Scheduling the cluster server node shutdown based on the hierarchical and k-means clustering combinations

G A Sudarsono\*, E P Nugroho and R R J Putra

Department of Computer Science Education, Universitas Pendidikan Indonesia, Bandung, Indonesia

*Corresponding author’s email: ginanjarajisudarsono@gmail.com

Abstract. The problem of server clusters is the use of electrical power. The data center in Indonesia consumed 1.5% of the national generating capacity in 2014. To solve this problem is to turn off the server node on the server cluster. With the turn off server nodes on server clusters scheduling, it’s expected to determine the time and duration of turning off server nodes. When the server cluster is running, data retrieval takes one, five and 15 minutes average load which is done every minute. Then the collected data is clustered using a combination of average linkage hierarchical clustering and K-Means clustering. The results of this clustering produce three clusters load averages that are "low", "medium" and "high". Load averages that included into the "low" category are sorted by the time of data retrieval to get the time and duration to turn off at each node. The results of the research result in scheduling turning off node one is 14.09 - 14.28, at node two is 14.47 - 06.57 and at node three is 06.57 - 07.18. Turn off server node scheduling is reduces the use of electrical power from 2,528 kWh to 2,519 kWh and doesn’t affect server quality parameters.

1. Introduction
Server clusters distribute load workload (load balancing) to a number of n server computers that are on the server cluster through a load balancer. One of the Load Balancing algorithms is least-connection. This algorithm works by distributing requests from the client to the cluster based on the number of connections. This algorithm is used because it distributes traffic fairly [1].

The problem of server clusters is the use of electrical power. The data center, as a place to store server clusters, in Indonesia consumed 1.5% of the national generating capacity in 2014. This percentage increased in 2017 to 2% of the national generating capacity or equivalent to the consumption of generating capacity for the combined Jambi, Riau and West Sumatra [2]. One way to solve the problem of using electricity is to turn off the server node on the server cluster [3]. Server node shutdown gives effect to the use of electric current. If all nodes on the server cluster are turned on without being loaded, the electrical current used is 0.9A. If one of the nodes is turned off, the electric current used is reduced to 0.7A [4].

Server node shutdown should not be done without planning because it will affect the server quality. Therefore, it is necessary to schedule effective server node shutdown by considering the server node performance. One of the server node performance indicators is load average. Load average indicates the workload that is being processed by the CPU [5].
The combination of Hierarchical clustering and K-Means clustering algorithms can result in better grouping of data when compared to K-Means clustering. The average linkage hierarchical clustering method is chosen because it produces a cophenetic correlation coefficient [6].

In this study server cluster will use the least connection load balancing method. When the server cluster is running the load average data retrieval is needed as a scheduling parameter. The parameters needed are load one, five and 15 minutes. Data retrieval is done every minute. Then the collected data is clustered using a combination of average linkage hierarchical clustering and K-Means algorithms. The results of the clustering resulted in three cluster load averages that were "low", "medium" and "high". Load averages that fall into the "low" category are sorted by the time of data retrieval so as to get the time and duration of shutdown at each node.

By shutting down the server node affects server quality and electrical power usage. Therefore, it is necessary to test the quality of the server and the use of electrical power before and after extinguishing based on the results of processing uptime data. Server quality testing uses parameters of availability, throughput and packet loss. Availability refers to systems that can provide services under normal operating conditions during the use of applications for data exchange. Availability shows the percentage of the system's normal operation duration [7]. Throughput is the rate at which data is successful in a time interval measured in bit / second. Packet loss is a condition where a package has been sent but never reaches its destination [8].

With the scheduling of shutting down server nodes on server clusters, it is expected to determine the time and duration of the shutdown so that it can efficiently use electric power and maintain maximum server quality.

2. Method

In this study, the system is four servers that are connected in a LAN where one server functions as a load balancer that functions to receive requests and forward them to three other server nodes in the cluster for processing. The network infrastructure used is adapted from a star topology. The network topology can be seen in Figure 1.

![Cluster server topology](image)

**Figure 1.** Cluster server topology

The software that is created is a software that provides the function of suppressing server nodes in order to save electricity usage. The development of this software requires several dependencies in the form of hardware, software and libraries. Server shutdown can be realized by applying a combination of average linkage search clustering and K-means clustering methods on uptime logs from each server cluster node. Uptime data consists of duration of uptime, number of users, load average at one minute, five minutes and the last 15 minutes. From the results of clustering, the highest and lowest load can be determined along with the time and duration of the outage of each node. Each node will be shutdown according to the time and duration of the outage the next day.

The input needed in this software is in the form of uptime data from each server cluster member and date range. The uptime data is captured in realtime from the server used. The data is in the form of a string that must be processed first before entering it into the software. The uptime consists of "<current
The data is then dispart, so that only the average load in one, five and 15 minutes and current time are taken. Users enter the date range that will be processed by the software so that it can provide output.

