CprE 530

Lecture 21

Topics

- HTML Protocol
- HTML Security
- Server side security
- Client Side security

HTML

- Hypertext Markup Language
- Two parts
  - Head: contains information for the browser
  - Body: contains information to display on the screen
- Contains markup codes which tell the browser how to display the page
- Each markup code is called an element or a tag
- Tags can be nested:

```
<tag1>
    <tag2>
    </tag2>
</tag1>
```

HTML

Start of an HTML document

```
<HMTL>
```

HEAD section

```
<HEAD>
<TITLE> The page title </TITLE>
</HEAD>
```

BODY section

```
<BODY>
        HTML CODE
</BODY>
```

End of the HTML document

```
</HTML>
```

HTML Tags

- Basic HTML tags

  <HTML> - tells browser where page starts

  <HEAD> - start of head section

  <TITLE> - text to be displayed in title bar

  <BODY> - start of body section

  <H1> - largest header size

  <P> - paragraph

  <BR> - break (new line)

  <UL> - unordered list

  <LI> - list item

  <a href="abc.com">link</a> - hyperlink to abc.com

  <img src ="red.gif"> - display the image red.gif

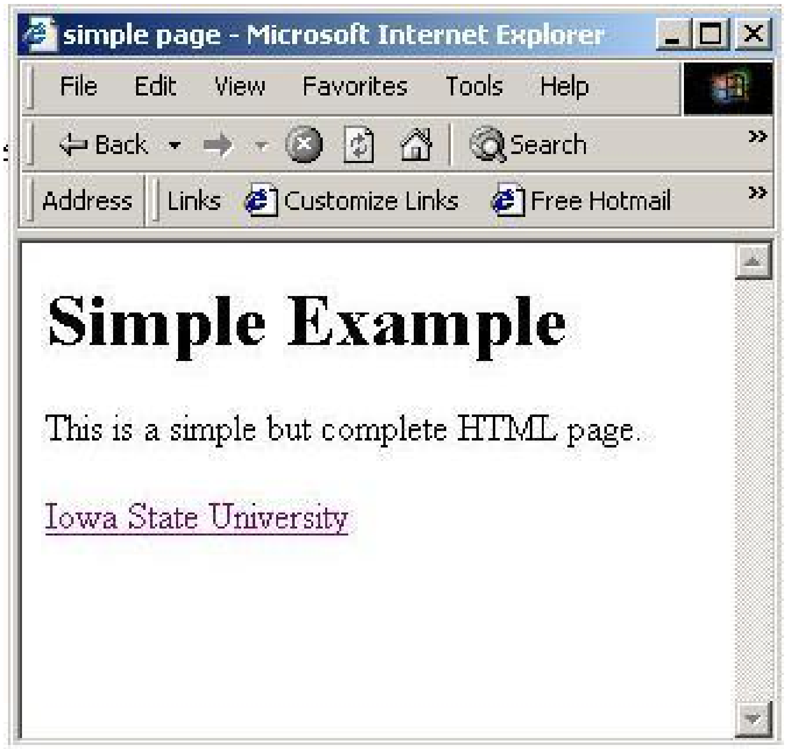
  <APPLET> CODE=XXX </APPLET> - java applet

HTML Example

- Here is a simple HTML page

```
<HTML>
<HEAD><TITLE>simple page</TITLE>
</HEAD>
<BODY>
<H1>Simple Example</H1>
<p>
This is a simple but complete HTML page.
<p>
<a href=http://www.iastate.edu>Iowa State University</a>
</BODY>
</HTML>
```

HTML Example


simple page - Microsoft Internet Explorer
File Edit View Favorites Tools Help
Back
Search
Address
Links
Customize Links
Free Hotmail

Simple Example

This is a simple but complete HTML page.

