HyperSDK

Quick Start & Proof of Concept

From zero to your first VM migration in under 30 minutes. Clone, build, configure, and export — everything you need to evaluate HyperSDK.

30-Minute Setup — No External Dependencies — Production Ready

Prerequisites

What you need before starting your HyperSDK proof of concept.

Go 1.24+

HyperSDK is built with Go 1.24. Install from `go.dev` or your distribution's package manager. Verify with `go version`.

Linux Host

Any modern Linux distribution: RHEL/CentOS 8+, Ubuntu 20.04+, Fedora 38+, Debian 11+. Runs on bare metal or in a VM. 2 GB RAM minimum.

Provider Credentials

Credentials for at least one source provider: vCenter admin account, AWS access keys, Azure service principal, GCP service account, or Proxmox API token.

Storage Space

Sufficient local disk or cloud storage for exported VM images. Plan for 1.5x the provisioned disk size of your target VMs (compression reduces final size).

Optional (Recommended)

Component	Purpose	Required?
systemd	Run HyperSDK daemon as a managed service	Recommended
Web browser	Access the dashboard UI at http://localhost:8080	Recommended

Cloud CLI (aws, az, gcloud)	Verify cloud credentials and storage buckets	Optional
make	Build automation (make build, make install)	Recommended
git	Clone the repository	Required

> *"Most teams have everything they need already installed. The only new dependency is Go 1.24 — everything else is standard Linux tooling."*

Installation Steps

Four commands from source code to running daemon.

1. **Clone the Repository**
   Get the latest source code from the HyperSDK repository.

2. **Build the Binaries**
   Compile the daemon (hypervisord) and CLI (hyperctl) with a single command.

3. **Install System-Wide**
   Copy binaries to /usr/local/bin and install the systemd service unit.

4. **Start the Daemon**
   Enable and start the hypervisord service. Dashboard immediately available.

```
# Step 1: Clone
git clone https://github.com/hypersdk/hypersdk.git
cd hypersdk

# Step 2: Build
make build

# Step 3: Install (requires sudo)
sudo make install

# Step 4: Start the daemon
sudo systemctl enable --now hypervisord

# Verify it's running
systemctl status hypervisord
hyperctl version
```

4	~2 min	0
Commands to Full Install	Build Time (Go 1.24)	External Dependencies

After installation, open `http://localhost:8080` in your browser. The dashboard is served by the daemon — no separate web server needed.

First Migration

Configure a provider, browse VMs, export, and monitor — all from the web UI.

1. **Configure vCenter (or Any Provider)**

   Navigate to `/providers-ui`. Enter your vCenter URL, username, and password. Click "Test Connection" to verify. Click "Save" to register the provider.

2. **Browse VMs at /vms**

   Navigate to `/vms`. All discovered VMs appear in a searchable, paginated table. Filter by provider, search by name, and identify your target VMs.

3. **Export via /export-ui**

   Click "Export" on any VM or navigate to `/export-ui`. Select the source VM, choose output format (QCOW2, VMDK, VHD, RAW), set the destination path, and click "Start Export".

4. **Monitor at /jobs-ui**

   Navigate to `/jobs-ui`. Watch real-time progress with percentage, bytes transferred, throughput, and ETA. The job auto-completes and generates an export manifest.

CLI Alternative

Prefer the command line? Use `hyperctl provider add vsphere` to configure, `hyperctl vm list` to browse, and `hyperctl export` to migrate. Same API, different interface.

Export Manifest

Every export produces a manifest file with checksums, VM metadata, hardware config, and timestamps. Use it for validation, auditing, or automated import into the target platform.

Ready to Start?

Your first VM export takes less than 30 minutes from clone to completion. No licensing, no registration, no vendor lock-in. Just clone, build, and migrate.