Borgmatic Director UI

Borgmatic Director UI for beginners

Borgmatic Director UI - Beginner's Guide

> Welcome to Borgmatic UI! This guide will help you understand the basics of backing up your data using this powerful yet easy-to-use backup management interface.

Understanding Key Concepts

Before diving in, it's important to understand the terminology used throughout the application. These terms come from the underlying backup tools (Borg and Borgmatic) and understanding them will make everything much clearer.

?? Repository

What is it? A **Repository** is the *destination* where your backups are stored. Think of it as a secure vault or storage container that holds all your backup data.

Key points:

- A repository is created once and can hold many backups over time
- Repositories are encrypted by default - you'll set a passphrase when creating one
- You can have multiple repositories (e.g., one local, one remote)
- Repositories can be stored locally (on your server) or remotely (SSH server, cloud storage)

> **Important:** Never lose your repository passphrase! Without it, your backup data cannot be recovered.

? Backup (Backup Job)

What is it? A **Backup** (or Backup Job) is a *configuration* that defines:

- **What** to back up (which files and folders)
- **Where** to store it (which repository)
- **How** to back it up (compression, exclusions, etc.)
- **When** to run (if scheduled)

Example: You might create a backup job called "Website Backup" that backs up `/var/www` to your remote repository every night at 2 AM.

? Archive

What is it? An **Archive** is a single *snapshot* of your data at a specific point in time. Every time a backup job runs successfully, it creates a new archive.

Key points:

- Archives are stored inside repositories
- Each archive has a unique name (usually including a timestamp)
- Archives are deduplicated - only changed data is stored, saving space
- You can browse, restore, or delete individual archives

Analogy: If a Repository is a photo album, then Archives are individual photos. Each photo captures a moment in time, and the album holds them all together.

? Schedules

Schedules allow you to automate your backups so they run without manual intervention.

How Schedules Work

- Schedules are attached to backup jobs
- You can set backups to run hourly, daily, weekly, or with custom cron expressions
- Scheduled backups run automatically in the background
- You'll see the status and history of scheduled runs in the dashboard

Common Schedule Patterns

Pattern	Description	Best For
Daily at 2:00 AM	Runs once per day during low-activity hours	Most use cases

Pattern	Description	Best For
Every 6 hours	Runs 4 times per day	Frequently changing data
Weekly on Sunday	Runs once per week	Large, stable datasets

? SSH Keys

SSH Keys are used for secure, passwordless authentication when connecting to remote servers for backup storage.

Why Use SSH Keys?

- **Security:** More secure than passwords
- **Automation:** Required for scheduled backups to remote servers (no password prompts)
- **Convenience:** No need to enter passwords repeatedly

How to Set Up SSH Keys

1. Go to **SSH Key Management** in the sidebar
2. Either **Generate** a new key pair or **Import** an existing key
3. Copy the **public key** to your remote server's `~/.ssh/authorized_keys`
4. Test the connection to verify it works

> **Tip:** When importing a key, you can select a file from the server or upload from your computer using the buttons in the import dialog.

?? Storage Types

When creating a repository, you can choose from several storage types depending on where you want to store your backups.

Local Repository

Path format: `/path/to/repository`

- Stored on the same server or a mounted drive
- Fastest backup and restore speeds
- Best for: Quick backups, staging before cloud sync

SSH/SFTP Repository

Path format: `ssh://user@hostname/path/to/repository`

- Stored on a remote server via SSH
- Requires SSH key or password authentication
- Best for: Off-site backups, dedicated backup servers

Cloud Storage (with Rclone)

For cloud storage like Amazon S3, Backblaze B2, Google Drive, etc.

Option 1: Local + Cloud Sync (Recommended)

- Backups are stored locally first
- Automatically synced to cloud after each backup
- Faster backups, cloud redundancy

Option 2: Native Cloud (Borg 2.x only)

- Borg 2.x can write directly to S3-compatible storage
- No Rclone required
- Simplest cloud setup

? How to Restore (Retrieve) Backups

Restoring your data is just as important as backing it up. Here's how to retrieve files from your backups.

Step 1: Navigate to Archives

1. Go to **Archives** in the sidebar
2. Select the repository containing your backup
3. You'll see a list of all archives (snapshots) in that repository

Step 2: Browse the Archive

1. Click on an archive to open the **Archive Browser**
2. Navigate through folders just like a file explorer
3. Preview text files directly in the browser

Step 3: Restore Files

You have three options when restoring:

Option	Description	Use Case
Restore to New Location	Extract files to a folder you choose	Safest option - review before replacing
Download	Download files to your computer (folders are zipped)	Quick access, small files
Restore to Original Location	Put files back where they came from	Full restore after data loss

> **Caution:** "Restore to Original Location" will overwrite existing files. Use with care!