Iowa State University

Header based

- HTML documents with hyperlinks where the text is different than the link
- Pictures can come from anywhere
- Links to rouge code.
- Countermeasures:
  - User education

Protocol Based

- Different that normal protocols (no message exchange)
- Client side downloads can be malicious (viruses, worms, Trojan horses)
- Countermeasures:
  - Scanners, filters
  - Education

Authentication Based

- HTML does not directly support authentication
- HTML can be used to direct you to the wrong site, and since there is no host to user authentication. The site may not be the true site.
- Countermeasures:
  - User education

Traffic Based

- Sniffing

Server Side Security

- HTML documents can cause applications to be run.
- Common method is via a CGI script
- HTML documents can also front end other applications like databases through a CGI script

CGI

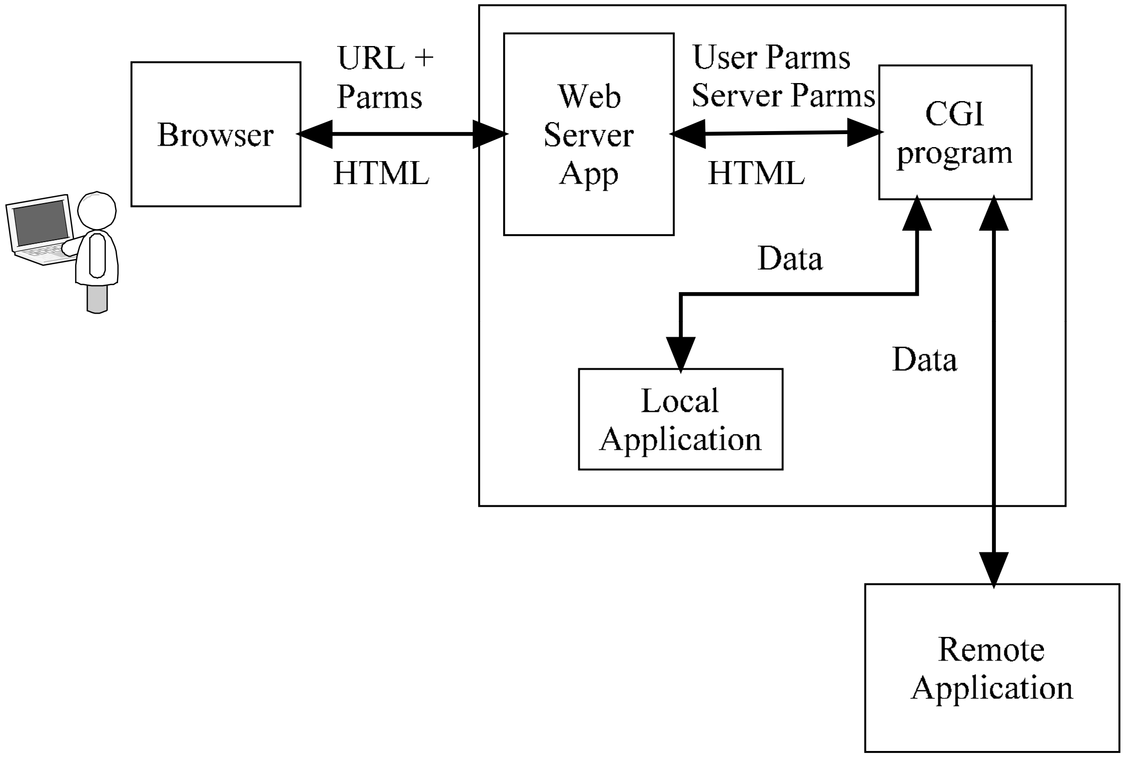
- Common Gateway Interface
- Allows a server to run programs and scripts
- CGI is the method for passing data back and forth between the server and the program or script
- Variables can be passed to the program or script either through a form or after the '?' in the URL
- Examples:

http://HOST/cgi-bin/program.pl?name=bob;state=ia

or

<FORM METHOD=POST ACTION=/cgi-bin/program.pl>

CGI



Browser

URL + Parms

HTML

Web Server App

User Parms
Server Parms

HTML

CGI program

Data

Local Application

Data

Remote Application

CGI

- CGI can access additional information through environment variables
- Environment variables are passed from the server to the program or script
- Environment variables include:

  Query_string
  Remote_addr
  Remote_host
  Remote_user
  Server_name
  HTTP_referrer
  HTTP_user_agent
  Path_info
  Server_port

Header Based

- Buffer overflow problems on CGI scripts
- Server can pass HTTP requests to back-end servers and applications so header problems are not just with the WEB server
- Some header-based attacks facilitate authentication-based attacks or allow direct access to the web server

Protocol Based

- Not many protocol based attacks since it is not a protocol.

