# Máster Interuniversitario en Seguridad de las TIC (MISTIC)

## Identidad Digital – 2o Semestre, Curso 2019/2020

## Segunda Práctica (PRAC2)

**Requisitos previos**

Conocimientos: Se requieren ciertos conocimientos de Java, JSPs, Servlets y HTML.

Software: Virtual Box + Debian; Java SE Development Kit (JDK); servlet container Apache Tomcat; Spring Security; Central Authentication Service (CAS); OpenLDAP.

**Recursos**

Para resolver la práctica podéis consultar los materiales de la asignatura que están en la sección de Recursos del aula o los materiales que, opcionalmente, os facilite el consultor; también podéis consultar fuentes externas como Internet, libros, etc.

Para cualquier duda y/o aclaración sobre el enunciado, tenéis que dirigiros al consultor responsable de vuestra aula.

**Formato y fecha de entrega**

Se tiene que entregar todos los archivos de la práctica en un archivo comprimido en formato **ZIP**.

La fecha límite de entrega es el: **8 de diciembre de 2019, 23:59 AoE (Anywhere on Earth)**

Las prácticas entregadas fuera del plazo establecido no se puntuarán y constarán como no presentadas.

**Criterios de valoración**

**Es necesario que expliquéis los pasos que habéis realizado para desarrollar la solución.**

En la memoria se valorará la claridad de la información aportada, el estilo comunicativo empleado, y la capacidad de síntesis.

**Si la práctica contiene preguntas teóricas, las respuestas sin justificación, las copias de una fuente de información y/o que no contengan las referencias utilizadas, no recibirán ninguna puntuación.**

**Si se detecta una copia de la práctica, esta será evaluada con una D y la incidencia será notificada a la dirección del Máster para actuar en consecuencia.**

El valor de esta práctica es del 50% de la nota global de prácticas de la asignatura y, por lo tanto, corresponde al 30% de la nota final de la evaluación continuada.

# Ejercicio 1. Crear un directorio de empresa *(2 puntos)*

En este ejercicio hay que crear un **directorio** usando **OpenLDAP** para la supuesta empresa de importación y distribución utilizada en la PRAC1. El directorio almacenará los datos básicos de trabajadores, proveedores y clientes.

## 1.1 Especificaciones

Para realizar la práctica emplearemos el organigrama de una empresa que denominaremos "**Insectores**", su página web o dominio es "**insectores.com**".

La empresa tiene los trabajadores siguientes:

| **Trabajador (cn)** | **Actividad (title)** | **Identificador de usuario (uid)** | **Password** | **Ubicación (roomNumber)** | **E-mail (mail)** |
|---|---|---|---|---|---|
| *Ender Wiggin* | *Gerente* | *ender* | *dragon78* | *Room1* | *ender@insectores.com* |
| *Petra Arkanian* | *Jefe de Ventas* | *petra* | *phoenix8* | *Room2* | *petra@insectores.com* |
| *Bonzo Madrid* | *Jefe de Personal* | *bonzo* | *salamand* | *Room3* | *bonzo@insectores.com* |
| *Carn Carby* | *Jefe de Compras* | *carn* | *rabbit78* | *Room4* | *carn@insectores.com* |
| *Dink Meeker* | *Auditor* | *dink* | *rat45678* | *Room5* | *dink@insectores.com* |

La empresa tiene los clientes siguientes:

| **Cliente (cn)** | **Empresa** | **Actividad (title)** | **Identificador de usuario (uid)** | **Password** | **Dirección** | **Teléfono (telephoneNumber)** | **E-mail (mail)** |
|---|---|---|---|---|---|---|---|
| Hyrum Graff | Battle School, INC | Director | hyrum | c0l0n3l | Lagrange P, Outearth | +1 (222) 123-4567 | hyrum@battles.com |
| Mazer Rackham | Command School, LTD | Director | mazer | 4ns1bl3 | Sol System, Eros | +1 (333) 987-6543 | mazer@commands.com |

La empresa tiene los proveedores siguientes:

| **Proveedor (cn)** | **Empresa** | **Actividad (title)** | **Identificador de usuario (uid)** | **Password** | **Dirección** | **Teléfono (phoneNumber)** | **E-mail (mail)** |
|---|---|---|---|---|---|---|---|
| Abra Tolo | Formic Machines,CO | Jefe Comercial | abra | f41ryl4nd | Colony P, Shakespeare | +1 (444) 345-6789 | abra@formicm.com |
| James Dap | Launch Group, LLC | Jefe Comercial | james | s3rg34nt | Bschool St, Earth | +1 (555) 321-9876 | james@launchg.com |

La estructura de la empresa consta de **los siguientes departamentos** a considerar: Dirección, Ventas, Personal y Compras. **Los trabajadores a cargo de cada departamento** son los siguientes: **Ender/Dirección, Petra/Ventas, Bonzo/Personal, Carn/Compras**. Esto tiene que quedar reflejado en el directorio.

