

**ADMINISTRACIÓN GENERAL DE TECNOLOGÍA
DE LA INFORMACIÓN**

**ADMINISTRACIÓN CENTRAL DE DESARROLLO
Y MANTENIMIENTO DE APLICACIONES**

ADMINISTRACIÓN DE DESARROLLO PARA ADUANAS

**SISTEMA DE AUTOMATIZACIÓN ADUANERA INTEGRAL
(S. A. A. I.)**

**PROTOCOLOS DE COMUNICACIÓN ENTRE
EQUIPOS DE TERCEROS (PREVALIDADORES) Y
EQUIPOS DEL SAT**

**PROTOCOLOS DE COMUNICACIÓN ENTRE EQUIPOS DE TERCEROS
(PREVALIDADORES) Y EQUIPOS DEL SAT.**

C O N T E N I D O

0. CONSIDERACIONES	2
1. TRANSMISOR DE ARCHIVOS A VALIDAR.....	3
2. SERVICIO DE RECEPCIÓN DE ARCHIVOS DE RESPUESTA DE VALIDACIÓN.....	5
ANEXO A: Códigos de error.....	6

**ADMINISTRACIÓN CENTRAL DE DESARROLLO
Y MANTENIMIENTO DE APLICACIONES.
ADMINISTRACIÓN DE DESARROLLO PARA ADUANAS**

**PROTOCOLOS DE COMUNICACIÓN ENTRE EQUIPOS DE
TERCEROS (PREVALIDADORES) Y EQUIPOS DEL SAT**

0. CONSIDERACIONES.

Las aplicaciones cliente-servidor residentes en los equipos de terceros para comunicación con aplicaciones del SAAI fueron desarrolladas originalmente para plataforma UNIX en equipos HP 9000, utilizando lenguaje "C". Posteriormente se buscó adaptar la programación a otras plataformas y actualmente se tienen versiones para:

- ✦ Equipos HP, arquitectura RISC 1 y RISC 2, sistema operativo HP-UX 9.x o superior.
- ✦ Equipos SCO-UNIX,
- ✦ Equipos PC con Windows 95 o Windows NT.

Dadas las anteriores limitaciones, se ha optado por dar a conocer las **secuencias de intercambio de información** entre las aplicaciones residentes en los equipos de terceros y los equipos de aplicaciones SAAI de las aduanas, para que las partes interesadas puedan desarrollar sus aplicaciones **utilizando la plataforma y lenguajes que más les convengan**.

A la fecha, las aplicaciones mínimas que deben tener los terceros para un servicio adecuado por parte de equipos con aplicaciones SAAI son las que se enlistan a continuación, todas ellas del tipo cliente-servidor cuya transferencia de información se lleva a cabo vía sockets:

- ✦ Programa tipo "demonio" para la transmisión de archivos a ser validados en el equipo del SAT.
- ✦ Programa "servidor" para la recepción de archivos de respuesta de validación, transmitidos desde el equipo del SAT.

En esta primera versión del documento, se describen los protocolos de comunicación para que un servidor con conexión a un equipo de aplicación SAAI pueda:

- ✦ transmitir archivos de pedimentos para que sean validados.
- ✦ recibir archivos de respuesta de validación, transmitidos desde el equipo de aplicación SAAI.

Se omiten los protocolos de las otras aplicaciones; en una segunda versión se incluirán dichos protocolos.

El software que se desarrolle para la transmisión y recepción de archivos debe poder utilizar los *sockets* para comunicación con un equipo remoto, con protocolo TCP/IP.

Dada la posible presencia de más de un equipo de terceros con conexión a un equipo de aplicaciones SAAI, es muy importante tener en cuenta el concepto del **carácter de identificación de un equipo de terceros, conocido en general como prevalidador**. Este carácter es una letra, dígito o carácter especial proporcionado por las autoridades del SAT, es único para cada prevalidador y se genera al terminar las tareas de configuración en el equipo del SAT para que el nuevo prevalidador pueda tener comunicación con el mismo. Actualmente, la configuración la lleva a cabo personal de ISOSA.

1. TRANSMISOR DE ARCHIVOS A VALIDAR.

En este documento se describe sólo la parte del intercambio de información, a partir de que se logra la conexión vía *sockets* con el equipo del SAT. No es propósito del documento indicar cómo se crea un socket, el usuario debe tener los conocimientos necesarios para lograr esto.

Los nombres de las funciones utilizadas en esta descripción se toman del **lenguaje "C"** bajo un ambiente **UNIX**. Los nombres de las funciones equivalentes en otro tipo de plataforma pueden cambiar, por lo que se recomienda fuertemente tener a la mano los manuales correspondientes para utilizarlas de manera correcta. En UNIX, la conexión se logra con la función *connect()*.