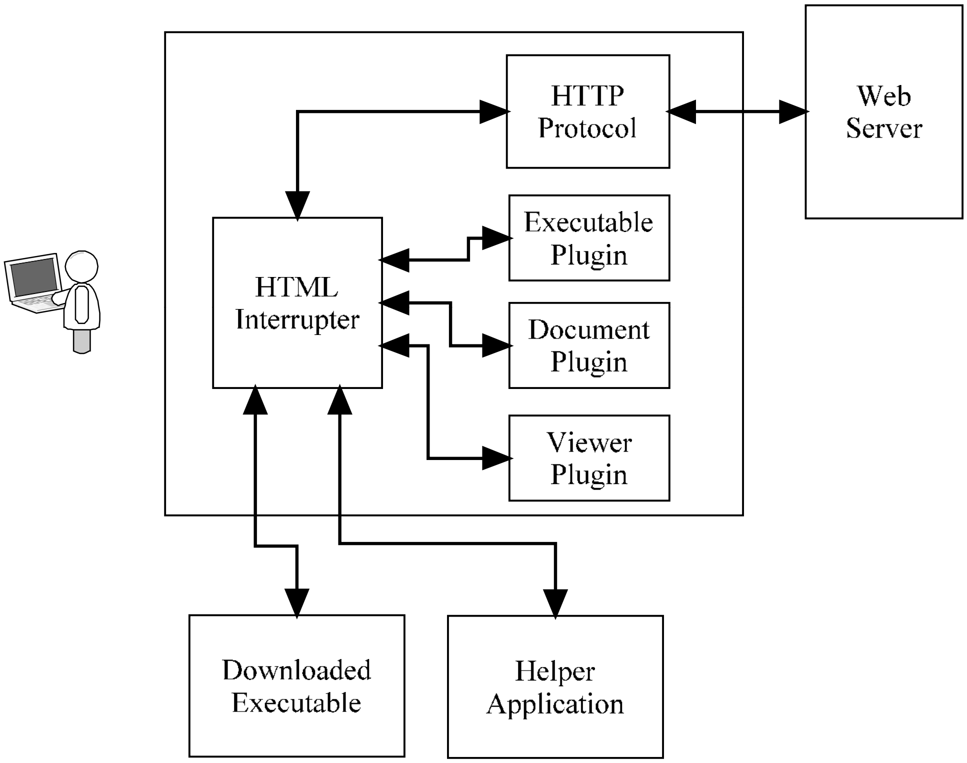
Authentication Based

- Provide access to application authentication methods.

Traffic Based

- No additional attacks due to CGI scripts

Client Side Security



Client Side Security

- Cookies are placed on the client
- Executable programs can be downloaded automatically by the browser.
  - Java Scripts
  - Active X
- They can send information back to the server.

Cookies

- A file on the users computer in which the website can store data
  - Why cookies?
    - HTTP is stateless protocol, websites like to keep state information on your information and habits
- First implementation of cookies allowed any site to read another website's cookie.
- Now only the site the storied the cookie can look at it
- Example of Amazon cookie
- Netscape has one cookie file whereas explorer has a file for each cookie
- Passwords can be in clear text

Clear Gifs

- One pixel gif
- Hyperlink to another site
- This allows people to track documents

Client side Executables

- Plugins: Applications that are part of the browser to help read different file types
- Scripts: Programs run by the browser often to provide inactive graphics or forms
- Downloads: Programs that are downloaded using the browser

Header/Protocol Based

- Not many attacks in these categories since there is not really a separate header or protocol.

Authentication Based

- No authentication of applications leads to malicious code
- Client side executables provide a method for attackers to interject code
  - Trojan horses
  - Spyware
  - Key loggers
- Can be coupled with email attacks (using phishing to direct a user to a web side which downloads code

Authentication based

- Mitigation:
  - Client side protection
  - User awareness

Traffic Based

- Not very common since, however some malicious programs may generate large amounts of network traffic.

