

Using UDP or TCP transport for WebRTC

Since build 2.6.21 it is possible to use UDP or TCP transport to publish or play WebRTC with option `FPWCSEApi2StreamOptions.transport`. The option is set while Stream object creation and accepts the following values:

```
typedef NS_ENUM(NSUInteger, kFPWCSTransport) {
    kFPWCSTransportDefault,
    kFPWCSTransportUDP,
    kFPWCSTransportTCP
};
```

Where

- `kFPWCSTransportUDP` enables UDP transport
- `kFPWCSTransportTCP` enables TCP transport

If the option is not set, the transport defined in server settings is used.

The usage example in application based on Objective C framework

[code](#)

```
FPWCSEApi2StreamOptions *options = [[FPWCSEApi2StreamOptions alloc] init];
options.name = [self getStreamName];
...
options.transport = _useTCPTransport.control.on ? kFPWCSTransportTCP : kFPWCSTransportUDP;
NSError *error;
_localStream = [_session createStream:options error:&error];
...

```

The usage example in application based on Swift framework

[code](#)

```
let options = FPWCSEApi2StreamOptions()
options.name = publishName.text
...
options.transport = tcpTransport.isOn ? kFPWCSTransport.fpwcsTransportTCP : kFPWCSTransport.fpwcsTransportUDP;
do {
    try publishStream = session!.createStream(options)
} catch {
    print(error);
}
```