viafirma fortress

manual de instalación

Tabla de contenido

Manual de instalación de Viafirma Fortress

El presente documento pretende ser una guía de instalación para el proyecto **Viafirma Fortress**. Información dirigida a perfiles técnicos y con solvencia en la instalación y despliegues de aplicaciones JEE.

Si lo desea puede descargar este manual en pdf aquí.

Viafirma Fortress roadmap

- Fortress Dashboard: https://www.viafirma.com/fortress/es/backlog/
- Fortress Desktop: https://www.viafirma.com/fortress/es/backlog-desktop/

Componentes instalación

Entregables

Los ficheros necesarios para llevar a cabo la instalación de la aplicación son los siguientes:

Cliente

No es necesaria la instalación de ningún tipo de fichero en el entorno cliente.

Servidor

Se entregarán los siguientes ficheros de configuración del entorno servidor:

- Los ficheros de configuración para el entorno (.zip):

Archivo: viafirma-fortress_v.4.6_cfg.zip

- La aplicación empaquetada (archivo WAR):

Archivo: viafirma-fortress_v.4.6.war

- Ficheros con scripts de creación de roles/usuarios y esquema de BBDD.

Archivo: viafirma-fortress_v.4.6_sql.zip

- Otros ficheros: driver Jdbc, certificados raíz de CAs a instalar, fichero licencia properties, fichero logback.xml.

Archivo: viafirma-fortress_v.4.6_ent.zip

Base de datos

Base de datos que gestiona la información de los usuarios, clientes, certificados asociados a los usuarios e factores de autenticación en los que se encuentra enrolado o se puede enrolar un usuario.

Los scripts que se facilitan incluyen:

- Creación de las tablas contenidas en el esquema de BD
- Creación de roles
- Creación de usuarios

- scripts sql
    - oracle
        - **schema-oracle.sql**
        - **data-oracle.sql**
    - postgres
        - **schema-postgresql.sql**
        - **data-postgresql.sql**

          Nota. Para realizar la migración de versiones anteriores de Viafirma Fortress que empleen PostgreSQL será necesario ejecutar el siguiente script de actualización:

        - **update-postgresql.sql-version_ini-version_end.sql**

- Drivers jdbc
    - **ojdbc6.jar** Driver de oracle
    - **postgresql-9.3-1102.jdbc41.jar** Driver de postgresql

Informes

No aplica.

Otros

No aplica

```
Última revisión: 12 Ene 2022
```

Requisitos

Los requisitos asociados a la instalación de **Viafirma Fortress** son:

Versión de JAVA

El aplicativo está optimizado para emplear la versión de JAVA 1.8.x de Oracle.

Para poder realizar encriptaciones compleja se necesitará instalar o sobrescribir las siguientes librerías:

```
local_policy.jar
US_export_policy.jar
```

ubicadas en:

```
<JAVA_HOME>/jre/lib/security
```

Servidor de aplicaciones

El Servidor de aplicaciones debe soportar Java 8. El aplicativo debe ser desplegado en Apache Tomcat 8 o superior.

Recomendaciones de Seguridad

Viafirma Fortress debe ser instalado en una vesión 8.x de Apache Tomcat siguiendo las pautas de seguridad recomendadas para la configuración antes de su publicación.

- https://tomcat.apache.org/tomcat-8.0-doc/security-howto.html
- https://wiki.owasp.org/index.php/Securing_tomcat

Dimensionamiento del servidor

Se tomará como ejemplo una instalación del backend en un único servidor de aplicaciones. En caso de optar por una instalación en varios servidores, o instalación en clúster, consultar con el soporte técnico de viafirma las opciones recomendadas para cada caso.

- RAM: 8GB
- Micro: recomendado, 2 micros de 6 núcleos cada uno; a 2Ghz.
- Disco: 20 GB (1)
- Logs: estimado 1GB por cada millón de operaciones (1)

(1): los logs incluyen configuración de rotación, por lo que la optimización de estos logs podrá definirse por el administrador del sistema en función a las políticas de espacio en disco deseadas.

Sistemas gestores de BD

Los gestores de Base de datos soportados por defecto serán PostgreSQL y Oracle.

El tamaño ocupado de base de datos depende de la volumetría estimada. El sistema no custodia ficheros en base de datos, al margen de claves públicas o claves wrappeadas de bajo peso. La tabla de mayor peso es la de auditoría (vf_activity), que almacena cualquier transacción realizada en la plataforma, y que es la más susceptible de crecer.

Una base de datos con 150 usuarios, 1.000 certificados y 15.000 operaciones registradas (uso razonable en un mes) pesa 30MB aproximadamente.

El servidor de base de datos debe tener un mínimo de 4GB de RAM para una volumetría como esta, entendiendo en todo caso que no es un servidor dedicado exclusivamente a Inbox.

HSM

La conexión de Fortress con el HSM dependerá de:

- El proveedor del HSM
- La versión del HSM y del cliente asociado, que proporcionará el proveedor del HSM
- Tipología del HSM: en Cloud, tarjeta PCI alojada en su servidor

Fortress se basa en el empleo de conectores con los distintos HSMs actualmente soportados, entre ellos:

- Gemalto/Safenet Luna, versiones Network y PCI
- Thales nCipher
- Realsec, Cryptosec Lan

  y para entornos de prueba conectores basados en JKS.

  Para la conexión con cualquiera de las soluciones indicadas será necesario instalar un cliente proporcionado por el fabricante del HSM.

Procedimiento de instalación

En los siguientes apartados se describe como instalar y configurar **Viafirma Fortress**.

Instalación

La instalación de la aplicación contempla los siguientes apartados:

Cliente

No aplica.

Servidor

Para la instalación de la aplicación en el servidor será necesario:

- Crear datasource
- Incluir el driver JDBC del sistema gestor de BD en Tomcat
- Creación de directorios donde alojará: licencia, logs, auditoria
- Configurar en el servidor de aplicaciones la ruta donde se encuentra el fichero de propiedades del aplicativo

El proceso de despliegue es sencillo. Con el servidor de aplicaciones parado (aunque puede estar levantado, es recomendable pararlo), se copia el war en la carpeta /webapps de Tomcat. Posteriormente se debe arrancar el servidor. Por defecto Tomcat tiene activo el autodeploy, por esta razón cuando encuentre un nuevo WAR en dicha carpeta lanzará el deploy de forma automática, descomprimiendo el WAR en /webapps en un directorio con el mismo nombre al del WAR sin la extensión. La aplicación quedará desplegada bajo el contexto /fortress.

Base de datos

Creación del DataSource

Debe definirse un DataSource para el acceso a la BBDD. Puede definir las propiedades del DataSource en el descriptor de contexto en tomcat/conf/Catalina/.

Dicho dataSource se puede definir de varias formas:

- Definir por su nombre jndi:

El DataSource debe tener por nombre “jdbc/fortress” y seguir un esquema similar al siguiente ejemplo, indicando el username, password y la URL que corresponda:

```
<Resource name="jdbc/fortress" auth="Container"
        type="javax.sql.DataSource" maxActive="100" maxIdle="30" maxWait="10000"
        username="fortress"
        password="********"
        url="jdbc:postgresql:fortress"/>
```

Y configurar en el contexto dicho Datasource

```
<Context>
    <Parameter name="spring.datasource.jndi-name" value="jdbc/fortress" />
</Context>
```

- Definir directamente el datasource en el contexto

```
<Context>
    <Parameter name="spring.datasource.url" value="your URL" />
    <Parameter name="spring.datasource.username" value="your username" />
    <Parameter name="spring.datasource.password" value="your password" />
</Context>
```

- Definir la ruta de acceso a un fichero de properties y establecer aquí su configuración:

```
<Context>
    <Parameter name="spring.config.location" value="/path/to/fortress.properties" />
</Context>
```

Establecer en el fichero de propiedades, las propiedades del datasource, tal y como se indica a continuación

```
# DATASOURCE
spring.datasource.url=jdbc:postgresql://[IP base datos]:[puerto]/fortress
spring.datasource.username=[usuario base de datos]
spring.datasource.password=[contraseña usuario de base de datos]
```

Inclusión del driver de Jdbc

Será necesario copiar el driver JDBC en la carpeta /lib de la instalación de Tomcat:

```
/<TOMCAT_HOME>/lib
```

Informes

No aplica.

