

Server-Side Web Programming: Python (Part 4)

Copyright © 2026 by
Robert M. Dondero, Ph.D.
Princeton University

Objectives

- We will cover:
 - Stateful web programming
 - Cookies
 - Web programming in COS 333

Agenda

- **Stateful web programming**
- Cookies
- Web programming in COS 333

Stateful Web Programming

- **Problem:**
 - HTTP is a *stateless* protocol
 - Neither the browser nor the HTTP server nor the application remembers previous interactions

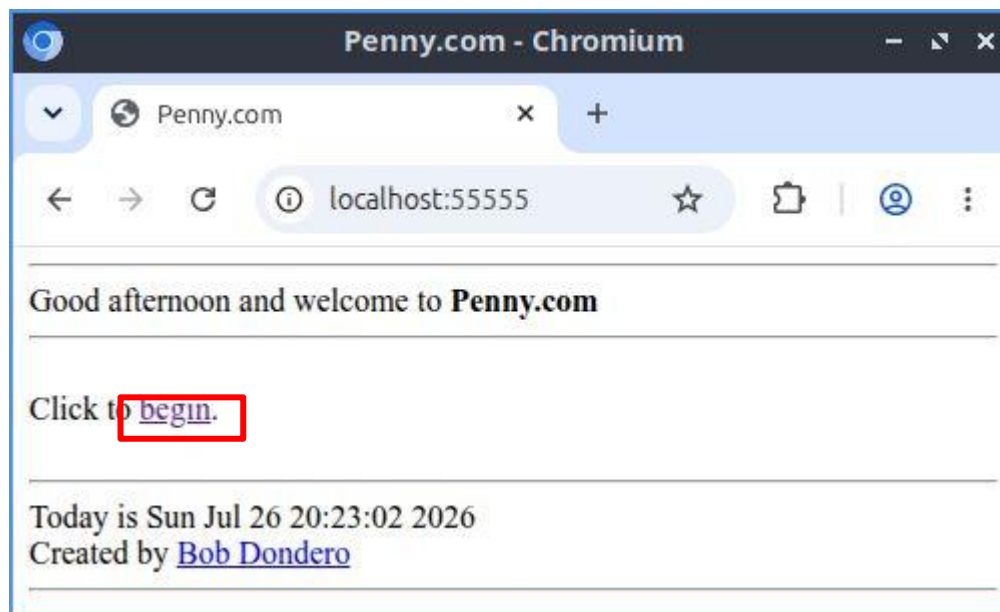
Stateful Web Programming

- See **PennyFlaskJinjaState** app

```
$ python runserver.py 55555
* Serving Flask app 'penny'
* Debug mode: on
WARNING: This is a development server. Do not use
it in a production deployment. Use a production
WSGI server instead.
* Running on all addresses (0.0.0.0)
* Running on http://127.0.0.1:55555
* Running on http://192.168.1.22:55555
Press CTRL+C to quit
* Restarting with stat
* Debugger is active!
* Debugger PIN: 129-723-306
```

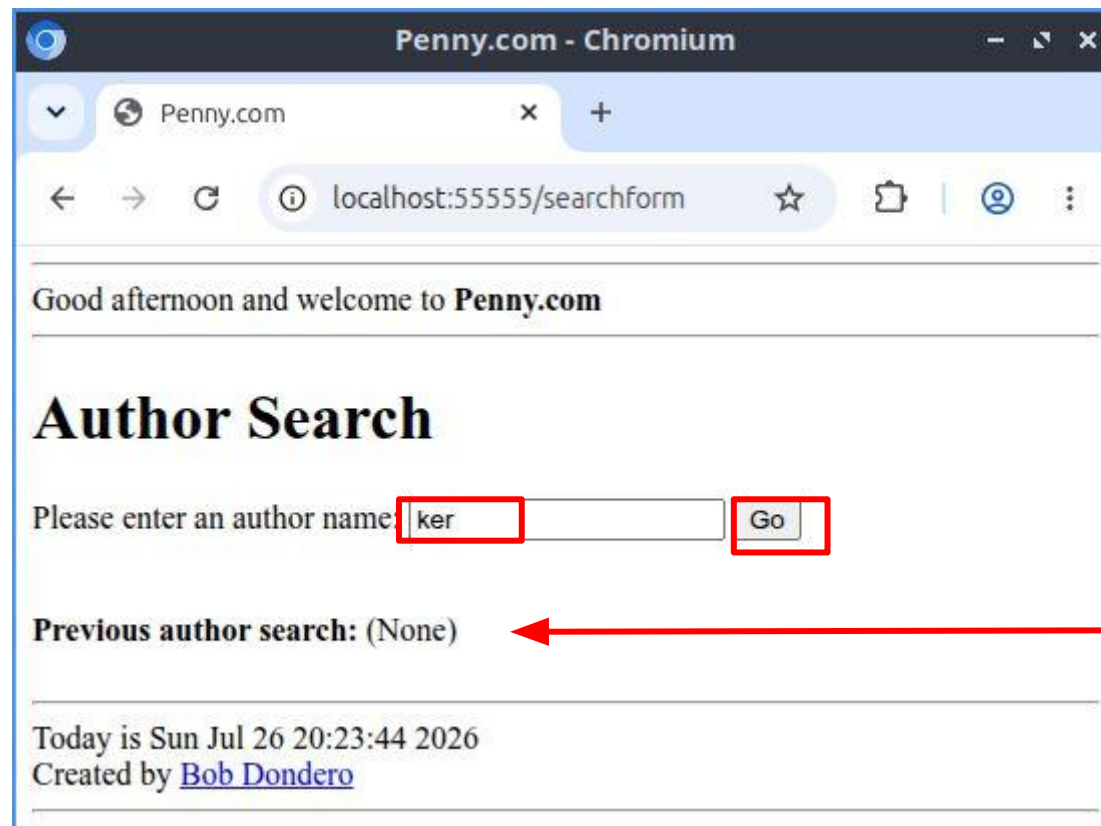
Stateful Web Programming

- See **PennyFlaskJinjaState** app (cont.)



Stateful Web Programming

- See **PennyFlaskJinjaState** app (cont.)

