

# SYSTEMD クイックリファレンス

サービス管理、ユニット、タイマー、journalctl

## サービス管理

### 基本的なサービスコマンド

```
systemctl start nginx
systemctl stop nginx
systemctl restart nginx
systemctl reload nginx # reload config
systemctl status nginx
```

### 有効化と無効化

```
systemctl enable nginx # start at boot
systemctl disable nginx # remove from boot
systemctl enable --now nginx # enable + start
systemctl is-enabled nginx
```

### サービスの状態

**active (running)** サービスが正常に動作中  
**active (exited)** 一度実行して正常終了  
**inactive (dead)** サービスが停止中  
**failed** サービスがクラッシュまたはエラー終了  
**activating** サービスが起動中

## ユニットファイル

### ユニットファイルの場所

**/etc/systemd/system/** 管理者作成ユニット (最高優先度)  
**/run/systemd/system/** 実行時生成ユニット  
**/usr/lib/systemd/system/** パッケージインストール済みユニット  
**~/ .config/systemd/user/** ユーザーレベルのユニット

### 基本的なサービスユニット

```
[Unit]
Description=My Application
After=network.target
[Service]
ExecStart=/usr/bin/myapp --config /etc/myapp.conf
Restart=on-failure
User=appuser
[Install]
WantedBy=multi-user.target
```

### 変更の適用

```
systemctl daemon-reload # reload unit files
systemctl restart myapp # apply changes
```

## タイマー

### タイマーユニット

```
[Unit]
Description=Run backup daily
[Timer]
OnCalendar=*-*-* 02:00:00
Persistent=true
[Install]
WantedBy=timers.target
```

### OnCalendarの構文

**\*-\*-\* 02:00:00** 毎日午前2時  
**Mon \*-\*-\* 09:00:00** 毎週月曜日午前9時  
**\*-\*-\* 01 00:00:00** 毎月1日  
**hourly / daily / weekly** 省略スケジュール

### タイマーの管理

```
systemctl list-timers -all
systemctl start backup.timer
systemctl enable backup.timer
systemd-analyze calendar "daily"
```

## ターゲット

### 主要ターゲット

**multi-user.target** 通常起動、マルチユーザー、GUIなし  
**graphical.target** フルGUIデスクトップ  
**rescue.target** シングルユーザーレスキューモード  
**emergency.target** 最小シールド、rootのみ  
**network-online.target** ネットワークが完全に設定済み  
**timers.target** すべてのタイマーユニットが準備済み

### ターゲットコマンド

```
systemctl get-default
systemctl set-default multi-user.target
systemctl isolate rescue.target
systemctl list-dependencies graphical.target
```

## Journalctl

### ログの表示

```
journalctl -u nginx # logs for unit
journalctl -u nginx -f # follow (tail)
journalctl -u nginx --no-pager
journalctl -b # current boot only
```

### ログのフィルタリング

```
journalctl --since "2026-03-01"
journalctl --since "1 hour ago"
journalctl -p err # errors and above
journalctl -PID=1234
```

### 優先度レベル

**emerg (0)** システムが使用不能  
**alert (1)** 即時対応が必要  
**crit (2)** 重大な状態  
**err (3)** エラー状態  
**warning (4)** 警告状態  
**info (6)** 情報メッセージ  
**debug (7)** デバッグレベルのメッセージ

### ログのメンテナンス

```
journalctl --disk-usage
journalctl --vacuum-size=500M
journalctl --vacuum-time=30d
```

## ネットワーク

### networkctl

```
networkctl list
networkctl status eth0
networkctl up eth0
networkctl down eth0
```

### systemd-resolve

```
resolvectl status
resolvectl query example.com
resolvectl flush-caches
resolvectl statistics
```

### ネットワーク待機

```
# In unit file [Unit] section:
After=network-online.target
Wants=network-online.target
```

## マウント

### マウントユニット

```
[Unit]
Description=Mount data volume
[Mount]
What=/dev/sdb1
Where=/mnt/data
Type=ext4
Options=defaults,noatime
[Install]
WantedBy=multi-user.target
```

### 自動マウントユニット

```
[Unit]
Description=Automount data on access
[Automount]
Where=/mnt/data
TimeoutIdleSec=300
[Install]
WantedBy=multi-user.target
```

### 命名規則

**/mnt/data** ユニットファイル: `mnt-data.mount`  
**/var/lib/app** ユニットファイル: `var-lib-app.mount`

マウントパスの `/` を `.` に置換、先頭のダッシュは除去

## 環境変数

### 環境変数の設定

```
[Service]
Environment=APP_ENV=production
Environment=PORT=8080
EnvironmentFile=/etc/myapp/env
```

### 環境変数ファイルの形式

```
# /etc/myapp/env
APP_ENV=production
DATABASE_URL=postgres://localhost/db
SECRET_KEY=changeme
```

### サービスのセキュリティ強化

**ProtectSystem=strict** 許可パス以外は読み取り専用ファイルシステム  
**ProtectHome=true** /home、/root、/run/user を非表示  
**NoNewPrivileges=true** 特権昇格を防止  
**PrivateTmp=true** サービス専用の分離 /tmp  
**ReadWritePaths=/var/lib/myapp** 特定パスへの書き込みを許可

## 依存関係

### 順序と依存関係のディレクティブ

**After=b.service** bの後に起動 (順序のみ)  
**Before=b.service** bの前に起動 (順序のみ)  
**Requires=b.service** 強い依存。bが失敗すると自分も失敗  
**Wants=b.service** 弱い依存。bが失敗しても自分は失敗しない  
**BindTo=b.service** bが停止すると自分も停止  
**Conflicts=b.service** bと同時に実行できない

### 依存関係の確認

```
systemctl list-dependencies nginx
systemctl list-dependencies --reverse nginx
systemd-analyze dot nginx.service | dot -Tsvg > deps.svg
```

## よく使うパターン

### 再起動ポリシー

**Restart=no** 再起動しない (デフォルト)  
**Restart=on-failure** 非ゼロ終了時に再起動  
**Restart=always** 常に再起動 (デーモン向け)  
**RestartSec=5** 再起動前に5秒待機  
**StartLimitBurst=3** インターバル内の最大再起動回数  
**StartLimitIntervalSec=60** ハーストカウントのインターバル

### 元ファイルを編集せずにオーバーライド

```
systemctl edit nginx # creates drop-in
# /etc/systemd/system/nginx.service.d/override.conf
systemctl cat nginx # show effective config
systemctl revert nginx # remove overrides
```

### システム分析

```
systemd-analyze # boot time
systemd-analyze blame # per-unit time
systemd-analyze critical-chain
systemctl list-units --failed
```