

# A Service Scheduling Security Model for a Cloud Environment

Abdullah Sheikh<sup>1</sup>, Malcolm Munro<sup>2</sup>, David Budgen<sup>3</sup>  
Department of Computer Science  
Durham University  
Durham, United Kingdom

**Abstract**—Scheduling tasks on a standalone system can be complex but applying it to a cloud environment can be even more complex because of the large amount of resources available. An added complexity in a Cloud environment is that of security. This paper addresses scheduling from a security point of view and presents a Scheduling Security Model and evaluates its effectiveness to meet user’s requirements with a number of worked examples with different scenarios.

**Keywords**—Scheduling; security; model; cost; cloud computing

## I. INTRODUCTION

Scheduling tasks on a standalone system to meet the needs of users can be complex but applying it to a cloud environment can be even more complex. This is because of the large amount of different resources available in a Cloud environment. An added complexity in a Cloud environment is that of security.

The National Institute of Standard and Technology (NIST) [1] gives a basic definition of Cloud Computing as a model for a customer to request an on-demand convenient services, that allow to use and access network resources such as servers, data storage, software and applications with indirect provider interactions. There are a various criteria to classify Scheduling Models in Cloud Computing to produce reliable, trusted and secure services, such as scheduling performance [2], cost scheduling [3], and scheduling based on security [4].

Also, NIST [1] specifies the deployment models to four deployment models, which are are: Public, Private, Community, Hybrid. These deployment models identify the accessibility to the cloud service [5]. Dillon et al. [6] indicate some security concerns about the deployment models that include data privacy and trust, service policies, and data transfer. So, these security concerns make the service provider aware to address these issues which have been focused by [7]. Also, [8] describe the main issues that can affect the cloud service models. Then security policies need to be applied to deal with data access and security.

As an example consider a service that consists of two tasks. First task is analysing data with basic security requirement. The second task is a high security task about saving private data. Thus this service requires two resources, one for each task, with different levels of security. In general, the higher the security the higher the cost of using a resource.

This requires that security levels need to be defined. In [9] the security levels for the SSM are driven from Watson [10]

as five security levels summarised on Fig. 1:

- 1) The first level 1: It is all about applying essential security level that can run tasks on trusted public resources.
- 2) The second level 2: It is all about applying more security setting that can run tasks on more trusted public resources.
- 3) The third level 3: It is all about applying highly security setting to run tasks on highly trusted public resources.
- 4) The fourth level 4: It is all about applying advance security setting to run tasks on trusted private resources.
- 5) The fifth level 5: It is all about applying more highly security setting to run tasks on highly trusted private resource.

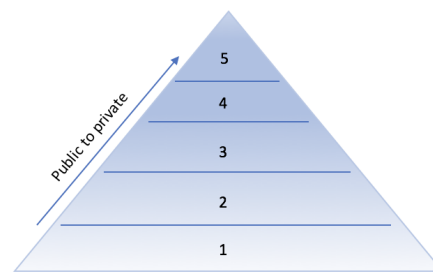


Fig. 1. The Five Security Levels Applied for Resources [9].

Thus for the previous example two resources are required. One of the tasks will be on a resource with Level 1 security and the other on a resource with Level 5 security.

A Scheduling Security Model (SSM) [11] [12] [13] is been devised to make security the primary factor for scheduling algorithm. Imbedded in the SSM is a cost calculation together with a scheduling algorithm. To resolve some scheduling issues within a resource each task is given an importance level. Other factors included in the model are times for each task, resource cost per hour, and a Quality of Service cost.

In the SSM a service consists of M tasks that can be executed on N resources. Each task is given a Security Level value of 1 to 5. Tasks with the same Security Level will be run on the same resource, thus N ranges from 1 to 5.