Otros

Creación de directorios:

Se deben crear los siguientes directorios para diferentes funcionalidades:

directorio licencia fortress

Fortress como producto, emplea una licencia para permitir su empleo, como se verá al configurar los parámetros de configuración del aplicativo, Fortress utiliza la variable **fortress.license.path** para alojar el path al archivo que contiene la licencia.

```
# LICENSE
fortress.license.path=file:/path/to/fortress.license/fortress.lic
```

directorio log fortress

Fortress emplea logback para generar el log del aplicativo, permitiendo personalizar el mismo según las necesidades del cliente. En caso de no alojar los ficheros de log en el propio servidor de aplicaciones, será necesario crear tantos directorios como se indiquen en el fichero **logback.xml**

Por ejemplo, se puede definir en el archivo logback.xml que los archivos de log se almacénen en la carpeta de logs del servidor de aplicaciones donde se encuentra desplegado fortress

```
<property name="LOGS_FOLDER" value="${catalina.base}/logs" />
```

O por el contrario, crear una estructura de carpetas externas al servidor de aplicaciones e indicar los valores correspondientes en las variables anteriormente definidas.

Directorio proceso de firma

Fortress emplea un directorio base, que emplea en el proceso de firma y que se define en la variable **fortress.signature.home**.

```
fortress.signature.home=/home/ubuntu/fortress
```

En dicho directorio, podremos encontrar diversas subcarpetas y archivos, entre otras:

- directorio **auditory**.- Aloja ficheros de auditoria de las firmas realizadas
- directorio **custody**.- Directorio donde persistirán los documentos firmados si la custodia está activada, la activación de la custodia se indica en la siguiente variable:

```
fortress.signature.custodyStorage=true
```

- directorio **cache-tsl**.- contiene certificados y ficheros xml con las estructuras de certificados que permitirá emplear Fortress en el proceso de firma.

Firectorio almacén de certificados

Fortress emplea un almacén de certificados, donde se alojan los distintos certificados asociados a las CAs que empleará Fortress en el proceso de firma. La ruta al almacén de certificados se indica en la siguiente variable del fichero de propiedades.

```
fortress.signature.trusted_keystore.path=/home/ubuntu/fortress/trusted_cacerts.jks
```

Configuración del log del aplicativo

Crear el archivo fortress-logback.xml y alojarlo en la estructura de carpetas deseada en el servidor. Es necesario indicar la ruta donde se encuentra el fichero fortress-logback.xml en la siguiente variable del fichero de propiedades.

```
logging.config==/home/ubuntu/fortress/fortress-logback.xml
```

Acontinuación mostramos un ejemplo de definición del archivo **fortress-logback.xml**

```
<?xml version="1.0" encoding="UTF-8"?>
<configuration scan="true" scanPeriod="30 seconds">
    <property name="LOGS_FOLDER" value="${catalina.base}/logs"/>

    <appender name="log-file"
              class="ch.qos.logback.core.rolling.RollingFileAppender">
        <file>${LOGS_FOLDER}/fortress.log</file>
        <rollingPolicy class="ch.qos.logback.core.rolling.TimeBasedRollingPolicy">
            <fileNamePattern>${LOGS_FOLDER}/fortress.%d{yyyyMMdd}.log</fileNamePattern>
        </rollingPolicy>
        <encoder>
            <pattern>%d{dd/MM/yyyy HH:mm:ss.SSS} [%thread] %level %logger{36} - %m %throwable%n</pattern>
        </encoder>
    </appender>

    <logger name="com.viafirma.fortress" level="INFO"/>
    <logger name="org.springframework.web" level="INFO"/>

    <root level="ERROR">
        <appender-ref ref="log-file"/>
    </root>
</configuration>
```

```
Revisión: 05-jul-2023
```

Configuración y parametrización

Viafirma Fortress permite configurar el aplicativo a través de uno o varios ficheros de propiedades sin necesidad de modificar el código del aplicativo. Únicamente deberemos definir la ruta donde se encuentra el fichero/ficheros de propiedades en la propiedad **spring.config.location**, esto lo podemos definir de las siguientes maneras:

- En el descriptor de contexto **tomcat/conf/Catalina/< host >**

  ```
  <Context>
      <Parameter name="spring.config.location" value="/path/to/application.properties" />
  </Context>
  ```

  por ejemplo:

  ```
  <Context>
      <Parameter name="spring.config.location" value="/home/ubuntu/fortress/fortress.properties" />
  </Context>
  ```

  o indicar las rutas donde se encontrarán los distintos ficheros de propiedades separados por comas:

  ```
  <Context>
      <Environment name="spring.config.location" value="/home/ubuntu/fortress/fortress.properties,/home/ubuntu/fortress/s
ecurity.properties" type="java.lang.String"/>
  </Context>
  ```

- Como un parámetro indicado en el **setenv.sh** de la JVM empleada por el tomcat:

  ```
      -Dspring.config.location=/"/home/ubuntu/fortress/fortress.properties"
  ```

La configuración y parametrización de la aplicación contempla los siguientes apartados

Cliente

No aplica

Servidor

Configuración del datasource

Como se indica en el apartado 3.1.3.1, hay diferentes formas de establecer el DataSource que empleará el aplicativo, permitiendo indicarlo en el fichero de propiedades:

```
spring.datasource.url=
spring.datasource.username=
spring.datasource.password=
```

En caso de emplear Oracle como motor de BD, será necesario agregar el siguiente parámetro para indicar el Dialect.

```
#hibernate config
spring.jpa.database-platform=org.hibernate.dialect.Oracle10gDialect
```

Si se desea emplear el datasource de HikariCP en lugar del datasource de tomcat, será necesario agregar las sigientes variables:

```
#HikariCP datasource
spring.datasource.type=com.zaxxer.hikari.HikariDataSource
# Hikari setup connection pooling
```

```
spring.datasource.hikari.minimumIdle=5
spring.datasource.hikari.maximumPoolSize=100 (Por defecto, el pool de Hikari está limitado a 10 conexiones)
spring.datasource.hikari.idleTimeout=30000
spring.datasource.hikari.poolName=SpringBootJPAHikariCP
spring.datasource.hikari.maxLifetime=2000000
spring.datasource.hikari.connectionTimeout=30000
```

Configuración del log del aplicativo

Tal y como se ha indicado en el apartado 3.1.5.2, en el fichero de propiedades es necesario indicar la ruta donde se encuentra el fichero fortress-logback.xml

```
# LOGGING
logging.config=
```

Configuración de la licencia del aplicativo

En el apartado3.1.5.1.1, en el fichero de propiedades es necesario indicar la ruta donde se encuentra la licencia del aplicativo

```
# LICENSE
fortress.license.path=
```

Configuración de las cookies de la aplicación

Es completamente configurable en caliente, y no tira del properties. Tira de cookies.md que se encuentran en la ruta:

```
    $fortress.signature.home\cookies.md
```

Para crear sus traducciones simplemente:

```
    $fortress.signature.home\cookies_ca.md`
    $fortress.signature.home\cookies_en.md`
```

Configuración i18n externo

Si se desea sobreescribir los mensajes del aplicativo, será necesario agregar el path donde se alojarán los ficheros i18n, que sobreescribirán aquellas variables que le usuario desee.

```
#EXTERNAL I18N PATH
fortress.messagePath=
```

