CS312 Homework #4

February 22, 2016

Instructions

Please submit all answers as a single text file via T.E.A.C.H using the naming format `$onidusername-hw4.txt`. This homework is due at 4pm on Monday, Feb 29th.

Questions

1. Ansible primarily uses a push driven model, however it can also support a pull driven model.
   (a) True
   (b) False
2. Nagios plugins keep track of core monitoring logic including service states.
   (a) True
   (b) False
3. NRPE checks are initiated by Nagios
   (a) True
   (b) False

4. Horizontal scaling means adding more resources to a node

   (a) True

   (b) False

5. Vertical scaling is easier and more simple than horizontal scaling.

   (a) True

   (b) False

6. Hosts in Nagios must have at least once service defined

   (a) True

   (b) False

7. Which Nagios configuration component defines what hosts and services to monitor?

   (a) Main configuration file

   (b) Resource files

   (c) Objection definition files

   (d) CGI configuration file

8. Which consensus algorithm does CoreOS use for `etcd`?

   (a) Paxos

   (b) Raft

   (c) Ripple

   (d) RAID5

9. Which configuration management system is the oldest platform that is still in use today?

   (a) Chef

   (b) Puppet

   (c) Ansible

   (d) CFEngine

10. Describe two resources you might want to monitor on a server. Elaborate on how you might decide on acceptable thresholds for each resource.

11. Name two primary differences between active checks vs. passive checks for monitoring.

12. Describe what the following command does in detail:

```
/usr/lib64/nagios/plugins/check_nrpe -H foo.example.org \
  -c check_cpu
```

13. Which contact configuration directive in Nagios configures whether or not a contact will receive notifications about service problems and recoveries?

14. What are specificity and sensitivity? Give a 1–2 sentence description of each.

15. Give an example of a highly sensitive test that has low specificity.

16. What is time-series data? Give an explanation of why it is important.

17. Explain the CoreOS update process. What is it modeled after? Do you think this is a good or bad update model? Explain.

18. Name three scenarios that virtual IP's are good for increasing redundancy.

19. Name three ways that horizontal scaling can add complexity. Briefly explain why the complexity is necessary to scale horizontally.

20. Suppose you have a system running with a single load balancer, three web nodes, and two database nodes. What is the single point of failure? How could you get rid of this single point of failure? After fixing it, what other points of failure might be singular?