

1 Bosh CheatSheet

VMWARE

- PDF Link: [cheatsheet-bosh-A4.pdf](#), Category: vmware
- Blog URL: <https://cheatsheet.dennyzhang.com/cheatsheet-bosh-A4>
- Related posts: PKS CheatSheet, Tile CheatSheet, Chef CheatSheet, #denny-cheatsheets

File me Issues or star this repo.

<https://raw.githubusercontent.com/dennyzhang/cheatsheet-bosh-A4/master/bosh.jpg>

1.1 Bosh Frequent-Used

Name	Command
List tasks	<code>bosh tasks -r=10, bosh -d \$deployment tasks -ar</code>
Tail latest bosh task	<code>bosh task -a, bosh task -a --debug</code>
List deployments and vms	<code>bosh deployments, bosh -d \$deployment vms</code>
Bosh get current task output	<code>bosh -d \$deployment task \$id --result</code>
Bosh get task all detail log	<code>bosh -d \$deployment task -a --debug, bosh -d \$deployment task \$id --debug</code>
Delete a deployment	<code>bosh delete-deployment -d \$deployment, bosh delete-deployment -d \$deployment</code>
Upload release	<code>bosh -n upload-release, bosh -n upload-release /tmp/my-release-0.22.0.tgz</code>
Create bosh release tar from source code	<code>bosh-create-release.md</code>
Get deployment manifest yaml	<code>bosh -d \$deployment manifest > my.yaml</code>
Run bosh deployment from yaml	<code>bosh -d \$deployment deploy my.yaml</code>
Run errand, and avoid deleting the vms	<code>bosh -n -d \$deployment run-errand \$errand_name --keep-alive</code>
Delete all or a given bosh deployment	<code>cleanup-bosh-deployment.sh</code>
Delete all or a given bosh release	<code>cleanup-bosh-releases.sh</code>
Common Bosh Releases	kubo-release, cfc-etc-release, docker-boshrelease, pxc-release, YouTube: BOSH Releases
Reference	Link: terminology, GitHub: cloudfoundry/bosh, CheatSheet: Cloudfoundry Tile & Ops
Reference	PKS CheatSheet, Bosh CheatSheet, Tile CheatSheet, UAA CheatSheet

1.2 Bosh Release

Name	Command
List release	<code>bosh -n releases</code>
Release folder	<code>/var/tempest/releases</code>
Create a local release	<code>bosh create-release --tarball=myrelease-0.4.0.tgz</code>
Export a release	<code>bosh -d \$deployment export-release <my-release>/<release-version> ubuntu-xenial/170.9</code>
Inspect a release	<code>bosh inspect-release <my-release>/<release-version></code>
Delete release	<code>bosh -n delete-release <my-release>/<release-version> --force</code>
Upload release	<code>bosh -n upload-release, bosh -n upload-release /tmp/my-release-0.22.0.tgz</code>
Reset release	<code>bosh reset-release</code>
Reference	Link: Release creation

1.3 Bosh Basic

Name	Command
bosh	<code>apt install bosh-cli, /var/vcap/bosh/log, /var/vcap/sys/log</code>
List all bosh deployments	<code>bosh -n deployments, bosh -n deployments --json</code>
Download log files	<code>bosh -d \$deployment_id logs</code>
Show deployment manifest	<code>bosh manifest -d \$deployment</code>
Show current cloud config	<code>bosh cloud-config, bosh cc, bosh configs</code>
Show current CPI config	<code>bosh cpi-config</code>
List environment	<code>bosh env, bosh env --json, bosh envs</code>
List all instances	<code>bosh instances</code>
Get vms per deployment	<code>bosh -e \$env -d \$deployment vms, bosh instances</code>
Cloud consistency check	<code>bosh cloud-check -d \$deployment</code>
List disks	<code>bosh disks -o</code>

