Towards Optimized Security-aware (O-Sec) VM Placement Algorithms

Motlatsi Isaac Thulo and J. H. P. Eloff
Cybersecurity and Big Data Science Research Group, Department of Computer Science, University of Pretoria, Pretoria, South Africa
u15277489@tuks.co.za, eloff@cs.up.ac.za

Keywords: Cloud Computing, VM Placement Algorithm, Security-aware VM Placement.

Abstract: Cloud computing is a technology that takes advantage of virtualization. Through virtualization, Virtual Machines (VMs) within the same host machine share physical resources. Cloud service providers (CSP) take advantage of virtualization by providing on-demand computing resources through the use of the Internet. In order to provide good Quality of Service (QoS) and to lower costs, CSPs need to optimize the cloud environment. This optimization can be achieved by the strategic placement of Virtual Machines (VMs) in cloud architecture, usually through VM placement algorithms. Despite these efforts, there are some remaining problems that need to be addressed. Amongst these are threats introduced by the cloud's architectural vulnerabilities. This paper, therefore, focuses on evaluating currently available VM placement algorithms. The objective is to identify VM placement algorithms that show potential to be further augmented with security features or that can be improved from a security perspective. Future work will investigate how these algorithms can be adapted to be security-aware.

1 INTRODUCTION

Cloud computing is a rapidly growing technology which takes advantage of virtualization as an underlying tool. Through virtualization, multiple instances of operating systems run on a single Physical Machine (PM) and share physical resources like CPU, memory and storage (Biran et al., 2012). Cloud Service Providers (CSPs) make use of this technology by providing an on-demand computing resources through the use of the Internet. This allows organizations to reduce their costs of maintenance and administration for their datacentres (Dong et al., 2015). The types of cloud computing models offered by CSPs are Platform as a Service (PaaS), Software as a Service (SaaS), and Infrastructure as a Service (IaaS) (Al-Haj et al., 2013). The ability of IaaS to provision VMs allows it to offer highest level of flexibility and scalability (Yuchi and Shetty, 2015). However, according to article (Lindemann, 2015) there are some failures in the sufficient protection of IaaS cloud resources. These failures are brought by, for example, the restricted view of the VMs by Intrusion Detection System (IDS) and limited anomaly detection techniques within the cloud environment. This therefore allows malicious activities, such as unauthorized inter-VM communication, that affect cloud computing users. CSPs are expected to implement countermeasures for minimizing the risk of such malicious activities but they are expected, at the same time, to provide good service at a reasonable cost.

CSPs need to provide good Quality of Service (QoS) to satisfy paying customers at lower costs. There are some ongoing research efforts on optimization of the cloud architecture to reduce costs without violating the Service Level Agreements (SLA) (Gao and Tang, 2013), (Xu and Fortes, 2011), (Kuo et al., 2015), (Ohta, 2013), (Meng et al., 2010). According to the last mentioned references, the strategic placement of VMs within the cloud architecture is the key to ensuring good QoS and reduced costs. These references, amongst others, focus on objectives such as energy-consumption minimization (Dong et al., 2015), (Vu and Hwang, 2014), resource-utilization (Ohta, 2013), (Zaouch and Benabbou, 2015), workload or network-management (Ilkhechi et al., 2015), (Vu and Hwang, 2014), overhead minimization (Andreolini et al., 2009) and time to complete job (Li et al., 2015). The said objectives of the last mentioned references all contribute towards one goal, which is optimization of the cloud environment. But they do not, however, minimize the security threats imposed in an IaaS cloud.

In addition to optimization, security also needs to be taken into consideration in the implementation of VM placement algorithms. IaaS cloud infrastructure

Thulo, M. and Eloff, J.
Towards Optimized Security-aware (O-Sec) VM Placement Algorithms.
DOI: 10.5220/0006206504110422
In *Proceedings of the 3rd International Conference on Information Systems Security and Privacy (ICISSP 2017)*, pages 411-422
ISBN: 978-989-758-209-7

has architectural vulnerabilities such as virtual machine escape (Venkata et al., 2004), and data breaches (Alani, 2014). These vulnerabilities in cloud infrastructure enable some cloud specific threats. These threats take advantage of the fact that VMs share physical resources (Afoulki et al., 2011). According to the article (Yuchi and Shetty, 2015), the compromised VMs within the cloud architecture use the shared resources as channels to disrupt, corrupt or spy on other VMs within the architecture. To address this, there are some ongoing security related research efforts in cloud computing. Amongst other security related efforts are those in articles (Li et al., 2012), (Yuchi and Shetty, 2015), (Afoulki et al., 2011), (Caron et al., 2013), (Al-Haj et al., 2013). The referenced articles incorporate security objectives for the strategic placement of VMs. Some of these, for example, focus on the creation of security groups for VMs based on the similarities of incoming traffic (Al-Haj et al., 2013). Most of the current work on addressing security objectives for IaaS cloud however ignore QoS and cost. It is therefore important to further conduct research into the notion of Optimized Security-aware VM placement algorithms (O-Sec VM Placement). This notion includes security, QoS as well as cost optimization objectives.

In this paper, currently available VM placement algorithms are studied. For each of the algorithms, QoS, cost and security objectives are evaluated. The next section discusses related work. Section 3 lists and discusses the most important currently available VM placement algorithms and then formulates the evaluation criteria. The criteria are used to evaluate the VM placement algorithms. Section 4 provides suggestions for future work.

