Web Servers

- This topic covers basic functionality found in most web servers as well as some specific instruction for accessing and modifying the parameters for one particular web server, Tomcat 5.0.

- Briefly looking at how web servers support secure communication with browsers.

Server History

- Just as the NCSA MosaicTM browser was the starting point for subsequent browser development efforts by Netscape and Microsoft, NCSA's httpd web server was also a starting point for server development.
- HTTPD was used on a large fraction of the early web servers, but the NCSA discontinued development of the server in the mid-1990s.

- When this happened,several individuals who were running httpd at their sites joined forces and began developingtheir own updates to the open-source httpd software.
- Their updates were called “patches,”and this led to calling their work “a patchy server,” which soon became known as “theApache server.”
- They made the first public release of their free, open-source server in April1995, and within a year Apache was the most widely used server on the Web.

Server Features

•The primary feature of every web server is to accept HTTP requests from web clientsand return an appropriate resource (if available) in the HTTP response.

• Even this basicfunctionality involves a number of steps (the quoted terms used in this list are defined in subsequent paragraphs):

1. The server calls on TCP software and waits for connection requests to one or more
ports.

2. When a connection request is received, the server dedicates a “subtask” to handling
this connection.
3. The subtask establishes the TCP connection and receives an HTTP request.
4. The subtask examines the Host header field of the request to determine which “virtual
host” should receive this request and invokes software for this host.
5. The virtual host software maps the Request-URI field of the HTTP request start line
to a resource on the server.

6. If the resource is a file, the host software determines the MIME type of the file (usually
by a mapping from the file-name extension portion of the Request-URI), and creates
an HTTP response that contains the file in the body of the response message.
7. If the resource is a program, the host software runs the program, providing it with
information from the request and returning the output from the program as the body
of an HTTP response message.

8. The server normally logs information about the request and response—such as the IP
address of the requester and the status code of the response—in a plain-text file.

9. If the TCP connection is kept alive, the server subtask continues to monitor the
connection until a certain length of time has elapsed, the client sends another request,
or the client initiates a connection close.

SERVER CONFIGURATION AND TUNING

•Modern servers have a large number of configuration parameters. In this section, we will

•cover many of the key configuration items found in Tomcat. Similar features, along with

•some not found in Tomcat, are included in the Apache and IIS servers.

•Broadly speaking, server configuration can be broken into two areas:

❖EXTERNAL COMMUNICATION

❖ INTERNAL PROCESSING.

- In Tomcat, this corresponds to two separate Java packages:
  - ➢Coyote, which provides the HTTP/1.1 communication
  - ➢Catalina, which is theactual servlet container.
- Some of the Coyote parameters, affecting external communication,include the following:
  - ❖ IP addresses and TCP ports that may be used to connect to this server.
  - ❖ Number of subtasks (called threads in Java) that will be created when the server is initialized. This many TCP connections can be established simultaneously with minimal overhead.

❖Maximum number of threads that will be allowed to exist simultaneously. If this is largerthan the previous value, then the number of threads maintained by the server may change,either up or down, over time.

❖ Maximum number of TCP connection requests that will be queued if the server is already running its maximum number of threads. Connection requests received if the queue is full will be refused

- ❖Length of time the server will wait after serving an HTTP request over a TCP connection before closing the connection if another request is not received.

Defining Virtual Hosts

The virtual host name should normally be a fully qualified domain name that would be used by visitors to your web site, although the Host supplied as part of the JWSDP Service is given the unqualified name localhost.