General Countermeasures

- Encryption and authentication
- URL Filtering
- Content filtering

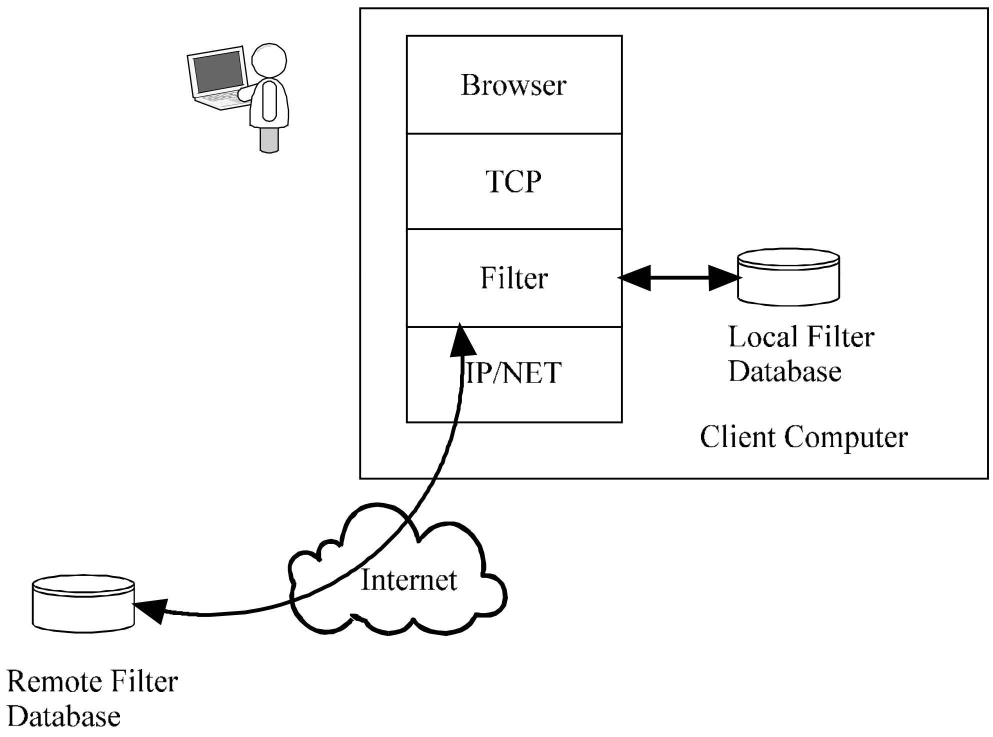
Encrypted Transactions

- SSL
  - Secure Socket Layer
  - Broader application then HTTP
  - Another layer to the mix, creates a secure layer between HTTP and TCP
  - Uses port 443
  - Browser is shipped with certificates for support of this service
  - Communicates through an encrypted channel

