This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TPDS.2014.2314672, IEEE Transactions on Parallel and Distributed Systems

A Decentralized Cloud Firewall Framework with Resources Provisioning Cost Optimization

Meng Liu, *Student Member, IEEE,* Wanchun Dou, *Member, IEEE,* Shui Yu, *Senior Member, IEEE,* and Zhensheng Zhang, *Senior Member, IEEE*

Abstract—Cloud computing is becoming popular as the next infrastructure of computing platform. Despite the promising model and hype surrounding, security has become the major concern that people hesitate to transfer their applications to clouds. Concretely, cloud platform is under numerous attacks. As a result, it is definitely expected to establish a firewall to protect cloud from these attacks. However, setting up a centralized firewall for a whole cloud data center is infeasible from both performance and financial aspects. In this paper, we propose a decentralized cloud firewall framework for individual cloud customers. We investigate how to dynamically allocate resources to optimize resources provisioning cost, while satisfying QoS requirement specified by individual customers simultaneously. Moreover, we establish novel queuing theory based model M/Geo/1 and M/Geo/m for quantitative system analysis, where the service times follow a geometric distribution. By employing Z-transform and embedded Markov chain techniques, we obtain a closed-form expression of mean packet response time. Through extensive simulations and experiments, we conclude that an M/Geo/1 model reflects the cloud firewall real system much better than a traditional M/M/1 model. Our numerical results also indicate that we are able to set up cloud firewall with affordable cost to cloud customers.

Index Terms—Cloud Computing, Firewall, Resources allocation, System modeling.

1 Introduction

Cloud computing is becoming popular as the next infrastructure of computing platform in the IT industry [1]. With large volume hardware and software resources pooling and delivered on demand, cloud computing provides rapid elasticity. In this service-oriented architecture, cloud services are broadly offered in three forms: Infrastructure-as-a-Service (IaaS), Platform-as-a-Service (PaaS) and Software-as-a-Service (SaaS). Besides, cloud computing also brings down both capital and operational expenditure for cloud customers by outsourcing their data and business.

Despite all these charming features surrounding, security is the major concern that people hesitate to transfer their applications to cloud [2]. On one hand, traditional attacks such as Distributed Denial of Service (DDoS), viruses and phishing still exist in clouds. On the other hand, new specific attacks on the computing mechanisms of cloud have also been found, including Economic Denial of Sustainability (EDoS) attack [3], cross Virtual Machine (VM) attack [4], and so on.

It is an effective and necessary choice of establishing a cloud firewall to protect cloud data centers from all these attacks. Firewalls are the first line when defending against malicious traffic, and rule based firewalls are the most widely deployed among traditional firewalls [5], [6]. Compared to a cloud platform, traditional firewalls are generally deployed for private networks which host relatively specific services [7], [8].

To the best of our knowledge, only a few work [9], [10] have been done on cloud firewall, and both proposed a centralized cloud firewall. However, the diversity of heterogeneous services and complex attacks definitely means a large rule set and high packet arrival rate if a centralized firewall is applied for a whole cloud data center. As a result, it is hard to guarantee QoS requirement specified by cloud firewall customers. Moreover, important design factors like packet arrival rate (in both attack and non-attack period), attack (non-attack) duration and number of rules are customers specific. Therefore, it is more practical to offer a cloud firewall for individual cloud firewall customers.

Naturally, a question that arises in setting up a cloud firewall is how to price this service. From cloud firewall customers' perspective, they prefer to rent a cloud firewall from firewall providers as cheap as possible. While for cloud firewall providers, their primary goal is financial reward. Therefore, cloud firewall providers need to optimize resources provisioning cost, which offers a chance of lowering the cloud firewall price on behalf of customers without reducing providers' profit. Meanwhile, QoS requirement about the cloud firewall system specified by customers should be satisfied. There is an inherent trade-off between the two goals: to guarantee a pressed response time, large volume of resources should be invested by cloud firewall providers, which in

- *M. Liu and W. Dou are with the State Key Laboratory for Novel Software Technology, the Department of Computer Science and Technology, Nanjing University, China, 210023.*
  E-mails: mengliunju@gmail.com, douwc@nju.edu.cn.
  W. Dou is the author for correspondence.
- *S. Yu is with the school of IT, Deakin University, Victoria, 3125, Australia.*
  E-mail: shui.yu@deakin.edu.au.
- *Z. Zhang is with Department of Electronic Engineering, University of California, Los Angeles, CA 90095 USA.*
  E-mail: zzhang@ieee.org.

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TPDS.2014.2314672, IEEE Transactions on Parallel and Distributed Systems

turn increase provisioning cost (and vice-versa).

