Phase 1 Manual Test Plan: Security Hardening

dbus-mqtt-bridge Debian Package Testing

This test plan covers manual testing of Phase 1 security features that cannot be fully automated. Run these tests after building the Debian package.

Prerequisites

Test Environment

- Debian-based system (Debian 11+, Ubuntu 20.04+, or similar)
- systemd-based init system
- D-Bus daemon running
- Root/sudo access
- MQTT broker (mosquitto) for integration tests

Build the Package

```bash
cd /path/to/dbus-mqtt-bridge
dpkg-buildpackage -us -uc -b
```

Test Suite 1: Package Installation

Test 1.1: Fresh Installation

Objective: Verify package installs correctly and creates necessary resources.

```bash
# Install package
sudo dpkg -i ../dbus-mqtt-bridge_0.1.0-1_amd64.deb

# May need to install dependencies
sudo apt-get install -f
```

Verification Checklist:

- □ Package installs without errors
- □ Post-installation message displayed
- □ User `dbus-mqtt-bridge` created
- □ Group `dbus-mqtt-bridge` created
- □ Directory `/etc/dbus-mqtt-bridge` exists
- □ Directory `/var/log/dbus-mqtt-bridge` exists
- □ Service file installed at `/lib/systemd/system/dbus-mqtt-bridge.service`
- □ Example config at `/etc/dbus-mqtt-bridge/config.yaml`
- □ D-Bus policy at `/etc/dbus-1/system.d/dbus-mqtt-bridge.conf`
- □ Man page at `/usr/share/man/man1/dbus-mqtt-bridge.1.gz` (when implemented)

Commands to verify:

```bash
# Check user/group
getent passwd dbus-mqtt-bridge
getent group dbus-mqtt-bridge

# Check directories
ls -ld /etc/dbus-mqtt-bridge
ls -ld /var/log/dbus-mqtt-bridge

# Check files
ls -l /etc/dbus-mqtt-bridge/config.yaml
ls -l /etc/dbus-1/system.d/dbus-mqtt-bridge.conf
systemctl cat dbus-mqtt-bridge
```

Expected Results:

```
User: dbus-mqtt-bridge:x:NNN:NNN:D-Bus MQTT Bridge Service:/nonexistent:/usr/sbin/nologin
Group: dbus-mqtt-bridge:x:NNN:

Directories:
drwxr-x--- 2 root dbus-mqtt-bridge 4096 ... /etc/dbus-mqtt-bridge
drwxr-x--- 2 dbus-mqtt-bridge dbus-mqtt-bridge 4096 ... /var/log/dbus-mqtt-bridge

Config file:
-rw-r----- 1 root dbus-mqtt-bridge ... /etc/dbus-mqtt-bridge/config.yaml
```

Test 1.2: File Permissions

Objective: Verify security-critical file permissions are correct.

```bash
# Check config directory
stat -c "%a %U:%G" /etc/dbus-mqtt-bridge

# Check config file
stat -c "%a %U:%G" /etc/dbus-mqtt-bridge/config.yaml

# Check log directory
stat -c "%a %U:%G" /var/log/dbus-mqtt-bridge

# Check D-Bus policy
stat -c "%a %U:%G" /etc/dbus-1/system.d/dbus-mqtt-bridge.conf

# Check service file
stat -c "%a %U:%G" /lib/systemd/system/dbus-mqtt-bridge.service
```

Expected Results:

```
/etc/dbus-mqtt-bridge: 750 root:dbus-mqtt-bridge
/etc/dbus-mqtt-bridge/config.yaml: 640 root:dbus-mqtt-bridge
/var/log/dbus-mqtt-bridge: 750 dbus-mqtt-bridge:dbus-mqtt-bridge
/etc/dbus-1/system.d/dbus-mqtt-bridge.conf: 644 root:root
/lib/systemd/system/dbus-mqtt-bridge.service: 644 root:root
```

Verification Checklist:

- □ Config directory: 0750, root:dbus-mqtt-bridge
- □ Config file: 0640, root:dbus-mqtt-bridge
- □ Log directory: 0750, dbus-mqtt-bridge:dbus-mqtt-bridge
- □ D-Bus policy: 0644, root:root
- □ Service file: 0644, root:root

