



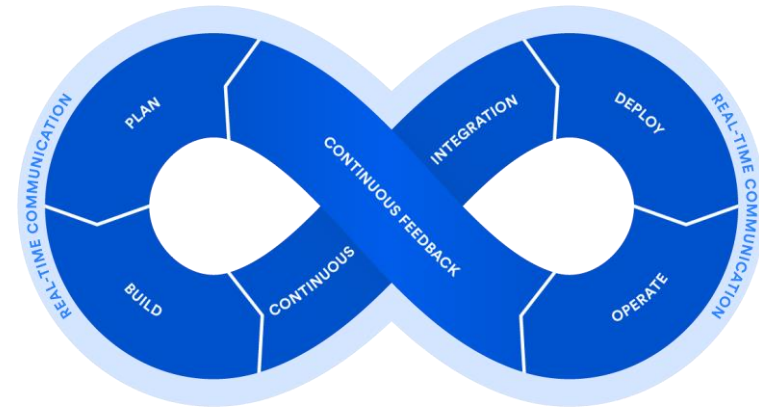
Automatización de Redes y Servicios usando Devops

Autor: MsC. Ing. Arian Trujillo Díaz





DevOps es un conjunto de prácticas que automatizan los procesos entre los equipos de desarrollo de software y TI para que puedan compilar, probar y publicar software con mayor rapidez y fiabilidad. El concepto de DevOps se basa en establecer una cultura de colaboración entre equipos que, tradicionalmente, trabajaban en grupos aislados...



En esencia, DevOps es una cultura, un movimiento, una filosofía.



Razones para Automatizar

- “Pereza” Profesional.
- Repetición.
- Consistencia.
- Garantía.
- Delegación al soporte.
- En las tareas complejas es mas fácil ... equivocarse.
- Es inevitable: Pensar en la escalabilidad.



Madurez del Modelo

Solicitudes Ad-hoc y (usualmente) Ejecución Ad-hoc.

Ad-Hoc

Se usan comandos basados en la experiencia/habilidad del usuario.

El procedimiento es documentado y se continua.

Repetible

Se usan comandos siguiendo documentación, Scripts/automatización parametrizada.

Todo el sistema esta "bien definido".

Definido

Entender el escenario
Los parámetros son secundarios.

Consistente, Enfoque estructurado.

Administrado

Infraestructura de Servicios con Soporte Automatizado.

Optimizado

Factor en el rendimiento y reutilización del código.



Gestores de Configuración



<https://forge.puppet.com/>





Script vs DSL

```
1 #!/bin/sh
2 # Test that a file exists, with given owner & group & permissio.
3 # Assumptions:
4 #   + The `stat` command is present on your system
5 #   + The owner and group given are alphanumeric, not UIDs
6 #   + Permissions are given in octal, and exclude SUID/SGID
7 #   + You have sufficient access to ALL directories above file
8
9 usage() {
10  echo 'Usage:  checkfile file owner group perms'
11  echo 'e.g.   checkfile /etc/passwd root root 644'
12  exit 1
13 }
14
15 case $# in
16 4) FILE="$1"
17   USER="$2"
18   GROUP="$3"
19   PERMS="$4"
20   ;;
21 *) usage()
22   ;;
23 esac
24
25 if [ ! test -e "${FILE}" ]; then
26  echo "${FILE} does not exist"
27  exit 11
28 fi
29
30 if [ ! test -f "${FILE}" ]; then
31  echo "${FILE} is not plain"
32  exit 11
33 fi
34
```

```
file { "/var/run/hello-world.txt":
  ensure => "present",
  mode => "644",
  owner => "mikec",
  group => "staff",
  content => "hello everyone",
}
```





