

CheatSheet: Concourse

VMWARE

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

File me Issues or star this repo

1.1 Concourse Concepts

Name	Summary
fly	Pipelines are configured entirely via fly CLI
Teams	Every pipeline and one-off build belongs to a team.
Targets	Alias of Concourse endpoints, which composes of server url, credentials, etc
Pipelines	A pipeline is the result of configuring Jobs and Resources together.
Jobs	Jobs determine the actions of your pipeline,
Resource	Allow you to define inputs and outputs to jobs in a pipeline
Tasks	The smallest configurable unit in a Concourse pipeline is a single task.
Reference	Link: Concourse Concepts, Link: concoursetutorial.com

1.2 Concourse Resources

Name	Summary
file-downloader-resource	enable downloading pivotal product files and stemcells
gcs-resource: google bucket	
pool-resource	Atomically manages external environments
github-release	
git-release	tracks commits in a branch of a Git repository
docker-image-resource: docker image	
Step: tags	Use tag to match workers
bbl (bosh-bootloader)	Command line utility for standing up a BOSH director on any IAAS
Reference	GitHub: pcf-pipelines

1.3 Concourse Examples

Name	Summary
Hello world pipeline	example: helloworld.yml
Pass output to input across different tasks	example: tasks-output-input.yml
Trigger pipeline test every 5 minutes	example: run-test-regularly.yml
git-resource	example: example-git.yml
concourse-curl-resource: http download link	example: url-download.yml

1.4 Concourse Containers

Name	Summary
Login to container	<code>fly -t \$target hijack -u \$job_url</code>
Run command without sudden exit	<code>nohup SOMECOMMAND & tail -f nohup.out</code>
Hijack to containers with a successful run	By design when task has passed, the container will be deleted

1.5 Concourse Pipeline

Name	Summary
List all authenticated end points	<code>fly targets</code>
Get fly version	<code>fly --version</code>
fly login	<code>fly -t \$target login -c \$concourse_url -n \$team_name</code>
Get pipeline definition	<code>fly -t \$target get-pipeline -p \$job_name > job.yaml</code>
Update pipeline	<code>fly -t \$target set-pipeline -p \$pipeline_name -c pipeline.yml -l credential</code>
Update pipeline	<code>fly -t \$target set-pipeline -p \$pipeline_name -c pipeline.yml -v env1=var1</code>
Destroy pipeline	<code>fly -t \$target destroy-pipeline -p \$pipeline_name</code>
Destroy pipeline without confirmation	<code>fly -t \$target destroy-pipeline -p \$pipeline_name -n</code>
Format pipeline	<code>fly format-pipeline -c pipeline.yml</code>
Validate pipeline	<code>fly validate-pipeline -c pipeline.yml</code>

<https://raw.githubusercontent.com/dennyzhang/cheatsheet.dennyzhang.com/master/cheatsheet-concourse-A4/concourse-sample.png>

1.6 Concourse Resource

Name	Summary
github-release resource get the old tag	<code>fly -t \$target check-resource -r \$pipeline-name/\$resource-name -f tag</code>
bosh-io-stemcell resource get the old version	<code>fly -t \$target check-resource -r \$pipeline-name/ubuntu-xenial-stemcell</code>
github release use a specific tag	<code>github-release-specific-tag.md</code>

1.7 Concourse Security

Name	Summary
Fly login to the default main team	<code>fly -t \$target login -n main -c \$concourse_url</code>
Fly login to a different team	<code>fly -t \$target login -n \$team_name -c \$concourse_url</code>
List all teams	<code>fly -t \$target teams, fly -t \$target teams -d</code>
Fly logout all targets	<code>fly logout --all</code>
Install ansible vault in mac	<code>brew install ansible</code>
Create a team	<code>fly -t \$target set-team -n \$team_name --github-org \$my_org</code>
Create a team	<code>fly -t \$target set-team -n \$team_name --allow-all-users</code>
Destroy a team	<code>fly -t \$target destroy-team --team-name \$team_name</code>
Login to a team	<code>fly -t \$target set-team -n \$team_name</code>
Fly conf file	<code>~/.flyrc</code>
Reference	After login to the main team, you can update other teams. Link: Concourse Team

1.8 Concourse Workers

Name	Summary
List all concourse workers	<code>fly -t \$target workers</code>
List all concourse containers	<code>fly -t \$target containers</code>

