OMPI
ORGANIZACIÓN MUNDIAL
DE LA **PROPIEDAD**
INTELECTUAL

1 de octubre de 2021

Servicios web del Sistema de La Haya

ÍNDICE

INTRODUCCIÓN

ÁMBITO

Este documento es una introducción a los servicios web del Sistema de La Haya (HWS), una interfaz entre máquinas para el Sistema de La Haya.

DEFINICIÓN

Los HWS es un API REST segura, fiable y de alta disponibilidad basada en el protocolo HTTPS diseñada para el intercambio de datos con el Sistema de La Haya, puede utilizarse tanto para enviar como para recibir datos.

Los HWS permiten:

- Enviar decisiones o solicitudes indirectas
- Comprobar el estado de las importaciones
- Consultar el estado de tramitación
- Obtener los boletines del Sistema de La Haya
- Obtener copias confidenciales (solo para las Oficinas de PI)

Los HWS son el canal de intercambio de datos preferente del Sistema de La Haya. Se recomienda encarecidamente que las OPI los utilicen desde el principio. Se anima a las Oficinas que ya intercambian datos con el Sistema de La Haya mediante la interfaz EDI, en papel o por otros canales a adoptar el canal proporcionado a trav'es de los HWS.

AUTENTICACIÓN Y SEGURIDAD

SEGURIDAD DE LA API

La API de los HWS está diseñada para la comunicación entre máquinas con contenido confidencial.

La autenticación se basa en la firma de claves asimétricas que forman parte del Perfil de seguridad de API financieras 1.0. Este perfil se puede aplicar a las API de cualquier sector que requiera un nivel de seguridad más elevado que el del protocolo estándar OAuth u OpenID Connect. Esto significa que tiene un mayor perfil de seguridad que OAuth, adecuado para proteger las API con un alto riesgo inherente.

GENERACIÓN DEL PAR DE CLAVES Y REGISTRO DEL IDENTIFICADOR DE CLIENTE

El siguiente diagrama muestra el proceso completo para registrar el identificador de cliente y la clave pública de la API en la OMPI, así como la dirección IP pública de la aplicación cliente.

Pasos correspondientes a la Oficina

1. Generar un par de claves pública y privada (Apéndice A: Cliente para la generación de pares de claves OpenSSL).

2. Generar el certificado x509 mediante la clave pública.

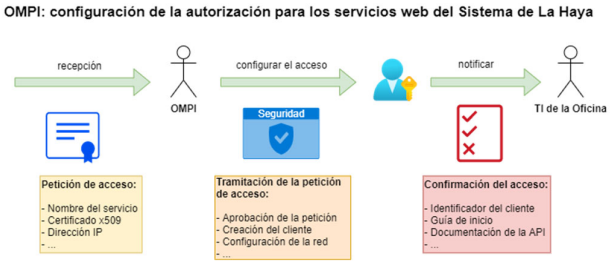
3. Pedir acceso a los HWS enviando un correo electrónico a hague.it@wipo.int con:

    a) El formulario de la OMPI cumplimentado (Apéndice B: Formulario de petición de acceso a la API del Departamento de Tecnologías de la Información y las Comunicaciones de la OMPI)

b) El certificado x509

Pasos correspondientes a la OMPI:

1. Una vez recibido lo anterior, generar el identificador del cliente.
2. Asignar/vincular la clave pública al identificador del cliente.
3. Autorizar la dirección IP.
4. Configurar los HWS para que autoricen las peticiones al identificador del cliente.
5. Confirmar el identificador del cliente a la Oficina de PI.

