**Q1. What is Puppet?**

Puppet is a Configuration Management tool which is used to automate administration tasks.

**Q2.How puppet Mater Agent communicates?**

Puppet has a Master-Slave architecture in which the Slave has to first send a Certificate signing request to Master and Master has to sign that Certificate in order to establish a secure connection between Puppet Master and Puppet Slave as shown on the diagram below. Puppet Slave sends request to Puppet Master and Puppet Master then pushes configuration on Slave.

**Q3. How Puppet Works?**

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. Catalogis 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.

**Q4. What are Puppet Manifests?**

Every node (or Puppet Agent) 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. Manifests are composed of Puppet code and their filenames use the .pp extension.

**Q5. What is Puppet Module and how it is different from Puppet Manifest?**

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

Puppet programs are called Manifests. Manifests are composed of Puppet code and their file names use the .pp extension.

**Q6. What is Facter in Puppet?**

Facter is basically a library that discovers and reports the per-Agent facts to the Puppet Master 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.

**Q7. What is Puppet Catalog?**

When configuring a node, Puppet Agent uses a document called a catalog, which it downloads from a Puppet Master. The catalog describes the desired state for each resource that should be managed, and may specify dependency information for resources that should be managed in a certain order.

**Q8. What size organizations should use Puppet?**

There is no minimum or maximum organization size that can benefit from Puppet, but there are sizes that are more likely to benefit. Organizations with only a handful of servers are unlikely to consider maintaining those servers to be a real problem, Organizations with many servers are more likely to find, difficult to manage those servers manually so using Puppet is more beneficial for those organizations.

**Q9. How should I upgrade Puppet and Facter?**

The best way to install and upgrade Puppet and Facter is via your operating system’s package management system, using either your vendor’s repository or one of Puppet Labs’ public repositories.

If you have installed Puppet from source, make sure you remove old versions entirely (including all application and library files) before upgrading. Configuration data (usually located in/etc/puppet or /var/lib/puppet, although the location can vary) can be left in place between installs.

**Q10. What is the Command to check requests of Certificates from Puppet Agent (Slave) to Puppet Master?**

To check the list of Certificate signing requests from Puppet Agent to Puppet Master execute **puppet cert list** command in Puppet Master**.**

**Q11. What is the use of etckeeper-commit-post and etckeeper-commit-pre on Puppet Agent?**

**etckeeper-commit-post**: In this configuration file you can define command and scripts which executes after pushing configuration on Agent.

**etckeeper-commit-pre:** In this configuration file you can define command and scripts which executes before pushing configuration on Agent.

**Q12. What characters are permitted in a class name? In a module name? In other identifiers?**

Class names can contain lowercase letters, numbers, and underscores, and should begin with a lowercase letter. “::” (Scope Resolution Operator) can be used as a namespace separator.

The same rules should be used when naming defined resource types, modules, and parameters, although modules and parameters cannot use the namespace separator.

Variable names can include alphanumeric characters and underscores, and are case-sensitive.

**Q13. Does Puppet runs on windows?**

Yes. As of Puppet 2.7.6 basic types and providers do run on Windows, and the test suite is being run on Windows to ensure future compatibility.

**Q14. Tell me about a time when you used collaboration and Puppet to help resolve a conflict within a team?**

The development team wanted root access on test machines managed by Puppet in order to make specific configuration changes. We responded by meeting with them weekly to agree on a process for developers to communicate configuration changes and to empower them to make many of the changes they needed. Through our joint efforts, we came up with a way for the developers to change specific configuration values themselves via data abstracted through Hiera. In fact, we even taught one of the developers how to write Puppet code in collaboration with us.

**Q15. Can I access environment variables with Facter in Puppet?**

Not directly. However, Facter reads in custom facts from a special subset of environment variables. Any environment variable with a prefix of FACTER\_ will be converted into a fact when Facter runs.

$ FACTER\_FOO=”bar”

$ export FACTER\_FOO</span>

$ facter | grep ‘foo’</span>

foo => bar

The value of the FACTER\_FOO environment variable would now be available in your Puppet manifests as $foo, and would have a value of ‘bar’. Using shell scripting to export an arbitrary subset of environment variables as facts is left as an exercise for the reader.

**Q16. What is the use of Virtual Resources in Puppet?**

Resources whose management depends on at least one of multiple conditions being met.

Overlapping sets of resources which might be needed by any number of classes.

Resources which should only be managed if multiple cross-class conditions are met.

**Q17. Command to check requests of Certificates?**

puppetca --list (2.6)  
puppetca --list (3.0)

**Q18. Command to sign Requested Certificates?**

puppetca –sign hostname-of-agent (2.6)

puppet ca sign hostname-of-agent (3.0)

**Q19.Where Puppet Master Stores Certificates?**

/var/lib/puppet/ssl/ca/signed

**Q20. What is Puppet Kick ?**

By default Puppet Agent request to Puppet Master after a periodic time which known as “runinterval”. Puppet Kick is a utility which allows you to trigger Puppet Agent from Puppet Master.

**Q21. What is MCollective ?**

MCollective is a powerful orchestration framework. Run actions on thousands of servers simultaneously, using existing plugins or writing your own.