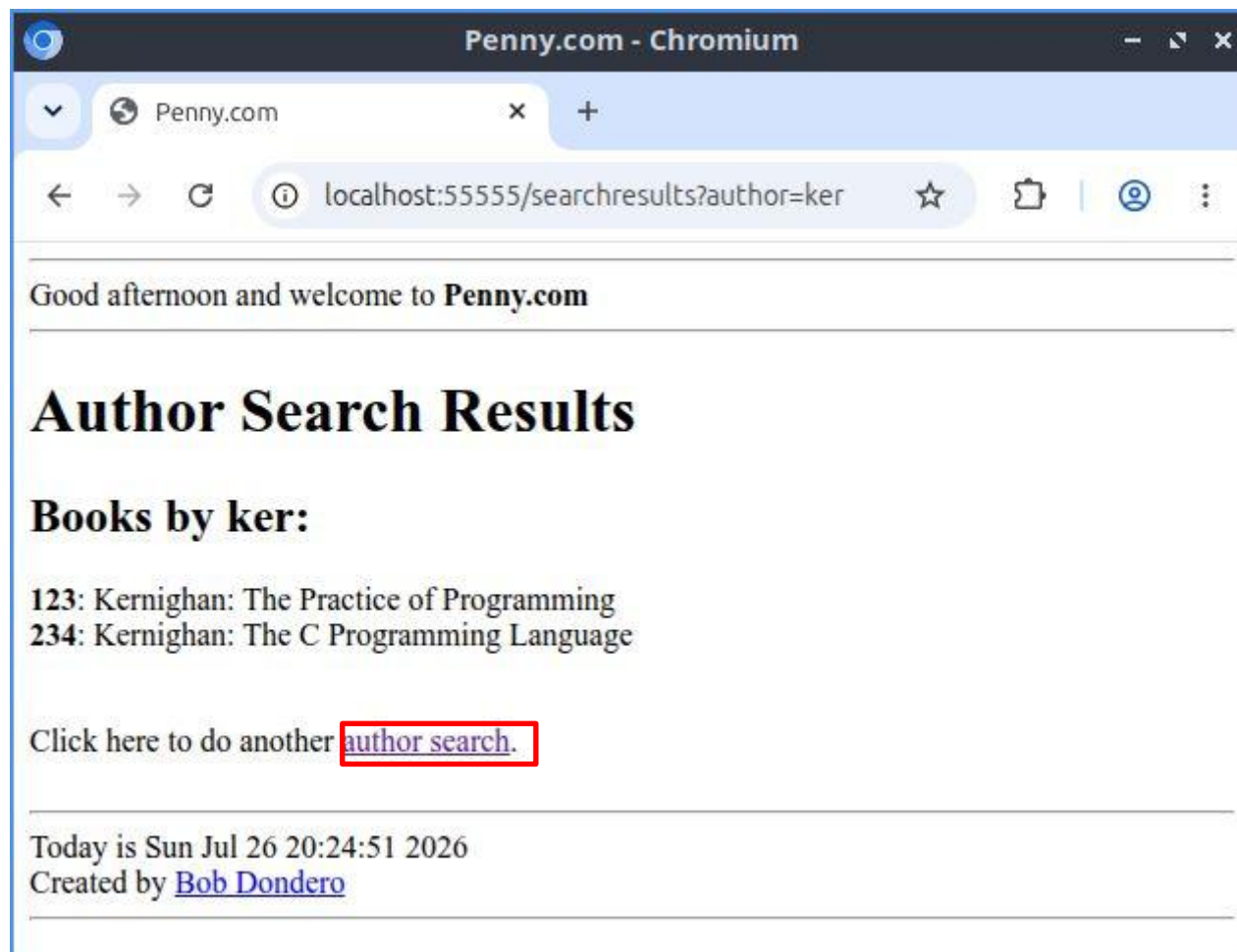


The screenshot shows a web browser window titled "Penny.com - Chromium". The address bar displays "localhost:55555/searchform". The page content includes a welcome message "Good afternoon and welcome to Penny.com", a section titled "Author Search", and a form with the label "Please enter an author name:". The input field contains the text "ker" and is highlighted with a red box. A "Go" button is also highlighted with a red box. Below the form, it says "Previous author search: (None)". At the bottom, it displays the date and time "Today is Sun Jul 26 20:23:44 2026" and the creator "Created by Bob Dondero".

← Note

Stateful Web Programming

- See **PennyFlaskJinjaState** app (cont.)



Stateful Web Programming

- See **PennyFlaskJinjaState** app (cont.)

