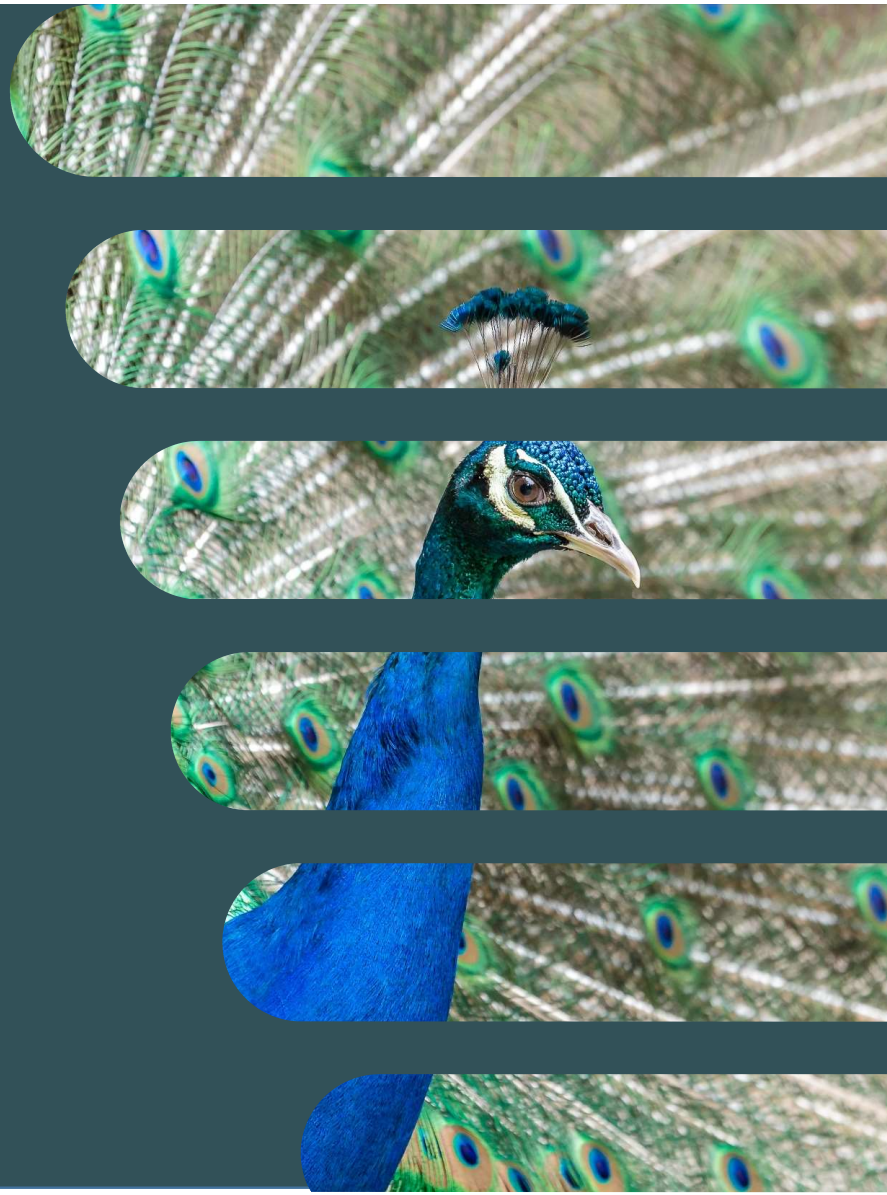
ZippyOPS
Making Automation Work
www.zippyops.com

What is Puppet ?