Test 1.3: Service Disabled by Default

Objective: Verify service is installed but not enabled or started.

```bash
# Check service status
systemctl is-enabled dbus-mqtt-bridge
systemctl is-active dbus-mqtt-bridge
systemctl status dbus-mqtt-bridge
```

Expected Results:

```
disabled
inactive
● dbus-mqtt-bridge.service - DBus-MQTT Bridge Service
   Loaded: loaded (/lib/systemd/system/dbus-mqtt-bridge.service; disabled; ...)
   Active: inactive (dead)
```

Verification Checklist:

□ Service is-enabled: disabled
□ Service is-active: inactive
□ Service status shows: disabled, inactive (dead)

Test Suite 2: Systemd Security Hardening

Test 2.1: Service User

Objective: Verify service runs as unprivileged user.

```bash
# Check service configuration
systemctl cat dbus-mqtt-bridge | grep -E "^User=|^Group="

# Attempt to start (will fail without proper config, but shows user)
sudo systemctl start dbus-mqtt-bridge
sleep 2
systemctl status dbus-mqtt-bridge

# Check process user (if service starts)
ps aux | grep dbus-mqtt-bridge | grep -v grep
```

Expected Results:

```
User=dbus-mqtt-bridge
Group=dbus-mqtt-bridge

Process should run as user 'dbus-mqtt-bridge', NOT root.
```

Verification Checklist:

□ Service unit has User=dbus-mqtt-bridge
□ Service unit has Group=dbus-mqtt-bridge
□ Service unit does NOT have User=root

Test 2.2: Security Hardening Options

Objective: Verify all systemd hardening options are present.

```bash
systemctl cat dbus-mqtt-bridge | grep -E "^NoNewPrivileges=|^ProtectSystem=|^ProtectHome=|^PrivateT
systemctl cat dbus-mqtt-bridge | grep -E "^ProtectKernel|^Restrict"
systemctl cat dbus-mqtt-bridge | grep -E "^Capability"
```

Verification Checklist:

□ NoNewPrivileges=true
□ PrivateTmp=true
□ ProtectSystem=strict
□ ProtectHome=true
□ ProtectKernelTunables=true
□ ProtectKernelModules=true
□ ProtectControlGroups=true
□ RestrictAddressFamilies=AF_UNIX AF_INET AF_INET6
□ RestrictNamespaces=true
□ LockPersonality=true
□ RestrictRealtime=true
□ RestrictSUIDSGID=true
□ RemoveIPC=true
□ PrivateMounts=true
□ SystemCallFilter=@system-service
□ CapabilityBoundingSet= (empty, no capabilities)

Test 2.3: Filesystem Protection

Objective: Verify filesystem access is properly restricted.

This requires the service to be running. Configure a minimal config first:

```bash
# Create minimal working config
sudo tee /etc/dbus-mqtt-bridge/config.yaml > /dev/null <<EOF
mqtt:
  broker: localhost
  port: 1883
bus_type: system
mappings:
  dbus_to_mqtt: []
  mqtt_to_dbus: []
EOF

# Update D-Bus policy to allow basic access
sudo tee /etc/dbus-1/system.d/dbus-mqtt-bridge.conf > /dev/null <<EOF
<!DOCTYPE busconfig PUBLIC "-//freedesktop//DTD D-BUS Bus Configuration 1.0//EN"
 "http://www.freedesktop.org/standards/dbus/1.0/busconfig.dtd">
<busconfig>
  <policy user="dbus-mqtt-bridge">
    <allow send_destination="org.freedesktop.DBus"/>
    <allow receive_sender="org.freedesktop.DBus"/>
  </policy>
</busconfig>
EOF

# Reload D-Bus
sudo systemctl reload dbus

# Start MQTT broker
sudo systemctl start mosquitto

# Start service
sudo systemctl start dbus-mqtt-bridge

# Check it's running
systemctl status dbus-mqtt-bridge
```

