Research on trusted virtual machine migration based on dynamic exponential smoothing prediction

Xin’an Zhuang a, Xuejun Yu b
Beijing University of Technology Beijing, China

a liar_zhuang@outlook.com, b yuxuejun@bjut.edu.cn

Abstract. Cloud service centers have the problems such as untrusted virtual machines and frequent untrusted migration caused by instantaneous peak. In this paper, the trusted virtual machine is measured and the dynamic exponential smoothing prediction algorithm is used to filter the migration requests, to ensure that a trusted virtual machine will migrate at the trusted migration time. The trustworthiness of the service level agreement (SLA) for cloud services is maintained, and the efficiency of the utilization of cloud service center resources is improved.

Keywords: Trusted computing; virtual machine migration; exponential smoothing prediction; cloud computing.

1. Introduction
In recent years, with the concept of cloud computing, virtualization technology has been widely concerned. As the carrier of virtualization, the load of the physical host will affect the utilization efficiency of cloud resources. In order to realize the overall load balancing of the physical host, when the load value of a physical host exceeds a reasonable range, some virtual machines running on the physical host need to be moved out, and the dynamic migration technology of the virtual machine will be used. Dynamic migration is the technique of migrating the entire system operating environment, including the operating system, from one physical host to another while keeping the virtual machine running, and resuming operations on the destination host[1].

Since cloud computing is offered externally as a service, the quality and credibility of service has become a key factor to determine the quality of cloud service providers. As the main body of the whole migration process, the credibility of the virtual machine is the basis of the service quality of cloud service center, therefore the integrity and credibility of the virtual machine should be ensured before migration. In the process of dynamic migration, although it will only cause a few tens of milliseconds of downtime that is difficult for cloud service users to perceive. The act of migration is also accompanied by the migration of a large amount of data, frequent migration will lead to data center performance degradation, affecting the quality of application operations, increasing the rate of data center SLA violations, thus making the SLA less credible[2].

On the credibility of virtual machine migration, a series of in-depth studies have also been done by domestic and international scholars: Zhu Yuanxi proposed a migration scheme to protect the atomicity of virtual trusted root migration, platform identity proof and integrity proof of platform and virtual machine in the process of trusted virtual machine migration, and proposed the security protection of...
active monitoring of migration behavior Mechanism[3]. Tal garginkel, Jim Chow, Mendel Rosenblum and Dan Boneh proposed Terra, a flexible trusted computing structure. Its main idea is to build a trusted virtual machine monitor TVMM, only when both the privileged virtual machine and the virtual machine monitor are trusted can the virtual machine be migrated [4]. For the filtering of migration requests, Liu Yuanyuan et al, used the autoregressive model to predict the load value, and achieved better load balancing[5]. Zhao Chun and others proposed to filter untrusted migration requests based on dynamic threshold adjustment[6].

Traditional virtual machine technology have been developed and innovated for many years, but there are still challenges in some aspects [7]. This paper mainly focuses on the following points for further research:

1. Virtual machines are software programs that run on hardware, and in terms of security, the system faces not only the threat of ordinary physical machines, but also the threat of specialized attacks on virtual machines [8]. The traditional server is lack of means to ensure the integrity and correctness of the BIOS startup, and it can not guarantee that the hardware configuration and operating system of the server have not been tampered with. If the source node of the virtual machine to be migrated can not be trusted, then the subsequent migration operation is also based on the untrusted virtual machine migration.

2. As for the timing of migration, some traditional manufacturers trigger migration when CPU utilization exceeds the threshold. For example, OpenStack uses overselling to provide cloud computing resources to third parties by multiplying the original number of CPUs in the compute nodes by a factor greater than one. This method is naturally prone to scramble for CPU resources, leading to frequent migration. Considering the cost of migrating virtual machines, it can be high, and untrusted migration may further reduce the performance of cloud computing data centers. Therefore, all migration requests should be screened to select a more accurate and reliable migration time.