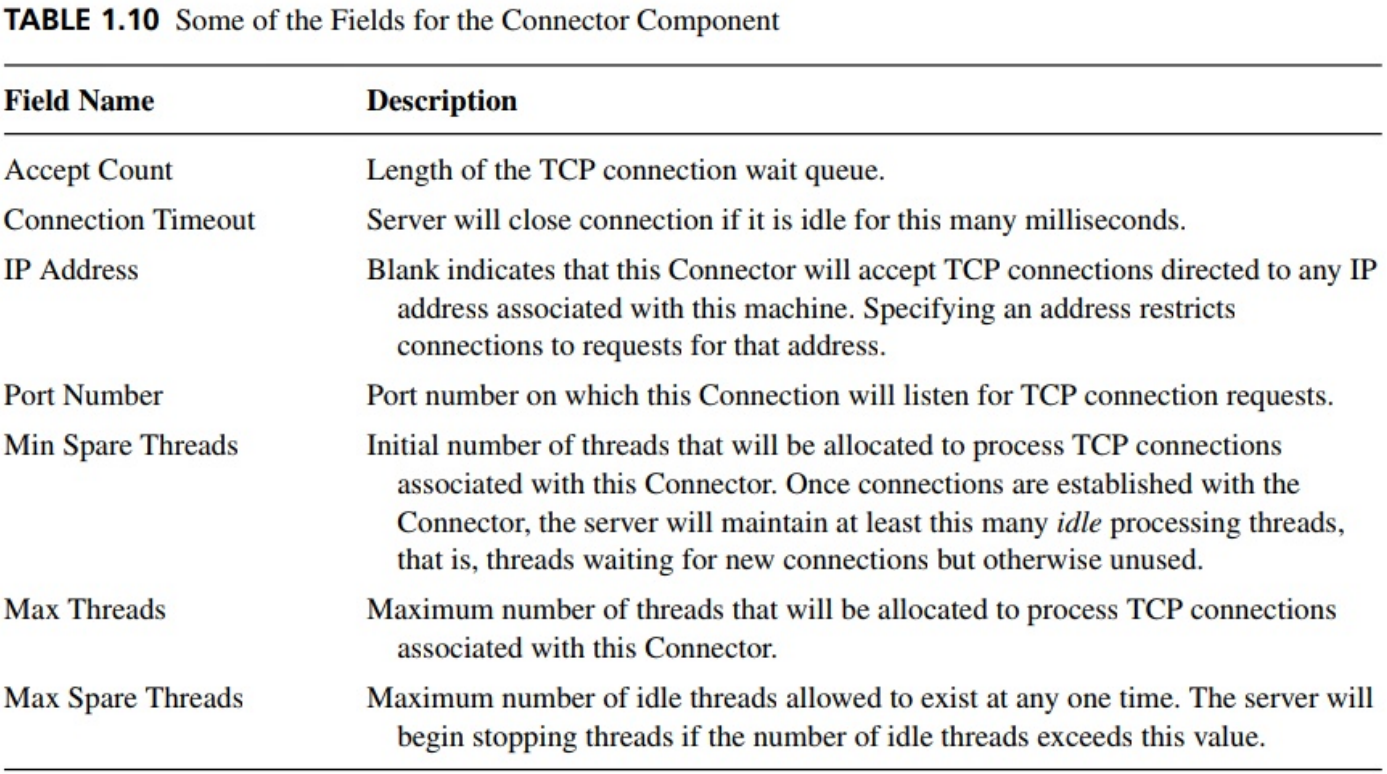
