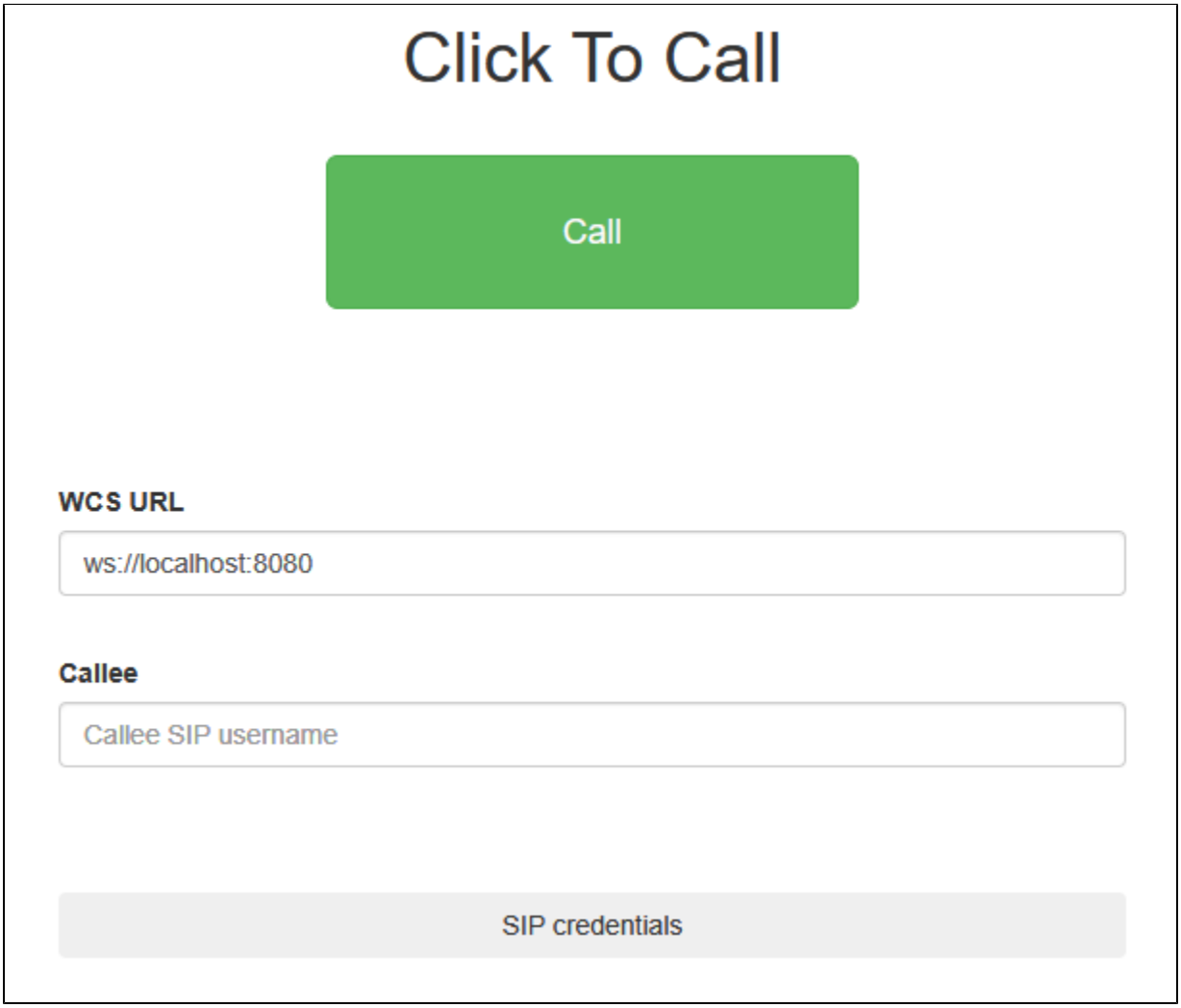
Click to Call

- Пример для совершения SIP аудиозвонка в один клик
- Код примера
- Работа с кодом примера

