TOMCAT

What is the difference between an Application Server and a Web Server?

1.	Web server encompasses web container only.	While the application server encompasses Web container as well as EJB container.
2.	Web server is useful or fitted for static content.	Whereas application server is fitted for dynamic content.
3.	Web server consumes or utilizes less resources.	While application servers utilize more resources.
4.	Web servers arrange the run environment for web applications.	While application servers arrange the run environment for enterprises applications.

5.	**In web servers, multithreading is not supported.**	**While in application server, multithreading is supported.**
6.	**Web server's capacity is lower than application server.**	**While application server's capacity is higher than web server.**

- **What is Catalina?**

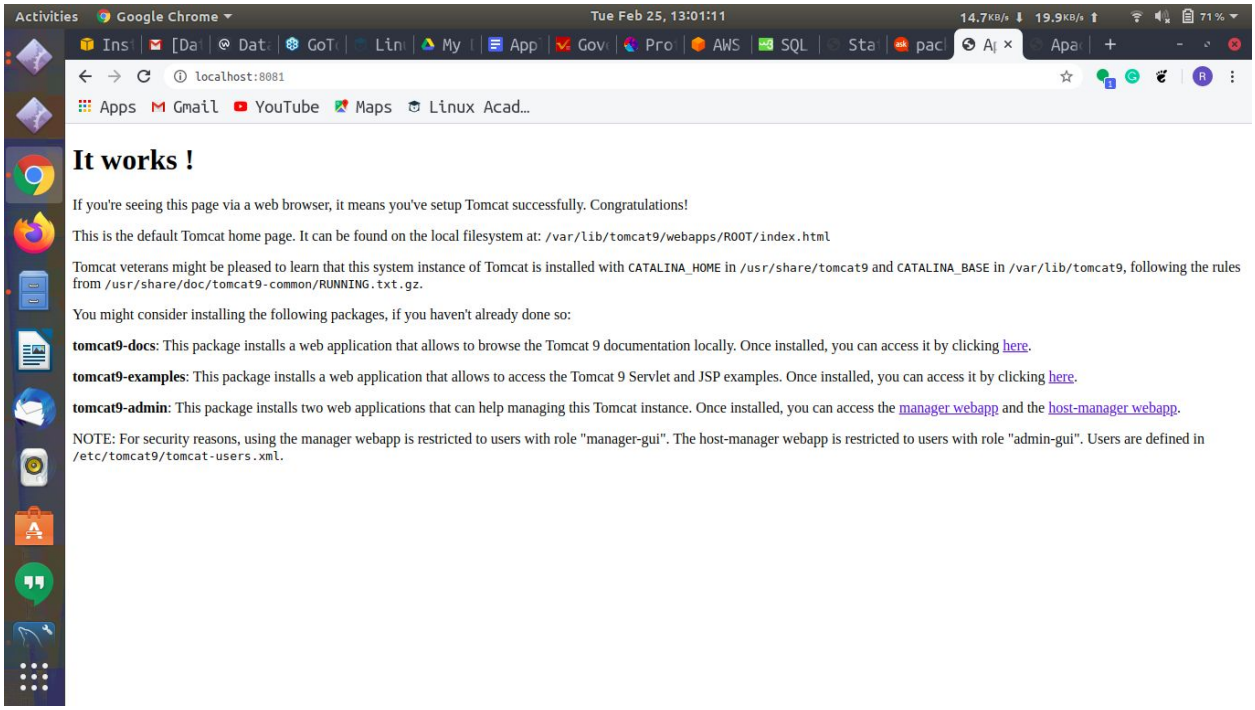
  Catalina is Tomcat's servlet container. Catalina implements Sun Microsystems' specifications for servlet and JavaServer Pages (JSP). In Tomcat, a Realm element represents a "database" of usernames, passwords, and roles (similar to Unix groups) assigned to those users. Different implementations of Realm allow Catalina to be integrated into environments where such authentication information is already being created and maintained, and then use that information to implement Container Managed Security as described in the Servlet Specification.