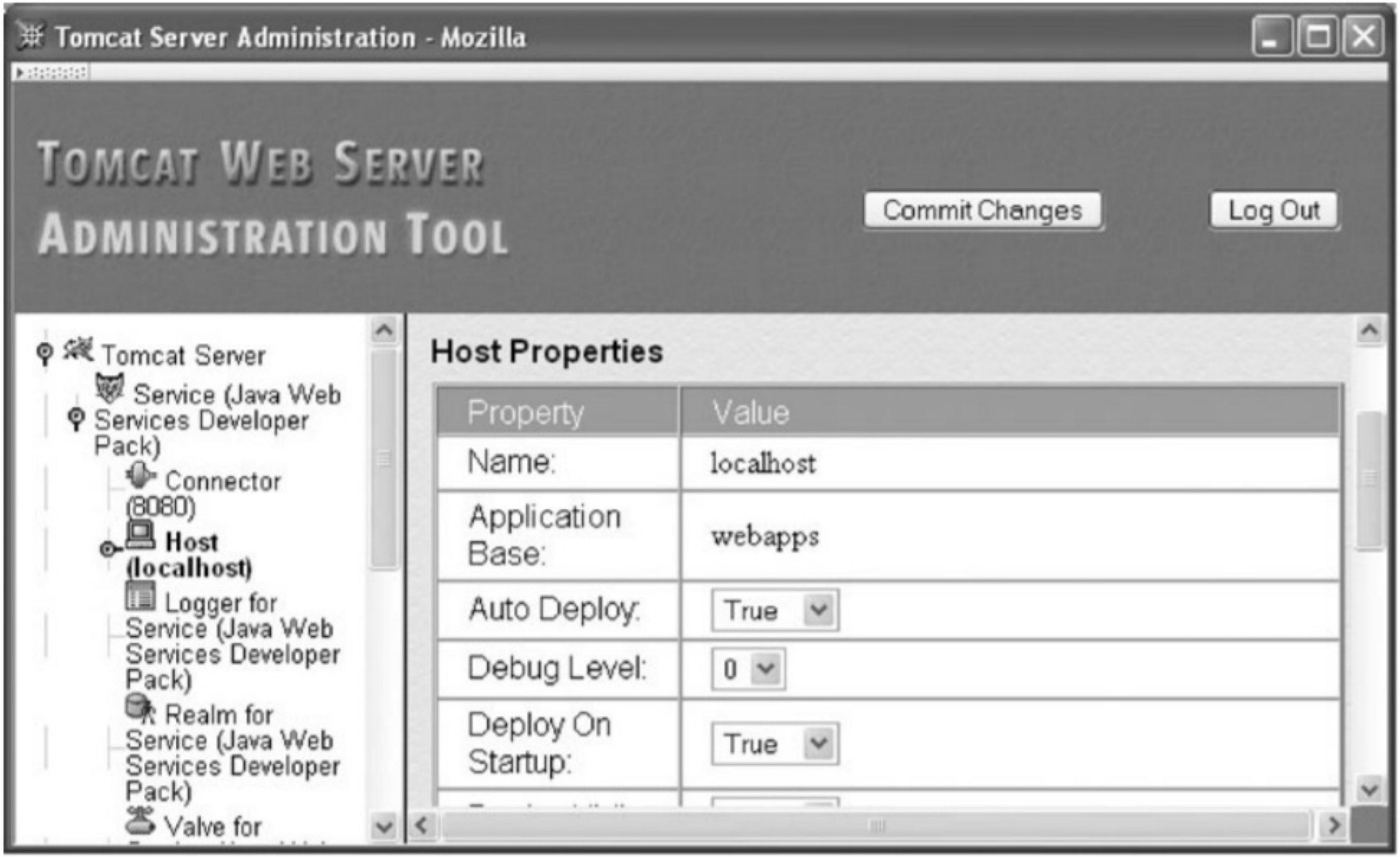