<https://raw.githubusercontent.com/dennyzhang/cheatsheet.dennyzhang.com/master/cheatsheet-concourse-A4/concourse-design.png>

1.9 Concourse Opportunities

Name	Summary
fly set-pipeline run into "error: forbidden"	Need to logout from both fly client (<code>fly logout --all</code>) and Concourse
List team details for all targets	<code>fly -t \$target teams -d, ~/.flyrc</code>
Concourse UI runs into a blank page	If pipeline has recursive dependencies, Concourse UI may have issue to run
Split one pipeline yaml into small reusable pieces	Not supported. An uncomfortable workaround: generating yaml with other
Add a readonly member to one team	Not supported. Concourse only support authentication control, but not aut
Does Concourse have REST API?	Not supported so far. Link: GitHub , Link: Concourse REST endpoints
Reference	concourse-tutorial , concourse-pipeline-samples

1.10 fly Online Usage

```
bash-3.2$ fly --help
```

```
error: Usage:
```

```
fly [OPTIONS] <command>
```

Application Options:

```
-t, --target=           Concourse target name
-v, --version           Print the version of Fly and exit
--verbose               Print API requests and responses
--print-table-headers  Print table headers even for redirected output
```

Help Options:

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

Available commands:

```
abort-build           Abort a build (aliases: ab)
builds                List builds data (aliases: bs)
check-resource        Check a resource (aliases: cr)
checklist             Print a Checkfile of the given pipeline (aliases: cl)
containers            Print the active containers (aliases: cs)
destroy-pipeline      Destroy a pipeline (aliases: dp)
destroy-team          Destroy a team and delete all of its data (aliases: dt)
execute               Execute a one-off build using local bits (aliases: e)
expose-pipeline       Make a pipeline publicly viewable (aliases: ep)
format-pipeline       Format a pipeline config (aliases: fp)
get-pipeline          Get a pipeline's current configuration (aliases: gp)
help                  Print this help message
hide-pipeline         Hide a pipeline from the public (aliases: hp)
hijack                Execute a command in a container (aliases: intercept, i)
jobs                  List the jobs in the pipelines (aliases: js)
login                 Authenticate with the target (aliases: l)
logout                Release authentication with the target (aliases: o)
order-pipelines       Orders pipelines (aliases: op)
pause-job             Pause a job (aliases: pj)
pause-pipeline        Pause a pipeline (aliases: pp)
pause-resource        Pause a resource (aliases: pr)
pipelines             List the configured pipelines (aliases: ps)
prune-worker          Prune a stalled, landing, landed, or retiring worker (aliases: pw)
rename-pipeline       Rename a pipeline (aliases: rp)
rename-team           Rename a team (aliases: rt)
set-pipeline          Create or update a pipeline's configuration (aliases: sp)
set-team              Create or modify a team to have the given credentials (aliases: st)
status                Login status
sync                  Download and replace the current fly from the target (aliases: s)
targets               List saved targets (aliases: ts)
teams                 List the configured teams (aliases: t)
trigger-job           Start a job in a pipeline (aliases: tj)
unpause-job           Unpause a job (aliases: uj)
unpause-pipeline      Un-pause a pipeline (aliases: up)
unpause-resource       Unpause a resource (aliases: ur)
validate-pipeline     Validate a pipeline config (aliases: vp)
volumes               List the active volumes (aliases: vs)
watch                 Stream a build's output (aliases: w)
workers               List the registered workers (aliases: ws)
```

1.11 fly set-pipeline Online Usage

```
> fly set-pipeline --help
```

error: Usage:

fly [OPTIONS] set-pipeline [set-pipeline-OPTIONS]

Application Options:

-t, --target=	Concourse target name
-v, --version	Print the version of Fly and exit
--verbose	Print API requests and responses
--print-table-headers	Print table headers even for redirected output

Help Options:

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

[set-pipeline command options]

-n, --non-interactive	Skips interactions, uses default values
--no-color	Disable color output
--check-creds	Validate credential variables against credential manager
-p, --pipeline=	Pipeline to configure
-c, --config=	Pipeline configuration file
-v, --var=[NAME=STRING]	Specify a string value to set for a variable in the pipeline
-y, --yaml-var=[NAME=YAML]	Specify a YAML value to set for a variable in the pipeline
-l, --load-vars-from=	Variable flag that can be used for filling in template values in configuration

1.12 More Resources

<http://www.mikeball.info/blog/concourse-git-resource/>

<https://concoursetutorial.com/>

License: Code is licensed under MIT License.