The output of this software is in the form of a shutdown clock and the longest duration of each server node that is processed using a combination of average linkage hierarchical clustering and K-Means clustering as shown in Figure 2. Data from clustering results is used to schedule server node shutdown.

![Figure 2. Combined average linkage hierarchical clustering and k-means clustering.](image)

Before shutdown is carried out, the power used is measured to be used as a comparison material with measurements after shutdown. Scenario testing before and after the shutdown as follows:

- Server clusters are given a concurrent load of 180 at 00.00-07.59
- Server clusters are given a concurrent load of 360 at 08.00-15.59
- Server clusters are given a concurrent load of 540 at 16.00-23.59

3. Research and Discussion
Getlog is a minute uptime data retrieval and processing uptime data to be the input format needed by the software. Figure 3 describes the getlog process.

![Figure 3. Getlog process](image)

Data uptime is processed to be input is required (preprocessing), the time, "load1", "load5" and "load15" added the date. The output of this process is the log data that is required by the software. The result of the process get log on a node of one to three are in table 1 to table 3.
Figure 4 is a power efficiency process that functions to get a shutdown schedule, the first thing to do is clustering with a combination of average linkage hierarchical and K-Means clustering. The results of the clustering resulted in three clusters. To get the shutdown schedule, objects included in the low cluster are sorted by date and time. The selected shutdown time is the longest span of time with the condition of the initial object and the next object at one minute's difference. "N" is the number of getlog data. "Temp_i" is the place to save the end time of shutdown. "Temp_counter" is where you can save the maximum shutdown duration. "Counter" is the duration of shutdown. "I" is an index. "Row" is the amount of data that is in the low cluster that has been sorted. "T" is data that is in a low cluster that has been sorted. "Delta t" is the time difference in data.

| Table 1. Node one getlog results |
|---------------------------------|
| date   | time | load1 | load5 | load15 |
| 02/05/2018 | 00:00 | 0.10  | 0.05  | 0.07   |
| 02/05/2018 | 00:01 | 0.03  | 0.04  | 0.07   |
| 02/05/2018 | 00:02 | 0.01  | 0.03  | 0.06   |
| ...    | ...  | ...   | ...   | ...    |
| 02/05/2018 | 23:57 | 0.03  | 0.08  | 0.10   |
| 02/05/2018 | 23:58 | 0.08  | 0.08  | 0.10   |
| 02/05/2018 | 23:59 | 0.06  | 0.08  | 0.09   |

| Table 2. Node two getlog results |
|---------------------------------|
| date   | time | load1 | load5 | load15 |
| 02/05/2018 | 00:00 | 0.00  | 0.00  | 0.07   |
| 02/05/2018 | 00:01 | 0.00  | 0.00  | 0.07   |
| 02/05/2018 | 00:02 | 0.00  | 0.00  | 0.06   |
| ...    | ...  | ...   | ...   | ...    |
| 02/05/2018 | 23:57 | 0.07  | 0.07  | 0.07   |
| 02/05/2018 | 23:58 | 0.14  | 0.08  | 0.08   |
| 02/05/2018 | 23:59 | 0.05  | 0.07  | 0.07   |

| Table 3. Node three getlog results |
|---------------------------------|
| date   | time | load1 | load5 | load15 |
| 02/05/2018 | 00:00 | 0.04  | 0.09  | 0.12   |
| 02/05/2018 | 00:01 | 0.09  | 0.10  | 0.12   |
| 02/05/2018 | 00:02 | 0.09  | 0.10  | 0.12   |
| ...    | ...  | ...   | ...   | ...    |
| 02/05/2018 | 23:57 | 0.18  | 0.21  | 0.19   |
| 02/05/2018 | 23:58 | 0.17  | 0.19  | 0.18   |
| 02/05/2018 | 23:59 | 0.35  | 0.23  | 0.20   |
In the powefficiency process, the data obtained from the getlog process is processed in the range from 2 May 2018 to 2 May 2018. The data number is 1440. The results of this process are the time and duration of shutdown. In Table 5 is the result of the powefficiency with the time span of 2 May 2018.

### Table 4. Time and duration of shutdown

| Node | Start Time | Finish Time | Duration |
|------|------------|-------------|----------|
| 1    | 14.09      | 14.28       | 19 minutes |
| 2    | 14.31      | 14.47       | 16 minutes |
| 3    | 06.57      | 07.18       | 21 minutes |

At node one, the shutdown will be carried out at 14.09 with a duration of 19 minutes and will be turned on at 14.28. At node two, the shutdown will be carried out at 14.31 with a duration of 16 minutes and will be turned on at 14.47. At node three, the shutdown will be done at 06.57 with a duration of 21 minutes and will be turned on at 07.18.