In this paper, we will apply the trustworthiness principle to ensure the trustworthiness of migration to address the above-mentioned problems in dynamic migration [9]. First, the TPM(trusted platform module) is used to build a trusted platform to measure the integrity of virtual machines and establish a complete chain of trust; and then the untrustworthy migration timing is filtered. The whole structure is divided into three sections:The first section is the basis of the research, which briefly introduces the related technologies involved in this paper. The second section is the scheme design of the experiment, and the third section is the simulation experiment and experimental data display.

2. Research Base

2.1. Trustworthiness measurement

Trustworthiness means that the process and result of execution are consistent with the expectation. The trustworthiness measurement is to collect the state of the detected software or system, and compare the measurement result with the reference value to see whether it is consistent. If it is consistent, it means that the verification has passed. If it is inconsistent, it means that the verification is failed[10]. TPM is the core security computing component of the trusted computing platform, and it is an important tamper resistant component in the platform [11]. It can provide an integrity measurement mechanism to provide credible evaluation for the start-up process of virtual machine, so that machine users can make correct judgments after the system is tampered.

PCRs (Platform Configuration Registers) are components dedicated to storing metric summary values, which are located in the TPM and represent the current configuration state of the platform [12]. Integrity metric is a process of storing metric values, stored in PCR, and the contents in PCR can only be changed during system operation by extending the operation as in Equation 1, old PCR[i] is the value stored in PCR[i] before the extension operation, new value is the integrity metric value to be stored in PCR[i], and the symbol "||" indicates a concatenation character. Therefore, the formula means that when an integrity measure hashvalue is to be extended into PCR[i], the two 160-bit hashes of old PCR[i] and hashvalue are concatenated, and the resulting string is hashed by the hashing algorithm, where the output 160-bit hash is stored as new PCR[i] into PCR[i]. An iterative way of computing hash values is used to
ensure the sequential nature of the extended operation. Since the hashing algorithm itself is irreversible, the extension operation is also irreversible.

\[
\text{new PCR}[i] = \text{HASH (old PCR}[i] || \text{new value})
\]  

(1)

Thus, each new measure is related to the previous one. On the premise of trusting the current link, the link evaluates the security of the next link. After confirming the trustworthiness of the next link, the control is transferred to the next link, and then the trust chain is formed. The biggest advantage of this chain trust measurement model is that it realizes the basic idea of trusted computing, which is, a step-by-step measurement and a step-by-step trust [13].

2.2. Exponential smoothing prediction algorithm

Exponential smoothing is proposed by Robert G. brown. He deliberates that the situation of the time series is stable and regular, therefore the time series can be reasonably postponed. The more recent past situation affect the more recent future situation to some extent, so a greater weight is given to the more recent data [14]. Exponential smoothing prediction algorithm is divided into the first exponential smoothing method, the second exponential smoothing method and the third exponential smoothing method. Their common idea is that the predicted value is the weighted sum of the previous observations, and different weights are given to different data, newer data are given more weight and older data are given less weight.

The three exponential smoothing methods have their own adaptive scenarios. The first exponential smoothing prediction is suitable for the scenario with no obvious trend change of the time series, the second exponential smoothing is suitable for the scenario with linear trend, and the third exponential smoothing is suitable for the scenario with quadratic trend change of the time series. Although the first exponential smoothing method can better predict the trend without obvious change, it has a large lag, and the load trend of the cloud service center is not a quadratic curve, so the third exponential smoothing method is not applicable. The second smoothing method can better predict the load trend and make a reasonable judgment on the load status of the cloud service center. Therefore, this paper uses the second exponential smoothing method to predict whether the migration request is credible.

The quadratic smoothing method is based on the first smoothing method, which can not forecast alone and, it must cooperate with the first exponential smoothing method to establish the mathematical model of forecast, and then use the mathematical model to determine the forecast model.