? Quick Start Guide

Ready to create your first backup? Follow these steps:

1. Create a Repository

1. Go to **Repositories** → **Create Repository**
2. Choose a storage type (Local is easiest to start)
3. Set a path (e.g., `/host/backups/my-repo`)
4. Enter a strong passphrase and save it somewhere safe!
5. Click **Create**

2. Create a Backup Job

1. Go to **Backups** → **Create Backup**
2. Give it a name (e.g., "My Documents")
3. Select your repository
4. Add source paths (what to back up)
5. Optionally set a schedule
6. Click **Create**

3. Run Your First Backup

1. Find your backup job in the list
2. Click the **Run** button
3. Watch the progress in the dashboard

4. Once complete, you'll have your first archive!

4. Verify Your Backup

1. Go to **Archives**
2. Select your repository
3. Click on the archive to browse its contents
4. Confirm your files are there

? Tips for Beginners

- **Start small:** Begin with a small folder to test the process
- **Use descriptive names:** Name your backups and repositories clearly
- **Test restores:** Regularly practice restoring files to ensure your backups work
- **Multiple destinations:** Consider both local and remote repositories for redundancy
- **Monitor the dashboard:** Check regularly that scheduled backups are running
- **Keep passphrases safe:** Store repository passphrases in a password manager

---

? Glossary

Term	Definition
Borg	The underlying backup program that handles deduplication and encryption
Borgmatic	A wrapper around Borg that simplifies configuration and automation
Deduplication	Only storing unique data chunks, saving significant storage space
Archive	A single backup snapshot at a point in time
Repository	The storage location containing all your archives
Passphrase	The password used to encrypt/decrypt your repository
Rclone	A tool for syncing files to cloud storage providers
SSH Key	A cryptographic key pair for secure, passwordless server authentication
Retention	Rules for how long to keep old archives before pruning them
Pruning	Removing old archives according to retention rules to save space

---

Congratulations! You now have a solid understanding of Borgmatic UI. Remember: a backup is only as good as its last successful restore test. Happy backing up! □□

Borgmatic Director UI for professionals

Borgmatic Director UI - Professional Administration Guide

> This guide is intended for system administrators and IT professionals who need comprehensive knowledge of the Borgmatic Director UI application, its operating modes, configuration options, and administrative features.

Operating Modes

Borgmatic Director UI supports three operating modes for different deployment scenarios. Understanding these modes is essential for proper deployment planning.

Mode Comparison Matrix

Feature	Standalone	Client	Director
Manage local backups	✅	✅	✅ (via client proxy)
Full local web UI	✅	✅	✅
Connect to Director	✅	✅	✅
Manage multiple machines	✅	✅	✅
Approve client connections	✅	✅	✅
Deploy templates to clients	✅	✅	✅
Aggregate monitoring dashboard	✅	✅	✅
Receive remote commands	✅	✅	✅

---

1. Standalone Mode (Default)

Standalone mode is the default operating mode. The instance operates independently, managing backups on a single server with no external connections.

Use Cases

- Single server deployments
- Home labs and personal servers
- Small businesses with one backup server
- Testing and development environments
- Air-gapped or isolated systems

Configuration

No additional configuration required. This is the default state after installation.

```
# Identity configuration file: /app/data/config/identity.json
{
  "mode": "standalone",
  "instance_id": "auto-generated-uuid",
  "instance_name": "My Backup Server"
}
```

Available Features

- Full backup job management (create, edit, delete, run)
- Repository management (local, SSH, S3, Rclone)
- Archive browsing and restore
- Schedule management
- SSH key management
- YAML configuration editor
- Real-time backup monitoring
- Notifications (email, webhooks, etc.)

---

2. Client Mode

Client mode allows the instance to be remotely managed by a central Director server while retaining full local functionality.

Use Cases

- Multiple servers managed by a central IT team
- Distributed infrastructure requiring unified management

- Managed Service Provider (MSP) client deployments
- Branch offices connected to headquarters

Configuration Steps

1. Navigate to **Settings → Operating Mode**
2. Click the toggle to switch from Standalone to Client mode
3. Configure connection settings:
   - **Client Name:** Human-readable identifier (e.g., "Production Web Server")
   - **Identification Phrase:** Secret phrase shown to Director admin for verification
   - **Director URL:** WebSocket URL (e.g., `wss://director.example.com`)
   - **Director Port:** Default is 9000
4. Save and initiate connection
5. Wait for Director admin to approve the connection request

Authentication Security

Client authentication uses Ed25519 cryptographic keypairs:

1. **Keypair Generation:** Client generates Ed25519 keypair on first setup
2. **Registration:** Client sends public key + identification phrase to Director
3. **Approval:** Director admin verifies phrase and approves connection
4. **Challenge-Response:** All subsequent connections authenticated via cryptographic challenge

```
Authentication Flow:
1. Client → Director: "Register me" + public key + phrase
2. Director Admin: Reviews request, verifies phrase
3. Director Admin: Clicks Approve (optionally locks IP)
4. Director → Client: Issues signed JWT token
5. Future connections:
   - Director: "Sign this random challenge: XYZ123"
   - Client: Signs with private key
   - Director: Verifies signature with stored public key
   - Connection established □
```

Connection Status Indicators

Status	Indicator	Meaning
Connected	□□ Green	Active connection to Director
Pending	□□ Yellow	Awaiting Director approval
Disconnected	□□ Red	No connection (network issue or Director offline)

Status	Indicator	Meaning
Rejected	□ Blocked	Connection rejected by Director admin

Client Mode Features

- All Standalone features remain available
- Receives template deployments from Director
- Reports status and backup results to Director
- Can be monitored remotely via Director dashboard
- Automatic reconnection with exponential backoff

---

3. Director Mode

Director mode transforms the instance into a central management hub that can monitor and control multiple Client instances.

