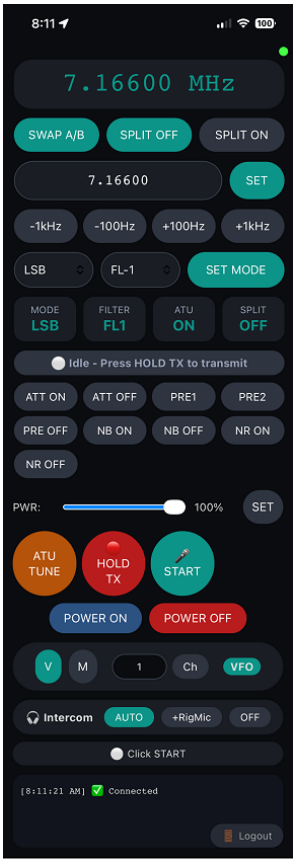
📡 IC-7300 Remote Server Setup Guide

This guide explains how to set up a **local IC-7300 remote server** and then expose it securely over the internet using **Cloudflare Tunnel**.

🌐 Web application

Desktop	Mobile
	

🧩 Part 1 — Local Setup (PC ↔ IC-7300)

1. Connect and Identify COM Port

- Plug your **Icom IC-7300** into your PC via USB.
- Open **Device Manager**.
- Plug/unplug the radio and identify the COM port (e.g. `COM4`).

---

2. Configure Radio Settings

On the IC-7300:

`[MENU] → [SET] → [Function] → [Time-Out Timer (CI-V)]`

- Set to **3 or 5 minutes**

---

3. Load Preset Configuration (Recommended)

For best and fastest results:

1. Copy preset file:

   `IC-7300-Settings-XX0YYY.dat`

   to:

   `SD Card → /Setting`

2. On the radio:

   `[MENU] → [SET] → [SD Card] → [Load Setting]`

   - Select file
   - Confirm:
     - Choose **[ALL]** (or specific settings)
     - Select **[YES]**

3. Update callsign:

   `[MENU] → [SET] → [Display] → [My Call]`

   - Replace `XX0YYY` with your callsign

---

4. Prepare Server Folder

On your PC:

`C:\myserver`

- Copy all files from:

  `\bin-out`

---

5. Detect Audio Devices

Run:

`sound_devices.exe`

Note device IDs:

- **Input device** (radio → PC)
  Example:

  `Microphone (USB Audio CODEC) → ID 2`

- **Output device** (PC → radio)
  Example:

  `Speakers (USB Audio CODEC) → ID 5`

---

6. Configure `config.json`

Edit:

```
{
    "serial_port": "COM4",
    "baud_rate": 115200,
    "http_port": 8080,
    "ws_port": 8081,
    "sample_rate": 8000,
    "frame": 1024,
    "min_buffer_frames": 3,
    "jitter_buffer_max": 4,
    "buffer_timeout": 2.0,
    "rx_gain": 1.0,
    "tx_gain": 1.0,
    "tx_timeout_seconds": 180,
    "tx_hang_time_ms": 550,
    "radio_audio_input_device": 2,
    "radio_audio_output_device": 5,
```

```
    "monitor_level": 64,
    "password": "radio2",
    "cookie_name": "radio_auth",
    "cookie_max_age": 2592000,
    "secret": "dolkiuytsgty7643hy5",
    "domain": "radio.yourdomain.com",
    "page_title": "IC-7300 Remote",
    "polling_interval_ms": 30000,
    "subpath": "",
    "default_memory_channel": 2,
    "license_key": "2dc57363851b329aa46150eb76f00d0c"
}
```

Notes

- Change:
  - `serial_port`
  - `radio_audio_*`
  - `password` ⚠️ **important**
  - `secret` (random string)
  - `license_key` ⚠️ **important** To get your key, look at the **logs/ic7300.log** for lines:

```
...
LICENSE ERROR: Invalid or missing license key!
TX period is limited to 5 seconds at once!
Email this ID to the software vendor to get your license key
============================================================
Your ID: ###############
============================================================
```

Send an e-mail with your ID#.
Replace the "license_key" to remove the 5 seconds TX limit.
⚠️ Your ID# / key is for one computer and user name, if you are running as service, the username would be different!

