



API de Integria IMS



From:

<https://pandorafms.com/manual/!776/>

Permanent link:

https://pandorafms.com/manual/!776/es/documentation/10_pandora_itsm/21_pandora_itsm_api

2024/06/10 14:34



API de Integria IMS

[Volver al índice de documentación de Pandora FMS](#)

La API externa de Integria IMS se utiliza haciendo llamadas remotas vía HTTP o HTTPS sobre el fichero `/include/api.php`. Con este método es posible la integración de aplicaciones de terceros con Integria IMS. El uso del API se hace mediante una llamada con unos parámetros definidos que devuelve un valor o una lista de valores.

El prototipo de llamada al API es el siguiente:

```
http://<URL>/include/api.php<parameters>
```

Donde `< URL >` es la dirección IP o URL pública donde está instalado Integria IMS. El método API sólo acepta los siguientes parámetros (`< parameters >`), cualquier otro parámetro será ignorado:

- `user` (obligatorio): Nombre del usuario de Integria IMS que accede a los datos, se usa para verificar [las reglas ACL](#).
- `user_pass` (obligatorio): Contraseña del usuario.
- `op` (obligatorio): Operación a realizar. Las operaciones están detalladas más abajo.
- `pass` (obligatorio si está habilitado): Contraseña específica del API, configurado en la Consola de Integria IMS.
- `params`: Lista de parámetros que se pasaran a la función separados por un *token*.
- `token`: Carácter que separa los diferentes parámetros (por defecto es utilizada la coma ,).
- `return_type`: Formato del resultado de la llamada, puede ser XML o CSV (CSV por defecto).
- `info`: Parámetro especial con el que puede obtenerse la versión antes de autenticarse.

Seguridad

Autenticación por dirección IP válida

La seguridad se basa en una lista de direcciones IP que tendrán acceso a la herramienta. Esta lista se puede crear en el menú de configuración de Integria IMS.

Es importante tener en cuenta que este método de autenticación siempre está activo por lo que si no desea usarlo será necesario poner el carácter asterisco * para indicar que cualquier dirección IP está autorizada.

Se recomienda usar siempre una lista específica de direcciones IP.

Autenticación por contraseña

Dentro del menú de configuración de Integria IMS es posible establecer una contraseña específica para la API. Esta contraseña deberá enviarse en la llamada al API *sin cifrar*, por ello siempre se debe usar HTTPS si se quiere usar este método de autenticación.

Si esta contraseña está vacía, significa que la autenticación por contraseña está deshabilitada y es innecesario colocar el *token pass* en la llamada a la API.

Autenticación por usuario y contraseña de usuario

Es necesario en todas las llamadas (*excepto las de información*) especificar el usuario (`user`) y la contraseña del usuario (`user_pass`).

No hay forma de deshabilitar este paso de la autenticación.

Autenticación por bearer token

Versión 5.0.99 OUM 99 MR 93: Se puede autenticar con API Token por medio de el envío en las **cabeceras HTTP** de un **bearer token** generado por cada usuario y para su propio uso privado y particular. Consulte el menú People → Edit my user para más detalles.

La autenticación basada en *bearer token* permite que cada uno de los usuarios registrados en Pandora FMS pueda generar su propio identificador en correspondencia unívoca. Dicho *bearer token* debe ser insertado en las cabeceras HTTP, es decir, no forma parte de la URL completa de la API. El mecanismo de envío corresponde a la aplicación que realiza la conexión con la API de PFMS.

Para los siguientes ejemplos se utiliza el programa de línea de comandos cURL, versión 7.68 .

- Verifique la versión de curl con `curl -V`.
- Si necesita más ayuda para utilizar curl, invoque `curl -h`.
- Si necesita depurar la conexión utilice el parámetro `-v` (no confundir con `-V` usado para mostrar la versión).
- Los parámetros `-k` y `-H` sirven para aceptar conexiones con certificados autofirmados para HTTPS y enviar la información de las cabeceras, respectivamente.
- Consulte el *bearer token* correspondiente al usuario que realizará la consulta API.
- Ahora ya puede realizar una consulta API, en el siguiente ejemplo se utiliza `get_incidents` (sustituya por su *bearer token* consultado anteriormente):

```
BEARER="Authorization: Bearer 9ea069a7240411ee95580242ac100002"
```

```
URL="http://192.168.7.117/integria/include/api.php"  
API_QUERY="?return_type=xml&user=admin&op=get_incidents"  
curl -k -H "$BEARER" $URL$API_QUERY && echo ""
```

Nótese que los *bearer token* de distintos usuarios, aunque son bastante parecidos, en realidad son valores únicos.

Funciones API

Retorno de información sobre Integria IMS

Esta es una función especial que puede ser usada sin autenticarse, ya que la información que devuelve se puede obtener también de la pantalla de inicio de sesión de manera pública (si se tiene desactivado el *token* Hide version).

Ejemplo:

```
http://127.0.0.1/integria/include/api.php?info=version
```

Resultado de ejemplo:

```
Integria IMS OpenSource Edition 5.0.98 Build 221221 MR93
```

get_incidents

Resultado: Devuelve una lista con los *tickets* a los que tiene acceso el usuario que hace la llamada.

Parámetros serializados:

- Filtro de cadena: Muestra solamente los *tickets* que contienen esa cadena en el título o en la descripción (no obligatorio). Vacío para todos los *tickets*. Es indiferente el uso mayúsculas y minúsculas para la búsqueda.
- Filtro de estado: Identificar del estado (no obligatorio). Puede tomar los siguientes valores:
 - 0 → Todos (por defecto).
 - 1 → Nuevo.
 - 2 → No confirmado.
 - 3 → Asignado.
 - 4 → Re-abierto.
 - 5 → Pendiente de ser cerrado.
 - 6 → Pendiente de una tercera persona.
 - 7 → Cerrada.
 - -10 → Para todos excepto los cerrados.
- Filtro de grupo: Identificador del grupo (obligatorio). Uno (1) para todos los grupos.

- Filtro de prioridad: Identificador de la prioridad (*opcional*). Puede tener alguno los siguientes valores:
 - 10 → Mantenimiento.
 - 0 → Informativo.
 - 1 → Bajo.
 - 2 → Medio.
 - 3 → Grave.
 - 4 → Muy grave.
 - -1 → Cualquiera.
- Filtro de inventario: Identificador del inventario relacionado (*opcional*). Cero (0) para todos o ninguno.
- Filtro de propietario: Identificador del usuario al que está asignada la incidencia (*opcional*).
- Filtro de creador: Identificador del usuario que ha creado la incidencia (*opcional*).
- Filtro de incidencia: Identificador de incidencia (*opcional*).