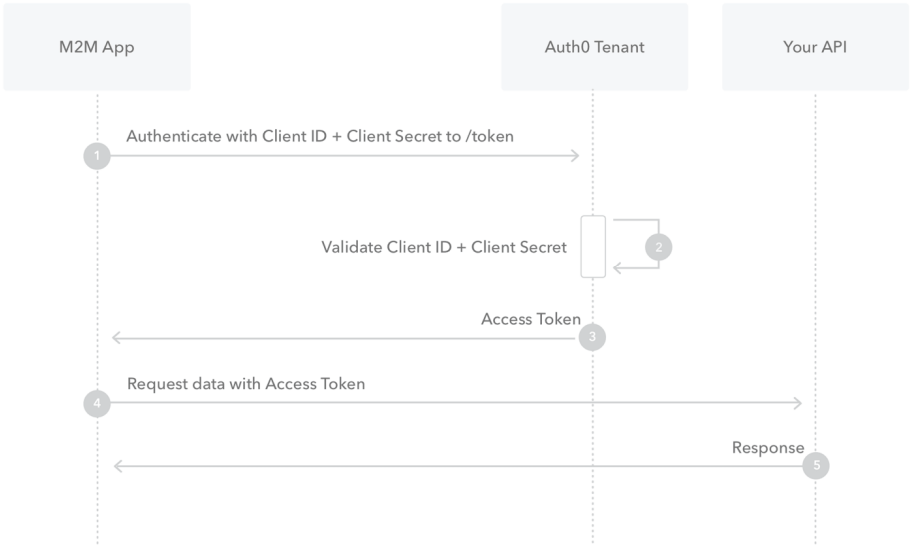


AUTENTICACIÓN EN LOS HWS

Una vez que el identificador del cliente, la clave pública y la dirección IP pública están registrados en la OMPI, y los HWS están configurados, la Oficina de PI está lista para utilizar la API.

El siguiente diagrama muestra la interacción:

1. Petición HTTPS al titular de Auth0 con el identificador del cliente y el testigo JWT firmado por la clave privada. Nota: La petición **debe** provenir de la dirección IP autorizada.
2. Se valida la petición HTTPS y se genera el testigo de acceso JWT.
3. Si el proceso es satisfactorio, se devuelve el testigo de acceso JWT con un vencimiento de una hora.
4. Las llamadas posteriores a los puntos finales HTTPS dentro del plazo de vencimiento pueden realizarse utilizando el mismo testigo de acceso JWT.



Traducción de la imagen anterior:

M2M App	Aplicación entre máquinas
Auth0 Tenant	Titular de Auth0
Your API	Su API
Authenticate with Client ID + Client Secret to /token	Autenticar con el identificador y el testigo de acceso secreto del cliente
Validate Client ID + Client Secret	Validar el identificador y el testigo secreto del cliente
Access Token	Testigo de acceso
Request data with Access Token	Petición de datos con el testigo de acceso
Response	Respuesta

DESCRIPCIÓN DE LA API

La API de los HWS utiliza los siguientes puntos finales REST:

1. Envío de solicitudes y decisiones indirectas al Sistema de La Haya (POST /request).

2. Comprobación del estado de importación de una solicitud o decisión indirecta enviada (GET /request/import).

3. Consulta del estado de una petición de servicio determinada (GET /request/{serviceRequestId}).

4. Obtención de un boletín (GET /publication/bulletin/{weekId}).

5. Obtención de una copia confidencial (GET /publication/copy/confidential/{weekId}).

Los detalles sobre la API de los HWS (parámetros, respuestas, etc.) figuran en el Apéndice D (API pública del Sistema de La Haya).

Todo el contenido se basa en el formato XML utilizado en la Norma ST.96 de la OMPI. Los detalles sobre la Norma ST.96, versión 4.0, y las definiciones XSD figuran en https://www.wipo.int/standards/es/st96/v4-0/. Actualmente se están normalizando pequeñas extensiones específicamente necesarias para los servicios web.

Nota: Existe un punto final denominado pingMe que permite comprobar la conexión entre el cliente y los HWS. No tiene ninguna función, pero está disponible a efectos de prueba y validación técnica.

Envío de solicitudes y decisiones indirectas al Sistema de La Haya (POST /request).

