

ICU 60: Overview and Cross-Distro Installation Guide

1 What ICU 60 is

- ICU (International Components for Unicode) is a widely used library for Unicode text handling, globalization, and locale-aware operations.
- Version 60 is older (e.g., shipped with Ubuntu 18.04).
- Some enterprise applications require ICU 60 at runtime.
- Modern systems often ship newer ICU versions (Ubuntu 20.04 → ICU 66, Fedora → ICU 70+), so binaries built for ICU 60 may fail to run if the library is missing.

2 Why side-by-side installation is necessary

- Do not replace system ICU — it's a core OS dependency.
- Installing ICU 60 alongside system ICU (e.g., in `/opt/icu60`) allows legacy binaries to run safely.
- Snap-based ICU 60 libraries exist but are sandboxed and not usable by non-Snap apps.

3 Cross-distro installation steps

Step 1: Install build tools and C++ compiler

- Fedora/Linux requires explicit C++ compiler installation:

```
sudo dnf groupinstall "Development Tools" -y
sudo dnf install gcc-c++ make autoconf libtool wget bzip2 -y
```

- Ubuntu/Debian:

```
sudo apt update
sudo apt install -y build-essential wget
```

- Arch/Manjaro:

```
sudo pacman -S base-devel wget
```

Step 2: Download ICU 60 source

```
cd /tmp
wget https://github.com/unicode-org/icu/releases/download/release-60-2/
icu4c-60_2-src.tgz
tar xf icu4c-60_2-src.tgz
cd icu/source
```

Step 3: Configure ICU 60 (enable shared libraries)

```
./runConfigureICU Linux --prefix=/opt/icu60 --enable-shared
```

- **Important:** `--enable-shared` is required; without it, ICU will only produce static `.a` libraries, and binaries like OTMaster will fail to find `libcui18n.so.60` at runtime. - On Fedora, ensure `gcc-c++` is installed; without it, configuration will fail.

Step 4: Build and install

```
make -j$(nproc)
sudo make install
```

- Verify shared libraries exist:

```
ls /opt/icu60/lib/libicu*.so.60
```

Expected output:

```
libcui18n.so.60
libcuc.so.60
libcudata.so.60
```

Step 5: Register ICU 60 with the linker

```
echo "/opt/icu60/lib" | sudo tee /etc/ld.so.conf.d/icu60.conf
sudo ldconfig
```

- Verify:

```
ldconfig -p | grep libcui18n.so.60
```

Step 6: Run your binary (DTL OTMaster)

• Directly:

```
/home/user/path/to/OTMaster
```

• Or temporarily using `LD_LIBRARY_PATH`:

```
LD_LIBRARY_PATH=/opt/icu60/lib:$LD_LIBRARY_PATH /home/user/path/to/OTMaster
```

4 Notes / Best Practices

- ICU 60 **must be built with** `--enable-shared` on Linux for runtime libraries.
- Installing `gcc-c++` explicitly is required on Fedora and some other distros.
- Always build ICU 60 **side-by-side** in `/opt/icu60`; do not replace system ICU.
- Use `ldconfig` or `LD_LIBRARY_PATH` to make libraries visible.
- Static `.a` libraries are not sufficient for binaries that expect `.so` files.

5 Optional cleanup

```
sudo rm -rf /opt/icu60
sudo rm /etc/ld.so.conf.d/icu60.conf
sudo ldconfig
```

This ensures the system is clean without touching system ICU.

✓ Summary - ICU 60 is required by legacy binaries.

- Snap ICU is sandboxed → unusable for system apps.
- Side-by-side installation in `/opt/icu60` with `--enable-shared` is safe, portable, and distro-independent.
- Update `LD_LIBRARY_PATH` or use `ldconfig` to make ICU 60 visible to binaries.