Ejemplo de llamada con resultado en formato CSV:

```
http://127.0.0.1/integria/include/api.php?user=user&user_pass=userpass&pass=api_pass&op=get_incidents&params=day,-10,1,-1,0,admin,user1
```

Resultado de ejemplo con formato CSV:

```
5,"2016-01-19 11:19:42","0000-00-00
00:00:00",incidence_title,admin,1,1,2,"2014-01-19
11:19:42",,1,0,0,,1,0,0,1,0,mail@some.com,,0,"2014-01-20
14:45:01",,,,0,0,0,0,0,0,0
```

Ejemplo de llamada con resultado en XML:

```
http://127.0.0.1/integria/include/api.php?user=user&user_pass=userpass&pass=api_pass&op=get_incidents&params=day,-10,1,-1,0,admin,user1&return_type=xml
```

Resultado de ejemplo en XML:

```
<xml version='1.0' encoding='UTF-8'>
  <incident>
    <id_incidencia><![CDATA[5]]></id_incidencia>
    <inicio><![CDATA[2016-01-19 11:19:42]]></inicio>
    < cierre><![CDATA[0000-00-00 00:00:00]]></ cierre>
    < titulo><![CDATA[Titulo incidencia]]></ titulo>
    < descripcion><![CDATA[Descripcion]]></ descripcion>
    < id_usuario><![CDATA[admin]]></ id_usuario>
    < estado><![CDATA[1]]></ estado>
    < prioridad><![CDATA[1]]></ prioridad>
    < id_grupo><![CDATA[2]]></ id_grupo>
    < actualizacion><![CDATA[2016-01-19 11:19:42]]></ actualizacion>
    < id_creador><![CDATA[]]></ id_creador>
    < notify_email><![CDATA[1]]></ notify_email>
    < id_task><![CDATA[0]]></ id_task>
    < resolution><![CDATA[0]]></ resolution>
    < epilogo><![CDATA[]]></ epilogo>
```

```

<id_parent><![CDATA[1]]></id_parent>
<sla_disabled><![CDATA[0]]></sla_disabled>
<affected_sla_id><![CDATA[0]]></affected_sla_id>
<id_incident_type><![CDATA[1]]></id_incident_type>
<score><![CDATA[0]]></score>
<email_copy><![CDATA[copyto@someone.com]]></email_copy>
<editor><![CDATA[]]></editor>
<id_group_creator><![CDATA[0]]></id_group_creator>
<last_stat_check><![CDATA[2016-01-20 14:55:01]]></last_stat_check>
<closed_by><![CDATA[]]></closed_by>
<extra_data><![CDATA[]]></extra_data>
<extra_data2><![CDATA[]]></extra_data2>
<blocked><![CDATA[0]]></blocked>
<old_status><![CDATA[0]]></old_status>
<old_resolution><![CDATA[0]]></old_resolution>
<old_status2><![CDATA[0]]></old_status2>
<old_resolution2><![CDATA[0]]></old_resolution2>
<workunits_hours><![CDATA[0]]></workunits_hours>
<workunits_count><![CDATA[0]]></workunits_count>
</incident>
</xml>

```

get_incident_details

Resultado: Devuelve los detalles de un *ticket*.

Parámetros:

- Identificador del *ticket* (obligatorio).

Ejemplo de llamada con resultado en formato CSV:

```
http://127.0.0.1/integria/include/api.php?user=user&user_pass=userpass&pass=api_pass&op=get_incident_details&params=2
```

Resultado de ejemplo en formato CSV:

```
1,"2010-02-17 01:35:35","0000-00-00 00:00:00",Myfddfe,,demo,1,1,3,3,"2010-02-17 01:50:05",user,1,0,0,,0,0,0,0
```

Ejemplo de llamada con resultado en XML:

```
http://127.0.0.1/integria/include/api.php?user=user&user_pass=userpass&pass=api_pass&op=get_incident_details&params=2&return_type=xml
```

Resultado de ejemplo en XML:

```
<xml>
  <id_incidencia>1</id_incidencia>

```

```

<inicio>2010-02-17 01:35:35</inicio>
<cierre>0000-00-00 00:00:00</cierre>
<titulo>Myfddfe</titulo>
<descripcion></descripcion>
<id_usuario>demo</id_usuario>
<origen>1</origen>
<estado>1</estado>
<prioridad>3</prioridad>
<id_grupo>3</id_grupo>
<actualizacion>2010-02-17 01:50:05</actualizacion>
<id_creator>admin</id_creator>
<notify_email>1</notify_email>
<id_task>0</id_task>
<resolution>0</resolution>
<epilog></epilog>
<id_parent></id_parent>
<sla_disabled>0</sla_disabled>
<affected_sla_id>0</affected_sla_id>
<id_incident_type>0</id_incident_type>
<score>0</score>
</xml>

```

create_incident

Resultado: Se crea un *ticket*.

User: Es el creador del *ticket*.

Parámetros:

- Título (obligatorio). Los títulos vacíos no están admitidos. Para introducir comas en este parámetro habrá que sustituirlas por (: : :), para introducir un espacio debe utilizar %20 . Por ejemplo, para introducir Doe, John habrá que insertar en la URL: Doe : : :%20John .
- Identificador del grupo (obligatorio).
- Prioridad (*no obligatorio*). Puede tener estos valores:
 - 10 → 0 Maintenance (mantenimiento).
 - 0 → 1 Informative (informativo, *por defecto*).
 - 1 → 2 Low (prioridad baja).
 - 2 → 3 Medium (prioridad media).
 - 3 → 4 Serious (prioridad seria).
 - 4 → 5 Very serious (prioridad muy seria).
- Descripción (*no obligatorio*). Vacío por defecto. Para introducir comas en este parámetro habrá que sustituirlas por (: : :), para introducir un espacio debe utilizar %20 . Por ejemplo, para introducir Doe, John habrá que insertar en la URL: Doe : : :%20John .
- Identificador de inventario (*no obligatorio*). Se pueden crear incidencias con varios objetos de inventario y tendrán que ir separados entre sí por : .
- Identificador de tipo de incidencia (*no obligatorio*). Cero 0 (ninguno) por defecto.
- *Email* en copia (*no obligatorio*): Valor por defecto vacío.
- Propietario (*no obligatorio*). Por defecto admin.
- Identificador del *ticket* padre (*no obligatorio*). Por defecto, cero 0 (sin padre).

- Estado (obligatorio). Puede tomar los siguientes valores:
 - 1 → Nuevo (por defecto).
 - 2 → No confirmado.
 - 3 → Asignado.
 - 4 → Re-abierto.
 - 5 → Pendiente de ser cerrado.
 - 6 → Pendiente de una tercera persona.
 - 7 → Cerrada.
- Info_extra (no obligatorio).
- Resolución:
 - 0 → None (ninguno, por defecto).
 - 1 → Fixed (solucionado).
 - 2 → Invalid (inválido).
 - 3 → Won't fix (no se solucionará).
 - 4 → Duplicate (duplicado).
 - 5 → Works for me (“funciona para mí”).
 - 6 → Incomplete (incompleto).
 - 7 → Expired (caducado).
 - 8 → Moved (movido).
 - 9 → In process (en proceso).
- Info_extra2 (no obligatorio).
- Info_extra3 (no obligatorio).
- Info_extra4 (no obligatorio).
- CampoTipoIncidencia1.
- CampoTipoIncidencia2
- CampoTipoIncidencia3
- ...
- CampoTipoIncidenciaN.

Llamada de ejemplo con resultado en formato CSV:

```
http://127.0.0.1/integria/include/api.php?user=user&user_pass=userpass&pass=api_pass&op=create_incident¶ms=Title,2,1,Description%20de%20la%incidence,1:2:3,1,copyto@example.com,admin,0,1
```

Ejemplo de resultado en formato CSV:

```
0
```

Llamada de ejemplo con resultado en XML:

```
http://127.0.0.1/integria/include/api.php?user=user&user_pass=userpass&pass=api_pass&op=create_incident¶ms=Titulo%20de%20la%20incidencia,2,1,Descripcion%20de%20la%incidencia,1:2:3,1,copyto@someone.com,admin,0&return_type=xml
```

Resultado de ejemplo en XML:

```
<xml>
0
</xml>
```

update_incident

Resultado: Actualiza los datos de un *ticket*.

