

# TapiServer – Dokumentacja

## 1. Wstęp

Aplikacja TapiServer służy do bezpośredniej interakcji z centralą telefoniczną, posiadającą wsparcie dla Microsoft Telephony Application Programming Interface (TAPI) w standardzie 2.0. Umożliwia inicjowanie połączeń wychodzących z urządzeń podłączonych do centrali pod wskazany numer oraz rejestrowanie połączeń przychodzących na te urządzenia wraz z pozyskaniem numeru osoby dzwoniącej.

## 2. Interfejs

Aplikacja posiada charakter aplikacji serwerowej, z którą należy komunikować się przy użyciu gniazd sieciowych TCP (ang. sockets). Po uruchomieniu TapiServer nasłuchuje na porcie 44444, na adresie IPv4 komputera na którym jest uruchomiony. Po nawiązaniu połączenia, TapiServer odbiera i odsyła informację do klienta. Aplikacja jest przeznaczona do komunikacji z jednym klientem.

### Zapytania do aplikacji TapiServer:

1) Rejestracja numeru (równoważna z nasłuchiowaniem na połączenia przychodzące na ten numer, umożliwia wykonywanie połączeń wychodzących).

```
<czterobajtowa liczba określająca rozmiar zapytania>  
LISTEN,<rejestrowany_numer>
```

2) Wykonanie połączenia wychodzącego (możliwe po wcześniejszej rejestracji numeru z którego chcemy wykonać połączenie).

```
<czterobajtowa liczba określająca rozmiar zapytania>  
CALL,<numer_źródłowy>,<numer_docelowy>
```

3) Wylogowanie numeru (równoważna z zakończeniem nasłuchiwania na połączenia przychodzące pod ten numer, oraz dezaktywacją funkcji dzwonienia).

```
<czterobajtowa liczba określająca rozmiar zapytania>  
STOPLISTEN,<zalogowany_numer>
```

### Wiadomości otrzymywane od TapiServer:

1) Po otrzymaniu połączenia przychodzącego na zalogowany numer, aplikacja wysyła do podłączonego klienta wiadomość

```
<czterobajtowa liczba określająca rozmiar zapytania>  
NEWCALL,<numer_połączenia_przychodzącego>,<zalogowany_numer>
```

Wszystkie dane są przesyłane w formacie little-endian.

### 3. Kod źródłowy

Niskopoziomowe operacje, wykorzystujące funkcję TAPI, zostały opakowane w 5 podstawowych funkcji, pozwalających zrealizować powyżej opisaną funkcjonalność, nagłówki tych funkcji znajdują się w pliku click2call.h.

```
void fnInitializeTapi(HWND hParentWnd, HINSTANCE hInst)
```

#### Opis

Funkcja inicjalizuje podstawowe zasoby centralki telefonicznej, wołana jest raz, tuż po uruchomieniu aplikacji.

#### Parametry

HWND hParentWnd – uchwyt do okna WinApi

HINSTANCE hInst – uchwyt do aplikacji WinApi

```
void fnClickToKnow(LPCSTR szListenerNumber)
```

#### Opis

Funkcja rejestruje wskazany numer do nasłuchiwania połączeń przychodzących oraz umożliwiającą wykonanie połączeń wychodzących przy użyciu funkcji fnClickToCall(...).

Ponowne wywołanie tej funkcji dla tego samego numeru powinno odbyć się dopiero po wywołaniu funkcji fnStopClickToKnow(...) dla tegoż numeru.

#### Parametry

LPCSTR szListenerNumber – rejestrowany numer.

```
void fnClickToCall(LPCSTR szCallerNumber, LPCSTR szCalledNumber, LPCSTR szId)
```

#### Opis

Funkcja służy do wykonania połączenia pod wskazany numer w argumencie szCalledNumber, z numeru wskazanego w szCallerNumber. Można ją wołać dla wcześniej zalogowanego przy użyciu fnClickToKnow, numeru.

#### Parametry

LPCSTR szCallerNumber – numer dzwoniącego

LPCSTR szCalledNumber – numer docelowy

LPCSTR szId – parametr obecnie nie wykorzystywany

```
void fnStopClickToKnow(LPCSTR szListenerNumber)
```

#### Opis

Funkcja służąca do wylogowania numeru w celu zwolnienia zasobów sprzętowych. Należy ją wykonać gdy nie chcemy już dalej wykonywać operacji związanych z danym numerem. Zawołana dla numeru niezalogowanego, funkcja nic nie wykonuje.

#### Parametry

**LPCSTR** szListenerNumber – numer który chcemy wyrejestrować

```
void fnShutdownTapi();
```

#### Opis

Funkcja służąca do wyrejestrowania aplikacji i zwolnienia wszystkich zasobów centralki. Należy ją wołać tuż przed zamknięciem programu.