2 BACKGROUND AND RELATED WORK

2.1 VM Placement Algorithm

The concept of VM placement algorithms originates from the consolidation of Operating Systems (OSs) within the same PM to avoid application compatibility issues (Bobroff et al., 2007). This consolidation of OSs within the same PM also increases the utilization of resources to lower running costs. In around 2007, researchers found the need to use algorithms to automatically consolidate these OSs (or VMs in this papers context) and allow dynamic placement into multiple PMs (Bobroff et al., 2007), (Kochut and Beaty, 2007). This concept is adopted by cloud computing and the VM placement algorithms are used to strategically allocate VMs to available PMs. The strategic placement of VMs is based on client requirements and aims to achieve particular objectives (Meng et al., 2010), (Ohta, 2013), (Xu and Fortes, 2011), (Lin et al., 2011). Amongst other objectives are energy consumption minimization (Lin et al., 2011),(Yang et al., 2012), load balancing (Zaouch and Benabbou, 2015), (Andreolini et al., 2009), traffic cost minimization (Biran et al., 2012), (Meng et al., 2010), (Vu and Hwang, 2014) and security-aware placement (Afoulki et al., 2011), (Al-Haj et al., 2013), (Caron et al., 2013).

In consideration of the optimization objectives deliberated in the currently available VM placement algorithms and also taking into consideration the objectives of this paper, it is important to categorize the placement algorithms into three broad categories. The following categories will be used in the remainder of this discussion: 1) VM placement algorithms that focus on optimizing QoS, 2) VM placement algorithms that focus on cost reduction, and 3) VM placement algorithms that focus on security-aware placement. The next subsections will discuss the three categories, with examples of the VM placement algorithms from the most cited papers for each category. Due to space constraints in the paper, only the first subsection that discusses the first category includes activity diagrams for the discussed VM placement.

2.2 VM Placement Algorithms Focusing on QoS

The main aim of strategic placement of VMs within the cloud architecture is to utilize the physical resource usage without violating SLA. According to studies (Biran et al., 2012), (Zaouch and Benabbou, 2015), (Xu and Fortes, 2011) this strategic placement of VMs strives to avoid performance degradation which can be caused by shared resources in the cloud. Some authors, (Zaouch and Benabbou, 2015), focus on the even distribution of VMs across the cloud architecture to utilize resources and therefore provide good QoS. Xu and Fortes (Xu and Fortes, 2011) on the other hand, strive to avoid performance degradations that are caused by heat imbalances in cloud architectures.

Among the mostly cited studies is (Biran et al., 2012), whereby the authors focus on complex but effective VM placement algorithms. This study focuses on both the physical resource and network constraints. It aims at implementing the VM placement algorithm that is capable of absorbing the varying traffic demands in cloud networks. The quadratic na-

ture of the proposed solutions brought by pairs of communicating VMs, and the involvement of multiple constraints, makes the implementation NP-Hard and complex. The authors therefore propose heuristic algorithms to solve the problem. These algorithms, as shown in Figure 1, first cluster VMs into Connected Components (CCs) where a connected component represent a number of VMs communicating to each other. They then iteratively place the CCs on different levels of a tree network starting at the root followed by the next levels of the tree, denoted by nextLevel in Figure 1. If the nextLevel is the lowest level of the tree then individual VMs are placed on PMs, otherwise CCs are placed on switches, called Virtual Hosts (VHs). If the aggregated resource requirements of a CC_d (for d = 1 to n) exceed the aggregated capacities of VH_z (for z = 1 to m), the CC_d is split among the subtrees of the root. The iterative placement of CCs on VHs result in communicating VMs being placed into PMs that are in close proximity to each other. This minimises the total traffic costs within the architecture. The aggregated multiple VM placement algorithms which originates from the work by (Biran et al., 2012) are interpreted by authors of this paper using one single activity diagram as shown in Figure 1.

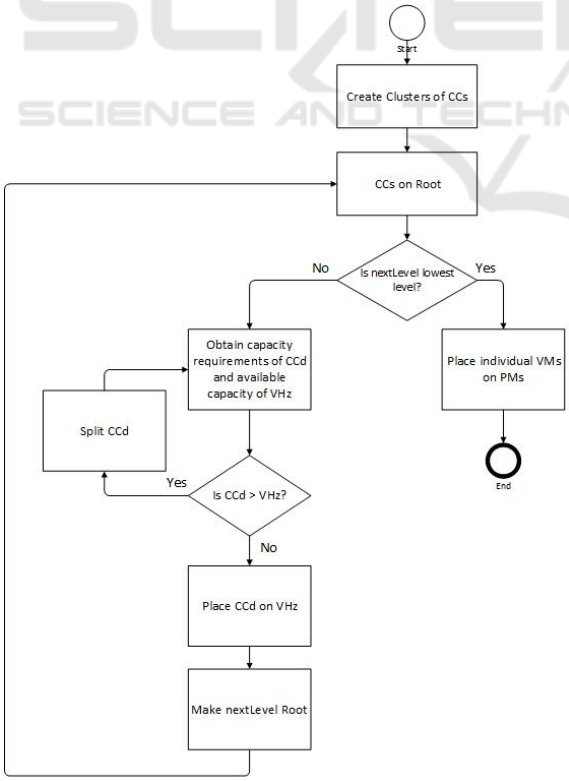

Figure 1: MCRVMP.

The problem with their solution (Biran et al., 2012) is that it works only with small to medium sized cloud architectures. In summary, the salient key points of this algorithm, from the perspective of the study in hand, are its two properties: 1) The algorithms is recursive, that is, it exploits the depth of the tree network structure by solving the placement of CCs or VMs on one level of the tree at the time. 2) The algorithm adopts a two-phase approach, it firstly places the CCs on the network and secondly expands the CCs to place the actual VMs on the PMs. The two properties together help the algorithm to place those VMs with high communication cost between them in the same PM or atleast in those that are in close proximity to each other. This reduces the overall traffic cost within the network structure and therefore minimises possibility of traffic bursts.