Parámetros:

- Identificador del *ticket* (obligatorio).
- Título (obligatorio). Los títulos vacíos no están admitidos.
- Descripción (*no obligatorio*). Vacío por defecto.
- Epílogo (*no obligatorio*). Vacío por defecto.
- Identificador del grupo (obligatorio).
- Prioridad (*no obligatorio*). Puede tener estos valores:
 - 10 → 0 Maintenance (mantenimiento).
 - 0 → 1 Informative (informativo, *por defecto*).
 - 1 → 2 Low (prioridad baja).
 - 2 → 3 Medium (prioridad media).
 - 3 → 4 Serious (prioridad seria).
 - 4 → 5 Very serious (prioridad muy seria).
- Resolución:
 - 0 → None (ninguno, por defecto).
 - 1 → Fixed (solucionado).
 - 2 → Invalid (inválido).
 - 3 → Won't fix (no se solucionará).
 - 4 → Duplicate (duplicado).
 - 5 → Works for me ("funciona para mí").
 - 6 → Incomplete (incompleto).
 - 7 → Expired (caducado).
 - 8 → Moved (movido).
 - 9 → In process (en proceso).
- Estado (obligatorio). Puede tomar los siguientes valores:
 - 1 → Nuevo (por defecto).
 - 2 → No confirmado.
 - 3 → Asignado.
 - 4 → Re-abierto.
 - 5 → Pendiente de ser cerrado.
 - 6 → Pendiente de una tercera persona.
 - 7 → Cerrada.
- Propietario (*no obligatorio*). Por defecto admin.
- Identificador del *ticket* padre (*no obligatorio*). Por defecto, cero 0 (sin padre).
- Identificador de tipo de incidencia (*no obligatorio*). Cero 0 (ninguno) por defecto.
- Info_extra (no obligatorio).
- Info_extra2 (no obligatorio).
- Info_extra3 (no obligatorio).
- Info_extra4 (no obligatorio).
- CampoTipoIncidencia1.
- CampoTipoIncidencia2
- CampoTipoIncidencia3
- ...
- CampoTipoIncidenciaN.

Si no se desea modificar el valor actual de los campos del tipo de *ticket* se deberán rellenar dichos campos con su valor correspondiente. En el caso de estar vacíos, se actualizarán sin valor.

Ejemplo de llamada con resultado en formato CSV:

```
http://127.0.0.1/integria/include/api.php?user=user&pass=api_pass&op=update_incident&params=4,nuevo%20titulo,nueva%20descripcion,nuevo%20epilogo,4,2,1,3,admin,0,6,,Soporte,Hardware
```

Ejemplo de llamada con resultado en XML::

```
http://127.0.0.1/integria/include/api.php?user=user&pass=api_pass&op=update_incident&params=4,nuevo%20titulo,nueva%20descripcion,nuevo%20epilogo,4,2,1,1,3,admin,0,6,Soporte,Hardware&return_type=xml
```

Ejemplo de resultado en XML:

```
<xml>  
0  
</xml>
```

add_inventory_to_ticket

Resultado: Permite asociar un objeto de inventario a un *ticket*.

Parámetros:

- Identificador del *ticket* (obligatorio).
- Identificador del objeto de inventario (obligatorio).

Ejemplo de llamada con resultado en formato CSV:

```
http://127.0.0.1/integria/include/api.php?user=user&user_pass=userpass&pass=api_pass&op=add_inventory_to_ticket&params=1,2
```

Resultado de ejemplo en CSV:

```
1
```

Ejemplo de llamada con resultado en XML:

```
http://127.0.0.1/integria/include/api.php?user=user&user_pass=userpass&pass=api_pass&op=add_inventory_to_ticket&params=1,2&return_type=xml
```

Resultado de ejemplo en XML:

```
<xml>
1
</xml>
```

delete_incident

Resultado: Se elimina un *ticket*.

Parámetros:

- Identificador del *ticket* (obligatorio).

Ejemplo de llamada con resultado en formato CSV:

```
http://127.0.0.1/integria/include/api.php?user=user&pass=api_pass&op=delete_incident&params=3
```

Ejemplo de resultado en formato CSV:

```
0
```

Ejemplo de llamada con resultado en XML:

```
http://127.0.0.1/integria/include/api.php?user=user&pass=api_pass&op=delete_incident&params=3&return_type=xml
```

Ejemplo de resultado en XML:

```
<xml>
0
</xml>
```

get_incident_workunits

Resultado: Devuelve una lista con los Workunits asociados a un *ticket*.

Parámetros:

- Identificador del *ticket* (obligatorio).

Ejemplo de llamada con resultado en formato CSV:

```
http://127.0.0.1/integria/include/api.php?user=user&user_pass=user_pass&pass=api_pass&op=get_incident_workunits&params=3
```

Resultado de ejemplo con formato CSV:

```
3,"2010-02-17 01:48:48",0.25,admin,"Workunit description",0,0,,1
2,"2010-02-17 01:47:41",0.25,admin,"Another workunit description",0,0,,1
```

Ejemplo de llamada con resultado en XML:

```
http://127.0.0.1/integria/include/api.php?user=user&user_pass=user_pass&pass=api_pass&op=get_incident_workunits&params=3&return_type=xml
```

Resultado de ejemplo en XML:

```
<xml>
  <workunit>
    <id>3</id>
    <timestamp>2010-02-17 01:48:48</timestamp>
    <duration>0.25</duration>
    <id_user>admin</id_user>
    <description>Workunit description</description>
    <have_cost>0</have_cost>
    <id_profile>0</id_profile>
    <locked></locked>
    <public>1</public>
  </workunit>
  <workunit>
    <id>2</id>
    <timestamp>2010-02-17 01:47:41</timestamp>
    <duration>0.25</duration>
    <id_user>admin</id_user>
    <description>Another workunit description</description>
    <have_cost>0</have_cost>
    <id_profile>0</id_profile>
    <locked></locked>
    <public>1</public>
  </workunit>
</xml>
```

create_workunit

Resultado: Crea un Workunit en un *ticket*.

Parámetros:

- Identificador del *ticket* (obligatorio).
- Descripción (*no obligatorio*). Por defecto, vacío.
- Duración en horas (*no obligatorio*). Por defecto 0.00 horas.
- Si ha tenido coste (*no obligatorio*). Por defecto, no.
- Si es público o no (*no obligatorio*). Por defecto, no.

- Identificador del perfil asociado (*no obligatorio*). Por defecto cero 0, sin perfil asociado.

Ejemplo de llamada con resultado en formato CSV:

```
http://127.0.0.1/integria/include/api.php?user=user&user_pass=userpass&pass=api_pass&op=create_workunit&params=1,Descripcion,0.5,0,1,1
```

Resultado de ejemplo con formato CSV:

```
0
```

Ejemplo de llamada con resultado en XML:

```
http://127.0.0.1/integria/include/api.php?user=user&user_pass=userpass&pass=api_pass&op=create_workunit&params=1,Descripcion,0.5,0,1,1&return_type=xml
```

Resultado de ejemplo en XML:

```
<xml>
0
</xml>
```

get_incident_files

Resultado: Devuelve una lista con los ficheros asociados a un *ticket*.

Parámetros:

- Identificador del *ticket* (obligatorio).

Ejemplo de llamada con resultado en formato CSV:

```
http://127.0.0.1/integria/include/api.php?user=user&user_pass=userpass=&pass=api_pass&op=get_incident_files&params=1
```

Resultado de ejemplo con formato CSV:

```
1,1,0,0,admin,file1.php,"php example",764026
2,1,0,0,admin,file2.png,"png example",34026
```

Ejemplo de llamada con resultado en XML:

```
http://127.0.0.1/integria/include/api.php?user=user&user_pass=userpass=&pass=api_pass&op=get_incident_files&params=1&return_type=xml
```

Resultado de ejemplo en XML:

```
<xml>
  <file>
    <id_attachment>1</id_attachment>
    <id_incidencia>1</id_incidencia>
    <id_task>0</id_task>
    <id_kb>0</id_kb>
    <id_usuario>admin</id_usuario>
    <filename>file1.php</filename>
    <description>php example</description>
    <size>764026</size>
  </file>
  <file>
    <id_attachment>2</id_attachment>
    <id_incidencia>1</id_incidencia>
    <id_task>0</id_task>
    <id_kb>0</id_kb>
    <id_usuario>admin</id_usuario>
    <filename>file2.png</filename>
    <description>png example</description>
    <size>34026</size>
  </file>
</xml>
```

download_file

Resultado: Devuelve el contenido de un fichero como texto, en Base64.

