

# ✓ Migration Complete: .env → Local Config

## What Was Fixed

Your update script now automatically migrates settings from `.env` to the new local configuration system.

## What Happened

### Before

```
# config/local.local.json (template defaults)
{
  "wolfpack": {
    "ip": "192.168.1.100", ← Default template value
    "port": 4999           ← Default template value
  }
}

# .env (your actual settings)
WOLFPACK_HOST=192.168.1.100
WOLFPACK_PORT=23           ← Your real port
```

### After Migration

```
# config/local.local.json (your actual settings)
{
  "wolfpack": {
    "ip": "192.168.1.100", ← From your .env
    "port": 23             ← From your .env ✓
  }
}
```

## Files Added/Updated

### 1. Migration Script

`scripts/migrate-env-to-local-config.sh`

- Reads settings from `.env`
- Updates `config/local.local.json` with real values
- Backs up config before making changes
- Can be run manually anytime

### 2. Update Script Enhanced

`update_from_github.sh`

- Now runs migration automatically when needed
- Detects if local config has template defaults but `.env` has real values
- Migrates on first run after update
- Shows clear status messages

## Current Status

### ✓ Your settings are now migrated:

```
$ cat config/local.local.json | grep -A 6 wolfpack
"wolfpack": {
  "ip": "192.168.1.100",
  "port": 23,
  "protocol": "tcp",
  "timeout": 5000,
  "enabled": true,
  "defaultMatrixSize": "8x8"
}
```

### ✓ Backup created:

- config/local.local.json.backup (template version saved)

## How Update Script Works Now

### Scenario 1: First Time Setup (No Local Config)

```
🔧 Checking local configuration...
📝 Local configuration not found. Initializing from templates...
✓ Created: config/local.local.json

🔄 Migrating existing .env settings to local config...
✓ Wolfpack IP: 192.168.1.100
✓ Wolfpack Port: 23

✓ Local configuration initialized with your existing settings
```

### Scenario 2: Existing Config (Already Migrated)

```
🔧 Checking local configuration...
✓ Local configuration files found and preserved
Checking for new configuration options...
✓ Configuration is up to date
```

### Scenario 3: Config Exists but Needs Migration


```
🔧 Checking local configuration...
✓ Local configuration files found and preserved
📝 Detected .env settings not yet in local config...
🔄 Migrating .env to local config...
✓ Wolfpack IP: 192.168.1.100
✓ Wolfpack Port: 23
```




## Manual Migration


You can run the migration script manually anytime:

```
cd /home/ubuntu/Sports-Bar-TV-Controller
./scripts/migrate-env-to-local-config.sh
```

Output:

```
 Migrating .env settings to local configuration...

 Backed up: config/local.local.json.backup
 Updating Wolfpack settings...
 Wolfpack IP: 192.168.1.100
 Wolfpack Port: 23

 Migration complete!
```

## Verify Your Settings

Check your local config:




```
# View Wolfpack settings
cat config/local.local.json | grep -A 10 wolfpack

# View full config
cat config/local.local.json | jq .

# Compare with .env
grep WOLFPACK .env
```

## What Settings Get Migrated

From `.env` → `config/local.local.json` :

.env Variable	Config Location	Status
WOLFPACK_HOST	.wolfpack.ip	 Migrated
WOLFPACK_PORT	.wolfpack.port	 Migrated
API_PORT	.network.apiPort	 Migrated (if set)

Other `.env` variables (like `DATABASE_URL` , `NEXTAUTH_SECRET` , etc.) remain in `.env` - they're not part of the local config system.

## Backup & Restore

### Automatic Backup

The migration script automatically creates:

- `config/local.local.json.backup` (before migration)

## Restore from Backup

```
cd /home/ubuntu/Sports-Bar-TV-Controller

# Restore from migration backup
cp config/local.local.json.backup config/local.local.json

# Or restore from update backup
tar -xzf ~/sports-bar-backups/config-backup-YYYYMMDD-HHMMSS.tar.gz
```

## Testing

Test the migration worked correctly:

```
cd /home/ubuntu/Sports-Bar-TV-Controller

# Check Wolfpack settings
cat config/local.local.json | jq .wolfpack

# Should show:
# {
#   "ip": "192.168.1.100",
#   "port": 23,
#   "protocol": "tcp",
#   "timeout": 5000,
#   "enabled": true,
#   "defaultMatrixSize": "8x8"
# }
```

## Next Steps

### 1. Review your settings:

```
bash
nano config/local.local.json
```

### 2. Add device inventory:

```
bash
nano config/devices.local.json
```

### 3. Set sports preferences:

```
bash
nano config/sports-teams.local.json
```

### 4. Update is safe:

```
bash
./update_from_github.sh
# Your settings will be preserved!
```

## Pushed to GitHub

✓ **Committed:** 575ded7

- Migration script

- Updated update script
- This documentation

**Your system is now fully configured!** 🎉

## Summary

---

- ✓ Settings migrated from .env to local config
  - ✓ Update script enhanced with auto-migration
  - ✓ Your Wolfpack port (23) now in local config
  - ✓ Automatic backup created
  - ✓ Safe to run updates anytime
  - ✓ All changes pushed to GitHub
- 

### Related Documentation:

- `UPDATE_SCRIPT_GUIDE.md` - Update script details
- `LOCAL_CONFIG_SYSTEM.md` - Config system guide
- `CONFIG_DEMO.md` - Visual examples
- `config/README.md` - Configuration reference

**Last Updated:** October 1, 2025