# New Server infrastructure created by Prakash

* Disclaimer
* Provisioning of servers
  + Configuration
* Servers
  + Api-(1-?)
  + Chace-1
  + Graph-Legacy-(1-?)
  + Graph-Util-(1-?)
  + Graph-(1-?)
  + Job-(1-4)
  + Mobile-(1-?)
  + Queue-(1-3)
  + Rest-(1-?)
  + Search-1
  + Swift-(1-?)
  + Web-(1-?)
  + Zoo-(1-3)

Disclaimer

Since there has been no documentation done prior to this document, in regards to servers domain, there probably are errors and things that has not been described or documented as intended when the server was first setup.

\*\*The question mark (?) mean that the server group is setup as an auto-scaling group.

Provisioning of servers

Servers started in the AWS cloud are normally all part of a security group. Based on that security group each server gets provisioned with puppet.

Many of the specific settings etc. has its place in the puppet recipes.

Structure of the puppet “code”/recipes:

.

├── manifests

│ ├── classes

│ │ ├── sg\_api.pp

│ │ ├── sg\_common.pp

│ │ ├── sg\_common\_front\_interface.pp

│ │ ├── sg\_common\_java.pp

│ │ ├── sg\_common\_mcollective.pp

│ │ ├── sg\_mobile.pp

│ │ ├── sg\_osm-rest.pp

│ │ ├── sg\_service.pp

│ │ ├── sg\_swift.pp

│ │ ├── sg\_test.pp

│ │ ├── sg\_vpn.pp

│ │ └── sg\_web.pp

│ ├── modules.pp

│ ├── nodes.pp

│ └── site.pp

├── modules

│ ├── apache

│ ├── apt

│ ├── backup-mysql-cnf

│ ├── base-cron

│ ├── base-dhclient

│ ├── base-dns-register

│ ├── base-history

│ ├── base-hostname

│ ├── base-hosts

│ ├── base-iptables

│ ├── base-java-jce

│ ├── base-java-tzupdate

│ ├── base-motd

│ ├── base-profile

│ ├── base-puppet-cert

│ ├── base-ssh-keys

│ ├── base-sudo

│ ├── base-sysstat

...

...

│ ├── user-mysql

│ ├── user-webmaster

│ └── vcsrepo

└── puppet.conf

**Configuration**

There are many places where the configuration is located. On the servers and in the application. Here is a short description of where to find what. Configuration for the server groups is also handled/described in each of the sections below.

The configuration for the different server applications like tomcat or equinox can be found under each of the puppet modules for these applications.

F.x. module "service-tomcat" - under this directory there is a "manifest" and a "file" directory. In the "file" directory you will find all the files used in the init.pp file for the setting up the module.

In here some of the files are postfixed with fully qualified domain name - fqdn, by doing this we have the ability to tweak a single server in a pool of servers, with different configurations. Files can also be postfixed with a security group like - job, there files are often the default values a sever pool should have at least.

"service-tomcat" model directory:

├── files

│ ├── Resound.jks

│ ├── FreePushCertificate.p12

│ ├── FreePushCertificate.p12\_old

│ ├── FreePushDevCertificate.p12

│ ├── ProPushCertificate.p12

│ ├── ProPushDevCertificate.p12

│ ├── context.xml

│ ├── context.xml-memcached

│ ├── context.xml-persistent

│ ├── context.xml-persistent-native

│ ├── jmxremote.access

│ ├── jmxremote.password

│ ├── keystore-test

│ ├── logging.properties

│ ├── memcached-session-manager

│ │ ├── context.xml

│ │ ├── memcached-session-manager-1.6.2.jar

│ │ ├── memcached-session-manager-1.6.4.jar

│ │ ├── memcached-session-manager-tc6-1.6.2.jar

│ │ ├── memcached-session-manager-tc6-1.6.4.jar

│ │ ├── memcached-session-manager-tc7-1.6.4.jar

│ │ └── spymemcached-2.8.2.jar

│ ├── mysql-connector-java-5.1.24-bin.jar

│ ├── smart-access-log-valve.jar

│ ├── smart-jdbc-store.jar

│ ├── smart-syslog-access-log-valve.jar

│ ├── syslog-access-log-valve-tomcat-7.jar

│ ├── tomcat-default-7

│ ├── tomcat-default-7-api

│ ├── tomcat-default-7-graph-legacy

│ ├── tomcat-default-7-job

│ ...