Datos para la conexión:

- ✧ Protocolo: TCP/IP
- ✧ Nombre del servicio: trans_recep
- ✧ Número de puerto: 50205
- ✧ Familia de sockets: AF_INET
- ✧ Dirección del host remoto: La correspondiente al equipo SAT

Secuencia de intercambio de información.

Transmisión de datos previos.

1. Se transmite una cadena de trece caracteres: los primeros doce son el nombre del archivo incluida la extensión; el carácter número 13 es el identificador del equipo prevalidador del que se transmite el archivo. Para este propósito, se utilizan las funciones *write()* o *send()*.
2. A continuación, se transmiten cuatro bytes (un entero largo) que especifican el tamaño del archivo a transmitir, en bytes. Estos deben ser transmitidos del byte más significativo al byte menos significativo. Se debe tener cuidado con el orden de los bytes utilizados para el almacenamiento de datos enteros en una plataforma en particular. Consulte en algún manual lo relativo a la función *htonl()* (host to net long), utilizada para cambiar el orden de los bytes que componen un dato entero largo para que sean conformes para transmitirse a la red.

Recepción y análisis del primer acuse.

3. Se recibe 1 byte (primer acuse de recibo) como resultado de la validación de los datos previos. Cuando el valor de este byte es diferente de 0, se tiene un error que debe interpretarse como se muestra en el anexo A para llevar a cabo las acciones correctivas correspondientes; a continuación se debe **cerrar el socket** y terminar la comunicación. Si el valor del byte recibido es cero, se procede a la transferencia de información.

Transmisión de la información (contenido del archivo a validar).

4. Se abre en modo lectura el archivo cuya información se transmitirá al equipo del SAT. Véase la función *open()*. Se utiliza la función *read()* para leer bloques de 1024 caracteres del archivo. Si se utiliza la función *fopen()* para abrir el archivo, la información se podría transmitir línea a línea, en vez de bloques de bytes.
5. Se lee un bloque de 1024 bytes desde el archivo (o una línea de información). Si el valor devuelto por la función *read()* es mayor que cero, se debe transmitir ese número de bytes al equipo del SAT; el valor devuelto por la función *write()* (número de bytes transmitidos) debe ser igual al número de bytes leídos. Si se transmiten líneas, el número de bytes a ser transmitidos por *write()* debería ser el del tamaño de la línea leída.

**ADMINISTRACIÓN CENTRAL DE DESARROLLO
Y MANTENIMIENTO DE APLICACIONES.
ADMINISTRACIÓN DE DESARROLLO PARA ADUANAS**

**PROTOCOLOS DE COMUNICACIÓN ENTRE EQUIPOS DE
TERCEROS (PREVALIDADORES) Y EQUIPOS DEL SAT**

6. Se repite el paso anterior hasta que la función *read()* devuelva un valor de cero, indicando que se ha terminado de leer el contenido del archivo. En este momento se bloquea la posibilidad de escritura al socket (véase la función *shutdown()*), indicando con esto el fin de la transmisión.

Recepción y análisis del segundo acuse.

7. Se recibe 1 byte (segundo acuse de recibo) como resultado de la verificación del número de bytes recibidos. Cuando el valor de este byte es diferente de 0, se tiene un error que debe interpretarse como se muestra en el anexo A para llevar a cabo las acciones correctivas correspondientes; a continuación se debe **cerrar el socket** y terminar la comunicación. Si el valor del byte recibido es cero, la información se ha recibido correctamente.

Cierre del socket.

8. Una vez concluida la transmisión de información, se debe cerrar por completo el socket. Se recomienda el uso combinado de las funciones *shutdown()* (en modo total) y *close()*. Es importante recalcar la importancia del cierre adecuado del socket una vez concluida la transmisión, pues de no hacerlo así, se puede dejar establecida la conexión en con el equipo remoto y tener activo el servicio de recepción. Con un número elevado de conexiones activas, se puede degradar significativamente el rendimiento del equipo del SAT.

Resumen de la secuencia de intercambio de información.

TRANSMISIÓN	RECEPCIÓN
13 bytes. Nombre del archivo (12 bytes) + carácter de identificación del equipo.	
4 bytes. Dato entero, el tamaño del archivo, del byte más significativo al menos significativo.	
	1 byte. Valor ASCII cero si todo está bien. Consulte anexo A para otro caso.
n bytes. Contenido del archivo, en bloques de 1024 bytes o línea por línea. Al terminar, bloquear la escritura en el socket (<i>shutdown()</i>)	
	1 byte. Valor ASCII cero si todo está bien. Consulte anexo A para otro caso.

2. SERVICIO DE RECEPCIÓN DE ARCHIVOS DE RESPUESTA DE VALIDACIÓN

Este servicio es la contraparte en el host local (prevalidador) de los programa encargados de la transmisión de archivos de respuesta desde el host remoto (equipo del SAT).

