### Cloud Resource Naming Conventions

Please follow the naming convention below to help consistent provisioning of AWS resources.

<table>
<thead>
<tr>
<th>Item</th>
<th>Formatting Style</th>
</tr>
</thead>
<tbody>
<tr>
<td>Determined by Construct</td>
<td>{{construct_name}}</td>
</tr>
<tr>
<td>Entered by User</td>
<td>{{item_name}}</td>
</tr>
<tr>
<td>Optional Text based on use</td>
<td>[<a href="naming"></a>]]</td>
</tr>
<tr>
<td>To be Included “AS IS”</td>
<td>wording</td>
</tr>
</tbody>
</table>

For Updates to items already present, please use the inline-comment feature. For new additions which are not below, use the page comments (which will be deleted when resolved).

- Naming Constructs
- AWS Resource Naming Standards
  - CloudFormation Naming Standards
  - AWS Account Items
  - VPC Resource Naming Standards
  - EC2 Resource Naming Standards
  - S3 Resource Naming Standards
  - Lambda Function Naming Standard
  - RDS Resource Naming Standards
  - ElastiCache Resource Naming Standards
  - CodeDeploy Resource Naming Standards
  - IAM Resource Naming Standards
  - Cloudwatch Resource Naming Standards
  - SNS Resource Naming Standards
  - SQS Resource Naming Standards
  - CloudOps Tool Naming Standards
  - Package Names for CAR
  - Splunk Index Naming
  - Git Repository
  - DynamoDB Tables
  - CloudWatch Log Groups
  - Elastic File System
  - Ansible Tower
- Page History

1. Dashes are required as entered, do not remove them
2. Spaces are left for legibility reasons around dashes, they are not to be included in the name
3. Delimiters are not to be substituted into areas which are components of the name. Where clarity is needed for a veryLongName use modifiedCamelCase

If you need to deviate from standards, please contact your team architect. It will get added to the standard as appropriate (e.g. "search" for a tier type)

### Naming Constructs

<table>
<thead>
<tr>
<th>AWS Resource</th>
<th>Resource Name</th>
<th>Comment</th>
<th>Example</th>
</tr>
</thead>
</table>

---
Account Naming Construct

\(\{\text{teamName}\} - \{\text{environment}\} - \{\text{context}\}\)

- **teamName** should be the name of the application team providing support
  - atsea
  - dms
- **Environment** should be one of:
  - prod
  - dev
- **Context** should be one of:
  - standard
  - level4 (for Level 4 specific accounts)
  - bcdr (only for bcdr exclusive accounts)

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
<th>Example</th>
</tr>
</thead>
<tbody>
<tr>
<td>admints-dev</td>
<td>standard</td>
<td>admints-dev-standard</td>
</tr>
<tr>
<td>admints-dev</td>
<td>prod</td>
<td>admints-dev-prod-standard</td>
</tr>
<tr>
<td>admints-dev</td>
<td>level4</td>
<td>admints-dev-prod-level4</td>
</tr>
</tbody>
</table>

App Naming Construct

\(\{\text{appName}\} - \{\text{environment}\} \{\text{number}\} - \{\text{scope}\} \{\text{number}\}\)

- **Environment** should be one of:
  - prod
  - int
  - uat
  - stage
  - test
  - dev
  - blue
  - green
- **Scope** should be used if necessary and can reflect components of an application
  - auth
  - solr
  - app
  - web
  - cache
  - master
  - worker
  - nfs
- If multiple are needed, increment number.
- Region is optional and should be used when necessary (e.g. BC/DR)

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
<th>Example</th>
</tr>
</thead>
<tbody>
<tr>
<td>takeasweater-dev</td>
<td>prod</td>
<td>takeasweater-dev-prod-prod</td>
</tr>
<tr>
<td>takeasweater-dev</td>
<td>stage</td>
<td>takeasweater-dev-stage</td>
</tr>
<tr>
<td>takeasweater-dev</td>
<td>test</td>
<td>takeasweater-dev-test</td>
</tr>
<tr>
<td>takeasweater-dev</td>
<td>prod</td>
<td>takeasweater-dev-prod</td>
</tr>
<tr>
<td>ask-prod-auth</td>
<td>prod</td>
<td>ask-prod-auth</td>
</tr>
<tr>
<td>coursecatalog</td>
<td>prod</td>
<td>coursecatalog-prod</td>
</tr>
<tr>
<td>prod-solr</td>
<td>prod</td>
<td>prod-solr</td>
</tr>
</tbody>
</table>

