ISSN: 2277-9655
Impact Factor: 3.785

IJESRT

INTERNATIONAL JOURNAL OF ENGINEERING SCIENCES & RESEARCH TECHNOLOGY

PERFORMANCE SCALABILITY IMPROVEMENTS FOR ENTERPRISE BACKUP, RESTORE AND CLONING SOFTWARE FOR ORACLE DATABASE

Kulkarni Sameer Anil*

* DEPARTMENT OF ISE BMS COLLEGE OF ENGINEERING, BENGALURU 560019

DOI: 10.5281/zenodo.51462

ABSTRACT

In modern era most sensitive thing is Data, everything on the internet deals with data. Data needs security and protection, data should not corrupt in any case, and in case it gets corrupted we need to get back that data.

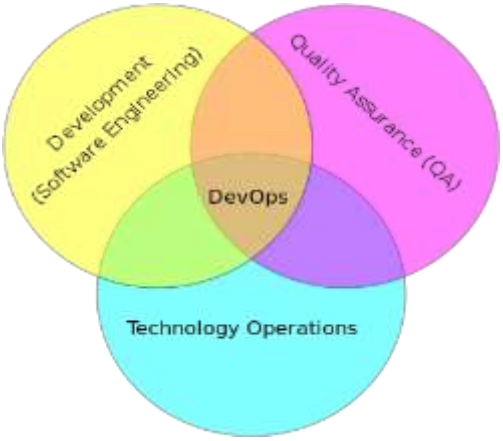
As I am working on oracle database this software supports various flours of oracle database like 11g, 12c as well as Real Application Cluster.

Challenge for me is to use this software in scaled environment so for development of the scaled environment I had many options like chef, puppet but because of simplicity I have chosen puppet, by using puppet I prepared one server, and have written entire code on that server and as per requirement I used puppet to simply push the puppet module which will install desired binaries as well as packages.

KEYWORDS: Puppet labs, Dev-ops, client-server paradigm.

INTRODUCTION

What is Dev Ops role?



As we can see in this Venn diagram dev-ops is a role which is intersection of all areas of IT departments. Dev ops engineers are mostly responsible for quick execution of agile model. Suppose there is a sale on www.abc.co.in website so Dev-ops engineer need to build an environment which will provide support for multiple requests coming from the client, and system should be stable and solid as well as error free enough to handle those requests.

Tools that can be used by Dev Ops people

1. **Chef**
2. **Puppet**

ISSN: 2277-9655
Impact Factor: 3.785

3. **Ensemble**
4. **Salt**

All of these tools provides facility to engineer to build more accurate, error free environment in less time.

PROJECT ARCHITECTURE

Server is running on windows host and as I am working on plugin side client or plugin is running on RHEL Linux host. So client will communicate with server by. This software facilitates user to take backup, restore and clone the oracle database.

I have done scalability performance analysis of this software previously at the end I found that, if you want to perform this task you need an error free environment and which can be built easily in quick time.

As part of scalability analysis we have spanned 200 databases across 20 hosts that means each host contains 10 databases. While doing this task I found that I am spending so much of time in preparing error free setup which can be used for this task.

Motivation to use puppet

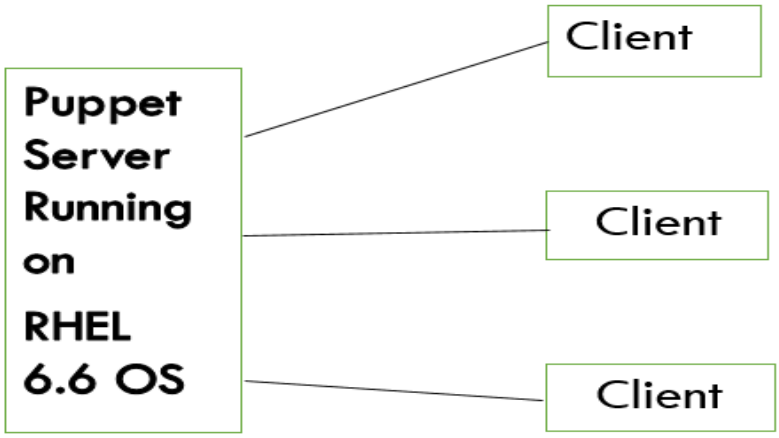


Figure: Proposed architecture of system.