The other similar and mostly cited paper is (Meng et al., 2010), also focusing on consumption of network resources. The proposed VM placement algorithms in the last mentioned article also strive to place the pairs of communicating VMs close to each other to reduce traffic costs and time to complete job. Vu et al. (Vu and Hwang, 2014) also focus on consumption of network resources, but they also include energy consumption minimization as an additional optimization objective. These studies collectively consider the concept of connected components (CCs) that share data (data in-transit). They however, do not focus on the security risks of that data, which are important in the virtualized environment where resources are shared. Security risks that are not implemented in these cited papers (Biran et al., 2012), (Meng et al., 2010), (Vu and Hwang, 2014) include, amongst others the following:

- Communications within the cloud architecture use shared network resources. This means that VMs belonging to malicious users can intercept communication between sibling VMs. This can be achieved by either sniffing or spoofing IP addresses to redirect packets (Wu et al., 2010). Sniffing, on one hand, is possible if the virtual network is configured in bridge mode where a virtual bridge acts as hub and broadcasts communication. Spoofing, on the other hand, is possible if virtual network is configured in route mode and a virtual switch is used to direct packets to specific virtual interfaces.
- In clustered CCs, some communicating VMs reside in either the same PM or those in close proximity to each other. This makes it easier to identify and target a cluster of VMs that are close to each other as opposed to identifying communicating VMs that are in different locations. The

denial-of-services, which compromises availability of VMs, is simple in this case as there are fewer PMs to target.

2.3 VM Placement Focusing on Cost

Virtualization in cloud computing enables consolidation of VMs within the PMs and reduces the costs of having to buy servers for every application. In addition to this, the strategic placement of VMs within the cloud architecture also contributes to the reduction of incurring running costs (Kochut and Beaty, 2007). CSPs take advantage of these strategies and place VMs in the cloud architecture such that the least number of PMs is powered on to reduce energy costs(Yang et al., 2012).

Among the literature studied, the article (Yang et al., 2012) is one of the most cited that focuses on energy consumption minimization to save costs. The VM placement algorithm proposed in the article (Yang et al., 2012) considers both workload characteristics and energy consumption. It is modelled using modified bin packing problem, where PMs are bins and VMs are items. The items are further categorized into red and green items for data-intensive and CPU-intensive VMs respectively. To save energy, the algorithm consolidates both red and green items into the same bins of capacity 1. To achieve this, it first places red items in red bins of capacity C and then the green items into green bins of capacity (1- C). It then combines the red and green items and places them into the first bin of capacity 1. If there is a need for migrations, the VM placement algorithm allows migrations of only the green items (CPU-intensive VMs). This strives to avoid performance degradation which results from migrating data-intensive VMs away from the PMs that contain their images. The data-intensive VMs lose their performance up to 40 percent if they are migrated away from those PMs that contain their images.

In summary, the prominent point of the proposed VM placement algorithm (Yang et al., 2012), based on the perspective of the paper in hand, is that performance might be compromised if consolidation of VMs to reduce energy consumption does not consider placement of VMs based on their workload characteristics. The VM placement algorithm discussed does not only save the costs but also strives to maintain the performance of the cloud environment even after any migrations. It however, does not take into account some security risks that are introduced by consolidation of VMs into the same VMs. Some potential security risks that can be imposed by this kind of VM placement, amongst others, are the following:

- Consolidation of VMs may result in adversary users sharing the same PM. This brings higher risks of inter-VM attacks such as Denial of Service (DOS) attacks. The malicious user who owns one of the VMs consolidated within a PM can willingly take all resources of the host and deny access to other co-located VMs (Wu et al., 2010).
- Although virtualization provides isolation between VMs, there are vulnerabilities exposed by narrow interfaces in the hypervisor which might result in possible fault leakages between VMs consolidated within the same PM (Soltesz et al., 2007).

2.4 VM Placement Algorithm Focusing on Security

Security in the cloud environment is a major concern, more especially in IaaS cloud architecture. This is due to the fact that IaaS supports multi-tenant services where users share physical resources leading to potential inter-VM attacks (Caron et al., 2013). There are some on-going research efforts which strive to minimize these cloud specific attacks using strategic placement of VMs.

In the article (Afoulki et al., 2011), authors propose one of the first and most cited VM placement algorithm that considers security as an objective. This security-aware VM placement algorithm makes use of adversary lists submitted by clients to create incompatibility groups. The idea is to place VMs in such a way that no adversary users share the same physical resources. Upon request for a new VM, the VM placement algorithm checks the available PMs for VMs belonging to adversary users. If such VMs are available, the algorithm checks if it is possible to either suspend or migrate those VMs. The suspension of running VMs depends on the priorities of both the running and the newly requested VM. If both suspensions and migrations are not possible, the algorithm checks if the new VM can be scheduled for later placement, otherwise the VM request is rejected. In cases where either preemptions or migrations are possible, the placement algorithm further checks if the new VM requirements fit the resource capacities of the available PM. If the requirements fit resource capacities, the VM request is accepted and scheduled to run immediately. In summary, the salient points of the discussed VM placement algorithm (Afoulki et al., 2011), based on the perspective of the paper in hand, are: 1) the consideration of adversary users and creating of incompatibility groups in the placement of VMs and, 2) the suspension and/or migration of running VMs, which are based on the priorities as

compared to new VM request, if there is there is no available PM that suits the requirements of the new VM request.