AWS Resource Naming Standards

CloudFormation Naming Standards

<table>
<thead>
<tr>
<th>AWS Resource</th>
<th>Resource Name</th>
<th>Comment</th>
<th>Example</th>
</tr>
</thead>
<tbody>
<tr>
<td>CloudFormation Stack</td>
<td>If this stack is universal and not application specific:</td>
<td>stack_group should be one of:</td>
<td>admints-prod-level4-vpc-cf or takeasweater-prod-all-cf or takeasweater-prod-iam-cf</td>
</tr>
<tr>
<td></td>
<td>({\text{account_naming_construct}} - {\text{stack_group}} - \text{cf})</td>
<td>- ALL - includes all stack types - VPC - All VPC resources - NAT - NAT servers for the VPC - SG - Security Groups - IAM - IAM roles used instances and CodeDeploy - ELB - Elastic Load Balancer - ASGLC - AutoScaling Group and Launch Configuration - CW - CloudWatch monitoring - RDS - Databases, options groups, parameter groups - RDSREPLICA - Replica/slave database instances - S3 - S3 buckets - CONFIG - Config Rules</td>
<td></td>
</tr>
<tr>
<td></td>
<td>If this stack is application specific:</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>({\text{appname_construct}} - {\text{stack_group}} - \text{cf})</td>
<td></td>
<td></td>
</tr>
<tr>
<td>CloudFormation ChangeSet</td>
<td>If this stack is universal and not application specific:</td>
<td>Date Generated should follow: YYYYMMDDHHMM</td>
<td>admints-prod-level4-201704201100-cset or takeasweater-prod-201704201100-cset</td>
</tr>
<tr>
<td></td>
<td>({\text{account_naming_construct}} - {\text{dateGenerated}} - \text{cset})</td>
<td>Any additional information about the change set should be included in the description of the changeset</td>
<td></td>
</tr>
<tr>
<td></td>
<td>If this stack is application specific:</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>({\text{appname_construct}} - {\text{dateGenerated}} - \text{cset})</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
## AWS Account Items

<table>
<thead>
<tr>
<th>AWS Resource</th>
<th>Resource Name</th>
<th>Comment</th>
<th>Example</th>
</tr>
</thead>
</table>
| AWS Account Name | {{account_naming_construct}} | | admints-dev-standard
| | | Not all P-env’s require a separate account
| | admints-prod-level4 |
| AWS Account Email | huit-cloudops-awsaccounts + {{teamName}} + {{environment}}@calists.harvard.edu | Environment should be one of:
| | P - prod
| | P-1 - stage
| | P-2 - test
| | P-3 - dev
| | huit-cloudops-awsaccounts+admints-dev@calists.harvard.edu |
| | Not all P-env’s require a separate account |

*There is a 64 character limit on the AWS Account Email Address form input*

## VPC Resource Naming Standards

<table>
<thead>
<tr>
<th>AWS Resource</th>
<th>Resource Name</th>
<th>Comment</th>
<th>Example</th>
</tr>
</thead>
</table>
| VPCs | {{account_naming_construct}} - vpc | level4
| | | for Level 4 VPC’s inside non-l4 specific accounts
| | | bcdr
| | | only for bcdr exclusive accounts
| | admints-dev-standard-vpc
| | admints-stage-standard-vpc
| | admints-prod-standard-vpc
| | admints-prod-level4-vpc
| | admints-prod-bcdr-vpc |

| Subnets | {{account_name_construct}} - {{subnetType}} - {{routeType}} - {{az}} - {{number}} | Subnet Type should be one of:
| | app
| | elb
| | db
| | cache
| | nat
| | web
| | data |
| Number: Starts at 1, only increments if we have outgrown that AZ’s subnet |
| admints-dev-standard-app-private-1a-1
| admints-stage-standard-elb-public-1b-1
| admints-prod-standard-db-private-1c-1
| admints-prod-level4-web-private-1d-1 |

| VPC Peering Link | {{account_naming_construct_of_target}} - {{scope_of_target}} - vpc-peerlink | The items to be written should be that of the remote end of the VPC Peering Link |
| Example link between admints and sharedservices |
| admints-prod-standard-vpc-peerlink
| sharedservices-prod-standard-vpc-peerlink |
Route Tables

```latex
\{\text{account\_naming\_construct}\} - \{\text{routeType}\} - \text{rt} \# \{\text{zone}\}
```