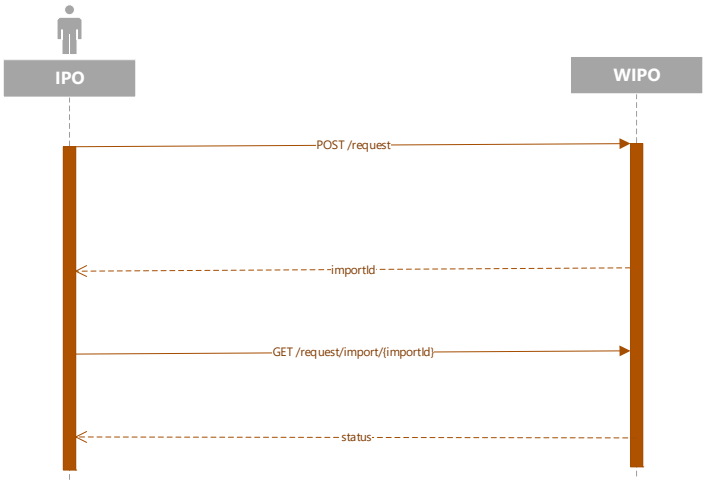
El envío de solicitudes y decisiones al Sistema de La Haya se efectúa mediante una petición POST, en la que el contenido es el paquete de importación (véase más abajo).

Si el proceso es satisfactorio, se devuelve un identificador de importación por paquete, lo que significa que el paquete de importación se ha recibido y será procesado por la Oficina Internacional.

Este identificador de importación del paquete puede utilizarse posteriormente para obtener el número de petición de servicio (GET request/import), el cual permite obtener el estado de la petición (GET request).

El contenido de una petición de solicitud y decisión es un único fichero ZIP que consiste en documentos e imágenes referenciados en el formato XML de la Norma ST.96.

- Esos archivos deben ubicarse en la ruta relativa indicada en el XML.
- Un fichero ZIP (el contenido de una petición) debe contener solo una solicitud o decisión.
- El contenido no puede incluir más de un fichero XML.
- Pueden consultarse ejemplos en ftp://ftpird.wipo.int/ST96_V_4_0_test/import-packages-4.0.zip.



Consulta del estado de una petición de servicio (GET /request/{serviceRequestId})

Una vez importado el paquete en el Sistema de La Haya, la transacción se denomina petición de servicio y se le atribuye un número de petición, el cual puede obtenerse mediante el punto final GET request/import (véase más arriba).

Una vez que el número de petición de servicio está disponible, se puede obtener el estado de la petición mediante el punto final correspondiente.

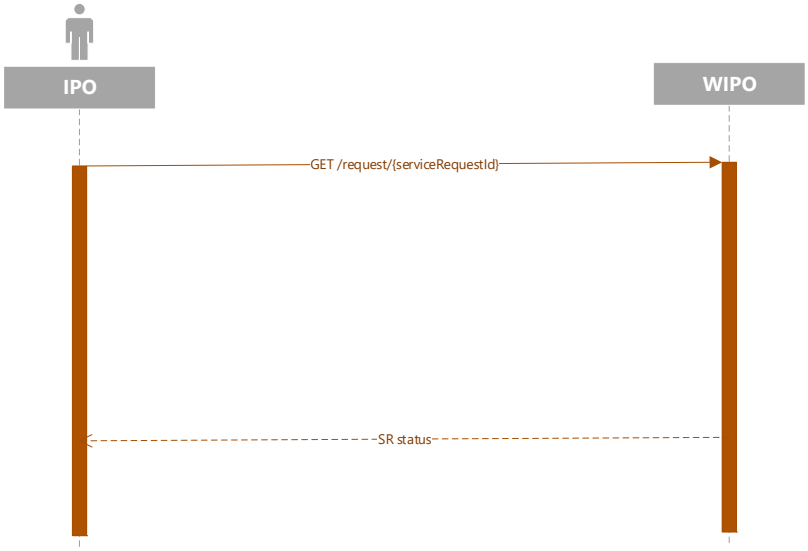
Los tipos de estado son:

- No definido
- En trámite
- Pendiente de regularización

- Registrado
- Abandonado
- Anulado

Como respuesta a la consulta del estado de una petición de servicio, los servicios web del Sistema de La Haya devuelven el contenido según la Norma ST.96, que incluye:

- Identificador de la petición
- Estado de tramitación
- Número de petición de servicio
- Número de registro internacional
- Fecha prevista de publicación (si procede)



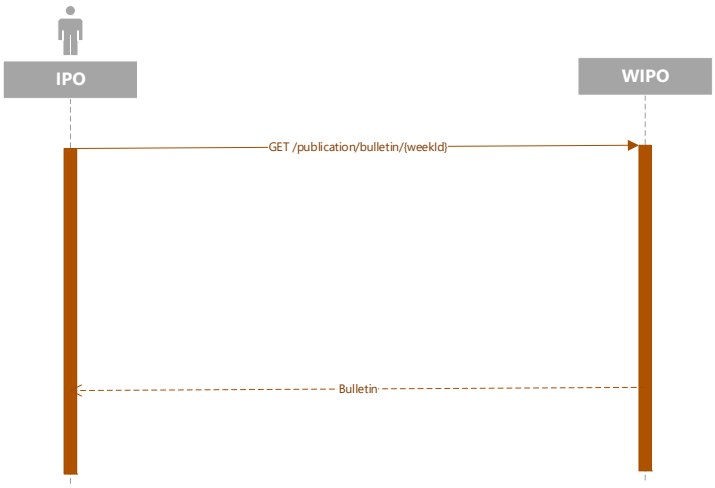
Obtención de un boletín (GET /publication/bulletin/{weekId})

La Oficina Internacional publica boletines semanales, normalmente el viernes por la noche (hora central europea). Los boletines se pueden solicitar en cualquier momento a partir del momento en que se generan. El formato del parámetro weekId es aaaass.

El contenido de la respuesta es un fichero ZIP que incluye:

- Los datos bibliográficos del boletín como fichero ST.96
- Carpetas de imágenes correspondientes a las inscripciones incluidas o correcciones de imágenes

Pueden encontrarse ejemplos de contenido de los boletines (las copias confidenciales tienen la misma arquitectura) en ftp://ftpird.wipo.int /ST96_V_4_0.



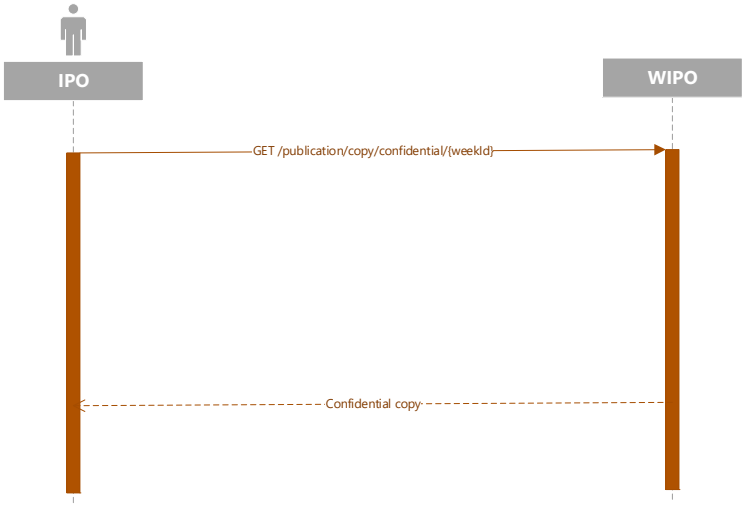
Obtención de una copia confidencial (GET /publication/copy/confidential/{weekId})

Las copias confidenciales tienen la misma arquitectura que los boletines.