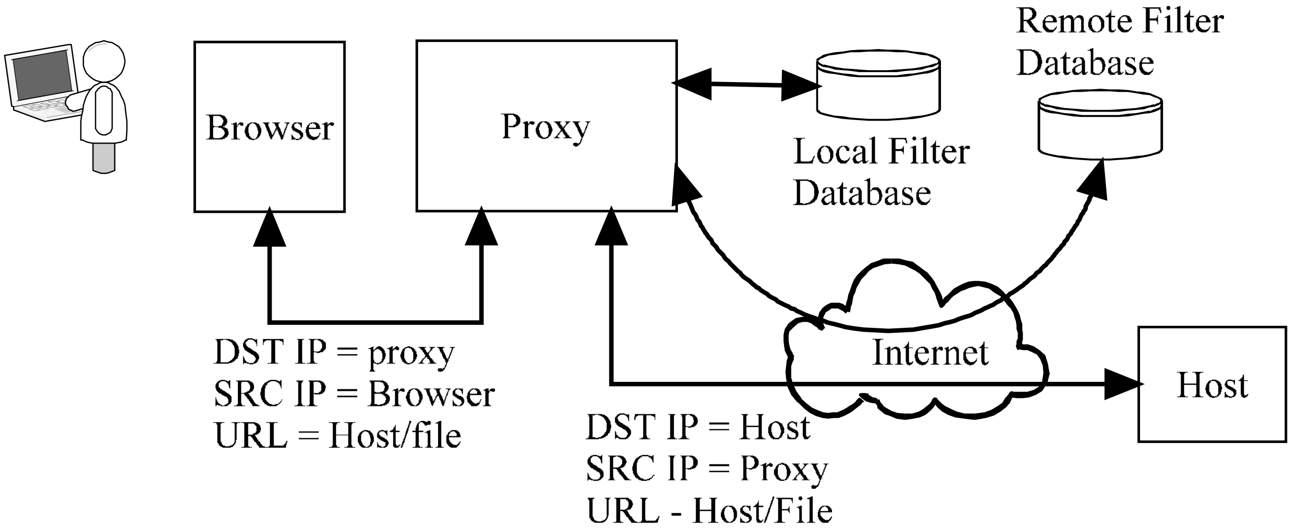
URL Filtering

- Client side
- Proxy based
- Network based

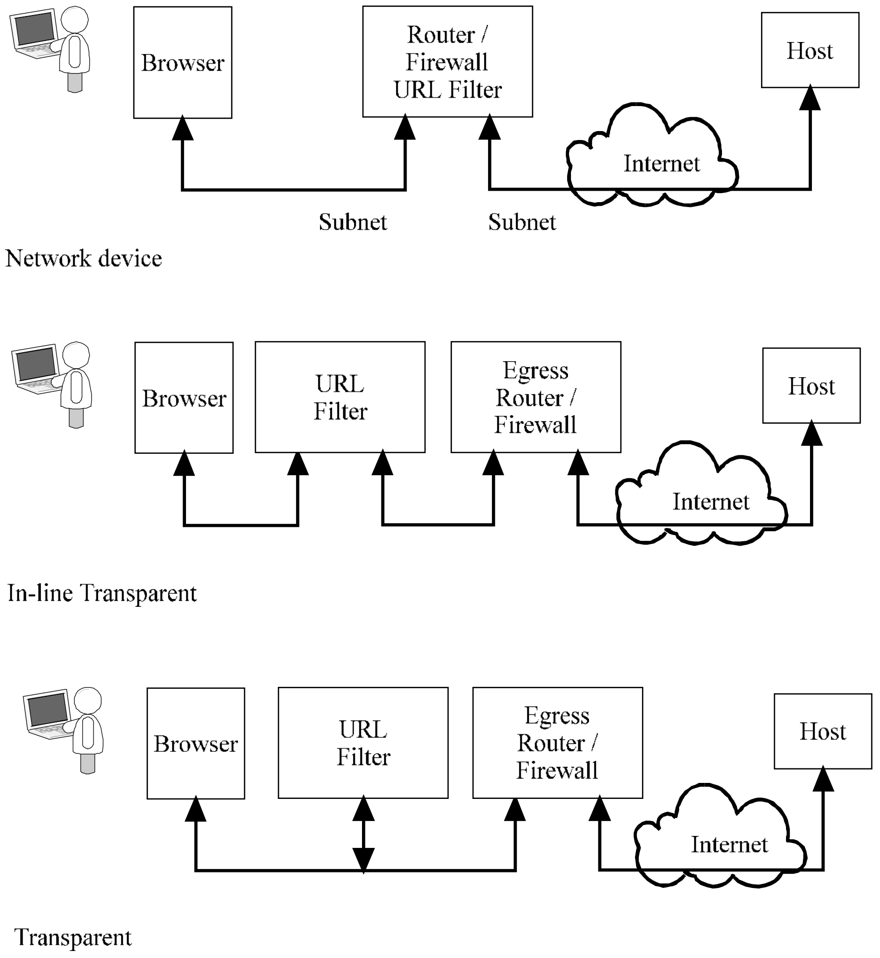
Client Side URL Filter



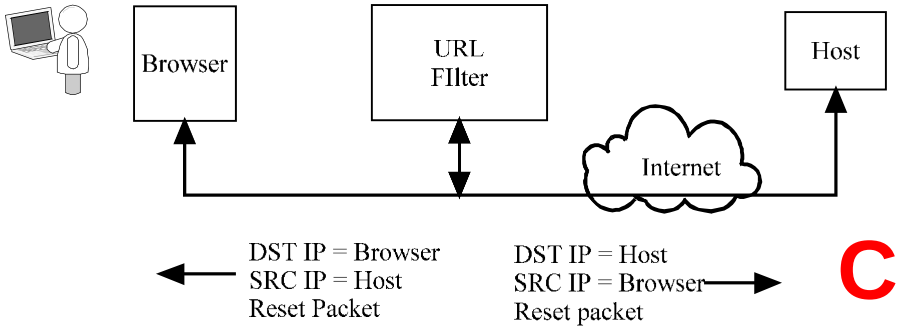
Proxy Based URL Filter



Network Based URL Filter



Connection Blocking



Browser

URL
FIlter

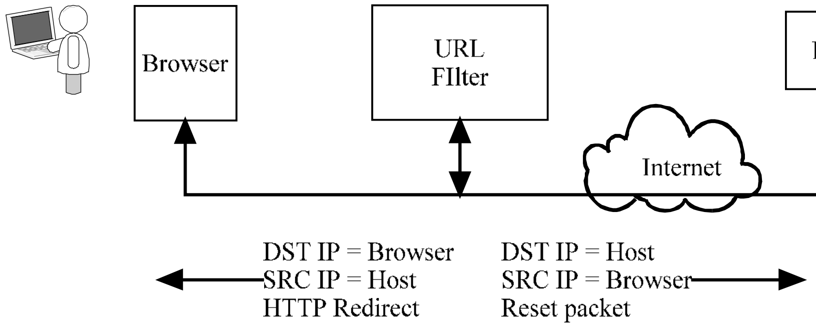
Host

Internet

DST IP = Browser
SRC IP = Host
Reset Packet

DST IP = Host
SRC IP = Browser
Reset packet

