World Wide Web – Part II

Lecture 12: World wide web – Part II

On completion, the student will be able to:

- **Design a skeletal web server for responding to basic HTTP queries.**
- **Explain the roles of proxy servers, and network address translators.**
- **Explain the various ways in which a network address translator can work.**

How a Web Server Works?

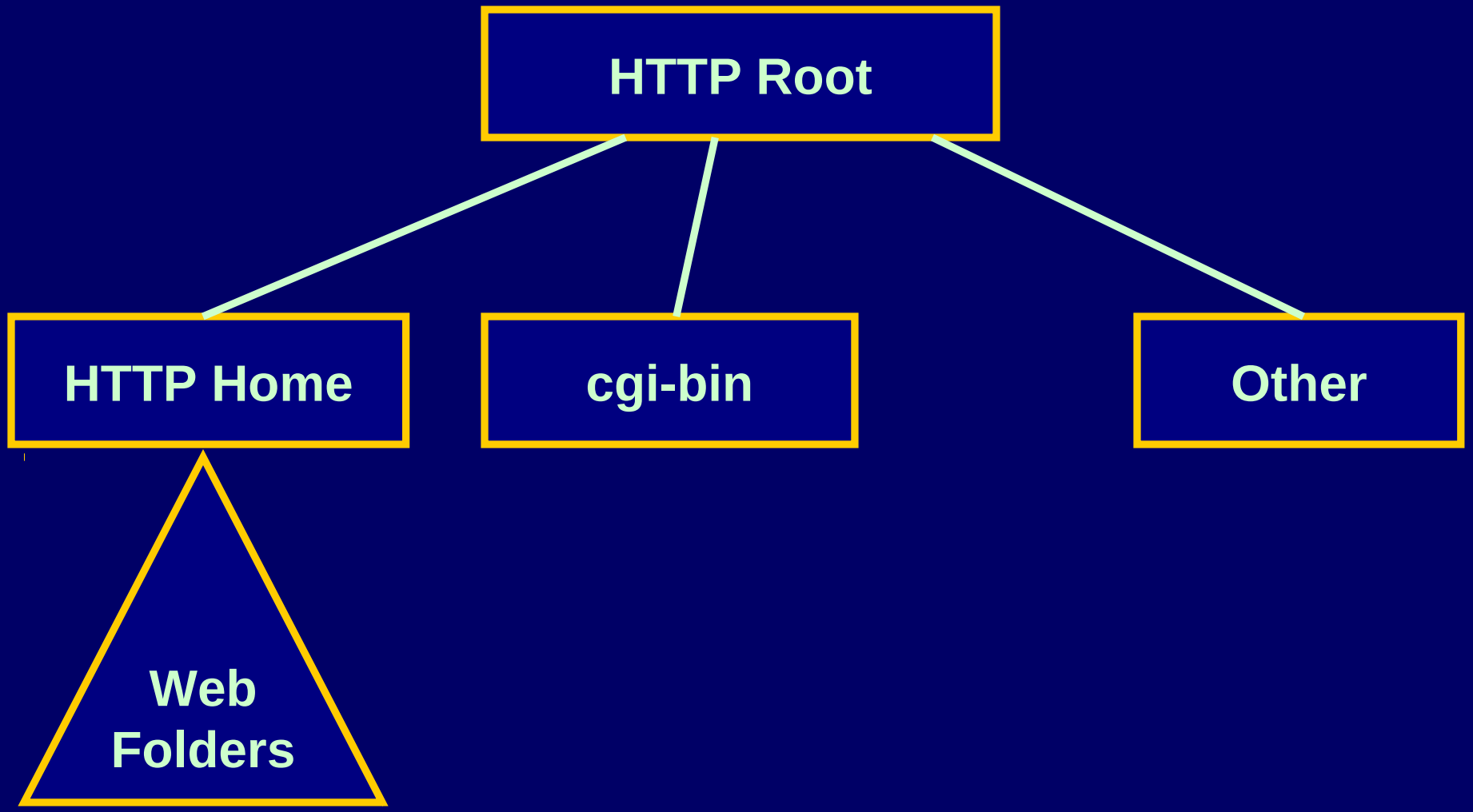
Requirements of a Web Server

- Simple requirements:
  - Able to accept HTTP requests, and respond to them.
  - Support for GET and HEAD, possibly also POST.
  - Able to handle server-side scripts.
    - Executables residing on the server.
    - They get executed when specified.
    - Their output sent back to the client; typically as a HTML page.