Route type is one of the following:
- public
- private

⚠️ If the route table is distinct for each AZ (e.g. you are routing to different NATs), you must add the following Zone

Zone should be #! (e.g. 1a, 1b, 1c, 1d, 1e)

```
admints-dev-standard-private-rt-1a
admints-dev-standard-public-rt
```

DHCP Option Sets

```latex
\{\text{account\_naming\_construct}\} - \text{dhcpoptions}
```

Please Reference VPC Setup: DHCP Option Sets Required Settings

```
admints-dev-standard-dhcpoptions
```

Network ACL

```latex
\{\text{account\_naming\_construct}\} - \{\text{nac1Type}\} - \text{nacl}
```

NACL Types should reference why they exist (dnsout, etc)

⚠️ Please note that these are not required and not great to use

```
admints-dev-dnsout-nacl
```

Virtual Private Gateway

```latex
\{\text{account\_naming\_construct}\} - \{\text{scope}\} - \text{vgw}
```

Scope should be one of (only if colocated in a "standard" account)
- level4
  - for Level 4 VPC’s inside non-l4 specific accounts
- bcdr
  - only for bcdr exclusive accounts

⚠️ This should match the VPC naming

```
```

EC2 Resource Naming Standards

<table>
<thead>
<tr>
<th>AWS Resource</th>
<th>Resource Name</th>
<th>Comment</th>
<th>Example</th>
</tr>
</thead>
<tbody>
<tr>
<td>Elastic Load Balancer</td>
<td>{\text{appname_construct}} - elb</td>
<td>All new ELBs should be internal</td>
<td>takeasweater-dev-elb</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>takeasweater-stage-web-elb</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>takeasweater-prod-web1-elb</td>
</tr>
<tr>
<td>Launch Configuration</td>
<td>{\text{appname_construct}} - {\text{dateGenerated}} - lc</td>
<td>DateGenerated only required if manually created</td>
<td>car-prod-web-20151041708-icc</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Date should be in the form: YYYYMMDDHHMM</td>
<td>maximo-dev-nfs-201511091000-icc</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Hours should be in 24 hour format</td>
<td>fastcat-prod-app-ic</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>coursecatalog-prod-app-ic</td>
</tr>
<tr>
<td>AutoScaling</td>
<td>{\text{appname_construct}} - asg</td>
<td></td>
<td>takeasweater-prod-app-asg</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>takeasweater-prod-asg</td>
</tr>
</tbody>
</table>
### Security Groups

Account level Security Group resources:

<table>
<thead>
<tr>
<th>Resource Name</th>
<th>Comment</th>
</tr>
</thead>
<tbody>
<tr>
<td>sg</td>
<td></td>
</tr>
</tbody>
</table>

Resource Name can be one of:

- elb
- efs
- nlb
- instance
- db
- mgmt

If multiple are needed, increment number.

Account level resource name:

- nat

### Instances

Account level instance resources:

<table>
<thead>
<tr>
<th>Resource Name</th>
<th>Comment</th>
</tr>
</thead>
<tbody>
<tr>
<td>nat</td>
<td></td>
</tr>
</tbody>
</table>

Account level Instance Scope:

- nat

If generated by an Autoscaling Group add "-ec2asg"

### AMIs

<table>
<thead>
<tr>
<th>Resource Name</th>
<th>Comment</th>
</tr>
</thead>
<tbody>
<tr>
<td>-ami</td>
<td></td>
</tr>
</tbody>
</table>

### SSH Pem Keys

<table>
<thead>
<tr>
<th>Resource Name</th>
<th>Comment</th>
</tr>
</thead>
<tbody>
<tr>
<td>-pem</td>
<td>Date should be in the form: YYYYMMDD</td>
</tr>
</tbody>
</table>

.pem will be added by AWS on download

### S3 Resource Naming Standards

<table>
<thead>
<tr>
<th>AWS Resource</th>
<th>Resource Name</th>
<th>Comment</th>
<th>Example</th>
</tr>
</thead>
</table>
| S3 Buckets   | ((teamName)) - ((environment)) - ((scope)) - bucket | Environment should be one of:
- P = prod
- P-1 = stage
- P-2 = test
- P-3 = dev
Scope can be one or the following
- elblogs | admints-dev-bucket |
|              |               |         | admints-dev-elblogs-bucket |

