

# Публикация потока из приложения Android в фоновом режиме

## Описание

Чтобы приложение Android не выгружалось из памяти устройства, и публикация видео не останавливалась при сворачивании приложения, необходимо устанавливать соединение с WCS сервером и публиковать видео из сервиса, который должен быть запущен из Activity приложения. В свою очередь, чтобы и сервис не был выгружен из памяти, необходимо создать уведомление, которое должно находиться в панели уведомлений, пока сервис работает. Рассмотрим пример модификации исходного кода приложения [Android Two Way Streaming](#)

## Пример модификации исходного кода приложения

1. Создание сервиса при нажатии кнопки **Publish** и успешном получении доступа к камере и микрофону

```
@Override
public void onRequestPermissionsResult(int requestCode,
                                     @NonNull String permissions[],
                                     @NonNull int[] grantResults) {
    switch (requestCode) {
        case PUBLISH_REQUEST_CODE: {
            if (grantResults.length == 0 ||
                grantResults[0] != PackageManager.PERMISSION_GRANTED ||
                grantResults[1] != PackageManager.PERMISSION_GRANTED) {
                Log.i(TAG, "Permission has been denied by user");
            } else {
                mPublishButton.setEnabled(false);
                ...
                Intent intent = new Intent(StreamingMinActivity.this,
                TestService.class);
                intent.putExtra("url", mWcsUrlView.getText().toString());
                intent.putExtra("streamName",
                mPublishStreamView.getText().toString());
                startService(intent);

                Log.i(TAG, "Permission has been granted by user");
            }
        }
    }
}
```

## 2. Создание сессии и публикация потока при старте сервиса

```
@Override
public int onStartCommand(Intent intent, int flags, int startId) {
    SessionOptions sessionOptions = new
    SessionOptions(intent.getStringExtra("url"));
    Session session = Flashphoner.createSession(sessionOptions);
    session.connect(new Connection());
    StreamOptions streamOptions = new
    StreamOptions(intent.getStringExtra("streamName"));
    Stream publishStream = session.createStream(streamOptions);
    publishStream.publish();
    Toast.makeText(this, "Start service", Toast.LENGTH_SHORT).show();
    return START_STICKY;
}
```

## 3. Создание уведомления

```
private void showNotification() {
    Intent notificationIntent = new Intent(this,
    StreamingMinActivity.class);
    notificationIntent.addFlags(Intent.FLAG_ACTIVITY_CLEAR_TOP);
    PendingIntent pendingIntent = PendingIntent.getActivity(this, 0,
    notificationIntent, 0);
    int iconId = R.mipmap.ic_launcher;
    int uniqueCode = new Random().nextInt(Integer.MAX_VALUE);
    Notification notification = new NotificationCompat.Builder(this)
        .setSmallIcon(iconId)
        .setContentText("Started stream")
        .setContentIntent(pendingIntent).build();
    startForeground(uniqueCode, notification);
}
```

## 4. Остановка сервиса при нажатии кнопки **Unpublish**

```
mPublishButton.setOnClickListener(new OnClickListener() {
    @Override
    public void onClick(View view) {
        if (mPublishButton.getTag() == null ||
        Integer.valueOf(R.string.action_publish).equals(mPublishButton.getTag()))
        {
            ...
        } else {
            mPublishButton.setEnabled(false);
            ...
            stopService(new Intent(StreamingMinActivity.this,
            TestService.class));
            publishStream = null;
        }
        ...
    }
});
```

## 5. Остановка публикации при остановке сервиса


```
@Override
public void onDestroy() {
    super.onDestroy();
    publishStream.stop();
    Toast.makeText(this, "Stop service",
        Toast.LENGTH_SHORT).show();
    stopForeground(true);
}
```

Полный код примера модификации файла `StreamingMinActivity.java`

 [StreamingMinActivity.java](#)



Полный код примера файла реализации сервиса `TestService.java`

 [TestService.java](#)



Полный код примера модификации манифеста приложения

 [AndroidManifest.xml](#)



## Известные ограничения

1. При публикации потока из сервиса невозможно отображение локального видео в приложении
2. В приведенном примере, при закрытии приложения из списка запущенных приложений, сервис также остановится.