Akamai As Code Training

GIT - CLI

2023-01-30 Mon 16:22

Table of Contents

- 1. GIT Basics
- 2. Akamai CLI
- 3. Akamai CLI - DNS
- 4. Akamai CLI - PM
- 5. Akamai CLI - Pipeline
- 6. Thank you

1. GIT Basics

GIT Basic Configuration - Identity

- `git config --global user.name Your Name`
- `git config --global user.email your@email.com`

GIT Basic Configuration - Alias

- `git config --global alias.co commit`
- `git config --global alias.st status`
- `git config --global alias.br branch`
- `git config --global alias.l log --oneline -10`
- `git config --global alias.ltree log --color --graph --pretty=format:'%Cred%h%Creset - %C(yellow)%d%Creset %s %Cgreen(%cr)%C(bold blue) <%an> %Creset' --abbrev-commit --date-order`

2. Akamai CLI

General Akamai CLI usage

```
akamai [global flags] command [command flags] [arguments...]
```

```
Global Flags:
  --edgerc value, -e value                Location of the credentials file [$AKAMAI_EDGERC]
  --section value, -s value               Section of the credentials file [$AKAMAI_EDGERC_SECTION]
  --accountkey value, --account-key value Account switch key [$AKAMAI_EDGERC_ACCOUNT_KEY]
  --help                                  show help (default: false)
  --bash                                  Output bash auto-complete (default: false)
  --zsh                                   Output zsh auto-complete (default: false)
  --proxy value                           Set a proxy to use
  --version                               Output CLI version (default: false)
```

This is an example command list; it can be different based on what is installed.

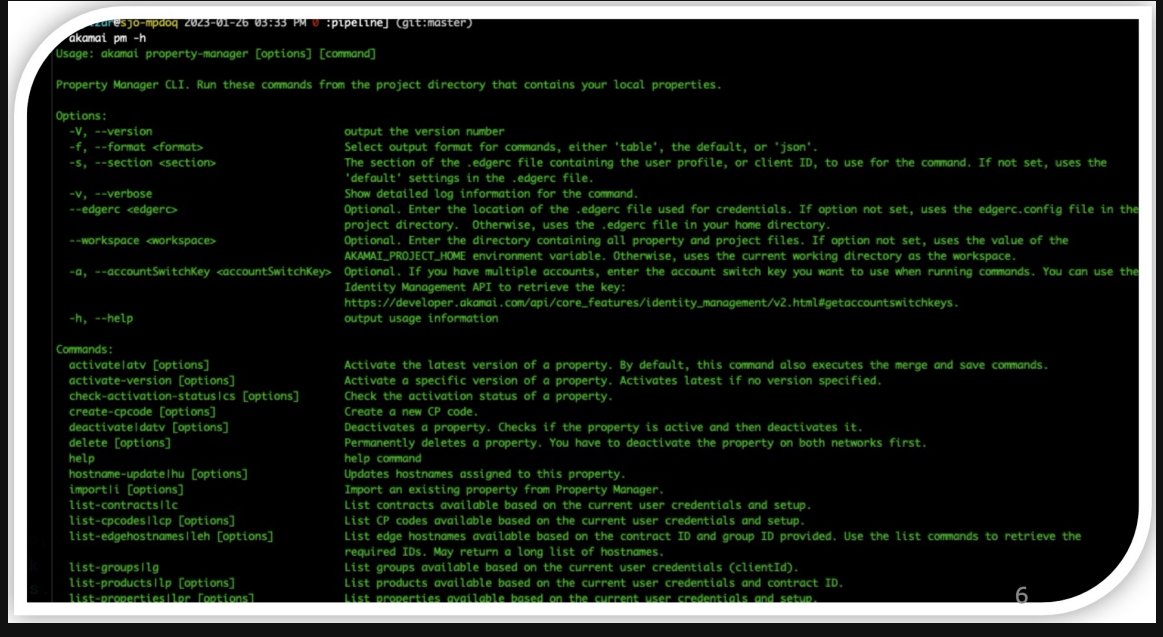
```
Commands:
  config
  install (alias: get)
  list
  search
  uninstall
  update
  upgrade
  help
Installed Commands::
  appsec (alias: appsec/appsec)
  cloudlets (aliases: cloudlets, cloudlets/cloudlets)
  cps (aliases: certs, cps/cps)
  dns (alias: dns/dns)
  edgekv (aliases: ekv, edgekv, edgeworkers/edgekv)
  edgeworkers (aliases: ew, edgeworkers, edgeworkers/edgeworkers)
  firewall (aliases: fw, firewall/firewall)
  gtm (alias: gtm/gtm)
  image-manager (aliases: im, image-manager/image-manager)
  pipeline (aliases: pl, pd, proddeploy, property-manager/pipeline)
  property-manager (aliases: pm, snippets, property-manager/property-manager)
  purge (alias: purge/purge)
  query (alias: query/query)
  sandbox (alias: sandbox/sandbox)
  site-shield (aliases: ss, firewall/site-shield)
```

