
Artículo

[Ricardo Paiva](#) · 9 jul, 2021 Lectura de 2 min

Cómo ejecutar el Portal de Administración (servidor web privado) a través de TLS/SSL/HTTPS

Hola todos,

Quiero compartir un sencillo y rápido método que puede usarse para habilitar ssl con un certificado auto-firmado en una instancia de desarrollo local de IRIS/HealthShare. Esto permite probar funciones específicas de https, como OAuth.

1. Instalar OpenSSL

Windows: https://slproweb.com/download/Win64OpenSSL_Light-1_1_1g.exe

Debian Linux: `$ sudo apt-get -y install openssl`

RHEL: `$ sudo yum install openssl`

2. Crear un par de certificados auto-firmados. En tu terminal (powershell, bash, zsh, etc.)

```
$ openssl req -x509 -nodes -days 365 -newkey rsa:2048 -keyout apache-selfsigned.key -out apache-selfsigned.crt
```

Nota: este comando anterior creará un certificado que durará un año.

3. Editar el servidor web privado para utilizar el nuevo par de certificados auto-firmados

En el directorio de instalación de tu instancia, edita tu configuración de pws <install-dir>/httpd/conf/httpd-local.conf. Añade la siguiente sección antes de las directivas "Incluir ...".

```
# Port to listen for secure traffic On. The default is 443
LoadModule ssl_module "modules/mod_ssl.so"
Listen 10443

# Listen Virtual Host Block to define the keys we should use for that port
# If you define a different port in the Listen directive, change that here as well
<VirtualHost *:10443>

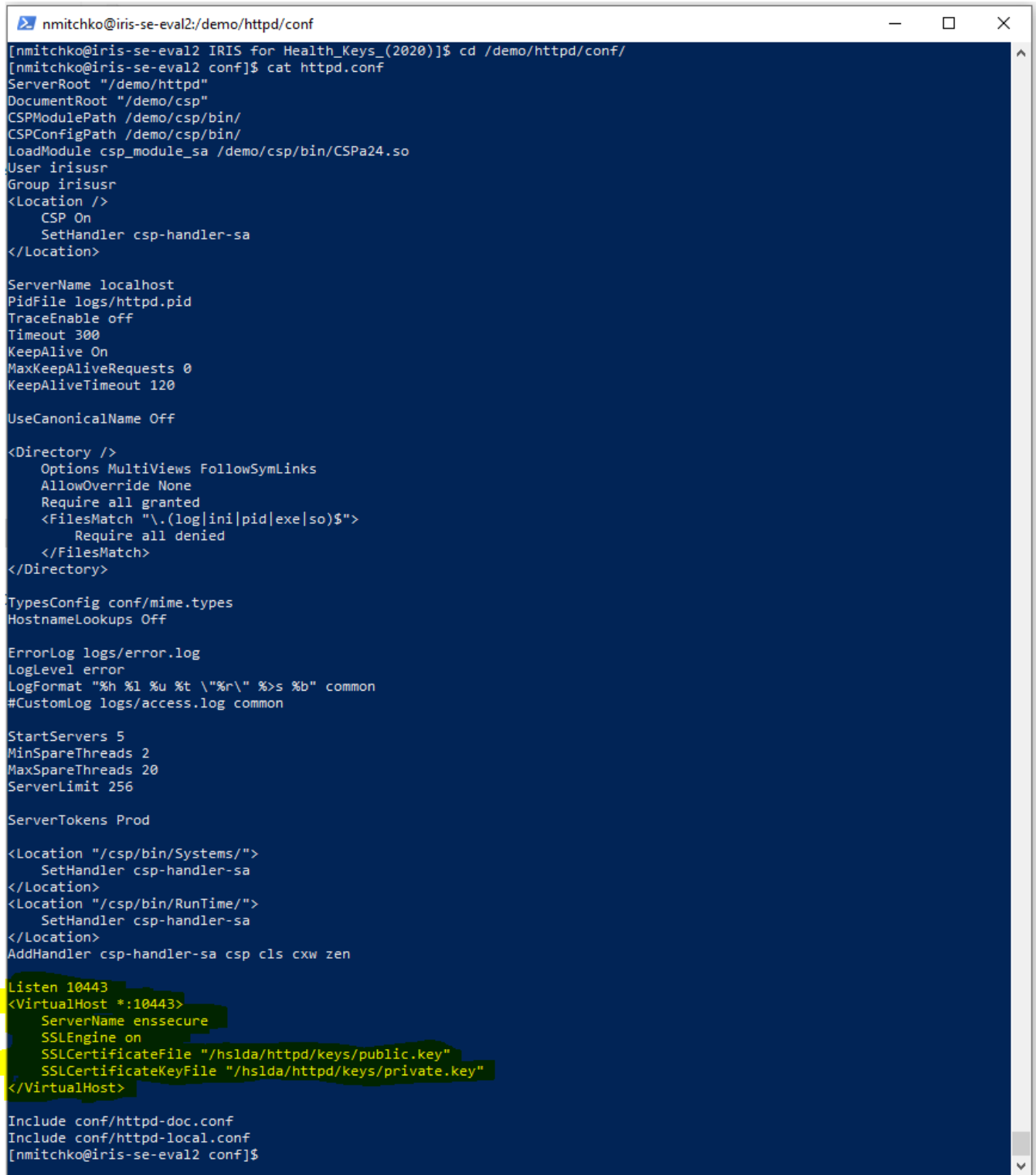
    # We need a servername, it has not effect but is required by apache
    ServerName mysecureinstance

    # Turn on SSL for this Virtual Host
    SSLEngine on

    #key files, replace these paths with the path you generated the keys from in step
    2.
```