Parámetros:

- Identificador del fichero (obligatorio).

Ejemplo de llamada con resultado en formato CSV:

```
http://127.0.0.1/integria/include/api.php?user=user&user_pass=userpass&pass=api_
pass&op=download_file&params=32
```

Resultado de ejemplo con formato CSV:

```
[contenido del fichero en base64]
```

Ejemplo de llamada con resultado en XML:

```
http://127.0.0.1/integria/include/api.php?user=user&user_pass=userpass&pass=api_
pass&op=get_incidents&params=day,-10,1&return_type=xml
```

Resultado de ejemplo en XML:

```
<xml>
```

```
[contenido del fichero en base64]
</xml>
```

attach_file

Resultado: Añade un fichero a la lista de ficheros de un *ticket*.

Parámetros:

- Identificador del *ticket*: (*no obligatorio*), cero 0 por defecto.
- Nombre del fichero: (*no obligatorio*).
- Tamaño del fichero en Bytes: (*no obligatorio*), por defecto cero 0 .
- Descripción del fichero: (*no obligatorio*), por defecto vacío.
- Contenido del fichero en Base64 (*no obligatorio*) por defecto se crea con contenido vacío.

Ejemplo de llamada con resultado en formato CSV:

```
http://127.0.0.1/integria/include/api.php?user=user&user_pass=userpass&pass=api_
pass&op=attach_file&params=1,example.txt,234,Fichero
%20de%20ejemplo,aW50ZWdyawEgcm9ja3M=
```

Resultado de ejemplo con formato CSV:

```
0
```

Ejemplo de llamada con resultado en XML:

```
http://127.0.0.1/integria/include/api.php?user=user&user_pass=userpass&pass=api_
pass&op=attach_file&params=1,example.txt,234,Fichero
%20de%20ejemplo,aW50ZWdyawEgcm9ja3M=&return_type=xml
```

Resultado de ejemplo en XML:

```
<xml>
0
</xml>
```

delete_file

Resultado: Elimina un fichero de la lista de ficheros de un *ticket*.

Parámetros:

- Identificador del fichero (*obligatorio*).

Ejemplo de llamada con resultado en formato CSV:


```
http://127.0.0.1/integria/include/api.php?user=user&user_pass=userpass&pass=api_pass&op=delete_file&params=32
```

Resultado de ejemplo con formato CSV:

```
0
```

Ejemplo de llamada con resultado en XML:

```
http://127.0.0.1/integria/include/api.php?user=user&user_pass=userpass&pass=api_pass&op=delete_file&params=32&return_type=xml
```

Resultado de ejemplo en XML:

```
<xml>
0
</xml>
```

create_user

Resultado: Llamada usada para crear un usuario y asignarlo a un rol o grupo. Para asignar más perfiles hay que usar una llamada diferente. Devuelve uno 1 si se crea correctamente, cero 0 en caso contrario.

Parámetros:

- Username (obligatorio). Para introducir comas en este parámetro habrá que sustituirla por (: : :), para introducir un espacio debe utilizar %20 . Por ejemplo, para introducir Doe , John habrá que insertar en la URL: Doe : : :%20John .
- Identificador del grupo (obligatorio).
- Identificador del perfil (obligatorio).
- *Email (no obligatorio)*. Por defecto vacío.
- *Password (no obligatorio)*. Contraseña del usuario, por defecto es el nombre del usuario.
- Descripción (*no obligatorio*). Por defecto vacío. Para introducir comas en este parámetro habrá que sustituirla por (: : :), para introducir un espacio debe utilizar %20 . Por ejemplo, para introducir Doe , John habrá que insertar en la URL: Doe : : :%20John .
- Nombre Real (*no obligatorio*). Por defecto vacío.
- *Externaluser (no obligatorio)*. Por defecto cero 0.

Ejemplo de llamada con resultado en formato CSV:

```
http://127.0.0.1/integria/include/api.php?user=user&user_pass=userpass&pass=api_pass&op=create_user&params=jorge,3,4,jorge@lux.com,userpass,description,Jorge%20Apellido,1
```

Resultado de ejemplo con formato CSV:

```
1
```

Ejemplo de llamada con resultado en XML:

```
http://127.0.0.1/integria/include/api.php?user=user&user_pass=userpass&pass=api_pass&op=create_user&params=jorge,3,4,jorge@lux.com,userpass,description,Jorge%20Apellido,1&return_type=xml
```

Resultado de ejemplo en XML:

```
<xml>
1
</xml>
```

validate_user

Resultado: Devuelve uno 1 si la autenticación es correcta, cero 0 en caso contrario.

Parámetros:

- User: Nombre del usuario registrado que se quiere validar (obligatorio).
- Pass: Contraseña del usuario (obligatorio).

Ejemplo de llamada con resultado en formato CSV:

```
http://127.0.0.1/integria/include/api.php?&user=user&user_pass=userpass&pass=api_pass&op=validate_user&params=jorge,jorge_pass
```

Resultado de ejemplo con formato CSV:

```
1
```

Ejemplo de llamada con resultado en XML:

```
http://127.0.0.1/integria/include/api.php?&user=user&user_pass=userpass&pass=api_pass&op=validate_user&params=jorge,jorge_pass
&return_type=xml
```

Resultado de ejemplo en XML:

```
<xml>
1
</xml>
```

delete_user

Resultado: Llamada usada para borrar un usuario. Devuelve uno 1 si se borra correctamente, cero 0 en caso contrario.

admin no puede ser borrado a través de la API.

Parámetros

- Usuario: (obligatorio).

Ejemplo de llamada con resultado en formato CSV:

```
http://127.0.0.1/integria/include/api.php?user=user&user_pass=userpass&pass=api_pass&op=delete_user&params=pepe
```

Resultado de ejemplo con formato CSV:

```
1
```

Ejemplo de llamada con resultado en XML:

```
http://127.0.0.1/integria/include/api.php?user=user&user_pass=userpass&pass=api_pass&op=delete_user&params=pepe&return_type=xml
```

Resultado de ejemplo en XML:

```
<xml>
1
</xml>
```

user_exists

Resultado: Llamada usada para verificar si un usuario existe. Devuelve uno 1 si existe, cero 0 si no existe.

Parámetros:

- Usuario: (obligatorio).

Ejemplo de llamada con resultado en formato CSV:

```
http://127.0.0.1/integria/include/api.php?user=user&user_pass=userpass&pass=api_pass&op=user_exists&params=pepe
```

Resultado de ejemplo con formato CSV:

1

Ejemplo de llamada con resultado en XML:

```
http://127.0.0.1/integria/include/api.php?user=user&user_pass=userpass&pass=api_pass&op=user_exists&params=pepe&return_type=xml
```

Resultado de ejemplo en XML:

```
<xml>
1
</xml>
```

create_lead

Resultado: Llamada usada para crear un *lead*. Devuelve uno 1 si se crea correctamente, de lo contrario devuelve cero 0.

