crow <-> Consumer
Landlord <-> Tenant

Right Model:

Trees (O₂) <-> Animals (CO₂)

Wrong Model:

Basketball

Decision Error

Wrong: hurt one side to help the other

Works in basketball

Does not work in a symbiosis

e.g. cut down all the trees

e.g. outlaw collecting royalties

e.g. restrict rents, tax landlords ...

Opposite result of what you wanted

- Editing Gap

More Content

Lack of quality control

New York Times

Not writing, but quality assurance

Want: content + good quality

e.g. Search engines

e.g. Yahoo ratings

e.g. edu content

Lots of it, but you can't tell what's good

• • "Attention Scarce"

Info tech -> lots of content

Lots of ads

Attention rare/expensive/hunted

"Scarce Eyeballs"

Having "eyeballs" is valuable

Content Sites have eyeballs

e.g. yahoo

e.g. google

e.g. New York Times

eyeballs = valuable

eyeballs + info about interests = more valuable

1. Consumers Realize This

Regular people sense that their attention is becoming more scarce

1. Better TV ads

2. Leave unappealing sites

2. Lame Corporate Site

Re-package bland, marketing half-truths

3. Good Corporate Site

1. Useful / Truthful

e.g. PDFs of all manuals

e.g. truthful troubleshooting database

2. Appealing

Pictures, interviews, real info on product development

The consumer is not a moron

Cluetrain Manifesto

cluetrain.com

Most corporations, on the other hand, only know how to talk in the soothing, humorless monotone of the mission statement, marketing brochure, and your-call-is-important-to-us busy signal. Same old tone, same old lies. No wonder networked markets have no respect for companies unable or unwilling to speak as they do.

••Security

Problem : Secrecy

Password sniffing
Credit card # sniffing

Solution: Encryption

1. Traditional "Shared Secret"

Both have secret key, k_1
Encrypt w/ k_1 , Send, Decrypt w/ k_1
Problem: key distribution

2. New "Public Key"

1. Bring up insecure connection
2. Use public key / key exchange

Public Key technology allows you to set up encryption for the dialog without a previous shared secret.

It's an amazing feat of mathematics that this can be done

"Public Key" Technology

One of the most amazing developments in applied mathematics in recent memory. Instead of one, secret key, there are two keys which go together as a pair: one "public" and one "private"

Original credit goes to Diffie and Hellman Stanford

Two Keys

Property #1: A message can be encoded with one key and decoded with the other. Either key may be used to encode, and the other key decodes.

Public vs. Private

someone can know the public key, but knowing that will not allow them to deduce/compute what the private key is.

Applications

1. Security

Secure communications works as before, but the public key can be published. I can send secure mail to someone without first arranging a secret key-- I just look up their public key knowing that their private key will decode the message. Like secret key encryption, but without the hassle of managing a secret key.

2. Digital signatures

Person X, encodes an agreed on text using their private key and sends it out or attaches it to a document. Anyone can apply X's public key to decode the signature. *If it comes out right, it could only have come from X.*

Message digest -- Authenticity / tamper-proof messages. Take message, compute a "digest" which is a function of all the characters in the document. Sign the digest as above. Anyone can decode the digest and verify that it matches the document. This verifies who the document is from and that it has not been changed. Example: this is what you need to have digital checks.