Default command list

Akamai CLI – Getting Help

Example how to get help from a command (it apply to any command)

- `akamai pm -h`
- `akamai pm command -h`



3. Akamai CLI - DNS

Akamai CLI – DNS – Create Zone

Example create DNS zone

```
akamai --accountkey accountID dns create-zoneconfig newzone.com
--type primary --contractid contractID --groupid groupID
--initialize
```

Akamai CLI – DNS – Create Record

Example create a CName entry

```
akamai dns --accountkey accountID create-recordset newzone.com
--name newrecord.newzone.com --type cname –ttl seconds
--rdata FQDN
```

Akamai CLI – DNS – List

Example list DNS zone information

```
akamai dns --accountkey accountID list-zoneconfig --search newzone.com
```

Example list DNS record information

```
akamai dns --accountkey accountID list-recordsets newrecord.newzone.com
```

4. Akamai CLI - PM

Akamai CLI – PM – New Property with Template

Create a new PM property based on template

```
akamai pm -a accountID new-property -e template.config.com_pm
-n template_version -p newconfig.name.com_pm
```

```
-e, --propertyId <propertyId/propertyName> Optional. Use an existing
property as the blueprint for the new one. Enter either a property ID
or an exact property name. The CLI looks up the group ID, contract ID,
and product ID of the existing property and uses that data to create a
new property.

-n, --propver <propver> Add only if using a property as a
template. Enter the version of the existing property to use as the
blueprint. Uses latest version if omitted.

-p, --property <propertyName> Property name. Optional if a default
property was previously set with the set-default command.
```

Akamai CLI – PM – New Property with no Template

Create a new PM property based on product ID

```
akamai pm -a accountID new-property -c contractId -g groupId
-d productId -p newconfig.name.com_pm
```

```
-c, --contractId <contractId> Enter the contract ID to use. Optional
if using -e with a property ID or name.

-d, --productId <productId> Enter the product ID to use. Optional if
using -e with a property ID or name.

-g, --groupId <groupId> Enter the group ID for the property. Optional
if using -e with a property ID or name.

-p, --property <propertyName> Property name. Optional if a default
property was previously set with the set-default command.
```