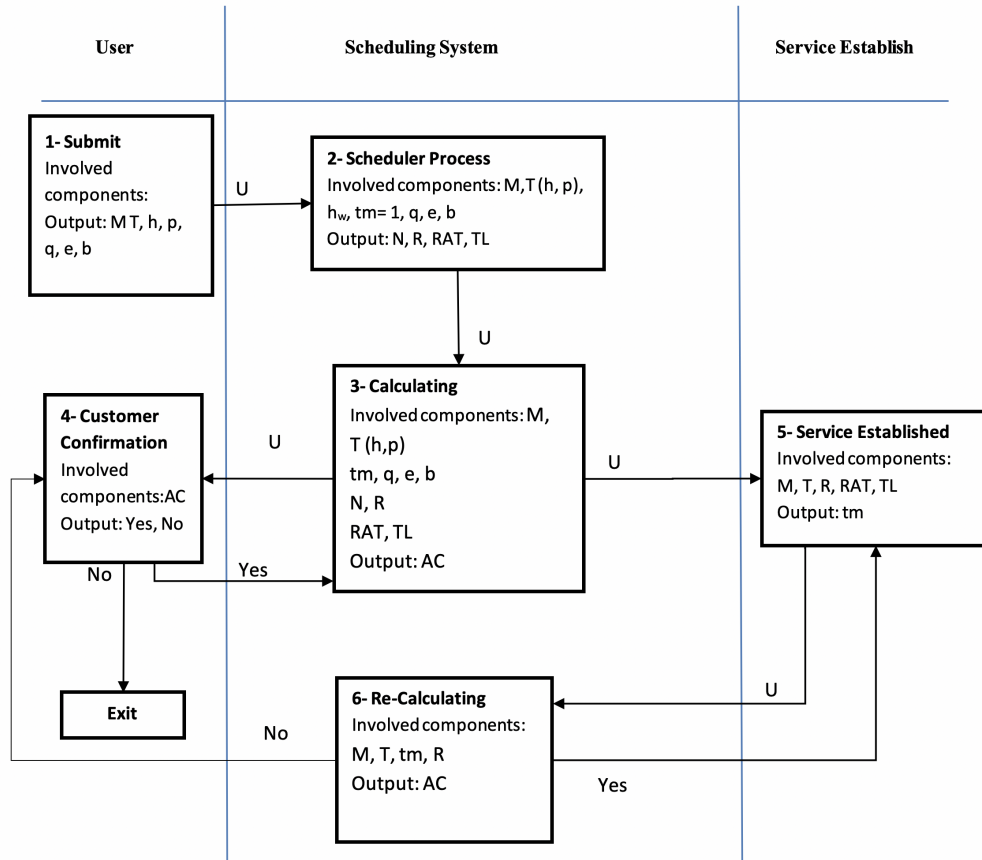


Fig. 2. Scheduling Security Model Function

The contribution of this paper is to address scheduling from a security point of view and present a Scheduling Security Model and evaluate its effectiveness to meet user's requirements with a number of worked examples with different scenarios.

## II. MODEL DEFINITION

This section serves as a definition of the SSM and its components. Then it summarises the SSM function.

The SSM [13] is defined formally as follows.

First, the SSM categorises all tasks based on the tasks security level. Then the SSM calculates initial cost for service. Then, the SSM requests confirmation from the customer to start the service. Next, establishing the service after the receiving the confirmations. Then the SSM recalculates the actual cost.

The suffixes used in the definitions are specified as follows:

$i$  : Tasks

$k$  : Resources

$j$  : Security Level

$l$  : Task Importance

The values in the definitions are:

- $tm$ : Time cost  
 $tm_i$ : time cost for task  $i$
- $q$ : Quality of Service for the service.
- $b$ : Customer budget for the service.
- $h$ : Security level for a task.  
 $h_j$ : Security level  $\in \{1, 2, 3, 4, 5\}$
- $hw$ : Security weight for the security level (for each task).  
 $hw_j$ : Security weight for security level  $j$ .
- Each task ( $t_i$ ) has a security level  $h_j$   
(and then a security weight  $hw_j$ )
- $R$ : Set of resources ( $R_k$ ,  $k \in \{1, 2, 3, 4, 5\}$ )  
(can use up to to 5 resources  $\{R_1, R_2, R_3, R_4, R_5\}$ )  
then, used resources will be numbered from 1 to  $N$   
 $N$ : is the number of resources used ( $N$  determined later)
- $R_k$ : is a set of tasks, where the  $hw_j = R w_k$
- $Rw$ : Resource security weight (for each resource)  
 $Rw_k$ : Security weight for resource  $k$  ( $R_k$ )

$$Rw_k = hw_j$$

- $p$ : Tasks Importance,  $p_l$ : is the importance  $\{1, 2, 3\}$  task  $t_i$  has importance  $p_l$
- $M$ : Number of tasks.
- $N$ : Number of resources.
- $e$ : Maximum time.

The main components of the SSM:

- All tasks including the total number of tasks  $M$ , and Tasks Set  $T$ .
- The time duration for executing the task time cost : The elapsed time (EPT)  $tm_i$ .
- The required Quality of Service (QoS) level of the service  $q$ .
- The overall requested service budget  $b$ .
- The Security Level for each task  $h_j$ . The SSM will identify the security weight  $hw_j$  for each task  $t_i$ . Then The SSM allocates the tasks to the correspondents resources  $R$  with similar resource weight  $Rw_k$ . The SSM allocates and executes all tasks on the same Resource with same security weight.
- Task Importance Level  $p_l$  for each task, and the SSM defines these levels into three levels.

The reason for making the Task Importance in three levels is that the scheduling in the SSM is serving the security as a category, then there is a need to give each task within the same category an order to be executed. So, the order will be as identified with this three levels but if there are some tasks with the same Task Importance level then the scheduling will be for first come first served.

Table I shows a simple example of ordering tasks in same resource depends on task importance.

TABLE I. SIMPLE EXAMPLE OF ORDERING TASKS BASED ON TASK IMPORTANCE

Security Level(Weight)/Importance	1	2	3	$R_k$
1 (0.00)				
2 (0.25)				
3 (0.50)	$t_3$	$t_1$	$t_2$	$R_1$
4 (0.75)				
5 (1.00)				

Another simple example with tasks dependencies will be discussed in details later in the Scheduling Process. Tasks submitted  $t_1: h_1 = 2, p_1 = 1, t_2: h_2 = 2, p_2 = 3$ , and  $t_3: h_3 = 4, p_3 = 2$ . Then the Task dependency  $t_3$  depends on  $t_1$ . The SSM will analyse this input and allocates Tasks to two Resources  $R_1$  and  $R_2$ . Tasks  $t_1$  and  $t_2$  will be allocated to  $R_1$  and  $t_3$  to  $R_2$ . But with the dependencies required  $t_1$  will assigned to the Fast-Track list. That means  $t_1$  will be executed first then  $t_3$  depends on  $t_1$  and it has the highest Security Level that lets the SSM to list first see Table II.

TABLE II. SIMPLE EXAMPLE OF ORDERING TASKS WITH DEPENDENCIES AND FAST-TRACK

Security Level(Weight)/Importance	1	2	3	$R_k$
1 (0.00)				
2 (0.25)	$t_1^{FT}$	$t_2$	$R_1$	
3 (0.50)				
4 (0.75)		$t_3$		$R_2$
5 (1.00)				

Table III shows a summary of the components that specify the customer requirement for requesting a service. Then SSM will analyse the requirements for the calculating step. Example of a service required is shown in Table IV.

