Tomcat容器的目录情况:  
bin : 存放启动、关闭、查看版本等脚本   
conf : 包含不同的配置文件,server.xml(Tomcat的主要配置文件)和web.xml   
work : 存放jsp编译后产生的class文件   
webapp: 存放应用程序示例，以后你要部署的应用程序也要放到此目录   
logs : 存放日志文件   
lib/japser/common : 这三个目录主要存放tomcat所需的jar文件

# Tomcat中内置的Manager WebApplication的功能:

* Deploy a new web application from the uploaded contents of a WAR file.
* Deploy a new web application, on a specified context path, from the server file system.
* List the currently deployed web applications, as well as the sessions that are currently active for those web apps.
* Reload an existing web application, to reflect changes in the contents of /WEB-INF/classes or /WEB-INF/lib.
* List the OS and JVM property values.
* List the available global JNDI resources, for use in deployment tools that are preparing <ResourceLink> elements nested in a <Context> deployment description.
* Start a stopped application (thus making it available again).
* Stop an existing application (so that it becomes unavailable), but do not undeploy it.
* Undeploy a deployed web application and delete its document base directory (unless it was deployed from file system).

注:使用该应用的时候得注意CSRF

# Tomcat的server.xml配置文件:

|  |  |  |
| --- | --- | --- |
| 元素名 | 属性 | 解释 |
| server | port | 指定一个端口，这个端口负责监听关闭tomcat 的请求 |
| shutdown | 指定向端口发送的命令字符串 |
| service | name | 指定service 的名字 |
| Connector ( 表示客户端和service之间的连接) | port | 指定服务器端要创建的端口号，并在这个断口监听来自客户端的请求 |
| minProcessors | 服务器启动时创建的处理请求的线程数 |
| maxProcessors | 最大可以创建的处理请求的线程数 |
| enableLookups | 如果为true ，则可以通过调用request.getRemoteHost() 进行DNS 查询来得到远程客户端的实际主机名，若为false 则不进行DNS 查询，而是返回其ip 地址 |
| redirectPort | 指定服务器正在处理http 请求时收到了一个SSL 传输请求后重定向的端口号 |
| acceptCount | 指定当所有可以使用的处理请求的线程数都被使用时，可以放到处理队列中的请求数，超过这个数的请求将不予处理 |
| connectionTimeout | 指定超时的时间数( 以毫秒为单位) |
| Engine ( 表示指定service 中的请求处理机，接收和处理来自Connector的请求) | defaultHost | 指定缺省的处理请求的主机名，它至少与其中的一个host 元素的name 属性值是一样的 |
| Context ( 表示一个web 应用程序，通常为WAR 文件，关于WAR 的具体信息见servlet 规范) | docBase | 应用程序的路径或者是WAR 文件存放的路径 |
| path | 表示此web 应用程序的url 的前缀，这样请求的url为[http://localhost:8080/path/](http://localhost:8080/path/" \t "_blank)\*\*\*\* |
| reloadable | 这个属性非常重要，如果为true ，则tomcat 会自动检测应用程序的/WEB-INF/lib 和/WEB-INF/classes 目录的变化，自动装载新的应用程序，我们可以在不重起tomcat 的情况下改变应用程序 |
| host ( 表示一个虚拟主机 ) | name | 指定主机名 |
| appBase | 应用程序基本目录，即存放应用程序的目录 |
| unpackWARs | 如果为true ，则tomcat 会自动将WAR 文件解压，否则不解压，直接从WAR 文件中运行应用程序 |
| Logger ( 表示日志，调试和错误信息) | className | 指定logger 使用的类名，此类必须实现org.apache.catalina.Logger 接口 |
| prefix | 指定log 文件的前缀 |
| suffix | 指定log 文件的后缀 |
| timestamp | 如果为true ，则log 文件名中要加入时间，如下例:localhost\_log.2001-10-04.txt |
| Realm ( 表示存放用户名，密码及role 的数据库) | className | 指定Realm 使用的类名，此类必须实现org.apache.catalina.Realm 接口 |
| Valve ( 功能与Logger 差不多，其prefix 和suffix 属性解释和Logger 中的一样) | className | 指定Valve 使用的类名，如用org.apache.catalina.valves.AccessLogValve 类可以记录应用程序的访问信息 |
| directory | 指定log 文件存放的位置 |
| pattern | 有两个值，common 方式记录远程主机名或ip 地址，用户名，日期，第一行请求的字符串，HTTP 响应代码，发送的字节数。combined 方式比common 方式记录的值更多 |

# Tomcat的tomcat-users.xml配置文件:

每个应用的Web.xml都可以配置不同的角色,然后设置每个角色在该应用中的权限,然后在tomcat-users.xml中配置上同名角色,然后为用户分配上对应的角色.

这个文件是用来配置TomcatWebApplications的访问用户和访问权限(角色)

例如Tomcat的Manager WebApplication的web.xml中就配置了四个角色:

manager-gui — Access to the HTML interface.

manager-status — Access to the "Server Status" page only.

manager-script — Access to the tools-friendly plain text interface that is described in this document, and to the "Server Status" page.

manager-jmx — Access to JMX proxy interface and to the "Server Status" page.