Configuración de parámetros globales del aplicativo

Viafirma Fortress emplea una serie de variables que permiten determinar el comportamiento del aplicativo:

```
# GLOBAL CONFIGURATION
fortress.delegations=
fortress.certificate_import=
fortress.base_url=
fortress.public_sign_up=
fortress.template_path=
fortress.certificate_password=
fortress.oauth2.token.millis_to_expire=
fortress.single_factor=
fortress.certificate_requests=
fortress.idp_user_fails_to_lock=
fortress.redirect=
fortress.redirect_url=
fortress.invitation_code=
fortress.extension_help=
fortress.extension_help_url=
```

```
fortress.download=
fortress.download_url=
fortress.subdelegations=
fortress.showAuthCache=
fortress.multifactor_auth=
fortress.multifactor_super_required=
fortress.multifactor_admin_auth=
fortress.captcha_private_key=
fortress.captcha_public_key=
fortress.terms_and_conditions_url=
fortress.delay_repudiation=
fortress.export=
fortress.timezone=
fortress.login_page_title=
fortress.login_page_subtitle=
fortress.activity.filter_max_days=
fortress.captcha_login_page=
fortress.captcha_create_user_page=
fortress.app_client_id=
fortress.disable_appQR=
fortress.locales=
fortress.default_locale=
fortress.disable_right_component_login_page=
fortress.show_activity=
fortress.statistics.active=
fortress.group.expiration.active=
fortress.show_group_expiration_date=
fortress.group_limits=
fortress.show_cookies_policy=
fortress.disabled_client_user_activity=
fortress.privacy_policy_url=
fortress.obfuscate=
fortress.inactivity_limit=
fortress.account.expiration=
fortress.password_expiration=
fortress.account.password.expiration=
fortress.password_limits=
fortress.single_token=
fortress.message_path=
fortress.show_cookies_policy=
fortress.cookies_path=
fortress.cookies_url=
fortress.template_path=
fortress.node=
fortress.theme=
fortress.title=
fortress.auth_cache=
fortress.certificate_password_fails_to_lock=
fortress.export_policies=
fortress.eidas_compliance=
fortress.max_delegations=
fortress.max_certificates=
fortress.max_groups=
fortress.max_users=
fortress.show_statistic=
fortress.show_alert=
fortress.custom_theme=
fortress.show_terms_and_conditions=
fortress.request_policy_modification=
fortress.version=
fortress.full_version=
fortress.allow_app_signature=
fortress.captcha_unlock_user_page=
fortress.captcha_activate_user_page=
fortress.captcha_reset_password_page=
fortress.delete_user_account=
fortress.configTsa=
fortress.captcha_signature_validation=
```

- **fortress.delegations**.- variable que indica si se permite la delegación de certificados entre usuarios de Viafirma Fortress.

- **fortress.certificate_import**.- variable que indica si está permitido importar certificados **P12**.
- **fortress.base_url**.- path base del aplicativo.
- **fortress.public_sign_up**.- variable que indica si está permitido que los usuarios puedan darse de alta en Viafirma Fortress.
- **fortress.templatePath**.- pathBase donde se alojarán las plantillas ftl, empleadas para las notificaciones vía email.
- **fortress.certificate_password**.- variable que indica si el certificado está protegido por una variable que únicamente conoce el usuario final.
- **fortress.oauth2.token.millis_to_expire**.- tiempo en el que expiran los tokens.
- **fortress.singleFactor**.- variable empleada, para solicitar únicamente la password del certificado, o un Factor de autenticación si el certificado no tiene password.
- **fortress.certificateRequests**.- variable que indica si se permite visualizar la sección solicitudes de certificado.
- **fortress.idpUserFailsToLock**.- variable que indica el número de intentos que podrá utilizar un usuario para autenticar contra un Factor de autenticación, por defecto dicho valor está configurado a 3.
- **fortress.redirect**.- variable que indica si la aplicación agregará una redirección tras activar, desbloquear un usuario en Viafirma Fortress, por defecto redirigirá a la pantalla de login de Viafirma Fortress, si se desea redirigir a una URL externa, será necesario indicar la Url en la variable **fortress.redirect_url**.
- **fortress.redirect_url**.- variable que almacena, la URL a donde redirigirá el sistema tras activar, desbloquear un usuario en Viafirma Fortress, esta variable depende de la variable, **fortress.redirect**.
- **fortress.invitation_code**.- variable que obliga a introducir un código de invitación asociado a un grupo, al darse de alta usuarios desde la parte pública.
- **fortress.download**.- variable que indica si se muestra la opción de descargas.
- **fortress.download_url**.- variable que contiene la URL donde redirigirá la opción de descargas, esta variable depende de la variable, **fortress.download**.
- **fortress.extension_help**.- variable que indica si se muestra la opción ayuda para habilitar la extensión del navegador de Fortress Desktop
- **fortress.extension_help_url**.- variable que contiene la URL donde se encuentra la ayuda para habilitar la extensión del navegador de Fortress Desktop, esta variable depende de la variable, **fortress.extension_help**.
- **fortress.subdelegations**.- variable que indica si se permite la subdelegación de certificados entre usuarios de Viafirma Fortress.
- **fortress.show_auth_cache**.- variable que indica si se muestra el check "caché de credenciales en el wizard de activación del usuario".
- **fortress.multifactor_auth**.- variable que indica si se obliga a emplear dos factores de autenticación a todos los usuarios de la plataforma
- **fortress.multifactor_super_required**.- variable que indica si se obliga a emplear dos factores de autenticación a los administradores globales
- **fortress.multifactor_admin_auth**.- variable que indica si se permite configurar a nivel de grupo el empleo de dos factores de autenticación para los administradores del grupo
- **fortress.captcha_private_key**.- variable que indica la private key del captcha v2 de google, si no se indica no se mostrará el captcha en las ventanas de inicio de sesión o crear usuario.
- **fortress.captcha_public_key**.- variable que indica la public key del captcha v2 de google, si no se indica no se mostrará el captcha en las ventanas de inicio de sesión o crear usuario
- **fortress.captcha_login_page**.- variable que permite activar el captcha en la pantalla de login
- **fortress.captcha_create_user_page=true**.- variable que permite activar el captcha en la pantalla de creacion de usuario
- **fortress.terms_and_conditions_url**.- variable que indica la url de redireccionamiento a términos y condiciones a la hora de crear un usuario.
- **fortress.delay_repudiation**.- variable que permite activar retardo/repudio a nivel de plataforma
- **fortress.export**.- variable que indica si se permite exportar los listados a CSV
- **fortress.timezone**.- variable que permite establecer el timezone por defecto empleado en la plataforma, por defecto "Europe/Madrid"
- **fortress.login_page_title**.-Titulo del menu derecho de la pagina de login
- **fortress.login_page_subtitle**.-Subtitulo del menu derecho de la pagina de login
- **fortress.activity.filter_max_days**. Número máximo de días que permite filtrar a un administrador en el filtro de

actividad

- **fortress.app_client_id**.- cliente que emplearán las APPS IOs y Android que interactuarán con esta instancia de Fortress.
- **fortress.disable_appQR**.- Deshabilitar el escaneo para configurar las Apps a la instancia de Fortress
- **fortress.locales**.- lenguajes soportados en el aplicativo, por defecto `es,en`
- **fortress.default_locale**.-Seleccionar el lenguaje por defecto que empleará el aplicativo, por defecto es: "es"
- **fortress.disable_right_component_login_page**.-Oculta el panel derecho, de la pantalla de login
- **fortress.show_activity**.-Permite deshabilitar la actividad
- **fortress.statistics.active**.-Permite deshabilitar las estadísticas
- **fortress.group.expiration.active**.- Permite deshabilitar mostrar la fecha de expiración asociada a los grupos
- **fortress.group_limits**.-Permite mostrar u ocultar la configuración de límites de uso asociadas al grupo
- **fortress.show_group_expiration_date**.-Permite mostrar u ocultar la sessión la configuración de la fecha de expiración del grupo
- **fortress.show_cookies_policy**.-Indica si se muestra la política de cookies
- **fortress.disabled_client_user_activity**.- Permite deshabilitar la actividad asociada a los sistemas clientes
- **fortress.privacy_policy_url**.- Url que muestra la política de privacidad
- **fortress.obfuscate**.- Valor booleano, que permite establecer si se ofuscará la información sensible de los usuarios
- **fortress.inactivity_limit**.- Booleano. Indica si se habilita el límite por inactividad
- **fortress.account.expiration**.- ntero. Número de días a partir del cual los usuarios que no interaccionan con la plataforma se bloquearán.
- **fortress.password_expiration**.- Booleano. Indica si se habilita la expiración de contraseñas
- **fortress.account.password.expiration**.- Entero. Número de días que el usuario podrá emplear la contraseña.
- **fortress.password_limits**.- Entero. Número de contraseñas utilizadas por el usuario.
- **fortress.single_token**.- Valor booleano, que indica si se limita el empleo de un token por usuario/equipo
- **fortress.message_path**.- Path indicado se alojarán los ficheros con i18n, se debe añadir un fichero de mensajes por cada lenguaje configurado en la plataforma. Por defecto "es,en", aunque estos pueden ser sobreescritos en la variable fortress.locales, del fichero fortress.properties.
- **fortress.show_cookies_policy**.- Inidica si se muestran la política de Cookies o no
- **fortress.cookies_path**.- Path donde se encuentra los archivos .md de las políticas de cookies
- **fortress.cookies_url**.- Url donde informará de la política de cookies
- **fortress.template_path**.-Path que alojará las plantillas personalizadas
- **fortress.node**.- Identificador del nodo
- **fortress.theme**.- Tema aplicado
- **fortress.title**.- Título del aplicativo
- **fortress.auth_cache**.- Valor booleano que permite indicar si se permite emplear caché de credenciales
- **fortress.certificate_password_fails_to_lock**.-Número máximo de intentos de validación de la password antes de bloquear el certificado
- **fortress.export_policies**.- Exportar Políticas
- **fortress.eidas_compliance**.- Adapta el aplicativo a la directiva eidas
- **fortress.max_delegations**.- Número maximo de delegaciones de la plataforma
- **fortress.max_certificates**.- Número maximo de certificados de la plataforma
- **fortress.max_groups**.- Número maximo de grupos de la plataforma
- **fortress.max_users**.- Número maximo de usuarios de la plataforma
- **fortress.show_statistic**.- Mostar las estadististicas
- **fortress.show_alert**.- Mostar las alertas
- **fortress.custom_theme**.- Permitir personalizar los estilos a nivel de grupo
- **fortress.show_terms_and_conditions**.- Mostrar opción de menú, términos y condiciones
- **fortress.request_policy_modification**.- Indica si se bloquearán las delegaciones, cuando se modificque aungún datos de la delegación
- **fortress.version**.- versión
- **fortress.full_version**.- versión completa
- **fortress.allow_app_signature**.- Indica si se permite firmar desde las APPs de Fortress