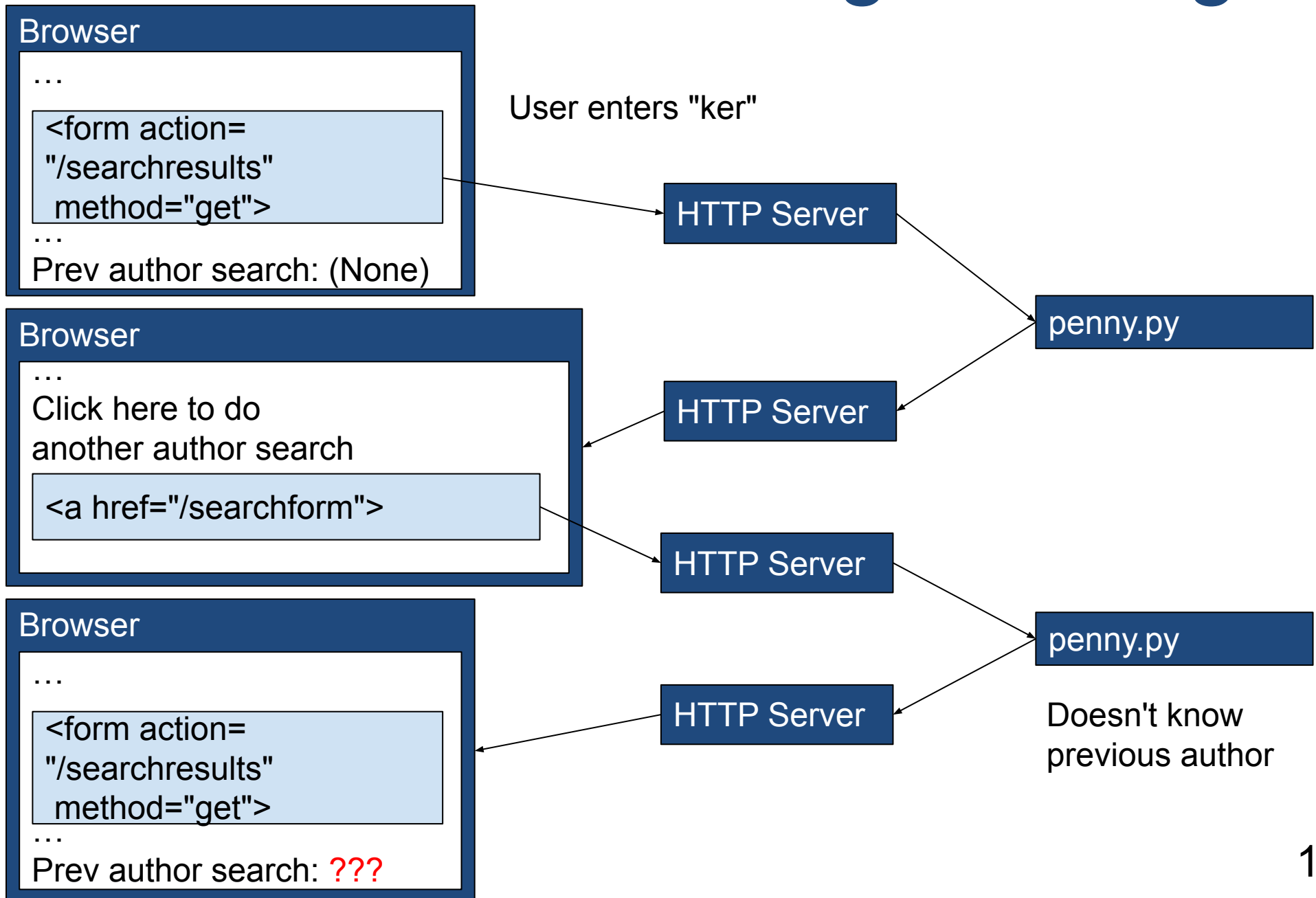


Note

Stateful Web Programming

- See **PennyFlaskJinjaState** app (cont.)
 - Displays name of previously-searched-for author in searchform page
 - But how???

Stateful Web Programming



Question (09webpython4a)

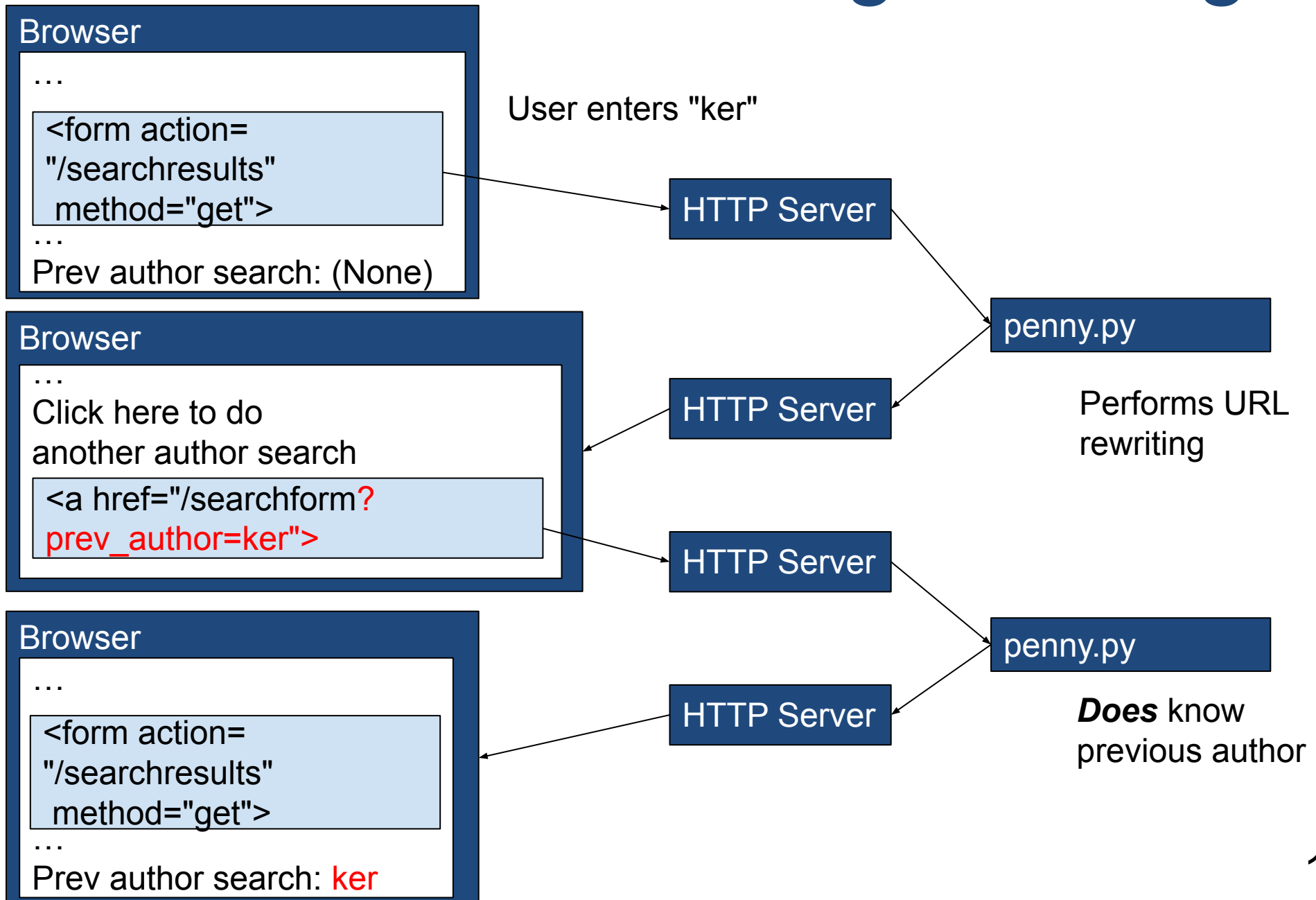
- To achieve the desired behavior, the Penny app could store the previous author in:
 - (a) A `prev_author` local variable within one of its route-handling functions
 - (b) A `prev_author` global variable
 - (c) A `PREVAUTHOR` environment variable
 - (d) None of the above
- Browse to <https://cos333attend.cs.princeton.edu> to answer