Some other closely related papers (Al-Haj et al., 2013), (Caron et al., 2013) propose VM placement algorithms that allow users to choose the security levels they require. The VM placement algorithms in this case take user requirements as inputs and provide VM placement accordingly. Yuchi and Shetty (Yuchi and Shetty, 2015) also propose a VM placement algorithm that separates VMs according to their vulnerability status. These research efforts minimize the malicious attacks that are specific to the cloud environment using strategic placement of VMs. The discussed research efforts however, focus solely on security-aware placement of VMs and ignore the QoS and cost optimization objectives. An important security feature that can further be augmented to improve the security-aware VM placement algorithms (Afoulki et al., 2011), (Yuchi and Shetty, 2015), (Al-Haj et al., 2013), (Caron et al., 2013) discussed in this subsection is:

- The use of databases (National Vulnerability Database) of known vulnerabilities to test survivability status of the PMs. In this case, the survivability status of the PMs will periodically be checked instead of checking only when there is a request for new VM as proposed in (Yuchi and Shetty, 2015). If the survivability drops for any PM, there should be an alert that triggers the migration of critical VMs to more secure PMs. This will ensure the availability of the critical VMs which if compromised, can cause massive harm to the users.

In summary, this paper categorizes the VM placement algorithms into three: VM placement algorithms that focus on QoS; VM placement algorithm that focus on cost; and VM placement algorithms that focus on security. The first two categories ensure optimal service delivery at reduced costs. Their strategic placement however imposes some security risks which can lead to cloud specific attacks. It is therefore necessary to enhance the security features to minimize these risks. The third category considers security as its major objectives, but with possibilities to be further improved. The next sections discuss and evaluate VM placement algorithms to find the one that qualifies to further be enhanced with security features or improved from a security point of view.

3 VM PLACEMENT ALGORITHMS EVALUATION

This section introduces the VM placement algorithms that are evaluated in this paper. The algorithms and their brief descriptions are presented in tabular format as shown in Table 3 in the Appendix. The next subsections will present the evalution criteria and the obtained results after evaluation.

3.1 Evaluation Criteria

The idea is to evaluate the currently implemented VM placement algorithms to identify the algorithm that qualifies to be used in implementation of O-Sec VM Placement algorithm. This is achieved by studying the currently developed VM placement algorithms and evaluating the reflected optimization objectives for every algorithm. The overall evaluation of the VM placement algorithm depends on the scaling of the objectives for every VM placement algorithm.

3.2 Selection of VM Placement Algorithms

To achieve the goal of this paper, a number of publications that focus on VM placement algorithms are chosen among the many that are currently implemented. These publications are found in different websites and online databases through the use of the internet. Only those articles published by ACM, IEEE, Springer, Elsevier, Tailor and Francis, and Science Direct were considered. The number of publications is reduced by eliminating those that are not relevant and/or of no interest to the authors of this article. These are selected through abstract reading of all publications. A thorough studying of the remaining publications narrows the final selection to only ten publications. These are the publications which focus on VM placement algorithms evaluated in this paper.

3.3 Step-by-step Evaluation of the Algorithms

1. Identify the optimization objectives required for the O-Sec VM placement algorithm.
2. Rate the objectives according to their importance, using constant values 1 to 4 which represent the importance of the objective.
   4 = *critical importance*,
   3 = *very important*,
   2 = *important*, *and*
   1 = *necessary*

3. Identify the VM placement algorithms, state their merits, and then evaluate the objectives for every algorithm using the scale below:
   2 = *objective exists fully*,
   1 = *objective partially exist*, *and*
   0 = *does not exist*

4. Use the formula to score each VM placement algorithm to find those that qualify to be modified in the implementation of O-Sec VM placement algorithm.

3.4 VM Placement Algorithms Evaluation

Step 1: Identifying Optimization Objectives for O-Sec VM Placement Algorithm

For the VM placement algorithm to qualify for the final selection, certain optimization and security objectives need to be taken into consideration. The identification of these objectives depends on their importance, taken from the previous studies (Suseela, 2013), (Vu and Hwang, 2014), (Andreolini et al., 2009), (Shneiderman, 1984), (Biran et al., 2012), (Caron et al., 2013). Next are the optimization and security objectives and motivation on why they are chosen among the rest.

- **Elasticity:** PMs within the cloud architecture have limited amount of physical resources (Suseela, 2013). With the increasing workloads within the VMs, which are hosted in PMs, there are chances of resource demands exceeding the limit. In such cases, there is a need to migrate some VMs with higher demands to other PMs that can accommodate the workload (Andreolini et al., 2009). It is therefore critically important to take migrations into consideration to allow changes in workloads.
- **Energy Consumption:** The rapid increase of energy consumption in cloud architectures is influenced by the increasing demand for cloud computing resources. Energy consumption is a contributing factor to cost in the cloud, this means high energy consumption results in high management costs. According to (Vu and Hwang, 2014), energy consumption in clouds is a worldwide concern. There are therefore some on-going research efforts that concentrate on trying to find different ways to save energy. It is therefore extremely important to include it in this study too.
- **Response Time:** In order to provide good QoS, time to complete job is an essential tool. The unexpected delays in computing frustrates users (Shneiderman, 1984). In the same manner for cloud computing services we require services that take the least time to complete. The response time in this regard is the time taken by an algorithm to place the VM into its host. Although it is not a critical objective, but it is necessary in cloud computing.
- **VM Communication:** Modern data-centers experience some dynamic traffic change due to the adoption of dynamic multi-path routing schemas and varying peak times (Biran et al., 2012). This means excessive change in traffic can result in traffic bursts. It is therefore important to consider the communication between VMs and ways to minimize these traffic bursts.
- **Reduced SLA Violations:** Normally in cloud computing, the clients submits the computing requirements known as SLA (Caron et al., 2013). For every provisioning of the cloud resources, these SLAs need to be taken into consideration not to be violated. It is therefore equally important to consider it in the new implementation as it also helps provision some cloud resources.
- **Security-aware Optimization:** In addition to these optimization objectives is the security-aware VM placement, which is the main objective of the study. The importance of this objective is to minimize some security risks within the cloud environment. This includes, but is not limited to, cloud specific attacks such as inter-VM attacks.