- **fortress.captcha_unlock_user_page**.- Variable que permite activar el captcha en la pantalla para solicitar el desbloqueo del usuario
- **fortress.captcha_activate_user_page**.- Variable que permite activar el captcha en la pantalla para solicitar la activación del usuario
- **fortress.captcha_reset_password_page**.- Variable que permite activar el captcha en la pantalla para solicitar el reseteo de la password del usuario
- **fortress.delete_user_account**.- Variable que habilita la eliminación del propio usuario desde su perfil
- **fortress.configTsa**.-Indica si se permite gestionar las TSAs en la plataforma
- **fortress.captcha_signature_validation**.-Agrega el captcha a la página de consulta de una firma y descargar documento. Debe estar configurado el captcha de google a nivel de plataforma

Deshabilitar la actividad

Permite deshabilitar el acceso a la actividad, para los usuarios de la plataforma será necesario agregar al archivo fortress.properties la propiedad fortress.showActivity=false

Eliminación los certificados expirados

Para eliminar los certificados expirados en los entornos no cualificables se ha agregado al archivo fortress.properties la propiedad:

```
fortress.removeKeys=true
```

Configuración del servidor de correo

Viafirma Fortress envía correos electrónicos y para ello, es necesario configurar el servidor de correo utilizado:

```
# EMAIL
fortress.email.host=
fortress.email.port=
fortress.email.username=
fortress.email.password=
fortress.email.from=
fortress.email.auth=
fortress.email.ssl_enable=
fortress.email.tls_enable=
fortress.email.sslProtocols=
fortress.email.connectiontimeout=
fortress.email.timeout=
```

- **fortress.email.connectiontimeout**.- variable que indica el número de milisegundos permitido para establecer la conexión con el servidor de SMTP, si no se indica dicha propiedad por defecto serán 30 segundos.
- **fortress.email.timeout**.- variable que indica el número de milisegundos permitido para enviar el correo, si no se indica dicha propiedad por defecto será 1 minuto.
- **fortress.email.sslProtocols**.- Protocolos SSL soportados, por ejemplo TLSv1.1 TLSv1.2

Configuración de la cuenta de usuario

- Ver sección Securización de las cuentas de usuario

Configuración empleada en el proceso de firma

Viafirma Fortress emplea el siguiente conjunto de variables en el proceso de firma:

```
# SIGNATURE
fortress.signature.home=
fortress.signature.trusted_keystore.path=
fortress.signature.trusted_keystore.password=
fortress.signature.custody_code=
fortress.signature.custody_storage=
fortress.signature.certificate_support_path=
```

```
fortress.signature.revocation_type_request=
fortress.signature.tslUrl=
```

Donde:

- **fortress.signature.home**.- Tal y como se ha indicado en el apartado 3.1.5.1.3, path base donde se alojarán los distintos ficheros que intervienen en el proceso de firma.
- **fortress.signature.trusted_keystore.path**.- Tal y como se ha indicado en el apartado 3.1.5.1.4, fortress emplea un almacén de certificados donde se alojan los certificados de las CAs y subCAs permitidas para la firma.
- **fortress.signature.trusted_keystore.password** .- clave del almacén de certificados
- **fortress.signature.custody_code**.- Código que identifica los archivos custodiados por fortress, dicho valor normalmente será FORTRESS.
- **fortress.signature.custody_storage**.- variable que indica si los documentos firmados persistirán en Viafirma Fortress.
- **fortress.signature.certificate_support_path**.- Fichero con formato Json, que contiene las políticas de los certificados soportadas por Viafirma Fortress.
- **fortress.signature.revocation_type_request**.-Dada la clave pública de un certificado en formato X509Certificate se facilita toda la información referente a la validez del certificado y de su cadena de confianza.

  Se permite la validación

  - **DEFAULT**: Se utilizará el sistema de validación definido en el sistema de soporte de certificados.
  - **ONLINE**: Para validar el estado de revocación de cada certificado se accede a la fuente de validación disponible preferente por OCSP.
  - **ONLINE_WITH_CACHE**: Para validar el estado de revocación de cada certificado se accede a la fuente de validación disponible preferente por OCSP, pero utilizando caché de CRL, se utilizará la crl ya descargada siempre que esta sea válida.
  - **CRL**: Se valida el estado de revocación consultando las listas de revocación de certificados CRL, siempre se descarga la CRL, no se utiliza cache. Si no puede consultar el estado de revocación mediante CRL se producirá un error.
  - **CRL_CACHE**: Se valida el estado de revocación consultando las listas de revocación de certificados CRL, utilizando caché de CRL, se utilizará la crl ya descargada siempre que esta sea válida.
  - **OCSP**: Se valida el estado de revocación del certificado haciendo uso del protocolo OCSP. Si no se puede consultar el estado de revocación se producirá un error.
  - **OFFLINE**: No se valida el estado de revocación de los certificados. En modo offline se pueden firmar documentos, pero no se pueden validar documentos ni validar certificados. Para poder verficar la cadena de confianza de un certificado es necesario informar las claves públicas de los certificados intermedios de las CA soportadas por el sistema facilitando el acceso a un KeyStore que contenga las claves públicas o facilitando una TSL (lista de confianza de prestadores de servicios de certificación formato XML).
- **fortress.signature.tslUrl**.- Url de la tsl a emplear por el aplicativo

Configuración eliminación de documentos custodiados

Viafirma Fortress emplea el siguiente conjunto de variables en el proceso de eliminación de documentos custodiados:

```
# DELETE CUSTODY DOCUMENTS
fortress.signature.custody_delete_active=
fortress.signature.default_custody_days=
fortress.signature.custody.expiration.schedule=
```

Donde:

- **fortress.signature.custody_delete_active**.- Indica si se ejecutará la eliminación de documentos custodiados.
- **fortress.signature.default_custody_days**.- Número de días por defecto que se custodiarán los documentos.
- **fortress.signature.custody.expiration.schedule** .- Indica cuando se lanzará el proceso.

```
A continuación se muestran algunos patrones de ejemplo que se pueden emplear en el parámentro fortress.signature.custody.expir
```

```
ation.schedule

* "0 0 * * * *" = the top of every hour of every day.
* "*/10 * * * * *" = every ten seconds.
* "0 0 8-10 * * *" = 8, 9 and 10 o'clock of every day.
* "0 0/30 8-10 * * *" = 8:00, 8:30, 9:00, 9:30 and 10 o'clock every day.
* "0 0 9-17 * * MON-FRI" = on the hour nine-to-five weekdays
* "0 0 0 25 12 ?" = every Christmas Day at midnight
```