Use Cases

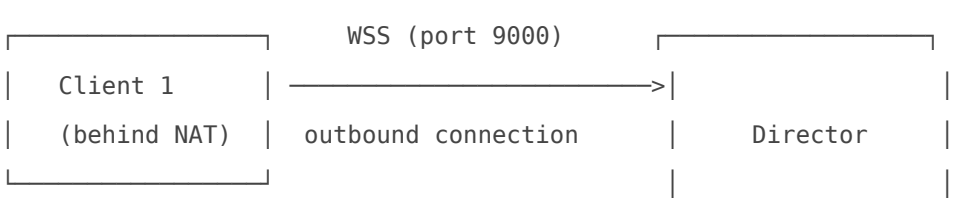
- Enterprise IT managing multiple servers
- Managed Service Providers (MSPs)
- DevOps teams managing infrastructure fleets
- Organizations requiring centralized backup oversight

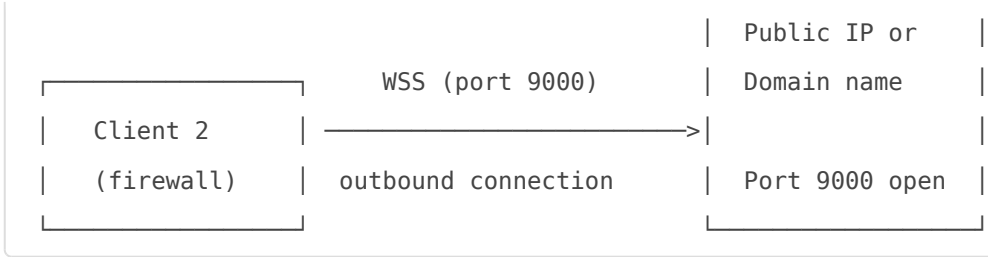
Configuration Steps

1. Navigate to **Settings → Operating Mode**
2. Click "Switch to Director Mode"
3. Type "switch" to confirm (this is a significant change)
4. Configure Director settings:
   - **Listen Port:** WebSocket port for client connections (default: 9000)
   - **SSL/TLS:** Enable for production (recommended)
   - **Auto-Approve:** Automatically approve new clients (not recommended)
5. Restart the server if prompted

> **Note:** Switching to Director mode enables HTTPS and may require server restart. Switching back to Standalone clears all client data.

Network Requirements





- Director needs a public IP or domain name
- Director port 9000 (or configured port) must be accessible
- Clients initiate outbound connections (NAT-friendly)
- No port forwarding required on client machines
- Can use CloudFlare Tunnel or reverse proxy if Director is also behind NAT

Client Management Dashboard

The Director dashboard provides:

- **Connected Clients:** Real-time list of online clients with status
- **Pending Approvals:** New clients awaiting approval
- **Offline Clients:** Previously connected clients that are currently unreachable
- **Aggregate Statistics:** Total backups, repositories, success/failure rates

Client Approval Workflow

1. New client connects and appears in "Pending Approvals"
2. Admin reviews:
   - Client name
   - Identification phrase (verify this matches expected)
   - IP address
   - Public key fingerprint
3. Admin decides:
   - **Approve:** Client can connect and be managed
   - **Approve + Lock IP:** Only allow connections from current IP
   - **Reject:** Deny connection permanently

Managing Remote Clients

When in Director mode, a client selector appears in the navigation:

- Select a client to view/manage that client's backups, repositories, etc.
- All pages (Backups, Repositories, Archives, etc.) show data from selected client
- Actions performed affect the selected client
- "View All Clients" returns to the aggregate dashboard

Template Deployment

Directors can create and deploy templates to multiple clients:

- **Backup Templates:** Pre-configured backup job definitions
- **Schedule Templates:** Standardized backup schedules
- **Repository Templates:** Common repository configurations

Deployment options:

- Deploy to selected clients
- Deploy to client groups
- Deployed items are created as **inactive** for security (must be activated locally)

Mode Switching Reference

From	To	Data Impact	Action Required
Standalone	Client	None (non-destructive)	Toggle switch + configure Director URL
Client	Standalone	None (non-destructive)	Toggle switch
Standalone/Client	Director	Enables HTTPS, adds Director features	Type "switch" to confirm + restart
Director	Standalone	Clears all client connection data	Type "switch" to confirm

---

Dashboard

The Dashboard provides an at-a-glance overview of backup system health and recent activity.

Dashboard Widgets

System Status

- **Backup Tools Health:** Shows installed status and versions of Borg 1.x, Borg 2.x, Borgmatic, and Rclone
- **Last Health Check:** Timestamp of most recent tool verification

Backup Statistics

- Total backup jobs configured
- Active vs. inactive backups
- Last 24-hour success/failure count
- Currently running backups

Recent Activity

- List of recent backup runs with status
- Click to view detailed logs
- Filter by status (success, failed, running)

Repository Overview

- Total repositories configured
- Storage usage (if available)
- Repository health status

Backup Jobs

Backup Jobs define what data to back up, where to store it, and when to run.