Parámetros:

- Nombre completo (obligatorio). Para introducir comas en este parámetro habrá que sustituirlas por (: : :), para introducir un espacio debe utilizar %20. Por ejemplo, para introducir Doe, John habrá que insertar en la URL: Doe : : :%20John.
- Compañía (obligatorio).
- *Email* (obligatorio).
- País (*no obligatorio*). Por defecto vacío.
- Venta estimada (*no obligatorio*). Cero 0 por defecto.
- Progreso (en tanto por ciento, *no obligatorio*). Por defecto 0%.
- Teléfono (*no obligatorio*). Por defecto, vacío.
- Teléfono móvil (*no obligatorio*). Por defecto, vacío.
- Posición (*no obligatorio*). Por defecto vacío.
- Propietario (obligatorio). Nombre del usuario.
- Idioma: Código de idioma (obligatorio).
 - de → *Deutch* (alemán).
 - en_GB → *English* (inglés).
 - es → Español.
 - fr → *Français* (francés).
 - pl → *Polski* (polaco).
 - ru → *Русский* (ruso).
 - zh_CN → 简化字 (chino).
- Comentarios (*no obligatorio*). Para introducir comas en este parámetro habrá que sustituirlas por (: : :), para introducir un espacio debe utilizar %20. Por ejemplo, para introducir Doe, John habrá que insertar en la URL: Doe : : :%20John.
- Identificador de categoría (*no obligatorio*). Por defecto cero 0 (ninguna).
- Dirigido por (*no obligatorio*). Por defecto cero 0.
- Campaña (*no obligatorio*). Por defecto cero 0.
- Separador de etiquetas o *tags* (obligatorio si se van a añadir varias *tags*).
- Nombres de *tags*: Divididos con el separador declarado anteriormente. Si los *tags* no existen, no se almacenarán.

Ejemplo de llamada con resultado en formato CSV:

```
http://127.0.0.1/integria/include/api.php?user=user&user_pass=userpass&pass=api_pass&op=create_lead¶ms=Dario%Rodriguez%Perez,Artica+ST,drp@example.com,Spain,230,0,55566677,6665577,Developer,admin,es,This%20comment%20is%20very%20important,1,0,7,-%7C-,Software-%7C-Hardware-%7C-Tag%20con%20espacios
```

Resultado de ejemplo con formato CSV:

```
1
```

Ejemplo de llamada con resultado en XML:

```
http://127.0.0.1/integria/include/api.php?user=user&user_pass=userpass&pass=api_pass&op=create_lead¶ms=Dario%Rodriguez%Perez,Artica+ST,drp@example.com,Spain,230,0,55566677,6665577,Developer,admin,es,This%20comment%20is%20very%20important,1,0,7,-%7C-,Software-%7C-Hardware-%7C-Tag%20con%20espacios&return_type=xml
```

Resultado de ejemplo en XML:

```
<xml>
1
</xml>
```

get_last_invoice_id

Resultado: Llamada usada para obtener el último identificador de factura. Devuelve el último identificador de factura, vacío si es infructuoso.

Parámetros: ninguno.

Ejemplo de llamada con resultado en formato CSV:

```
http://127.0.0.1/integria/include/api.php?user=user&user_pass=userpass&pass=api_pass&op=get_last_invoice_id
```

Resultado de ejemplo con formato CSV:

```
140601_0082
```

Ejemplo de llamada con resultado en XML:

```
http://127.0.0.1/integria/include/api.php?user=user&user_pass=userpass&pass=api_pass&op=get_last_invoice_id&return_type=xml
```

Resultado de ejemplo en XML:

```
<xml>
  <last_invoice_id>
    <id><![CDATA[140601_0082]]></id>
  </last_invoice_id>
</xml>
```

get_invoice

Resultado: Llamada para obtener la información de una factura. Devuelve la información de la factura, vacío en caso contrario.

Parámetros:

- Identificador de la factura: Es el valor del campo Bill ID (obligatorio).

Ejemplo de llamada con resultado en formato CSV:

```
http://127.0.0.1/integria/include/api.php?user=user&user_pass=userpass&pass=api_pass&op=get_invoice&params=140601_0071
```

Resultado de ejemplo con formato CSV:

```
2,admin,,1,140601_0071,ads,,,,,222.00,0.00,0.00,0.00,0.00,0.00,EUR,,0,0,,2014-01-24,0000-00-00,pending,,,Submitted,en_GB
```

Ejemplo de llamada con resultado en XML:

```
http://127.0.0.1/integria/include/api.php?user=user&user_pass=userpass&pass=api_pass&op=get_invoice&params=140601_0071&return_type=xml
```

Resultado de ejemplo en XML:

```
<xml>
  <invoice>
    <id><![CDATA[2]]></id>
    <id_user><![CDATA[admin]]></id_user>
    <id_task><![CDATA[]]></id_task>
    <id_company><![CDATA[1]]></id_company>
    <bill_id><![CDATA[140601_0071]]></bill_id>
    <concept1><![CDATA[ads]]></concept1>
    <concept2><![CDATA[]]></concept2>
    <concept3><![CDATA[]]></concept3>
    <concept4><![CDATA[]]></concept4>
    <concept5><![CDATA[]]></concept5>
    <amount1><![CDATA[222.00]]></amount1>
    <amount2><![CDATA[0.00]]></amount2>
    <amount3><![CDATA[0.00]]></amount3>
    <amount4><![CDATA[0.00]]></amount4>
```

```

<amount5><![CDATA[0.00]]></amount5>
<tax><![CDATA[0.00]]></tax>
<currency><![CDATA[EUR]]></currency>
<description><![CDATA[]]></description>
<id_attachment><![CDATA[0]]></id_attachment>
<locked><![CDATA[0]]></locked>
<locked_id_user><![CDATA[]]></locked_id_user>
<invoice_create_date><![CDATA[2014-01-24]]></invoice_create_date>
<invoice_payment_date><![CDATA[0000-00-00]]></invoice_payment_date>
<status><![CDATA[pending]]></status>
<reference><![CDATA[]]></reference>
<internal_note><![CDATA[]]></internal_note>
<invoice_type><![CDATA[Submitted]]></invoice_type>
<id_language><![CDATA[en_GB]]></id_language>
</invoice>
</xml>

```

create_invoice

Resultado: Llamada para crear una factura. Devuelve dos valores, el código de estado (uno 1 si se realizó la operación, cero 0 si se detectó un fallo) y un mensaje con el resultado de la operación.

Parámetros:

- Identificador del usuario (debe existir o vacío).
- Identificador de tarea (*no obligatorio*). Cero 0 por defecto.
- Identificador de compañía (obligatorio).
- Identificador de factura (obligatorio).
- Concepto (obligatorio). Para introducir comas en este parámetro habrá que sustituirlas por (: : :), para introducir un espacio debe utilizar %20 . Por ejemplo, para introducir Doe , John habrá que insertar en la URL: Doe : : :%20John .
- Cantidad (obligatorio).
- Impuestos: Porcentaje de impuestos (*no obligatorio*). Por defecto cero 0.
- Moneda: Cadena de texto (obligatorio).
- Descripción: (*no obligatorio*). Por defecto vacío. Para introducir comas en este parámetro habrá que sustituirlas por (: : :), para introducir un espacio debe utilizar %20 . Por ejemplo, para introducir Doe , John habrá que insertar en la URL: Doe : : :%20John .
- *Flag* bloqueado: *Booleano (no obligatorio)*. Por defecto cero 0 (no bloqueado).
- Identificador usuario bloqueador (obligatorio si *flag* bloqueado está activo). Por defecto vacío.
- Fecha de creación (*no obligatorio*). Por defecto el día actual.
- Fecha de pago: (*no obligatorio*). Por defecto el día actual.
- Estado (*no obligatorio*). Por defecto pendiente.
- Referencia (*no obligatorio*). Por defecto vacío.
- Nota interna (*no obligatorio*). Por defecto vacío.
- Tipo de factura (obligatorio: Submitted o Received).
- Identificador idioma (*no obligatorio*).
 - de → *Deutch* (alemán).
 - en_GB → *English* (inglés).
 - es → Español.
 - fr → *Français* (francés).

- pl → *Polski* (polaco).
- ru → *Русский* (ruso).
- zh_CN → 简化字 (chino).