Configuración notificación de próxima expiración de certificados

Viafirma Fortress emplea el siguiente conjunto de variables en el proceso de notificación de próxima expiración de certificados:

```
# CERTIFICATE EXPIRATION
fortress.notification.active=
fortress.notification.certificate_expiration_days=
fortress.notification.schedule=
```

Donde:

- **fortress.notification.active**.- Indica si se notificará o no la próxima expiración de los certificados.
- **fortress.notification.certificate_expiration_days**.- Número de días que el procedimiento comprobará si debe notificar la próxima expiración de los certificados. Permite indicar varios valores númericos separados por ",".
- **fortress.notification.schedule** .- Indica cuando se lanzará el proceso.

```
A continuación se muestran algunos patrones de ejemplo que se pueden emplear en el parámentro fortress.notification.schedule

* "0 0 * * * *" = the top of every hour of every day.
* "*/10 * * * * *" = every ten seconds.
* "0 0 8-10 * * *" = 8, 9 and 10 o'clock of every day.
* "0 0/30 8-10 * * *" = 8:00, 8:30, 9:00, 9:30 and 10 o'clock every day.
* "0 0 9-17 * * MON-FRI" = on the hour nine-to-five weekdays
* "0 0 0 25 12 ?" = every Christmas Day at midnight
```

Configuración calculo estadísticas por grupo

Viafirma Fortress emplea el siguiente conjunto de variables en el proceso de obtención de datos estadísticos por grupo

```
# STATISTIC
fortress.statistics.active=
fortress.statistics.schedule=
```

Donde:

- **fortress.statistics.active**.- Indica si se ejecutará el cálculo de estadísticas.
- **fortress.statistics.schedule** .- Indica cuando se lanzará el proceso.

```
A continuación se muestran algunos patrones de ejemplo que se pueden emplear en el parámentro fortress.statistics.schedule

* "0 0 * * * *" = the top of every hour of every day.
* "*/10 * * * * *" = every ten seconds.
* "0 0 8-10 * * *" = 8, 9 and 10 o'clock of every day.
* "0 0/30 8-10 * * *" = 8:00, 8:30, 9:00, 9:30 and 10 o'clock every day.
* "0 0 9-17 * * MON-FRI" = on the hour nine-to-five weekdays
* "0 0 0 25 12 ?" = every Christmas Day at midnight
```

Configuración control expiración de grupos

Viafirma Fortress emplea el siguiente conjunto de variables en el proceso de validación de grupos expirados

```
# STATISTIC
```

```
fortress.group.expiration.active=
fortress.group.expiration.schedule=
```

Donde:

- **fortress.group.expiration.active**.- Indica si se ejecutará la comprobación de grupos expirados
- **fortress.group.expiration.schedule** .- Indica cuando se lanzará el proceso.

```
A continuación se muestran algunos patrones de ejemplo que se pueden emplear en el parámentro fortress.group.expiration.schedu
le

* "0 0 * * * *" = the top of every hour of every day.
* "*/10 * * * * *" = every ten seconds.
* "0 0 8-10 * * *" = 8, 9 and 10 o'clock of every day.
* "0 0/30 8-10 * * *" = 8:00, 8:30, 9:00, 9:30 and 10 o'clock every day.
* "0 0 9-17 * * MON-FRI" = on the hour nine-to-five weekdays
* "0 0 0 25 12 ?" = every Christmas Day at midnight
```

Configuración del HSM que empleará Viafirma Fortress

Como se indicó en el apartado 2.4, fortress permite configurar el almacén centralizado de certificados donde se alojarán los certificados de los usuarios. Dependiendo del proveedor empleado serán necesarios unos valores y otros, para poder realizar la conexión y operar con este. Por ejemplo si empleamos un keystore para almacénar los certificados, las variables a configurar serían:

```
# KEYSTORE
fortress.keystore.provider=JKS
fortress.keystore.jks.path=/pathToKeystore/keystore.jks
fortress.keystore.jks.password=*******
fortress.keystore.masterkey.alias=fortress-master-key
fortress.keystore.masterkey.autogenerated=true
```

Donde:

- **fortress.keystore.provider**.- tipo de proveedor, JKS
- **fortress.keystore.jks.path**.- ruta donde se aloja el amacen de certificados de tipo jks
- **fortress.keystore.jks.password**.- password del amacen de certificados
- **fortress.keystore.masterkey.alias**.- alias de la master key
- **fortress.keystore.masterkey.autogenerated**.- genera automáticamente la master key

Configuración del HSM Safenet Luna

En caso de emplear el HSM Luna de Safenet, las variables a indicar serían:

```
# KEYSTORE
fortress.keystore.provider=SafeNet
fortress.keystore.safenet.partition.label=
fortress.keystore.safenet.partition.password=*******
fortress.keystore.safenet.partition.slotNumber=
fortress.keystore.safenet.v7=true
fortress.keystore.mechanism=CKM_AES_KWP
fortress.keystore.migrate_mechanism=
```

Donde:

- **fortress.keystore.provider**.- tipo de proveedor, SafeNet
- **fortress.keystore.safenet.partition.label**.- etiqueta asociada a la partición
- **fortress.keystore.safenet.partition.password**.- password de la partición
- **fortress.keystore.safenet.partition.slotNumber**.- número de slot
- **fortress.keystore.safenet.v7**.- Parámetro que debe incorporarse en la configuración para activar el soporte a clientes lunas >7.3.

- **fortress.keystore.mechanism**.- Parámetro necesario para versiones Firmware 7.7.0 y superior, activando con ello el mecanismo CKM_AES_KWP en lugar del mecanismo usado en versiones anteriores CKM_AES_CBC_PAD.
- **fortress.keystore.migrate_mechanism**.- Valor booleano que indica si se migrarán las claves al nuevo mecanismo de WRAPEO.

Configuración del HSM ncipher

En caso de emplear el HSM ncipher, las variables a indicar serían:

```
# KEYSTORE
fortress.keystore.provider=SunPKCS11-ncipherProvider
fortress.keystore.pkcs11.password=
fortress.keystore.pkcs11.configName=
```

Donde:

- **fortress.keystore.provider**.- tipo de proveedor, SunPKCS11-ncipherProvider
- **fortress.keystore.pkcs11.password**.- password del HSM
- **fortress.keystore.pkcs11.configName**.- ruta a donde se encuentra el fichero de configuración del cliente

Configuración del HSM Realsec

En caso de emplear el HSM Realsec Cryptosec Lan, las variables a indicar serían:

```
# KEYSTORE
fortress.keystore.provider=SunPKCS11-realSecProvider
fortress.keystore.realsec.label=
fortress.keystore.realsec.password=*******
fortress.keystore.realsec.configName=
```

Donde:

- **fortress.keystore.provider**.- tipo de proveedor, SunPKCS11-realSecProvider
- **fortress.keystore.realsec.label**.- etiqueta asociada a la partición
- **fortress.keystore.realsec.password**.- password de la partición
- **fortress.keystore.realsec.configName**.- ruta a donde se encuentra el fichero de configuración del cliente de realsec.

Configuración Soporte CA Support

En caso de desear emplear CASupport para validar los certificados, será necesario agregar la siguiente información:

```
# CA SUPPORT
fortress.signature.caSupport.active=
fortress.signature.caSupport.url=
fortress.signature.caSupport.paths=
```

Donde:

- **fortress.signature.caSupport.active**.- Variable que permite activar o desactivar la validación empleando CA Support, por defecto está desactivado
- **fortress.signature.caSupport.url**.- URL del servicio CASupport
- **fortress.signature.caSupport.paths**.- paths a los que se desea suscribir

Configuración del Tema

Para establecer un tema propio para la instalación de Viafirma Fortress en las dependencias de un cliente, será necesario indicar la siguiente variable

```
# THEME
```

```
fortress.theme=
```

Donde:

- **fortress.theme**.-identificador del tema que describe la apariencia del aplicativo para un cliente en cuestión, solo para instalaciones propias en cliente.