1.4 Bosh Interpolate

Name	Command
Bosh Interpolates variables	<code>bosh int /tmp/build/4dcXX/pks-lock/metadata --path=/ops_manager/username</code>
Bosh use a property within a template	<code><%= p('some-property') %></code>
Bosh use a property with default value	<code><%= p('cluster-name', <%= spec.deployment %>) %></code>
Bosh use a property conditionally	<code>bosh-if-property.md</code>
Parse json output	<code>bosh int --path=/uuid -, similar to jq command</code>

1.5 Bosh Manifest

Name	Command
Show bosh manifest	<code>bosh manifest -d \$deployment</code>
Run bosh deployment from manifest	<code>bosh -d \$deployment deploy my.yaml</code>

1.6 Bosh VM

Name	Command
List vms	<code>bosh -d \$deployment vms</code>
List vms with verbose info	<code>bosh -d \$deployment vms --vitals</code>
Delete a vm	<code>bosh -d \$deployment delete-vm <vm-cid></code>
SSH to vm	<code>bosh -d \$deployment ssh \$vm_name/\$GUID</code>
Run SSH commands	<code>bosh -d \$deployment ssh \$vm_name>/\$GUID -c "sudo cat /etc/hosts"</code>
bosh scp	<code>bosh -d \$deployment scp /tmp/myfile \$vm_name/\$GUID:/tmp/</code>

<https://raw.githubusercontent.com/dennyzhang/cheatsheet-bosh-A4/master/bosh-workflow-intro.jpg>

1.7 Bosh Errand

Name	Command
List errands	<code>bosh errands -d \$deployment</code>
Run errand in a new vm	<code>bosh -n -d \$deployment run-errand <errand_name> Link: run-errand</code>
Run errand in a new vm with env alive	<code>bosh -n -d \$deployment run-errand <errand_name> --keep-alive</code>
Run errand in an existing vm	<code>bosh -n -d \$deployment run-errand <errand_name> --instance=worker/instance</code>

1.8 Bosh Blob

Name	Command
Reference	Link: Working with Blobs
Configure credential	Prepare <code>config/private.yml</code>
Add local file to blob	<code>bosh add-blob /tmp/<telegraf:1.7-alpine.tgz> container-images/telegraf:1.7-alpine.tgz</code>
Upload blobs	<code>bosh upload-blobs</code>
List blob	<code>bosh blobs</code>
Remove blob	<code>bosh remove-blob container-images/cadvisor-v0.31.0.tgz</code>

1.9 Bosh Task

Name	Command
List recent tasks	<code>bosh -d \$deployment tasks --recent</code>
List all tasks	<code>bosh -d \$deployment tasks -ar</code>
Attach to current task	<code>bosh -d \$deployment task -a</code>
Get task by id	<code>bosh -d \$deployment task \$id</code>
Get task detail msg	<code>bosh -d \$deployment task \$id --debug</code>
Get all logs for a deployment	<code>bosh -d \$deployment logs</code>
Get log for one vm	<code>bosh -d \$deployment logs \$instance_name</code>
List bosh events	<code>bosh events</code>

1.10 Bosh Job - Skeleton

Name	Command
Reference	Example: <code>kubo-release/jobs</code>
provides	Export resources for other bosh releases
consumes	Use resources from other bosh releases
packages	Packages get installed by bosh
templates	Conf files

1.11 Bosh Deployment

Name	Command
List bosh deployments	<code>bosh deployments</code>
Show detail of one deployment	<code>bosh -d \$deployment deployment</code>
Delete bosh deployment	<code>bosh -n -d \$deployment delete-deployment --force</code>
Get deployment manifest yaml	<code>bosh -d \$deployment manifest > my.yaml</code>
Run bosh deployment from yaml	<code>bosh -d \$deployment deploy my.yaml</code>