ZippyOPS
Making Automation Work
www.zippyops.com

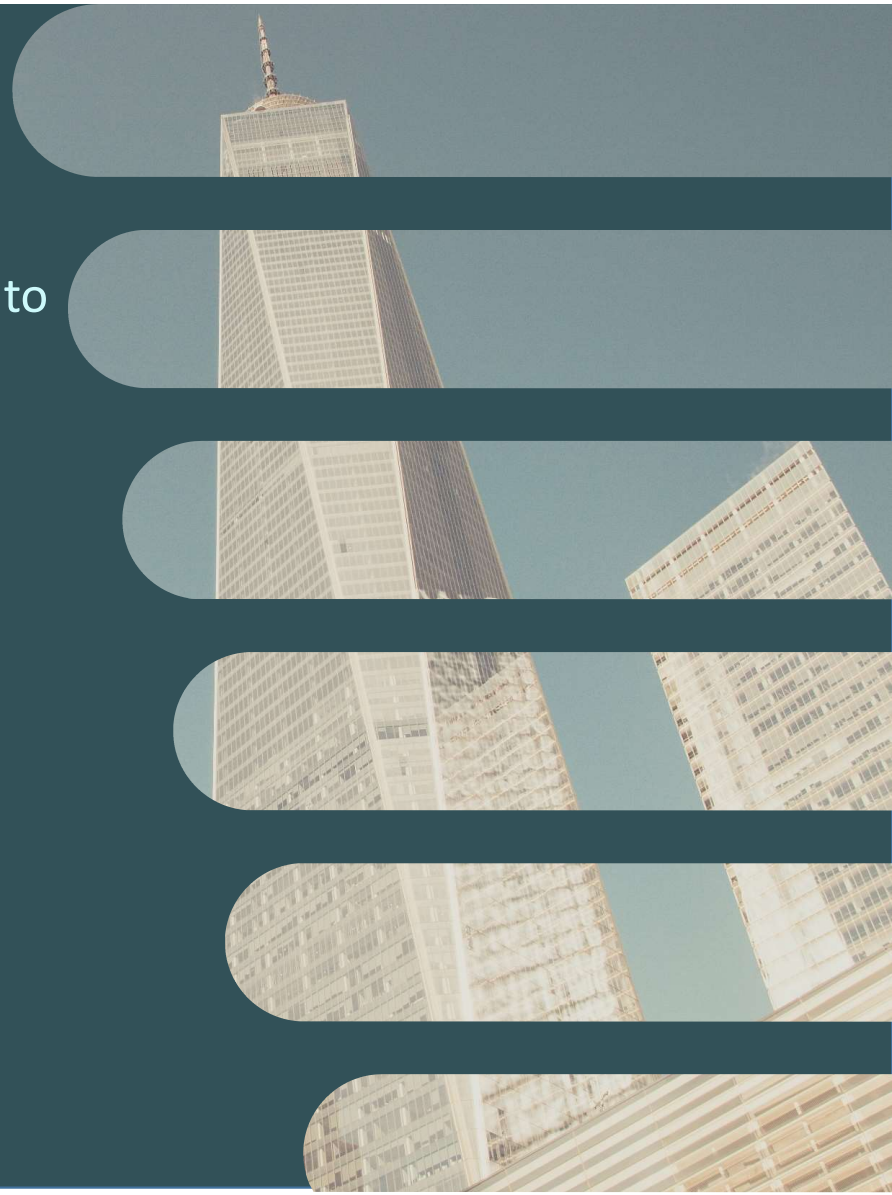
Puppet

- Configuration Management tool that is used for deploying, configuring and managing servers.
- Defining distinct configurations for each and every host, and continuously checking and confirming whether the required configuration is in place and is not altered on the host.
- Dynamic scaling-up and scaling-down of machines.
- Providing control over all your configured machines, so a centralized change gets propagated to all, automatically.

ZippyOPS
Making Automation Work
www.zippyops.com

Features

- It controls all the step, right from the bootstrapping to the end of server life.
- Can define configuration at the node level
- Can group them according to roles.
  - Example: Webserver , DB server
- Maintains consistency across nodes.



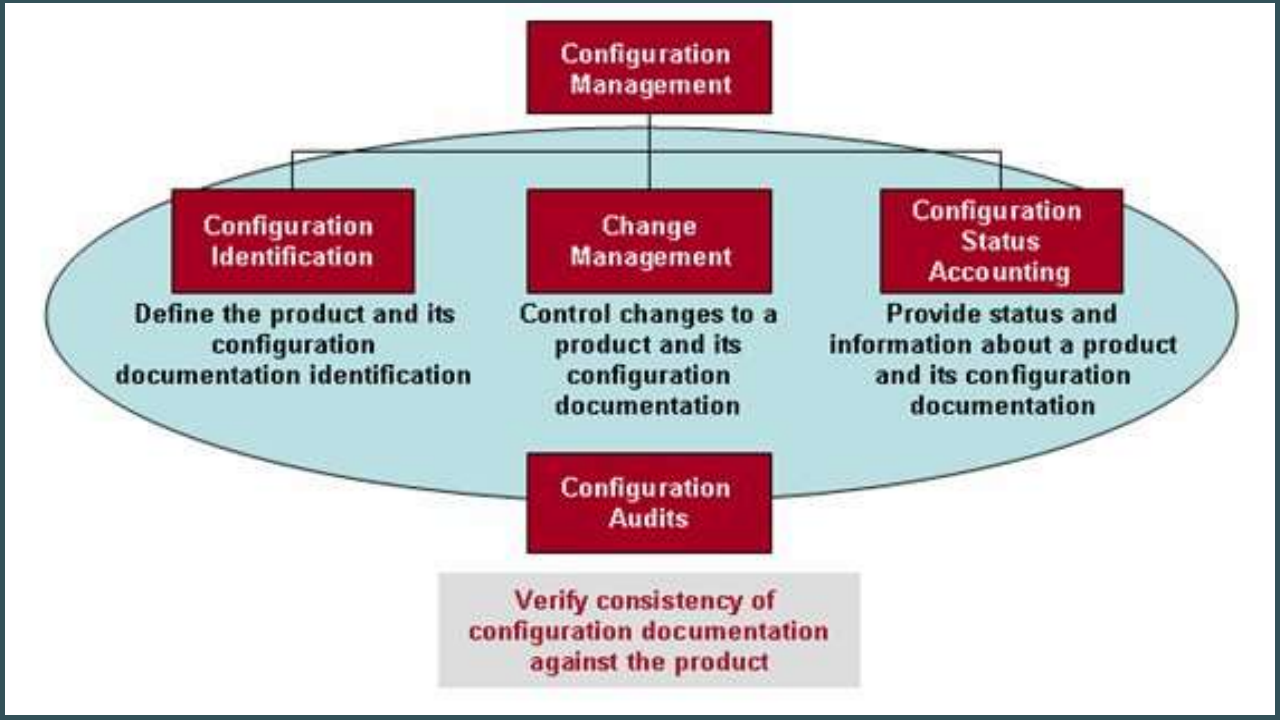
ZippyOPS
Making Automation Work
www.zippyops.com

What is Configuration Management?