There are some other optimization objectives that are not considered in this paper even though they are also important, these are: resource utilization, overheads minimization, heterogeneity of data-centres, and clustering of VMs.

Step 2: Rating of the Optimization Objectives for O-Sec VM Placement Algorithm

The rating of the optimization objectives based on their importance are presented in tabular format shown in Table 1.

Step 3: Evaluation of the Optimization Objectives of each VM Placement Algorithm

The next step after rating the required objectives for the O-Sec VM placement algorithm is to evaluate the objectives considered in the implementation of the currently available VM placement algorithms. The currently implemented VM placement algorithms, their merits and the optimization objectives evaluation are presented in Table 4 in the Appendix.

Table 1: Objectives Rating Table.

Objectives	Ratings
Elasticity	4
Response time	2
Power Consumption	4
Reduced SLA Violations	3
VM Communication	2
Security	2

Step 4: Scoring VM Placement Algorithms for Final Selection

The final selection process is based on a process introduced in (Eloff, 1983). In this process, a formula is used to calculate the percentage score in order to find the VM placement algorithm that best fit the evaluation criteria. The formula is simplified before usage to suit the research efforts of the paper in hand.

3.4.1 Deriving the Formula

In order to calculate the required optimization objectives percentage, the equation below can be used:

$$O_j = \frac{ObjectiveEvaluation}{TotalObjectiveRating} \times 100 \quad (1)$$

$$O_j = \frac{\sum_{i=1}^{N} x_i k_i}{\sum_{i=1}^{N} x_{i(TOT)} k_{i(TOT)}} \times 100 \quad (2)$$

Where, x_i is the evaluation factor allocated to the presence of the objective, k_i is the importance rating of each objective, $x_{i(TOT)}$ is the evaluation factor allocated to the presence of the objective in an ideal situation, $k_{i(TOT)}$ is the importance rating for an ideal situation, and N is total number of objectives used for evaluation.

For an ideal VM placement $x_{i(TOT)} = 2$ for all placement algorithms. This makes,

$$O_j = \frac{\sum_{i=1}^{N} x_i k_i}{2\sum_{i=1}^{N} k_{i(TOT)}} \times 100 \quad (3)$$

And therefore,

$$O_j = 50\frac{\sum_{i=1}^{N} x_i k_i}{\sum_{i=1}^{N} k_{i(TOT)}} \quad (4)$$

In this paper, $k_{i(TOT)}$ is a constant as the importance of the objectives is known (objectives already rated).

The value for $k_{i(TOT)} = 18$, and $N = 6$, therefore,

$$O_j = 0.46\sum_{i=1}^{N} x_i k_i \quad (5)$$

For all,

$$1 \leq j \leq M$$

Where M is total number of evaluated placement algorithms

3.4.2 Using the Derived Formula

Using the derived formula, we calculate the objectives percentage for each of the studied VM placement algorithms. The algorithm that has the highest percentage is assumed to be the one that considers the most important objectives according to our scaling. After the calculations, it is found that the best placement algorithm to be modified in order to implement the O-Sec placement algorithms is the **Traffic and Power-aware VM Placement (TPVMP)**. The percentage score of all the algorithms evaluated are shown in Table 2.

Table 2: VM Placement Algorithms Scores.

VM Placement Algorithm	**Score**
RFFA	7.36
VM Placement Algorithm to Minimize PM Count	9.2
Dynamic Load Management	3.68
MCRVMP	3.68
SEP-Pack and Dynamic Algorithm	8.28
TPVMP	11.04
Migration Based VM Placement and Direct Placement	5.52
A security-aware Scheduler	6.9
Modified k-means Clustering in VM Placement	5.52
Security Metrics Security-aware VM Placement Algorithm	3.68

4 TPVMP AS A CANDIDATE TO BE CONSIDERED FOR O-SEC VM PLACEMENT

The results obtained from the evaluation criteria in this paper show that TPVMP qualifies to be a VM placement algorithm with potential to be further augmented with security features. The implementation of TPVMP discussed in article (Vu and Hwang, 2014) focuses on the following objectives: VM communication , energy consumption, and elasticity. Taking into consideration the VM communication objective, the algorithm strives to reduce the total traffic costs within the cloud networks by placing communicating VMs into clusters. These clusters are then placed in those PMs that are close by to each other. It is envisaged that this VM communication objective of TPVMP algorithm can be expanded and employed to possibly include the following security objectives:

- Initial placement of VMs using TPVMP on virtual PMs, and then rearrangement of these VMs based on user security requirements and the co-located VMs. This must not violate the maximum distance between communicating VMs considered in original TPVMP. This requires submission of user security requirements which state the security levels required using metrics as proposed in (Caron et al., 2013). In addition to this, users need also to submit adversary user lists in order to create incompatibility groups. These groups will help minimize planned security attacks brought by adversary users owning the VMs located in the same PM.
- The use of a virtual switch within the hypervisors which allows creation of vNets can help isolate user traffic. This concept is discussed by Alfoulki et al. (Afoulki et al., 2011) and ensures that the traffic generated by cloud users cannot be seen by other unauthorized cloud users. This reduces the probability of interception and therefore strives to maintain the confidentiality, integrity and sometimes availability of user data that is in transit.