│ ├── tomcat-default-7-job-4.aws.resound.com

│ ├── tomcat-default-7-master.aws.resound.com

│ ├── tomcat-default-7-mobile

│ ├── tomcat-default-7-service

│ ├── tomcat-default-7-swift

│ ├── tomcat-default-7-test

│ ├── tomcat-default-7-test-3.ec2-int.resound.com

│ ├── tomcat-default-7-web

│ ├── tomcat-default-7-web-test.aws.resound.com

│ ├── tomcat-init-7

│ ├── tomcat-log-maintenance-cron

│ ├── tomcat-log-maintenance.sh

│ ├── tomcat-logrotate

│ ├── tomcat-manager.tar.gz

│ ├── tomcat-process-dump.sh

│ ├── tomcat-users.xml

│ └── tomcat-users.xml-test

├── manifests

│ ├── deploy.pp

│ ├── group.pp

│ ├── init.pp

│ ├── log-maintenance.pp

│ ├── package.pp

│ ├── service.pp

│ ├── session-manager.pp

│ └── user.pp

└── templates

├── server.xml-7-api

├── server.xml-7-graph-legacy

├── server.xml-7-job

...

├── server.xml-7-test-4.aws.resound.com

├── server.xml-7-web

└── server.xml-7-web-test.aws.resound.com

ResoundApplication.java

Servers

**Api-(1-?)**

Server function:

Running Tomcat.

Details:

This servers host the 3rd. Party API’s.

**Chace-1**

Server function:

Running a Mysql database.

Details:

Sessions from a web server are being stored here, so it is accable if auto-scaling kills a machine that a user’s session was "living" on.

To see the settings for the server, please refer to the puppet recipes, found in

“ ./manifests/nodes.pp, line:-> 355 -> if $fqdn =~ /cache-1.aws/ { ...“

The server can be considered as a single point of failure. Hench, this server needs to be up at all times. Since we have moved away from “wicked” framework and are using stateless AngularJ these setting for the “sticky” transactions might be ok to remove, making the setup not relying on a single server for the web part.

**Graph-Legacy-(1-?)**

Server function:

Running Tomcat.

Details:

**Graph-Util-(1-?)**

Server function:

Running equinox.

Details:

This server group is split into two groups. Graph-util (1-4) and Graph-util (5-8).

Graph-util (1-4):

To get an idea of what is running on the instances check the "ServerGroups.cfg" found in the "~/equinox/appconfig/" directory.

ServerGroups define which service and jobs that can be run on the instance.

root@graph-util-1:~/equinox/appconfig# cat ServerGroups.cfg

# Component enabled/disabled based on this configuration file (comma separated; should be at least one; will work with none; see com.resound.common.aws.ServerGroup)

# Enable components in these groups

enableGroups=graphConsumer,job,services,logConsumer

# Disable components in these groups

disableGroups=

# Run all groups except explicitly disabled

runAllGroups=false

Here is the .cfg file for graph-util-1.

root@graph-util-1:~/equinox/appconfig# ll

total 144

drwxr-xr-x 2 equinox equinox 4096 Sep 18 11:19 ./

drwxr-xr-x 8 equinox equinox 4096 Sep 3 19:59 ../

-rw-r--r-- 1 equinox equinox 201 Sep 3 19:59 Cluster-config.cfg

-rw-r--r-- 1 equinox equinox 82 Sep 3 19:59 com.resound.live.impl.WorkoutRepositoryImpl.cfg