Something about Puppet:

Puppet can manage configuration like unix and windows. The user provides description of system resources and current position, either using Puppet's declarative language or a Ruby. Puppet manifests or manifest file can store this information. Facter is the utility provided by puppet to discover the system, and compiles the Puppet manifests in operating system catalog which contains resources and various dependencies, which are applied against the target systems.

Puppet uses a custom language to describe system configuration, which can be either applied directly on the system, or compiled into a catalog and distributed to the target system via client–server paradigm (using a REST API), and the agent uses system specific providers to enforce the resource specified in the manifests. The resource abstraction layer enables administrators to describe the configuration in high-level terms, such as users, services and packages without the need to specify OS specific commands (such as rpm, yum, apt).

ISSN: 2277-9655
Impact Factor: 3.785

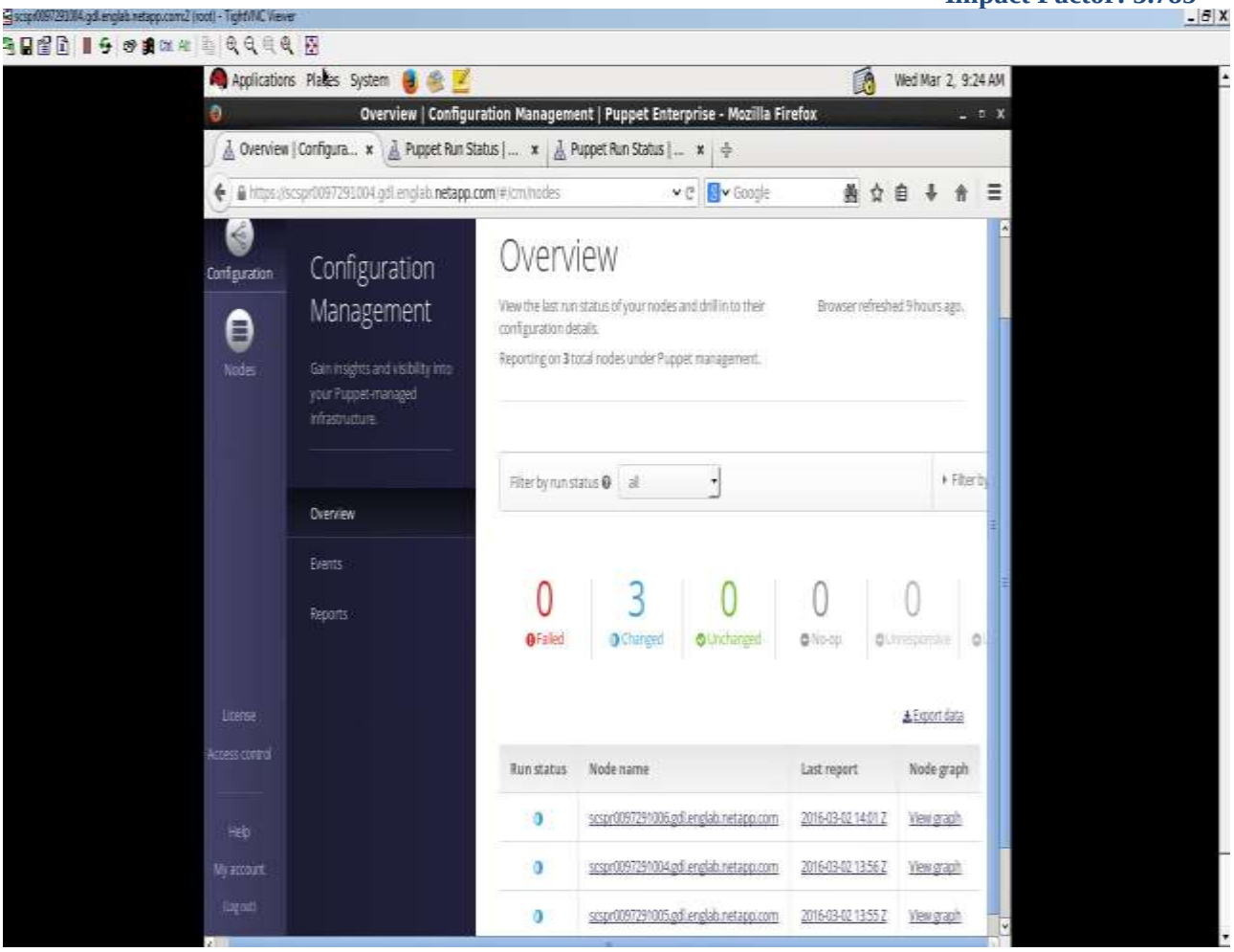


Figure: Screenshot of puppet server and managed hosts.

Idea of using puppet is if a software is in testing phase lots of builds will come, QA engineers have to test features again and again each time they have to configure the system. So to reduce this headache it will be simpler with Puppet. We are going to maintain single server with some managed hosts that is clients. So when new build will come we are just going to replace this build using puppet, puppet will push this build on all managed hosts.

Puppet is written on top of the ruby, and works with manifests, to use puppet it requires some knowledge of some ruby. In manifests directory of puppet we need to write different modules for different tasks, like to install the oracle database this approach will take care of installation of all dependencies, and setting up kernel parameters etc. While new build comes for testing puppet will just push this build on to clients, as puppet can install rpms packages and binaries.

Which will be very helpful in scaled environment in case of 20 or more than 20 hosts we can manage the infrastructure by using puppet.