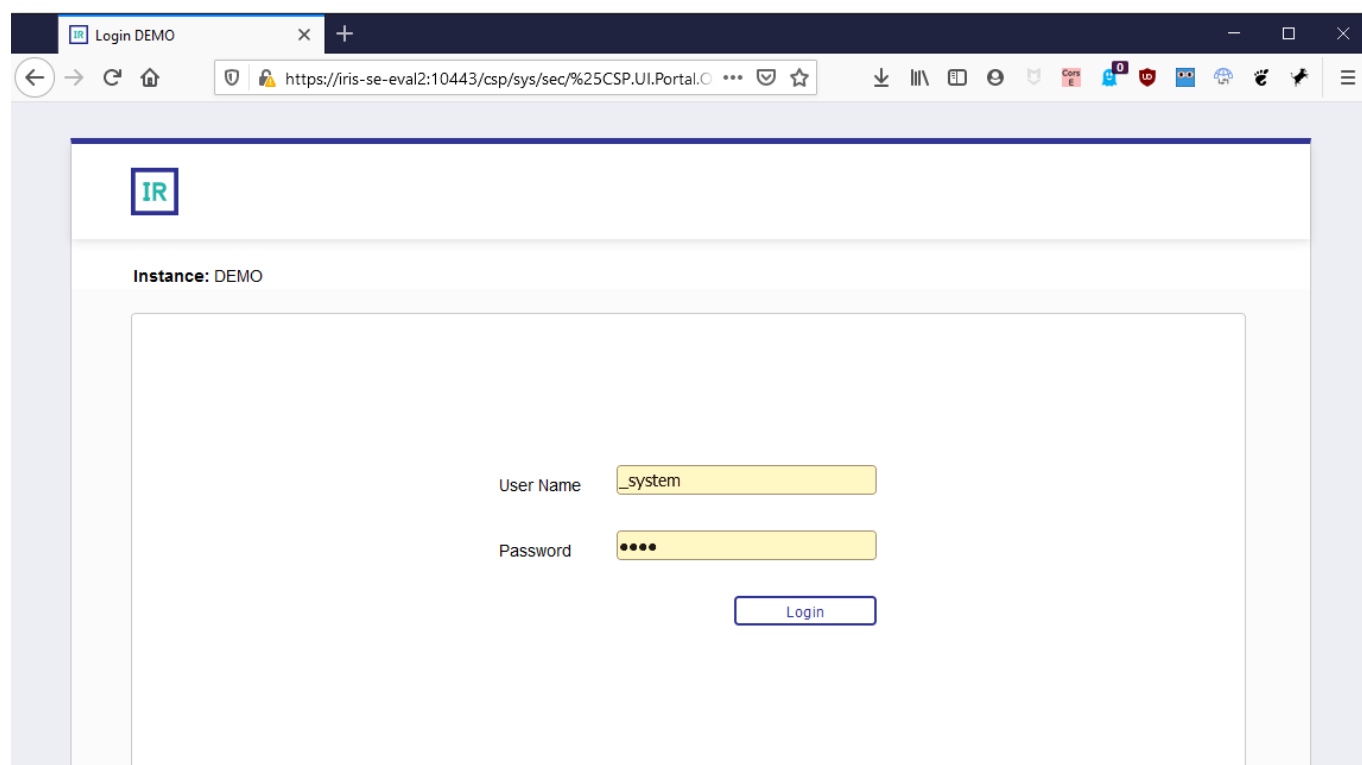
```
SSLCertificateFile "/path/to/apache-selfsigned.crt"  
  
SSLCertificateKeyFile "/path/to/apache-selfsigned.key"  
</VirtualHost>
```