Aprovechamiento de la Repetitividad

Configuración sin DSL

```
object Zone "master" {
  //...
}

object Zone "satellite region 1" {
  parent = "master"
  //...
}

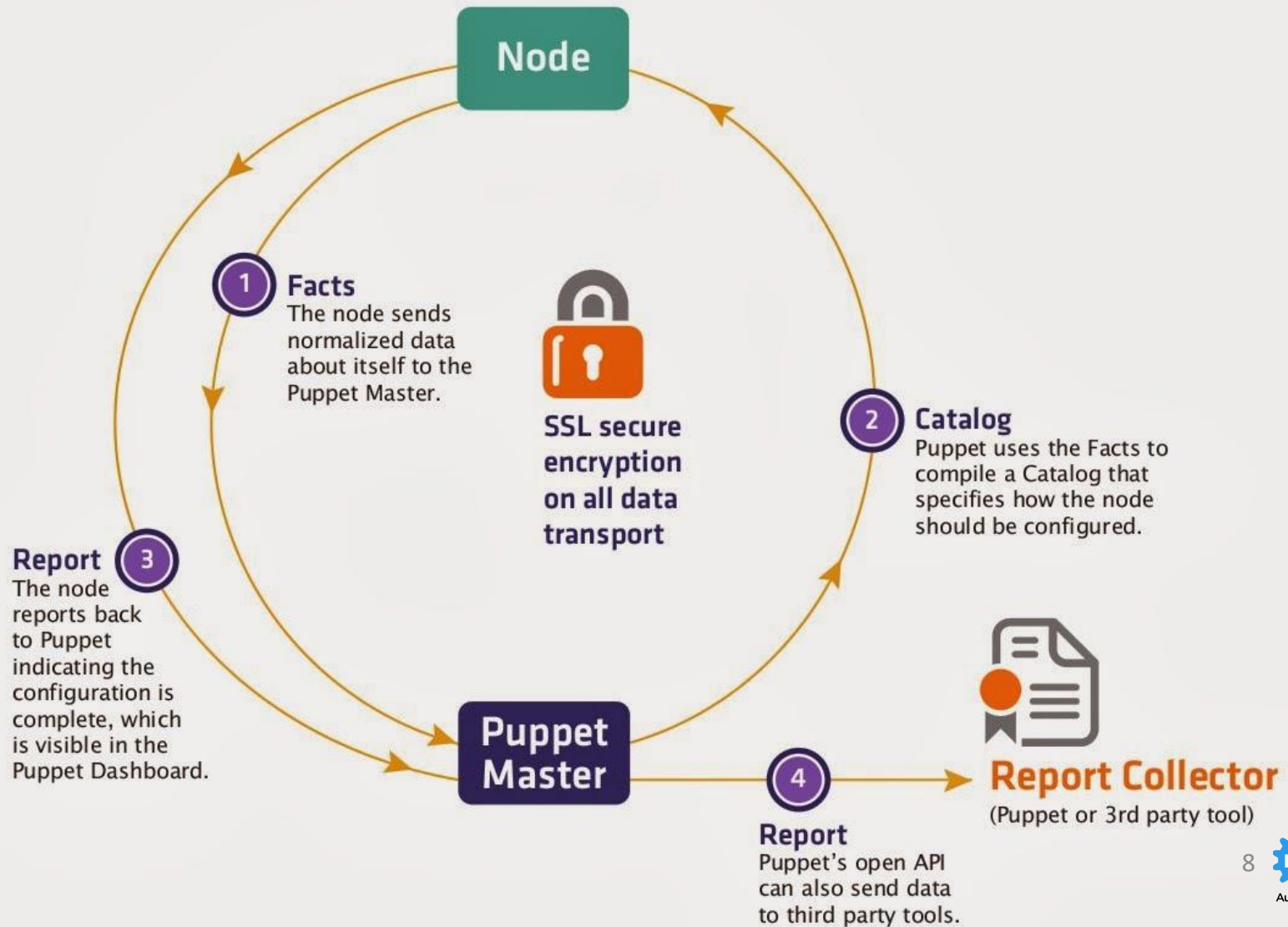
object Zone "satellite region 2" {
  parent = "master"
  //...
}
```

Configuración DSL (Puppet)

```
#Configure EndPoints
icinga2::object::endpoint { $::fqdn:
  host => $::ipaddress,
}
each($::monitoring::params::icinga_servers) [|Integer $index, String $value| {
  icinga2::object::endpoint { $value:
    host => $::monitoring::params::icinga_ipservers[$index],
  }
}
#Configure Zones
icinga2::object::zone { $::fqdn:
  endpoints => [ $fqdn ],
  parent    => 'master',
}
icinga2::object::zone { 'master':
  endpoints => $::monitoring::params::icinga_servers,
}
```