Creating a Backup Job

Basic Settings

Field	Description	Required
Name	Human-readable identifier for the backup	Yes
Description	Optional notes about the backup purpose	No
Repository	Target repository for storing backups	Yes
Borg Version	Use Borg 1.x or Borg 2.x (must match repository)	Yes
Active	Enable/disable the backup job	Yes

Source Configuration

Field	Description	Example
Source Directories	Paths to back up	`/host/home`, `/host/var/www`
Exclude Patterns	Glob patterns to exclude	`*.tmp`, `**/node_modules/**`
Exclude If Present	Skip directories containing these files	`.nobackup`, `CACHEDIR.TAG`

Advanced Options

Option	Description	Default
Compression	Compression algorithm and level	`zstd,3`
One File System	Don't cross filesystem boundaries	Enabled

Option	Description	Default
Read Special	Read special files (devices, FIFOs)	Disabled
Numeric Owner	Store numeric user/group IDs	Disabled
No Atime	Don't store access time	Enabled

Retention Policy

Configure how many archives to keep:

Setting	Description	Recommended
Keep Hourly	Number of hourly archives	24
Keep Daily	Number of daily archives	7
Keep Weekly	Number of weekly archives	4
Keep Monthly	Number of monthly archives	6
Keep Yearly	Number of yearly archives	2

Hooks

Execute commands before/after backups:

Hook	When Executed	Use Case
Before Backup	Before archive creation starts	Database dumps, stop services
After Backup	After successful backup	Cleanup temp files, start services
On Error	When backup fails	Send alerts, cleanup

Running Backups

- **Manual Run:** Click the "Run" button on any backup job
- **Scheduled Run:** Automatically triggered by configured schedule
- **Stop Running:** Click "Stop" to abort a running backup

Backup Status

Status	Indicator	Meaning
Idle	○ Gray	Not currently running
Running	◫ Blue (spinner)	Backup in progress
Success	◫ Green	Last run completed successfully
Failed	◫ Red	Last run failed

Status	Indicator	Meaning
Inactive	● Disabled	Backup job is disabled

Repositories

Repositories are the storage destinations for backup archives.

Repository Types

Local Repository

- **Path format:** `/path/to/repository`
- **Use case:** Local disk, mounted NAS, attached storage
- **Performance:** Fastest backup/restore speeds
- **Configuration:** Just specify the path

SSH/SFTP Repository

- **Path format:** `ssh://user@hostname:port/path`
- **Use case:** Remote backup servers, dedicated storage boxes
- **Authentication:** SSH key (recommended) or password
- **Configuration:**
  - Host, port, username
  - SSH key selection or password
  - Remote path (browse with file explorer)

S3/Cloud Storage (Rclone)

- **Providers:** Amazon S3, Backblaze B2, Wasabi, MinIO, Google Cloud, etc.
- **Storage Modes:**
  - **Local + Cloud Sync:** Fast local backups, automatic cloud sync
  - **Native Cloud (Borg 2.x):** Direct S3 writes, no Rclone needed
- **Configuration:**
  - Select Rclone remote (pre-configured in Rclone)
  - Specify bucket/path
  - Choose storage mode

Creating a Repository

1. Navigate to **Repositories → Create Repository**
2. Select repository type (Local, SSH, S3/Rclone)
3. Configure type-specific settings
4. Set Borg version (1.x or 2.x)

5. Enter encryption passphrase
6. Click **Create** to initialize the repository

> **Critical:** Save your repository passphrase securely! Without it, your backups cannot be accessed or restored.

Repository Actions

Action	Description	When to Use
Check	Verify repository integrity	Periodically or after errors
Compact (Borg 2.x)	Reclaim space from deleted archives	After pruning many archives
Update Passphrase	Change stored passphrase	If passphrase was entered incorrectly
Delete	Remove repository from UI	Optionally delete data on disk

---

Archives (View/Restore)

Archives are point-in-time snapshots stored in repositories.

Archive Browser

The Archive Browser provides file-system navigation of backup contents:

- **Directory Navigation:** Click folders to explore
- **Breadcrumb Trail:** Navigate back to parent directories
- **Search/Filter:** Filter files by name
- **File Preview:** View text file contents directly
- **Selection:** Check items for restore/download

Restore Options

Option	Description	Best For
Restore to New Location	Extract to a specified folder	Safest - review before replacing
Download	Download to your browser (ZIP for folders)	Small files, quick access
Restore to Original Location	Put files back where they were	Full disaster recovery

Restore History

Each archive tracks its last restore operation:

- Destination path or "Downloaded"
- Timestamp of restore
- Persisted across sessions

Archive Actions

Action	Description
Browse	Open archive in file browser
Info	View archive metadata (size, file count, etc.)
Delete	Permanently remove archive from repository

Schedules

Schedules automate backup execution at specified intervals.