-rw-r--r-- 1 equinox equinox 195 Sep 3 19:59 ConnectionManager-connection.cfg

-rw-r--r-- 1 equinox equinox 0 Sep 3 19:59 DeferUtilProxy.cfg

-rw-r--r-- 1 equinox equinox 58 Sep 3 20:06 EventLoggingConsumer.cfg

-rw-r--r-- 1 equinox equinox 832 Sep 10 20:38 HostManager-ConnectionRead.cfg

-rw-r--r-- 1 equinox equinox 772 Sep 10 20:38 HostManager-ConnectionWrite.cfg

-rw-r--r-- 1 equinox equinox 182 Sep 3 19:59 HTTPUtil.cfg

-r--r----- 1 equinox equinox 2367 Sep 3 19:59 jwk.json

-rw-r--r-- 1 equinox equinox 746 Sep 3 19:59 KafkaConsumerHandler.cfg

-rw-r--r-- 1 equinox equinox 104 Sep 3 19:59 KafkaMessageService.cfg

-rw-r--r-- 1 equinox equinox 60 Sep 3 20:06 KeyGenerator-RememberMe.cfg

-rw-r--r-- 1 equinox equinox 59 Sep 3 20:06 KeyGenerator-Session.cfg

-rw-r--r-- 1 equinox equinox 14 Sep 3 19:59 log4j.cfg

-rw-r--r-- 1 equinox equinox 50110 Sep 18 08:49 log4j.xml

-rw-r--r-- 1 equinox equinox 991 Sep 3 19:59 MailMessageMk2Consumer.cfg

-rw-r--r-- 1 equinox equinox 36 Sep 3 19:59 ManagedOrmLiteConnection-resound.cfg

-rw-r--r-- 1 equinox equinox 29 Sep 3 19:59 ManagedOrmLiteConnection-inbox.cfg

-rw-r--r-- 1 equinox equinox 27 Sep 3 19:59 ManagedOrmLiteConnection-push.cfg

-rw-r--r-- 1 equinox equinox 0 Sep 3 19:59 MessageServiceSelector.cfg

-rw-r--r-- 1 equinox equinox 0 Sep 3 19:59 MessageSystem.cfg

-rw-r--r-- 1 equinox equinox 140 Sep 3 19:59 org.ops4j.pax.web.cfg

-rw-r--r-- 1 equinox equinox 43 Sep 3 20:06 QuartzScheduler.cfg

-rw-r--r-- 1 equinox equinox 374 Sep 18 11:19 ServerGroups.cfg

-rw-r--r-- 1 equinox equinox 26 Sep 3 19:59 ServiceHandler.cfg

-rw-r--r-- 1 equinox equinox 0 Sep 3 19:59 SSOConfigImpl.cfg

root@graph-util-1:~/equinox/appconfig#

There are also specific settings for each server found in the puppet recipes.

"<puppet - code - git repo>/modules/service-equinox/manifests/group-graph-util.pp" - Here there are settings about what .cfg files is copied to which server instance. Be aware that if there are no .cfg file for a job running in equinox the job will not run. (This only apply if the job uses config files).

Graph-util (5-8):

This group is all running jobs related to the SSO, they all have same configuration.

ServerGroups.cfg file from graph-util-5:

root@graph-util-5:~# cat equinox/appconfig/ServerGroups.cfg

# Component enabled/disabled based on this configuration file (comma separated; should be at least one; will work with none; see com.resound.common.aws.ServerGroup)

# Enable components in these groups

enableGroups=ssoService,identityValidation,mmf\_entity\_sync

# Disable componentes in these groups

disableGroups=

# Run all groups except explicitly disabled

runAllGroups=false

root@graph-util-5:~#

.cfg files on graph-util-5:

root@graph-util-5:~# ll equinox/appconfig/

total 144

drwxr-xr-x 2 equinox equinox 4096 Sep 10 20:40 ./

drwxr-xr-x 8 equinox equinox 4096 Sep 1 09:33 ../

-rw-r--r-- 1 equinox equinox 201 Sep 1 09:33 Cluster-config.cfg

-rw-r--r-- 1 equinox equinox 84 Sep 1 09:33 com.resound.inbox.consumer.InboxCreateConsumer.cfg

-rw-r--r-- 1 equinox equinox 56 Sep 1 09:33 com.resound.inboxpush.consumer.InboxPushConsumer.cfg