- A Configuration Item is any service component, infrastructure element, or other item that needs to be managed in order to ensure the successful delivery of services.
- Examples of CI include individual requirements documents, software, models, and plans.
- Configuration Management consists of the following elements:
  - Configuration Identification
  - Change Management
  - Configuration Status Accounting
  - Configuration Audits

ZippyOPS
Making Automation Work
www.zippyops.com

Configuration Management



Configuration Identification:

- Labeling software and hardware configuration items with unique identifiers
- Identifying the documentation that describes a configuration item
- Grouping related configuration items into baselines
- Labeling revisions to configuration items and baselines.

ZippyOPS
Making Automation Work
www.zippyops.com

Change Management:

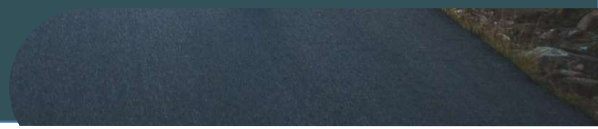
- It is a systematic approach to dealing with change both from the perspective of an organization and the individual.

ZippyOPS
Making Automation Work
www.zippyops.com

Configuration Status Accounting:

- It includes the process of recording and reporting configuration item descriptions
- All departures from the baseline during design and production.
- In the event of suspected problems, the verification of baseline configuration and approved modifications can be quickly determined.



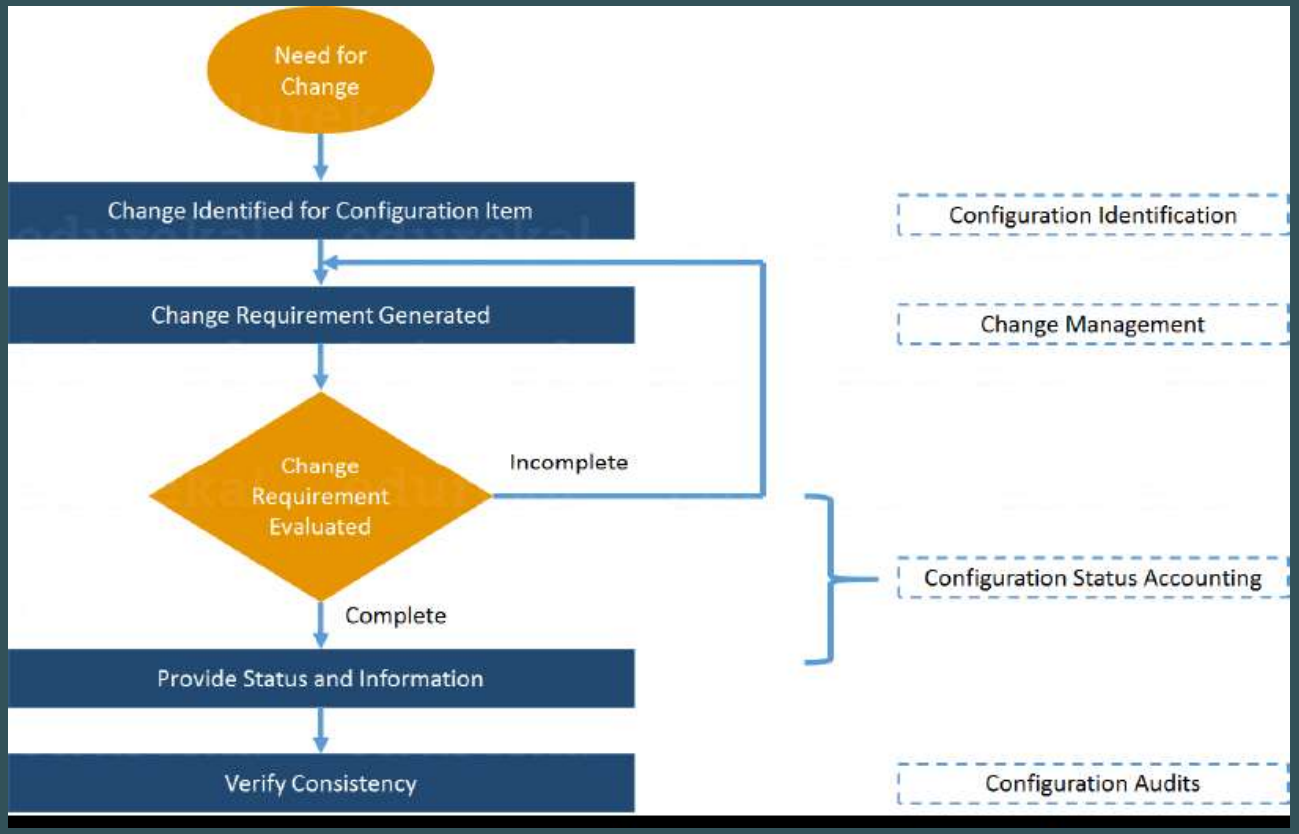
ZippyOPS
Making Automation Work
www.zippyops.com

Configuration Audits:

- Configuration audits provide a mechanism for determining the degree to which the current state of the system is consistent with the latest baseline and documentation.
- Basically, it is a formal review to verify that the product being delivered will work as advertised, promoted or in any way promised to the customers.
- It uses the information available as an outcome of the quality audits and testing along with the configuration status accounting information
- To provide assurance that what was required has been build.