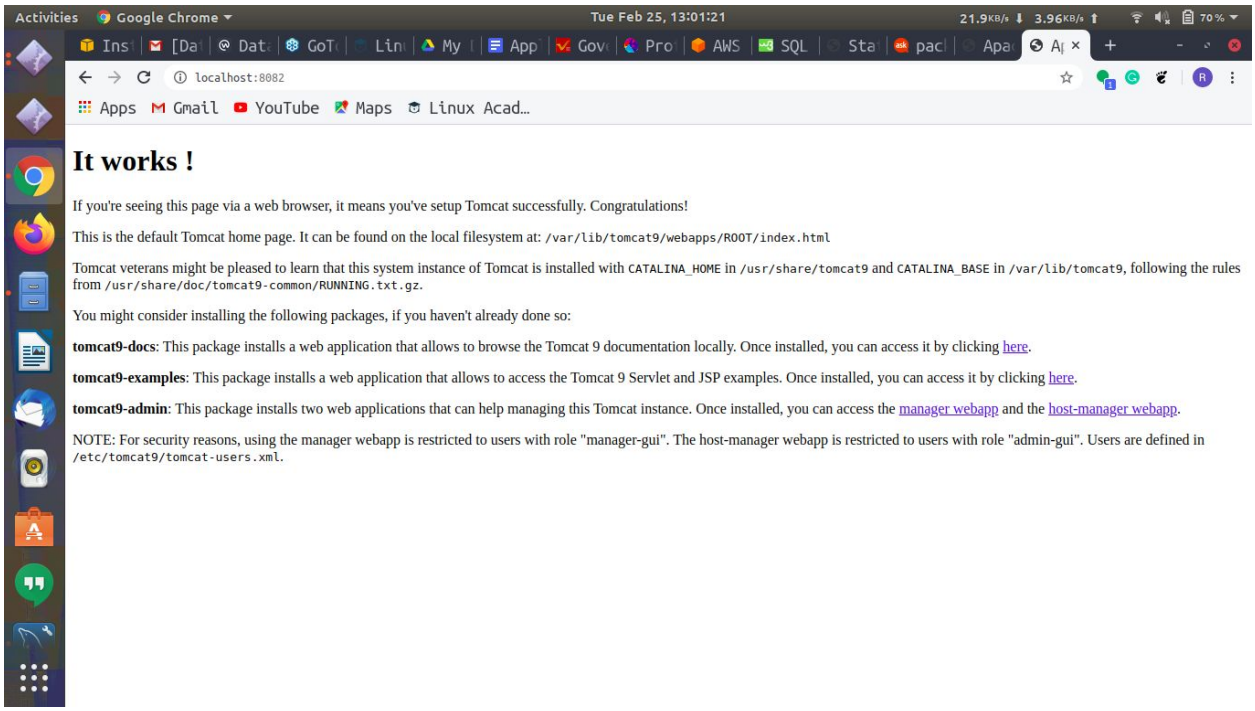
- **Describe tomcat directory structure.**
- **bin**: for Tomcat's binaries and startup scripts.
- **`conf`**: global configuration applicable to all the webapps. The default installation provides:
  - One Policy File: `catalina.policy` for specifying security policy.
  - Two Properties Files: `catalina.properties` and `logging.properties`,
  - Four Configuration XML Files: `server.xml` (Tomcat main configuration file), `web.xml` (global web application