Schedule Configuration

Field	Description	Example
Schedule Type	Preset or custom cron	Daily, Weekly, Custom
Time	When to run	02:00
Day (weekly)	Day of week for weekly schedules	Sunday
Cron Expression	Custom cron for advanced scheduling	`0 */6 * * *`
Timezone	Timezone for schedule evaluation	Europe/Berlin

Cron Expression Reference

```
# Format: minute hour day-of-month month day-of-week
#         0-59   0-23 1-31         1-12  0-6 (0=Sunday)

0 2 * * *      # Daily at 2:00 AM
0 */6 * * *    # Every 6 hours
0 3 * * 0      # Weekly on Sunday at 3:00 AM
0 4 1 * *      # Monthly on the 1st at 4:00 AM
*/30 * * * *   # Every 30 minutes
```

Common Schedule Patterns

Pattern	Cron	Use Case
Daily at 2 AM	`0 2 * * *`	Standard daily backup
Every 6 hours	`0 */6 * * *`	Frequently changing data
Weekdays at 6 PM	`0 18 * * 1-5`	End of business day
Sunday at 3 AM	`0 3 * * 0`	Weekly full backup

SSH Key Management

SSH keys enable secure, passwordless authentication to remote servers.

Key Operations

Import Existing Key

1. Click **Import SSH Key**
2. Provide a name for the key
3. Paste the private key content, OR
4. Click **Select from Server** to browse server filesystem, OR
5. Click **Upload Key File** to upload from your computer
6. If key is encrypted, enter the passphrase
7. Click **Create**

Generate New Key

1. Click **Generate SSH Key**
2. Provide a name
3. Select key type (Ed25519 recommended, RSA for compatibility)
4. Click **Generate**
5. Copy the public key to remote server's `~/.ssh/authorized_keys`

Test Connection

1. Click the test icon on a key
2. Enter remote host, username, and port
3. Click **Test Connection**
4. Verify connection succeeds and Borg is detected

Supported Key Types

Type	Format Header	Recommendation
OpenSSH	`-----BEGIN OPENSSH PRIVATE KEY-----`	✅ Recommended (modern)
RSA PEM	`-----BEGIN RSA PRIVATE KEY-----`	✅ Good (legacy compatible)
EC PEM	`-----BEGIN EC PRIVATE KEY-----`	✅ Good
PKCS#8	`-----BEGIN PRIVATE KEY-----`	✅ Good

Key Security

- Private keys are stored encrypted in the database
- Passphrase-protected keys are supported
- Temporary key files are created with mode 0600
- Temporary files are cleaned up immediately after use

YAML Editor

The YAML Editor provides direct access to Borgmatic configuration files.

Features

- **Syntax Highlighting:** YAML-aware editor with color coding
- **Validation:** Real-time syntax and schema validation
- **Backup/Restore:** Automatic backups of config changes
- **File Browser:** View all configuration files in `/etc/borgmatic.d/`

Use Cases

- Advanced configuration not exposed in UI
- Bulk editing of multiple options
- Importing existing Borgmatic configs
- Troubleshooting configuration issues

> **Caution:** Manual YAML edits may conflict with UI-managed settings. Use the UI for standard configurations.

Logs

The Logs page provides access to backup execution logs and system events.

Log Types

Log Type	Content	Location
Backup Logs	Individual backup execution output	Per-backup log files
Borgmatic Logs	Borgmatic wrapper output	System logs
Application Logs	Borgmatic UI application events	Container stdout/stderr

Log Features

- **Real-time Streaming:** Watch backup progress live
- **Search:** Find specific entries
- **Filter by Level:** Info, Warning, Error
- **Download:** Export logs for analysis

Settings

General Settings

Setting	Description
Instance Name	Display name for this server
Theme	Light or dark mode
Timezone	Default timezone for schedules

Operating Mode

See the Operating Modes section above.

Notifications

Configure alerts for backup events:

Provider	Configuration
Email (SMTP)	SMTP server, credentials, recipients

Provider	Configuration
Webhook	URL to POST events to
Slack	Webhook URL
Discord	Webhook URL
Gotify	Server URL and token
Ntfy	Topic and server URL

Security Settings

Setting	Description
Change Password	Update admin password
Session Timeout	Auto-logout after inactivity
API Tokens	Generate tokens for API access

Factory Reset

Completely reset the instance:

- Removes all configurations
- Clears all credentials
- Optionally regenerates secret key
- Requires typing "RESET" to confirm

> **Warning:** Factory reset is irreversible. Backup your data first!

Environment Variables

Configure the application via environment variables:

Variable	Default	Description
`NODE_ENV`	production	Environment mode (production/development)
`PORT`	3000	HTTP server port
`JWT_SECRET`	(auto-generated)	Secret for JWT token signing
`ADMIN_USERNAME`	admin	Default admin username
`ADMIN_PASSWORD`	admin	Default admin password

Variable	Default	Description
`DATA_DIR`	/app/data	Persistent data directory
`TZ`	UTC	Container timezone
`DIRECTOR_PORT`	9000	WebSocket port (Director mode)
`RESTORE_ALLOWED_ROOTS`	/host,/tmp,/data	Allowed restore destination paths
`DEBUG_REPOSITORIES`	false	Enable verbose repository logging