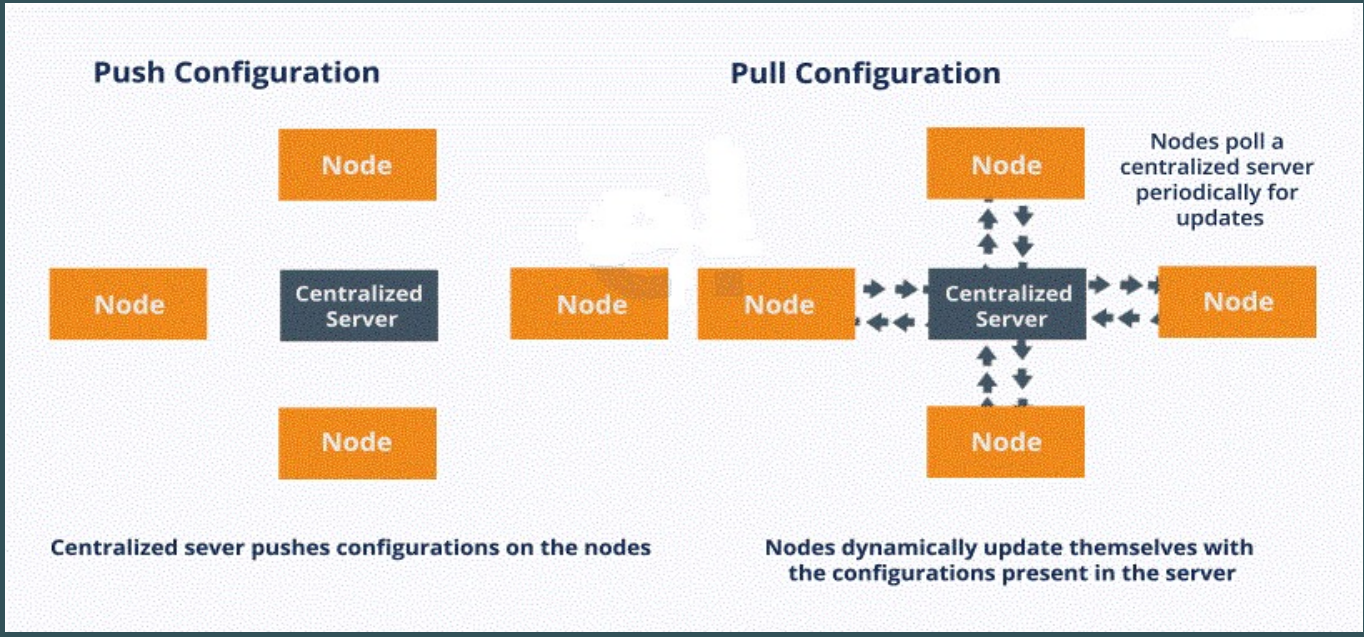
ZippyOPS
Making Automation Work
www.zippyops.com