Пример для совершения SIP аудиозвонка в один клик

Данный пример позволяет сделать SIP аудиозвонок одним кликом кнопки, используя заданный SIP аккаунт



Код примера

Код данного примера находится на WCS-сервере по следующему пути:

/usr/local/FlashphonerWebCallServer/client2/examples/demo/sip/click-to-call

click-to-call.css - таблица стилей для страницы примера
click-to-call.html - страница примера
click-to-call.js - скрипт, обеспечивающий работу примера

Тестировать данный пример можно по следующему адресу:

https://host:8888/client2/examples/demo/sip/click-to-call/click-to-call.html

Здесь host - адрес WCS-сервера.

Работа с кодом примера

Для разбора кода возьмем версию файла click-to-call.js с хешем 02c8028, которая находится здесь и доступна для скачивания в соответствующей сборке 2.0.178.

1. Инициализация API.

Flashphoner.init() code

```
try {
    Flashphoner.init();
} catch(e) {
    $("#notifyFlash").text("Your browser doesn't support WebRTC technology needed for this example");
    return;
}
```

2. Подключение к серверу.

createSession() code

Методу createSession() передается объект с параметрами для подключения

- urlServer - URL для WebSocket-соединения с WCS-сервером
- sipOptions - параметры SIP соединения, взятые из полей скрытой формы

```
    var url = $('#urlServer').val();
    var sipOptions = {
        login: $('#sipLogin').val(),
        authenticationName: $('#sipAuthenticationName').val(),
        password: $('#sipPassword').val(),
        domain: $('#sipDomain').val(),
        outboundProxy: $('#sipOutboundProxy').val(),
        port: $('#sipPort').val(),
        registerRequired: true
    };

    var connectionOptions = {
        urlServer: url,
        sipOptions: sipOptions
    };

    //create session
    console.log("Create new session with url " + url);
    Flashphoner.createSession(connectionOptions).on(SESSION_STATUS.ESTABLISHED, function(session){
        ...
    });
```

3. Получение от сервера события, подтверждающего успешное соединение.

ConnectionStatusEvent ESTABLISHEDcode

При обработке данного события создается исходящий звонок

```
    Flashphoner.createSession(connectionOptions).on(SESSION_STATUS.ESTABLISHED, function(session){
                setStatus("Session", SESSION_STATUS.ESTABLISHED);
                    //session connected, place call
                call(session);
        }).on(SESSION_STATUS.DISCONNECTED, function(){
        ...
        }).on(SESSION_STATUS.FAILED, function(){
        ...
            });
```

4. Исходящий звонок.

session.createCall(), call() code

При создании в метод передаются следующие параметры

- callee - SIP имя пользователя вызываемого
- visibleName - отображаемое имя
- localVideoDisplay - div-элемент локального дисплея (в случае Flash медиа-провайдера будет использован для диалога настроек Flash-плеера)
- remoteVideoDisplay - div-элемент для получаемого аудио
- constraints - ограничения для звонка (в данном случае параметр 'video' установлен в 'false' для аудиозвонка)
- receiveAudio - установлен в 'true' для получения аудио
- receiveVideo - установлен в 'false', чтобы не получать видео

```
    var constraints = {
        audio: true,
        video: false
    };

        //prepare outgoing call
    var outCall = session.createCall({
                callee: $("#callee").val(),
                visibleName: "Click To Call",
                localVideoDisplay: localDisplay,
                remoteVideoDisplay: remoteDisplay,
                constraints: constraints,
                receiveAudio: true,
        receiveVideo: false
        ...
        });

        outCall.call();
```

5. Завершение звонка.

call.hangup() code

```
    $("#callBtn").text("Hangup").removeClass("btn-success").addClass("btn-danger").off('click').click(function()
{
        $(this).prop('disabled', true);
            outCall.hangup();
    }).prop('disabled', false);
```