install and use chocoatley

Docker – small containers for Linux only, can run in windows

Testing environments fast! Vagrant

install vagrant and virtual box - default everything

add system user variable VARGRANT\_DEFAULT\_PROVIDER = virtualbox

HASHICORP

packer - create images for all environments

**Jenkins:**

Hey Ken - So your thought on jenkins is we should use the one on the jump/jump.  Should we also create a slave for it to use instead of running stuff on the master?

to do it the right way yep. but--you really won't hit scaling issues until WAY later

so if you want to wait on creating a slave, no worries there it just keeps the interface fast

you won't run into issues until you have >100 jobs running all the time

ok.  good to know.  It looks like you have done your most recent work on the internal open stack one.

Were you having issues using the outside one?

use interanl the reason being you can't hit the instances from the external one

because alex/tom never added the route

external was there so i could do plumgrid calls

but internal was more conducive to spinning up instances etc

Windows slave – run powershell remote sessions from windows slave to windows servers you want to take action on.

**Setup a windows slave:**

On new slave server:

* install Java JDK on server
* create remote root dir, name and location doesn’t matter

In Jenkins:

1. manage jenkins, manage nodes(slaves), new node,
2. name = name of server (leave off domain)
3. # of executors should = # of processors
4. Labels – windows – used to load balance across slaves
5. Usage – as much as possible
6. Launch method – java web start
7. Availability default
8. go to jenkins, nodes, copy link to newslave
9. launch – little box, file launch as service

Note – for linux process is much simplier (still need to install Java SDK on slave server)

Note – Currently a windows slave setup on Phase 3 MGMT tenant.

Note – use label when creating new job to show what server(s) to run on.

**Add powershell plugin to Jenkins:**

manage jenkins, manage plugins, available – use filter and type powershell

**Security:** manage jenkins, global security

**Jenkins Internal:**

62.193.13.84– This one lives within Openstack on the Management tenant.  It has access to all of the instances via management network.  This is the one that you are supposed to use

ID: ubuntu, pw: I…4

**Jenkins External:**

**62.193.8.59** jenkins.oncaas.com**, but my favorite.**  – This one has access to the Plumgrid directors, Piston bootnode, and lives on one of the infrastructure servers.  To get to this instance from the infrastructure node, first SSH into 62.193.8.35, do NOT sudo –i.  Run sudo virt-manager, the KVM console will pop up, it’s named ubuntujump.  We call it the Ubuntu jump jump server.  The creds for the infrastructure nodes, like 62.193.8.35 is username: caasadmin, password: see pw manager.

If you would prefer to SSH (as you should lol) to 62.193.8.59, you can, from the 85.15.17.117 box, using creds:

admin

Plumgrid

This server has Jenkins, the Plumgrid API installed, all of the Openstack python clients, OMD (our Nagios monitoring platform), it’s pretty much awesome.

62.193.8.59 is also where Tom and Alex run ALL Plumgrid scripts from.

Start jenkins - service jenkins start

Jenkins logs - /var/log/jenkins

curl -X post <http://62.193.8.59/jenkins/job/Test-20140430/buildWithParameters>

curl -X post http://62.193.8.59/jenkins/job/Test-20140430/build --data-urlencode json='{"parameter": [{"name":"yaml", "value":""}]}'

**MONITORING**

Monitoring is also hosted on 62.193.8.59, you can hit it by going to monitoring.oncaas.com, the username is omdadmin, password is Interactive2014.

Server: Looks like the OMD package was installed: included nagios

<http://mathias-kettner.com/checkmk_install_with_omd.html>

/opt/omd/sites/openstack/

web site reference: (apache2 I assume)

/var/www/html/index.php

**clients**

check\_mk client

<https://mathias-kettner.de/checkmk_linuxagent.html>

snmp client

<https://mathias-kettner.de/checkmk_snmp.html>

**ELK**

current host location: 62.193.8.35 (INF03)

* sudo virt-manage

vm – elk (62.193.8.60)

Install:

install java sdk

install apache2 for Kibana webserver (could be nxigx)

moved to /var/www/html

Install ELK pieces from website:

<http://www.elasticsearch.org/overview/elkdownloads/>

Kibana – web interface default is 9200 (blocked at ININ), changed to port 80

Ellisticsearch port moved to 443 (/etc/yaml -> http.port

Logstash config - /etc/logstash/conf.d/ logstash.conf

ssh key to access boot server

logstash conf –

* components – grok -> similar to grep but more powerfull
* grok heroku – grok debugger
* logstash patterns
* can filter by program and timestamp

Shipper – clientside way to send to logstash

logstash start script:

ssh -t [admin@62.193.10.10](mailto:admin@62.193.10.10) "tail -F /var/log/cmessages" | /opt/logstash/bin/logstash -f /etc/logstash/conf.d/logstash.test.conf

ssh -t [admin@62.193.10.10](mailto:admin@62.193.10.10) "tail -F /var/log/cmessages" | /opt/logstash/bin/logstash -f /etc/logstash/conf.d/logstashGOOD.conf