Para que el programa transmisor residente en el equipo del SAT pueda transmitir información, el programa que opera como servicio de recepción de archivos debe estar activo y a la escucha de una petición de conexión. Para que esto pueda lograrse, los servicios deben estar registrados en el archivo **C:\Windows\services** (para plataformas Windows 95 o 98). Para una plataforma Windows NT, el archivo es **C:\WinNT\system32\drivers\etc\services**. Para otras plataformas, consulte el manual respectivo.

Este programa lee desde el descriptor del socket correspondiente lo que le es enviado a través del socket utilizando la función *read()* o *recv()* y responde al programa transmisor por la misma vía con las funciones *write()* o *send()*.

Secuencia de intercambio de información.

Recepción de datos previos.

1. Se recibe una cadena de doce caracteres conteniendo el nombre del archivo, incluida la extensión (.err).
2. A continuación, se reciben cuatro bytes (un entero largo) que especifican el tamaño del archivo a recibir, en bytes. Estos bytes son transmitidos del más significativo al menos significativo. Si en el *host* receptor el orden de los bytes para el almacenamiento de datos enteros no es el mismo que el orden de transmisión en la red, se debe convertir, utilizando la función ***ntohl()*** (***net to host long***). Consulte en algún manual lo relativo a ésta función.

Envío del primer acuse de recibo.

3. Se revisa que el nombre de archivo recibido sea correcto en cuanto a sintaxis, además de otras condiciones que pudieran interferir con la recepción correcta de la información. Si todo está bien, se envía un byte con valor 0. En caso de error, el valor del byte enviado será alguno de los mostrados en el anexo A, según corresponda a la causa de error.
4. Se abre el archivo donde se almacenará la información recibida desde el equipo del SAT, con el nombre tal como se tiene en la cadena de doce caracteres de los datos previos y en modo escritura. Vea lo relativo a la función *open()* en el manual correspondiente.

Recepción de la información (contenido del archivo transmitido desde el SAT).

5. Con la función *read()* o *recv()* se recibe la información transmitida desde el equipo del SAT, en bloques de 1024 bytes. Los bytes recibidos se escriben inmediatamente al archivo donde se almacena la información. Estos dos pasos se repiten cíclicamente, hasta que la función de lectura regrese un valor de cero, en cuyo caso se cierra el archivo receptor y se compara el número de bytes recibidos contra el número de bytes especificado en los datos previos.

Envío del segundo acuse de recibo.

6. Si se recibieron tantos bytes como especificados en los datos previos, se envía un byte con valor cero (nulo). En otro caso debe enviar un valor tal como se especifica en el anexo A. En este momento termina la función del receptor.

**ADMINISTRACIÓN CENTRAL DE DESARROLLO
Y MANTENIMIENTO DE APLICACIONES.
ADMINISTRACIÓN DE DESARROLLO PARA ADUANAS**

**PROTOCOLOS DE COMUNICACIÓN ENTRE EQUIPOS DE
TERCEROS (PREVALIDADORES) Y EQUIPOS DEL SAT**

**ANEXO A
CÓDIGOS DE ERROR**

RECIBIDOS POR EL PROGRAMA TRANSMISOR DE ARCHIVOS A VALIDAR.

VALOR	SIGNIFICADO
Primer acuse de recibo.	
1	El servicio receptor en el equipo del SAT no pudo leer correctamente alguno de los datos previos: los 13 bytes del nombre del archivo o los cuatro bytes que especifican el tamaño. Cerrar el socket y reintentar.
2	Ya existe una copia del archivo en el equipo del SAT. Cerrar el socket. Se puede intentar la transmisión de la información en un archivo con nombre distinto.
- 1	Error grave en el host remoto (equipo del SAT). Dar aviso al operador para que corrija el problema.
Segundo acuse de recibo.	
1	Recepción incompleta de la información. Tamaño del archivo diferente al transmitido en los datos previos. Cerrar el socket y volver a transmitir la información.
- 1	Error grave en el host remoto (equipo del SAT). Dar aviso al operador para que corrija el problema.

ENVIADOS POR EL SERVICIO RECEPTOR DE ARCHIVOS DE RESPUESTA.

VALOR	SIGNIFICADO
Primer acuse de recibo.	
1	No se pudo leer correctamente alguno de los datos previos: los 12 bytes del nombre del archivo o los cuatro bytes que especifican el tamaño.
2	Ya existe una copia del archivo que se pretende recibir y ésta no se pudo respaldar (renombrar). Cerrar el socket. Revisar a detalle el por qué no se pudo efectuar el respaldo..
- 1	Error grave en el sistema (equipo local). No fue posible posicionarse en directorio de trabajo o abrir el archivo para recibir la información que transmitirá el equipo del SAT. Dar aviso al administrador del equipo para que corrija el problema.
Segundo acuse de recibo.	
1	Recepción incompleta de la información. Tamaño del archivo diferente al transmitido en los datos previos..
- 1	Error grave en el sistema (equipo local). No se pudo registrar recepción del archivo. Dar aviso al administrador del equipo para que corrija el problema.