How Puppet Manages Data Flow for Individual Nodes



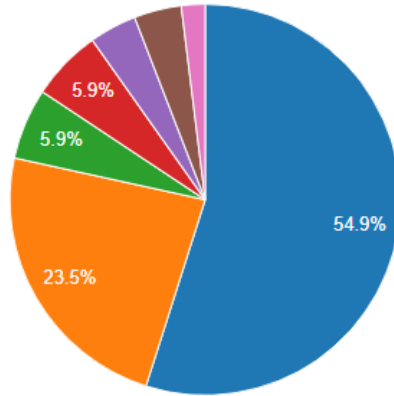


Configuración de Puppetserver

```
1  # Class: puppetdevserver
2  # =====
3  #
4  # Full description of class puppetdevserver here.
5  #
6  class puppetdevserver{
7
8      class {'::basesys':
9          uпрinfo_usage => 'Puppet dev server',
10         application   => 'puppetserver',
11         puppet_enabled => false;
12     }
13
14     class { ' '::puppetdb_server'::;}
15
16     class { ' '::puppetserver':
17         puppetdb_server => $::fqdn,
18
19     }
20 }
```



PuppetDB



■ 9.5 ■ 16.04 ■ 18.04 ■ 14.04 ■ 9.4 ■ 8.11 ■ 6.10

lsbdistrelease

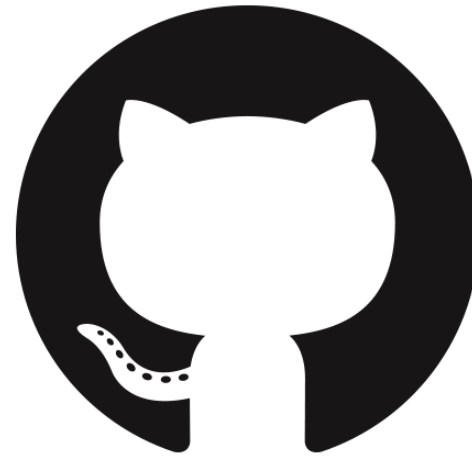
Show entries

Search:

Node	Value
wh.upr.edu.cu	8.11



Control de Versiones (CVS)







Reutilización de pki

```
class { '::icinga2::feature::api':  
  accept_commands => true,  
  accept_config   => true,  
  pki              => 'puppet',  
  endpoints       => {},  
  zones           => {},  
}
```