The definition of one-time exponential smoothing method is as follows:

\[
S_t^{(1)} = \alpha y_t + (1 - \alpha)S_{t-1}^{(1)}
\]  

(2)

Where \(S_t^{(1)}\) denotes the predicted value of time period \(T\), \(\alpha\) denotes the smoothing coefficient, \(y_t\) is the actual value of time period \(T\), \(S_{t-1}^{(1)}\) represents the predicted value of time period \(t-1\). Based on the first exponential smoothing, the second exponential smoothing method is defined as:

\[
S_t^{(2)} = \alpha S_t^{(1)} + (1 - \alpha)S_{t-1}^{(2)}
\]  

(3)

Therefore, the model of quadratic smoothing method is as follows:

\[
\hat{Y}_{t+T} = a_t + b_t \cdot T
\]  

(4)

\[
a_t = 2S_t^{(1)} - S_t^{(2)}
\]  

(5)

\[
b_t = \frac{\beta}{1 - \beta} \left( S_t^{(1)} - S_t^{(2)} \right)
\]  

(6)

Where \(t\) is the current time, \(T\) is the time period from the current time to the future time \(t + T\), \(\hat{Y}_{t+T}\) represents the predicted value of the \(T + T\) period, \(a_t\) is the intercept, \(b_t\) is the slope.
3. Trusted virtual machine migration based on dynamic exponential smoothing prediction

This section is the scheme design. As shown in Figure 1, the first part for the virtual machine level using level metrics, metrics and expected results to determine whether the virtual machine is trusted, metrics completed the virtual machine is a trusted virtual machine. The single-threshold migration policy is then improved: when the load of the source host exceeds 80% in the figure, a migration request is generated according to the single-threshold policy, and the time series is introduced as a variable to determine whether the migration request is credible by observing the load values under multiple consecutive time series and making a prediction of the next load value, thus filtering out the migration requests that are not credible due to transient peaks.

![Figure 1. Overall scheme design](image1.png)

3.1. Trusted measurement of virtual machine

Figure 2 shows the process of trust chain establishment: TPM conforms to TCG specification and has unchangeable BIOS boot block code. The rest of the BIOS is a short code that can not be measured. Taking BIOS boot block as the root of reliability measure (RTM), the RTM is used to measure most of the remaining BIOS, namely power on self checking BIOS, and the hashvalue of the measure is extended to PCR of TPM to generate measure log and complete the first level measure. The BIOS measures the next level, namely operating system layer (VM OS), and the measure value is extended to PCR to generate measure log and give control to the VM, thus, the trusted chain of virtual machine startup is established.

![Figure 2. Trust chain establishment process](image2.png)

The Linux system used in this paper is a open source software. The source code can be obtained from the official website, and the integrity ratio can also be obtained from the integrity information database published by the trusted evaluation center, and stored in the protected area of the hard disk. If the
integrity measurement value is inconsistent with the comparison value in the whole process of establishing the trusted chain, the TPM will stop loading the operating system.

3.2. Filtering trusted migration requests based on dynamic exponential smoothing prediction

To ensure that small transient load peaks trigger untrustworthy VM migration, the migration is only triggered when the load exceeds a certain migration set rule. Here the migration timing that does not reach the set rule is identified as untrustworthy migration timing, and the migration timing that meets the set rule is considered trusted migration timing. Table 1. shows the specific credibility statement.

| Serial number | Behavior                                                                 | Credible or not |
|---------------|--------------------------------------------------------------------------|-----------------|
| 1             | Migrate if threshold is exceeded in a single instance                     | not credible    |
| 2             | Migrate if the threshold is exceeded n times in a row but the predicted value is below the threshold | not credible    |
| 3             | Migrate if the threshold is exceeded n times in a row and the predicted value is also above the threshold | credible        |

The setting of the threshold itself will affect the radicality of the migration behavior: low threshold will lead to more frequent migration, which will lead to low resource utilization of the host. High threshold will improve resource utilization, but it will affect the performance of the application, and often trigger untrusted migration requests by transient peak load. Therefore, a simple strategy based on fixed threshold will cause serious problems compared with other measures to reduce the migration time, such as memory compression algorithm, to determine the credible migration time is actually to reduce the unnecessary migration in the upstream.

