FacetPhone NetTAPI

The NetTAPI interface allows a process to connect to the FacetPhone server to get information about calls, user status changes, and other telephone system activities for use in an application. It also allows applications to initiate calls and turn call recording on and off. This document describes the details of this interaction. In this document "the application" refers to the third party application that is connecting to the FacetPhone server.

1. Getting the FacetPhone User name, FacetPhone server address, and TCP port number from the PC

In order for the application to connect to the FacetPhone server, it will need an IP address, a TCP port number, and it will need to supply a FacetPhone user name. This information can be stored as part of the application's configuration data, or if the FacetPhone user interface is installed on the PC, the application can get this information from the user interface configuration file that can be found at:

\Program Files\FacetCorp\FacetPhone\ui\local.cfg

This file's contents look like this:

```
<server_url=http://facetphone.sssi.com/facetphone/>
<server_port=6500>
<udp_port_1=6503>
<udp_port_2=6501>
<udp_port_3=6502>
<ka_interval=2>
<username=clark>
<dflt_loc=office>
<log_number=2>
<automatic_login=y>
<password=1234>
```

The application should find the "username" line. That will contain the FacetPhone user name for the last person who logged in to the FacetPhone User Interface on this PC. The "server_url" and "server_port" parameters tell the DNS name of the FacetPhone server and the TCP port number to use to connect. The application should use these values to make a TCP connection to the FacetPhone server.

FacetPhone NetTAPI

2. Initial connection

Using the server_url and server_port, the application should open a TCP socket and connect to the FacetPhone server. It should send a message with the following contents:

!<connect_nettapi>^D

All messages to or from the server start with a single byte message type (!, ?, or #) and are terminated with a single byte terminator (^D, /004, 0x04). Each unit of the message is enclosed in angle brackets ("<" and ">") and consists of a keyword optionally followed by an equals sign and a value. The first unit in the message is the command verb.

In all the following messages, the ^D will be omitted, but it should be understood that every NetTAPI message ends with ^D.

Once the server receives this initial connect message, it will respond with a challenge string like this:

#<challenge=936bedf7d79995b7e29ec73ad214dc8>

In the future, the challenge value may be used to construct an encrypted password. At the current time, passwords are not implemented for NetTAPI. The application sends the login message with a blank pin value for now:

!<login_nettapi><uname=ken><pass=>

If the login is successful, the server will respond:

#<login_nettapi=y><uid=128><priv=oper,mgr,admin>

If the login is unsuccessful the server will respond:

#<login_nettapi=n>

3. Notification request and response

Next, the application will send commands describing what activities it is interested in receiving information about.

FacetPhone NetTAPI

This command says that the application wants to be notified of incoming calls from outside callers (not from an internal extension):

!<interest><incoming_outside_calls><uid=128>

The <uid=> parameter is optional and if it is left off, the notifications will be for calls to the user specified on the NetTAPI login.

This command would cause notification for outgoing calls made by the user to outside parties:

!<interest><outgoing_outside_calls>

Currently NetTAPI only supports outside calls. Call types can be combined in one command so that the following command would ask for notifications on incoming and outgoing calls for the specified user:

!<interest><incoming_outside_calls><outgoing_outside_calls><uid=128>

If the request is successful, the server will respond:

#<interest><ok>

If the request is unsuccessful, the server will respond:

#<interest><err=This message would say what went wrong>

4. Notification delivery

When the user receives a call type that the application has asked for notification of, the server will send the information about the call to the application:

#<uid=107><call=3><incoming_outside_call><caller_name=FACETCORP><caller_number=9729859901><serial=1>

The "uid" parameter specifies the user related to the interest request.

The "call" parameter indicates that this is notification of a call. The "=3" says that the call is in slot three. Once this call is finished, it can be replaced with another call in the same slot, so this call id is not unique to this call. The "serial" parameter described below is unique to the call.

FacetPhone NetTAPI

The next parameter tells the type of call using the same values that were used in the interest command.

The "caller_name" and "caller_number" are caller ID information from the phone company.

If the call is an outgoing call, the caller_name and caller_number will not be present in the notification, and the called_number will be included instead:

#<uid=107><call=3><outgoing_outside_call><called_number=9729859901><serial=1>

When this call is over (or when the party of interest leaves the call), the server will send:

#<uid=107><call=3><serial=1><end>

5. Dialing

When the application wants to place a call, it sends a dial command:

!<dial><uid=107><addr=9,2148441111>

The uid is the person who is placing the call. The addr is the outside number to dial including the leading 9 for the default trunk group or 8n to select a trunk group. The comma to separate the trunk group specifier from the outside number is optional.

If the command was processed, the response will be:

#<dial><ok>

If it failed for some reason, it will be:

#<dial><err=message about what was wrong>

6. Call recording

To turn on recording for a user's currently active call, the application can send the request:

!<record><uid=107>

FacetPhone NetTAPI

To turn off recording for a user's currently active call, send the request:

!<stop_record><uid=107>

If there is no currently active call for the specified user, then the recording requests will be ignored.

7. Changing a user's status

To change a user's current status, the application can send the request:

!<change_status><uid=nnn><status=xxx>

where nnn is the user's extension number and xxx is a status that the user would find in his drop list for choosing his current status. These can be system wide status definitions or status definitions for that user.

8. Requesting information about a call

The application can ask for information about a call after receiving the end message:

?<call_info=1>

Where the number is the serial number of the call. If the call is not completed for the user being watched, then the server will respond with:

#<call_info=1><not_completed>

If the call is completed, the following information will be returned:

#<call_info=1><call_type=O><start_time=2006/02/27@15:04:03><duration=0:00
:58><ring_time=0:00><hold_time=0:00><job_code=><note=>
#<call_info=1><party=1><voice_type=S><device_index=144><party_number=14
4><party_name=Jim Bryant><duration=0:00:58><ring_time=0:00>
#<call_info=1><party=2><voice_type=L><device_index=1><party_number=9728
441111><party_name=Unknown><duration=0:00:58><ring_time=0:00>
#<call_info=1><recording_file=/u/fpdev/test/ivr/jim/vmail/new/44036959_3a2c.wa
v>
#<call_info=1><recording_file=/u/fpdev/test/ivr/jim/vmail/new/44036977_3a3f.wa
v>
#<call_info=1><end>

FacetPhone NetTAPI

The first message of the response will have the statistics for the call in general.

The call_type is 'I' for an incoming outside call and 'O' for an outgoing outside call.

The start_time is the date and time the call started.

The duration is the duration of the call to the point that the person whose calls are being watched left the call.

The ring_time is the amount of time the call was ringing before being answered.

The hold_time is the amount of time the call was on hold or parked.

The job_code and note fields are data that the user can enter with the FacetPhone user interface while the call is in progress.

Then there will be one additional message for each party that has been in the call, not counting IVR parties like auto-attendants.

The voice_type will be 'L' for an outside line and 'S' for an internal station.

The device_index will be the line number or station number.

Note that if the call continues, but the user of interest is no longer a party to it, then the call is considered completed for that user and the statistics refer to the period of time up until the user left the call.

Next, there will be one message for each recording file associated with the call. There can be multiple recording files if recording was turned on and off multiple times during the call, creating a different file for each period that was recording.

Finally, an end message will be sent to let the application know that there are no more call_info messages about the call.

9. Disconnecting from NetTAPI

When the application is ready to end its NetTAPI connection, it can send the command:

!<logout_nettapi>

Or, the application may simply close the TCP connection to the FacetPhone server.