Akamai CLI – PM – Property Update

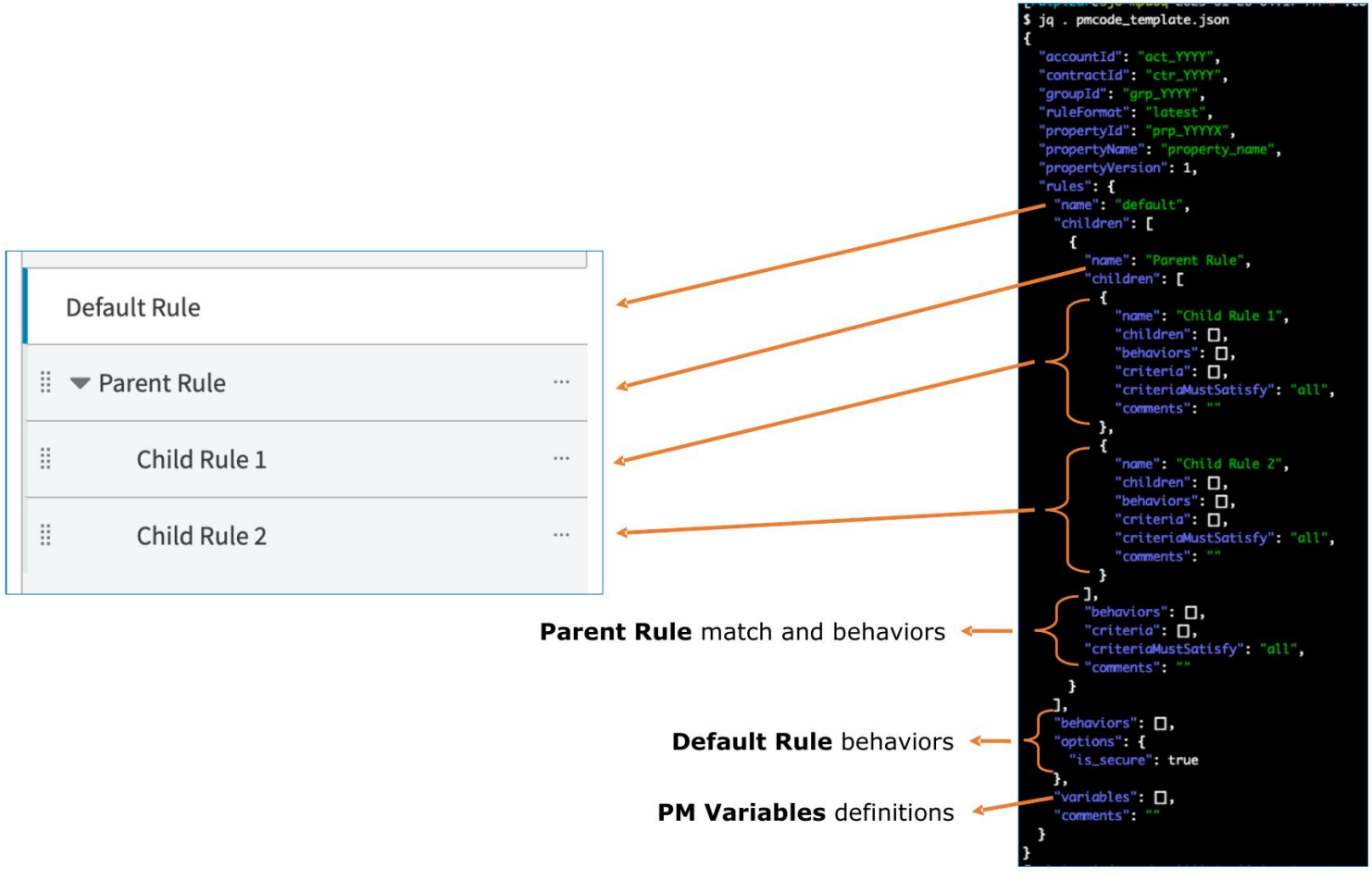
Update PM property configuration with json file

```
akamai pm -a accountID property-update --note "Version note"
-p newconfig.name.com_pm --propver configversion --file logic.json
```

```
--file <file> Specify the JSON file containing the rules. PM json
specifications:
https://techdocs.akamai.com/property-mgr/reference/rule-trees

--note <message> Alias of --message. Add comments for the
property. -p, --property <property> The name or ID of the property you
are updating. --propver <propver> Optional. Select the property
version to update
```

Akamai CLI – PM – JSON



[PM Rule Tree Specification]

Akamai CLI – PM – Match

```
"criteria": [
  {
    "name": "contentType",
    "options": {
      "matchOperator": "IS_ONE_OF",
      "matchWildcard": true,
      "matchCaseSensitive": false,
      "values": [
        "text/html*",
        "text/css*",
        "application/x-javascript*"
      ]
    }
  }
],
```

Rule tree criteria specification

Akamai CLI – PM – Behaviors

```
"behaviors": [
  {
    "name": "origin",
    "options": {
      "originType": "CUSTOMER",
      "hostname": "example.com",
      "forwardHostHeader": "REQUEST_HOST_HEADER",
      "cacheKeyHostname": "ORIGIN_HOSTNAME",
      "compress": true,
      "tcipEnabled": false,
      "httpPort": 80
    }
  },
  {
    "name": "cpCode",
    "options": {
      "value": {
        "id": 12345,
        "name": "main site"
      }
    }
  }
```

[Rule tree behavior specification]