Docker Deployment

Basic Docker Compose

```
version: '3.8'

services:
  borgmatic-ui:
    image: borgmatic-ui:latest
    container_name: borgmatic-ui
    restart: unless-stopped
    ports:
      - "8080:3000"      # Web UI
      - "9000:9000"      # Director WebSocket (if using Director mode)
    volumes:
      - ./data:/app/data                    # Persistent data
      - ./borgmatic.d:/etc/borgmatic.d      # Borgmatic configs
      - /:/host:ro                          # Host filesystem access
      - ./borg-cache:/root/.cache/borg      # Borg cache
    environment:
      - ADMIN_PASSWORD=change-me-please
      - TZ=Europe/Berlin
```

Volume Mounts Explained

Mount	Purpose	Required
`/app/data`	Application data, credentials, SSH keys	Yes
`/etc/borgmatic.d`	Generated Borgmatic YAML configs	Yes

Mount	Purpose	Required
`/:/host`	Access to host filesystem for backups	Yes (for local backups)
`/root/.cache/borg`	Borg cache (improves backup speed)	Recommended

Troubleshooting

Common Issues

Problem	Cause	Solution
"Cannot connect to server"	Backend not running or port blocked	Check container logs, verify port mapping
"Authentication failed" (SSH)	Wrong key or not in authorized_keys	Verify public key on remote server
"Passphrase wrong"	Incorrect repository passphrase	Update passphrase in repository settings
"Repository locked"	Previous backup didn't finish	Wait or use "Break Lock" action
Backup stuck at 0%	Network issue or SSH timeout	Check connectivity, test SSH connection
Director: Client not connecting	Firewall, wrong URL, or DNS issue	Verify port 9000 accessible, check client logs

Debug Mode

```
# Enable verbose logging
docker run -e DEBUG_REPOSITORIES=true ...

# View container logs
docker logs -f borgmatic-ui

# Check real-time backup output
# (visible in the UI during backup execution)
```

API Access

Borgmatic Director UI provides a REST API for automation and integration.

Authentication

```
# Login to get JWT token
POST /api/auth/login
Content-Type: application/json
{
  "username": "admin",
  "password": "your-password"
}

# Use token in subsequent requests
Authorization: Bearer <token>
```

Key Endpoints

```
# Backups
GET  /api/backups              # List all backup jobs
POST /api/backups/:id/run      # Trigger backup
POST /api/backups/:id/stop     # Stop running backup

# Repositories
GET  /api/repositories         # List repositories
POST /api/repositories/:id/check  # Check integrity

# Archives
GET  /api/archives/:repo       # List archives
GET  /api/archives/:repo/:archive/browse  # Browse contents

# Health
GET  /api/dashboard/health     # System health
GET  /api/dashboard/tools-health  # Tool versions
```

> **Need more help?** Check the container logs (`docker logs borgmatic-ui`) for detailed error messages. Most issues are related to SSH connectivity, passphrases, or file permissions.

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

1. Repository		2. Schedule		3. Backup Job		4. Automatic
Create storage location		Define when to run		Configure what to backup		Backups run automatically

Repository Guide

Repository Types & Performance Guide

Understand the differences between repository types and choose the best option for your needs

About Repositories in Borgmatic Director UI

A **repository** is where your backups are stored. Borgmatic Director UI supports multiple repository types, each with different performance characteristics. The choice of repository type significantly impacts backup speed, resource usage, and reliability. Understanding these differences helps you make informed decisions for your backup strategy.

Direct Mode

Borg writes directly to the remote storage. This is the fastest method as there's no intermediate step.

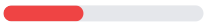
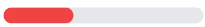
Sync Mode

Borg writes locally first, then a sync tool (like Rclone) copies to cloud. This adds overhead and delay.

Performance Comparison

REPOSITORY TYPE	SPEED RATING	STORAGE MODE	BEST USE CASE

	Local Filesystem Direct filesystem access on the same machine or fast local network	Fastest 100%	Direct	Same machine backups, fast local network storage ⚠ **Not recommended due to potential data loss in case of disk crash or malicious attacks**
	SSH (Native Borg) Borg's native SSH protocol with optimized deduplication and compression	Very Fast 90%	Direct	Remote servers with Borg installed, production backups
	S3 Direct (Native) Borg's native S3 support using boto3, optimized for cloud object storage	Fast 75%	Direct	Cloud storage (AWS, Hetzner, Wasabi, Backblaze B2, MinIO)
	SFTP SSH-based file transfer protocol, works without Borg on remote	Moderate 60%	Direct	Remote servers without Borg installed

Rclone Direct (Mounted) Rclone FUSE mount for 100+ cloud providers	Moderate-Slow  50%	Direct (FUSE)	Cloud providers not natively supported (Google Drive, Dropbox, etc.)
Network Mounts (NFS/SMB) Network filesystems like NFS, SMB/CIFS	Slow  45%	Direct (mount)	Existing network storage infrastructure
S3 Sync Mode Write locally first, then sync to S3 using Rclone	Slow  40%	Sync	When S3 direct mode is not available
Rclone Sync Mode Write locally first, then sync to cloud	Slowest  35%	Sync	When direct mounting is not possible