Test filesystem restrictions:

```bash
# Get PID
PID=$(systemctl show -p MainPID --value dbus-mqtt-bridge)

# Check what filesystems are visible
sudo ls -la /proc/$PID/root/

# Check if /home is accessible (should not be)
sudo ls /proc/$PID/root/home 2>&1

# Check if /etc/dbus-mqtt-bridge is readable (should be)
sudo ls /proc/$PID/root/etc/dbus-mqtt-bridge 2>&1
```

Verification Checklist:

- □ Service starts successfully with minimal config
- □ /home is not accessible to service (ProtectHome=true)
- □ /etc/dbus-mqtt-bridge is readable (ReadOnlyPaths)
- □ /var/log/dbus-mqtt-bridge is writable (ReadWritePaths)

Test Suite 3: D-Bus Policy Enforcement

Test 3.1: Policy File Installed

Objective: Verify D-Bus policy is installed and valid.

```bash
# Check policy file exists
ls -l /etc/dbus-1/system.d/dbus-mqtt-bridge.conf

# Validate XML syntax
xmllint --noout /etc/dbus-1/system.d/dbus-mqtt-bridge.conf
echo "Exit code: $?"

# Check D-Bus can parse it
sudo dbus-send --system --print-reply \
  --dest=org.freedesktop.DBus \
  / org.freedesktop.DBus.ReloadConfig
```

Verification Checklist:

- □ Policy file exists
- □ XML is valid (xmllint returns 0)
- □ D-Bus reloads config without error
- □ Policy contains user="dbus-mqtt-bridge"

Test 3.2: Policy Enforcement (Access Denied)

Objective: Verify D-Bus policy actually restricts access.

Setup: Create a test that tries to access a D-Bus service not in the policy.

```bash
# Try to call a method that should be denied
# This should fail if policy is working
sudo -u dbus-mqtt-bridge dbus-send \
  --system \
  --print-reply \
  --dest=org.freedesktop.systemd1 \
  /org/freedesktop/systemd1 \
  org.freedesktop.systemd1.Manager.ListUnits 2>&1 | grep -i "denied\|not allowed"
```

Expected Result:

```
Error: Access denied
```

Verification Checklist:

- □ D-Bus denies access to services not in policy
- □ Error message indicates permission denied

Test 3.3: Policy Enforcement (Access Allowed)

Objective: Verify D-Bus policy allows configured access.

```bash
# Modify policy to allow DBus introspection
sudo tee -a /etc/dbus-1/system.d/dbus-mqtt-bridge.conf > /dev/null <<'EOF'
  <policy user="dbus-mqtt-bridge">
    <allow send_destination="org.freedesktop.DBus"
          send_interface="org.freedesktop.DBus"
          send_type="method_call"/>
  </policy>
EOF

# Reload D-Bus
sudo systemctl reload dbus

# Now this should work
sudo -u dbus-mqtt-bridge dbus-send \
  --system \
  --print-reply \
  --dest=org.freedesktop.DBus \
  / \
  org.freedesktop.DBus.ListNames
```

Expected Result:

```
method return time=... sender=... -> destination=...
  array [
    string "org.freedesktop.DBus"
    ...
  ]
```

Verification Checklist:

□ Allowed D-Bus calls succeed

□ Response received without errors

Test Suite 4: Package Upgrade

Test 4.1: Upgrade with Existing Config

Objective: Verify upgrade preserves user configuration.

```bash
# Modify config
echo "# My custom comment" | sudo tee -a /etc/dbus-mqtt-bridge/config.yaml

# Rebuild package with new version
# (In practice, bump version in debian/changelog)

# Install new version
sudo dpkg -i ../dbus-mqtt-bridge_0.1.1-1_amd64.deb

# Check config preserved
grep "My custom comment" /etc/dbus-mqtt-bridge/config.yaml
```

Verification Checklist:

- □ Package upgrades without removing config
- □ User modifications to config preserved
- □ Service remains in same state (disabled if was disabled)

Test Suite 5: Package Removal

Test 5.1: Remove (Keep Config)