The other objective of focus in TPVMP is energy consumption. The algorithm addresses this objective by consolidating VM into high capacity servers and switching of as many low capacity servers as possible. This objective goes hand-in-hand with the third objective of focus in the implementation of TPVMP. The third objective, Elasticity and/or migrations, considered in this VM placement algorithm build towards the consolidation of VMs to save energy. For future implementation, these objectives conceive as a possibility that they can also be extended to include the following security objectives:

- Consolidate the VMs within the high capacity PMs taking into consideration the security level requirements of the users, incompatibility lists, and the maximum allowed distance between the communicating VMs.
- For a new VM request, there are possibilities of having PMs that meet user security requirements containing adversary users. In such cases, migrations and/or suspension of already running VMs should be taken into consideration. The suspension of the VMs, on one hand, should depend on the priority of the new VM compared to the already running VMs (Afoulki et al., 2011). The migrations, on the other hand, should not violate the maximum length constraint of the communicating VMs considered in original TPVMP.

There are other possible security features that can further extend the TPVMP but not specific to the objectives of focus in TPVMP. The possible novice improvement to TPVMP that can enhance security to the placement of VMs is as follows:
The VM placement algorithm will first categorize VMs into two: critical and non-critical VMs. The examples of critical VMs are back-end databases that store sensitive data, and those of non-critical VM are application engines. The critical VMs will be consolidated into the same PMs with a constraint that: no two communicating critical VMs share the same PM. The other non-critical VMs will be consolidated on the other remaining PMs. The placement of both critical and non-critical VMs, however, needs to ensure that the distance constraint between communicating VMs considered in original TPVMP is not violated. Again the VM placement needs to ensure that the least number of PMs is used in order to minimize energy consumption. To ensure the availability of the critical VMs, PMs containing the critical VMs will continuously be monitored. An attempt to initiate communication between any of these critical VMs in the same PM will be marked as a security threat. The initiating VM will therefore be migrated to a different PM to disallow the possible unauthorized communication. This proposed security-aware placement strives to avoid inter-VM attacks through shared network channels within the hypervisor. It further ensures the availability of the critical VMs through strategic placement.

5 CONCLUSION

In this paper, authors have developed an evaluation

method to compare the optimization objectives reflected in the different VM placement algorithms. This evaluation method uses an equation which scores the different optimization objectives based on their importance rating. According to the evaluation method, the VM placement algorithm which scores the highest is assumed to have the most important optimization objectives. The idea is to further expand such an algorithm to include security-aware objectives in order to achieve an Optimized Security-aware (O-Sec) VM placement algorithm.

To achieve their goal, the authors of this paper have categorized VM placement algorithms into three. These categories are based on what the algorithms aim to accomplish; cost reduction, good QoS and security. The authors further discuss how the VM placement algorithms for each category can further be adapted or modified to accomplish the notion of O-Sec VM placement algorithm. The optimization objectives reflected in these VM placement algorithms help to identify and rate the objectives suitable for the evaluation criteria. After the evaluation process, the Traffic and Power-aware VM Placement algorithm (TPVMP) is found as a potential candidate to further be augmented with security features.

For future envision, the authors propose the possible extension to the TPVMP to include some security features. This extension takes into consideration the optimization objectives of TPVMP, which are; energy consumption, VM communication and elasticity. In addition to these, the proposed implementation will strive to place critical and non-critical VMs separately with the provision to detect potential attacks to the VMs. Whenever there is an alert for potential attacks, migrations will be used to neutralize the situation. The proposed VM placement algorithm aims to ensure the availability of the critical VMs through strategic placement.

ACKNOWLEDGEMENTS

We would like to thank Moseme Anna Thulo, who is a sister to the first author, for helping in editing the article in hand.

REFERENCES

Afoulki, Z., Bousquet, A., and Rouzaud-Cornabas, J. (2011). A security-aware scheduler for virtual machines on IaaS clouds. *Report 2011*.

Al-Haj, S., Al-Shaer, E., and Ramasamy, H. V. (2013). Security-aware resource allocation in clouds. In *Services Computing (SCC), 2013 IEEE International Conference on*, pages 400–407. IEEE.

Alani, M. M. (2014). Securing the cloud: Threats, attacks and mitigation techniques. *Journal of Advanced Computer Science & Technology*, 3(2):202.

Andreolini, M., Casolari, S., Colajanni, M., and Messori, M. (2009). Dynamic load management of virtual machines in cloud architectures. In *International Conference on Cloud Computing*, pages 201–214. Springer.

Biran, O., Corradi, A., Fanelli, M., Foschini, L., Nus, A., Raz, D., and Silvera, E. (2012). A stable network-aware VM placement for cloud systems. In *Proceedings of the 2012 12th IEEE/ACM International Symposium on Cluster, Cloud and Grid Computing (ccgrid 2012)*, pages 498–506. IEEE Computer Society.

Bobroff, N., Kochut, A., and Beaty, K. (2007). Dynamic placement of virtual machines for managing SLA violations. In *2007 10th IFIP/IEEE International Symposium on Integrated Network Management*, pages 119–128. IEEE.