- $AC$  Actual Cost of the service.
  - Resource Cost  $RC$  = Cost of resources for  $RT_k$  hours.

$$Resource\ Cost(RC) = \sum_{k=1}^N (RC_k * RT_k) \quad (1)$$

Where  $RC_k$  is Resource Cost for Resource  $k$  per hour and  $RT_k$  is the actual time used by Resource  $k$  in hours.

- QoS Cost =  $RC * Quality\ of\ Service\ (q)$  required

$$QoS\ Cost = RC * q \quad (2)$$

- Security Cost for each Resource

$$SC_k = RC_k * Rw_k * RT_k \quad (3)$$

Where  $Rw_k$  is Security weight for Resource  $k$ .

- Security Cost for all Resources

$$SC = \sum_{k=1}^N SC_k = \sum_{k=1}^N (RC_k * Rw_k * RT_k) \quad (4)$$

- Actual Cost ( $AC$ ) =  $RC + SC + QoS\ Cost$

Therefore

$$AC = \sum_{k=1}^N (RC_k * RT_k) + q * \sum_{k=1}^N (RC_k * RT_k) + \sum_{k=1}^N (RC_k * Rw_k * RT_k) \quad (5)$$

Then

$$AC = \sum_{k=1}^N ((RC_k * RT_k) * (1 + q + Rw_k)) \quad (6)$$

Fig. 2 shows the SSM function and summarised as the following:

TABLE III. SUMMARY OF THE CUSTOMER INPUTS

Component	Values	Range
Budget	$b$	$b > 0$
Maximum Time	$e$	$0 < e \leq 60$
QoS	$q$	0.0, 0.1, 0.2, ..., 1.0
Tasks	$t_1, t_2, t_3, \dots, t_i$ , indexed by $i, i \in \{1 - M\}$	M Tasks
Task Security Level	$h_1, h_2, h_3, \dots, h_j$ , indexed by $j$	$j \in \{1, 2, 3, 4, 5\}$
Task Importance	$p_1, p_2, p_3, \dots, p_l$ , indexed by $l$	$l \in \{1, 2, 3\}$

TABLE IV. EXAMPLE OF A SERVICE REQUIRED

Security Level(Weight)/Importance	1	2	3	$R_k$	$RC_k$
1 (0.00)					
2 (0.25)		$t_1$		$R_1$	20
3 (0.50)	$t_2$			$R_2$	20
4 (0.75)					
5 (1.00)					

1) **Submit:**

- Budget  $b$ .
- Maximum Time Required  $e$ : Minimum 60 minutes (one hour).
- The QoS  $q$ .
- The Number of Tasks  $M$ .
- $t_i: h_j \in \{1, 2, 3, 4, 5\}, p_l \in \{1, 2, 3\}$ .

2) **Scheduler Process:**

The SSM creates categories for all tasks. The number of categories is equal to the number of the security levels required. For each category, tasks will be ordered depending on the tasks importance. The task with higher task importance will be run first down to the task with lowest task importance. If there are more than one task with the same task importance the SSM will put them in the task number order.

3) **Calculations:**

Actual Cost  $AC$  calculated and compared to budget  $b$ .

4) **Customer Confirmation:**

The customer takes last decision to confirm the service.

5) **Establishing the Service:**

The SSM starts the service and all tasks allocated to resources.

6) **Re-Calculating:**

At this step, the SSM aims to Re-calculates the  $AC$  depends on the actual running time for all resources considering dependencies that might cause delays over other resources.

A. Example: 1

Table V shows the details of a service request:

TABLE V. SERVICE REQUIREMENT FOR EXAMPLE 1

Security Level(Weight)/Importance	1	2	3	Resource	$RC_k$
1 (0.00)					
2 (0.25)			$t_1$	$R_1$	20
3 (0.50)		$t_2$		$R_2$	20
4 (0.75)					
5 (1.00)					

1) *First Scenario: 1.1:* The running time for each resource identified as follows:

- $R_1 : t_1$  and  $tm_1 = 18$  minutes
- $R_2 : t_2$  and  $tm_2 = 13$  minutes

Therefore  $AC: tm_1 = 18, RT_1: \text{Time for } R_1 = 18$

$tm_2 = 13, RT_2: \text{Time for } R_2 = 13$

$$AC = ((20*1) * (1+0+0.25)) + ((20*1) * (1+0+0.50))$$

$$AC = 25 + 30 = 55$$

Fig. 3 shows tasks timeline over the allocated resources.

**Re-Calculation:**

The SSM calculates the actual running time as follows:  
 $AC = ((20*18/60)*(1+0+0.25)) + ((20*13/60)*(1+0+0.50))$

$$\text{Therefore, } AC = 7.5 + 6.5 = 14$$

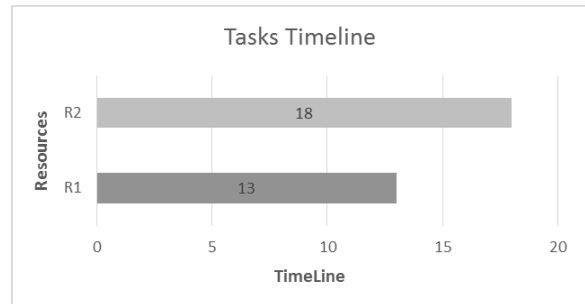


Fig. 3. Example 1. Scenario 1.1. Tasks Timeline

III. EXAMPLES OF COSTS

The following examples discusses Re-calculation with different scenarios where QoS cost  $q = 0$ , and all times represented in minutes.