Este es un ejemplo de mi archivo de configuración:



```
[nmitchko@iris-se-eval2 IRIS for Health_Keys_(2020)]$ cd /demo/httpd/conf/  
[nmitchko@iris-se-eval2 conf]$ cat httpd.conf  
ServerRoot "/demo/httpd"  
DocumentRoot "/demo/csp"  
CSPModulePath /demo/csp/bin/  
CSPConfigPath /demo/csp/bin/  
LoadModule csp_module_sa /demo/csp/bin/CSPa24.so  
User irisusr  
Group irisusr  
<Location />  
    CSP On  
    SetHandler csp-handler-sa  
</Location>  
  
ServerName localhost  
PidFile logs/httpd.pid  
TraceEnable off  
Timeout 300  
KeepAlive On  
MaxKeepAliveRequests 0  
KeepAliveTimeout 120  
  
UseCanonicalName Off  
  
<Directory />  
    Options MultiViews FollowSymLinks  
    AllowOverride None  
    Require all granted  
    <FilesMatch "\.(log|ini|pid|exe|so)$">  
        Require all denied  
    </FilesMatch>  
</Directory>  
  
TypesConfig conf/mime.types  
HostnameLookups Off  
  
ErrorLog logs/error.log  
LogLevel error  
LogFormat "%h %l %u %t \"%r\" %>s %b" common  
#CustomLog logs/access.log common  
  
StartServers 5  
MinSpareThreads 2  
MaxSpareThreads 20  
ServerLimit 256  
  
ServerTokens Prod  
  
<Location "/csp/bin/Systems/">  
    SetHandler csp-handler-sa  
</Location>  
<Location "/csp/bin/RunTime/">  
    SetHandler csp-handler-sa  
</Location>  
AddHandler csp-handler-sa csp cls cxw zen  
  
Listen 10443  
<VirtualHost *:10443>  
    ServerName enssecure  
    SSLEngine on  
    SSLCertificateFile "/hsl/da/httpd/keys/public.key"  
    SSLCertificateKeyFile "/hsl/da/httpd/keys/private.key"  
</VirtualHost>  
  
Include conf/httpd-doc.conf  
Include conf/httpd-local.conf  
[nmitchko@iris-se-eval2 conf]$
```

En ejecución:



Nota: InterSystems no soporta este tipo de configuración HTTPS y, si necesitas un producto de producción, debes seguir las instrucciones para instalar apache2/IIS/nginx de manera completa.

[#Mejores prácticas](#) [#SSL](#) [#HealthShare](#) [#InterSystems IRIS](#) [#InterSystems IRIS for Health](#)

URL de
fuente: <https://es.community.intersystems.com/post/c%C3%B3mo-ejecutar-el-portal-de-administraci%C3%B3n-servidor-web-privado-trav%C3%A9s-de-tlshttps>