# Key Rotation Runbook

## RichesReach Security Operations

**Purpose:** Operational guide for rotating API keys and secrets
**Owner:** Security Team
**Estimated Time:** 15-30 minutes per key

## Overview

This runbook provides step-by-step instructions for rotating API keys and secrets stored in AWS Secrets Manager. All rotations follow a "create new, test, deploy, deactivate old" pattern to minimize downtime.

## Prerequisites

- AWS CLI configured with appropriate permissions
- Access to AWS Secrets Manager
- Access to deployment environment
- Access to third-party provider dashboards (Yodlee, Alpaca, etc.)

## Rotation Process

### Step 1: Create New Secret in AWS Secrets Manager

```
 # Get current secret
aws secretsmanager get-secret-value \
  --secret-id richesreach/production/yodlee_client_id \
  --query SecretString --output text

# Create new secret version (staged)
aws secretsmanager put-secret-value \
  --secret-id richesreach/production/yodlee_client_id \
  --secret-string "NEW_CLIENT_ID_VALUE" \
  --version-stages AWSPENDING
```

### Step 2: Test New Secret

```
 # Update local environment temporarily
export YODLEE_CLIENT_ID="NEW_CLIENT_ID_VALUE"

# Test endpoint
curl -X GET https://api.richesreach.com/api/yodlee/fastlink/start \
  -H "Authorization: Bearer <token>"

# Verify response is successful
```

### Step 3: Promote New Secret

```
 # Promote to current version
aws secretsmanager put-secret-value \
  --secret-id richesreach/production/yodlee_client_id \
  --secret-string "NEW_CLIENT_ID_VALUE" \
  --version-stages AWSCURRENT
```

### Step 4: Restart Services

```
 # Restart backend services to load new secret
# (ECS will auto-restart if using task definition secrets)
aws ecs update-service \
  --cluster richesreach-production \
  --service backend \
  --force-new-deployment
```

### Step 5: Verify New Secret in Use

```
 # Check logs for successful initialization
# Should see: "YodleeClient initialized: client_id_length=XX"

# Test production endpoint
curl -X GET https://api.richesreach.com/api/yodlee/fastlink/start \
  -H "Authorization: Bearer <token>"
```

### Step 6: Deactivate Old Secret (After 24-48h)

```
 # After confirming new secret works for 24-48 hours
# Deactivate old secret in third-party provider dashboard
# (Yodlee, Alpaca, etc.)

# Optionally delete old secret version in AWS
aws secretsmanager delete-secret \
  --secret-id richesreach/production/yodlee_client_id \
  --recovery-window-in-days 7
```

---

## Key-Specific Instructions

### Yodlee Client ID & Secret

**Location:** AWS Secrets Manager - `richesreach/production/yodlee_client_id` - `richesreach/production/yodlee_secret`

**Third-Party Steps:** 1. Log into Yodlee Developer Portal 2. Generate new client credentials 3. Update in AWS Secrets Manager 4. Test FastLink flow 5. Deactivate old credentials after 48h

**Rollback:** Revert to previous secret version in AWS Secrets Manager

---

### Market Data API Keys

**Location:** AWS Secrets Manager - `richesreach/production/alpha_vantage_key_1` - `richesreach/production/polygon_api_key` - `richesreach/production/finnhub_api_key`

**Third-Party Steps:** 1. Generate new API key in provider dashboard 2. Update in AWS Secrets Manager 3. Test market data endpoints 4. Deactivate old key after 24h

**Note:** Multiple keys supported for rotation (key_1, key_2, etc.)

---

### Database Passwords

**Location:** AWS Secrets Manager - `richesreach/production/db_password`

**Steps:** 1. Create new password in RDS 2. Update secret in AWS Secrets Manager 3. Update RDS master password 4. Restart services 5. Verify connection

**Rollback:** Revert RDS password to previous value

---

## Emergency Rotation

**If key is compromised:**

1. **Immediately rotate** (skip testing phase)
2. **Revoke old key** in third-party dashboard
3. **Update AWS Secrets Manager**
4. **Restart all services**
5. **Monitor for errors**
6. **Document incident**

---

## Verification Checklist

After rotation, verify:

- [ ] New secret loaded successfully (check logs)
- [ ] API endpoints respond correctly
- [ ] No authentication errors
- [ ] Third-party integrations working
- [ ] Old secret deactivated (after grace period)
- [ ] Rotation documented in security log

## Automation (Future)

**Planned:** Automated rotation via AWS Lambda - Scheduled rotation (90 days for Yodlee, 180 days for market data) - Automatic testing - Rollback on failure - Notification on completion

**Status:** Manual rotation for now, automation planned for Q2 2026

## Contact

**Security Team:** security@richesreach.com
**On-Call:** See PagerDuty rotation

**Last Updated:** 2026-01-XX
**Next Review:** Quarterly