Ejemplo de llamada sin errores con resultado en formato CSV:

```
http://127.0.0.1/integria/include/api.php?user=user&user_pass=userpass&pass=api_pass&pass=1234&op=create_invoice¶ms=admin,,1,140601_0082,paso,3.00,,EUR,,,,,paid,,,Submitted,
```

Resultado de ejemplo sin errores con formato CSV:

```
1,"invoice created"
```

Resultado de ejemplo con errores con formato CSV:

```
0,"invalid billing id"
```

Ejemplo de llamada con resultado en XML:

```
http://127.0.0.1/integria/include/api.php?user=user&user_pass=userpass&pass=api_pass&op=create_invoice¶ms=admin,,1,140601_0082,paso,3.00,,EUR,,,,,paid,,,Submitted,&return_type=xml
```

Resultado de ejemplo sin errores en XML:

```
<xml>
  <invoice>
    <status><![CDATA[1]]></status>
    <error><![CDATA[invoice created]]></error>
  </invoice>
</xml>
```

Resultado de ejemplo con errores en XML:

```
<xml>
  <invoice>
    <status><![CDATA[0]]></status>
    <error><![CDATA[invalid billing id]]></error>
  </invoice>
</xml>
```

create_company

Resultado: Devuelve el identificador de la compañía creada.

Parámetros:

- Nombre (obligatorio). Para introducir comas en este parámetro habrá que sustituirlas por (: : :), para introducir un espacio debe utilizar %20 . Por ejemplo, para introducir Doe, John habrá que insertar en la URL: Doe : : :%20John .
- Dirección (*no obligatorio*). Por defecto vacío. Para introducir comas en este parámetro habrá que sustituirlas por (: : :), para introducir un espacio debe utilizar %20 . Por ejemplo, para introducir Doe, John habrá que insertar en la URL: Doe : : :%20John .
- Identificación fiscal (*no obligatorio*). Por defecto vacío.
- Identificador del rol de compañía (*no obligatorio*). Por defecto cero 0 (ningún rol).
- País (*no obligatorio*). Por defecto, vacío.
- *Manager*: Nombre del usuario que se encarga de la compañía (debe existir o ponerlo vacío).
- Identificador de la compañía padre (*no obligatorio*). Por defecto cero 0.

Ejemplo de llamada con resultado en formato CSV:

```
http://127.0.0.1/integria/include/api.php?user=user&user_pass=userpass&pass=api_pass&op=create_company¶ms=Artica,AlbertoAguilera,12312312,1,Pais,admin,0
```

Resultado de ejemplo con formato CSV:

```
6
```

Ejemplo de llamada con resultado en XML:

```
http://127.0.0.1/integria/include/api.php?user=user&user_pass=userpass&pass=api_pass&op=create_company¶ms=Artica,AlbertoAguilera,12312312,1,Pais,admin,0&return_type=xml
```

Resultado de ejemplo en XML:

```
<xml>
  <company>
    <id><![CDATA[6]]></id>
  </company>
</xml>
```

add_address_to_newsletter

Resultado: Devuelve el identificador de la dirección de correo creada. Si falla, devuelve un valor cero 0.

Parámetros:

- *Id newsletter* (obligatorio y debe de existir).
- Nombre (obligatorio).
- *Email* (obligatorio).

Ejemplo de llamada con resultado en formato CSV:

```
http://127.0.0.1/integria/include/api.php?user=user&user_pass=userpass&pass=integria&op=add_address_to_newsletter&params=2,juan,juan@mail.com
```

Resultado

```
10
```

Ejemplo de llamada con resultado en XML:

```
http://127.0.0.1/integria/include/api.php?user=user&user_pass=userpass&pass=integria&op=add_address_to_newsletter&params=2,juan,juan@mail.com&return_type=xml
```

Resultado de ejemplo en XML:

```
<xml>
10
</xml>
```

get_incident_tracking

Resultado: Devuelve una lista con el seguimiento o *tracking* de un *ticket*.

Parámetros:

- Identificador del *ticket* (obligatorio).

Ejemplo de llamada con resultado en formato CSV:

```
http://127.0.0.1/integria/include/api.php?user=user&user_pass=userpass&pass=api_pass&op=get_incident_tracking&params=2
```

Resultado de ejemplo con formato CSV:

```
9,2,0,"2010-02-17 01:56:33",demo,0,Created
10,2,2,"2010-02-17 02:06:58",demo,0,"Workunit added"
```

Ejemplo de llamada con resultado en XML:

```
http://127.0.0.1/integria/include/api.php?user=user&user_pass=userpass&pass=api_pass&op=get_incident_tracking&params=2&return_type=xml
```

Resultado de ejemplo en XML:

```
<xml>
  <tracking>
    <id_it>9</id_it>
    <id_incident>2</id_incident>
```

```

        <state>0</state>
        <timestamp>2010-02-17 01:56:33</timestamp>
        <id_user>demo</id_user>
        <id_aditonal>0</id_aditonal>
        <description>Created</description>
</tracking>
<tracking>
        <id_it>10</id_it>
        <id_incident>2</id_incident>
        <state>2</state>
        <timestamp>2010-02-17 02:06:58</timestamp>
        <id_user>demo</id_user>
        <id_aditonal>0</id_aditonal>
        <description>Workunit added</description>
</tracking>
</xml>

```

get_incidents_resolutions

Resultado: Devuelve una lista con los identificadores y nombres de las resoluciones de los *tickets* de Integria IMS.

Parámetros: Ninguno.

Ejemplo de llamada con resultado en formato CSV:

```
http://127.0.0.1/integria/include/api.php?user=user&user_pass=userpass&pass=api_pass&op=get_incidents_resolutions
```

Resultado de ejemplo con formato CSV:

```
1,Fixed
2,Invalid
3,"Wont fix"
```

Ejemplo de llamada con resultado en XML:

```
http://127.0.0.1/integria/include/api.php?user=user&user_pass=userpass&pass=api_pass&op=get_incidents_resolutions&return_type=xml
```

Resultado de ejemplo en XML:

```

<xml>
  <resolution>
    <id>1</id>
    <name>Fixed</name>
  </resolution>
  <resolution>

```

```
        <id>2</id>
        <name>Invalid</name>
    </resolution>
    <resolution>
        <id>3</id>
        <name>Wont fix</name>
    </resolution>
</xml>
```

get_incidents_status

Resultado: Devuelve una lista con los identificadores y nombres de los estados de los *tickets* de Integria IMS.

Parámetros: Ninguno.

Ejemplo de llamada con resultado en formato CSV:

```
http://127.0.0.1/integria/include/api.php?user=user&user_pass=userpass&pass=api_pass&op=get_incidents_status
```

Resultado de ejemplo con formato CSV:

```
1,New
2,Unconfirmed
3,Assigned
```

Ejemplo de llamada con resultado en XML:

```
http://127.0.0.1/integria/include/api.php?user=user&user_pass=userpass&pass=api_pass&op=get_incidents_status&return_type=xml
```

Resultado de ejemplo en XML:

```
<xml>
  <status>
    <id>1</id>
    <name>New</name>
  </status>
  <status>
    <id>2</id>
    <name>Unconfirmed</name>
  </status>
  <status>
    <id>3</id>
    <name>Assigned</name>
  </status>
</xml>
```

get_incident_priorities

Resultado: Devuelve una lista con los identificadores y nombres de las prioridades de los *tickets* de Integria IMS.

Parámetros: Ninguno.