Caron, E., Le, A. D., Lefray, A., and Toinard, C. (2013). Definition of security metrics for the cloud computing and security-aware virtual machine placement algorithms. In *Cyber-Enabled Distributed Computing and Knowledge Discovery (CyberC), 2013 International Conference on*, pages 125–131. IEEE.

Chowdhury, M. R., Mahmud, M. R., and Rahman, R. M. (2015). Implementation and performance analysis of various vm placement strategies in cloudsim. *Journal of Cloud Computing*, 4(1):1.

Dong, Z., Liu, N., and Rojas-Cessa, R. (2015). Greedy scheduling of tasks with time constraints for energy-efficient cloud-computing data centers. *Journal of Cloud Computing*, 4(1):1.

Eloff, J. H. (1983). Selection process for security packages. *Computers & Security*, 2(3):256–260.

Gao, J. and Tang, G. (2013). Virtual machine placement strategy research. In *Cyber-Enabled Distributed Computing and Knowledge Discovery (CyberC), 2013 International Conference on*, pages 294–297. IEEE.

Ilkhechi, A. R., Korpeoglu, I., and Ulusoy, Ö. (2015). Network-aware virtual machine placement in cloud data centers with multiple traffic-intensive components. *Computer Networks*, 91:508–527.

Kochut, A. and Beaty, K. (2007). On strategies for dynamic resource management in virtualized server environments. In *2007 15th International Symposium on Modeling, Analysis, and Simulation of Computer and Telecommunication Systems*, pages 193–200. IEEE.

Kuo, C.-F., Yeh, T.-H., Lu, Y.-F., and Chang, B.-R. (2015). Efficient allocation algorithm for virtual machines in cloud computing systems. In *Proceedings of the ASE BigData & SocialInformatics 2015*, page 48. ACM.

Li, K., Zheng, H., Wu, J., and Du, X. (2015). Virtual machine placement in cloud systems through migration process. *International Journal of Parallel, Emergent and Distributed Systems*, 30(5):393–410.

Li, M., Zhang, Y., Bai, K., Zang, W., Yu, M., and He, X. (2012). Improving cloud survivability through depen-

dency based virtual machine placement. In *SECRYPT*, pages 321–326.

Lin, C.-C., Liu, P., and Wu, J.-J. (2011). Energy-efficient virtual machine provision algorithms for cloud systems. In *Utility and Cloud Computing (UCC), 2011 Fourth IEEE International Conference on*, pages 81–88. IEEE.

Lindemann, J. (2015). Towards abuse detection and prevention in IaaS cloud computing. In *Availability, Reliability and Security (ARES), 2015 10th International Conference on*, pages 211–217. IEEE.

Meng, X., Pappas, V., and Zhang, L. (2010). Improving the scalability of data center networks with traffic-aware virtual machine placement. In *INFOCOM, 2010 Proceedings IEEE*, pages 1–9. IEEE.

Ohta, S. (2013). Virtual machine placement algorithms to minimize physical machine count. In *APNOMS*, pages 1–3.

Shneiderman, B. (1984). Response time and display rate in human performance with computers. *ACM Computing Surveys (CSUR)*, 16(3):265–285.

Soltesz, S., Pötzl, H., Fiuczynski, M. E., Bavier, A., and Peterson, L. (2007). Container-based operating system virtualization: a scalable, high-performance alternative to hypervisors. In *ACM SIGOPS Operating Systems Review*, volume 41, pages 275–287. ACM.

Suseela, B. B. J. (2013). Survey on VM placement algorithms. *International Journal of Engineering Trends and Technology (IJETT)*, 6(7):349–352.

Venkata, S., Kumar, K., and Padmapriya, S. (2004). A survey on cloud computing security threats and vulnerabilities.

Vu, H. T. and Hwang, S. (2014). A traffic and power-aware algorithm for virtual machine placement in cloud data center. *International Journal of Grid & Distributed Computing*, 7(1):350–355.

Wu, H., Ding, Y., Winer, C., and Yao, L. (2010). Network security for virtual machine in cloud computing. In *Computer Sciences and Convergence Information Technology (ICCIT), 2010 5th International Conference on*, pages 18–21. IEEE.

Xu, J. and Fortes, J. (2011). A multi-objective approach to virtual machine management in datacenters. In *Proceedings of the 8th ACM international conference on Autonomic computing*, pages 225–234. ACM.

Yang, J.-S., Liu, P., and Wu, J.-J. (2012). Workload characteristics-aware virtual machine consolidation algorithms. In *Cloud Computing Technology and Science (CloudCom), 2012 IEEE 4th International Conference on*, pages 42–49. IEEE.

Yuchi, X. and Shetty, S. (2015). Enabling security-aware virtual machine placement in IaaS clouds. In *Military Communications Conference, MILCOM 2015-2015 IEEE*, pages 1554–1559. IEEE.

Zaouch, A. and Benabbou, F. (2015). Load balancing for improved quality of service in the cloud. *International Journal of Advanced Computer Science and Applications IJACSA*, 6(7):184–189.

APPENDIX

Table 3: VM Placement Algorithms.