Detailed Explanations

Local Filesystem

Direct filesystem access on the same machine or fast local network

? Advantages

- Fastest performance
- No network overhead
- Lowest latency
- Simple setup

? Limitations

- Requires local storage
- No off-site backup
- Vulnerable to local disasters

SSH (Native Borg)

Borg's native SSH protocol with optimized deduplication and compression

? Advantages

- Highly optimized
- Native Borg protocol
- Efficient deduplication
- Secure

? Limitations

- Requires Borg on remote server
- Needs SSH access

S3 Direct (Native)

Borg's native S3 support using boto3, optimized for cloud object storage

? Advantages

- Native S3 support
- Optimized for cloud
- Scalable
- Works with many providers

? Limitations

- Network latency
- S3 API overhead
- Costs per request

SFTP

SSH-based file transfer protocol, slower than native SSH but works without Borg on remote

? Advantages

- Works without Borg on remote
- Secure
- Standard protocol

? Limitations

- Protocol overhead
- Slower than native SSH
- Less optimized

Quick Decision Guide

Choose SSH (Native) if:

- You have a remote server with Borg installed

- You want the fastest remote backup performance
- You need production-grade reliability
- You have SSH access to the remote server

Choose S3 Direct if:

- You're using cloud object storage (AWS, Hetzner, Wasabi, etc.)
- You want native cloud integration
- You need scalable storage

Choose Rclone Direct (Mounted) if:

- You need providers not natively supported (Google Drive, Dropbox, etc.)
- You can accept moderate performance
- You need unified access to multiple providers

Avoid Sync Mode if possible:

- It's the slowest option due to double write overhead
- Requires local storage space
- Adds complexity and potential failure points

Performance Tips

- **SSH is fastest:** If you have a remote server, SSH (native Borg) is almost always the fastest option
- **Direct beats Sync:** Always prefer direct mode over sync mode when possible
- **Avoid double writes:** Sync mode writes data twice (locally + cloud), significantly slowing backups
- **Compression helps:** Enable compression (LZ4) to reduce data transfer over network
- **Deduplication is key:** Borg's deduplication works best with direct protocols like SSH

Introduction Help

Intro Help & Documentation

Learn how to use Borgmatic Director UI to manage your backups

Quick Links

Overview | Operating Modes | Connecting Client | Repositories | Backup Jobs | Schedules | Restore | Settings

Overview - What is Borgmatic UI?

Borgmatic UI is a web-based interface for managing{' '} BorgBackup{' '} through{' '} Borgmatic. It provides an intuitive interface for configuring backups, managing repositories, scheduling automated backups, and restoring files.

Key Features

- Create and manage backup jobs with an easy-to-use wizard
- Configure repositories (local, SSH, SFTP, S3, Rclone)
- Schedule automated backups with cron-like scheduling
- Browse and restore files from any backup archive
- Multi-node management with Director mode
- Real-time backup monitoring and logging

Operating Modes

Borgmatic UI can operate in two main modes, depending on your needs:

Client / Standalone Mode

Run Borgmatic UI on a single machine to manage its local backups. This is the default mode and is perfect for individual servers or workstations.

Use when:

- You have a single server to back up
- You want a simple, self-contained backup solution
- You don't need centralized management

Note: A standalone instance can later connect to a Director to become a managed client.

Director Mode

A central management server that can monitor and control multiple client instances. The Director doesn't run backups itself – it manages others.

Use when:

- You have multiple servers to back up
- You want centralized monitoring and control
- You need to manage backups across your infrastructure

Features: Client management, templates, deployments, centralized passphrase vault.

Choosing Your Mode

You choose the operating mode during initial setup. The mode can be changed later in **Settings ? Operating Mode**, but this will reset mode-specific configurations.

Connecting a Client to a Director

To connect a Client (or Standalone instance) to a Director for centralized management:

1

On the Director: Get Connection Details

In the Director's **Settings ? Connection Configuration**, find or create a **Connection Token**. Also note the Director's URL (e.g., `https://director.example.com:8000`).

2

On the Client: Configure Connection

Go to **Settings ? Client Configuration**. Enter:

- **Client Name:** A friendly name for this client (e.g., "Web Server 1")
- **Director URL:** The full URL including port (use `https://`)
- **Connection Token:** The token from the Director (leave empty if open access)

3

Save and Connect

Click **Save Configuration**, then **Test Connection** to verify, and finally **Connect** to establish the connection.

Connected!

Once connected, the client will appear in the Director's dashboard and can be managed remotely. The connection uses WebSocket over SSL for real-time communication.

Troubleshooting Connection Issues

- **Connection refused:** Ensure the Director is running and the URL is correct (including https://)
- **Invalid token:** Double-check the connection token matches exactly
- **SSL errors:** Director mode automatically uses HTTPS – make sure to use https:// in the URL
- **Firewall:** Ensure port 8000 (default) is open between client and Director

Switching Between Clients (Director Mode)

When logged into a Director, you can switch between viewing the Director's own interface and any connected client's interface.