[2 points]

Stateful Web Programming

- **Problem (recall):**
 - HTTP is a stateless protocol
- **Solution 1: *URL rewriting***
 - Append state data to end of URL

Stateful Web Programming



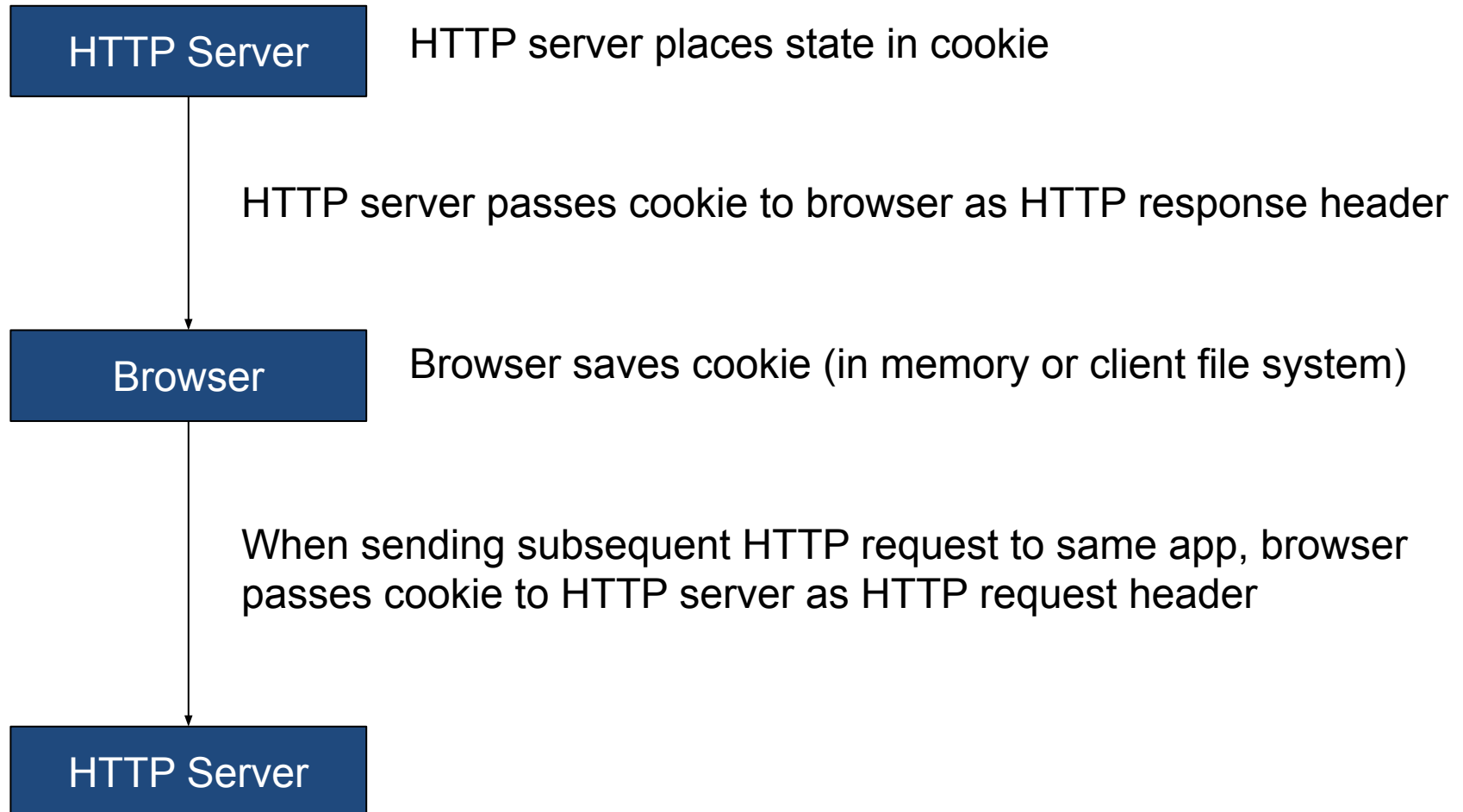
Agenda

- Stateful web programming
- **Cookies**
- Web programming in COS 333

Cookies

- **Problem (recall):**
 - HTTP is a stateless protocol
- **Solution 1: URL rewriting**
 - Append state data to end of URL
- **Solution 2: *Cookies***