## 1.2 Configurar un servidor OpenLDAP en el entorno de trabajo de la asignatura (Virtual Box + Debian)

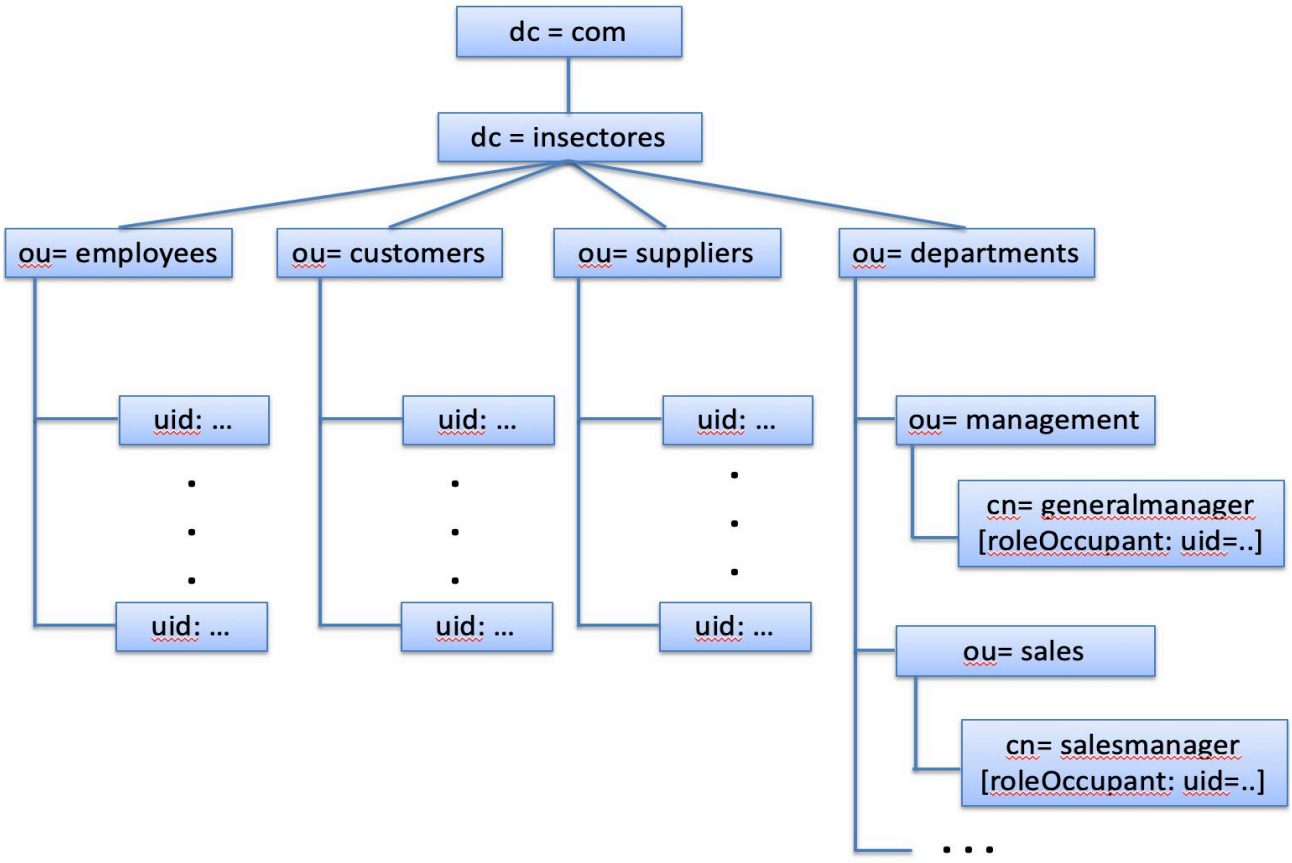
Primero configuraremos un **servidor OpenLDAP** considerando que el dominio de la empresa es "**soledad100.com**".

Se recomienda seguir el manual de instalación de OpenLDAP en Debian facilitado por el consultor.