In the shutdown process, the start and end time of shutdown obtained from the powefficiency process will be entered automatically so that the shutdown and turn on schedule will occur automatically. The result of this process is that the server node succeeds in turning off and on at a time that matches the software output.
Figure 5 shows the difference in power usage before and after shutdown. During the time period 00.00-07.59, before the shutdown, the use of power was 0.84 kWh while after the shutdown was done according to the schedule issued by the software at 06.57 with a duration of 21 minutes then turned on again at 07.18 at node three there was a decrease in power usage to 0.837 kWh, or a decrease of 0.003 kWh. During the period 08.00-15.59, before shutdown, the use of power is 0.84 kWh while after two shutdowns according to the schedule issued by the software. At node one, the shutdown occurred at 14.09 with a duration of 19 minutes so that it was turned on again at 14.28 and the second node was turned off at 14.31 with a duration of 16 minutes so that it was turned on at 14.47, there was a decrease in power usage to 0.834 kWh, or a decrease of 0.006 kWh. In the span of 16.00-23.59, there was no shutdown because there was no scheduling of shutdown in that range. At this time span the use of power is 0.848 kWh. Total power usage before scheduling was 2,528 kWh and after scheduling it was 2,519 kWh.

Figure 6 shows the difference in throughput before and after shutdown. During the period 00.00-07.59, before shutdown, the throughput value was 11.01 Mbps while after the shutdown was carried out according to the schedule issued by the software at 06.57 to 07.18 with a duration of 21 minutes at node three, the throughput value remained 11.01 Mbps. During the time period 08.00-15.59, before shutdown the throughput value was 11.01 Mbps while after the shutdown was done according to the schedule issued by the software at 14.09 to 14.28 with a duration of 19 minutes at node one and at 14.31 to 14.47 with a duration of 16 minutes at node two, fixed throughput 11.01 Mbps. During the 16.00-23.59 timeframe, there was no shutdown because there was no scheduling of termination in that time span. At this time, the throughput value is 11.02 Mbps.
Figure 7 shows the availability differences before and after shutdown. During the time period 00.00-07.59, before the shutdown, availability value was 100% while after the shutdown was carried out according to the schedule issued by the software at 06.57 to 07.18 with a duration of 21 minutes on the node three fixed availability values were 100%. During the period 08.00-15.59, before shutdown, availability value was 100% while after the shutdown was done according to the schedule issued by the software at 14.09 to 14.28 with a duration of 19 minutes at node one and at 14.31 to 14.47 with a duration of 16 minutes on node two the availability value remains 100%. In the span of 16.00-23.59, there was no shutdown because there was no scheduling of shutdown in that range. During this time the availability value is 100%.

Figure 8 shows the packet loss differences before and after shutdown. During the time period 00.00-07.59, before the shutdown, packet loss value was 0% while after the shutdown was carried out according to the schedule issued by the software at 06.57 to 07.18 with a duration of 21 minutes on the node three fixed availability values were 0%. During the period 08.00-15.59, before shutdown, packet loss value was 0% while after the shutdown was done according to the schedule issued by the software at 14.09 to 14.28 with a duration of 19 minutes at node one and at 14.31 to 14.47 with a duration of 16 minutes on node two the packet loss value remains 0%. In the span of 16.00-23.59, there was no shutdown because there was no scheduling of shutdown in that range. During this time the availability value is 0.02%.
4. Conclusion
The conclusion of the study are as follows (1) The system for managing the scheduling of server node shutdown on the server cluster is successfully built; (2) Scheduling successfully gives the server node time and duration of shutdown on the server cluster; (3) The use of scheduling the shutdown of server nodes on server clusters is successful in reducing electrical power and maintaining server quality; (4) The suggestions given for the development of further research are as follows; (5) Need to do further research on server load balancing with different load balancing algorithms taking into account hardware specifications; (6) Need to do research using other algorithms to group server nodes based on their level of load average.

5. References
[1] M. E. Mustafa 2017 Load Balancing Algorithms Round-Robin ( Rr ), Least- Connection , and Least Loaded Efficiency Comput. Sci. Telecommun. 51 1 25–29
[2] Asian Development Bank 2017 The Emerging Indonesian Data Center Market and Energy Efficiency
[3] A C Orgerie, M D de Assuncao and L Lefevre 2014 A Survey on Techniques for Improving the Energy Efficiency of Large-scale Distributed Systems ACM Comput. Surv. 46 41-47
[4] Y S Harja 2016 Kontrol Ketersediaan Node Server pada Server Cluster dengan Kluserisasi Data Uptime Menggunakan Algoritma K-Means (Bandung: Universitas Pendidikan Indonesia)
[5] R Walker 2006 Examining Load Average Linux J. 2006 152 5
[6] T Alfina, B Santosa and R Barakbah 2012 Analisa Perbandingan Metode Hierarchical Clustering , K-means dan Gabungan Keduanya dalam Cluster Data ( Studi kasus : Problem Kerja Praktek Jurusan Teknik Industri ITS ) J. Tek. ITS 1 521–525
[7] H Lyu, P Li, R Yan, H Qian and B Sheng 2017 High-availability Deployment for Large Enterprises Proc. 2016 IEEE Int. Conf. Prog. Informatics Comput. 2 503–507
[8] J F Kurose and K W Ross 2008 Computer networking : A Top-down Approach, 6th ed New Jersey: Pearson