

User Migration Checklist

WP Subdomain Login Plugin

Version: 1.0.0

Author: Douglas LaMar - LaMarSOFT.com

Pre-Migration Phase

Backup Everything

- Backup main site database

```
mysqldump -u username -p main_database > main_site_backup.sql
```

- Backup subdomain database

```
mysqldump -u username -p subdomain_database > subdomain_backup.sql
```

- Backup WordPress files (both sites)
- Store backups in a secure location
- Verify backups can be restored

Document Current State

- Count total users on main site: _____
- Count total users on subdomain: _____

- List user roles on main site
- List custom user meta fields to preserve
- Document any custom user functionality

Environment Preparation

- Set up staging environment
- Install WP Subdomain Login plugin on main site (v1.0.1+)
- Install WP Subdomain Login plugin on subdomain (v1.0.1+)
- Configure shared secret key on both sites
- Test plugin on staging first

Database Access

- Obtain main site database credentials
 - Username: _____
 - Password: _____
 - Database name: _____
 - Host: _____
 - Obtain subdomain database credentials
 - Test database connections
 - Verify user has sufficient privileges
-

Migration Phase

Choose Migration Method

- Method 1:** SQL Direct Export/Import (recommended for 100+ users)
- Method 2:** CSV Export/Import (best for < 100 users)
- Method 3:** Custom PHP Script (for complex requirements)

Prepare Migration Script

- Download migrate-users.php script
- Update database credentials in script
- Set allowed IP addresses
- Configure migration options:
 - Skip existing users: Yes / No
 - Copy user metadata: Yes / No
 - Default role: _____
- Enable DRY_RUN mode for testing

Test Migration (Dry Run)

- Upload script to subdomain root
- Run script in DRY_RUN mode
- Review output for potential issues
- Verify user count matches expectations
- Check for any error messages

Execute Migration

- Disable DRY_RUN mode
- Run migration script
- Monitor progress
- Note any errors or warnings
- DELETE migration script immediately after completion**

Verify Migration

- Check user count on subdomain matches main site
- Verify sample usernames exist on subdomain
- Confirm password hashes match (use SQL query)

- Check user metadata was copied correctly
 - Verify user roles are assigned properly
-

Testing Phase

Test User Login Flow

- [] **Test 1:** User exists on both sites
 - Login via main site form
 - Verify redirect to subdomain
 - Confirm logged in on subdomain
 - Check user role and permissions
- [] **Test 2:** User only on main site
 - Attempt login
 - Should see: “User account not found on this site”
 - Create user on subdomain
 - Retry login - should succeed
- [] **Test 3:** Password mismatch
 - Change password on main site only
 - Attempt login - should fail
 - Update password on subdomain
 - Retry - should succeed
- [] **Test 4:** Token expiry
 - Initiate login
 - Wait 6 minutes
 - Complete login
 - Should see: “Invalid or expired authentication token”

- [] **Test 5: Different user IDs**
 - Find user with different IDs on each site
 - Login via main site
 - Verify correct user logged in on subdomain
 - Check user data matches

SQL Verification Queries

- Run on main site:

```
SELECT user_login, user_email FROM wp_users ORDER BY user_login;
```

- Run on subdomain (compare results):

```
SELECT user_login, user_email FROM wp_users ORDER BY user_login;
```

- Verify password hashes match:

```
-- Main site
SELECT user_login, user_pass FROM wp_users WHERE user_login =
'testuser';
-- Subdomain (hash should be identical)
SELECT user_login, user_pass FROM wp_users WHERE user_login =
'testuser';
```

Post-Migration Phase

User Communication

- Prepare user notification email

- Send email to all users explaining changes
- Include instructions for password reset if needed
- Provide support contact information
- Create help documentation

Monitoring Setup

- Install login monitoring plugin
- Set up alerts for failed login attempts
- Monitor for authentication errors
- Track successful cross-site logins
- Review logs daily for first week

Password Synchronization

- Install password sync plugin on main site
- Install password sync plugin on subdomain
- Test password change on main site
- Verify password updated on subdomain
- Document sync process for team

Security Measures

- Delete migration script from server
- Remove database credentials from any config files
- Review file permissions
- Enable SSL on both sites
- Update security headers

Documentation

- Document migration process
- Record any issues encountered and solutions

- Update internal wiki/knowledge base
- Create troubleshooting guide for support team
- Save migration logs for reference

Regular Maintenance

- Schedule weekly user count audits
 - Set up monthly password sync verification
 - Plan for ongoing user synchronization
 - Review and update documentation quarterly
 - Keep WP Subdomain Login plugin updated
-

Troubleshooting Reference

Common Issues

Issue: User account not found on this site

Solution: User doesn't exist on subdomain - run migration again or create user manually

Issue: Authentication fails but user exists

Solution: Password hashes don't match - copy hash from main site to subdomain

Issue: Wrong user logged in on subdomain

Solution: Ensure plugin version 1.0.1+ (uses username, not user ID)

Issue: Some users can login, others cannot

Solution: Incomplete migration - find missing users and migrate them

Issue: User roles are wrong on subdomain

Solution: Reset user role using WordPress admin or PHP script

Emergency Rollback Plan

If Migration Fails

1. **Stop immediately** - Do not continue migration
2. **Restore subdomain database** from backup

```
mysql -u username -p subdomain_database < subdomain_backup.sql
```

3. **Verify backup restoration** - Check user count
4. **Investigate issue** - Review error logs
5. **Fix problem** - Update script or configuration
6. **Test in staging** - Verify fix works
7. **Retry migration** - Run script again

Backup Restoration Commands

```
# Restore database
mysql -u username -p database_name < backup_file.sql

# Restore files
tar -xzf backup_files.tar.gz -C /path/to/restore/

# Verify restoration
mysql -u username -p -e "SELECT COUNT(*) FROM database_name.wp_users;"
```

Support Resources

- **Plugin Documentation:** [Link to docs]
- **Migration Guide:** [Link to full guide]
- **Video Tutorials:** [Link to videos]

- **Support Email:** support@lamarsoft.com
 - **Website:** <https://lamarsoft.com>
-

Sign-Off

Migration Completed By: _____

Date: _____

Time: _____

Total Users Migrated: _____

Issues Encountered: _____

Notes: _____

Document Version: 1.0.0

Last Updated: January 7, 2026

Author: Douglas LaMar - LaMarSOFT.com