Dashboard Guide

Borgmatic Guide

Learn how Borgmatic works and get started with your backup strategy

What is Borgmatic Director UI?

Borgmatic Director UI is a modern web-based management interface for **Borgmatic**, which is a simple, configuration-driven backup software built on top of **Borg Backup**. Borgmatic automates the creation of backups, handles encryption, compression, and provides a powerful deduplication system that saves storage space by only storing unique data chunks. Borgmatic Director UI provides an intuitive interface to manage your backups, repositories, schedules, and archives without editing configuration files manually.

Key Features

- Automatic backup scheduling
- Deduplication (saves storage)
- Encryption and compression
- Multiple repository support
- Pre/post-backup hooks
- Template-based setup
- Database auto-discovery

How It Works

Borgmatic Director UI runs in a single Docker container that includes everything you need. Configuration files (YAML) define what to backup, where to store it, and when to run. Backups execute automatically according to your schedule, creating incremental archives that only store changes. You can also use templates to quickly set up common backup configurations.

Quick Start Guide

⚡ Fast Track: Use Templates

If you're setting up backups for common applications (like WordPress, BookStack, or other Infinity Tools), you can use **Templates** to automatically configure everything. Go to "Templates" ? Select a template ? Test connection ? Activate. This creates repositories, schedules, and backup jobs automatically!

Manual Setup (follow these steps in order):

1. Create SSH Keys (if using SSH/SFTP repositories)
2. Create a Repository (where backups will be stored)
3. Create a Schedule (when backups should run)
4. Create a Backup Job (what to backup and how)

1

SSH Keys (Optional)

If you plan to use SSH or SFTP repositories, you'll need to create SSH keys first. These keys allow secure, passwordless access to remote servers.

When needed: Only if using SSH or SFTP repository types
Where: Go to "SSH Keys" in the navigation menu

Create a Repository

A **repository** is where your backups are stored. It can be local, on a remote server (SSH/SFTP), or in the cloud (S3, Rclone). Think of it as the "destination" for your backups.

Where: Go to "Repositories" ? Click "Create Repository"
Tip: Click "Read this first!" on the Repositories page to learn about different repository types and performance.

? Example: Create an SSH repository pointing to `ssh://user@server.com/var/backups/borg`

Create a Schedule

A **schedule** defines when backups should run. It uses cron syntax to specify the frequency (e.g., daily at 2 AM, weekly on Sundays, etc.).

Where: Go to "Schedules" ? Click "Create Schedule"
Tip: You can reuse the same schedule for multiple backup jobs

? Example: Create a schedule with cron expression `0 2 * * *` (runs daily at 2:00 AM)

Create a Backup Job

A **backup job** (or "backup") ties everything together. It specifies:

- What files/directories to backup (source paths)
- Which repository to use (destination)
- Which schedule to follow (when to run)
- Retention policies (how long to keep backups)
- Pre/post-backup commands (optional)

Where: Go to "Backups" ? Click "Create Backup"
Tip: You can create multiple backup jobs using the same repository and schedule

? Example: Create a backup job that backs up `/home` and `/etc` to your SSH repository, running daily at 2 AM, keeping 7 daily, 4 weekly, and 12 monthly backups.

Key Terminology

Repository

A **repository** is the storage location where all your backups are stored. It's like a "vault" that contains multiple backup archives.

Key points:

- One repository can hold multiple backup jobs
- Repositories can be local, remote (SSH), or cloud (S3)
- Repositories are encrypted and deduplicated
- You need at least one repository before creating backups

Archive

An **archive** is a single backup snapshot created at a specific point in time. Each time a backup runs, it creates a new archive in the repository.

Key points:

- Each archive has a unique name (usually timestamp-based)
- Archives are incremental (only store changes)
- You can restore from any archive
- Old archives are pruned based on retention policies

Schedule

A **schedule** defines when backups should run using cron syntax. It's reusable across multiple backup jobs.

Examples:

- `0 2 * * *` - Daily at 2:00 AM
- `0 0 * * 0` - Weekly on Sunday
- `0 */6 * * *` - Every 6 hours

Backup Job

A **backup job** (or simply "backup") is a configuration that defines what to backup, where to store it, when to run, and retention policies.