Objective: Verify remove keeps configuration.

```bash
# Remove package
sudo apt-get remove dbus-mqtt-bridge

# Check files
ls /etc/dbus-mqtt-bridge/config.yaml
ls /var/log/dbus-mqtt-bridge
getent passwd dbus-mqtt-bridge
```

Verification Checklist:

- □ Binary removed
- □ Service file removed
- □ Config directory remains
- □ Log directory remains
- □ User/group remain

Test 5.2: Purge (Remove All)

Objective: Verify purge removes all files and user.

```bash
# Purge package
sudo apt-get purge dbus-mqtt-bridge

# Check everything removed
ls /etc/dbus-mqtt-bridge 2>&1
ls /var/log/dbus-mqtt-bridge 2>&1
getent passwd dbus-mqtt-bridge
getent group dbus-mqtt-bridge
ls /etc/dbus-1/system.d/dbus-mqtt-bridge.conf 2>&1
```

Verification Checklist:

- □ /etc/dbus-mqtt-bridge removed
- □ /var/log/dbus-mqtt-bridge removed
- □ User dbus-mqtt-bridge removed
- □ Group dbus-mqtt-bridge removed
- □ D-Bus policy removed
- □ All package files removed

Test Suite 6: Security Validation

Test 6.1: Privilege Escalation Attempt

Objective: Verify service cannot escalate privileges.

Test: Try to get root access from within service context.

```bash
# This test requires the service to be running
# The hardening should prevent any privilege escalation

# Check capabilities
sudo systemd-analyze security dbus-mqtt-bridge

# Look for FAIL or EXPOSED issues
```

Verification Checklist:

- □ systemd-analyze shows good security score
- □ No capabilities granted
- □ No SUID/SGID allowed

Test 6.2: System Call Filtering

Objective: Verify system call filtering is active.

```bash
# Check which syscalls are blocked
systemctl show dbus-mqtt-bridge | grep SystemCallFilter

# The service should only allow @system-service syscalls
```

Verification Checklist:

□ SystemCallFilter is configured
□ Only safe syscalls allowed

Test Suite 7: Integration Testing

Test 7.1: End-to-End with Real Config

Objective: Verify service works with actual D-Bus and MQTT.

Setup:

1. Install and start mosquitto
2. Create a working config that monitors a safe D-Bus signal
3. Configure appropriate D-Bus policy
4. Start service
5. Trigger D-Bus signal
6. Verify MQTT message received

```bash
# Example: Monitor NetworkManager signals (if available)
# Configure this in /etc/dbus-mqtt-bridge/config.yaml and D-Bus policy
# Then:

sudo systemctl restart dbus-mqtt-bridge
mosquitto_sub -t 'dbus/#' -v &
# Trigger network change (e.g., disconnect/reconnect WiFi)
```

Verification Checklist:

- □ Service starts without errors
- □ D-Bus signals are received
- □ MQTT messages are published
- □ No permission errors in logs

Summary Checklist

After completing all tests, verify:

- □ All Test Suite 1 tests passed (Installation)
- □ All Test Suite 2 tests passed (Systemd Hardening)
- □ All Test Suite 3 tests passed (D-Bus Policy)
- □ All Test Suite 4 tests passed (Upgrade)
- □ All Test Suite 5 tests passed (Removal)
- □ All Test Suite 6 tests passed (Security)
- □ All Test Suite 7 tests passed (Integration)

Known Issues / Expected Failures

Document any expected failures or environment-specific issues:

1. **Service fails to start without MQTT broker**
   - Expected: Service should fail gracefully with clear error
   - This is correct behavior
2. **D-Bus policy too restrictive**
   - If integration test fails, may need to adjust policy
   - Document which permissions are actually needed
3. **systemd version differences**
   - Some hardening options may not be available on older systemd
   - Check systemd version: `systemd --version`

Regression Testing

For future changes, re-run:

- Test Suite 1 (Installation)
- Test Suite 2.1 (Service User)
- Test Suite 3 (D-Bus Policy)
- Test Suite 5 (Removal/Purge)

These are the critical security tests that must always pass.