Directory Structure

- When a web server is installed:
  - A http root directory gets created.
    - For example, “/home/httpd”
  - There is a directory under the http root that acts as the http home directory.
    - For example, “/home/httpd/docs”
  - There is a directory under the http root under which all the CGI and other scripts are to be stored.
    - For example, “/home/cgi-bin”
    - Server-side exec permission provided

Directory Structure (contd.)



Default Web Page

- There is a default web page that gets returned by the server if no explicit document path is specified.
  - index.htm or index.html.
  - Any other name can also be specified through server script configuration.
- Can be accessed as:

  GET www.xyz.com HTTP/1.0

How are Scripts Handled?

- Server-side scripts
  - A file that is to be executed by the server, and the output sent back to the client.
- How does the server know?
  - Two ways:
    - GET command with a “?”.
    - POST command.

GET Command with a “?”

- Consider an example HTTP command:

  GET /cgi-bin/xyz.pl ? roll=1234 & sex=M

- What happens?
  - Server identifies the “?” following the GET.
  - Identifies xyz.pl as a program to be executed.
  - Allows the xyz.com program to read the values present in the string following the “?”.
    - How, to be discussed later
  - The output generated by the xyz.com program is sent back to the client.

POST Command

- Works similar to get.
- Differences:
  - The name-value pairs are present as data following the header lines and a blank line.
  - Not limited by the maximum size of a string (as in GET).
  - The executable program can read the data values.
    - How, to be discussed later.

POST Command :: Example

```
POST  /cgi-bin/myscript.cgi  HTTP/1.0
From:  isg@hotmail.com
User-Agent: HTTPTool/1.0
Content-Type: application/x-www-form-
  urlencoded
Content-Length: 32

Roll = 1234 & Sex = M & Age = 20
```

Points to Note

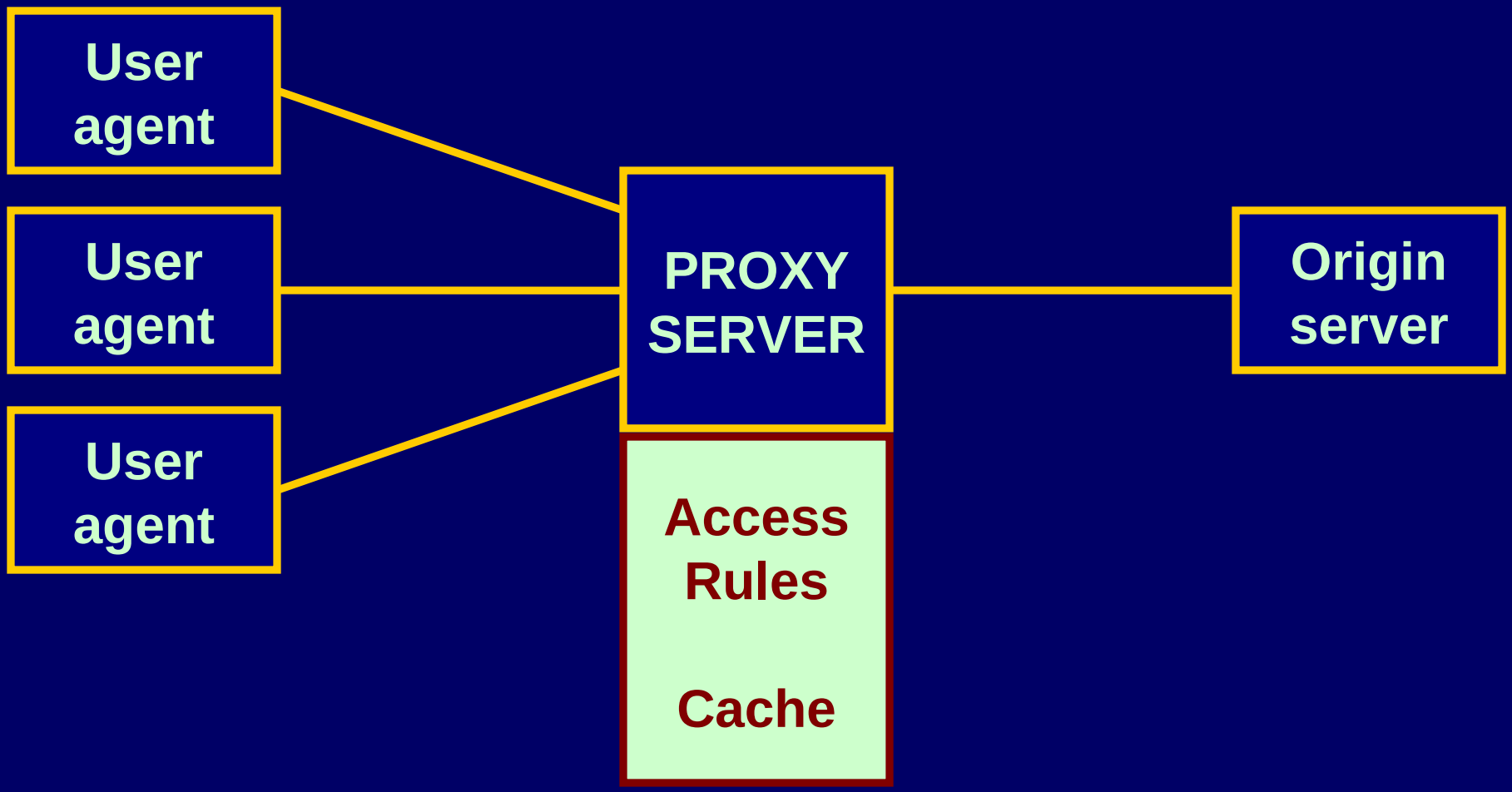
- The executable program that runs on the server can be written in any language.
  - Shell script (C shell, bourne shell, etc.)
  - Perl
  - ASP
  - PHP
  - C, Java (requires servlet support)
    - Necessary support for executing must be there in the server.
    - For example, ASP can run under IIS but not under Apache.