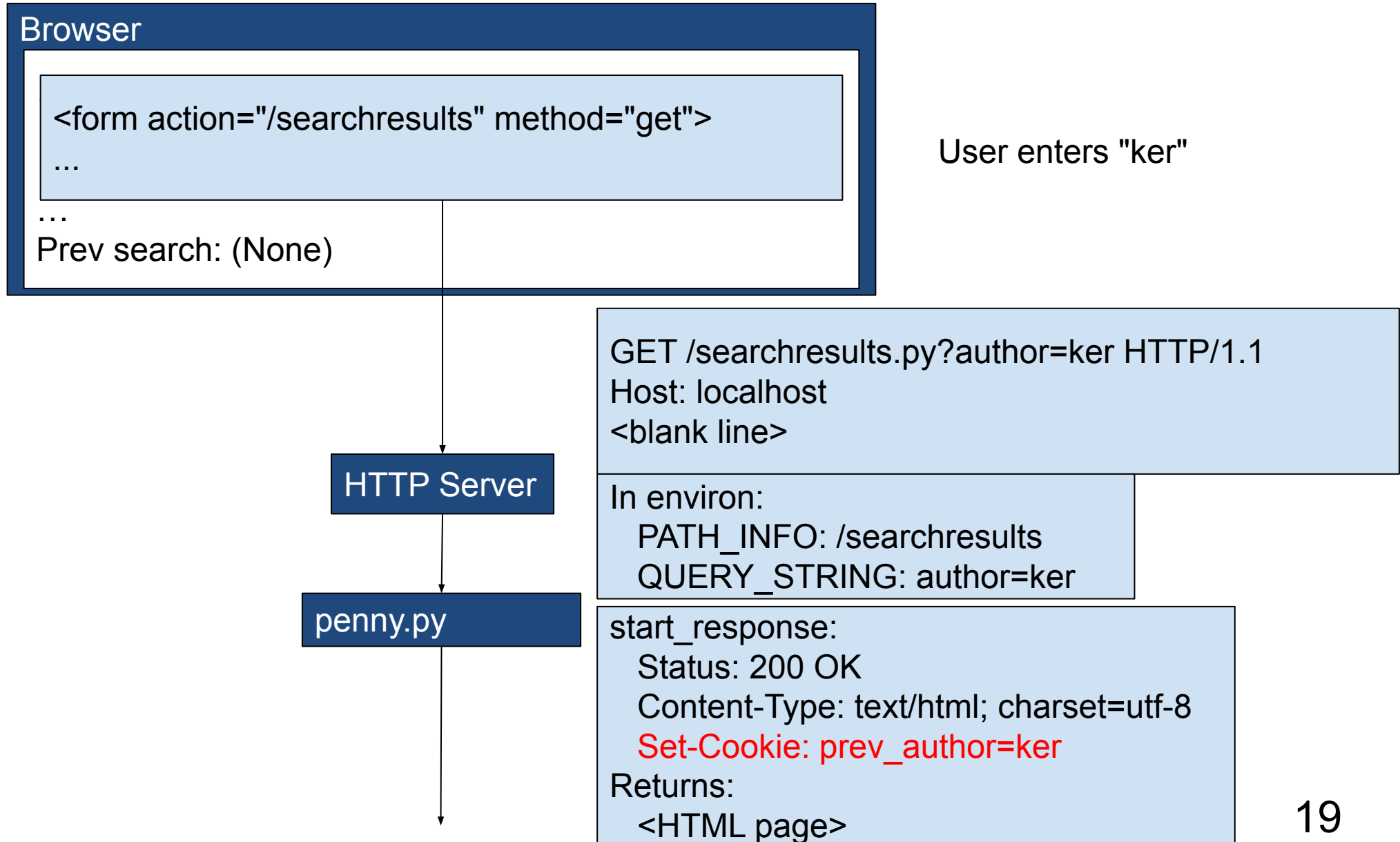
Cookies



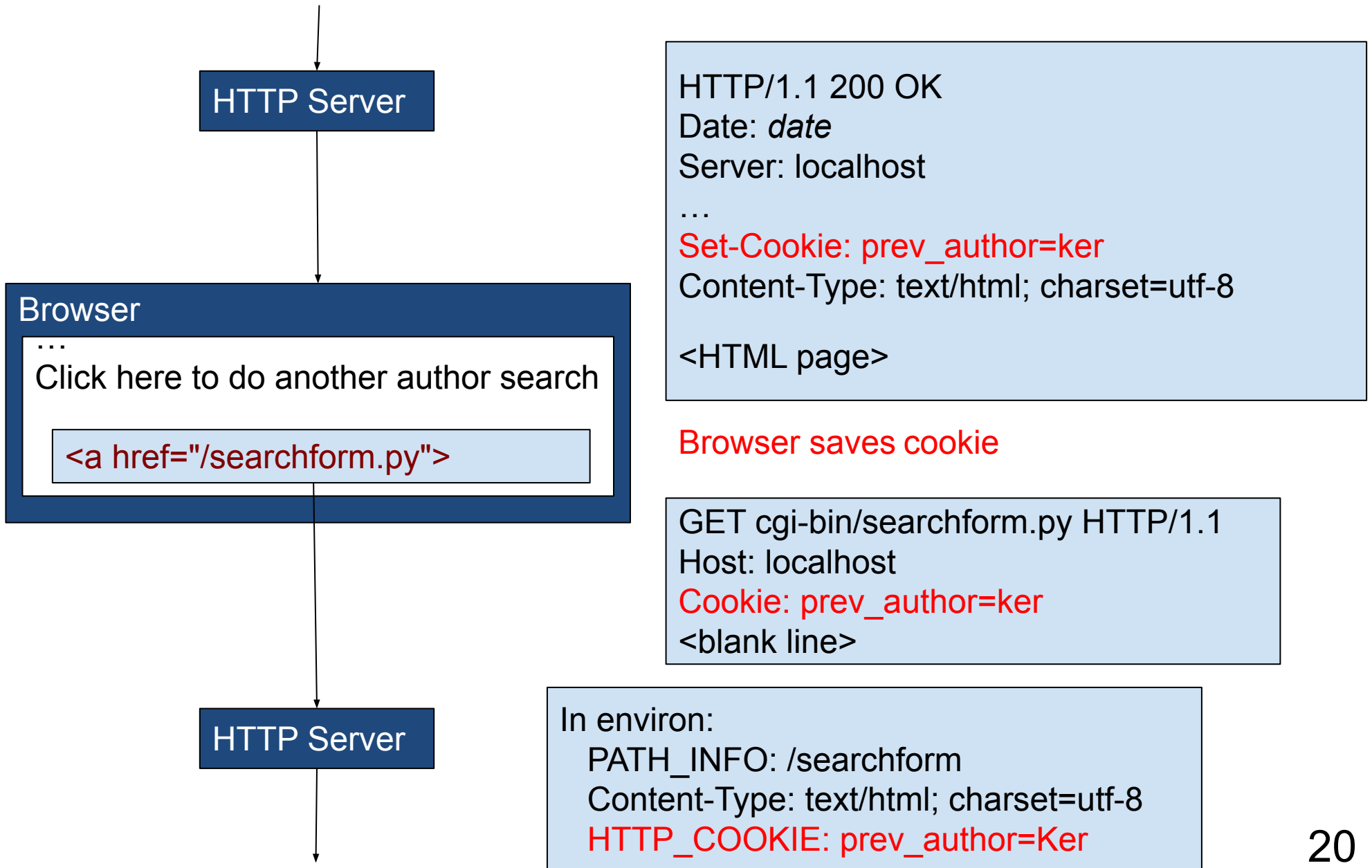
Cookies

- Cookie attributes:
 - Name
 - Content
 - Host & path
 - Expiration date
 - ...