1.12 Bosh vs Kubernetes

Bosh - VM Lifecycle Management	Kubernetes - Container Lifecycle Management
Bosh Envs	Kubernetes Context
Bosh Deployment	Kubernetes Deployment
Bosh Stemcell	Docker Image
Bosh Manifest	Kubernetes Configmap & Yaml
Bosh Monit	Kubernetes Kubelet
Bosh Errand	Kubernetes Job
Bosh Release	Kubernetes Chart
Bosh Tile	Kubernetes Helm
Bosh Director	Kubernetes API server
Bosh Event	Kubernetes Event
Bosh Task	
Bosh Agent	
Bosh Addon	
Bosh Ops manager	No related feature: UI to customize your service

1.13 Bosh Concepts List

Name	Comment
Bosh Stemcells	A versioned Operating System image. Download stemcell
Bosh Deployment	An encapsulation of software and configuration that BOSH can deploy to the cloud.
Bosh Release	Everything needed to make a software component deployable by BOSH.
Bosh Job	Each release job represents a specific chunk of work that the release performs.
Bosh Package	It contains vendored in software source and scripts to compile it.
Bosh Errand	A short-lived job that can be triggered by an operator any time after the deploy.
Bosh Tile	A *.pivotal file with everything needed to install/upgrade a PCF service. Tile CheatSheet
Bosh Manifest	A YAML file that identifies one or more releases, stemcells and configurations
Bosh ops file	A YAML file that includes multiple operations to be applied to a different YAML file.
Bosh Event	Actions taken by the Director (via user or system control) are recorded as events
Bosh Director	The Director is the orchestrator of deployments.
Bosh Addon	A release job that is colocated on all VMs managed by the Director.
Bosh Agent	A process that runs continuously on each VM that BOSH deploys. one Agent process per VM
Bosh Jumpbox	A VM that acts as a single access point for the Director and deployed VMs.
Director Blobstore	A repository where BOSH stores release artifacts, logs, stemcells, and other content, etc
Cloud ID (CID)	ID returned from the Cloud identifying particular resource such as VM or disk.

1.14 Bosh Stemcell

Name	Command
Floating stemcell	Floating stemcells allow upgrade to the minor versions of stemcells but not the major versions.
Upload stemcell	<code>bosh -n upload-stemcell /tmp/stemcell.tgz</code>
List stemcells	<code>bosh -n stemcells</code>
Delete a stemcell	<code>bosh delete-stemcell <stemcell-name>/<stemcell-version></code>

1.15 Bosh Monit

Name	Comment
List bosh deployed daemon processes	<code>sudo su -, monit summary, watch 'monit summary'</code>
Restart a given daemon processes	<code>monit restart <process_name></code>
Check a given daemon processes status	<code>monit status <process_name></code>

1.16 Bosh Tools

Name	Comment
bbr bosh-backup-and-restore	A framework for backing up and restoring BOSH deployments and BOSH Directors.
bbl bosh-bootloader	A command line utility for standing up BOSH on multi-clouds
bpm BOSH process manager	Isolation of BOSH jobs to make they run on many different work schedulers instead of monit only

1.17 Bosh Challenges

Name	Comment
Create deployment is very slow	Need to compile packages again, and canary update for all VMs
Errands on non co-located vms	It takes 5+ minutes to spin up and destroy a VM
No repo mechanism for Bosh releases	Duplicate packages are required to make no external dependencies
Support <code>tail -f</code>	Personally I don't like the way of <code>bosh logs</code>
Retry a failed errand/task	Currently we can do it. But it's a lengthy manual process
More lightweight	Use <code>containers</code> , instead of <code>VMs</code> to run errands
Support rolling upgrade	Just like <code>kubect1 rollout</code>
More declarative	End users define requirements via <code>yaml files</code>

1.18 Online Help Usage

kubo@jumper:~\$ `bosh --help`

Usage:

`bosh [OPTIONS] <command>`

Application Options:

```

-v, --version          Show CLI version
--config=              Config file path (default: ~/.bosh/config) [$BOSH_CONFIG]
-e, --environment=     Director environment name or URL [$BOSH_ENVIRONMENT]
--ca-cert=             Director CA certificate path or value [$BOSH_CA_CERT]
--sha2                 Use SHA256 checksums [$BOSH_SHA2]
--parallel=            The max number of parallel operations (default: 5)
--client=              Override username or UAA client [$BOSH_CLIENT]
--client-secret=       Override password or UAA client secret [$BOSH_CLIENT_SECRET]
-d, --deployment=      Deployment name [$BOSH_DEPLOYMENT]
--column=              Filter to show only given column(s)
--json                 Output as JSON
--tty                  Force TTY-like output
--no-color              Toggle colored output
-n, --non-interactive  Don't ask for user input [$BOSH_NON_INTERACTIVE]
```

Help Options:

```

-h, --help             Show this help message
```

Available commands:

add-blob	Add blob	https://bosh.io/docs/cli-v2#add-blob
alias-env	Alias environment to save URL and CA certificate	https://bosh.io/docs/cli-v2#alias-env
attach-disk	Attaches disk to an instance	https://bosh.io/docs/cli-v2#attach-disk
blobs	List blobs	https://bosh.io/docs/cli-v2#blobs
cancel-task	Cancel task at its next checkpoint	https://bosh.io/docs/cli-v2#cancel-task
clean-up	Clean up releases, stemcells, disks, etc.	https://bosh.io/docs/cli-v2#clean-up
cloud-check	Cloud consistency check and interactive repair	https://bosh.io/docs/cli-v2#cloud-check
cloud-config	Show current cloud config	https://bosh.io/docs/cli-v2#cloud-config
config	Show current config for either ID or both type and name	https://bosh.io/docs/cli-v2#config
configs	List configs	https://bosh.io/docs/cli-v2#configs
cpi-config	Show current CPI config	https://bosh.io/docs/cli-v2#cpi-config
create-env	Create or update BOSH environment	https://bosh.io/docs/cli-v2#create-env
create-release	Create release	https://bosh.io/docs/cli-v2#create-release
delete-config	Delete config	https://bosh.io/docs/cli-v2#delete-config
delete-deployment	Delete deployment	https://bosh.io/docs/cli-v2#delete-deployment
delete-disk	Delete disk	https://bosh.io/docs/cli-v2#delete-disk
delete-env	Delete BOSH environment	https://bosh.io/docs/cli-v2#delete-env
delete-release	Delete release	https://bosh.io/docs/cli-v2#delete-release
delete-snapshot	Delete snapshot	https://bosh.io/docs/cli-v2#delete-snapshot
delete-snapshots	Delete all snapshots in a deployment	https://bosh.io/docs/cli-v2#delete-snapshots
delete-stemcell	Delete stemcell	https://bosh.io/docs/cli-v2#delete-stemcell
delete-vm	Delete VM	https://bosh.io/docs/cli-v2#delete-vm
deploy	Update deployment	https://bosh.io/docs/cli-v2#deploy
deployment	Show deployment information	https://bosh.io/docs/cli-v2#deployment
deployments	List deployments	https://bosh.io/docs/cli-v2#deployments
diff-config	Diff two configs by ID	https://bosh.io/docs/cli-v2#diff-config
disks	List disks	https://bosh.io/docs/cli-v2#disks
environment	Show environment	https://bosh.io/docs/cli-v2#environment
environments	List environments	https://bosh.io/docs/cli-v2#environments
errands	List errands	https://bosh.io/docs/cli-v2#errands
event	Show event details	https://bosh.io/docs/cli-v2#event
events	List events	https://bosh.io/docs/cli-v2#events
export-release	Export the compiled release to a tarball	https://bosh.io/docs/cli-v2#export-release
finalize-release	Create final release from dev release tarball	https://bosh.io/docs/cli-v2#finalize-release
generate-job	Generate job	https://bosh.io/docs/cli-v2#generate-job
generate-package	Generate package	https://bosh.io/docs/cli-v2#generate-package
help	Show this help message	https://bosh.io/docs/cli-v2#help
ignore	Ignore an instance	https://bosh.io/docs/cli-v2#ignore
init-release	Initialize release	https://bosh.io/docs/cli-v2#init-release
inspect-release	List release contents such as jobs	https://bosh.io/docs/cli-v2#inspect-release
instances	List all instances in a deployment	https://bosh.io/docs/cli-v2#instances
interpolate	Interpolates variables into a manifest	https://bosh.io/docs/cli-v2#interpolate
locks	List current locks	https://bosh.io/docs/cli-v2#locks
log-in	Log in	https://bosh.io/docs/cli-v2#log-in
log-out	Log out	https://bosh.io/docs/cli-v2#log-out
logs	Fetch logs from instance(s)	https://bosh.io/docs/cli-v2#logs
manifest	Show deployment manifest	https://bosh.io/docs/cli-v2#manifest
orphan-disk	Orphan disk	https://bosh.io/docs/cli-v2#orphan-disk
recreate	Recreate instance(s)	https://bosh.io/docs/cli-v2#recreate
releases	List releases	https://bosh.io/docs/cli-v2#releases
remove-blob	Remove blob	https://bosh.io/docs/cli-v2#remove-blob
repack-stemcell	Repack stemcell	https://bosh.io/docs/cli-v2#repack-stemcell
reset-release	Reset release	https://bosh.io/docs/cli-v2#reset-release
restart	Restart instance(s)	https://bosh.io/docs/cli-v2#restart
run-errand	Run errand	https://bosh.io/docs/cli-v2#run-errand
runtime-config	Show current runtime config	https://bosh.io/docs/cli-v2#runtime-config
scp	SCP to/from instance(s)	https://bosh.io/docs/cli-v2#scp
snapshots	List snapshots	https://bosh.io/docs/cli-v2#snapshots
ssh	SSH into instance(s)	https://bosh.io/docs/cli-v2#ssh