Proxy Server

Introduction

- What is a proxy server?
  - Acts on behalf of other clients, and presents requests from other clients to a server.
  - Acts as a server while talking with a client, and as a client while talking with a server.
- Commonly used HTTP proxy server:
  - Squid
    - available on all platforms.

Where it is located?



Functions of a HTTP Proxy

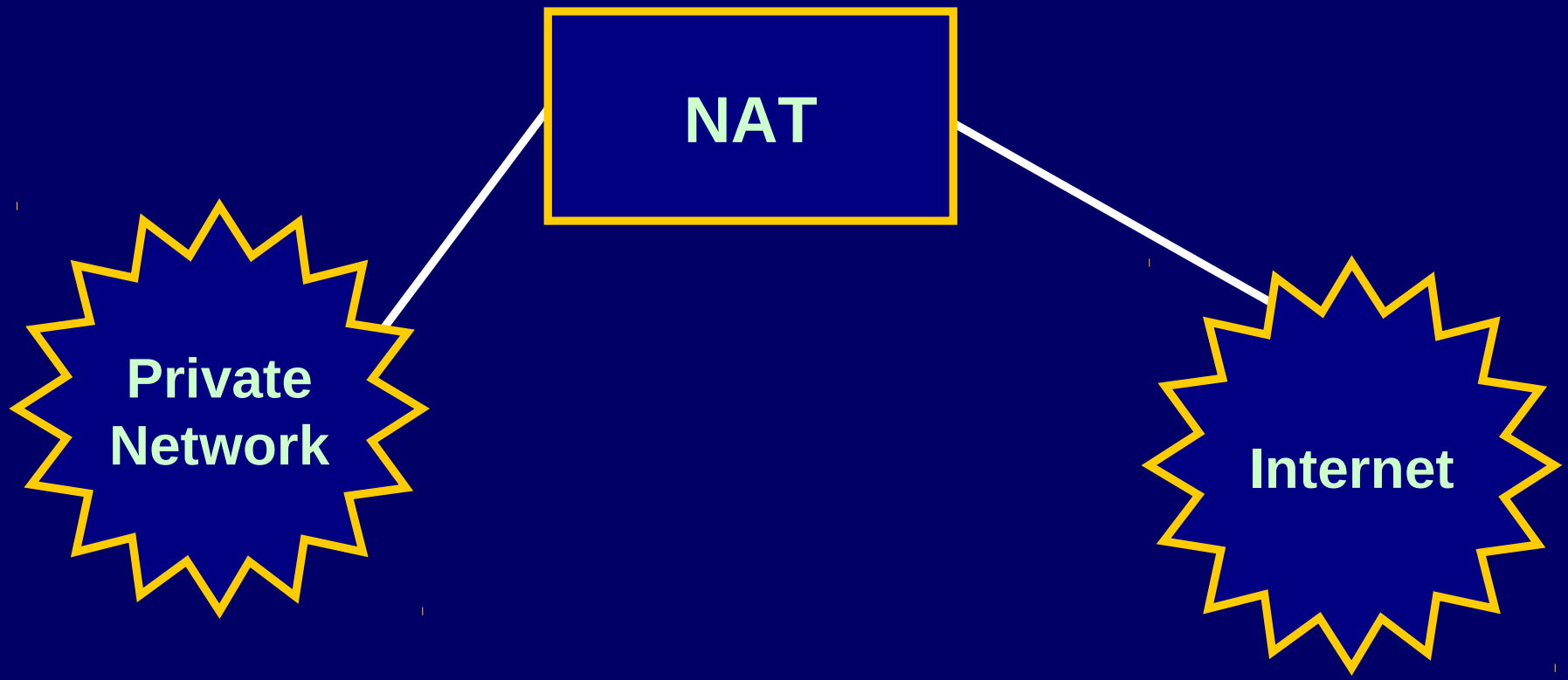
- Request forwarding
  - Primary function.
  - Acts as a rudimentary firewall.
- Access control
  - Allow or deny accesses, based on
    - Contents
    - Location
- Cache management
  - Efficient utilization of bandwidth.
  - Faster access.