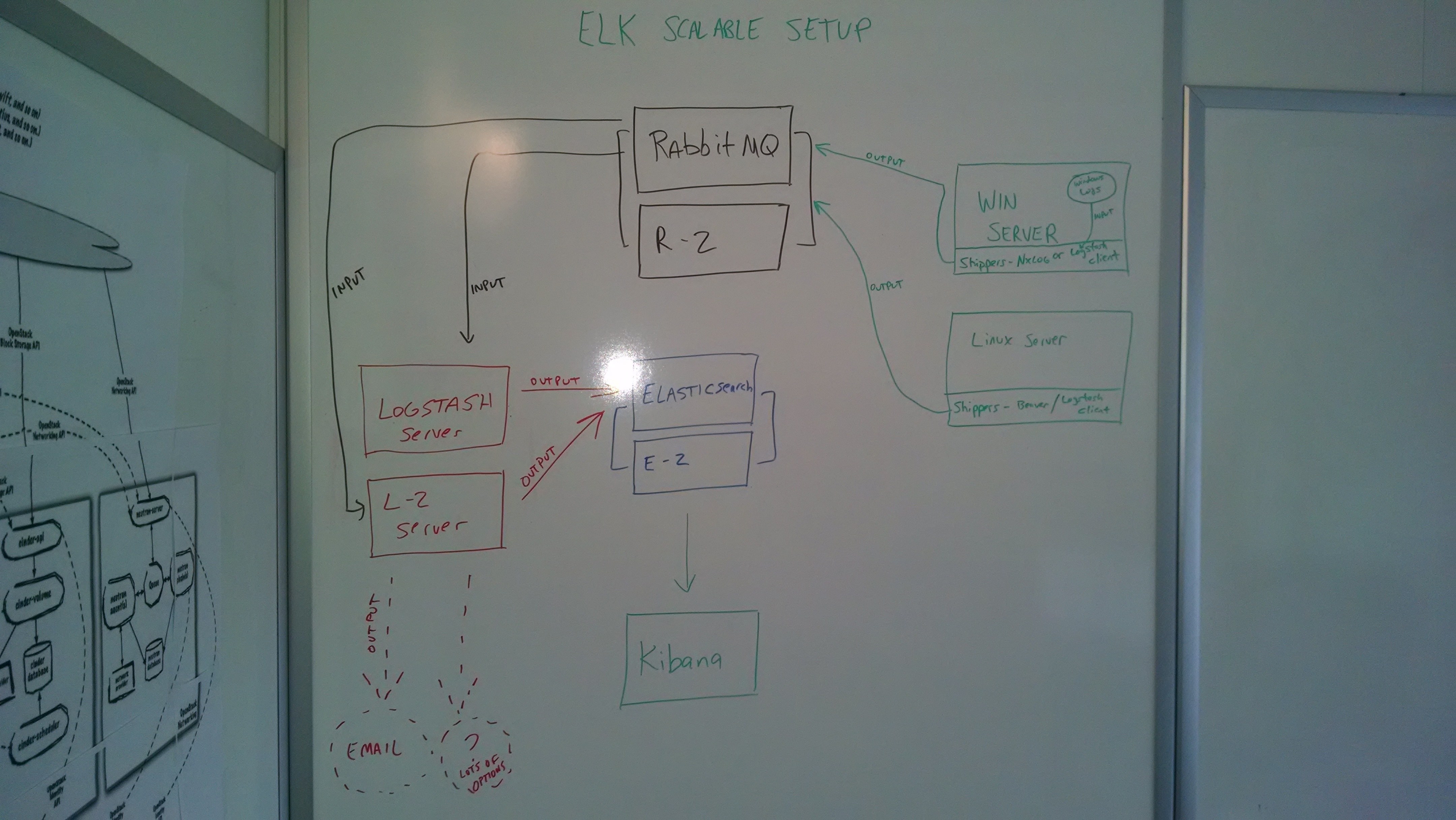
ElasticSearch Address:

ElasticSearch Frontend <http://62.193.8.60:443/_plugin/head/>

Kibana <http://62.193.8.60>

/usr/share/elasticsearch/bin/plugin -install mobz/elasticsearch-head

ELK scalable diagram:



Note – nxlog won’t talk directly with Reddis or Rabbit MQ.

Note – Reddis can be used instead of Rabbit MQ.

**Best Images located:**

c:\bestimages – on 117 jump server, location of the best images we currently have.

Misc ideas from Ken

SMNP Walk – troubleshooting tool

smpd – install on linux – default access is local host

PagerDuty – contact / alert people

Jira / Confluence – same company

new relic – expensive but really good monitoring for important servers

gomez – monitoring tool

checkmk – stores data in nagios DB, Nagios is a monitoring tool

wordpress – for small website / blogs