B. Example: 2

Table VI shows the details of a service request:

TABLE VI. SERVICE REQUIREMENT FOR EXAMPLE 2

Security Level(Weight)/ Importance	1	2	3	Resource	$RC_k$
1 (0.00)					
2 (0.25)	$t_1$	$t_2$		$R_1$	20
3 (0.50)					
4 (0.75)	$t_3$	$t_4$		$R_2$	20
5 (1.00)	$t_5$	$t_6$		$R_3$	20

1) *Second Scenario: 2.1:* This scenario shows the time for each resource and the tasks allocated with Fast-Track as follows:

- $R_1 : t_1^{FT}$  and  $tm_1 = 18$ ,  $t_2$  and  $tm_2 = 15$
- $R_2 : t_3^{FT}$  and  $tm_3 = 13$ ,  $t_4$  and  $tm_4 = 10$
- $R_3 : t_6$  and  $tm_6 = 10$ ,  $t_5$  and  $tm_5 = 5$

The dependencies are:  $t_5$  depends on  $t_1$  and  $t_6$  depends on  $t_3$ . As a result of the dependencies, the SSM finds a delay of executing tasks  $t_6$  and  $t_5$ . So,  $AC$  calculated as follows:

$$tm_1 = 18, tm_2 = 15, RT_1 \text{ Time for } R_1 = 18+15 = 33$$

$$tm_3 = 13, tm_4 = 10, RT_2: \text{ Time for } R_2 = 13+10 = 23$$

$$tm_6 = 10, tm_5 = 5, RT_3: \text{ Time for } R_3 = 10+5 = 15$$

See Fig. 4 for tasks timeline. So, the SSM adds the total waiting time to  $RT_3$  because  $tm_3 < tm_1$ :  
 $R_3 = 13+10+5 = 28$ .

**Re-Calculation:**

So,  $AC = 13.75 + 13.42 + 18.67 = 45.84$   
If the SSM ignores the delay time as follows:

$$tm_1 = 18, tm_2 = 15, RT_1 \text{ Time for } R_1 = 18 + 15 = 33$$

$$tm_3 = 13, tm_4 = 10, RT_2: \text{ Time for } R_2 = 13 + 10 = 23$$

$$tm_6 = 10, tm_5 = 5, RT_3 \text{ Time for } R_3 = 10 + 5 = 15$$

**Re-Calculation:**

Then,  $AC = 13.75 + 13.42 + 10.00 = 37.17$   
As a result,  $AC$  of  $EPT < AC$  of the actual running time.

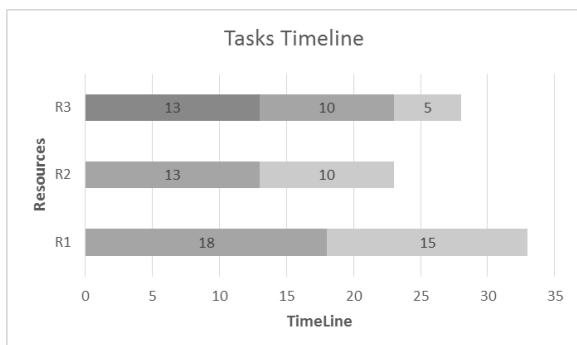


Fig. 4. Example 2. Scenario 2.1. Tasks Timeline

2) *Scenario: 2.2:* This scenario shows the time for each resource and the tasks allocated with Fast-Track as follows:

- $R_1 : t_1^{FT}$  and  $tm_1 = 8$ ,  $t_2$  and  $tm_2 = 3$
- $R_2 : t_3^{FT}$  and  $tm_3 = 10$ ,  $t_4$  and  $tm_4 = 4$
- $R_3 : t_6$  and  $tm_6 = 5$ ,  $t_5$  and  $tm_5 = 7$

The dependencies are:  $t_5$  depends on  $t_1$  and  $t_6$  depends on  $t_3$ . As a result of the dependencies, the SSM finds a delay of executing tasks  $t_6$  and  $t_5$ . So,  $AC$  calculated as follows:

$$tm_1 = 8, tm_2 = 3, RT_1: \text{ Time for } R_1 = 8 + 3 = 11$$

$$tm_3 = 10, tm_4 = 4, RT_2: \text{ Time for } R_2 = 10 + 4 = 14$$

$$tm_6 = 5, tm_5 = 7, RT_3: \text{ Time for } R_3 = 5 + 7 = 12$$

See Fig. 5 for tasks timeline. So, the SSM adds the total waiting time to  $RT_3$  because  $tm_1 < tm_3$ :  
Time for  $R_3 = 8 + 7 + 5 = 20$ .

**Re-Calculation:**

So,  $AC = 4.58 + 8.17 + 13.33 = 26.08$   
If, the SSM ignores the delay time as follows:

$$tm_1 = 8, tm_2 = 3, RT_1: \text{ Time for } R_1 = 8 + 3 = 11$$

$$tm_3 = 10, tm_4 = 4, RT_2: \text{ Time for } R_2 = 10 + 4 = 14$$

$$tm_6 = 5, tm_5 = 7, RT_3: \text{ Time for } R_3 = 5 + 7 = 12$$

**Re-Calculation:**

Then,  $AC = 4.58 + 8.17 + 8.00 = 20.75$   
As a result,  $AC$  of  $EPT < AC$  of the actual running time.