Configuration management flowchart



ZippyOPS
Making Automation Work
www.zippyops.com

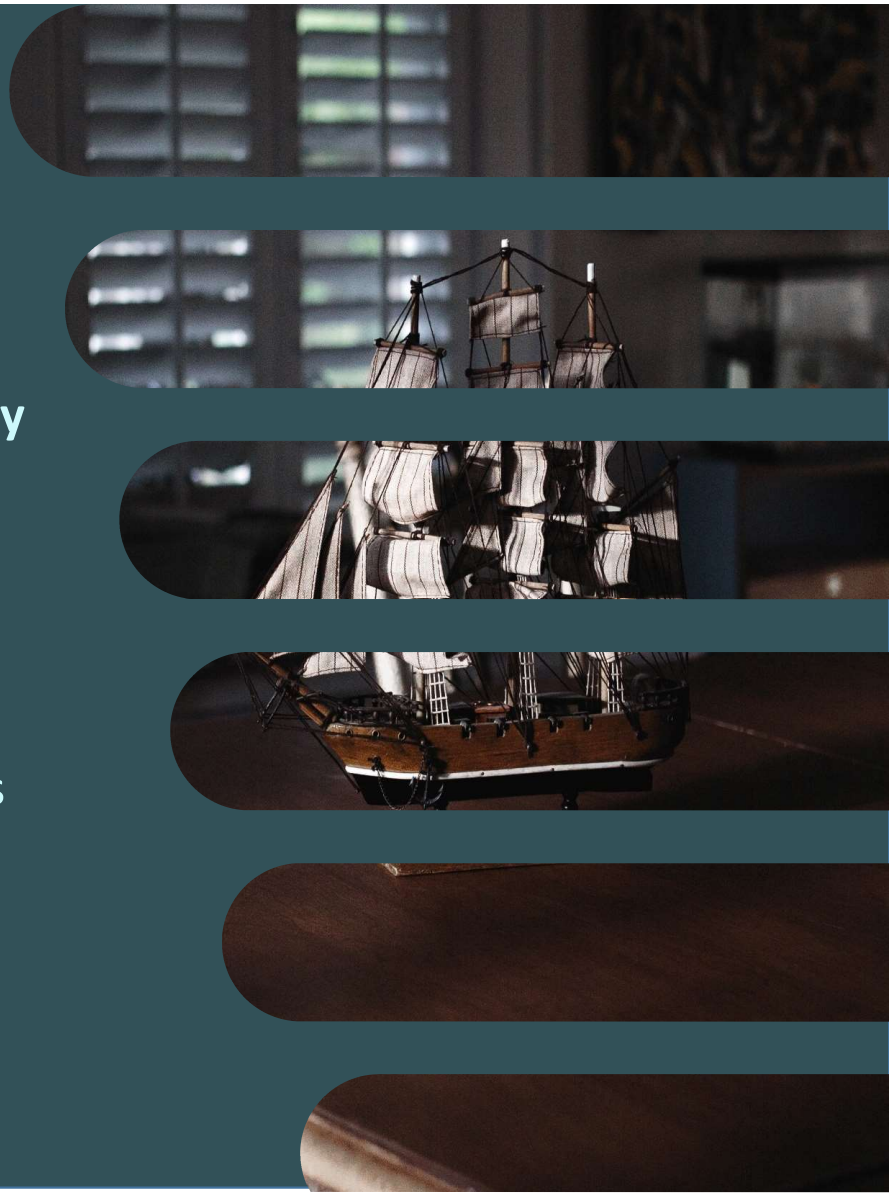
Configuration management components



ZippyOPS
Making Automation Work
www.zippyops.com

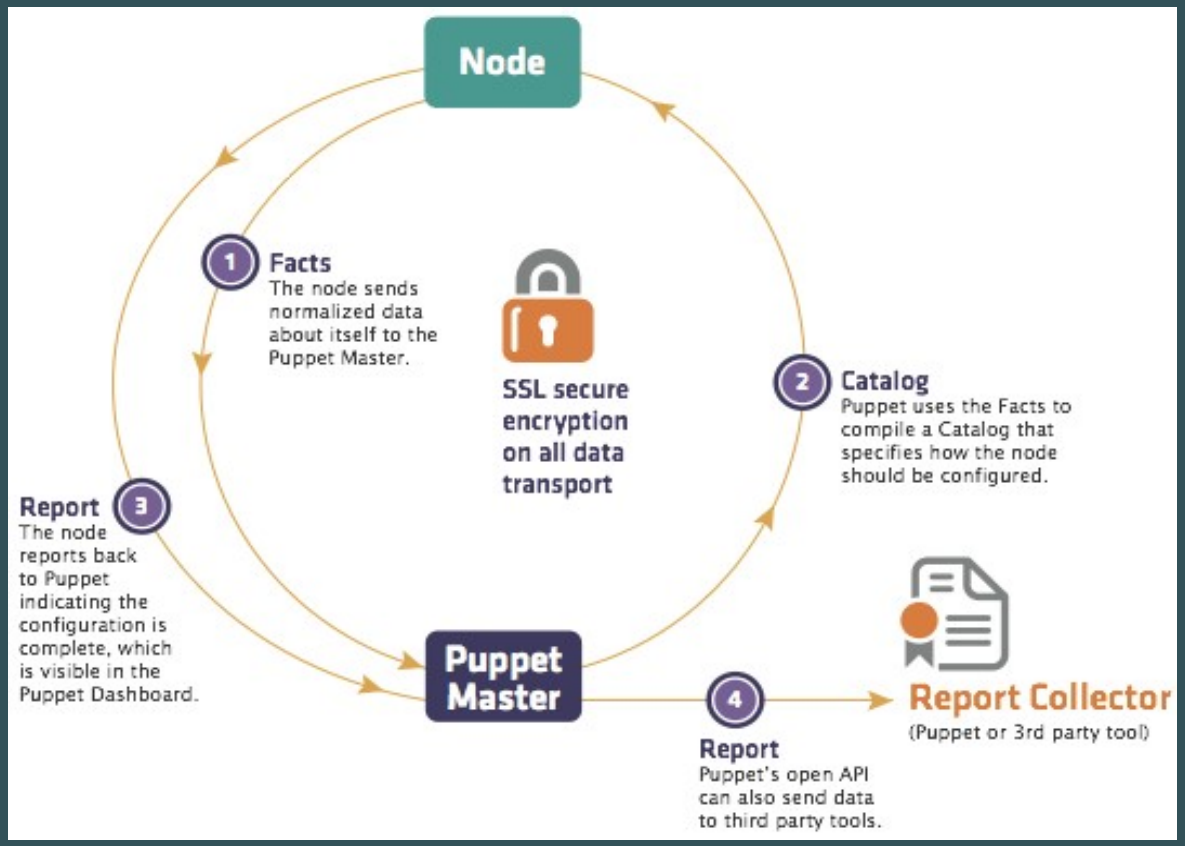
Configuration management Advantages :