Termination Blocking



Browser

URL
FIlter

Host

Internet

DST IP = Browser
SRC IP = Host
HTTP Redirect

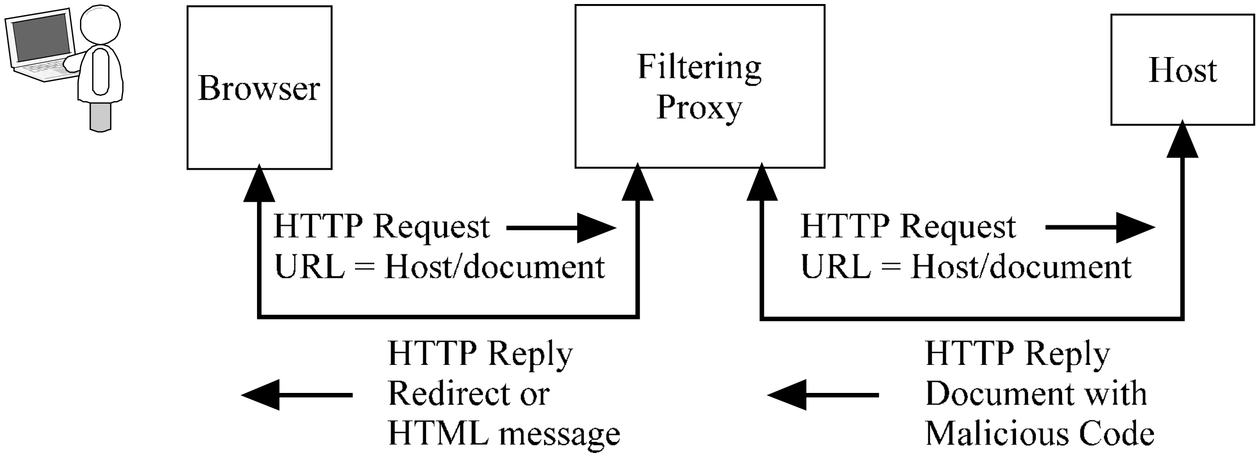
DST IP = Host
SRC IP = Browser
Reset packet

Redirection Blocking

Content Filters

- Proxy based
- Network based

Proxy Based Content Filter



Browser
Filtering Proxy
Host
HTTP Request
URL = Host/document
HTTP Request
URL = Host/document
HTTP Reply
Redirect or
HTML message
HTTP Reply
Document with
Malicious Code