Network Address Translator (NAT)

What is NAT?

- Allows a single device (router or a dedicated box) to act as an agent between the Internet (public network) and a local (private) network.
  - Tries to address the IP address distribution problem.
  - RFC 1631.
  - Only one unique IP address is required to represent an entire group of computers.
  - Several variations possible.

Where does NAT reside?



Various Forms of NAT

- Static NAT
  - Used to map an unregistered IP address to a registered IP address.
  - One-to-one mapping.
    - N registered addresses for N machines.
- Dynamic NAT
  - Used to map an unregistered IP address to a registered IP address.
    - From a given pool of registered IP addresses.
  - Addresses are assigned dynamically.
    - Any number of internal computers.
    - A limit N to the number communicating at a time.

Various Forms of NAT (contd.)

- Overloading
  - A special form of dynamic NAT.
  - Used to map multiple unregistered IP addresses to a single registered IP address by using different ports.
    - Also called port address translation (PAT).
    - Each computer on the private network gets translated to the same IP address, but with a different port number assignment.
  - Widely used.

NAT Overloading ….

- Utilizes the multiplexing feature of TCP/IP stack.
  - A computer maintains several concurrent connections with a remote computer, using different port numbers.
- The header of an IP packet contains:
  - Source IP address (32 bit)
  - Source port number (16 bit)
  - Destination IP address (32 bit)
  - Destination port number (16 bit)
  - The combination of above four elements define a TCP/IP connection.

- **Notations:**
  - **Stub domain: the internal or the private network.**
  - **Address translation table (ATT): maintained by router/NAT for address and port mapping.**
- **Easy to implement dynamic NAT.**
  - **Address translation table need only contain IP address mappings.**
    - **Private to public, and vice versa.**
    - **No port numbers needed.**

How NAT overloading works?

- The scenario:
  - Internal network has non-routable IP addresses.
  - NAT-enabled router contains a registered IP address assigned by IANA.
  - An internal host X tries to connect to, say, an outside Web server.
  - The router receives the packet from X.

- **The router will now:**
  - **Save IP address and port number from X's packet to an ATT.**
  - **In the packet, replace the IP address with the router's IP address.**
  - **Replace the port number with a port number from the ATT (look for match). For new connection, generate a unique port number.**

- When a packet comes back.
  - Its destination port is used to search ATT.
  - Source IP address and port numbers can be obtained.
  - Addresses changed accordingly.

- The ATT looks like:

Source Computer	Source IP address	Source port number	NAT IP address	NAT port number
A	10.5.17.112	500	203.11.16.5	1
B	10.5.17.85	75	203.11.16.5	2
C	10.23.10.5	2480	203.11.16.5	3
D	10.22.5.118	1120	203.11.16.5	4

Capability Limit of a NAT

- **Maximum number of concurrent translations:**
  - **Mainly determined by the size of the memory to store the ATT.**
  - **Typical entry in the ATT takes about 160 bits.**
  - **Memory size of 8 Mbyte will support about**

    8 x 1024 x 1024 x 8 / 160 = 4,19,000

    concurrent translations.

Which addresses to use inside?

- Private address classes.
  - Set aside by IANA an non-routable.
  - These addresses are considered unregistered.
  - Routers discard these addresses, if used as destination.
    - A packet from a host with a private unregistered address can reach a registered destination host, but not the reverse.

The Private Address Classes

- Class A (one)
  - 10.0.0.0 to 10.255.255.255
- Class B (sixteen)
  - 172.16.0.0 to 172.31.255.255
- Class C (256)
  - 192.168.0.0 to 192.168.255.255

Other Benefits of NAT

- Use of NAT automatically creates a firewall between the internal and external networks.
  - NAT will only allow connections that has originated from within the internal network.
  - An outside host cannot initiate a connection with an internal host.
- Inbound mapping requires static NAT.

Is NAT a Proxy Server?

- The answer is “NO”.
  - NAT is transparent to both source and destination hosts. But a proxy server is not transparent.
  - NAT is a layer 3 (network) protocol. In contrast, a proxy server works at layer 4 (transport) or higher.

End of Lecture 12