- **Infrastructure as Code**: Track, Test, Deploy, Reproduce, Scale Code commits log shows the **history of change** on the infrastructure
- **Reproducible setups**: Do once, repeat forever
- **Scale** quickly: Done for one, use on many
- **Coherent** and consistent server setups
- **Aligned Environments** for devel, test, qa, prod nodes

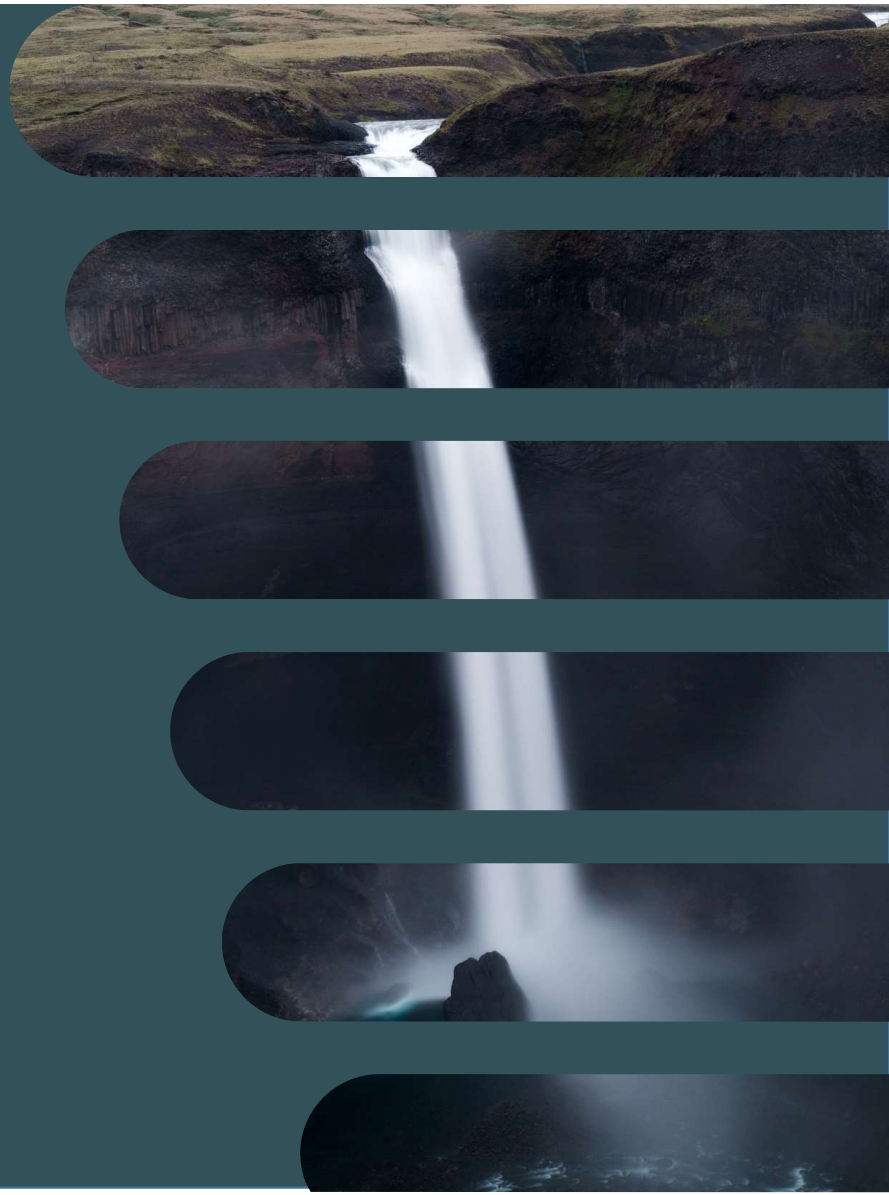


ZippyOPS
Making Automation Work
www.zippyops.com

Architecture of Puppet :