Cookies



Cookies



Cookies

penny.py

Knows previous author via environ['HTTP_COOKIE']

start_response:

Status: 200 OK

Content-type: text/html; charset=utf-8

Return:

<HTML page containing ker>

HTTP Server

HTTP/1.1 200 OK

Date: *date*

Server: localhost

...

Content-Type: text/html; charset=utf-8

<HTML page containing ker>

Browser

...

<form action="/searchresults.py" method="get">

...

Prev search: ker

Cookies

- See **PennyFlaskJinjaState** app (cont.)
 - runserver.py
 - penny.sql
 - penny.sqlite
 - database.py
 - header.html, footer.html
 - index.html
 - **searchform.html**, searchresults.html
 - **penny.py**

Cookies

- Viewing cookies in Chrome:
 - Browse to a route in the Penny application after the cookie has been created
 - From the menu at the upper right...
 - More tools →
 - Developer tools →
 - Application →
 - Cookies →
 - Select `http://localhost:55555`

Cookies

The screenshot shows the Chrome DevTools Application tab. The left sidebar has a tree view with 'Application' expanded, showing 'Storage' > 'Cookies'. The 'Cookies' list shows a single cookie for 'http://localhost:55555'. The main panel displays a table of cookies.

Name	Value	Domain	Path	Expir...	Size	H...	Se...	Sam...	Partitio...	Priority
prev_author	ker	localhost	/cgi-bin	Session	14					Medium

Select a cookie to preview its value

Cookies

- Viewing cookies in Firefox:
 - Browse to a page in the Penny application after the cookie has been created
 - From the menu at the upper right...
 - More tools →
 - Web Developer Tools →
 - Storage →
 - Cookies →
 - Select `http://localhost:55555`

Cookies

Penny.com — Mozilla Firefox

File Edit View History Bookmarks Tools Help

Penny.com

localhost:5555/cgi-bin/searchform.py

Getting Started CS Department Web ... COS217 COS333 IntroCsBook IndepWork Princeton Health Linux Mint Other Bookmarks

Good afternoon and welcome to **Penny.com**

Author Search

Please enter an author name:

Inspector Console Debugger Network Style Editor Performance Memory **Storage** Accessibility Application

Cache Storage

Cookies

http://localhost:5555

Indexed DB

Local Storage

Session Storage

Name	Value	Domain	Path	Expires / Max-Age	Size	HttpOnly	Secure
prev_auth...	ker	localhost	/cgi-bin	Session	14	false	false

Filter values

Data

prev_author:"ker"

Created:"Tue, 03 Oct 2023 01:53:17 GMT"

Domain:"localhost"

Expires / Max-Age:"Session"

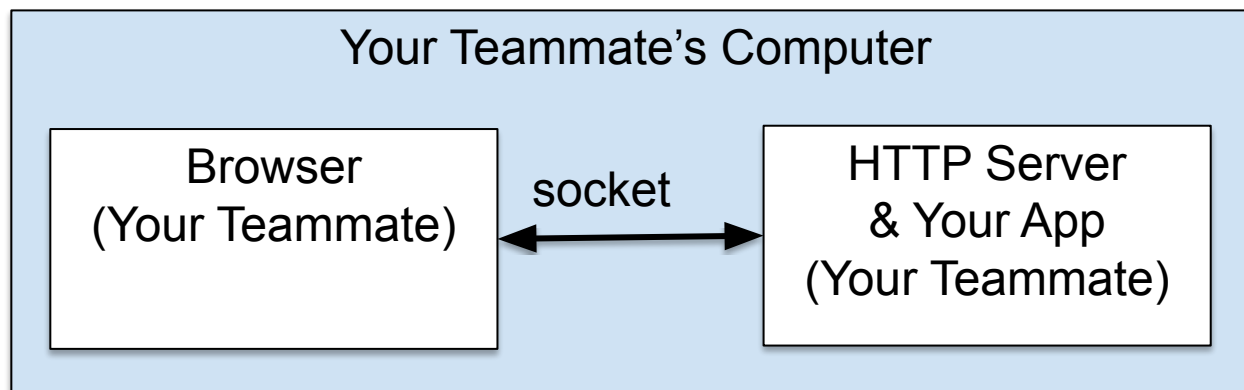
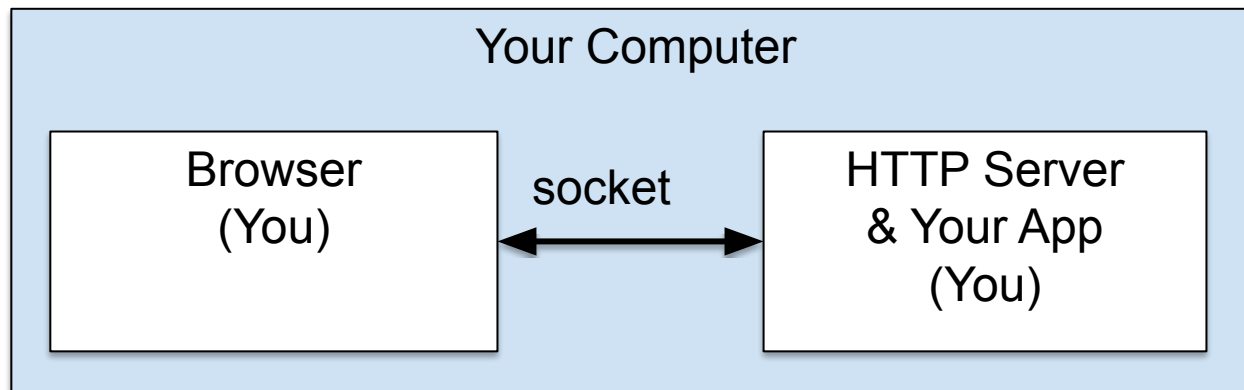
HostOnly:true