Akamai CLI – PM – Property Activation

Activate an Akamai PM property

```
akamai pm -a accountID activate-version -m "Activation note"
-e "email list" -p newconfig.name.com_pm --propver configversion
-n <network> -w
```

```
-m, --message <message>     Enter a message describing the reason for
                            activating.
-n, --network <network>     Network, either 'production' or 'staging'. You can
                            shorten 'production' to 'prod' or 'p' and
                            'staging' to 'stage' or 's'.
-p, --property <property>   Property name or property ID. Optional if default
                            property was previously set using set-default.
--propver <propver>        Optional. The property version to activate. Uses
                            latest version if not specified.
-w, --wait-for-activate     Prevents you from entering more commands until
                            activation is complete. May take several minutes.
```

5. Akamai CLI - Pipeline

Akamai CLI – Pipeline - Create

```
Usage: akamai pipeline new-pipeline|np [options] <environments...>

akamai pipeline -a accountID new-pipeline -e template_property_pm
-n template_version
-p new_pipeline_name <space separated environments>
```

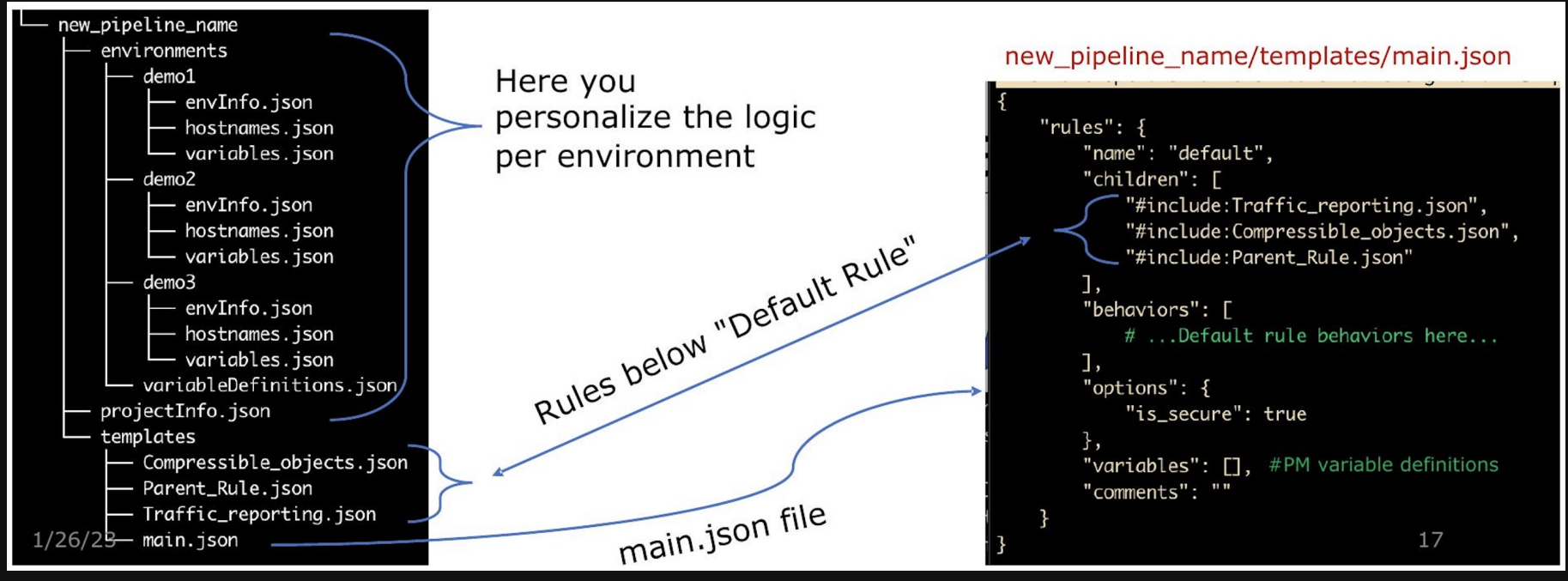
```
-e, --propertyId <propertyId/propertyName> Optional. Use an existing
property as the blueprint for the new one. Enter either a property ID
or an exact property name. The CLI looks up the group ID, contract ID,
and product ID of the existing property and uses that data to create a
new property.

-n, --propver <propver> Add only if using a property as a
template. Enter the version of the existing property to use as the
blueprint. Uses latest version if omitted.

-p, --property <propertyName> Property name.
```