According to the description and definition of the quadratic exponential smoothing method in the research base, this paper uses continuous threshold observation and threshold prediction based on exponential smoothing to determine the credible migration opportunity. First, the instantaneous peak value is processed, and the load value of s times CPU is continuously monitored. Only when s times load exceeds the threshold value, the subsequent load will be predicted, otherwise it will be judged as an untrustworthy migration opportunity. In the prediction stage, the above quadratic exponential smoothing method is used to calculate the intercept \( a_t \) and slope \( b_t \). According to the model, the load value of time \( t \) is obtained, and the load value is compared with the threshold value. If the load value is lower than the threshold value, it is an untrusted migration opportunity. If the load value is still higher than the threshold value, it is a trusted migration opportunity. It indicates that the load of the current physical host is in a high load state and will continue, so the virtual machine migration is triggered.

In the traditional exponential smoothing prediction model, the selection of smoothing coefficient determines the prediction accuracy. When the detected data is horizontal, rising and falling, different numbers within 0-1 will be selected as the smoothing coefficient. However, the load value of PM is dynamic, and a single smoothing coefficient is not enough to meet the changing forecasting demand. Therefore, the dynamic smoothing coefficient is selected, and the smoothing coefficient adjusts itself with the change of data to obtain more accurate forecasting results, the specific prediction process is shown in Figure 3:
Figure 3. A metric process for credible migration timing

(1) Obtain the actual data, and take the continuously detected historical load value of group t PM as the initial data of prediction.

(2) By iterating to find the optimal smoothing coefficient, select 0.0001 as the step size, and then the quadratic exponential smoothing method is used to calculate the sum of error squares of T group data. The smoothing coefficient corresponding to the minimum sum of error squares is the optimal smoothing coefficient.

(3) The load value of group T + 1 is predicted, and the smoothing coefficient obtained in step (2) is substituted into the quadratic exponential smoothing method for prediction to obtain the PM load value of group T + 1.

3.3. Experimental Environment
The experimental environment utilizes two PC servers in the school LAN, where the memory is 4G, the hard disk is 500g, the operating system is Ubuntu 16.04, the version of Xen is 4.4, the TPM is simulated by the open source software TPM emulator and, the version is 0.7.4. The simulator simulates the main functions of TPM, which can complete the functions required in this paper.

3.3.1. Experimental Results and Analysis. Trusted measurement of virtual machine
Before installing TPM emulator on PM, cmake, GMP, GTK and other algorithm libraries and dependent software were first installed. After the installation, first initialize and start the TPM, then start the TSS protocol stack, and finally compile and run the PCR read and extension program. First, output the value in the PCR register before modification, as shown in the figure 4, all PCR values are in the initial state.
Secondly specify the serial number of the register to be expanded by the parameter - P. Here, select PCR [0] to expand. The figure 5 shows the situation of all the values in the register after the PCR expansion operation. At this time, it can be automatically expanded in the actual scene and compared with the values saved in the protected hard disk area. If the two values do not match, the loading of the operating system will be stopped.

3.3.2. Credible measurement of migration timing. Figure 6 in Linux is used to simulate the high load scenario. The principle is to provide a random byte data stream that is never empty and decompress it, so that the CPU utilization of the thread exceeds a single threshold. The sudden overload and persistent overload are simulated by the execution time of the command.

Compared with the single threshold trigger mechanism and the single smooth coefficient trigger mechanism, the single threshold selection is 80% the same as the mainstream, and the number of times
of continuous monitoring exceeding the threshold is 4. The relatively optimal choice of the fixed smoothing coefficient and group number needs further experiments.

The first is the selection of smoothing coefficient. In this paper, 0.4, 0.5, 0.6, 0.7 and 0.8 are selected for prediction analysis. The experimental data are shown in Figure 7. It can be seen that when the smoothing coefficient is 0.6, the standard deviation is small, that is, the prediction accuracy is high. Therefore, 0.6 is selected as the fixed smoothing coefficient for subsequent experiments.

Secondly, when the number of actual data groups is different, it also has an impact on the error. It can be seen from Figure 8 that when the number of groups is greater than 10, the error and growth will slow down. Therefore, 10 groups of historical load values are selected as the actual predicted values.