Ejemplo de llamada con resultado en formato CSV:

```
http://127.0.0.1/integria/include/api.php?user=user&user_pass=userpass&pass=api_pass&op=get_incident_priorities
```

Resultado de ejemplo con formato CSV:

```
10,Mantenimiento
0,Informativo
1,Bajo
2,Medio
3,Grave
4,Muy grave
```

Ejemplo de llamada con resultado en XML:

```
http://127.0.0.1/integria/include/api.php?user=user&user_pass=userpass&pass=api_pass&op=get_incident_priorities&return_type=xml
```

Resultado de ejemplo en XML:

```
<xml>
  <priority>
    <id>10</id>
    <name>Mantenimiento</name>
  </priority>
  <priority>
    <id>0</id>
    <name>Informativo</name>
  </priority>
  <priority>
    <id>1</id>
    <name>Bajo</name>
  </priority>
  <priority>
    <id>2</id>
    <name>Medio</name>
  </priority>
  <priority>
    <id>3</id>
```

```

        <name>Grave</name>
    </priority>
    <priority>
        <id>4</id>
        <name>Muy grave</name>
    </priority>
</xml>

```

get_incident_fields

Resultado: Devuelve los campos personalizados de un tipo de *ticket* dado. Si además se especifica un identificador de *ticket*, se devolverán sus valores.

Parámetros:

- Identificador del tipo de *ticket* (obligatorio).
- Identificador del *ticket* (opcional).

Ejemplo de llamada con resultado en formato CSV:

```
http://127.0.0.1/integria/include/api.php?user=user&user_pass=userpass&pass=api_pass&op=get_incident_fields&params=4,32
```

Resultado de ejemplo con formato CSV:

```
cHJvYmFuZG8mI3gyMDtsaXN0YQ==,8,8,4,4,"probando lista","probando lista",text,text,,,1,1,,,0,0,,,1,1,"prueba de contenido",0
```

Ejemplo de llamada con resultado en XML:

```
http://127.0.0.1/integria/include/api.php?user=user&user_pass=userpass&pass=api_pass&op=get_incident_fields&params=4,32&return_type=xml
```

Resultado de ejemplo en XML:

```

<xml>
  <field>
    <label_enco>cHJvYmFuZG8mI3gyMDtsaXN0YQ==</label_enco>
    <id>8</id>
    <id_incident_type>4</id_incident_type>
    <label>probando lista</label>
    <type>text</type>
    <combo_value></combo_value>
    <show_in_list>1</show_in_list>
    <parent>0</parent>
    <linked_value></linked_value>
    <order>1</order>
    <data>prueba de contenido</data>
  </field>
</xml>

```

```
<blocked>0</blocked>
</field>
</xml>
```

get_groups

Resultado: Devuelve una lista con los identificadores y nombres de los grupos de Integria IMS.

Parámetros: Ninguno.

Ejemplo de llamada con resultado en formato CSV:

```
http://127.0.0.1/integria/include/api.php?user=user&user_pass=userpass&pass=api_
pass&op=get_groups
```

Resultado de ejemplo con formato CSV:

```
2,"Customer #A"
3,"Customer #B"
4,Engineering
```

Ejemplo de llamada con resultado en XML:

```
http://127.0.0.1/integria/include/api.php?user=user&user_pass=userpass&pass=api_
pass&op=get_groups&return_type=xml
```

Resultado de ejemplo en XML:

```
<xml>
  <group>
    <id>1</id>
    <name>"Customer #A"</name>
  </group>
  <group>
    <id>2</id>
    <name>"Customer #B"</name>
  </group>
  <group>
    <id>3</id>
    <name>Engineering</name>
  </group>
</xml>
```

get_users

Resultado: Devuelve una lista con los usuarios sobre los que tiene visibilidad el usuario que hace la llamada.

Parámetros: Ninguno.

Ejemplo de llamada con resultado en formato CSV:

```
http://127.0.0.1/integria/include/api.php?user=user&user_pass=userpass&pass=api_pass&op=get_users
```

Resultado de ejemplo con formato CSV:

```
usuario1  
usuario2  
usuario3
```

Ejemplo de llamada con resultado en XML:

```
http://127.0.0.1/integria/include/api.php?user=user&user_pass=userpass&pass=api_pass&op=get_incidents&params=day,-10,1&return_type=xml
```

Resultado de ejemplo en XML:

```
<xml>  
  <id_user>usuario1</id_user>  
  <id_user>usuario2</id_user>  
  <id_user>usuario3</id_user>  
</xml>
```

get_stats

Resultado: Devuelve un valor correspondiente con la consulta realizada.

Parámetros:

- **metric:** Métrica de valores estadísticos, pueden ser los siguientes:
 - **total_incidents:** Número total de *tickets* (limitado por el número máximo de *tickets* de Integria IMS).
 - **opened:** Número total de *tickets* abiertos.
 - **closed:** Número total de *tickets* cerrados.
 - **avg_life:** Vida media de los *tickets* en segundos.
 - **sla_compliance:** Porcentaje de cumplimiento de la SLA.
 - **avg_scoring:** Puntuación media de los *tickets* de 0 a 10.
 - **avg_worktime:** Tiempo medio de trabajo en el *ticket* en segundos.
 - **string:** Cadena para filtrar la búsqueda de *tickets*.
 - **status:** Cadena con valores separados por comas, puede tomar valores entre 1 y 7 (por defecto 1,2,3,4,5,6,7).
 - **id_user:** Filtrado por usuario (por defecto "").
 - **id_group:** Filtrado por grupo (por defecto uno 1).
 - **id_company:** Filtrado por compañía (por defecto cero 0).
 - **id_inventory:** Filtrado por inventario (por defecto cero 0).

- **priority**: Cadena con valores separados por comas, puede tomar los valores 0 (Informativo), 1 (Bajo), 2 (Medio), 3 (Grave), 4 (Muy grave) y 10 (Mantenimiento). Por defecto, todos.

Ejemplo de llamada para obtener porcentaje de SLA con resultado en formato CSV:

```
http://127.0.0.1/integria/include/api.php?user=user&user_pass=userpass&pass=api_pass&op=get_stats¶ms=sla_compliance
```

Obtener el total de *tickets* abiertos con estado NUEVO (codigo 1), en cualquier grupo:

```
http://127.0.0.1/integria/include/api.php?user=user&user_pass=userpass&pass=api_pass&op=get_stats¶ms=opened,,1
```

Obtener el total de tickets abiertos (cualquier estado, en cualquier grupo).

```
http://127.0.0.1/integria/include/api.php?user=user&user_pass=userpass&pass=api_pass&op=get_stats¶ms=opened
```

Resultado de ejemplo con formato CSV:

```
1,"Sample inventory object"  
2,"Another inventory"  
3,"Thirdventory"
```

Ejemplo de llamada con resultado en XML:

```
http://127.0.0.1/integria/include/api.php?user=user&user_pass=userpass&pass=api_pass&op=get_stats¶ms=sla_compliance&return_type=xml
```

Resultado de ejemplo en XML:

```
<xml>  
97  
</xml>
```

get_incidents_summary

Resultado: Devuelve el resumen del estado de los *tickets*.

Parámetros: Campo, si se añade el campo solo mostrará dicho valor. Los campos permitidos son:

- new .
- unconfirmed .
- assigned .
- reopened .
- verified .
- pending_third_person .

- closed .
- not_closed .
- total .