ISSN: 2277-9655
Impact Factor: 3.785

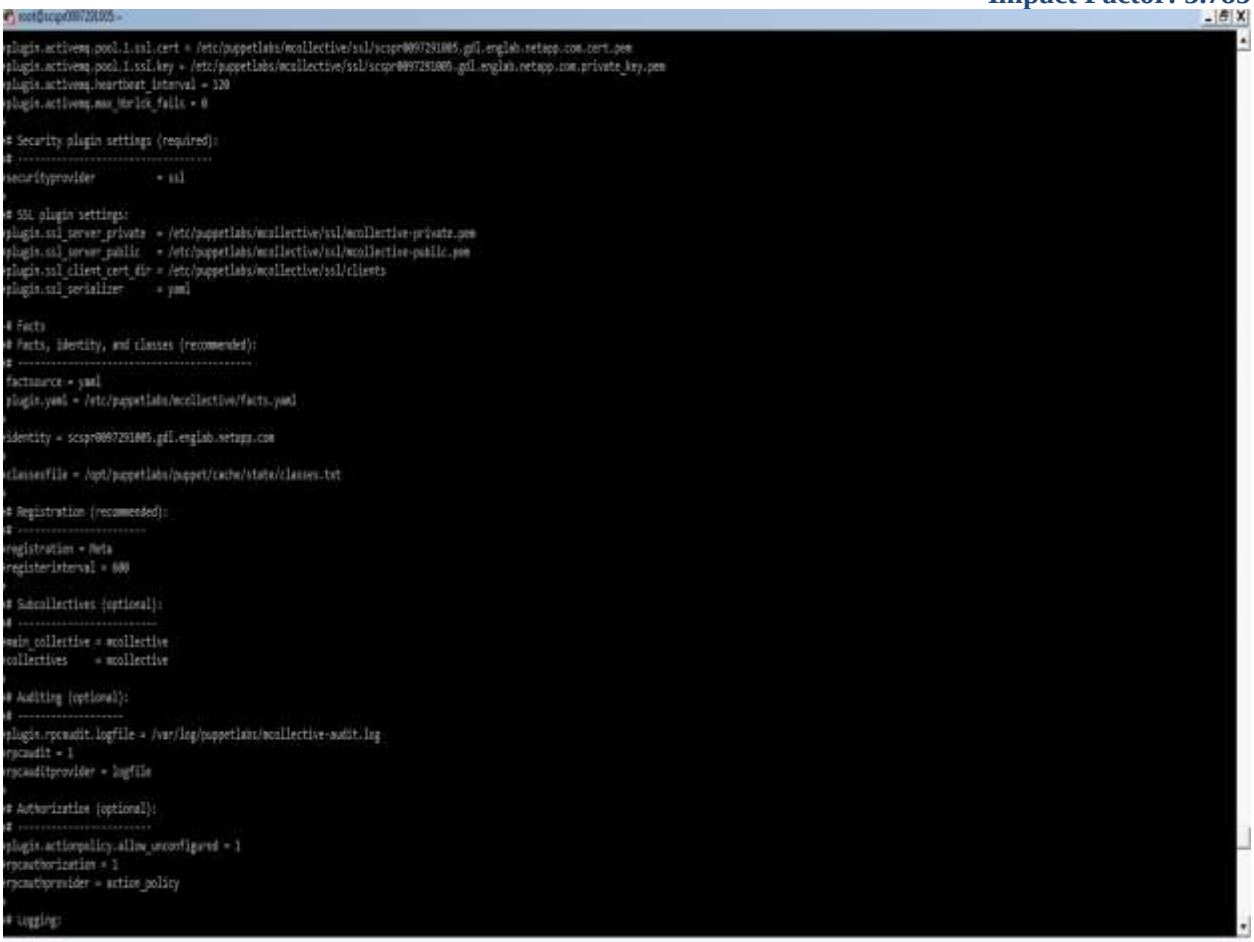


Figure: Installation of packages on client machine through server

IMPLIMENTATION

Phases involved in project:

1. Install prerequisites and edit configuration settings like kernel parameters etc.
2. Push oracle binaries to client machines
3. Install oracle binaries to client machines at time
4. Push software packages
5. Push various scripts which were written to automate operation
6. Perform desired operation in one shot on all the machines