After determining the more accurate smoothing coefficient and the number of groups, firstly test the prediction error comparison between single smoothing coefficient and dynamic smoothing coefficient, select 10 groups of physical host historical load value as the initial value to compare the prediction value, and the figure 9 below shows that the prediction error obtained by using dynamic smoothing coefficient is significantly lower.
Finally, we compare the migration times triggered by the three virtual machine migration strategies. Because the single threshold trigger strategy can not filter the untrusted migration caused by the instantaneous peak, and the single smoothing coefficient can not adapt to the scene of changing load, the prediction accuracy is not high, which may also cause untrusted virtual machines. Through multiple threshold observation and prediction of adaptive smoothing coefficient, we can better judge whether the migration request is necessary, that is, reliable migration.

From the experimental data shown in figure 10, it can be seen that the method used in this paper can filter the most number of untrusted migration requests. The difference between the method used in this paper and the single threshold trigger method is the filtered untrusted migration requests.

4. Conclusion
In this paper, some improvements are made to the credibility problems in the process of virtual machine dynamic migration. One is to build a trust chain with the help of TPM to measure the credibility of the virtual machine before the migration; the other is to combine an improved quadratic exponential smoothing algorithm. The migration request was screened, and the credibility measurement of the migration timing was completed. However, the study also has some shortcomings, such as the random generation of CPU load data. The study can be further researched where the simulation of the production environment load is strengthened or directly look for the real historical load.

References
[1] Chang Decheng, Xu Gaogao. Dynamic migration method of virtual machine [J]. Computer application research, 2013, 30 (4): 971-976
[2] Zhao pingting, Han Zhen, he Yongzhong. Cloud computing dynamic trust evaluation model based on SLA [J]. Journal of Beijing Jiaotong University: Natural Science Edition, 2013
[3] Zhu Yuanxi. Research and implementation of trusted virtual machine migration scheme for cloud environment [D]. 2018

[4] Garfinkel T, Pfaff B, Chow J, et al. Terra: A Virtual Machine-Based Platform for Trusted Computing [J]. ACM SIGOPS Operating Systems Review, 2003, 37(5).

[5] Liu, Yuanyuan, Gao, Qingyi, Chen, Yang. A load balancing approach for virtual machine resources in virtual computing environment [D]. 2010

[6] Zhao Chun, Yan Lianshan, Cui Yunhe, et al. Virtual machine migration algorithm based on dynamic threshold adjustment [J]. Computer applications, 2017 (9)

[7] Upadhyay A, Lakkadwala P. Secure live migration of VM's in Cloud Computing: A survey [C]// 2014 3rd International Conference on Reliability, Infocom Technologies and Optimization (ICRITO) (Trends and Future Directions). IEEE, 2014.

[8] Liu Gang, Wu Baoxi, Zhang Yao. Research on Key Technologies of trusted server platform in cloud environment [J]. Information security research, 2017 (4)

[9] Shen C X, Zhang H G, Wang H M, et al. Research on trusted computing and its development [J]. Science China Information Sciences, 2010, 53(3):405-433.

[10] Liu zwen, Feng Dengguo. Dynamic integrity measurement architecture based on Trusted Computing [J]. Acta electronica Sinica, 2010, 32 (4): 875-879

[11] Tian Jiansheng, Zhan Jing. Research and implementation of active dynamic measurement mechanism based on TPCM [J]. Information network security, 2016 (6): 22-27

[12] Ming-Fang L, Wen-Feng L I, Yang Z. An XEN Platform Based Trusted Virtual Machine Migration Protocol [J]. Computer Security, 2013.

[13] Tang Linlin. Research on integrity measurement of virtual machine migration based on Trusted Computing [D]

[14] Luo Chen Hui, Jie Jing Fang, Zhang Wei, et al. Load forecasting of cubic exponential smoothing algorithm based on dynamic coefficient [J]. Computer measurement and control, 2018, 26 (10): 147-150

[15] Shi Zhen. Improvement of exponential smoothing algorithm for time series [J]. Information systems engineering, 2017, 06 (v.42; No.245): 140 + 142