Para facilitar el proceso de corrección, usad **el password "admin"** del servidor OpenLDAP desplegado, tal y cómo se indica a la documentación de apoyo.

## 1.3 Crear y cargar la estructura del directorio (0,5 puntos)

A continuación, se muestra en alto nivel **la estructura del directorio** que se quiere implementar mediante OpenLDAP:




Puntos que destacar:

- Usamos "uid" para construir el "dn" de cada persona de la organización.
- Agrupamos todos los trabajadores dentro de una *organizational unit* general. Esto nos será útil para el Ejercicio2 de esta práctica, pero también implica que tendremos que indicar por medio de un **atributo** dentro de cada instancia de trabajador a qué departamento pertenece.
- En cada departamento se indica que hay un encargado. Enlazamos cada "encargado de departamento" con la persona que se encarga de esta responsabilidad.

Algunos recursos que pueden ser útiles para entender la estructura planteada y como implementarla:

ldapman_introduction.pdf
http://www.zytrax.com/books/ldap/ch5/index.html#step1-dit
http://www.zytrax.com/books/ldap/apa/structure.html

*Crea un fichero LDIF que permita crear las organizational units (ou) necesarias para construir la estructura de directorio requerido. El fichero debe tener el nombre:*

- *Ex1_3.ldif*

*Proporciona un script que cargue el fichero .ldif anterior. El script debe tener el nombre:*

- *Ex1_3.sh*

## 1.4 Cargar datos (1,5 puntos)

En esta sección crearemos los usuarios de la empresa y llenaremos la estructura creada en su punto anterior. Se tienen que usar los ObjectClass y attributes adecuados en cada caso. Idealmente, **los passwords no se tienen que almacenar en claro en el directorio**, sino que tenemos que guardar su **hash**. **NOTA:** si los passwords se guardan en claro el apartado **se valorará con un máximo de 0,75 puntos**.

*Crea un fichero LDIF que contenga las sentencias para añadir los trabajadores de la empresa al directorio junto con la información básica proporcionada. El fichero tiene que tener el nombre siguiente:*

- *Ex1_4_1.ldif*

*Crea un fichero LDIF que contenga las sentencias para añadir los clientes y proveedores de la empresa al directorio junto con la información básica proporcionada. El fichero tiene que tener el nombre siguiente:*

- *Ex1_4_2.ldif*

*Crea un fichero LDIF que contenga las sentencias para crear los encargados de departamento y enlazarlos con los trabajadores correspondientes. El fichero tiene que tener el nombre siguiente:*

- *Ex1_4_3.ldif*

*Proporciona un script que cargue los ficheros .ldif anteriores. El script tiene que tener el nombre siguiente:*

- *Ex1_4.sh*

## 1.5 Entrega del ejercicio

La entrega de este ejercicio debe incluir los ficheros siguientes:

- Ficheros LDIF: a continuación, se detallan todos los archivos que hay que entregar y se han indicado a lo largo del enunciado de la práctica.
    - Ex1_3.ldif
    - Ex1_4_1.ldif
    - Ex1_4_2.ldif
    - Ex1_4_3.ldif
- Scripts: Deben adjuntarse todos los ficheros con los comandos que se ha especificado a lo largo del enunciado de la práctica.
    - Ex1_3.sh
    - Ex1_4.sh

**Toda la información anterior se tiene que comprimir en un fichero ZIP llamado "Ejercicio1" el cual se comprimirá dentro del zip principal** que se tiene que entregar mediante la "Entrega y registro de EC".

## 1.6 Corrección (**IMPORTANTE**)

Para verificar el funcionamiento del ejercicio, se realizará una instalación de OpenLDAP en el entorno de trabajo de la asignatura (Virtual Box + Debian). A continuación, se ejecutarán los ficheros LDIF y los scripts facilitados para verificar los diversos puntos requeridos. Aseguraos que este proceso se puede seguir convenientemente en el entorno fijado y que se respetan los nombres de los ficheros (LDIF y scripts). El objetivo es evitar problemas en el proceso de evaluación.

# Ejercicio 2. Securizar una aplicación Web con autenticación vía servidor single sign-on. *(8 puntos)*

## 2.1 Introducción

El objetivo de este ejercicio es **securizar la aplicación Web "IDwebapp"** la cual se entrega junto con este enunciado (y que ya se usó a la PRAC1) **añadiendo un control de acceso basado en roles (RBAC)** que haga que ciertas operaciones sólo puedan ser ejecutadas por los usuarios con las credenciales y roles adecuados.