La Oficina Internacional emite copias confidenciales semanalmente, normalmente el viernes por la noche (hora central europea). Las copias confidenciales se pueden solicitar en cualquier momento a partir del momento en que se generan. El formato del parámetro weekId es aaaass.

El contenido de la respuesta es un fichero ZIP que incluye:

- Los datos bibliográficos de la copia confidencial como fichero ST.96
- Carpetas de imágenes correspondientes a las inscripciones incluidas o correcciones de imágenes



APÉNDICE A: CLIENTE PARA LA GENERACIÓN DE PARES DE CLAVES OPENSSL

Generar un par de claves asimétricas privada y pública y un certificado x509 para el registro entre máquinas de la OMPI.

Generar objetos para la inscripción OIDC de la OMPI

```
#!/bin/bash

# Set the environment
PRIVATE_KEY_ES256=hague4offices_private.pem
PUBLIC_KEY_ES256=hague4offices_public.pem
CLIENT_NAME=DAS

# Generates the ES256 keys
openssl ecparam -genkey -name prime256v1 -noout -out "${PRIVATE_KEY_ES256}"

# Extracts the public key
openssl ec -in "${PRIVATE_KEY_ES256}" -pubout -out "${PUBLIC_KEY_ES256}"

# Generates an x509 certificate
CERT_KEY_ES256=es256_cert.pem
OPENSSL_CONF=./openssl.cnf
CERT_CN="${CLIENT_NAME} private_key_jwt authentication"

# Build the certificate config file
printf '[ req ]\n' > "${OPENSSL_CONF}"
printf 'prompt = no\n' >> "${OPENSSL_CONF}"
printf 'distinguished_name = req_distinguished_name\n' >> "${OPENSSL_CONF}"
printf '[ req_distinguished_name ]\n' >> "${OPENSSL_CONF}"
printf 'CN = %s\n' "${CERT_CN}" >> "${OPENSSL_CONF}"

# Creates the x509 certificate
openssl req -x509 -new -config "${OPENSSL_CONF}" -key "${PRIVATE_KEY_ES256}" -out
"${CERT_KEY_ES256}"
```

1. Enviar **es256_cert.pem** a la OMPI para configurar el acceso a los servicios web del Sistema de La Haya (**hague4offices_private.pem** debe mantenerse siempre en secreto y nunca debe compartirse).

2. Esperar a que la OMPI comunique el **identificador del cliente** y el **ámbito** después de la configuración.

3. Probar la comunicación mediante la aplicación cliente facilitada por el Sistema de La Haya (enlace por confirmar).

APÉNDICE B: FORMULARIO DE PETICIÓN DE ACCESO A LA API DEL DEPARTAMENTO DE TECNOLOGÍAS DE LA INFORMACIÓN Y LAS COMUNICACIONES DE LA OMPI

Los formularios que figuran a continuación son borradores de la OMPI. Su versión final está pendiente y se confirmará más adelante (01/09/2021).

Rellene este formulario para proporcionar información general sobre el contexto.

Información general	
¿Tipo de petición?	Petición de creación ☐ Petición de actualización ☐
Descripción de la información actualizada[1]	Información de contacto ☐ IP del cliente/rango de IP ☐ Certificado ☐ Ámbitos ☐
Entorno[2]	Producción
Nombre de la aplicación API	
Descripción de la aplicación	
Direcciones URL de la API	
Titular comercial de la aplicación	
Correo electrónico del titular comercial de la aplicación[3]	
Nombre del contacto técnico de la aplicación	
Correo electrónico del contacto técnico de la aplicación[4]	
¿Quién tendrá acceso a la aplicación?	Personal interno ☐ Personal externo ☐
¿Cómo se protege la API?	Mediante un testigo de acceso obtenido a través del flujo de credenciales de cliente de Oauth2

1 Proporcione esta información solo en caso de una petición de actualización.
2 Seleccione un entorno.
3 Se utilizará para notificaciones cuando se planifique el despliegue del componente de un proveedor de Oauth2 en producción que pueda repercutir en la aplicación.
4 Se utilizará para notificaciones cuando se planifique el despliegue del componente de un proveedor de Oauth2 en producción que pueda repercutir en la aplicación.