Agenda

- Stateful web programming
- Cookies
- **Web programming in COS 333**

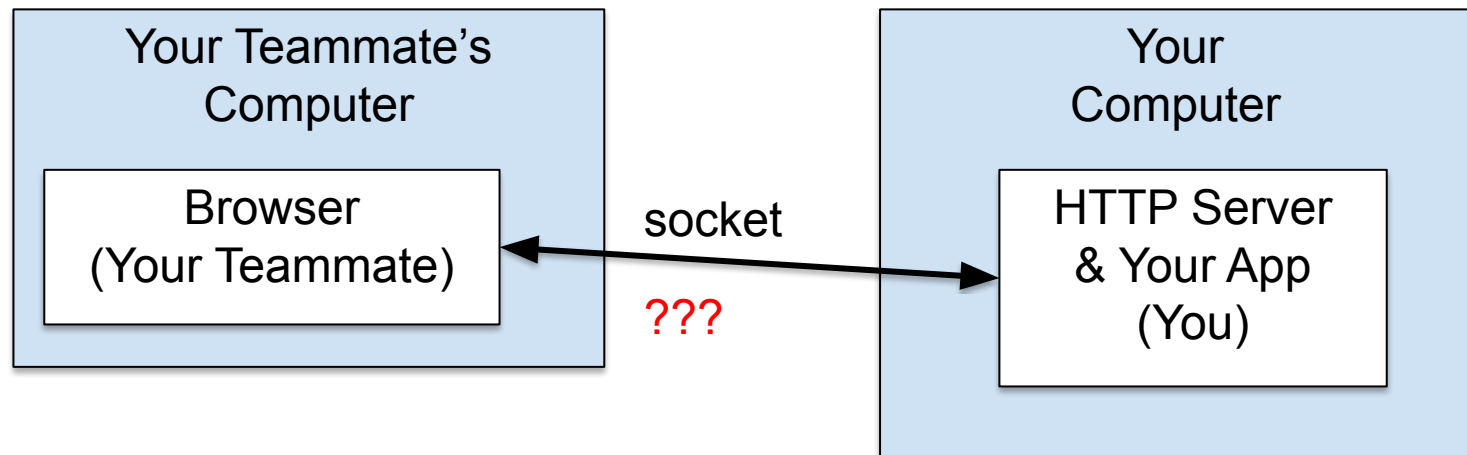
Web Pgmming in COS 333

Option 1: Run HTTP server on local computer
Run browser on same local computer



Web Pgmming in COS 333

Option 2: Run HTTP server on local computer
Run browser on different local computer



Won't work unless both computers are on Eduroam

Web Pgmming in COS 333

- Suggestion:
 - Use **option 1** during development
 - Use **option 2** to test network comm
 - If working alone:
 - Feel free to see an instructor, or...
 - Run the browser on your smart phone!

Lecture Summary

- In this lecture we covered:
 - Stateful web programming
 - Cookies
 - Web programming in COS 333
- See also:
 - **Appendix 1: Cookie Problems**

Appendix 1:

Cookie Problems

Cookie Problems

- **Problem 1:**
 - Cookie size is limited to 4K bytes
- **Solution:**
 - Cookie content stored on server-side (in database), indexed by a unique key
 - Cookie contains key only

Cookie Problems

- **Problem 2:**

- Browser user may block cookies for *all* websites

- Chrome

- Settings → Privacy & Security → Site Settings → Cookies and site data → Block all cookies

- Firefox:

- Settings → Privacy & Security → Custom → Cookies → All Cookies

Cookie Problems

- **Problem 2 (cont.):**
 - Browser user may block cookies for *some* websites
 - Encouraged by the *European General Data Protection Regulation (GDPR)*

Cookie Problems

- **Problem 2 (cont.):**

The **GDPR legislation** requires all multinational companies to provide an opt-in whereby website owners receive a user's permission to use cookies before they can be stored on a user's web browsers...

<https://us.norton.com/blog/privacy/should-i-accept-cookies>

Cookie Problems

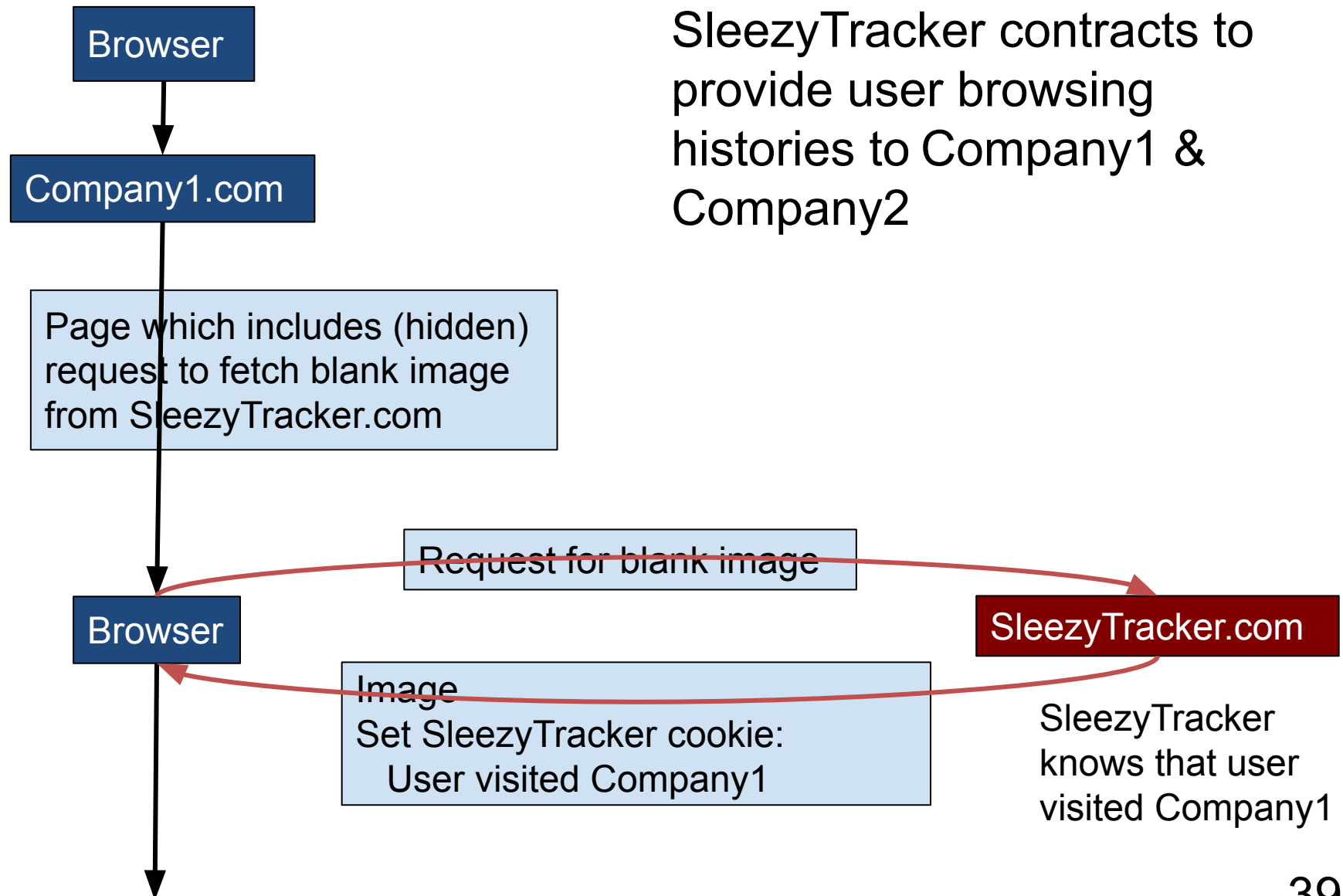
- **Solution:**
 - Ask the user to enable cookies! Or...
 - Add more logic
 - Use URL rewriting
 - Legal?

