



Interconexión con el Websocket

Interconexión con el Websocket de Innova360°

Introducción

Realizando una conexión contra el servidor de websocket de Innova360°, conseguimos recibir con todo detalle la información generada en el sistema por cada una de sus extensiones.

Todas las comunicaciones cliente/servidor, se realizan en formato JSON.

Disponemos de dos tipos de conexión:

- Supervisor: donde podremos comprobar el estado de todas las extensiones.
- Agente: donde podremos comprobar el estado de la extensión logada, además de disponer de determinadas acciones para lanzarlas sobre dicha extensión (realizar, transferir y colgar llamadas).

Conexión

Para establecer una conexión con el servidor, es necesario abrir una conexión a `ws.innova360.es` por el protocolo websocket.

Para conectarse correctamente hay que tener en cuenta dos cosas:

- El nivel de seguridad:

	URL	Puerto
Conexión sin cifrar	<code>ws://ws.innova360.es</code>	9000
Conexión cifrada	<code>wss://ws.innova360.es</code>	9090

El tipo de conexión:

- Supervisor (todas las cuentas): hay que añadir el directorio **ws** a la ruta de conexión.
- Agente (una única cuenta): hay que añadir el directorio **agent** a la ruta de conexión.

Para conectarse a un agente a través de una conexión segura, la URL de conexión sería la siguiente:

`wss://ws.innova360.es:9090/agent`

Autenticación

Una vez que se ha realizado la conexión al servidor, recibiremos un mensaje de bienvenida que nos solicitará autenticar la conexión.

Al que, deberemos responder con los datos de acceso adecuados dependiendo del tipo de conexión que hayamos escogido.

- Para conexiones de supervisor, los datos de acceso son proporcionados por InnoVA360°, y son los mismos que se utilizan para acceder al panel web.
- Para conexiones de agente, los datos de acceso son gestionados desde el panel web, donde a cada extensión se le podrá configurar un usuario y clave para acceder, tanto a la web de agente como para conectarse al servidor WebSocket.

Eventos

Cada evento recibido nos proporciona información en tiempo real sobre los cambios que ocurren en la extensión durante la conexión.

Estos son los eventos que nos podemos encontrar:

Evento	Datos	Descripción
Bienvenida	servername: 'BOBIf' version: '1.0' type: 'welcome' auth: 'required' server: <server>	Mensaje recibido justo después de realizar la conexión, donde solicita que se envíe la autenticación.
Autenticación correcta	extension: <extension> agent: <agent> server: <server> token: <token> auth: 'ok' type: 'auth-resp'	Si se han enviado datos de autenticación válidos, el sistema responde con un OK y los datos de la extensión. Justo después envía la iniciación de extensiones.
Autenticación incorrecta	type: 'auth-resp' auth: 'forbidden' server: <server>	Si se reciben datos erróneos, el sistema responde con este mensaje, por lo que habrá que volver a enviar la autenticación.
Iniciación de extensiones	data: <extension_data>* type: 'init' server: <server> *<extension_data>: Status: <status> Calls: <calls_data>** Extension: <extension> Hint: <hint> **<calls_data>: Monitored: true CallDirection: <calldirection> PhoneA: <phonea> PhoneB: <phoneb> Status: <call status> Bridge: <call bridged> StatusDesc: <status desc> DIDA: <dida> DIDB: <didb> CallID: <call_ID> Hold: <callonhold>	Una vez aceptadas las credenciales, se nos enviará un mensaje con el estado actual de la (o las) extensión y la información de todas sus llamadas si es que las hay.
Cambio de estado de extensión	server: <server> extension: <extension> type: 'update'	Cada vez que haya un cambio en la extensión (en espera, en uso, caída,

	action: 'extension-status' extension: <extension> status: <state>	sonando, etc...) recibiremos este evento, notificándonos de dicho cambio.
--	---	---

Evento	Datos	Descripción
Nueva llamada	server: <server> extension: <extension> type: 'update' action: 'new-call' callid: <call_ID> direction: <calldirection> phonea: <phonea> phoneb: <phoneb> dida: <dida> didb: <didb>	Cuando se genera una nueva llamada, ya sea entrante o saliente, nos llegará la información de toda la llamada (origen, destino, etc...)
Cambio de estado de llamada	extension: <extension> type: 'update' action: 'call-state' callid: <call_ID> state: <state> statedesc: <statedesc>	Cuando una llamada de la extensión monitorizada cambia de estado, recibimos la información actualizada a través de este evento.
Llamada colgada	server: <server> extension: <extension> type: 'update' action: 'hangup' callid: <call_ID> cause: <hangupcause>	Cuando una llamada de la extensión monitorizada es colgada, nos notifica cuelgue y de la causa del mismo.
Inicio de puesta en espera	server: <server> extension: <extension> type: 'update' action: 'start-moh' callid: <call_ID>	Recibimos este evento cuando una llamada en curso se pone en espera
Fin de puesta en espera	server: <server> extension: <extension> type: 'update' action: 'stop-moh' callid: <call_ID>	Recibimos este evento cuando una llamada en curso deja de estar en espera
Inicio de grabación de llamada	server: <server> extension: <extension> type: 'update' action: 'start-record' callid: <call_ID>	Recibimos este evento cuando se inicia la grabación en una llamada en curso, normalmente este evento suele llegar justo después de iniciar la llamada.
Fin de grabación de llamada	server: <server> extension: <extension> type: 'update' action: 'end-record' callid: <call_ID>	Recibimos este evento cuando una llamada en curso deja de ser grabada.

Acciones

Durante la conexión al servidor, podemos enviar diferentes acciones, con las cuales podemos autenticarnos en el sistema u ordenarle que realice determinadas funciones.

Las acciones disponibles son:

Acción	Datos	Descripción
Enviar autenticación	type: 'auth-plain' user: <user> passwd: <password>	Esta acción se envía al conectarse, para logarse contra el servidor. Las credenciales que se deben enviar vienen detalladas en la sección autenticación.
Originar llamada	type: 'action' action: 'call' phone: <phoneTo> [customfield1: <value>] [customfield2: <value>]	Una vez hemos accedido al sistema, podemos originar una llamada, especificando el número al que se quiere llamar. Esta acción generará una llamada en la extensión del agente, una vez que se conteste, se lanzará la llamada al número de destino. Además, si disponemos de acceso al servicio de obtención del registro de llamadas en CSV, podemos incluir dos campos personalizados en los que podremos establecer cualquier información referente a la llamada realizada, para posteriormente recibir dicha información en el CSV y así facilitar la identificación de la llamada o completar la información de la misma.
Transferir llamada	type: 'action' action: 'blindtransfer' call: <call_ID> phone: <phoneTo>	Cuando tenemos una llamada establecida, si conocemos el ID de la llamada (facilitado por los eventos de llamada), podemos transferirla a otro destino.
Colgar llamada	type: 'action' action: 'hangup' call: [<call_ID>, <call_ID>, ...]	Cuando tenemos una o varias llamadas en curso, podemos colgarlas si conocemos los IDs de las llamadas. Podemos colgar una o varias llamadas en simultáneo. Se deben establecer como una lista de valores, aunque solamente haya una llamada.