start	Start instance(s)	https://bosh.io/docs/cli-v2#start
stemcells	List stemcells	https://bosh.io/docs/cli-v2#stemc
stop	Stop instance(s)	https://bosh.io/docs/cli-v2#stop
sync-blobs	Sync blobs	https://bosh.io/docs/cli-v2#sync-
take-snapshot	Take snapshot	https://bosh.io/docs/cli-v2#take-
task	Show task status and start tracking its output	https://bosh.io/docs/cli-v2#task
tasks	List running or recent tasks	https://bosh.io/docs/cli-v2#tasks
unignore	Unignore an instance	https://bosh.io/docs/cli-v2#unign
update-cloud-config	Update current cloud config	https://bosh.io/docs/cli-v2#updat
update-config	Update config	https://bosh.io/docs/cli-v2#updat
update-cpi-config	Update current CPI config	https://bosh.io/docs/cli-v2#updat
update-resurrection	Enable/disable resurrection	https://bosh.io/docs/cli-v2#updat
update-runtime-config	Update current runtime config	https://bosh.io/docs/cli-v2#updat
upload-blobs	Upload blobs	https://bosh.io/docs/cli-v2#uploa
upload-release	Upload release	https://bosh.io/docs/cli-v2#uploa
upload-stemcell	Upload stemcell	https://bosh.io/docs/cli-v2#uploa
variables	List variables	https://bosh.io/docs/cli-v2#varia
vendor-package	Vendor package	https://bosh.io/docs/cli-v2#vendo
vms	List all VMs in all deployments	https://bosh.io/docs/cli-v2#vms

Succeeded

1.19 More Resources

<https://github.com/bosh-tips/tips>

<http://engineering.pivotal.io/post/compiled-releases-for-pipelines/>

License: Code is licensed under MIT License.