En este caso, **la autenticación se realizará usando un servicio externo de single sign-on** mientras que **la autorización se hará localmente** en la misma aplicación Web.

La aplicación se basa en la supuesta empresa de importación y distribución que se ha usado en la PRAC1 y el ejercicio1 de la PRAC2.

Los **trabajadores de la empresa** y sus datos ya se han presentado en el Ejercicio1 de esta práctica (también se usaron en la PRAC1).

Los roles necesarios para gestionar la aplicación son los siguientes:

| ***Rol*** | ***Descripción*** |
|---|---|
| *AC* | *Autoriza las compras.* |
| *GCFP* | *Gestión de clientes, facturas y presupuestos.* |
| *GNT* | *Gestión de nóminas y de trabajadores.* |
| *GCP* | *Gestión de compras y de proveedores.* |
| *A* | *Auditor* |

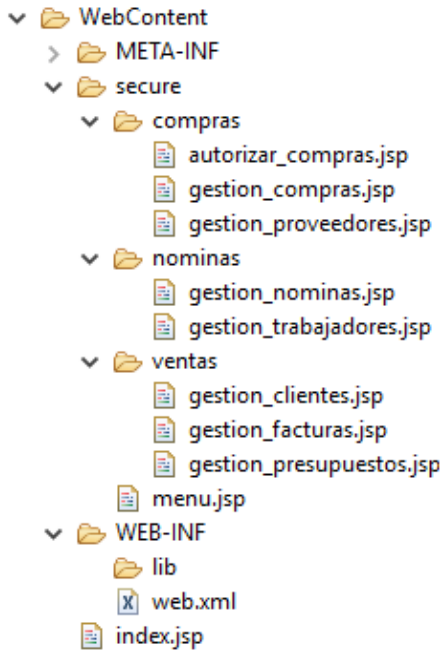
La asignación de roles a cada usuario es la siguiente:

| ***Personal de la empresa*** | ***Rol/es*** |
|---|---|
| *Ender Wiggin* | *AC, GCFP, GNT, GCP* |
| *Petra Arkanian* | *GCFP* |
| *Bonzo Madrid* | *GNT* |
| *Carn Carby* | *GCP* |
| *Dink Meeker* | *A* |

Los roles necesarios para ejecutar las operaciones de cada módulo son los siguientes:

| ***Módulo*** | ***Operación*** | ***Rol*** |
|---|---|---|
| *Personal* | *Gestión de nóminas* | *GNT* |
| | *Gestión de trabajadores* | *GNT, A* |
| *Compras* | *Autorizar compras* | *AC* |
| | *Gestión de compras* | *GCP* |
| | *Gestión de proveedores* | *GCP, A* |
| *Ventas* | *Gestión de clientes* | *GCFP* |
| | *Gestión de facturas* | *GCFP* |
| | *Gestión de presupuestos* | *GCFP, A* |

La aplicación web proporcionada tiene una primera página de portada (“index.jsp”) con un link al menú de la aplicación (menu.jsp) donde se muestran todas las operaciones disponibles según los 3 módulos indicados anteriormente. Cada página de operación tiene un link para volver al menú principal. La jerarquía de la aplicación web es la siguiente:



## 2.2 Configuración inicial

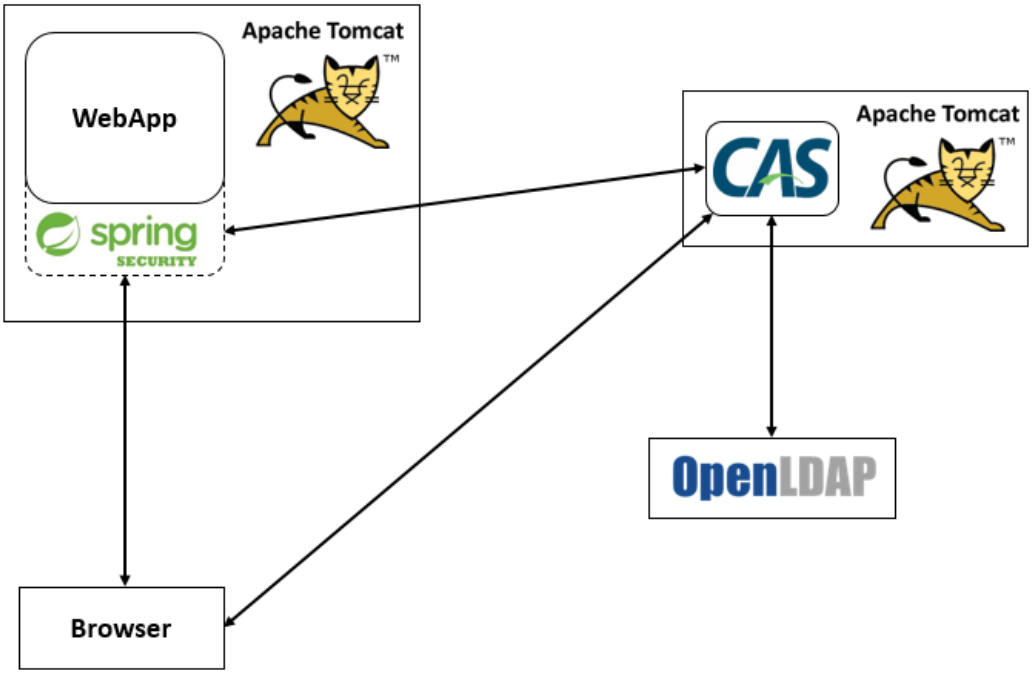
En este ejercicio básicamente **eliminaremos el módulo JAAS** que se usó en la PRAC1 y configuraremos la aplicación para que **externalice la etapa de autenticación** utilizando un **servidor CAS single sign-on**. La etapa de autorización necesaria para trabajar con los roles de usuario se implementará a nivel local usando el framework **Spring Security**.

**Como punto de partida**, se proporciona junto con este enunciado **un manual de despliegue de un servidor CAS single sign-on y una aplicación web sencilla donde se usa Spring Security para controlar el acceso a un recurso** (.jsp).

Tal y cómo se indica en el manual, la aplicación web CASificada de ejemplo está configurada para trabajar únicamente con un usuario llamado “casuser”. Este usuario tiene un rol asignado que no le permite acceder a la carpeta “extreme” donde hay otro recurso protegido. Un primer paso para familiarizarse con el funcionamiento de la aplicación de ejemplo sería crear un nuevo usuario con los roles necesarios para acceder a los dos recursos protegidos.

## 2.3 Puntos a tratar para completar el ejercicio

A continuación, se muestra una imagen de la arquitectura que hay que construir:



### 2.3.1 Aplicación web securizada con Spring Security+CAS *(3 puntos)*

Partiendo de la aplicación web proporcionada ("IDwebapp"), **es <u>obligatorio</u> tratar los puntos siguientes**:

- La página de portada "index.jsp" será accesible por todo el mundo, ya sean usuarios como no usuarios.

- La página de menú "menu.jsp" será accesible sólo para los usuarios de la aplicación web definidos anteriormente

- La página de menú "menu.jsp" mostrará el login del usuario autenticado, su nombre real y su rol

- La página de menú "menu.jsp" mostrará links a todas las operaciones existentes a la aplicación web pero mostrará, usando un icono o similar, qué operaciones puede realizar el usuario autenticado y qué no.

- Seguir un link a la página de una operación no permitida mostrará a una página donde se indicará el intento de acceso denegado.

- La página de menú "menu.jsp" tendrá un link que permitirá al usuario autenticado hacer **logout** y cerrar su sesión, devolviendo a la página de portada "index.jsp".

Debe considerarse que, idealmente, todos estos puntos ya se habrán tratado a la versión securizada de "IDwebapp" resultante de la PRAC1, por lo tanto, no habrá que hacerlos a no ser que el estudiante no haya hecho la PRAC1 o no lo haya resuelto correctamente.

Adicionalmente, en este ejercicio, **es <u>obligatorio</u> tratar los puntos siguientes**:

- El **control de acceso a las diferentes páginas** de menú y operaciones se realizará mediante **Spring Security, el cual se debe integrar a la aplicación web**. Un usuario sólo debe poder acceder a las operaciones correspondientes a sus roles.

- **Spring Security conectará con el servidor CAS** para autenticar al usuario que intente acceder a una zona protegida.

- El **cierre de sesión** implica hacer **un log-out al servidor CAS.**

- No se almacenará ningún password de usuario junto a la aplicación web. **Los passwords quedan íntegramente bajo el control del servidor CAS**. La aplicación web (o más concretamente **el Spring Security** integrado en la aplicación web) sí tendrá acceso a los roles disponibles y a la relación entre usuarios y roles asociados.

### 2.3.2 Utilización de CAS + OpenLDAP para realizar la autenticación *(2,5 puntos)*

El servidor CAS se conectará a un **servidor OpenLDAP** para consultar las credenciales de los usuarios y **realizar su autenticación**.