- Optional tuning:
  - `min_buffer_frames = 4..5` → smoother audio, more delay

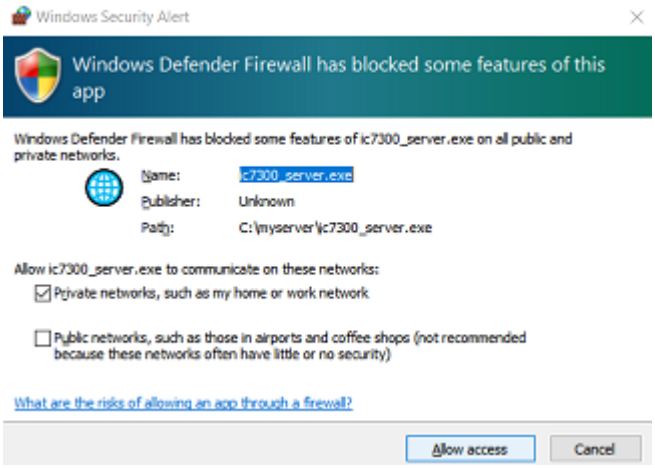
---

7. Start Server

Run:

```
ic7300_server.exe
```

- Allow access in **Windows Defender Firewall**



---

8. Test Locally

Open browser:

```
http://localhost:8080
```

---

9. Optional: Batch Start

Use:

```
test_server.bat
```

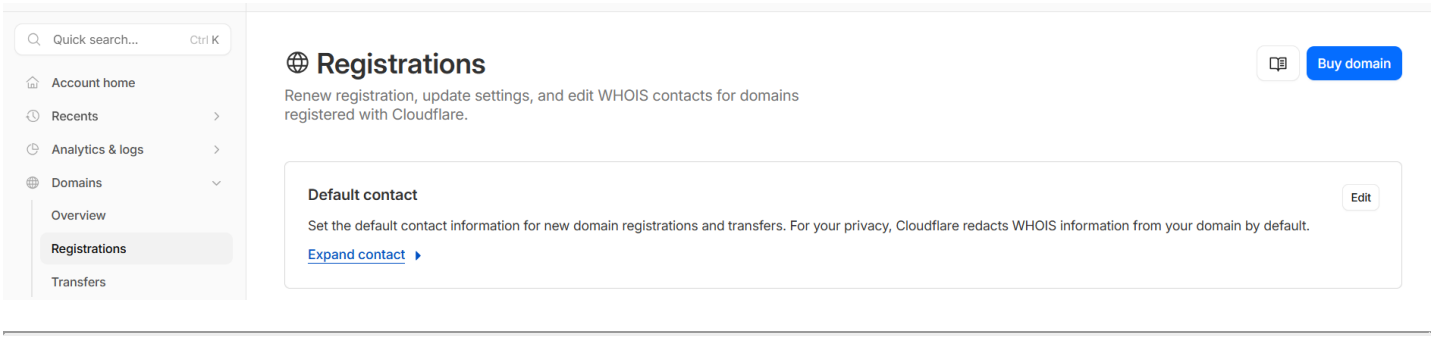
🌐 Part 2 — Remote Access via Cloudflare Tunnel

---

1. Create Cloudflare Account

- Go to:
  https://dash.cloudflare.com/sign-up
- Purchase a domain (~$US15 / ~10€ per year):

```
Domains → Registrations → Buy Domain
```



---

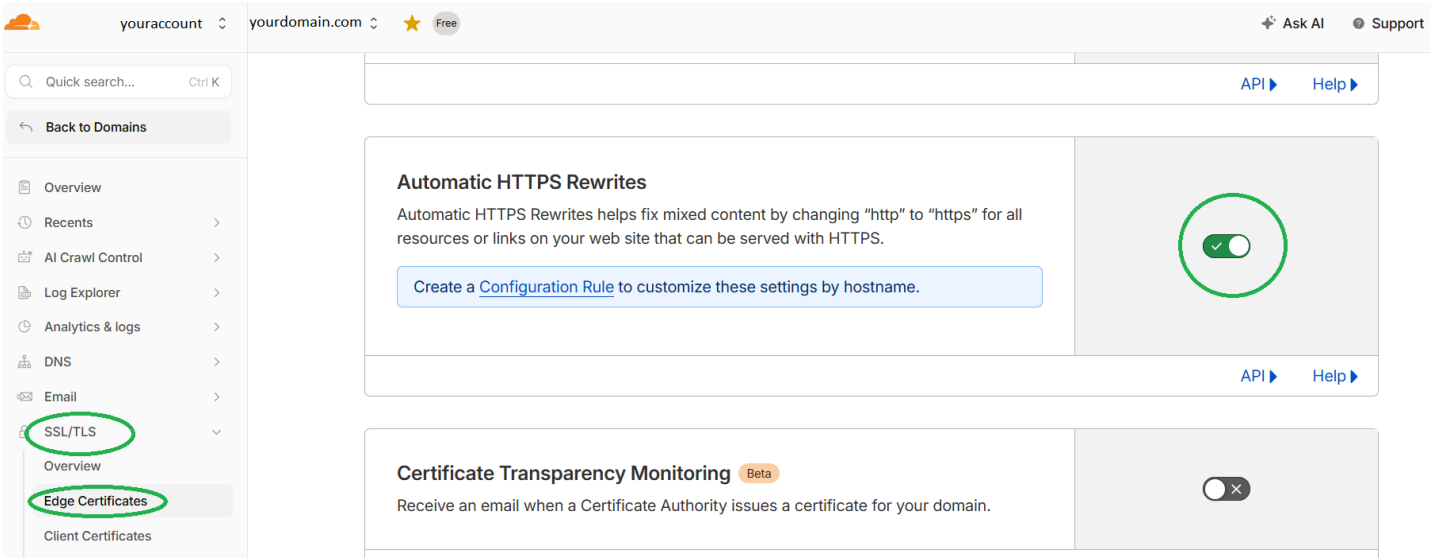
2. Force HTTPS

Navigate:

```
Domains → (your domain) → SSL/TLS → Edge Certificates
```

Enable:

```
Automatic HTTPS Rewrites → ON
```



3. Download Cloudflare Tunnel

From:
https://github.com/cloudflare/cloudflared/releases/

Download:

```
cloudflared-windows-amd64.exe
```

Rename and place:

```
C:\myserver\cloudflared.exe
```

4. Create `config.yml`

Create file:

`C:\myserver\config.yml`

```
tunnel: my-mydomain-tunnel
credentials-file: C:\myserver\mydomain.json

ingress:
  # WebSocket (must be first)
  - hostname: radio.mydomain.com
    path: /ws
    service: http://localhost:8081
    originRequest:
      noTLSVerify: true
      websocket: true

  # HTTP
  - hostname: radio.mydomain.com
    service: http://localhost:8080
    originRequest:
      noTLSVerify: true

  - service: http_status:404

log-directory: C:\myserver\logs
loglevel: debug
```

---

5. Authenticate Tunnel

Run:

```
cloudflared tunnel login
```

- Browser opens
- `cert.pem` created in: `%USERPROFILE%\.cloudflared` or `%HOME%\.cloudflared`

---

6. Create Tunnel

```
cloudflared tunnel create my-mydomain-tunnel
```

- Generates:
  - Tunnel ID
  - JSON credentials file

---

7. Prepare Files

Copy into:

```
C:\myserver
```

- `cert.pem`
- tunnel JSON → rename to:

  `mydomain.json`

---

8. Create DNS Route

```
cloudflared tunnel route dns my-mydomain-tunnel radio.mydomain.com
```

- Creates CNAME → tunnel

---

9. Run Tunnel

```
cloudflared tunnel --config config.yml run
```

---

10. Test Remote Access

Open:

```
https://radio.mydomain.com
```

- Autometed way, instead of steps 9. and 10. run: `test_server.bat radio.mydomain.com`

⚙️ Part 3 — Run as Windows Service

---

1. Install Service

```
cloudflared service install
```

---

2. Fix Service Path

Open:

```
regedit.exe
```

Navigate:

```
HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Services\Cloudflared
```

Edit:

```
ImagePath
```

Set:

```
C:\myserver\cloudflared.exe --config=C:\myserver\config.yml tunnel run
```

⚠️ Ensure:

- No extra spaces
- Correct paths

---

3. Control Service

```
net stop cloudflared
net start cloudflared
```

---

4. Test via Batch

```
test_server.bat radio.mydomain.com
```

ℹ Troubleshoot

Cleanup Tunnel Setup (if something goes wrong)

```
cloudflared tunnel list
cloudflared tunnel delete <NAME or UUID>
cloudflared tunnel list
```

- Then you can repeat the Tunnel setup (Part 2, Step 3)

Older Windows OS

The Server is designed to work on Win 10-11
Win 7, even older OS may work, you need follwing **KB3063858** update:

[Windows Vista SP2 x86]
[Windows Vista SP2 x64]
[Windows Server 2008 SP2 x86]
[Windows Server 2008 SP2 x64]
[Windows 7 SP1 x86]
[Windows 7 SP1 x64]
[Windows Server 2008 R2 SP1 x64]

Logs

Check:

```
C:\myserver\logs
```

✅ Summary

Local Setup

- Configure radio
- Configure audio + COM
- Run server

- Test on localhost

Remote Setup

- Cloudflare account + domain
- Tunnel + config
- DNS route
- HTTPS access

📃 End User License Agreement

By installing or using the IC-7300 Remote Web Server, you agree to the following:

1.Grant of License:
VE2OPN grants you a non-exclusive, non-transferable license for personal use.

2.Restrictions:
You may not decompile, reverse-engineer, or attempt to extract the source code of this application.
You may not redistribute the software without written permission.

3.Warranty:
The software is provided "As-Is" without warranty of any kind.

The developer is not liable for any damages arising from its use.

🎖️ Credits

- THIRD-PARTY OPEN SOURCE NOTICES:

This software bundles the following third-party components under their
respective licenses.

---

1. PYTHON INTERPRETER AND STANDARD LIBRARIES

Copyright (c) 2001-2026 Python Software Foundation.
This software includes the Python interpreter, which is distributed under the
Python Software Foundation License Version 2. (https://docs.python.org/3/license.html)

---

2. PERMISSIVE OPEN SOURCE LIBRARIES (MIT, BSD, APACHE 2.0)

The following libraries are included under permissive licenses (MIT, BSD, or
Apache 2.0). These licenses generally allow for commercial use provided that
the original copyright notices and disclaimers are preserved.

Libraries: altgraph,cffi,importlib_metadata,numpy,packaging,pefile,pip,pycparser,
pyinstaller,pyinstaller-hooks-contrib,pyserial,pywin32,pywin32-ctypes,setuptools,
sounddevice,websockets,wheel,WMI,zipp

---

3. PYINSTALLER BOOTLOADER

Copyright © 2010-2026, PyInstaller Development Team.
PyInstaller is licensed under the GPL license with a special exception
allowing the bundling of proprietary programs without making them subject
to the GPL.