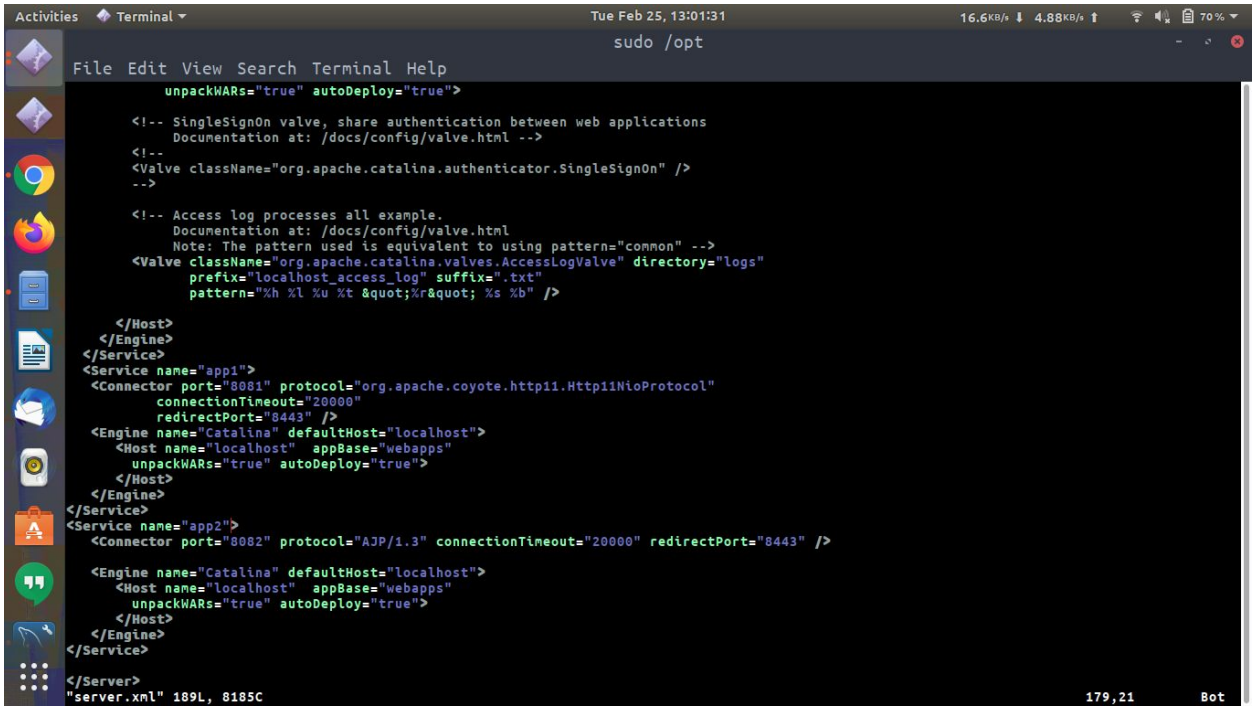
deployment descriptors), `context.xml` (global Tomcat-specific configuration options) and `tomcat-users.xml` (a database of user, password and role for authentication and access control).

- The `conf` also contains a sub-directory for each engine, e.g., `Catalina`, which in turn contains a sub-sub-directory for each of its hosts, e.g., `localhost`. You can place the host-specific context information (similar to `context.xml`, but named as *`webapp`*`.xml` for each webapp under the host).
- **`lib`**: Keeps the JAR-file that are available to all webapps. The default installation includes `servlet-api.jar` (Servlet), `jasper.jar` (JSP) and `jasper-el.jar` (EL). You may also keep the JAR files of external packages here, such as MySQL JDBC driver (`mysql-connector-java-5.1.{xx}-bin.jar`) and JSTL (`jstl.jar` and `standard.jar`).
- **`logs`**: contains the engine log file `Catalina.{yyyy-mm-dd}.log`, host log file `localhost.{yyyy-mm-dd}.log`, and other application log files such as `manager` and `host-manager`. The access log (created by the `AccessLogValve`) is also kept here.
- **`webapps`**: the default `appBase` - web applications base directory of the host `localhost`.
- **`work`**: contains the translated servlet source files and classes of JSP/JSF. Organized in hierarchy of engine name (`Catalina` **`bin`**), host name (`localhost`), webapp name, followed by the Java classes package structure.
- **`temp`**: temporary files.