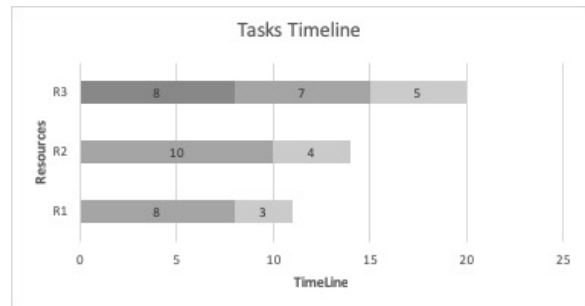


Fig. 5. Example 2. Scenario 2.2 Tasks Timeline

3) *Scenario: 2.3:* This scenario shows the time for each resource and the tasks allocated with Fast-Track as follows:

- $R_1 : t_1$  and  $tm_1 = 5$ ,  $t_2$  and  $tm_2 = 5$
- $R_2 : t_3^{FT}$  and  $tm_3 = 5$ ,  $t_4$  and  $tm_4 = 5$
- $R_3 : t_6$  and  $tm_6 = 5$ ,  $t_5$  and  $tm_5 = 5$

The dependencies are:  $t_5$  depends on  $t_3$  and  $t_6$  depends on  $t_3$ . As a result of the dependencies, the SSM finds a delay of executing tasks  $t_6$  and  $t_5$ . So,  $AC$  calculated as follows:

$$tm_1 = 5, tm_2 = 5, RT_1: \text{ Time for } R_1 = 5+5 = 10$$

$$tm_3 = 5, tm_4 = 5, RT_2: \text{ Time for } R_2 = 5+5 = 10$$

$$tm_6 = 5, tm_5 = 5, RT_3: \text{ Time for } R_3 = 5+5 = 10$$

See Fig. 6 for tasks timeline. So, the SSM adds the total waiting time to  $RT_3$ . Therefore,  $RT_3 = 5+5+5 = 15$ .

**Re-Calculation:**

So,  $AC = 4.17 + 5.83 + 10.00 = 20$

If, the SSM ignores the delay time as follows:

$tm_1 = 5, tm_2 = 5, RT_1$ : Time for  $R_1 = 5+5 = 10$

$tm_3 = 5, tm_4 = 5, RT_2$ : Time for  $R_2 = 5+5 = 10$

$tm_6 = 5, tm_5 = 5, RT_3$ : Time for  $R_3 = 5+5 = 10$

**Re–Calculation:**

Then  $AC = 4.17 + 5.83 + 6.67 = 16.67$

As a result,  $AC$  of  $EPT < AC$  of the actual running time.

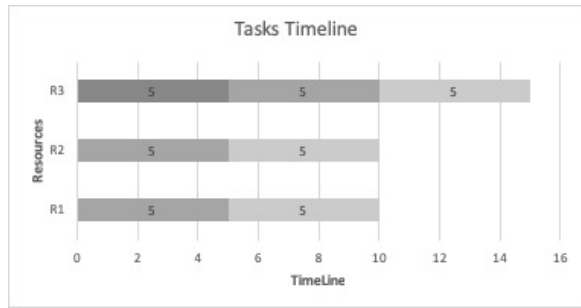


Fig. 6. Example 2. Scenario 2.3 Tasks Timeline

**C. Example: 3**

Table VII shows the details of a service request:

TABLE VII. SERVICE REQUIREMENT FOR EXAMPLE 3

Security Level(Weight)/Importance	1	2	3	Resource	$RC_k$
1 (0.00)					
2 (0.25)	$t_1$	$t_2$		$R_1$	20
3 (0.50)					
4 (0.75)	$t_3$	$t_4$		$R_2$	20
5 (1.00)	$t_5, t_7$	$t_6$		$R_3$	20

1) *Scenario: 3.1*: This scenario shows the time for each resource and the tasks allocated with Fast-Track as follows:

- $R_1 : t_1^{FT}$  and  $tm_1 = 8, t_2$  and  $tm_2 = 3$
- $R_2 : t_3^{FT}$  and  $tm_3 = 10, t_4$  and  $tm_4 = 4$
- $R_3 : t_6$  and  $tm_6 = 5, t_5$  and  $tm_5 = 7, t_7$  and  $tm_7 = 7$

The dependencies are:  $t_5$  depends on  $t_1$  and  $t_6$  depends on  $t_3$ . As a result of the dependencies, the SSM finds a delay of executing tasks  $t_6$  and  $t_5$ . So,  $AC$  calculated as follows:

$tm_1 = 8, tm_2 = 3, RT_1$ : Time for  $R_1 = 8+3 = 11$

$tm_3 = 10, tm_4 = 4, RT_2$ : Time for  $R_2 = 10+4 = 14$

$tm_6 = 5, tm_5 = 7, tm_7 = 7, RT_3$ : Time for  $R_3 = 5+7+7=19$

See Fig. 7 for tasks timeline. So, the SSM adds the delay time to  $RT_3$  because  $tm_1 < tm_3$ . Therefore,  $RT_3 = 8+7+5+7 = 27$ .

**Re–Calculation:**

So,  $AC = 4.58 + 8.17 + 18.00 = 30.75$

If, the SSM ignores the delay time as follows:

$tm_1 = 8, tm_2 = 3, RT_1$ : Time for  $R_1 = 8 + 3 = 11$

$tm_3 = 10, tm_4 = 4, RT_2$ : Time for  $R_2 = 10 + 4 = 14$

$tm_6 = 5, tm_5 = 7, tm_7 = 7, RT_3$ : Time for  $R_3 = 5+7+7= 19$ .