### Lambda Function Naming Standard

<table>
<thead>
<tr>
<th>AWS Resource</th>
<th>Resource Name</th>
<th>Comment</th>
<th>Example</th>
</tr>
</thead>
</table>
| Lambda Function | ((appname_construct)) - ((function_scope)) - lambda-function | Function scope should be one of the following:
- autodeploy
- cfnoutputs | takeasweater-prod-autodeploy-lambda-function |
|              |               |         | campussvcs-prod-standard-cfoutputs-lambda-function |

### RDS Resource Naming Standards

<table>
<thead>
<tr>
<th>AWS Resource</th>
<th>Resource Name</th>
<th>Comment</th>
<th>Example</th>
</tr>
</thead>
</table>
| RDS (DB Instance Name) | ((appname_construct)) – ((db_type)) – ((deployment_type)) | DB Type is one of the following:  
- oracle  
- mysql  
- postgres  
- maria  
- aurora  
Deployment Type is one of the following:  
- mutliz  
- standalone  
- slave  
  - Used when making Read Slaves in RDS | takeawesweater-prod-oracle-standalone  
takeawesweater-prod-mysql-multiaz  
takeawesweater-prod-aurora-slave |
|----------------------|------------------------------------------------|--------------------------------------------------|
| RDS (Subnet Group)   | ((appname_construct)) – ((db_type)) – ((deployment_type)) – subnetgroup | DB Type is one of the following:  
- oracle  
- mysql  
- postgres  
- maria  
- aurora  
Deployment Type is one of the following:  
- mutliz  
- standalone  
- slave  
  - Used when making Read Slaves in RDS | takeawesweater-prod-oracle-standalone-subnetgroup  
takeawesweater-prod-mysql-multiaz-subnetgroup  
takeawesweater-prod-aurora-slave-subnetgroup |
| RDS (Option Group)   | ((appname_construct)) – ((db_type)) – ((deployment_type)) – optiongroup | DB Type is one of the following:  
- oracle  
- mysql  
- postgres  
- maria  
- aurora  
Deployment Type is one of the following:  
- mutliz  
- standalone  
- slave  
  - Used when making Read Slaves in RDS | takeawesweater-prod-oracle-standalone-optiongroup  
takeawesweater-prod-mysql-multiaz-optiongroup  
takeawesweater-prod-aurora-slave-optiongroup |
| RDS (Parameter Group) | ((appname_construct)) – ((db_type)) – ((deployment_type)) – paramgroup | DB Type is one of the following:  
- oracle  
- mysql  
- postgres  
- maria  
- aurora  
Deployment Type is one of the following:  
- mutliz  
- standalone  
- slave  
  - Used when making Read Slaves in RDS | takeawesweater-prod-oracle-standalone-paramgroup  
takeawesweater-prod-mysql-multiaz-paramgroup  
takeawesweater-prod-aurora-slave-paramgroup |

Elasticache Resource Naming Standards
### ElastiCache

<table>
<thead>
<tr>
<th>AWS Resource</th>
<th>Resource Name</th>
<th>Comment</th>
<th>Example</th>
</tr>
</thead>
<tbody>
<tr>
<td>ElastiCache</td>
<td>((\text{appname_construct}) - (\text{cache_type}) - (\text{deployment_type}))</td>
<td>Cache Type is one of the following: * memcached * redis Deployment Type is one of the following: * multiaz * standalone * slave • Used when making Redis Read Slaves in RDS</td>
<td>takeasweater-prod-memcached-standalone takeasweater-prod-redis-multiaz takeasweater-prod-redis-slave</td>
</tr>
</tbody>
</table>

### CodeDeploy Resource Naming Standards

<table>
<thead>
<tr>
<th>AWS Resource</th>
<th>Resource Name</th>
<th>Comment</th>
<th>Example</th>
</tr>
</thead>
<tbody>
<tr>
<td>Code Deploy Deployment Group</td>
<td>((\text{appname_construct}) - (\text{scope}) - \text{dg})</td>
<td>Scope Can be one of the following: * app * web * cache</td>
<td>takeasweater-prod-app-dg takeasweater-dev-web-dg</td>
</tr>
</tbody>
</table>