TABLE 1.10 Some of the Fields for the Connector Component

Field Name	Description
Accept Count	Length of the TCP connection wait queue.
Connection Timeout	Server will close connection if it is idle for this many milliseconds.
IP Address	Blank indicates that this Connector will accept TCP connections directed to any IP address associated with this machine. Specifying an address restricts connections to requests for that address.
Port Number	Port number on which this Connection will listen for TCP connection requests.
Min Spare Threads	Initial number of threads that will be allocated to process TCP connections associated with this Connector. Once connections are established with the Connector, the server will maintain at least this many *idle* processing threads, that is, threads waiting for new connections but otherwise unused.
Max Threads	Maximum number of threads that will be allocated to process TCP connections associated with this Connector.
Max Spare Threads	Maximum number of idle threads allowed to exist at any one time. The server will begin stopping threads if the number of idle threads exceeds this value.

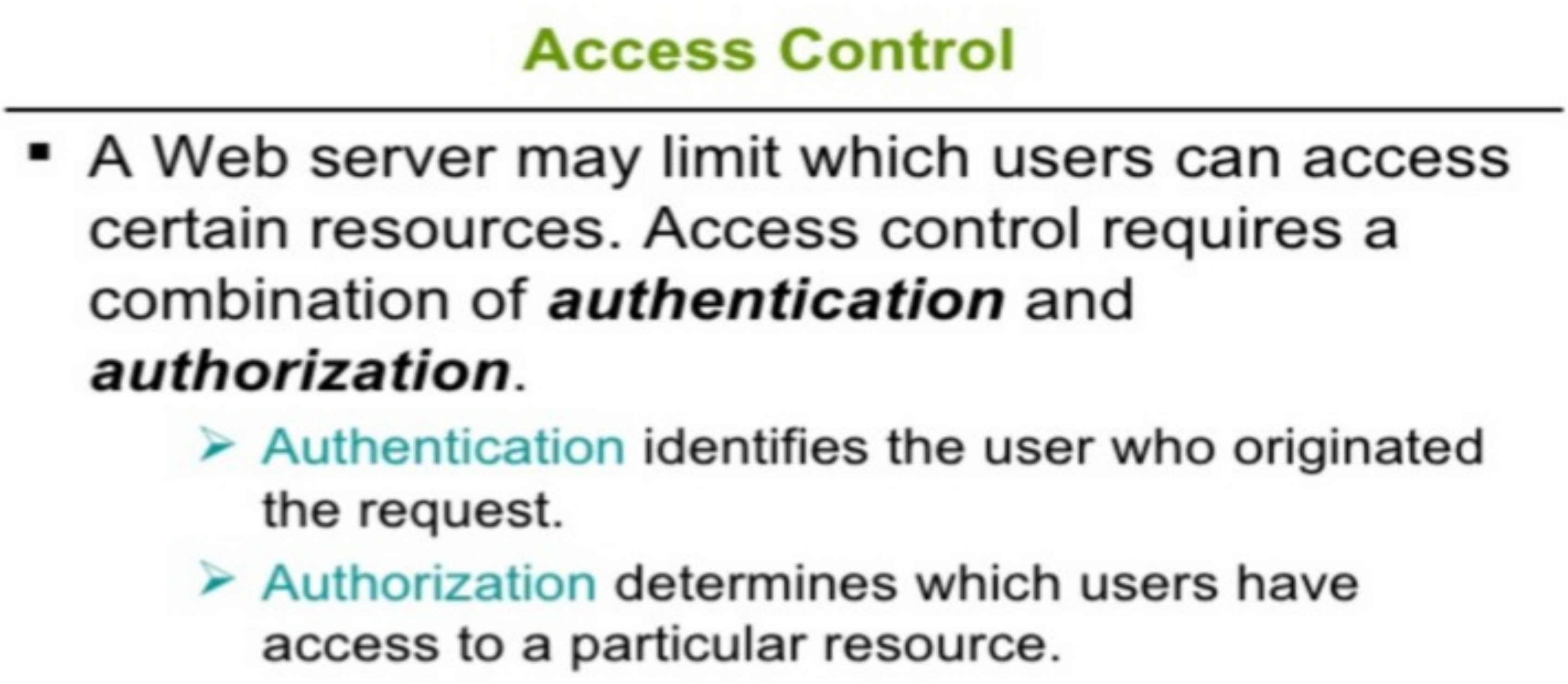


Tomcat Server Administration - Mozilla
Tomcat Web Server
Administration Tool
Commit Changes
Log Out
Tomcat Server
Service (Java Web Services Developer Pack)
Connector (8080)
Host (localhost)
Logger for Service (Java Web Services Developer Pack)
Realm for Service (Java Web Services Developer Pack)
Valve for

Host Properties

Property	Value
Name:	localhost
Application Base:	webapps
Auto Deploy:	True
Debug Level:	0
Deploy On Startup:	True

- This is a special name that the DNS system treats as a reference to a special IP address, 127.0.0.1.
- If an IP message is sent to this address, the IP software causes the message to loop back to itself for receipt.
- In short,browsing to a URL with domain name localhost causes the browser to send the HTTP request to a web server on the machine running the browser.



Access Control

- A Web server may limit which users can access certain resources. Access control requires a combination of ***authentication*** and ***authorization***.
    - ➢ Authentication identifies the user who originated the request.
    - ➢ Authorization determines which users have access to a particular resource.

AUTHORIZATION

- To control access to Web resources, the server must employ an authorization policy.
- A policy typically expressed in terms of an access control list that enumerates the users who are granted or denied access to the resources.
- In addition to checking the user name, the server may allow or deny access to the resource based on other information associated with the HTTP request, such as the host name or IP address of the requesting client.
- Authenticating HTTP requests can impose a heavy load on the Web server.

AUTHENTICATION

- Most client-server systems authenticate a user by asking for a name and password.
- Web server must perform authentication for ***every*** request for a resource that has access restrictions.
- The server returns an HTTP response that indicates that the request requires authorization.
- The response also identifies what kind of authentication is required.
- The response also identifies the ***realm***
  - a string that associates a collection of resources at the server

Thank you