

SonosPlayer

- [Install Sonos Node Library](#)
- [Scripts](#)
 - [Discover.js](#)
 - [PlayRadioParadise.js](#)
- [Add Start Script](#)
- [References](#)

Install Sonos Node Library

```
mkdir sonos
cd sonos

npm init

npm install sonos

npm install git://github.com/bencevans/node-sonos.git
```

Scripts

Discover.js

> vi discover.js

```
const Sonos = require('sonos')

console.log('Searching for Sonos devices for 5 seconds...')

const discovery = new Sonos.AsyncDeviceDiscovery()

discovery.discover().then((device, model) => {
  console.log('Found one sonos device %s getting all groups', device.host)
  return device.getAllGroups().then((groups) => {
    console.log('Groups %s', JSON.stringify(groups, null, 2))
  })
}).catch(e => {
  console.warn(' Error in discovery %j', e)
})
```

Run it

> node discover.js

PlayRadioParadise.js

vi playRadioParadise.js

```
const Sonos = require('sonos').Sonos
const sonos = new Sonos(process.env.SONOS_HOST || '192.168.1.26')

// This example demonstrates playing radio stations
// the Sonos built-in support for tunein radio.

// CAUTION: ID is with leading s since version 1.12.1
const stationId = 's13606'
const stationTitle = 'Radio Paradise'

sonos.playTuneinRadio(stationId, stationTitle).then(success => {
  console.log('Yeay')
}).catch(err => { console.log('Error occurred %j', err) })
```

Run it

```
> node playRadioParadise.sh
```

Run It using environment variables

```
> SONOS_HOST=192.168.1.21 node playRadioParadise.sh
```

Add Start Script

If you want to add your program to npm, edit the package.json file

```
{
  "name": "sonosplayer",
  "version": "1.0.0",
  "description": "",
  "main": "index.js",
  "scripts": {
    "start": "node discover.js",
    "test": "echo \"Error: no test specified\" && exit 1"
  },
  "author": "",
  "license": "ISC",
  "dependencies": {
    "sonos": "git://github.com/bencevans/node-sonos.git"
  }
}
```

Run it

```
> npm start
```

References

Reference	URL
node-sonos	https://www.npmjs.com/package/sonos