Ejemplo de llamada con resultado en formato CSV:

```
http://127.0.0.1/integria/include/api.php?user=user&user_pass=userpass&pass=api_pass&op=get_incidents_summary&return_type=csv
```

Resultado de ejemplo con formato CSV:

```
NEW, 1
UNCONFIRMED, 0
ASSIGNED, 70
REOPENED, 3
VERIFIED, 12
PENDING_THIRD_PERSON, 308
CLOSED, 3382
NOT_CLOSED, 393
TOTAL, 3776
```

Ejemplo de llamada con resultado en XML:

```
http://127.0.0.1/integria/include/api.php?user=user&user_pass=userpass&pass=api_pass&op=get_incidents_summary&return_type=xml
```

Resultado de ejemplo en XML:

```
<xml>
  <new>1</new>
  <unconfirmed>0</unconfirmed>
  <assigned>70</assigned>
  <reopened>3</reopened>
  <verified>12</verified>
  <pending_third_person>308</pending_third_person>
  <closed>3382</closed>
  <not_closed>393</not_closed>
  <total>3776</total>
</xml>
```

Ejemplo de llamada con un campo específico y resultado en formato CSV:

```
http://127.0.0.1/integria/include/api.php?user=user&user_pass=userpass&pass=api_pass&op=get_incidents_summary&return_type=csv&params=not_closed
```

Resultado de ejemplo con formato CSV:

393

Ejemplo de llamada con resultado en XML:

```
http://127.0.0.1/integria/include/api.php?user=user&user_pass=userpass&pass=api_pass&op=get_incidents_summary&return_type=xml&params=not_closed
```

Resultado de ejemplo en XML:

```
<xml>
  <data>393</data>
</xml>
```

get_inventories

Resultado: Devuelve una lista con los identificadores y nombres de los inventarios de Integria IMS.

Parámetros: Ninguno.

Ejemplo de llamada con resultado en formato CSV:

```
http://127.0.0.1/integria/include/api.php?user=user&user_pass=userpass&pass=api_pass&op=get_inventories
```

Resultado de ejemplo con formato CSV:

```
1,"Sample inventory object"
2,"Another inventory"
3,"Thirdventory"
```

Ejemplo de llamada con resultado en XML:

```
http://127.0.0.1/integria/include/api.php?user=user&user_pass=userpass&pass=api_pass&op=get_inventories&return_type=xml
```

Resultado de ejemplo en XML:

```
<xml>
  <inventory>
    <id>1</id>
    <name>Sample inventory object</name>
  </inventory>
  <inventory>
    <id>2</id>
    <name>Another inventory</name>
  </inventory>
  <inventory>
    <id>3</id>
    <name>Thirdventory</name>
  </inventory>
```

```
</xml>
```

get_user_exists_in_group

Resultado: Devuelve uno 1 si el usuario dado existe en alguno de los grupos dados. Cero 0 en caso contrario.

Parámetros serializados:

- `id_usuario` .
- `id_grupo` : Pueden ser varios grupos separados por un *token*.

Ejemplo de llamada:

```
http://127.0.0.1/integria/include/api.php?user=user&user_pass=userpass&pass=api_pass&op=user_exists_in_group&params=test,4,5
```

get_incident_inventories

Resultado: Devuelve una lista con objetos de inventario asociados a un *ticket*.

Parámetros serializados:

- Filtro de *ticket*: Identificador del *ticket* (obligatorio).

Ejemplo de llamada con resultado en formato CSV:

```
http://127.0.0.1/integria/include/api.php?user=user&user_pass=userpass&pass=api_pass&op=get_incident_inventories&params=10
```

Resultado de ejemplo con formato CSV:

```
2,1,admin,Sanchez,0,Sancho's
computer,0,0,0,0,0000-00-00,new,2013-08-19,0000-00-005,3,admin,InventarioCSV,1,d
descripcion,0,0,0,1,2017-08-02,new,2016-10-09,2017-10-09
```

Ejemplo de llamada con resultado en XML:

```
http://127.0.0.1/integria/include/api.php?user=user&user_pass=userpass&pass=api_pass&op=get_incident_inventories&params=10&return_type=xml
```

Resultado de ejemplo en XML:

```
<xml version='1.0' encoding='UTF-8'>
  <inventory>
    <id><![CDATA[2]]></id>
```

```
<id_object_type><![CDATA[1]]></id_object_type>
<owner><![CDATA[admin]]></owner>
<name><![CDATA[Sanchez]]></name>
<public><![CDATA[0]]></public>
<description><![CDATA[Sancho's computer]]></description>
<id_contract><![CDATA[0]]></id_contract>
<id_manufacturer><![CDATA[0]]></id_manufacturer>
<id_parent><![CDATA[0]]></id_parent>
<show_list><![CDATA[0]]></show_list>
<last_update><![CDATA[0000-00-00]]></last_update>
<status><![CDATA[new]]></status>
<receipt_date><![CDATA[2013-08-19]]></receipt_date>
<issue_date><![CDATA[0000-00-00]]></issue_date>
</inventory>
</xml>
```

get_last_cron_execution

Resultado: Devuelve el tiempo que ha pasado en minutos desde la última ejecución del Crontask.

Parámetros: Ninguno.

Llamada de ejemplo:

```
http://127.0.0.1/integria/include/api.php?user=user&user_pass=userpass&pass=api_
pass&op=get_last_cron_execution
```

Resultado:

```
3
```

Ejemplo de llamada con resultado en XML:

```
http://127.0.0.1/integria/include/api.php?user=user&user_pass=userpass&pass=api_
pass&op=get_last_cron_execution&return_type=xml
```

Resultado de ejemplo en XML:

```
<xml>
  <cronjob>
    <last_exec>3</last_exec>
  </cronjob>
</xml>
```

get_num_queued_emails

Resultado: Devuelve el número de *emails* que están en cola pendientes de ser enviados.

Parámetros: Ninguno.

Llamada de ejemplo:

```
http://127.0.0.1/integria/include/api.php?user=user&user_pass=userpass&pass=api_pass&op=get_num_queued_emails
```

Resultado:

```
20
```

Ejemplo de llamada con resultado en XML:

```
http://127.0.0.1/integria/include/api.php?user=user&user_pass=userpass&pass=api_pass&op=get_num_queued_emails&return_type=xml
```

Resultado de ejemplo en XML:

```
<xml>
  <pending_email>
    <num>20</num>
  </pending_email>
</xml>
```

Extensión del API

Para desarrollar nuevas llamadas para el API tiene que tener en cuenta que:

- La llamada tiene que estar inscrita como una función en el fichero <instalación Integria>/include/functions_api.php, además de estar contemplada en entre las funciones válidas en el archivo <instalación Integria>/include/api.php.
- La función debe devolver un único valor o un *array* (matriz o arreglo) de pares, por ejemplo nombre-valor.
- El formateo del resultado de la llamada a XML o CSV que se realiza en el archivo api.php debe ser de forma automática.

Ejemplos de funciones para el API

Ejemplo de función que devuelve un único valor

Definición en el archivo functions_api.php:

```
function host_risk ($host_name) {
    $host_id = get_hosts_id($host_name);
```

```
//If there is not a host with this name return false
if (!$host_id) {
    return false;
}

$risk = return_last_risk_agent($host_id);

return $risk;
}
```

Llamada en el archivo `api.php`:

```
switch ($op) {

    ...

    case "host_risk":
        if (isset($params['0'])) {
            $result = host_risk ($params['0']);
        } else {
            $result = false;
        }
        break;

    ...

}
```

Ejemplo de función que devuelve una lista de valores

Definición en el archivo `functions_api.php`:

```
function all_host_risk() {

    //Get all host
    $hosts = get_hosts();

    //Check if there is host or not
    if (empty($hosts)) {
        return false;
    }

    $result = array();
    foreach ($hosts as $host) {
        $host_name = get_host_name($host['id']);
        $risk = return_last_risk_agent($host['id']);
        $aux = array();
        $aux['name'] = $host_name;
        $aux['data'] = $risk;

        array_push($result, $aux);
    }
}
```

```
}  
  
    return $result;  
}
```

Llamada en el archivo `api.php`:

```
switch ($op) {  
  
    ...  
  
    case "all_host_risk":  
        $result = all_host_risk ();  
        break;  
  
    ...  
  
}
```

[Volver al índice de documentación de Pandora FMS](#)