



NetPlay Cloud API

Revision 1.0
8/11/17

Overview

NetPlay Cloud is a Google Firebase cloud service and includes the following features

- Firebase authentication via email/password or Google login
- Firebase Real Time Database for device discovery and control
- Cloud control for your NetPlay Home or Pro system
- Cloud control for your standalone IrUSB products

Video Storm provides drivers for many control systems and/or applications. This document is intended for integrators who wish to access the cloud API directly or write their own drivers/applications.

Firestore

Firestore can be integrated using several methods. Full documentation can be found on Google help pages: <https://firebase.google.com/docs/>

For Linux, we recommend using the pyrebase python library (which uses Firestore REST API). This is included in our driver set for IrUSB and can also be used for NetPlay Home/Pro.

For Android, iOS, or Web based applications please refer to the Google documents for a simple way to include Firestore support in your application.

NetPlay Cloud connection configuration:

```
apiKey: "AIzaSyBzS58fmZasjyR05FHqyLAIbXeh4SGHovY",  
authDomain: "netplay-429de.firebaseio.com",  
databaseURL: "https://netplay-429de.firebaseio.com",  
projectId: "netplay-429de",  
storageBucket: "netplay-429de.appspot.com",  
messagingSenderId: "504916577319"
```

Authentication

You must first create a NetPlay Cloud account either using NetPlay Manager web interface OR the IrUSB Android app. Accounts can be created using an email & password or directly linked via your Google sign in.

We recommend using email & password for maximum compatibility across different platforms.

External applications or drivers must first “authenticate” to connect to your NetPlay Cloud account. Authentication is handled as a method call in the library you are using. Please insure you use the same method to authenticate (email or google) as you used to create the account.

We recommend that you use ONE NetPlay Cloud account PER INSTALLATION/HOME.

Real Time Database

The Firebase Real Time Database is used to facilitate all communication between external drivers/applications and your NetPlay Manager and/or IrUSB devices.

Once authenticated, you can use the library methods to read or write to NetPlay Cloud devices. Please see the Google docs for examples based on your library.

Firebase RTDB can also provide streaming sockets for immediate notification of events on cloud connected devices. Use of the streaming sockets is not required to control NetPlay devices, but can be beneficial if you need very low latency status readback or event notifications. Google docs also describe how to implement this.

The NetPlay Cloud database JSON structure is as follows. Please note that you may only read/write nodes under the UID (user id) that you have authenticated with.

JSON node tree:

users:

 \$UID:

 devices: (control interface for all devices)

 \$UUID: (below is for VRX020/NetPlay Manager)

 DT:

 connected: Boolean

 last_connection: date/time

 (if IrUSB, also these keys below)

 devids:

 \$DEVID:

 name: String

 txcodes: Strings (space delim list)

 rxcodes: Strings (space delim list)

 TX:

 commandToSend: String (NetPlay protocol, /r term)

 SendCnt: Int (inc on each new commandToSend)

 RX:

 commandToRecv: String (NetPlay protocol, /r term)

 RecvCnt: Int (inc on each new commandToRecv)

remotes: (saved remote data from devices)

 \$NAME:

 \$CODE:

 source: boolean

 number: int

 hexdata: String (hex code)

services: (service and driver related variables)

 \$NAME:

Service specific key/value pairs

Driver steps

To use NetPlay Cloud API, use the following steps:

1. Authenticate
2. Connect to RTDB using your token and our config
3. Find the device you will control from RTDB node `/users/$UID/devices` (`$UID` is your `userId` returned by Authentication). NetPlay manager devices will have `VRX020` in the `uuid`. IrUSB devices will have `IRUSB` in the `uuid`.
4. Verify that node `/users/$UID/devices/$UUUID/DT/connected = true`. This indicates the device is online.
5. Write your NetPlay protocol (or IrUSB protocol) command to node `/users/$UID/devices/$UUUID/TX/ commandToSend`. Make sure you include the `\r` to terminate the command.
6. Increment node `/users/$UID/devices/$UUUID/TX/ SendCnt`
7. If needed, you can read the device response from node `/users/$UID/devices/$UUUID/RX/ commandToRecv`

If you are using a streaming socket connection to node `/users/$UID/devices/$UUUID/RX` you will automatically receive any changes on `commandToRecv`.

Otherwise, you can read the node directly for device feedback. The node `/users/$UID/devices/$UUUID/RX/ RecvCnt` will increment each time `commandToRecv` is updated. You may also just check `commandToRecv` for your command echo (via NetPlay Protocol) for verification.

Please see our IrUSB driver package for an example of NetPlay Cloud implementation.