Cookie Problems

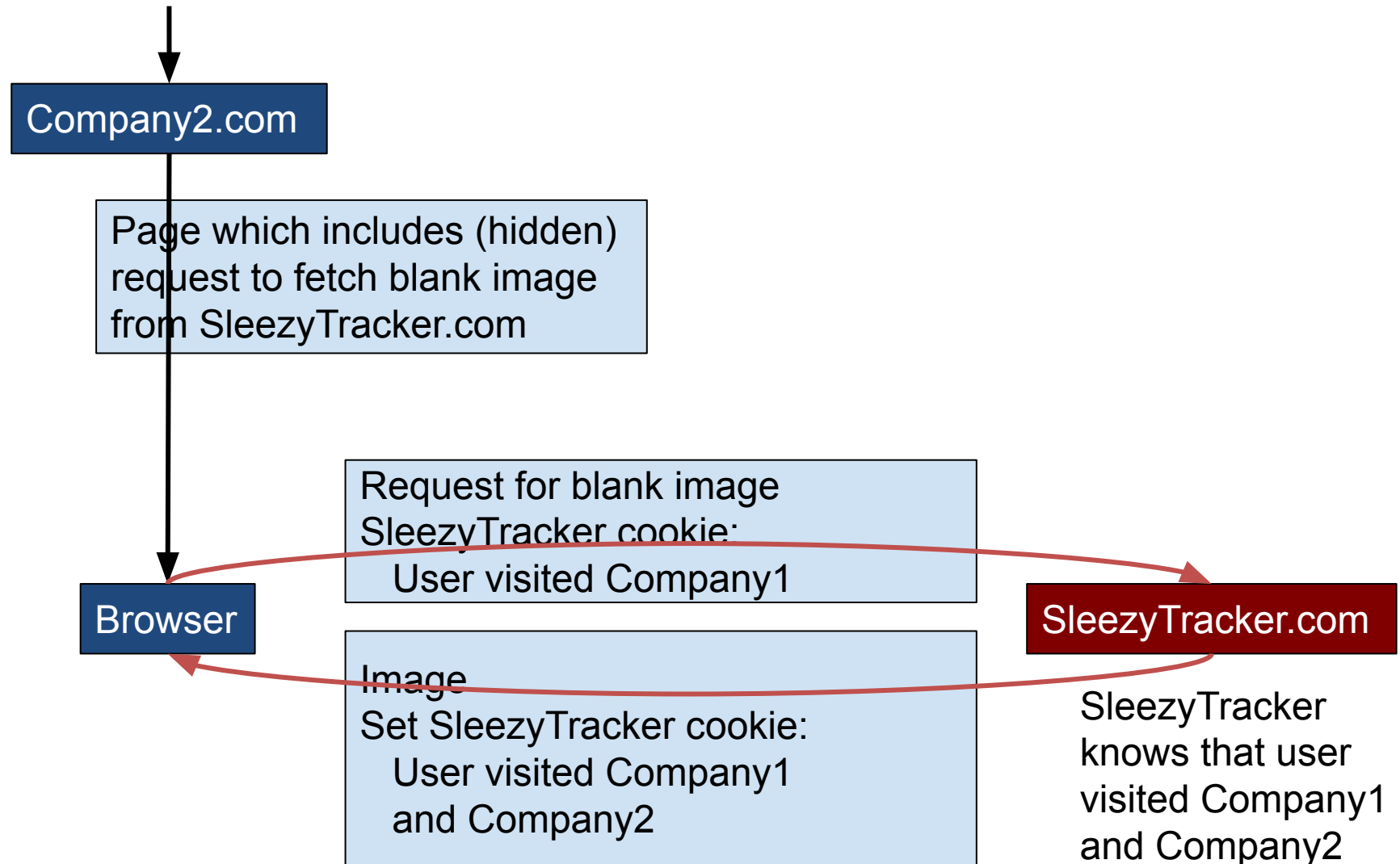
- **Problem 3:**

- *Third-party cookies* can invade privacy
 - See https://en.wikipedia.org/wiki/Cookie_stuffing

Cookie Problems



Cookie Problems



SleezyTracker provides user browsing histories to Company1 & Company2

Cookie Problems

- **Solution:** Tell browser to refuse third-party cookies
 - Chrome
 - Settings → Privacy & Security → Site Settings → Cookies and site data → Block third-party cookies
 - Firefox:
 - Settings → Privacy & Security → Custom → Cookies → Cross-site tracking cookies, and isolate other cross-site cookies
 - But disables some websites