Configuración de las RAs

Viafirma Fortress permite configurar 0 a n RAs para solicitar certificados asociados a la instancia de fortress desplegada. Si no se configura ninguna RA, Viafirma Fortress únicamente podrá emplear los certificados importados por los usuarios.

Viafirma Fortress ha implementado conectores con las siguientes RAs:

Ra CSR.- permite generar el CSR para emplearlo en aquellas RAs que permitan generar los certificados empleando dicho CSR.

```
# RA CSR
fortress.ra.csr.active=true
```

- **fortress.ra.csr.active=true**.- variable que indica si la generación del CSR esta activa

Ra Avansi.- permite interactuar con la RA Avansi

```
# RA AVANSI
fortress.ra.avansi.active=
fortress.ra.avansi.api.url=
fortress.ra.avansi.api.consumer_key=
fortress.ra.avansi.api.consumer_secret=
fortress.ra.avansi.schedule=
fortress.ra.avansi.certificate_password=
fortress.ra.avansi.change_password=
```

Donde:

- **fortress.ra.avansi.active**.- variable que indica si la RAde avansi esta activa
- **fortress.ra.avansi.api.url**.- URL de la RA
- **fortress.ra.avansi.api.consumer_key**.- consumer key
- **fortress.ra.avansi.api.consumer_secret**.-consumer secret
- **fortress.ra.avansi.schedule**.- Periodcidad en la que se comprobará la creación de nuevos certificados
- **fortress.ra.avansi.certificate_password**.- variable que indica si el certificado creado desde la RAde optic está protegido por una PIN.
- **fortress.ra.avansi.change_password**.- variable que indica si es necesario cambiar la password

Ra Thomas Signe.- permite interactuar con la RA de Thomas Signe

```
# RA TSIGNE
fortress.ra.tsigne.active=
fortress.ra.tsigne.api.url=
fortress.ra.tsigne.api.consumer_key=
fortress.ra.tsigne.api.consumer_secret=
fortress.ra.tsigne.schedule=
fortress.ra.tsigne.certificate_password=
fortress.ra.tsigne.change_password=
```

Donde:

- **fortress.ra.tsigne.active**.- variable que indica si la RAde tsigne esta activa
- **fortress.ra.tsigne.api.url**.- URL de la RA

- **fortress.ra.tsigne.api.consumer_key**.- consumer key
- **fortress.ra.tsigne.api.consumer_secret**.-consumer secret
- **fortress.ra.tsigne.schedule**.- Periodcidad en la que se comprobará la creación de nuevos certificados
- **fortress.ra.tsigne.certificate_password**.- variable que indica si el certificado creado desde la RAde optic está protegido por una PIN.
- **fortress.ra.tsigne.change_password**.- variable que indica si es necesario cambiar la password

Ra Optic.- permite interactuar con la RA de la Optic

```
fortress.ra.optic.active=
fortress.ra.optic.api.url=
fortress.ra.optic.api.consumer_key=
fortress.ra.optic.api.consumer_secret=
fortress.ra.optic.schedule=
fortress.ra.optic.certificate_password=
fortress.ra.optic.change_password=
```

Donde:

- **fortress.ra.optic.active**.- variable que indica si la RAde optic esta activa
- **fortress.ra.optic.api.url**.- URL de la RA
- **fortress.ra.optic.api.consumer_key**.- consumer key
- **fortress.ra.optic.api.consumer_secret**.-consumer secret
- **fortress.ra.optic.schedule**.- Periodcidad en la que se comprobará la creación de nuevos certificados
- **fortress.ra.optic.certificate_password**.- variable que indica si el certificado creado desde la RAde optic está protegido por una PIN.
- **fortress.ra.optic.change_password**.- variable que indica si es necesario cambiar la password

Ra Ogtic.- permite interactuar con la RA de la Ogtic

```
fortress.ra.ogtic.active=
fortress.ra.ogtic.api.url=
fortress.ra.ogtic.api.consumer_key=
fortress.ra.ogtic.api.consumer_secret=
fortress.ra.ogtic.schedule=
fortress.ra.ogtic.certificate_password=
fortress.ra.ogtic.change_password=
```

Donde:

- **fortress.ra.ogtic.active**.- variable que indica si la RAde ogtic esta activa
- **fortress.ra.ogtic.api.url**.- URL de la RA
- **fortress.ra.ogtic.api.consumer_key**.- consumer key
- **fortress.ra.ogtic.api.consumer_secret**.-consumer secret
- **fortress.ra.ogtic.schedule**.- Periodcidad en la que se comprobará la creación de nuevos certificados
- **fortress.ra.ogtic.certificate_password**.- variable que indica si el certificado creado desde la RAde optic está protegido por una PIN.
- **fortress.ra.ogtic.change_password**.- variable que indica si es necesario cambiar la password

Ra firmaprofesional.- permite interactuar con la RA de firma profesional

```
fortress.ra.firmaprofesional.active=
fortress.ra.firmaprofesional.public_key_path=
fortress.ra.firmaprofesional.certificate_password=
fortress.ra.firmaprofesional.change_password=
```

Donde:

- **fortress.ra.firmaprofesional.active**.- variable que indica si la RAde firmaprofesional esta activa

- **fortress.ra.firmaprofesional.public_key_path**.- ruta donde se aloja la clave pública proporcionada por firmaprofesional
- **fortress.ra.firmaprofesional.certificate_password**.- variable que indica si el certificado creado desde la RAde firmaprofesional está protegido por una PIN.
- **fortress.ra.firmaprofesional.change_password**.- variable que indica si es necesario cambiar la password

Ra camerfirma.- permite interactuar con la RA de camerfirma

```
fortress.ra.camerfirma.active=
fortress.ra.camerfirma.certificate_password=
fortress.ra.camerfirma.change_password=
```

Donde:

- **fortress.ra.camerfirma.active**.- variable que indica si la RAde camerfirma esta activa
- **fortress.ra.camerfirma.certificate_password**.- variable que indica si el certificado creado desde la RAde camerfirma está protegido por una PIN.
- **fortress.ra.camerfirma.change_password**.- variable que indica si es necesario cambiar la password

Ra accv.- permite interactuar con la RA de Accv

```
fortress.ra.accv.active=
fortress.ra.accv.certificate_password=
fortress.ra.accv.change_password=
```

Donde:

- **fortress.ra.accv.active**.- variable que indica si la RAde accv esta activa
- **fortress.ra.accv.certificate_password**.- variable que indica si el certificado creado desde la RAde accv está protegido por una PIN.
- **fortress.ra.accv.change_password**.- variable que indica si es necesario cambiar la password

Configuración del Ldap

Viafirma Fortress permite interactuar contra el LDAP configurado para autenticar a un usuario, para poder emplear la conexión con LDAP será necesario indicar las siguientes variables:

```
# LDAP
fortress.ldap.active=
fortress.ldap.host=
fortress.ldap.domain=
fortress.ldap.path=CN=
fortress.ldap.field.username=
fortress.ldap.field.nif=
fortress.ldap.field.entirename=
fortress.ldap.field.mail=
fortress.ldap.field.phone=
fortress.ldap.username=
fortress.ldap.password=
```

Donde:

- **fortress.ldap.active**.- Indica si está activo o no.
- **fortress.ldap.host**.-URL de conexión con el LDAP, por ejemplo: `LDAP://localhost:389`
- **fortress.ldap.domain**.- Nombre del dominio
- **fortress.ldap.path**.-Path base donde se encuentran los usuarios, por ejemplo `CN=Users,DC=prueba,DC=local`
- **fortress.ldap.field.username**.- campo que indica el código de usuario almacénado en el ldap
- **fortress.ldap.field.entirename**.- campo que contiene el nombre del usuario en el LDAP, normalmente `cn`
- **fortress.ldap.field.nif**.- campo que indica el nif del usuario

- **fortress.ldap.field.mail**.- campo que contiene el email del usuario en el LDAP, normalmente `mail`
- **fortress.ldap.field.phone**.- campo que contiene el teléfono del usuario en el LDAP, normalmente `telephoneNumber`
- **fortress.ldap.username**.- Usuario que conecta con el LDAP
- **fortress.ldap.password**.- clave del usuario que conecta con el LDAP