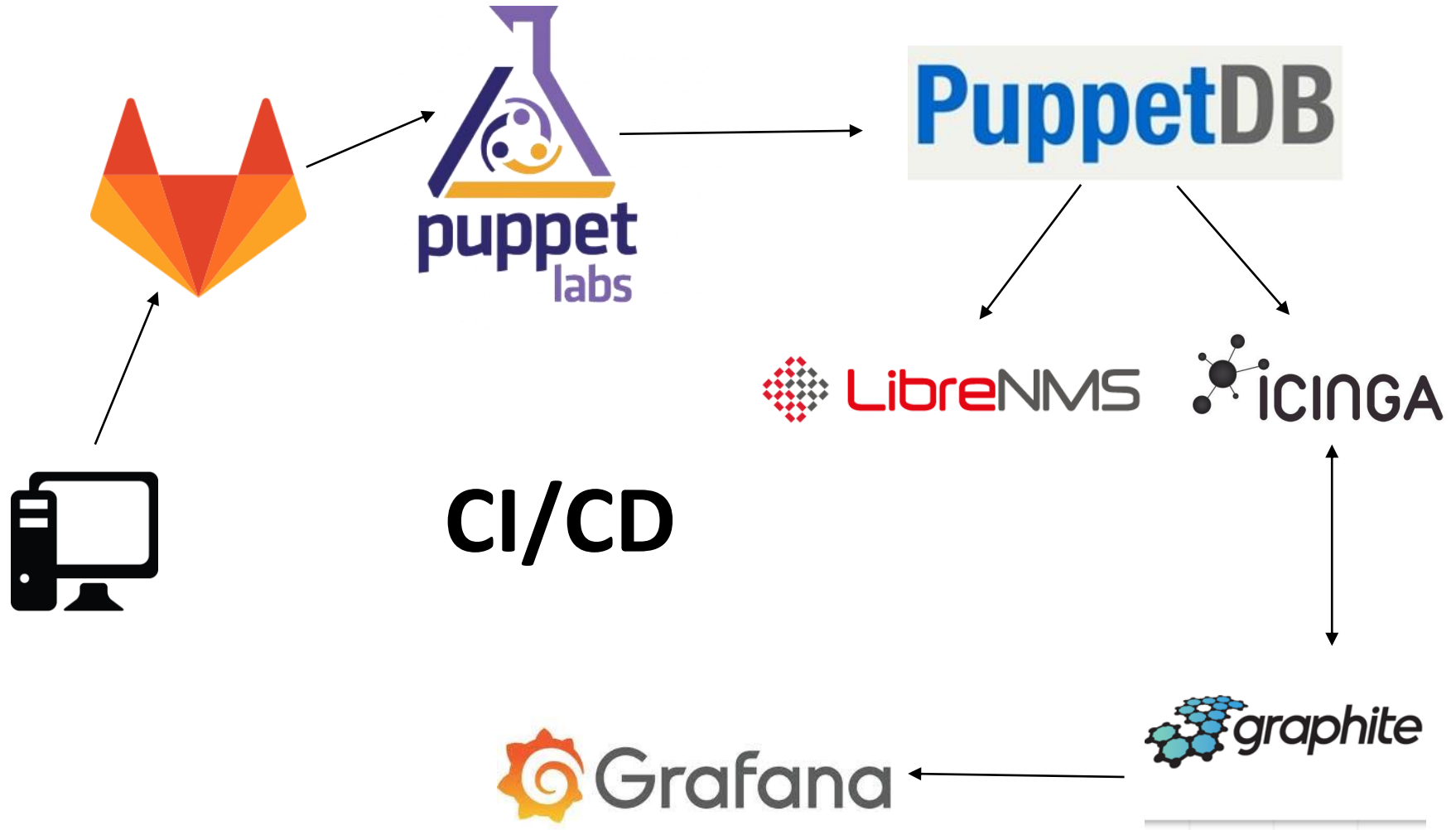


Herramientas de Graficado



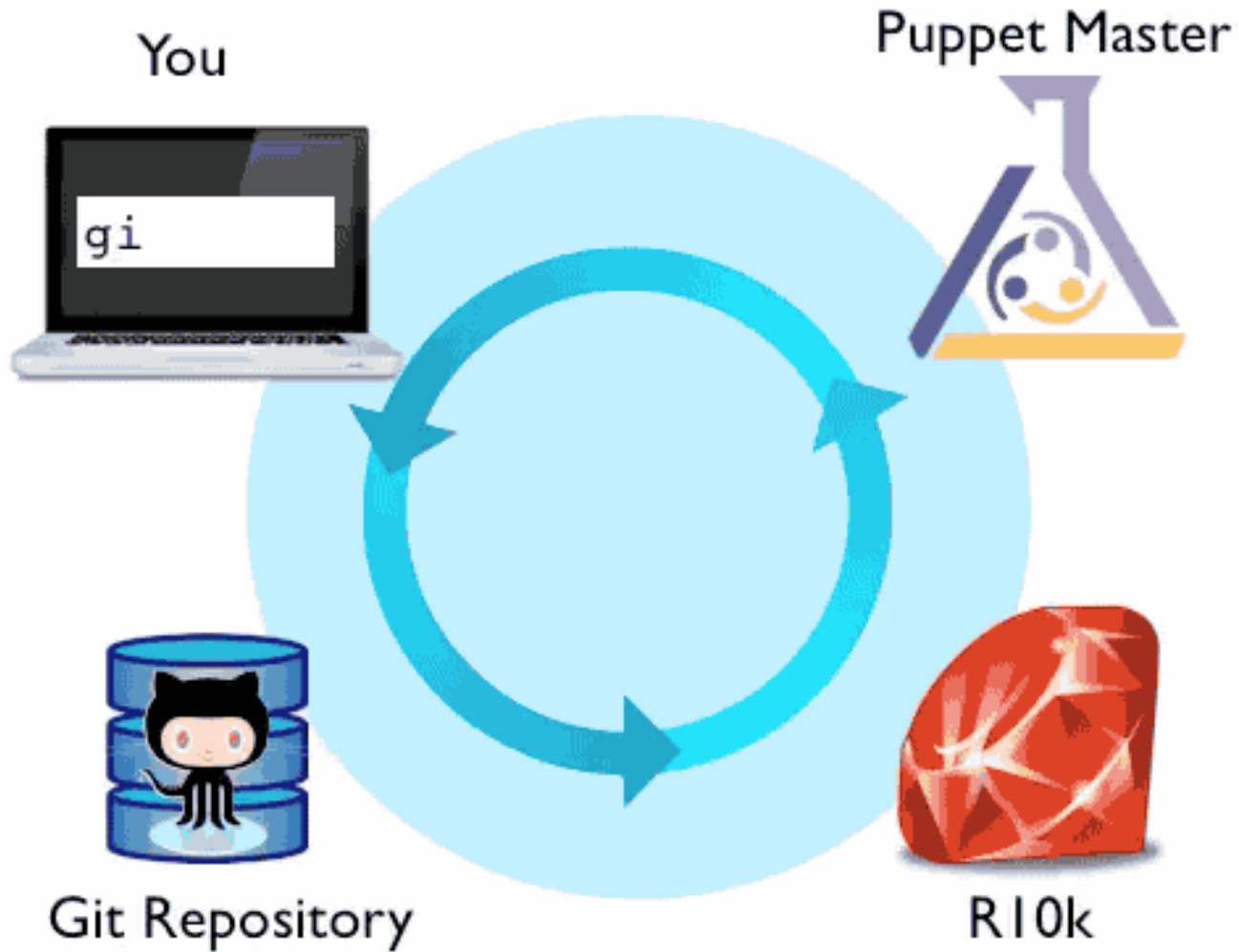


Escenario





Pipeline CI/CD





Integración icinga2-puppetDB

The screenshot shows the Icinga2 web interface with the 'Automation' menu selected. The 'Modify' tab is active, displaying the configuration for an import source named 'Puppetdb'. The configuration fields are as follows:

Field	Value
Import source name*	Puppetdb
Description	Conexión con el servidor puppetdb
Source Type*	Puppetdb module
Key column name	certname
API version*	v4: PuppetDB 2.3 (experimental), 3.0, 3.1, 3.2, 4.0 (PE 3.8 (experimental), 2015.2, ;
PuppetDB Server*	puppet-master.upr.edu.cu
Client Certificate*	master-icinga0.upr.edu.cu
Query type*	Nodes
Resource type*	Node (65)

At the bottom of the configuration form, there are two buttons: 'Store' and 'Delete'.

The left sidebar contains the following navigation items: Dashboard, Problems (86), Overview, Icinga Director (Hosts, Services, Commands, Notifications), Automation (selected), Activity log, Deployments, History, Graphite, Reporting, Documentation, System, Configuration, and arián.



Propiedades importadas de PuppetDB

The screenshot shows the Icinga web interface. The top navigation bar includes 'Sync rule', 'Modify', 'Properties', and 'History'. The left sidebar contains navigation options: Dashboard, Problems (82), Overview, Icinga Director (Hosts, Services, Commands, Notifications), Automation (Activity log, Deployments), History, Graphite, Reporting, Documentation, System, Configuration, and a user profile for 'arian'. The main content area is titled 'Sync properties: Puppet Nodes' and features a search bar and a '+ Add sync property rule' button. Below this is a table with the following data:

Source name	Source field	Destination	Prio
Puppetdb	\$(facts.ipaddress)	address	↓ ↑
Puppetdb	\$(facts.operatingsystem)	vars.os	↓ ↑
Puppetdb	\$(facts.fqdn)	display_name	↓ ↑
Puppetdb	hostalive	check_command	↓ ↑
Puppetdb	\$(classes)	vars.\$(classes)	↓ ↑
Puppetdb	\$(facts.operatingsystem)	groups	↓ ↑
Puppetdb	\$(facts.operatingsystem)	import	↓ ↑



Integración icinga2-graphite

```
#Graphite Feature Conf
class { '::icinga2::feature::graphite':
    host          => 'graphite.upr.edu.cu',
    port          => 2003,
    enable_send_thresholds => true,
    enable_send_metadata => true,
}
}
```



Configuración librenms_agent

lacnic31

06/10 DE MAYO 2019
REPUBLICA DOMINICANA

```
class monitoring::librenms_agent {
class { 'snmpd':
    package      => true,
    service      => true,
    allowed_hosts => [ 'localhost', '10.2.4.200/32', '10.2.9.0/24', '10.2.4.0/23' ],
    community    => 'UPRadmin4all',
    syslocation  => 'UPR, Nodo Central',
    syscontact   => 'di@upr.edu.cu'
}
}
```



Retos

- Curva de aprendizaje de Puppet.
- Comprensión del equipo de trabajo y del jefe de grupo.
- Resistencia al cambio.



Ventajas de un entorno Devops

- Aumenta el trabajo equipo y la colaboración.
- Disminuyen los errores humanos.
- Se automatizan los procesos.
- Se optimiza el proceso respaldo de información.
- Estandarización de la configuración.