Components:

- Source paths (what to backup)
- Repository (where to store)
- Schedule (when to run)
- Retention policy (how long to keep)

Director & Client Modes

Borgmatic Director UI supports two operating modes: **Standalone** (default) and **Director/Client** (for managing multiple backup servers from a central location).

Standalone Mode

Default mode - Each server runs Borgmatic Director UI independently. Perfect for single-server deployments.

Characteristics:

- Single server deployment
- No network communication required
- Simple setup and management
- Best for small deployments

Director/Client Mode

Centralized management - One Director server manages multiple Client servers remotely.

Characteristics:

- Central management dashboard
- Multiple client servers
- Secure WebSocket connections
- Best for enterprise deployments

Director Mode

The **Director** is the central management server that oversees multiple backup clients. It provides a unified dashboard to monitor and manage all connected clients.

Director Capabilities:

- **Unified Dashboard:** View backup status across all clients
- **Remote Sessions:** Switch between clients to view their data
- **Template Management:** Create and deploy backup configurations to multiple clients
- **Centralized Reporting:** Aggregate statistics and logs from all clients
- **Client Management:** Monitor, approve, and manage connected clients

Client Mode

Clients are backup servers that connect to a Director. They execute backups locally and report status back to the Director via secure WebSocket connections.

Client Capabilities:

- **Secure Connection:** Connects to Director via encrypted WebSocket (wss://)
- **Local Execution:** Runs backups on the client server
- **Status Reporting:** Sends backup status, logs, and statistics to Director
- **Configuration Receipt:** Accepts backup templates deployed from Director
- **Automatic Reconnection:** Reconnects automatically if connection is lost

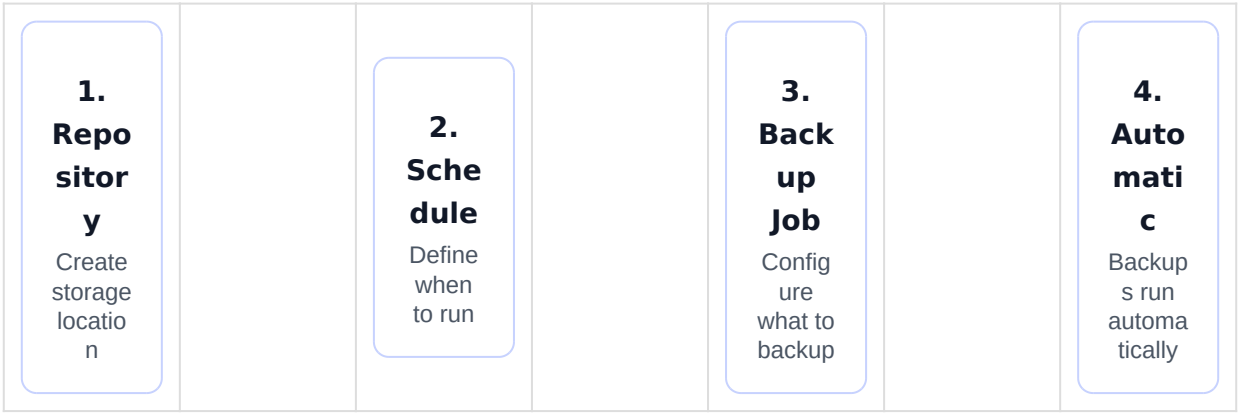
Security Architecture

Director/Client mode uses cryptographic authentication to ensure secure communication:

1. **Connection Token:** Client connects with a shared connection token
2. **Challenge-Response:** Director sends a cryptographic challenge
3. **Digital Signature:** Client signs challenge with private key (Ed25519)
4. **Verification:** Director verifies signature with client's public key
5. **Approval:** Connection is approved or rejected

Protection: Maximum 10 failed authentication attempts, then 1-hour lockout period to prevent brute-force attacks.

Typical Workflow



Revision #2
Created 17 January 2026 09:31:47 by Admin
Updated 17 January 2026 10:22:25 by Admin