ZippyOPS
Making Automation Work
www.zippyops.com

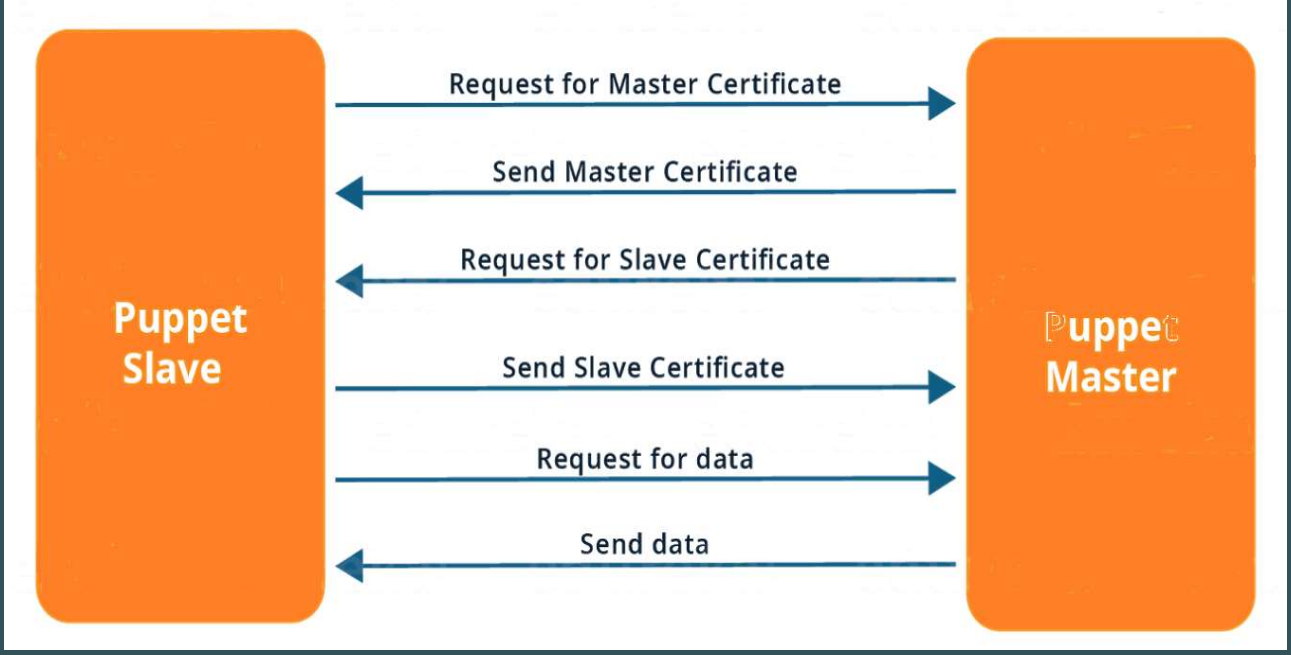


Puppet functions :

- The Puppet Agent sends the Facts to the Puppet Master. Facts are basically key/value data pair that represents some aspect of Slave state, such as its IP address, up-time, operating system, or whether it's a virtual machine.
- Puppet Master uses the facts to compile a Catalog that defines how the Slave should be configured. Catalog is a document that describes the desired state for each resource that Puppet Master manages on a Slave.
- Puppet Slave reports back to Master indicating that Configuration is complete, which is visible in the Puppet dashboard.