El directorio LDAP que se usará es el mismo que se ha **implementado en el Ejercicio1.**

Para que el servidor CAS pueda interactuar con un servidor OpenLDAP será necesario **añadir ciertas librerías adicionales (.jar)** en la carpeta \lib del servidor CAS (...\cas-server-webapp-4.0.0\WEB-INF\lib) y modificar **la configuración donde sea necesario**. Se recomienda la siguiente documentación:

https://apereo.github.io/cas/4.0.x/installation/LDAP-Authentication.html

### 2.3.3 Utilización SSL (y certificados) para proteger las comunicaciones *(2,5 puntos)*

Utilizaremos SSL (y certificados) para securizar todas las comunicaciones entre el browser, la aplicación CASificada, el servidor CAS y el servidor OpenLDAP. Algunos recursos web que pueden ser útiles son:

https://wiki.jasig.org/display/casum/demo
http://tomcat.apache.org/tomcat-8.0-doc/ssl-howto.html
https://wiki.jasig.org/display/CASC/JA-SIG+Java+Client+Simple+WebApp+Sample

## 2.4 Entrega

La entrega del ejercicio tiene que incluir los 3 recursos siguientes:

- Documento en PDF que contenga:
  - **Explicación de los pasos que habéis realizado para desarrollar las distintas modificaciones** (utilización de CAS+OpenLDAP o la alternativa elegida, conexiones securizadas con SSL, etc.).
  - URL necesaria para acceder a la aplicación.
- Archivos de la aplicación
  - **La carpeta *entera* del apache-tomcat** correspondiente a la **aplicación web securizada con Spring Security+CAS** con todos los ficheros necesarios para ejecutar la aplicación.
  - **Vuestra carpeta *entera* del apache-tomcat** correspondiente al **servidor CAS** con todos los ficheros necesarios para ejecutar el servidor.
  - **El código fuente de todas las clases que se hayan implementado.** No se evaluarán ejercicios donde no se proporcione el código fuente.
  - Carpeta con los certificados necesarios parar securizar el sistema con SSL. Aquí se puede incluir cualquier fichero que el estudiante crea necesario para conseguir esto. Esto también incluye la configuración bajo SSL del servidor OpenLDAP. **Los pasos necesarios para realizar las diferentes configuraciones deben detallarse en el PDF requerido**.
- **Juego de pruebas**
  - **Video 1: un video que muestre el funcionamiento de la aplicación.** Se tiene que poder seguir el proceso desde la página inicial de login, hasta la pantalla (o pantallas) donde se muestran las diversas operaciones que pueden realizar los usuarios escogidos como representativos del funcionamiento de la aplicación. En concreto, no debería faltar el funcionamiento de la aplicación web para los usuarios con roles Gerente, Jefe de Ventas y Auditor.
  - **Video 2: otro video mostrando las partes más representativas del código modificado/generado**. No es necesario ser muy detallado, pero si mostrar la ubicación de ese código y su funcionamiento general. Si este video es lo suficientemente explicativo el PDF requerido anteriormente no es necesario que sea tan exhaustivo. Concretamente, no debería faltar la explicación de:
    - Control de acceso a los recursos
    - Logout
    - Definicion de roles
    - Configuración LDAP
    - Configuración CAS
    - Definición de certificados

**Toda la información anterior se tiene que comprimir en un fichero ZIP** que se tiene que entregar mediante la “Entrega y Registro de EC”. Todo lo requerido anteriormente es de **entrega obligatoria.** Las prácticas entregadas sin los recursos requeridos **no serán evaluadas**.

## 2.5 Corrección (**IMPORTANTE**)

Para verificar el funcionamiento de la práctica, se cogerán las dos carpetas apache-tomcat facilitadas y se pondrán en la carpeta /opt del entorno de trabajo de la asignatura (Virtual Box + Debian). En caso de ser necesario, se aplicarán las indicaciones facilitadas por el alumno en el PDF entregado (por ejemplo, respecto al posicionamiento de certificados, modificaciones al OpenLDAP, etc.). Tras esto, se iniciarán los tomcats y se introducirá en el browser la URL de acceso a la aplicación facilitada.

Por favor, antes de hacer la entrega, seguid vosotros mismos este proceso y aseguraos que todo se ejecuta convenientemente en el entorno fijado para evitar problemas con el proceso de evaluación. **Comprobad también que vuestra entrega contiene todos los puntos indicados a la sección 2.4 Entrega.**