

# Cheatsheet: Cloudfoundry UAA

VMWARE

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

File me Issues or star this repo.

## 1.1 uaac

| Name                          | Command   |
|-------------------------------|---|
| Run command                   | <code>uaac apt-install ruby ruby-dev build-essential rubygems; gem install uaac-cf</code>                   |
| Set a new target              | <code>uaac target https://&lt;uaa-dnsname&gt;:8443 --skip-ssl-validation</code>                             |
| List all context              | <code>uaac contexts</code>  |
| Log in as a UAA admin         | <code>uaac token client get admin -s &lt;uaa-management-admin-client&gt;</code> Link: Log in as a UAA Admin |
| List users                    | <code>uaac users</code>   |
| List all client registrations | <code>uaac clients</code>   |
| Add user                      | <code>uaac user add alana --emails alana@example.com -p password</code> Link: Grant PKS Access to a User    |
| Add user to group             | <code>uaac member add pks.clusters.admin alana</code> Link: Grant PKS Access to a User                      |
| Get specific user account     | <code>uaac user get alana</code>  |
| Get user                      | <code>uaac user get pks-admin</code>  |
| Reference                     | Bosh CheatSheet, Tile CheatSheet, UAA CheatSheet, PKS CheatSheet  |

## 1.2 UAA Concept

| Name     | Command |
|----------|---------|
| target   |         |
| context  |         |
| client   |         |
| token    |         |
| password |         |
| secret   |         |

## 1.3 credhub

| Name          | Command   |
|---------------|---|
| CredHub       | It centralizes and secures credential generation, storage, lifecycle management, and access.                              |
| Reference     | GitHub: credhub, GitHub: credhub-cli, Link: credhub   |
| credhub login | <code>credhub login -client-name credhub-admin -client-secret "admin_secret" -s "{api_url}" -ca-cert "\${ca_file}"</code> |

<https://raw.githubusercontent.com/dennyzhang/cheatsheet.dennyzhang.com/master/cheatsheet-uaa-A4/credhub-service.png>

## 1.4 Online Help Usage

kubo@jumper:~\$ uaac --help

UAA Command Line Interface

### Miscellaneous

|                         |   |
|-------------------------|---|
| version                 | Display version   |
| help [topic command...] | Display summary or details of command or topic                              |
| target [uaa_url]        | Display current or set new target   |
|                         | -f   --[no-]force, set even if target does not respond                      |
|                         | --ca-cert [file], use the given CA certificate to validate the target's SSL |
|                         | --skip-ssl-validation, do not attempt to validate ssl certificate           |
| targets                 | Display all targets   |
| context [name]          | Display or set current context  |

|                                  |   |
|----------------------------------|---|
| contexts                         | Display all contexts  |
| System Information               |   |
| info                             | get information about current target  |
| me                               | get authenticated user information  |
| prompts                          | Show prompts for credentials required for implicit grant post   |
| signing key                      | get the UAA's token signing key(s)<br>-c   --client <name><br>-s   --secret <secret>, client secret   |
| stats                            | Show UAA's current usage statistics<br>-c   --client <name><br>-s   --secret <secret>, client secret  |
| password strength [password]     | calculate strength score of a password  |
| Tokens                           |   |
| token get [credentials...]       | Gets a token by posting user credentials with an implicit grant request<br>-c   --client <name><br>--scope <list>   |
| token client get [id]            | Gets a token with client credentials grant<br>-s   --secret <secret>, client secret<br>--scope <list>   |
| token owner get [client] [user]  | Gets a token with a resource owner password grant<br>-s   --secret <secret>, client secret<br>-p   --password <password>, user password<br>--scope <list>   |
| token sso get [client]           | Gets a token based on a one time passcode after successful SSO via browser<br>-s   --secret <secret>, client secret<br>--passcode <passcode><br>--scope <list>  |
| token refresh [refreshtoken]     | Gets a new access token from a refresh token<br>-c   --client <name><br>-s   --secret <secret>, client secret<br>--scope <list>   |
| token authcode get               | Gets a token using the authcode flow with browser<br>-c   --client <name><br>-s   --secret <secret>, client secret<br>--scope <list><br>--[no-]cf, save token in the ~/.cf_tokens file<br>--port <number>, pin internal server to specific port |
| token implicit get               | Gets a token using the implicit flow with browser<br>-c   --client <name><br>--scope <list><br>--[no-]cf, save token in the ~/.cf_tokens file<br>--port <number>, pin internal server to specific port  |
| token decode [token] [tokentype] | Show token contents as parsed locally or by the UAA. Decodes locally unless<br>--key <key>, Token validation key<br>-c   --client <name><br>-s   --secret <secret>, client secret   |
| token delete [contexts...]       | Delete current or specified context tokens and settings<br>--[no-]all, remove all contexts  |
| User Accounts                    |   |
| users [filter]                   | List user accounts<br>-a   --attributes <names>, output for each user<br>--start <number>, start of output page<br>--count <number>, max number per page  |
| user get [name]                  | Get specific user account<br>--origin <identity provider origin, defaults to UAA><br>-a   --attributes <names>, output for each user  |