ISSN: 2277-9655
Impact Factor: 3.785

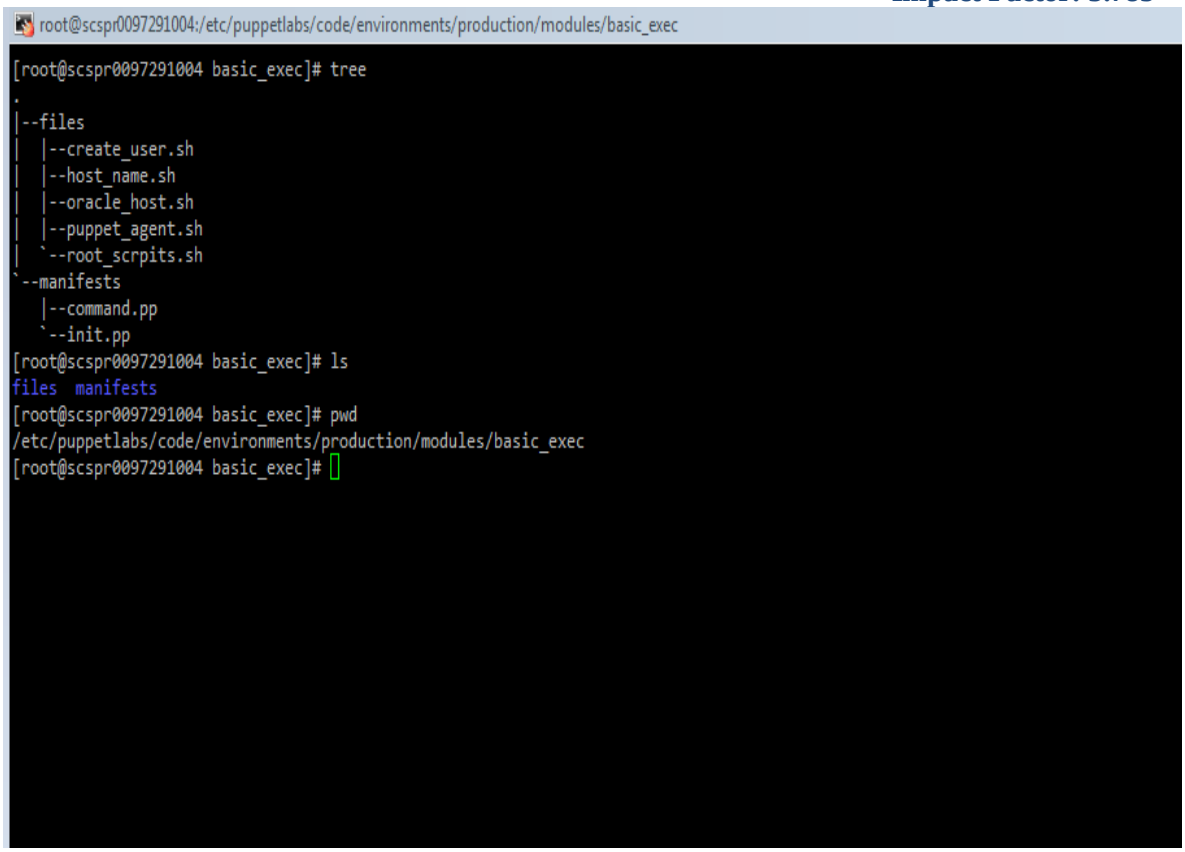


Figure: Directory Structure of puppet module

Above figure shows directory structure of puppet module which I have written for installation of oracle binaries, Basic module contains manifest files in which I have kept init.pp for module declaration and the other file is actual file which contains the puppet code for my module.
Another component of my module is file server which contains various scripts puppet has ability to execute scripts on the client node if the scripts are stored on file server location.

ISSN: 2277-9655
Impact Factor: 3.785

root@scspr0097291004:/etc/puppetlabs/code/environments/production/modules/basic_exec

```
file_line {'kernel.shmmni':
    line => 'kernel.shmmni = 4096',
    path => '/etc/sysctl.conf',
}

file_line {'kernel.shmmax':
    line => 'kernel.shmmax = 4398046511104',
    path => '/etc/sysctl.conf',
}

file_line {'kernel.shmall':
    line => 'kernel.shmall = 1073741824',
    path => '/etc/sysctl.conf',
}

file_line {'kernel.sem':
    line => 'kernel.sem = 250 32000 100 128',
    path => '/etc/sysctl.conf',
}

file_line {'fs.aio-max-nr':
    line => 'fs.aio-max-nr = 1048576',
    path => '/etc/sysctl.conf',
}

file_line {'fs.file-max':
    line => 'fs.file-max = 6815744',
    path => '/etc/sysctl.conf',
}

file_line {'net.ipv4.ip_local_port_range':
    line => 'net.ipv4.ip_local_port_range = 9000 65500',
    path => '/etc/sysctl.conf',
}

file_line {'net.core.rmem_default':
    line => 'net.core.rmem_default = 262144',
    path => '/etc/sysctl.conf',
}

file_line {'net.core.rmem_max':
```

Figure: Setting up prerequisites for oracle database

Above screenshot shows the settings of kernel parameters. Each parameter should be specified properly in /etc/sysctl.conf file.

For this feature puppet provides file_line feature with this feature it is very easy to add lines in any files, like this puppet facilitates mounting of directories on local system, user creation, group creation, execution of script from file server location of the module.

ISSN: 2277-9655
Impact Factor: 3.785

RESULTS:

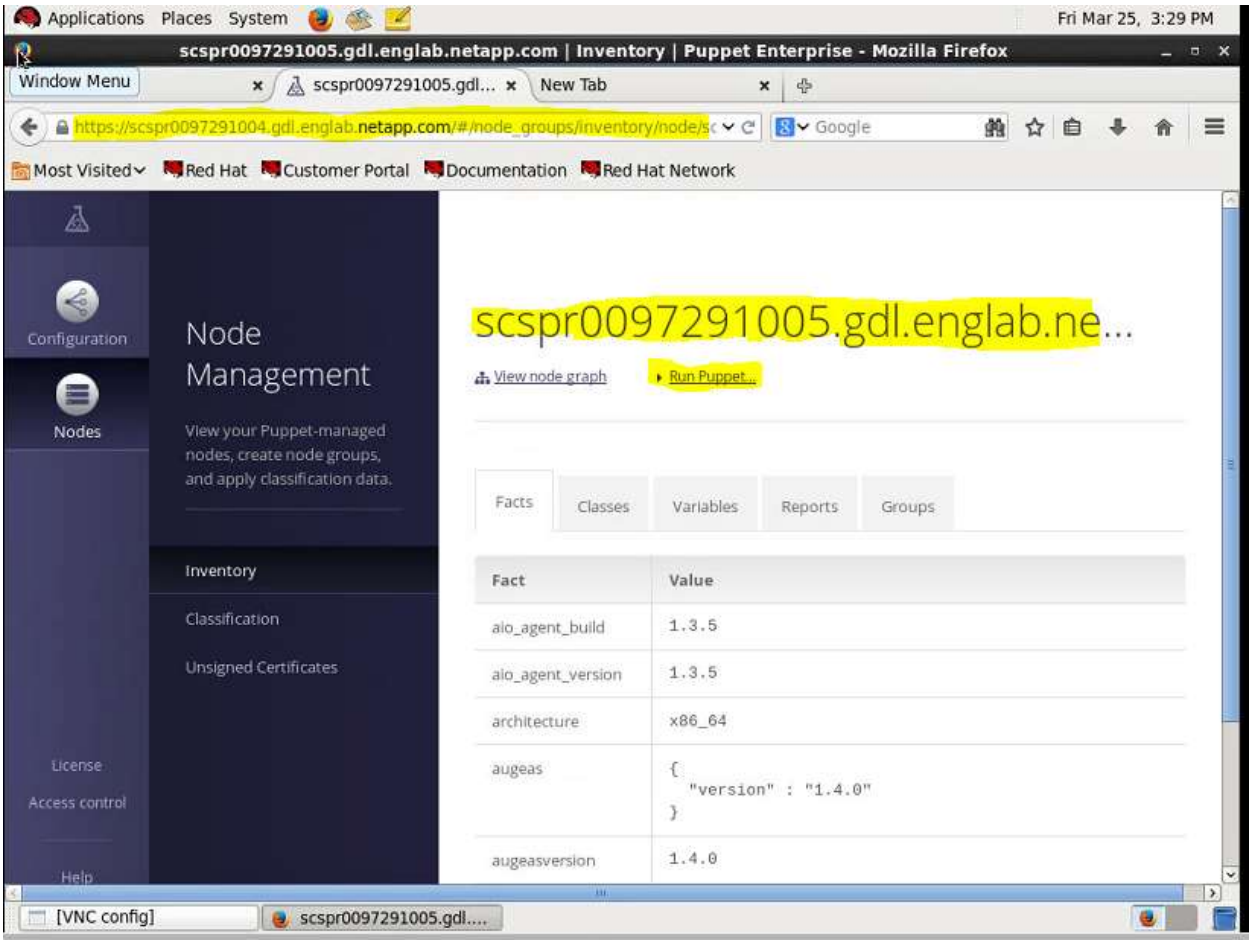


Figure: How to run puppet from server on the desired node

Like this on a single click puppet server will run the configuration settings on the desired node, by simply pushing proper modules, in other way if client is connected to server and client wants to install module on itself then client can simply run *puppet apply –t* which will install modules on the client if the client is certified by puppet server, but installation via server is bit easier so we should not log in to client for every time if any code change appears.

CONCLUSION

Hence with the help of puppet managing a software in scaled environment became easy and error free, which lead to more productivity. This approach resulted into error free development and deployment of the software.

REFERENCES

[1] Johannes Wettinger; Vasilios Andrikopoulos; Frank Leymann "Automated Capturing and Systematic Usage of DevOps Knowledge for Cloud Applications" Cloud Engineering (IC2E), 2015 IEEE International Conference
[2] Abubaker Wahaballa; Osman Wahballa; Majdi Abdellatief; Hu Xiong; Zhiguang Qin "Toward unified DevOps model" Software Engineering and Service Science (ICSESS), 2015 6th IEEE International
[3] Matthew A. McCarthy; Lorraine M. Herger; Shakil M. Khan; Brian M. Belgodere " Composable DevOps: Automated Ontology Based DevOps Maturity Analysis" Services Computing (SCC), 2015 IEEE International Conference