### IAM Resource Naming Standards

<table>
<thead>
<tr>
<th>AWS Resource</th>
<th>Resource Name</th>
<th>Comment</th>
<th>Example</th>
</tr>
</thead>
<tbody>
<tr>
<td>IAM User</td>
<td>Administrators: * first_last Service Accounts: * service_name</td>
<td>IAM user names should mirror the users @harvard.edu email address</td>
<td>robert_ruma thomas_vachon cloud_endure</td>
</tr>
<tr>
<td>IAM Group</td>
<td>((\text{appname_construct}) - (\text{group_purpose}) - \text{iam} - \text{group})</td>
<td>Application group purpose can be one of the following: * readonly * poweruser * administrator Account level group purpose can be one of the following: * forcemfa * cloudendure * administrator Option account level group access can be one of the following: * administrator * poweruser * readonly * isolated-by-tag</td>
<td>takeasweater-prod-readonly-iam-group takeasweater-dev-administrator-iam-group admints-dev-standard-forcemfa-iam-group</td>
</tr>
</tbody>
</table>
| IAM Role (via SAML) | `{{account_naming_construct}} - saml - {{group_purpose}} - iam - role@us-east-1` | Account group purpose can be one of the following:  
  - admin  
  - poweruser  
  - readonly  
  - (More can be added as required) | cloudhacks-dev-standard-saml-admin-iam-role@us-east-1  
  
  Note: There will be a standard naming structure for app level roles resources:  
  `{{account_naming_construct}} - saml - {{group_purpose}} - iam - role@us-east-1`  
  
  Account level group purpose can be one of the following:  
  - fullaccess  
  - readonly | takeasweater-prod-saml-readonly-iam-role@us-east-1 |
|---|---|---|---|
| IAM Roles | `{{appname_construct}} - {{role_purpose}} - iam - role` | Role Purpose can be one of the following:  
  - autodeploy  
  - s3access  
  - securitymonkey  
  - datapipeline  
  - ebssnapshot  
  - lambdaexecute | takeasweater-prod-autodeploy-iam-role  
  
  takeasweater-dev-s3access-iam-role |  
  
  | Instance Role | `{{appname_construct}} - {{role_type}} - iam - ec2role` | Role Purpose can be one of the following:  
  - app  
  - web  
  - cache  
  - master  
  - worker  
  - nfs | takeasweater-prod-app-iam-ec2role  
  
  takeasweater-dev-cache-iam-ec2role |  
  
  | IAM Policy | `{{appname_construct}} - {{product_used}} - {{level_of_access}} - iam - policy` | Product Used must be one of the AWS product names such as:  
  - ec2  
  - s3  
  - lambda  
  - codedeploy  
  - sqs  
  
  Account level product names can be one of the following:  
  - securitymonkey  
  - osaccounts  
  - cloudendure  
  - cfoutputs |  
  
  Level of Access must be one of the following:  
  - readonly  
  - readwrite  
  - isolated-by-tag | takeasweater-prod-s3-readwrite-iam-policy  
  
  takeasweater-dev-codedeploy-readonly-iam-policy  
  
  Note: Instance profiles are a collection of policies added to a role |  
  
  |
KMS

{{appname_construct}} - {{scope}} - {{type}} - kms

Context should be one of:
- standard
- level4 (for Level 4 Data)
- bcdr (only for bcdr exclusive accounts)

Type should be one of:
- ebs
- rds
- snowball
- s3
- redshift
- codecommit
- cloudtrail
- elastictranscoder
- ses

takeasweater-dev-standard-rds-kms
takeasweater-prod-standard-ebs-kms

SSL Certificates (for ELB or Cloudfront)

{{appname_construct}} - {{product_used}} - {{certificate_type}} - {{certificate_expiry}} - sslcert

Product Used must be one of the following AWS product names:
- elb
- cloudfront

Certificate Type should be one of the following:
- domain
- wildcard
- san

Certificate Expiry should be in the format "YYYYMM"

takeasweater-dev-elb-domain-201601-sslcert
takeasweater-prod-cloudfront-wildcard-201601-sslcert