Akamai CLI – Pipeline - Tree

```
# example
akamai pipeline -a accoundID new-pipeline -e a_config_as_template_pm -n template_version -p new_pipeline demo1 demo2 demo3
```



Akamai CLI – Pipeline - Variables

Template File | variableDefinitions.json | variables.json (environment)

```
$ jq . main.json
{
  "rules": {
    "name": "default",
    "children": [
      "#include:Traffic_reporting.json",
      "#include:Compressible_objects.json",
      "#include:Parent_Rule.json"
    ],
    "behaviors": [
      {
        "name": "origin",
        "options": {
          "cacheKeyHostname": "REQUEST_HOST_HEADER",
          "compress": true,
          "enableTrueClientIp": true,
          "forwardHostHeader": "REQUEST_HOST_HEADER",
          "httpPort": 80,
          "httpsPort": 443,
          "originCertificate": "",
          "originSni": true,
          "originType": "CUSTOMER",
          "ports": "",
          "trueClientIpClientSetting": false,
          "trueClientIpHeader": "True-Client-IP",
          "verificationMode": "CUSTOM",
          "hostname": "${env.originHostname}",
          "customValidCnValues": [
            "{{Origin Hostname}}",
```

```
$ jq . environments/variableDefinitions.json
{
  "definitions": {
    "originHostname": {
      "type": "hostname",
      "default": null
    },
    "cpCode": {
      "type": "cpCode",
      "default": 1430518
    }
  }
}
```

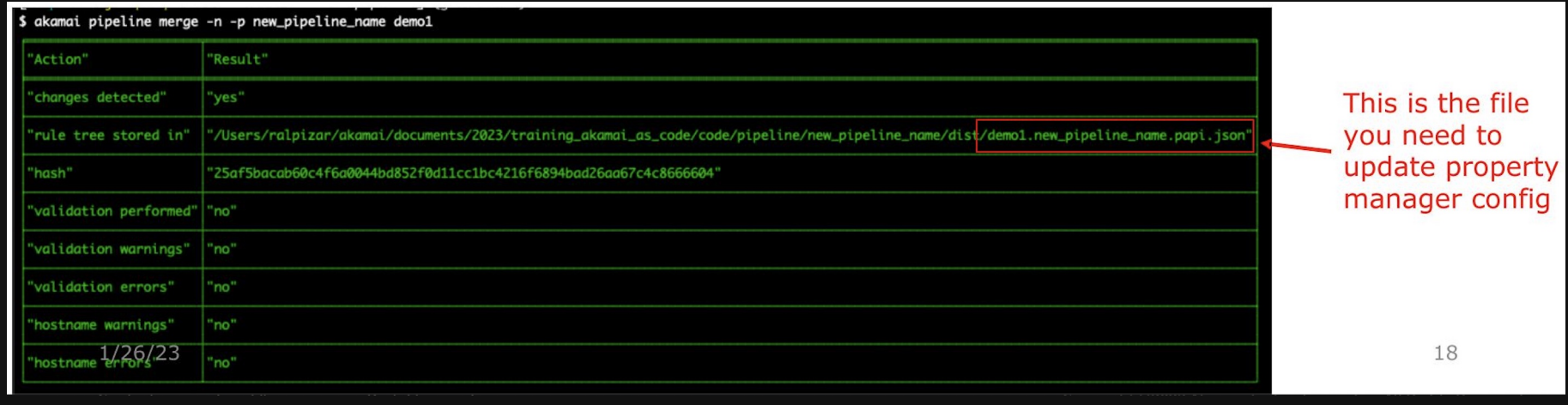
```
$ jq . environments/demo1/variables.json
{
  "originHostname": "origin-demo1.demos.training.ranet.me",
  "cpCode": null
}
```

Akamai CLI – Pipeline - Merge

Pipeline merge an environment

```
akamai pipeline merge -n -p pipeline_name environment
```

```
-n, --no-validate Merge the environment without validating.
-p, --pipeline <pipelineName> Pipeline name.
```



6. Thank you

Any question?