- **Run multiple services on different ports with different connectors (AJP/HTTP) on same tomcat installation.**


Activities
Google Chrome
Tue Feb 25, 13:01:11
localhost:8081
Apps
Gmail
YouTube
Maps
Linux Acad...
It works !
If you're seeing this page via a web browser, it means you've setup Tomcat successfully. Congratulations!
This is the default Tomcat home page. It can be found on the local filesystem at: /var/lib/tomcat9/webapps/ROOT/index.html
Tomcat veterans might be pleased to learn that this system instance of Tomcat is installed with CATALINA_HOME in /usr/share/tomcat9 and CATALINA_BASE in /var/lib/tomcat9, following the rules from /usr/share/doc/tomcat9-common/RUNNING.txt.gz.
You might consider installing the following packages, if you haven't already done so:
tomcat9-docs: This package installs a web application that allows to browse the Tomcat 9 documentation locally. Once installed, you can access it by clicking here.
tomcat9-examples: This package installs a web application that allows to access the Tomcat 9 Servlet and JSP examples. Once installed, you can access it by clicking here.
tomcat9-admin: This package installs two web applications that can help managing this Tomcat instance. Once installed, you can access the manager webapp and the host-manager webapp.
NOTE: For security reasons, using the manager webapp is restricted to users with role "manager-gui". The host-manager webapp is restricted to users with role "admin-gui". Users are defined in /etc/tomcat9/tomcat-users.xml.