Cloudwatch Resource Naming Standards

<table>
<thead>
<tr>
<th>AWS Resource</th>
<th>Resource Name</th>
<th>Comment</th>
<th>Example</th>
</tr>
</thead>
</table>
| CloudWatch Alarm | {{appname_construct}} - {{instance_id}} - {{alarm_product}} - {{alarm_metric}} - {{check_status}} - cw-alarm | Alarm Product Should be one of the following (lowercased):
  - elb
  - asg
  - rds
  - billing
  - ebs
  - lambda
  - s3
  - sns
  - linuxsystem (instance_id must be used)

Alarm Metric should match the metric name in lowercase dashed style such as:
- http-4xx
- http-5xx
- volume-queue-length
- cpu
- memory
- swap
- diskspace

Check Status should be one of:
- high
- low | takeasweater-prod-elb-http-4xx-high-cw-alarm
takeasweater-prod-rds-max-connections-high-cw-alarm |
SNS Resource Naming Standards

<table>
<thead>
<tr>
<th>AWS Resource</th>
<th>Resource Name</th>
<th>Comment</th>
<th>Example</th>
</tr>
</thead>
<tbody>
<tr>
<td>SNS Topic</td>
<td>([appname_construct]) - ([topic_name]) - ([topic_subscription_type]) - sns-topic</td>
<td>Topic name should be a lowercased name for the topic. Topic Subscription Type should be one of the following: email, http, sqs, sms, application, lambda.</td>
<td>takeasweater-prod-elbcw-email-sns-topic, takeasweater-prod-codedeploy-lambda-sns-topic</td>
</tr>
</tbody>
</table>

SOS Resource Naming Standards

<table>
<thead>
<tr>
<th>AWS Resource</th>
<th>Resource Name</th>
<th>Comment</th>
<th>Example</th>
</tr>
</thead>
<tbody>
<tr>
<td>SQS Queue</td>
<td>([appname_construct]) - ([queue_name]) - sqs-queue</td>
<td>The Queue name should be a lowercased name for the queue.</td>
<td>takeasweater-prod-myqueue-name-sqs-queue</td>
</tr>
</tbody>
</table>

CloudOps Tool Naming Standards

Package Names for CAR

<table>
<thead>
<tr>
<th>Type</th>
<th>Format</th>
<th>Example</th>
<th>Known Standards Reference</th>
</tr>
</thead>
</table>

DEB

Splunk Index Naming

<table>
<thead>
<tr>
<th>Format</th>
<th>Comment</th>
<th>Example</th>
</tr>
</thead>
</table>

⚠️ Version and architecture are added in the Jenkins build.
cloud - {{ teamName }} - {{ environment }}

Optional for level 4 isolation:

cloud - {{ teamName }} - {{ inner-group }} - {{ environment }}

Environment should be one of:
- prod
- uat
- stage
- test
- dev

Values should be lowercase, no spaces

Already created indexes:
- cloud-acts-dev
- cloud-acts-prod
- cloud-admints-dev
- cloud-admints-test
- cloud-admints-uat
- cloud-admints-prod
- cloud-campusservices-dev
- cloud-campusservices-prod
- cloud-cloudops-dev
- cloud-cloudops-prod
- cloud-lts-dev
- cloud-lts-prod
- cloud-sharedservices-dev
- cloud-sharedservices-prod

Optional for level 4:
- cloud-admints-researchadmin-prod
- cloud-admints-researchadmin-dev

---

Git Repository

<table>
<thead>
<tr>
<th>Type</th>
<th>Format</th>
<th>Comment</th>
<th>Example</th>
</tr>
</thead>
<tbody>
<tr>
<td>Automation Code Repositories</td>
<td>hcd0 - {{ code context }} - {{ resource name }}</td>
<td>Code context should be one of the following:</td>
<td>hcd0-ansible-role-apache</td>
</tr>
<tr>
<td></td>
<td></td>
<td>• ansible-role</td>
<td>hcd0-ansible-role-postfix</td>
</tr>
<tr>
<td></td>
<td></td>
<td>• cloudformation</td>
<td>hcd0-ansible-role-splunk</td>
</tr>
<tr>
<td></td>
<td></td>
<td><img src="image" alt="This name should mirror any associated CAR package" /></td>
<td>hcd0-cloudformation-rds</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Resource name should be the name of the resource the automation code manages.</td>
<td></td>
</tr>
<tr>
<td>Tools Repositories</td>
<td>hcd0 - {{ tool name }}</td>
<td>Tool name is specific to the function of the tool.</td>
<td>hcd0-splunk-certs</td>
</tr>
<tr>
<td></td>
<td></td>
<td>It should reflect a meaningful name.</td>
<td>hcd0-cloud-os-accounts</td>
</tr>
<tr>
<td></td>
<td></td>
<td><img src="image" alt="This name should mirror any associated CAR package" /></td>
<td></td>
</tr>
</tbody>
</table>
### Application Repositories