Using the Client Selector

In the top navigation bar, you'll see a dropdown showing **"Director"** or the currently selected client's name. Click it to:

- Switch to **Director** - View the Director dashboard with all connected clients
- Switch to a **Client** - View that client's backups, schedules, and settings as if you were logged in directly

	Director View
	• See all connected clients • Manage templates and deployments • Configure the vault for passphrases • View aggregated statistics

	Client View
	• Manage backup jobs • View/restore archives • Configure schedules • Edit repositories

Quick Access from Dashboard

In the Director Dashboard, each connected client has a **"View"** button. Clicking it switches to that client's view and highlights the dropdown selector so you know where to switch back.

Setting Up Repositories

A **repository** is where your backup data is stored. Before creating backup jobs, you need to set up at least one repository.

Supported Repository Types

Local
Backup to a local directory or mounted drive

SSH/SFTP
Backup to a remote server via SSH

Borg Server
Connect to a dedicated Borg backup server

S3/MinIO
Amazon S3 or compatible object storage

Rclone
Use Rclone for cloud storage (Google Drive, Dropbox, etc.)

Creating a Repository

1. Go to **Repositories** in the sidebar
2. Click **Add Repository**
3. Select the repository type and fill in the details:
   - **Name:** A friendly name for reference
   - **Path/URL:** Where the repository will be stored
   - **Encryption:** Choose encryption mode (repokey-blake2 recommended)
   - **Passphrase:** A strong password to encrypt your backups
4. Click **Create & Initialize** to create the repository

Important: Save Your Passphrase!
The repository passphrase is required to access your backups. If you lose it, your backup data will be **permanently inaccessible**. Store it securely (e.g., in a password manager).

Creating Backup Jobs

A **backup job** defines what to back up and where. Each job specifies source directories, a target repository, and optional exclusions.

Creating a Backup Job

1. Go to **Backup Jobs** in the sidebar
2. Click **Create Backup Job**
3. Fill in the backup configuration:
   - **Job Name:** A descriptive name (e.g., "Daily Website Backup")

- **Source Directories:** Paths to back up (e.g., /var/www, /home)
- **Repository:** Select a configured repository
- **Exclusions:** Patterns to exclude (e.g., *.log, node_modules/)

4. Click **Save** to create the job

Running Backups

	Manual Backup Click the **Run Now** button on any backup job to start an immediate backup. Progress is shown in real-time.

	Scheduled Backup Set up a schedule (see next section) to run backups automatically at specified times.

Backup Archives

Each time a backup runs, it creates an **archive** – a point-in-time snapshot. Borg uses deduplication, so subsequent backups only store changed data, making them fast and space-efficient.

Setting Up Schedules

Schedules automate your backups by running them at specified times. You can create multiple schedules for different backup jobs.

Creating a Schedule

1. Go to **Schedules** in the sidebar
2. Click **Add Schedule**
3. Configure the schedule:
   - **Backup Job:** Select which job to run
   - **Frequency:** Daily, Weekly, Monthly, or Custom (cron)

- **Time:** When to run the backup
- **Days:** For weekly schedules, select which days

4. Click **Save** to activate the schedule

Schedule Examples

Daily at 2:00 AM	`0 2 * * *`
Every Sunday at midnight	`0 0 * * 0`
Every 6 hours	`0 */6 * * *`
First of every month at 3 AM	`0 3 1 * *`

Best Practices

- Run backups during low-activity periods (e.g., night)
- Stagger multiple backup jobs to avoid resource contention
- For critical data, consider running backups multiple times per day
- Monitor backup logs to ensure schedules are running successfully

Viewing and Restoring Archives

The **View/Restore** page lets you browse your backup archives and restore files when needed.

Browsing Archives

1. Go to **View/Restore** in the sidebar
2. You'll see a list of your repositories
3. Click on a repository to expand it and see archives grouped by backup job
4. Each archive shows:
   - Creation date and time

- Size (original and deduplicated)
- Number of files

Restoring Files

1	**Select an Archive** Click the **View** button on the archive you want to restore from.
2	**Browse Files** Navigate through the archive's directory structure to find the files you need.
3	**Select Files to Restore** Check the files or directories you want to restore. You can select individual files or entire folders.
4	**Choose Restore Location** Specify where to restore the files: • **Original location:** Restore files to their original paths • **Custom location:** Restore to a different directory
5	**Start Restore** Click **Restore** and wait for the process to complete. Progress is shown in real-time.

Restore Tips

- Restoring to the original location will overwrite existing files
- For safety, consider restoring to a temporary location first
- Large restores may take time – don't close the browser window
- Check file permissions after restore if needed

Settings Overview

The **Settings** page contains various configuration options depending on your operating mode.

Operating Mode

View and change your current mode (Client/Standalone or Director). Changing modes will reset mode-specific configurations.

Client Configuration

Configure connection to a Director server. Set client name, Director URL, and connection token.

Connection Configuration

(Director only) Manage connection tokens and security settings for client connections.

Domain & Security

Configure custom domains, SSL certificates, and security settings for your installation.

Vault

(Director only) Securely store and manage repository passphrases for connected clients.

System Settings

Configure backup timeouts, concurrent backup limits, log retention, and other system-wide settings.

User Management

(Admin only) Create and manage user accounts, reset passwords, and configure permissions.