So,  $AC$  of  $EPT < AC$  of the actual running time. On other hand , if the SSM runs  $t_7$  first as follows:  $RT_3 = (8-7) + 7 + 5 + 7= 20$

**Re–Calculation:**

So,  $AC = 4.58 + 8.17 + 13.33 = 26.08$

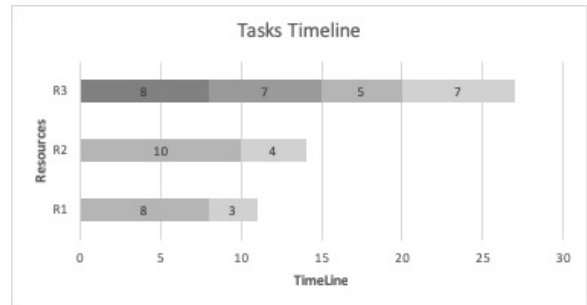


Fig. 7. Example 3. Scenario 3.1. Tasks Timeline

2) *Scenario: 3.2*: This scenario shows the time for each resource and the tasks allocated with Fast-Track as follows:

- $R_1 : t_1^{FT}$  and  $tm_1 = 8, t_2$  and  $tm_2 = 3$
- $R_2 : t_3^{FT}$  and  $tm_3 = 10, t_4$  and  $tm_4 = 4$
- $R_3 : t_6$  and  $tm_6 = 5, t_5$  and  $tm_5 = 7, t_7$  and  $tm_7 = 12$

The dependencies are:  $t_5$  depends on  $t_1$  and  $t_6$  depends on  $t_3$ . As a result of dependencies, the SSM finds a delay of executing tasks  $t_6$  and  $t_5$ . So,  $AC$  calculated as calculated follows:

$tm_1 = 8, tm_2 = 3, RT_1$ : Time for  $R_1 = 8+3= 11$

$tm_3 = 10, tm_4 = 4, RT_2$ : Time for  $R_2 = 10+4= 14$

$tm_6 = 5, tm_5 = 7, tm_7 = 12, RT_3$ : Time for  $R_3 = 5+7+12 = 24$

See Fig. 8 for tasks timeline. So, the SSM adds the delay time to  $RT_3$ , because  $tm_1 < tm_3$ . Therefore,  $RT_3 = 8+7+5+12 = 32$ .

**Re–Calculation:**

$AC = 4.58 + 8.17 + 21.33 = 34.08$

If, the SSM ignores the delay time as follows:

$tm_1 = 8, tm_2 = 3, RT_1$  Time for  $R_1 = 8 + 3 = 11$

$tm_3 = 10, tm_4 = 4, RT_2$ : Time for  $R_2 = 10 + 4 = 14$

$tm_6 = 5, tm_5 = 7, tm_7 = 12, RT_3$ : Time for  $R_3 = 5+7+12= 24$

On other hand, if the SSM runs  $t_7$  first as follows:

**Re–Calculation:**

So,  $AC = 4.58 + 8.17 + 16.00 = 28.75$

**D. Example: 4**

Table VIII shows the details of a service request:

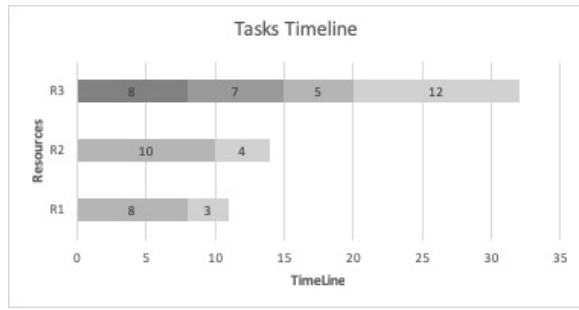


Fig. 8. Example 3. Scenario 3.2 Tasks Timeline

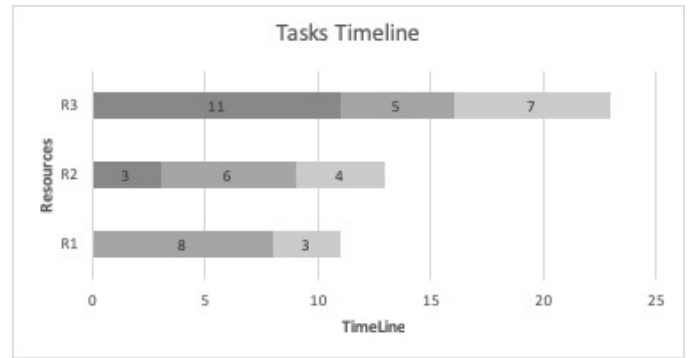


Fig. 9. Example 4. Scenario 4.1 Tasks Timeline

TABLE VIII. SERVICE REQUIREMENT FOR EXAMPLE 4

Security Level(Weight)/Importance	1	2	3	Resource	$RC_k$
1 (0.00)					
2 (0.25)	$t_1$	$t_2$		$R_1$	20
3 (0.50)					
4 (0.75)	$t_3$	$t_4$		$R_2$	20
5 (1.00)	$t_5$	$t_6$		$R_3$	20

1) Scenario: 4.1: This scenario shows the time for each resource and the tasks allocated with Fast-Track. The dependencies as follows:

- $t_6$  depends on  $t_1$
- $t_5$  depends on  $t_3$
- $t_3$  depends on  $t_2$

The resources allocated as follows:

- $R_1 : t_1^{FT}$  and  $tm_1 = 8$ ,  $t_2^{FT}$  and  $tm_2 = 3$
- $R_2 : t_3^{FT}$  and  $tm_3 = 6$ ,  $t_4$  and  $tm_4 = 4$
- $R_3 : t_6$  and  $tm_6 = 5$ ,  $t_5$  and  $tm_5 = 7$

See Fig. 9 for tasks timeline with tasks orders. So, the SSM finds a delay of executing tasks. As a results, the SSM adds  $tm_2$  to  $RT_2$ , then the SSM adds  $RT_1$  to  $RT_3$  to avoid the delays of running this service:

- $R_1 : t_1^{FT}$  and  $tm_1 = 8$ ,  $t_2^{FT}$  and  $tm_2 = 3$ ,  $RT_1 = 8+3 = 11$
- $R_2 : t_3^{FT}$  and  $tm_3 = 6$ ,  $t_4$  and  $tm_4 = 4$ ,  $RT_2 = 3+6+4 = 13$
- $R_3 : t_6$  and  $tm_6 = 5$ ,  $t_5$  and  $tm_5 = 7$ ,  $RT_3 = 11+5+7 = 23$

#### IV. COMPARISON AGAINST CURRENT MODELS

This section provide a comparison of the SSM with a current work.