Autentitación con Ldap

Viafirma Fortress permite autenticar a un usuario contra el LDAP configurado:

```
#LDAP Authentication
fortress.ldap.auth_search_filter=
fortress.ldap.user_mappings=
fortress.ldap.auth_group_search_filter=
fortress.ldap.user_search_base=
fortress.ldap.group_search_base=
fortress.ldap.role_prefix=
```

Donde:

- **fortress.ldap.auth_search_filter**.- filtro de búsqueda de los usuarios, por ejemplo: `(&(objectClass=User)(sAMAccountName={0})(|(userAccountControl=512)(userAccountControl=66048))`
- **fortress.ldap.user_mappings**.-Mapeo de los campos del usuario con los devueltos desde el LDAP, por ejemplo: `code=sAMAccountName\nemail=mail\nfirstName=givenName\nlastName=sn\nnif=department\ndateAdded=whenCreated\ndateUpdated=whenChanged\nstatus=userAccountControl\nmobile=telephoneNumber`
- **fortress.ldap.auth_group_search_filter**.- filtro de búsqueda de los grupos
- **fortress.ldap.user_search_base**.- cadena base donde se alojan los usuarios
- **fortress.ldap.group_search_bas**.- cadena base donde se alojan los grupos
- **fortress.ldap.role_prefix**.- Prefijo del rol

Configuración de factores de autenticación

Viafirma Fortress permite habilitar distintos factores de autenticación que permitan verificar a un usuario en las aplicaciones que integren con Viafirma Fortress, a continuación se indican los distintos factores de autenticación disponibles, así como los parámetros que permiten habilitar los mismos.

Configuración factor de autenticación OTP

El token OTP únicamente emplea el siguiente parámetro de configuración **fortress.idp.otp.active** encargado de habilitar o deshabilitar el factor de autenticación

```
# IDP OTP
fortress.idp.otp.active=true
```

Configuración factor de autenticación SMS

Viafirma Fortress ha realizado varios conectores que permiten enviar códigos de verificación a los usuarios mediante envíos de SMS.

Actualmente se han implementado conectores con los siguientes proveedores:

- **SMS Publi**
- **SMS Arena**

Acontinuación indicaremos los distintos parámetros de configuración que debemos indicar para habilitar el factor de autenticación de SMS:

```
# IDP SMS
fortress.idp.sms.active=
fortress.idp.sms.provider=
```

Donde:

- **fortress.idp.sms.active**.- Indica si está o no habilitado el factor de autenticación para ser empleado en la verificación de los usuarios
- **fortress.idp.sms.provider**.- Debemos indicar el proveedor a emplear, en este caso los valores son: **SMSPubli** o **SMSArena** .

Configuración proveedor SMSpubli

Para configurar este proveedor, deberemos indicar el apiKey proporcionada por SMSPubli al registrarnos en su plataforma. Dicha apiKey debemos almacenarla en la variable **fortress.idp.sms.smspubli.apiKey**.

```
fortress.idp.sms.smspubli.apiKey=
```

Configuración proveedor SMS Arena

Para configurar este proveedor, deberemos indicar las siguientes variables

```
fortress.idp.sms.smsarena.url="https://api.smsarena.es/http/";
fortress.idp.sms.smsarena.authKey=
```

Donde:

- **fortress.idp.sms.smsarena.url**.- URL al api proporcionado por el proveedor
- **fortress.idp.sms.smsarena.authKey**.- código de autenticación en el proveedor

Configuración factor de autenticación Email

La configuración del factor de autenticación de Email, empleará la configuración del servidor de correo indicado en el Apartado 3.2.2.4, por lo que únicamente deberemos habilitarlo o deshabilitarlo mediante la variable **fortress.idp.email.active**.

```
# IDP EMAIL
fortress.idp.email.active=
```

Configuración del factor de autenticación PIN

Dicho Factor de autenticación como su propio nombre indica permite verificar a un usuario mediante un código PIN, gestionado por fortress.

La configuración del factor de autenticación de PIN, únicamente permite habilitarlo o deshabilitarlo mediante la variable **fortress.idp.pinactive**.

```
# IDP PIN
fortress.idp.pin.active=
```

Configuración del factor de autenticación LDAP

Factor de autenticación que permite autenticar a un usuario contra el LDAP configurado, para poder emplear este Factor de autenticación será necesario indicar las siguientes variables:

```
# IDP LDAP
fortress.idp.ldap.active=true
```

Donde:

- **fortress.idp.ldap.active**.- Indica si está activo o no.

Configuración del factor de autenticación Password

Factor de autenticación que permite autenticar a un usuario empleando su password de Viafirma Fortress, para poder emplear este Factor de autenticación será necesario indicar las siguientes variables:

```
# IDP PASSWORD
fortress.idp.password.active=true
```

Donde:

- **fortress.idp.password.active**.- Indica si está activo o no.

Configuración para la importación de usuario mediante archivos CSV

Acontinuación se indican las variables, necesarias para habilitar la importación de usuarios empleando archivos CSV

```
#CSV Import
fortress.import.user.path.separator=;
```

Donde:

- **fortress.import.user.path.separator**.- Caracter empleado como separador en el archivo.

Configuración para la importación de usuario mediante LDAP

Acontinuación se indican las variables, necesarias para habilitar la importación de usuarios contra un LDAP

```
#LDAP Import
fortress.ldap.schedule=
fortress.ldap.filter=
fortress.ldap.group_search_filter=
fortress.ldap.user_imports=
fortress.import.user.path.ldap_user_process=
```

Donde:

- **fortress.ldap.schedule**.- Variable que indica la periodicidad en la que se ejecutará la importación
- **fortress.ldap.filter**.- filtro de búsqueda de los usuarios a importar
- **fortress.ldap.group_search_filter**.- filtro de búsqueda de los grupos
- **fortress.ldap.user_imports**.- Variable que indica si la importación de usuarios está activa
- **fortress.import.user.path.ldap_user_process**.- Path que almacenará los archivos generados con el resultado de la importación.

Configuración integración CAS

El uso de CAS en la plataforma requerirá que se den de alta en el fichero de propiedades los siguientes campos:

```
# CAS
fortress.cas.active=
fortress.cas.server_url=
fortress.cas.createUser=true
fortress.cas.logoutSSO=true
fortress.cas.userGroup=
fortress.cas.enrollerGroup=
fortress.cas.adminGroup=
fortress.cas.auditorGroup=
fortress.cas.attribute.username=
fortress.cas.attribute.displayName=
fortress.cas.attribute.mail=
fortress.cas.attribute.phone=mobile
fortress.cas.attribute.memberOf=isMemberOf
```

Donde:

- **fortress.cas.active**.- Valor booleano, que permite establecer si se activa el empleo de CAS a nivel de plataforma.
- **fortress.cas.server_url**.- URL de CAS contra el que se quiere autenticar a los usuarios.
- **fortress.cas.logoutSSO**.- Valor booleano, que permite establecer si se realizará o no el logout del SSO cuando se realice logout en la aplicación. Por defecto, true.
- **fortress.cas.createUser**.- Valor booleano, que indica si se creará en la plataforma el usuario, en caso de no estar dado de alta en la misma y autenticar correctamente contra el CAS, valor por defecto true, ojo si dicho valor es false, la plataforma mostrará un mensaje indicando que el usuario no se encuentra en la paltaforma.

Mapeo de valores asociados a los usuarios en caso de crear los mismos, es decir, fortress.cas.createUser=true.

- **fortress.cas.attribute.username**.- Campo de los valores devueltos por el CAS que se empleará como código de usuario. Por defecto, employeeNumber, si no hay ningún usuario en el sistema que coincida con el username indicado buscaremos por el principal.
- **fortress.cas.attribute.displayName**.- Campo de los valores devueltos por el CAS que se empleará para obtener el nombre completo del usuario. Por defecto, displayName.
- **fortress.cas.attribute.mail**.- Campo de los valores devueltos por el CAS que se empleará para obtener el email del usuario. Por defecto, mail.
- **fortress.cas.attribute.phone**.- Campo de los valores devueltos por el CAS que se empleará para obtener el teléfono móvil del usuario (El teléfono debe contener el prefijo). Por defecto, mobile
- **fortress.cas.attribute.memberOf**.- Campo de los valores devueltos por el CAS que se empleará para obtener los grupos del usuario. Por defecto, isMemberOf.