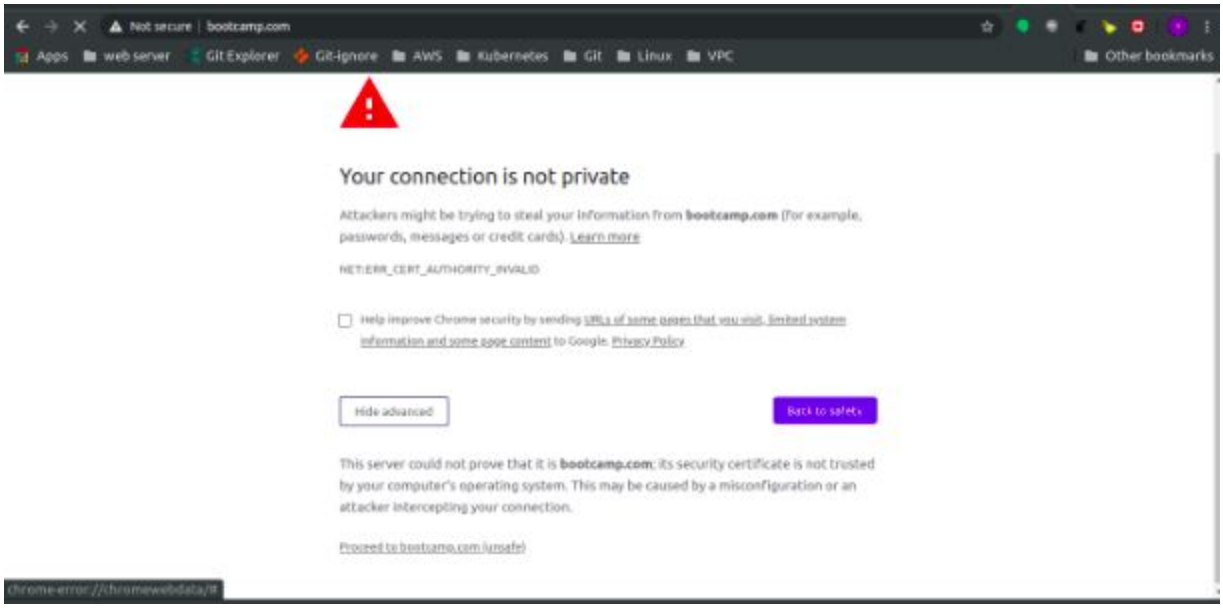

Activities
Google Chrome
Tue Feb 25, 13:01:21
localhost:8082
Apps
Gmail
YouTube
Maps
Linux Acad...
It works !
If you're seeing this page via a web browser, it means you've setup Tomcat successfully. Congratulations!
This is the default Tomcat home page. It can be found on the local filesystem at: /var/lib/tomcat9/webapps/ROOT/index.html
Tomcat veterans might be pleased to learn that this system instance of Tomcat is installed with CATALINA_HOME in /usr/share/tomcat9 and CATALINA_BASE in /var/lib/tomcat9, following the rules from /usr/share/doc/tomcat9-common/RUNNING.txt.gz.
You might consider installing the following packages, if you haven't already done so:
tomcat9-docs: This package installs a web application that allows to browse the Tomcat 9 documentation locally. Once installed, you can access it by clicking here.
tomcat9-examples: This package installs a web application that allows to access the Tomcat 9 Servlet and JSP examples. Once installed, you can access it by clicking here.
tomcat9-admin: This package installs two web applications that can help managing this Tomcat instance. Once installed, you can access the manager webapp and the host-manager webapp.
NOTE: For security reasons, using the manager webapp is restricted to users with role "manager-gui". The host-manager webapp is restricted to users with role "admin-gui". Users are defined in /etc/tomcat9/tomcat-users.xml.

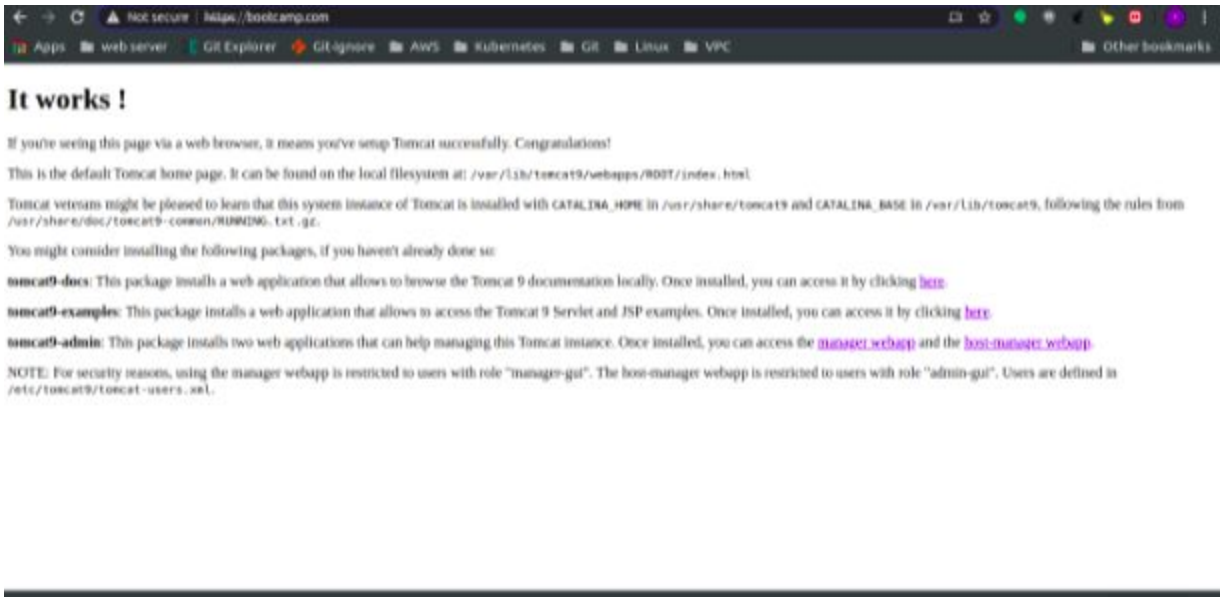


- **Use nginx as reverse proxy for tomcat application.**
    - **Setup self signed certificate on that nginx for bootcamp.com.**

```
server {
    listen 80;
    server_name bootcamp.com;
    return 302 https://$host$request_uri;
}
server {
    listen 443 ssl ;
    server_name bootcamp.com;

    ssl_certificate /etc/nginx/ssl/public.pem;
    ssl_certificate_key /etc/nginx/ssl/private.key;
    location / {
                proxy_pass http://localhost:8080;
            }
}
```





- **What is the difference between proxy_pass & proxy_pass reverse?**
  In forward proxy client identity is hidden when they request(ex. VPN)
  In Reverse Proxy server identity is hidden, (ex. Client will hit the Nginx and it will serve content from Tomcat).