Rellene este formulario para proporcionar información sobre el cliente:

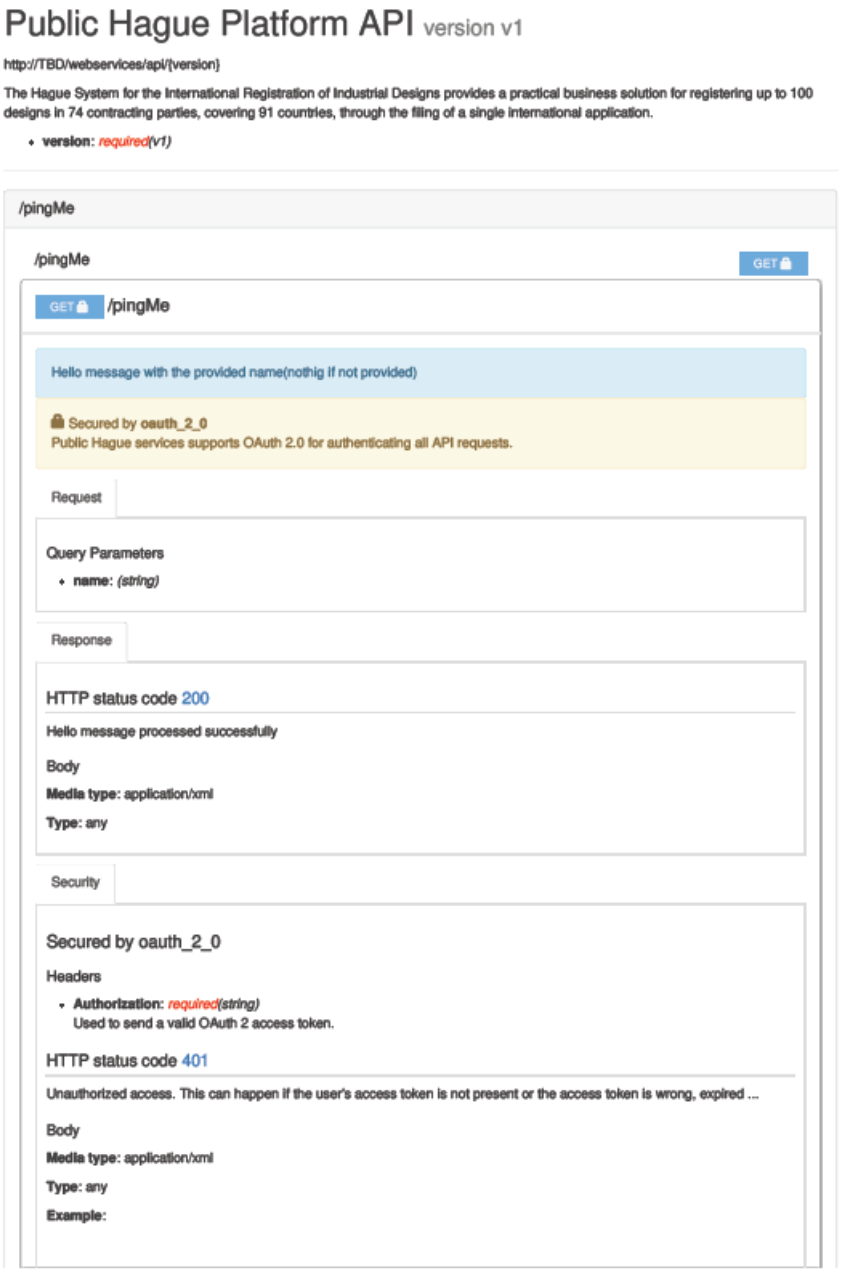
Protección de la API mediante OAuth2	
Identificador del cliente	Facilitado por la OMPI
Tipo de cliente	Confidencial
Ámbitos compatibles (opcional)	Perfil predeterminado
Certificado (X509V3 - ES256)	
IP del cliente/rango de IP	
Método de autentificación del cliente	private_key_jwt (el cliente envía sus credenciales como testigo JWT)