The aim of Tripathy and Patra [14] approach is to execute the high priority tasks first over the allocated resources  $R_K$  needed. On the other hand, the SSM justified the priority just to order the tasks with five levels.

Tripathy and Patra approach allows a single task to run over a number of resources  $R_k$  at the same time. The Task set  $job_i(j,k,l)$  identified as the following:

- $i$  serves as tasks id
- $j$  identifies the number of resources required
- $k$  serves as task time
- $l$  to set task priority

For example, for a service request:

- $t_1:(N = 2, tm_1 = 5, pl_1 = 1)$
- $t_2:(N = 6, tm_2 = 10, pl_2 = 5)$
- $t_3:(N = 2, tm_3 = 5, pl_3 = 4)$
- $t_4:(N = 2, tm_4 = 5, pl_4 = 2)$

See Fig. 10 for tasks timeline and allocated resources for each task.

The SSM analyses this service request as follows:

- Set  $q = 0$ .
- $hw_j = 0.00$  for all task  $st_i$ , therefore,  $Rw_i = 0.00$  for all  $R_k$
- All  $RT_k$  for  $R_k = 15$ .
- $AC$  will be calculated from equation No. 6.  
 $R_k \text{ Cost} = (15*20)/60$   
 $AC = 5 * 6 = 30$

#### V. DISCUSSION

This section discusses previous examples and the comparison against current models.

In all examples, the SSM has applied the calculating steps produced by [9]. The calculating steps help to identify if the service includes tasks dependencies or not. Because of the dependencies can make the scheduling and the executing process very complicated.

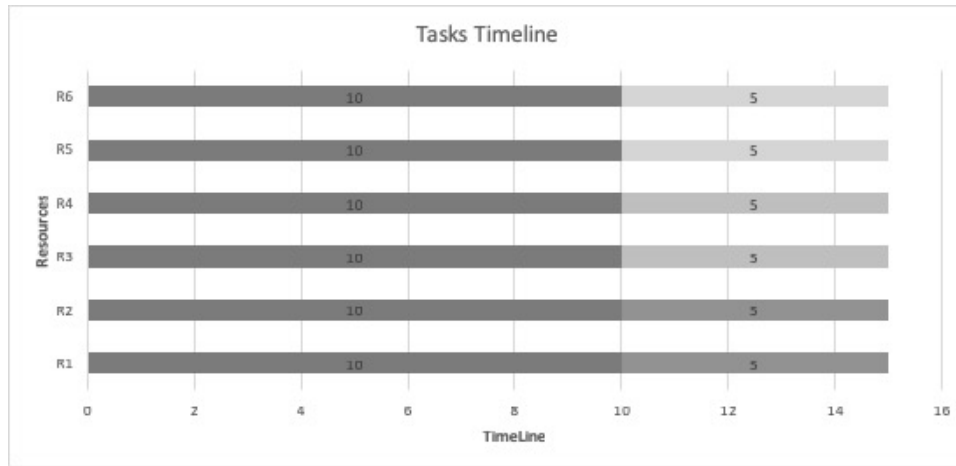


Fig. 10. Resources and Tasks Timeline

Each examples of different possible scenarios present different calculated service *AC* that can fulfil the customer expectations of service security, time, cost, and QoS.

serves these five features. However, there might be a slight delay of service time when applying the SSM on the same example included in Tripathy and Patra [14] approach.

TABLE IX. SSM FEATURES OVER ALL EXAMPLES

Feature/Examples	1	2	3	4	5
No. of Tasks (single task or set of tasks)	2	6,7	7	6	4
Dependencies	NA	✓	✓	✓	NA
Security	✓	✓	✓	✓	basic
QoS q	0	0	0	0	0
No. of Task Importance/Priority levels	5	5	5	5	5,3
<i>AC</i> vs Initial Cost	less	less	less	less	less
<i>AC</i> of <i>EPT</i> vs <i>AC</i> of Running Time	equal	less	less	less	less

Table IX shows the differences in the services for all examples. The differences include the Number of tasks for each examples, Security, QoS, Number of Task Importance levels, *AC* against Initial Cost, and *AC* of *EPT* against *AC* of Running Time.

For all examples,  $AC < \text{the Initial cost}$ . However, it is possible to have a scenario that finds there is no big different of *AC* and the Initial cost.

Also, The number of tasks running in the service are different as the SSM has defined that a task can be a single task or set of tasks.