<table>
<thead>
<tr>
<th>Format</th>
<th>Comment</th>
<th>Example</th>
</tr>
</thead>
<tbody>
<tr>
<td>hcd - applications - {{teamName}} - {{appName}}</td>
<td>teamName should be the application team name providing support. appName should be the application name without the environment.</td>
<td>hcd-applications-admints-gmas</td>
</tr>
</tbody>
</table>

### Feature Branch

<table>
<thead>
<tr>
<th>Format</th>
<th>Comment</th>
<th>Example</th>
</tr>
</thead>
<tbody>
<tr>
<td>feature/CLOPS-xxx_short_description</td>
<td>CLOPS-xxx should be the JIRA Story or task used to track progress.</td>
<td>feature/CLOPS-2713_adding_environment_tag</td>
</tr>
</tbody>
</table>

### Bugfix Branch

<table>
<thead>
<tr>
<th>Format</th>
<th>Comment</th>
<th>Example</th>
</tr>
</thead>
<tbody>
<tr>
<td>bugfix/CLOPS-xxx_short_description</td>
<td>CLOPS-xxx should be the JIRA Story or task used to track progress.</td>
<td>bugfix/CLOPS-2713_fix_race_condition</td>
</tr>
</tbody>
</table>

### DynamoDB Tables

<table>
<thead>
<tr>
<th>Format</th>
<th>Comment</th>
<th>Example</th>
</tr>
</thead>
<tbody>
<tr>
<td>{{appName_construct}} - {{ context }}</td>
<td>Context may be arbitrary but reflects the use of the table.</td>
<td>pidash-dev-osaccounts pidash-prod-osaccounts</td>
</tr>
</tbody>
</table>

### CloudWatch Log Groups

<table>
<thead>
<tr>
<th>Format</th>
<th>Comment</th>
<th>Example</th>
</tr>
</thead>
<tbody>
<tr>
<td>{{appName_construct}} - {{ source }} - {{ context }}</td>
<td>Source should be one of the following (the AWS service the CloudWatch Log is configured with): - lambda - vpc Context may be arbitrary but reflects the context of the use of the log.</td>
<td>vpc-flowlog lambda-dev-autodeploy lambda-prod-autodeploy pidash-dev-lambda-deleteebssnapshot pidash-prod-solr-lambda-deleteebssnapshot pidash-prod-lambda-ebssnaphsha pi pidash-prod-lambda-ebssnaphsha</td>
</tr>
</tbody>
</table>

### Elastic File System

<table>
<thead>
<tr>
<th>Format</th>
<th>Comment</th>
<th>Example</th>
</tr>
</thead>
<tbody>
<tr>
<td>{{appName_construct}} - efs</td>
<td>Context may be arbitrary but reflects the purpose of the storage</td>
<td>github-prod-backup-efs maximo-test-efs</td>
</tr>
</tbody>
</table>