APÉNDICE C: FRAGMENTO PARA OBTENER UN TESTIGO DE ACCESO DE DEV OPENAM DE LA OMPI

El siguiente fragmento bash es un ejemplo de petición de autenticación de la OMPI utilizando la clave privada de la OPI:

```
#!/bin/bash
PRIVATE_KEY_ES256=es256_private.pem
CLIENT_ID=das-api-auth
SCOPE="das-api/das-access"
ISSUER="https://logindev.wipo.int/am/oauth2"

# https://logindev.wipo.int/am/oauth2/.well-known/openid-configuration
OIDC_CONFIG_JSON=$(curl -k "${ISSUER}/.well-known/openid-configuration")

# Generic way to obtain the token endpoint
TOKEN_ENDPOINT=$(printf '%s' ${OIDC_CONFIG_JSON} | jq -r ".token_endpoint")
UTC_TIME=$(date -u +%s)
EXP_TIME=$(expr "$UTC_TIME" + 10)
JWT_ID=Un1qu3i0

JSON='{'
JSON=${JSON}$(printf '"iss":"%s"' ${CLIENT_ID})
JSON=${JSON}$(printf ,"sub":"%s"' ${CLIENT_ID})
JSON=${JSON}$(printf ,"aud":"%s"' ${TOKEN_ENDPOINT})
JSON=${JSON}$(printf ,"exp":%s' ${EXP_TIME})
JSON=${JSON}'}'

JSON_HEADER_B64=$(printf '{"alg":"ES256","typ":"JWT"}' | jq -cj | base64 -w0 | tr -d
'\n=' | tr '+/' '-_')

JSON_PAYLOAD_B64=$(printf $JSON | jq -cj | base64 -w0 | tr -d '\n=' | tr '+/' '-_')

JSON_SIGNATURE_ASN1_B64=$(printf '%s.%s' $JSON_HEADER_B64 $JSON_PAYLOAD_B64 | openssl
dgst -sha256 -sign"${PRIVATE_KEY_ES256}" | openssl asn1parse -inform DER | base64 -w0)
JSON_SIGNATURE_HEX=$(printf $JSON_SIGNATURE_ASN1_B64 | base64 -d | sed -n '/INTEGER/p'
| sed 's/.*INTEGER\s*://g' | sed -z 's/[^0-9A-F]//g')
JSON_SIGNATURE_B64=$(printf $JSON_SIGNATURE_HEX | xxd -p -r | base64 -w0 | tr -d '\n='
| tr '+/' '-_')

JWT_ASSERTION=$(printf '%s.%s.%s' $JSON_HEADER_B64 $JSON_PAYLOAD_B64
$JSON_SIGNATURE_B64)

# echo $JWT_ASSERTION
# Access token private_key_jwt
# --insecure is only needed when testing within WIPO premises (because of the
proxy...)
curl \
 --header "Content-Type: application/x-www-form-urlencoded" \
 --data-urlencode "grant_type=client_credentials" \
 --data-urlencode scope="${SCOPE}" \
 --data-urlencode "client_assertion_type=urn:ietf:params:oauth:client-assertion-
type:jwt-bearer" \
 --data-urlencode "client_assertion=${JWT_ASSERTION}" \
 --url "${TOKEN_ENDPOINT}"
```

APÉNDICE D: API PÚBLICA DEL SISTEMA DE LA HAYA



Nota: Se añadirá información en versiones posteriores de este documento.