-rw-r--r-- 1 equinox equinox 82 Sep 1 09:33 com.resound.live.impl.WorkoutRepositoryImpl.cfg

-rw-r--r-- 1 equinox equinox 215 Sep 1 09:33 com.resound.notificationsender.NotificationSenderConsumer.cfg

-rw-r--r-- 1 equinox equinox 195 Sep 10 04:49 ConnectionManager-connection.cfg

-rw-r--r-- 1 equinox equinox 0 Sep 1 09:33 DeferUtilProxy.cfg

-rw-r--r-- 1 equinox equinox 832 Sep 10 20:40 HostManager-ConnectionRead.cfg

-rw-r--r-- 1 equinox equinox 772 Sep 10 20:40 HostManager-ConnectionWrite.cfg

-rw-r--r-- 1 equinox equinox 182 Sep 1 09:33 HTTPUtil.cfg

-r--r----- 1 equinox equinox 2367 Sep 1 09:33 jwk.json

-rw-r--r-- 1 equinox equinox 747 Sep 4 08:44 KafkaConsumerHandler.cfg

-rw-r--r-- 1 equinox equinox 104 Sep 1 09:33 KafkaMessageService.cfg

-rw-r--r-- 1 equinox equinox 14 Sep 1 09:33 log4j.cfg

-rw-r--r-- 1 equinox equinox 50110 Sep 17 12:49 log4j.xml

-rw-r--r-- 1 equinox equinox 991 Sep 1 09:33 MailMessageMk2Consumer.cfg

-rw-r--r-- 1 equinox equinox 36 Sep 1 09:33 ManagedOrmLiteConnection-resound.cfg

-rw-r--r-- 1 equinox equinox 29 Sep 1 09:33 ManagedOrmLiteConnection-inbox.cfg

-rw-r--r-- 1 equinox equinox 27 Sep 1 09:33 ManagedOrmLiteConnection-push.cfg

-rw-r--r-- 1 equinox equinox 0 Sep 1 09:33 MessageServiceSelector.cfg

-rw-r--r-- 1 equinox equinox 0 Sep 1 09:33 MessageSystem.cfg

-rw-r--r-- 1 equinox equinox 30 Sep 1 09:33 NotificationCleanupJob.cfg

-rw-r--r-- 1 equinox equinox 140 Sep 1 09:33 org.ops4j.pax.web.cfg

-rw-r--r-- 1 equinox equinox 381 Sep 10 20:40 ServerGroups.cfg

-rw-r--r-- 1 equinox equinox 24 Sep 1 09:33 ServiceHandler.cfg

-rw-r--r-- 1 equinox equinox 0 Sep 1 09:33 SSOConfigImpl.cfg

root@graph-util-5:~#

**Graph-(1-?)**

Server function:

Running equinox.

Details:

**Job-(1-4)**

Server function:

Running Tomcat

Details:

**Mobile-(1-?)**

Server function:

Running Tomcat.

Details:

**Queue-(1-3)**

Server function:

Running Kafka.

Details:

**Rest-(1-?)**

Server function:

Running equinox.

Details:

**Search-1**

 Server function:

 Running Solr

Details:

Server is used to search for users by name or email.

**Swift-(1-?)**

Server function:

Running Tomcat.

Details:

**Web-(1-?)**

Server function:

Running Tomcat.

Details:

**Zoo-(1-3)**

Server function:

Running Zookeeper.

Details:

We have a cluster consisting of 3 nodes.

There is one leader and to followers.

It can be looked as a key value store.

Used by Kafka to store consumer state and topic state.

Each service that uses the topics use Helix lib to register in Zookeeper. Helix used for the auto-discovery of the services in Zookeeper.

Resound uses two ways of configuration, v1 and v2.

The v1 can be found here: /resound/config/instances/ (Remember to use the ZkCli.sh to connect to the cluster - and from there you can cd and ls your way to the mentioned path.)

The v1 configuration and the /consumers settings is backed up every 10 minutes. - Scheduled by cron. It makes a incremental backup - hence only new stuff.