|   |   |
|---|---|
| user add [name]                           | Add a user account<br>--given_name <name><br>--family_name <name><br>--emails <addresses><br>--phones <phone_numbers><br>--origin <identity provider origin, defaults to UAA><br>-p   --password <password>, user password  |
| user update [name]                        | Update a user account with specified options<br>--given_name <name><br>--family_name <name><br>--emails <addresses><br>--phones <phone_numbers><br>--origin <identity provider origin, defaults to UAA><br>--del_attrs <attr_names>, list of attributes to delete |
| user delete [name]                        | Delete user account<br>--origin <identity provider origin, defaults to UAA>   |
| user ids [username id...]                 | Gets user names and ids for the given users   |
| user unlock [name]                        | Unlocks the user account  |
| user deactivate [name]                    | Deactivates user  |
| user activate [name]                      | Activates user  |
| password set [name]                       | Set password<br>-p   --password <password>, user password   |
| password change                           | Change password for authenticated user in current context<br>-o   --old_password <password>, current password<br>-p   --password <password>, user password  |
| <b>Groups</b>                             |   |
| groups [filter]                           | List groups<br>-a   --attributes <names>, output for each user<br>--start <start>, show results starting at this index<br>--count <count>, number of results to show  |
| group get [name]                          | Get specific group information<br>-a   --attributes <names>, output for each user   |
| group add [name]                          | Adds a group  |
| group delete [name]                       | Delete group  |
| group mappings                            | List all the mappings between uaa scopes and external groups<br>--start <start>, show results starting at this index<br>--count <count>, number of results to show  |
| group map [external_group]                | Map uaa groups to external groups<br>--id <id>, map uaa group using group id<br>--name <name>, map uaa scope using group name<br>--origin <origin>, map uaa scope to external group for this origin. Default  |
| group unmap [group_name] [external_group] | Unmaps an external group from a uaa group<br>--origin <origin>, map uaa scope to external group for this origin. Default  |
| member add [name] [users...]              | add members to a group  |
| member delete [name] [users...]           | remove members from a group   |
| <b>Client Application Registrations</b>   |   |
| clients [filter]                          | List client registrations<br>-a   --attributes <names>, output for each user<br>--start <start>, show results starting at this index<br>--count <count>, number of results to show  |
| client get [id]                           | Get specific client registration<br>-a   --attributes <names>, output for each user   |
| client add [id]                           | Add client registration<br>--name <string><br>--scope <list><br>--authorized_grant_types <list>   |

|                         |   |
|-------------------------|---|
|                         | --authorities <list><br>--access_token_validity <seconds><br>--refresh_token_validity <seconds><br>--redirect_uri <list><br>--autoapprove <list><br>--signup_redirect_url <url><br>--clone <other>, get default settings from other<br>-s   --secret <secret>, client secret<br>-i   --[no-]interactive, interactively verify all values                                      |
| client update [id]      | Update client registration  |
|                         | --name <string><br>--scope <list><br>--authorized_grant_types <list><br>--authorities <list><br>--access_token_validity <seconds><br>--refresh_token_validity <seconds><br>--redirect_uri <list><br>--autoapprove <list><br>--signup_redirect_url <url><br>--del_attrs <attr_names>, list of attributes to delete<br>-i   --[no-]interactive, interactively verify all values |
| client delete [id]      | Delete client registration  |
| secret set [id]         | Set client secret   |
|                         | -s   --secret <secret>, client secret   |
| secret change           | Change secret for authenticated client in current context   |
|                         | --old_secret <secret>, current secret   |
|                         | -s   --secret <secret>, client secret   |
| CURL                    |   |
| curl [path]             | CURL to a UAA endpoint  |
|                         | -X   --request <method>, request method type, defaults to GET   |
|                         | -d   --data <data>, data included in request body   |
|                         | -H   --header <header>, header to be included in the request  |
|                         | -k   --insecure, makes request without verifying SSL certificates   |
| Global options:         |   |
| -h   --[no-]help        | display helpful information   |
| -v   --[no-]version     | show version  |
| -d   --[no-]debug       | display debug information   |
| -t   --[no-]trace       | display extra verbose debug information   |
| --config [string file]  | file to get/save configuration information or yaml string   |
| -z   --zone <subdomain> | subdomain of zone to manage   |

## 1.5 More Resources

License: Code is licensed under MIT License.

<https://docs.cloudfoundry.org/credhub/>

<https://github.com/cloudfoundry/uaa>