### Ansible Tower

<table>
<thead>
<tr>
<th>Type</th>
<th>Format</th>
<th>Comment</th>
<th>Example</th>
</tr>
</thead>
<tbody>
<tr>
<td>Projects (Generic role)</td>
<td>{{ organization }} - ansible - role - {{ name }} - {{ environment }}</td>
<td>Organization is the Tower organization the resource is a member of Name should be a unique identifying term related to the technology in use and closely related to the SCM repository name containing the code Environment refers to the software deployment environment and may be one of: - dev - test - stage - prod The directory layout should follow Ansible best practices</td>
<td>huit-ansible-role-carbonblack-prod huit-ansible-role-rhn-prod</td>
</tr>
<tr>
<td>Projects (Application specific)</td>
<td>Organization is the Tower organization the resource is a member of. Appname construct should be closely related to the SCM repository name containing the code. Scope is optional and should be used to add context to the function of the project and may be one of: - infra - deploy - role - playbook The directory layout should follow Ansible best practices.</td>
<td>huit-ansible-deploy-cours ecatalog-prod-solr1 huit-ansible-ask-prod-auth</td>
<td></td>
</tr>
<tr>
<td>Inventories (Cloud)</td>
<td>Applies to Amazon EC2 sourced inventories</td>
<td>acts-dev-standard acts-prod-standard ebs-prod-level4</td>
<td></td>
</tr>
<tr>
<td>Inventories (On-prem)</td>
<td>teamName should be the organizational team providing operational support. Environment refers to the software deployment environment and may be one of: - dev - test - stage - prod</td>
<td>onprem-its-csi-dev-hosts</td>
<td></td>
</tr>
<tr>
<td>Job Templates</td>
<td>Inventory name is the name of the inventory used in the template configuration. Template function should be a meaningful name describing the functionality of the template. Environment refers to the software deployment environment and may be one of: - dev - test - stage - prod</td>
<td>atsea-dev-standard-deploy-account-sshkey-prod atsea-dev-standard-scan-template-prod</td>
<td></td>
</tr>
<tr>
<td>Workflow Job Templates</td>
<td>Function should be a meaningful name describing the functionality of the template. Environment refers to the software deployment environment and may be one of: - dev - test - stage - prod</td>
<td>all-aws-accounts-deploy-account-sshkey-prod all-aws-accounts-scan-template-prod</td>
<td></td>
</tr>
<tr>
<td>API User Accounts (Generic)</td>
<td>Organization is the Tower organization the resource is a member of. Function is optional and is a meaningful name describing the functionality of the API user. Access permission must be one of the following: - ro - rw</td>
<td>huit_api_ro huit_api_rw huit_billing_api_ro</td>
<td></td>
</tr>
<tr>
<td>API User Accounts (Inventory specific)</td>
<td>Inventory name is the name of the inventory the user will have access from. Access permission must be one of the following: - ro - rw</td>
<td>cloudhacks_dev_standard_d_api_rw sharedsvcs_prd_standard_d_api_rw</td>
<td></td>
</tr>
<tr>
<td>Version</td>
<td>Date/Time</td>
<td>Author</td>
<td></td>
</tr>
<tr>
<td>---------</td>
<td>----------------</td>
<td>------------------</td>
<td></td>
</tr>
<tr>
<td>v. 13</td>
<td>Oct 31, 2017 12:56</td>
<td>Ruma, Robert D</td>
<td></td>
</tr>
<tr>
<td>v. 12</td>
<td>Oct 31, 2017 12:04</td>
<td>Ruma, Robert D</td>
<td></td>
</tr>
<tr>
<td>v. 11</td>
<td>Jun 06, 2017 11:25</td>
<td>Vachon, Thomas</td>
<td></td>
</tr>
<tr>
<td>v. 10</td>
<td>Jun 03, 2017 12:31</td>
<td>Ruma, Robert D</td>
<td></td>
</tr>
<tr>
<td>v. 9</td>
<td>Apr 20, 2017 11:24</td>
<td>Vachon, Thomas</td>
<td></td>
</tr>
<tr>
<td>v. 8</td>
<td>Oct 19, 2016 09:14</td>
<td>Ruma, Robert D</td>
<td></td>
</tr>
<tr>
<td>v. 7</td>
<td>Sep 28, 2016 21:00</td>
<td>Ruma, Robert D</td>
<td></td>
</tr>
<tr>
<td>v. 6</td>
<td>Sep 28, 2016 20:49</td>
<td>Ruma, Robert D</td>
<td></td>
</tr>
<tr>
<td>v. 5</td>
<td>Sep 28, 2016 20:41</td>
<td>Ruma, Robert D</td>
<td></td>
</tr>
<tr>
<td>v. 4</td>
<td>Sep 28, 2016 13:58</td>
<td>Ruma, Robert D</td>
<td></td>
</tr>
<tr>
<td>v. 3</td>
<td>Sep 27, 2016 11:15</td>
<td>Ruma, Robert D</td>
<td></td>
</tr>
<tr>
<td>v. 2</td>
<td>Aug 25, 2016 17:44</td>
<td>Ferry, Mark</td>
<td>Added isolated-by-tag for the policy where access is limited by tags</td>
</tr>
<tr>
<td>v. 1</td>
<td>Aug 11, 2016 13:31</td>
<td>Vachon, Thomas</td>
<td></td>
</tr>
</tbody>
</table>