ZippyOPS
Making Automation Work
www.zippyops.com

Puppet Master and Slave Communication



ZippyOPS
Making Automation Work
www.zippyops.com

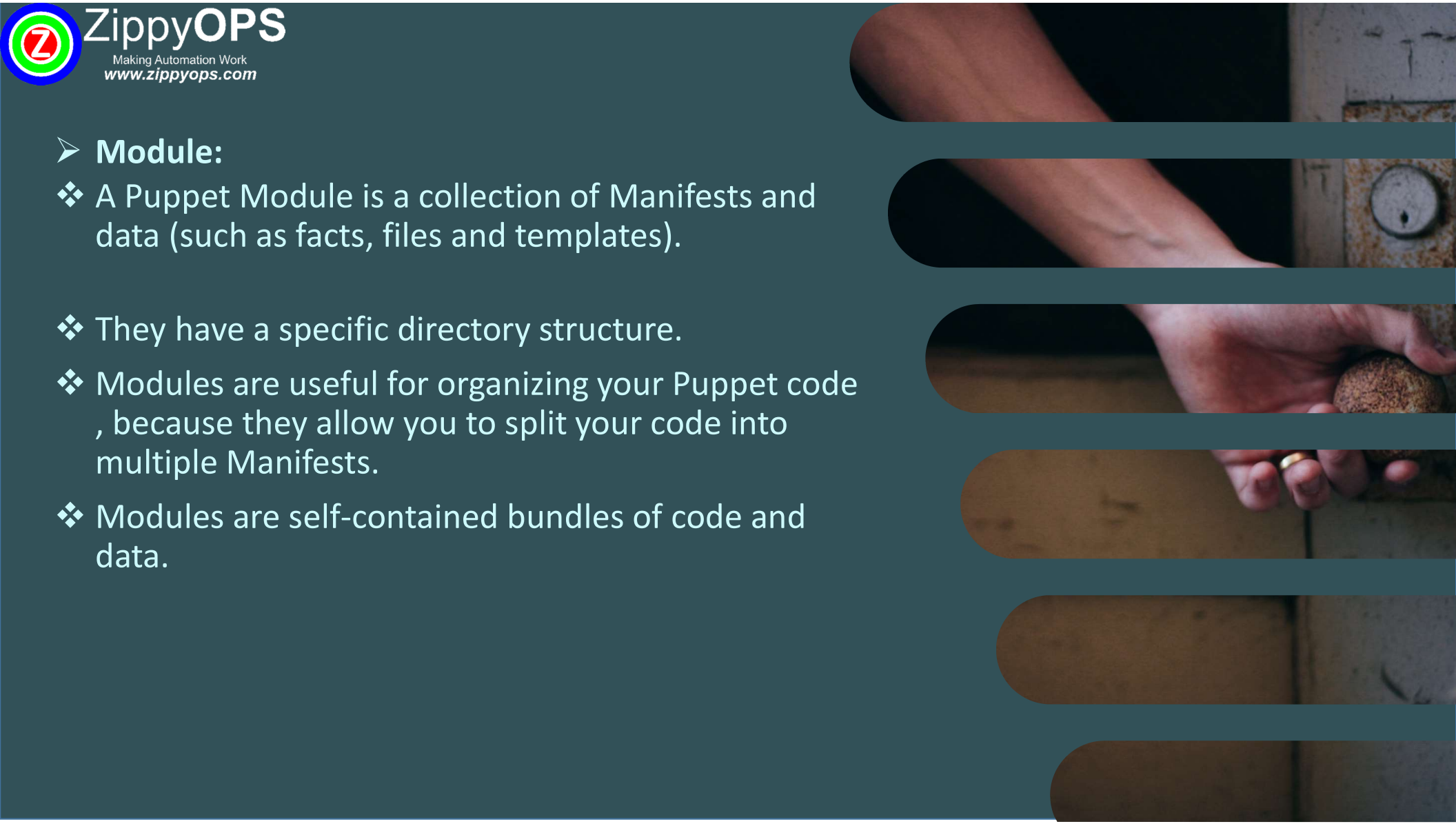
Puppet Master and Slave Communication

- Puppet Slave asks for Puppet Master certificate.
- After receiving Puppet Master certificate, Master requests for Slave certificate.
- Once Master has signed the Slave certificate, Slave requests for configuration/data.
- Finally, Puppet Master will send the configuration to Puppet Slave.

ZippyOPS
Making Automation Work
www.zippyops.com

Components of Puppet

- **Manifests:**
- Every Slave has got its configuration details in Puppet Master, written in the native Puppet language.
- These details are written in the language which Puppet can understand and are termed as Manifests.
- They are composed of Puppet code and their filenames use the *.pp* extension.



- **Module:**
- A Puppet Module is a collection of Manifests and data (such as facts, files and templates).

- They have a specific directory structure.
- Modules are useful for organizing your Puppet code , because they allow you to split your code into multiple Manifests.
- Modules are self-contained bundles of code and data.

- **Resource:**
  - Resources are the fundamental unit for modeling system configurations.
  - Each Resource describes some aspect of a system, like a specific service or package.

- **Facter :**
  - Facter gathers basic information (facts) about Puppet Slave such as hardware details, network settings, OS type and version, IP addresses, MAC addresses, SSH keys, and more.
  - These facts are then made available in Puppet Master's Manifests as variables.

ZippyOPS
Making Automation Work
www.zippyops.com

- **Mcollective**
- It is a framework that allows several jobs to be executed in parallel on multiple Slaves.
- Interact with clusters of Slaves, whether in small groups or very large deployments.
- Use a broadcast paradigm to distribute requests. All Slaves receive all requests at the same time, requests have filters attached, and only Slaves matching the filter will act on requests.
- Use simple command-line tools to call remote Slaves.
- Write custom reports about your infrastructure.

- **Catalogs:**
- A Catalog describes the desired state of each managed resource on a Slave.
- It is a compilation of all the resources that the Puppet Master applies to a given Slave, as well as the relationships between those resources.
- Catalogs are compiled by a Puppet Master from Manifests and Slave-provided data .
- As well as an optional external data (such as data from an external Slave classifier, exported resources, and functions)
- The Master then serves the compiled Catalog to the Slave when requested.

Installation Puppet Enterprise

- Puppet Master install command: ./puppet-enterprise-installer
- Installation Flags
  - -c – Use a pe.conf file to configure the Puppet server.
  - -D – Displays debugging information.
  - -h – Display help.
  - -q – Run in quite mode; the installation process is not displayed.
  - -V – Display very verbose debugging information.
  - -y – Assumes yes/default and bypass any prompts for user input.

ZippyOPS
Making Automation Work
www.zippyops.com

pe.conf

§ The pe.conf file is a HOCON-formatted file that declares parameters and values needed to install and configure Puppet Enterprise.

§ When the installation completes, you can find an updated pe.conf file in /etc/puppetlabs/enterprise/conf.d.

```
{
  "console_admin_password": "password",
  "puppet_enterprise::puppet_master_host":
  "<puppet-master-fqdn>",
  "pe_install::puppet_master_dnsaltnames": [
    "puppet"
  ]
}
```

Installation Directories

- Puppet Enterprise configuration files are installed in /etc/puppetlabs/puppet for

  *nix nodes and <COMMON_APPDATA>\\PuppetLabs for Windows nodes.

- Puppet Enterprise software binaries are installed in /opt/puppetlabs.
- Executable binaries are in /opt/puppetlabs/bin and /opt/puppetlabs/sbin.
- The installer automatically creates symlinks in /usr/local/bin.

Code and Data Directories

- R10k: /etc/puppetlabs/r10k
- Environments: /etc/puppetlabs/code/environments
- modules: Main directory for puppet modules (applies to master only)
- manifests: Contains the main starting point for catalog compilation (applies to master only)
- ssl: Contains each node's certificate infrastructure (all nodes) /etc/puppetlabs/puppet/ssl

Important Ports

- 3000: Used for the web-based installer of the Puppet master.
- 8140: The port on which the Puppet master and agents communicate.
- 61613: Used by MCollective for orchestration requests by Puppet agents.
- 443: The web port used to access the Puppet Enterprise Console.
- 5432: PostgreSQL runs on this port. It is used by PuppetDB in a split stack configuration.
- 8081: The PuppetDB traffic/request port.
- 8142: Used by Orchestration services to accept inbound traffic/responses from the Puppet agents.