VM Placement Algorithm	Brief Summary
Resource-based First-Fit Algorithm (RFFA)	These algorithms, proposed in (Kuo et al., 2015) allocate VMs according to the resource requirements and the availability of resource amounts in the running PMs. To save energy, the algorithms minimize the number of running PMs by switching off those that are on idle mode. The algorithms assign new VMs to the first PMs that best suit the requirements. There are some possible migrations if the resource requirement of the running VMs change and exceed the resource capacity of the hosting PM
VM Placement Algorithm to Minimize PM Count	The proposed heuristic VM placement algorithm by (Ohta, 2013) rearranges VMs within the cloud environment in response to the load changes. The algorithm is made up of three procedures: the $judgestate$, $incrstate$ and $decrstate$. These procedures first determines the state of the PMs within the cloud architecture, then choose the sending and receiving PMs for migrations to take place in order to increase or decrease number of PMs respectively
Dynamic Load Management	The proposed VM placement algorithm (Andreolini et al., 2009) strives to reduce overheads caused by excessive migrations. To achieve this, the algorithm considers the load profile of the PM and load trend behaviour of the VMs over time to avoid false alarms that trigger migrations. The migrations are triggered only by changes that are significant and persistent. For migrations to take place, the algorithm identifies those hosts that are over-utilized then sorts VMs in those hosts according to their loads. Then the subset of VMs with highest loads is chosen for migration.
Min Cut Ratio-aware VM Placement (MCRVMP)	The proposed VM placement algorithm (Biran et al., 2012) addresses issues of unpredicted traffic bursts. This solution achieves its goal by placing VMs in such a way that there is spare capacity in every network cut. This solution consists of two algorithms that cluster VMs into CCs. They place VMs with high communicating weight in PMs that are in close proximity.
SEP-Pack and Dynamic Algorithm	The proposed VM placement algorithm (Yang et al., 2012) considers both energy consumption minimization and performance. It is a modified bin-packing algorithm where VMs are items and PMs are bins. The items are further categorized into two: red items for data-intensive VMs and green items for CPU-intensive VMs.
Traffic and Power Aware VM Placement (TPVMP)	The proposed VM placement algorithm (Vu and Hwang, 2014) considers both traffic and power consumption. In order to save energy, the algorithm consolidates VMs into high capacity servers and tries to switch off as many low capacity servers as possible. To reduce traffic costs within the cloud architecture, the algorithm tries to place VMs with high traffic weight in PMs that are close range to each other.
Migration Based VM Placement	The proposed VM placement algorithm (Li et al., 2015) considers total completion time to place a VM. The placement algorithm has two options: migration based placement, and direct placement. For migration based placement, VM is placed in a PM that takes least job completion time, but if there are limited resources in that PM to host the new VM, then some already hosted VMs within the PM are migrated.
A security-aware Scheduler	The proposed VM placement algorithm (Afoulki et al., 2011) takes lists of adversary users from clients as input. These lists are used to create incompatibility groups used by scheduler to place VMs. The algorithm ensures that no adversary users reside in the same PM.
Modified k-means Clustering in VM Placement	The proposed VM placement algorithm (Chowdhury et al., 2015) explores the clustering technique where VMs are clustered according to their CPU utilization and RAM allocation. The algorithms migrates VMs from over-utilized or under-utilized PMs using the modified k-means centroid clustering. Unlike the normal k-means centroid clustering where the initial centroid is random, this algorithm makes the initial centroid the average of the n-objects, which are VMs in this case. After clustering the VMs, the bin-packing algorithms are used to allocate VMs, starting with those that are in the highest dense cluster.
Security-aware VM Placement Algorithm	The VM placement algorithm (Caron et al., 2013) uses metric vectors to allow users to detail their security requirements, presented as $bitsets$. The clients submit their security requirements using the $bitsets$ which are easily interpreted by the scheduler.

Table 4: Objectives Evaluation Table.

VM Placement Algorithm	Merits	Objective Evaluation	
Resource-based First-Fit Algorithm (RFFA)	The VMs are placed according to their resource requirements. The idle PMs are switched off to save energy	Elasticity Response Time Power Consumption SLA Violations Communication Security	2 0 2 0 0 0
VM placement algorithms to minimize PM count	Live migrations allow rearrangement of VMs in response to load changes. This helps minimize number of running PMs	Elasticity Response Time Power Consumption SLA Violations Communication Security	2 2 2 0 0 0
A Security-aware Scheduler	The adversary users do not share the physical machine. vNets are created to isolate traffic	Elasticity Response Time Power Consumption SLA Violations Communication Security	2 0 0 2 0 2
Security metrics security-aware VM placement algorithm	Ability to test the security levels of the physical machines, allows users to state the security levels they require using metrics	Elasticity Response Time Power Consumption SLA Violations Communication Security	0 0 0 2 0 2
Dynamic Load Management	The algorithm ignores the absolute or average load state, but triggers migrations depending on the load profile over time	Elasticity Response Time Power Consumption SLA Violations Communication Security	2 0 0 0 0 0
Min Cut Ratio-aware VM Placement (MCRVMP)	Addresses issues of unpredicted traffic bursts	Elasticity Response Time Power Consumption SLA Violations Communication Security	0 0 0 0 2 0
SEP-Pack and Dynamic Algorithm	Optimizes the use of resources by ensuring placement both data-intensive and CPU-intensive VMs in every PM	Elasticity Response Time Power Consumption SLA Violations Communication Security	2 1 2 0 0 0
Traffic and Power Aware VM Placement (TPVMP)	Tries to reduce the traffic costs within the datacenter while preserving energy	Elasticity Response Time Power Consumption SLA Violations Communication Security	2 0 2 0 2 0
Migration Based VM Placement and Direct placement	Tries to minimize the total completion time to place the new VM	Elasticity Response Time Power Consumption SLA Violations Communication Security	2 2 0 0 0 0
Modified k-means Clustering in VM Placement	Concentrates on placement of VMs that are in the migration list. Modifies the available bin-packing solution algorithms	Elasticity Response Time Power Consumption SLA Violations Communication Security	2 0 1 0 0 0