Mapeo de roles

Para poder asignar roles a los usuarios creados en la plataforma, es necesario asociar cada rol empleado con un grupo asociado al usuario, devueltos en la variable fortress.cas.attribute.memberOf.

- **fortress.cas.adminGroup**.- Todos los usuarios pertenecientes al grupo asignado, se le asignará el rol ADMIN
- **fortress.cas.auditorGroup**.- Todos los usuarios pertenecientes al grupo asignado, se le asignará el rol AUDITOR
- **fortress.cas.enrollerGroup**.- Todos los usuarios pertenecientes al grupo asignado, se le asignará el rol ENROLLER
- **fortress.cas.userGroup**.- Todos los usuarios pertenecientes al grupo asignado o que no pertenezcan a ninguno de los anteriores, se le asignará el rol USER.

Configuración integración RA HORUS

El uso de RAHORUS en la plataforma requerirá que se den de alta en el fichero de propiedades los siguientes campos:

```
# RA HORUS

fortress.ra.horus.active=
fortress.ra.horus.api.url=
fortress.ra.horus.api.debug=
```

Donde:

- **fortress.ra.horus.active**.- Variable que indica si la RAde horus esta activa
- **fortress.ra.horus.api.url**.- Url de la RAde HORUS
- **fortress.ra.horus.api.debug**.- Variable que indica swi se activa o no el modo depuración

Configuración Activación Factor de autenticación PUSH

Para activar el factor de autenticación PUSH en la plataforma, requerirá que se den de alta en el fichero de propiedades los siguientes campos:

```
# IDP PUSH
fortress.idp.push.active=
fortress.idp.push.url=
```

```
fortress.idp.push.username=
fortress.idp.push.password=
fortress.idp.push.environment=
```

Donde:

- **fortress.idp.push.active**.- Valor booleano, que permite establecer si se activa el fator de autenticación PUSH, por defecto se encuentra deshabilitada valor false
- **fortress.idp.push.url**.- URL plataforma de envio de PUSHES
- **fortress.idp.push.username**.- Código de usuario para autenticar en la plataforma de PUSHES
- **fortress.idp.push.password**.- Password para autenticar en la plataforma de PUSHES
- **fortress.idp.push.environment**.- Identiciador del entorno empleado para autenticar en la plataforma de PUSHES

Configuración integración Audit-Trail

Para activar Audit-Trail, requerirá que se den de alta en el fichero de propiedades los siguientes campos:

```
# AUDITTRAIL
fortress.auditTrailActive=
fortress.auditTrailUrl=
fortress.auditTrailUser=
fortress.auditTrailPass=
```

Donde:

- **fortress.auditTrailActive**.- Indica si esta activa la cominicación con Audit-Trail
- **fortress.auditTrailUrl**.- Url de auditTrail
- **fortress.auditTrailUser**.- Usuario de auditTrail
- **fortress.auditTrailPass**.- Password de auditTrail

Configuración de TSA

Para activar una TSA, requerirá que se den de alta en el fichero de propiedades los siguientes campos:

```
#TSA
fortress.tsa.primary.type=
fortress.tsa.primary.url=
fortress.tsa.primary.user=
fortress.tsa.primary.password=
fortress.tsa.primary.policyId=
fortress.tsa.primary.certificateCode=
fortress.tsa.primary.timestampAlgorithm=
fortress.tsa.primary.extensionOid=
fortress.tsa.primary.extensionValue=
# backup tsa TSA
fortress.tsa.secondary.type=
fortress.tsa.secondary.url=
fortress.tsa.secondary.user=
fortress.tsa.secondary.password=
fortress.tsa.secondary.policyId=
fortress.tsa.secondary.certificateCode=
fortress.tsa.secondary.timestampAlgorithm=
fortress.tsa.secondary.extensionOid=
fortress.tsa.secondary.extensionValue=
```

Donde:

- **fortress.tsa.primary.type**.- Tipo de TSA: URL, USER, CERTIFICATE, CERTIFICATE_TLS
- **fortress.tsa.primary.url**.- Url de conexión con la tsa
- **fortress.tsa.primary.user**.- Usuario de conexión con la tsa
- **fortress.tsa.primary.password**.- Password de conexión con la tsa
- **fortress.tsa.primary.policyId**.- Identificador de la política

- **fortress.tsa.primary.certificateCode**.- Código de certificado, se debe administrar desde la administración de fortress
- **fortress.tsa.primary.timestampAlgorithm**.- Algortimo de sellado de tiempo
- **fortress.tsa.primary.extensionOid**.- OID de la extensión
- **fortress.tsa.primary.extensionValue**.-Valor de la extensión
- **fortress.tsa.secondary.type**.- Tipo de TSA: URL, USER, CERTIFICA
- **fortress.tsa.secondary.url**.- Url de conexión con la tsa
- **fortress.tsa.secondary.user**.- Usuario de conexión con la tsa
- **fortress.tsa.secondary.password**.- Password de conexión con la tsa
- **fortress.tsa.secondary.policyId**.- Identificador de la política
- **fortress.tsa.secondary.certificateCode**.- Código de certificado, se debe administrar desde la administración de fortress
- **fortress.tsa.secondary.timestampAlgorithm**.- Algortimo de sellado de tiempo
- **fortress.tsa.secondary.extensionValue**.- OID de la extensión
- **fortress.auditTrailPass**.- Password de auditTrail

Configuración de Seguridad - Properties

```
Revisión: 13-ene-2022
```

Securización de las cuentas de usuario

Viafirma Fortress permite configurar ciertos parámetros relacionados con la cuenta de usuario, tales como:

```
# ACCOUNT CONFIGURATION
fortress.account.password.regexp=^([a-zA-Z0-9_-]{4,15})$
fortress.account.password.messageError=global.password.format.error.alphanumericHyphenDash
fortress.account.password.minLength=4
fortress.account.password.maxLength=15
fortress.account.failsToLock=3
fortress.account.recoverMinutesToExpire=60
fortress.account.unlockMinutesToExpire=60
```

- **fortress.account.password.regexp**.- variable que indica la expresión regular del formato que debe cumplir la contraseña del usuario en Viafirma Fortress. Por defecto, admite caracteres alfanuméricos, guion alto y bajo.
- **fortress.account.password.messageError**.- variable que indica el mensaje de error que debe aparecer en caso de que la contraseña del usuario no cumpla el formato deseado.
- **fortress.account.password.minLength**.- variable que indica la longitud mínima de la contraseña del usuario en Viafirma Fortress
- **fortress.account.password.maxLength**.- variable que indica la longitud máxima de la contraseña del usuario en Viafirma Fortress.
- **fortress.account.failsToLock**.- variable que indica el número de intentos de acceso a Viafirma Fortress erróneos antes de bloquer al usuario.
- **fortress.account.recoverMinutesToExpire**.- variable que indica los minutos que tarda en expirar el token de recuperación de usuario.
- **fortress.account.unlockMinutesToExpire**.- variable que indica los minutos que tarda en expirar el token de activación de usuarios empleará las siguientes propiedades.

Procedimiento de Desinstalación

Desinstalación

Para el proceso de desinstalación de la aplicación, se recomienda en primer lugar parar el servidor de aplicaciones en el que estará corriendo la aplicación, para posteriormente eliminar de la carpeta /webapps:

El directorio /viafirma_fortress El fichero: viafirma_fortress_v.xx.yy.zz.WAR

Procedimiento para la actualización de versiones

Actualización de versiones

El proceso de actualización de versiones básicamente consiste en:

1) Desinstalar la aplicación

2) Desplegar el nuevo WAR

3) Adecuar los parámetros de configuración

4) Ejecutar los scripts de actualización de BD

Procedimiento de Backup y recuperación

La política de Backup dependerá de cada cliente, y las políticas de seguridad definidas internamente.

Se recomienda realizar backups de los siguientes elementos:

- Base de Datos del aplicativo
- HSM donde se alojarán los certificados
- Ficheros de configuración del aplicativo
- Almacén de certificados **trusted_cacerts.jks**
- Si se permite la custodia de documentos firmados, backup incremental de los documentos firmados.