The SSM has more features presented in Fig. 11 that can be applied to the services than the current models. These Features are Security, Priority, QoS, Cost, and Time. In addition, these features can make the cloud services very flexible to meet customers requirements. For example, Security defined by the SSM to have five different levels, also there are a range of QoS levels to ensure that the service will be delivered in the required level. So, the comparison of the SSM with Tripathy and Patra [14] showed that the SSM can

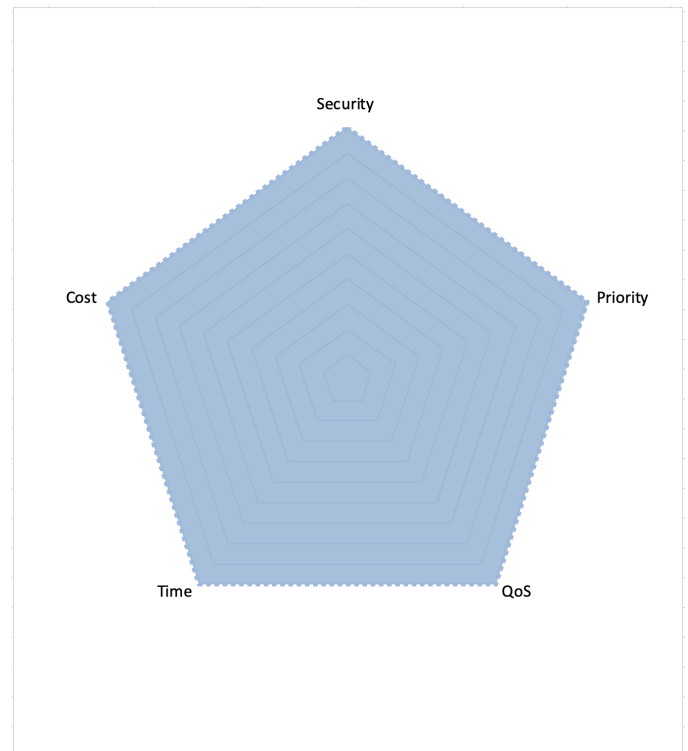


Fig. 11. SSM Features [13]

## VI. FUTURE WORK

This paper outcomes of the cost and effect suggested more investigating of the SSM. Also, further investigation required on QoS levels introduced by [15] that can be applied and considered for a service.



## VII. CONCLUSIONS

In this paper, the SSM has been examined through various worked examples. The Re-calculating step showed a major effect on the service cost  $AC$ . Furthermore, it discussed the SSM features over the examples and it showed the effect of the dependencies on the service scheduling and the service time. Finally, SSM works with more benefits than current models as it serves more features that can be applied to the services such as security, priority, QoS, Cost, and Time.

## ACKNOWLEDGMENT

Thanks to Taif University in Kingdom of Saudi Arabia for the funding.

## REFERENCES

- [1] P. Mell and T. Grance, *The NIST Definition of Cloud Computing*, Computer Security Division, Information Technology Laboratory, National Institute of Standards and Technology Gaithersburg, 2011.
- [2] L. Qiang, H. Qinfen, X. Limin and L. Zhoujun, *Adaptive management of virtualized resources in cloud computing using feedback control*, First International Conference on Information Science and Engineering, IEEE, pp. 99–102, 2009.
- [3] Y. Hu, HCH. Chen, PPC. Lee and Y. Tang, *NCCloud: applying network coding for the storage repair in a cloud-of-clouds*, FAST, pp.21, 2012.
- [4] V. Chang, D. Bacigalupo, G. Wills and D. De Roure, *A categorisation of cloud computing business models*, FAST, pp. 509–512, 2010.
- [5] D. Goutam, A. Verma, N. Agrawal, *The performance evaluation of proactive fault tolerant scheme over cloud using CloudSim simulator*, International Conference on the Applications of Digital Information and Web Technologies (ICADIWT), IEEE, 171–176, 2014.
- [6] T. Dillon, C. Wu, and E. Chang, *Cloud computing: issues and challenges*, *International Conference on Advanced Information Networking and Applications*, IEEE, 27–33, 2010.
- [7] S. Subashini and V. Kavitha, *A survey on security issues in service delivery models of cloud computing*, *Journal of network and computer applications*, Elsevier, 34, 1, 1–11, 2011.
- [8] A. Sheikh, M. Munro and D. Budgen, *Systematic Literature Review (SLR) of Resource Scheduling and Security in Cloud Computing*, *International Journal of Advanced Computer Science and Applications (IJCSA)*, IEEE, Vol. 10, No. 4, pp. 35–44, 2019.
- [9] A. Sheikh, M. Munro and D. Budgen, *Cost and Effect of Using Scheduling Security Model in a Cloud Environment*, *International Conference on Cyber Security and Cloud Computing and Edge Computing and Scalable Cloud (CSCloud/EdgeCom)*, IEEE, pp. 95–101, 2019.
- [10] P. Watson, *A multi-level security model for partitioning workflows over federated clouds*, *Journal of Cloud Computing*, Springer, 1, 1, pp 1-15, 2012.
- [11] A. Sheikh, M. Munro and D. Budgen, *SSM: Scheduling Security Model for a Cloud Environment*, *International Conference on Cloud and Big Data Computing (ICCBDC 2018)*, ACM, pp. 11–15, 2018.
- [12] A. Sheikh, M. Munro and D. Budgen, *Evaluating a Cloud Service using Scheduling Security Model (SSM)*, *International Journal of Advanced Computer Science and Applications (IJCSA)*, IEEE, Vol. 10, No. 10, pp. 525–532, 2019.
- [13] A. Sheikh, *Scheduling Security Model for a Cloud Environment*, PhD thesis, Durham University, 2020.
- [14] L. Tripathy and R.R. Patra, *Scheduling in cloud computing*, *International Journal on Cloud Computing: Services and Architecture (IJCCSA)*, 4, 5, pp.21-7, 2014.
- [15] H. Ramadan, and D. Kashyap, *Quality of Service (QoS) in Cloud Computing*, *International Journal of Computer Science and Information Technologies (IJCSIT)*, vol. 8, No. 3, pp. 318–320, 2017.