In this paper, we propose a decentralized cloud firewall framework. The cloud firewall is offered by Cloud Service Providers (CSP) and placed at access points between cloud data center and the Internet. Individual cloud customer rents the firewall for protecting his cloud hosted applications. Hosting servers of applications are grouped into several clusters, and resources are then dynamically allocated to set up an individual firewall for each cluster. All these parallel firewalls will work together to monitor incoming packets, and guarantee QoS requirement specified by cloud customers at the same time. By covering the vast cloud and firewall related parameter space, we formulate the resources provisioning cost.

As aforementioned, the essential issue to achieve a financial balance between firewall providers and customers is to optimize resources provisioning cost. In order to conduct the optimization, we need to capture mean packet response time through the firewall system. As widely adopted in cloud performance analysis [5] [11], we employ queuing theory to undertake system modeling. However, we have to point out that the cloud firewall service times follow a geometric distribution according to rule match discipline.

The contributions of this paper are summarized as follows,

- We propose a decentralized cloud firewall framework for individual cloud firewall customers. Resources are dynamically allocated to optimize the provisioning cost, and guarantee QoS requirement specified by customers at the same time.
- We introduce novel queuing theory based model M/Geo/1 or M/Geo/m for performance analysis of the proposed cloud firewall. By employing Z-transform and embedded Markov chain techniques, a closed-form expression of mean packet response time is derived.
- Extensive simulations and experiments are conducted to verify our analytical model. The simulation results claim that geometric distribution is more suitable for firewall system modeling, and give a deep insight into tradeoff among optimal resources provisioning cost, QoS requirement and packet arrival rate.

The remainder of this paper is organized as follows. In Section 2, we first introduce some preliminary knowledge about cloud firewall and then present a decentralized cloud firewall framework. In Section 3, we formulate the resources provisioning cost optimization problem with firewall service rate modeling. The novel M/Geo/1 and M/Geo/m model for system analysis are described in Section 4. Performance evaluations are conducted in Section 5, and we present further discussion in Section 6. Finally, we summarize this paper and discuss future work in Section 7. The related work can be found from the online supplementary file of this paper.

2 Cloud Firewall Framework

In this section, we first discuss several important characteristics about cloud firewall, and then present our decentralized cloud firewall framework.

2.1 Preliminary knowledge about cloud firewall

Dynamic packet arrival rate. In general, cloud services are hired by legitimate customers. However, cloud applications are also vulnerable to various attacks, and a long time attack is usually rare as they can easily detected [11]. Therefore, incoming packets to cloud firewalls are composed of long term legitimate packets and bursty attack packets. In addition, packet arrival rate is dynamically changing over the time. Moreover, arrival rate of legitimate packets from benign customers is relatively low, while attack packets for malicious purposes are usually at a high rate. In conclusion, it requires a feasible model to capture the dynamic packet arrival rate in both attack and normal period.

As a main threat to cloud availability [2], here we take DDoS attack for example. Moore et al. [12] indicated that the average DDoS attack duration is around 5 minutes, with the average DDoS attack rate being around 500 requests per second. While Yu et al. [11] presented that the mean arrival rate to an observed e-business site in normal period is lower than 10 requests per second.

On-demand resources provisioning. In order to provide a cloud firewall, firewall service providers should invest various resources to fight against possible attacks. Current CSPs usually pack resources such as CPU, bandwidth and storage into Virtual Machine (VM) instances for service. Generally, multiple VM instance types are offered and each type has a limited service capacity for a particular application, which is evident by analysis results in [13].

In our case, VM instances are launched by providers to host the cloud firewall. When packet arrival rate increases, a single VM instance tends to be incapable of handling the massive incoming packets, or the response time will violate QoS requirement specified by customers. According to QoS requirement, packet arrival rate and VM instances service rate, firewall service providers need to invest more resources on-demand by launching additional VM instances. New VM instances can be cloned based on the image file of the original firewall using the existing clone technology [14], [15]. Specifically, firewall providers have to invest different volume of resources in attack and normal period.

Cost and performance trade-off. There is an inherent trade-off between the following two goals:

- QoS requirement satisfaction. Mean packet response time requirement specified in QoS should be satisfied.
- Resources provisioning cost optimization. Resources provisioning cost of cloud firewall should be minimized as long as QoS requirement is satisfied.

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TPDS.2014.2314672, IEEE Transactions on Parallel and Distributed Systems

2.2 A decentralized cloud firewall framework

As aforementioned, each VM instance has a limited service capacity for a cloud firewall application. Hosting a cloud firewall in a single VM instance (even the most powerful one) tends to be incapable of satisfying customer specific QoS requirement. In other words, it's hard to guarantee response time through a centralized cloud firewall. Therefore, we propose a decentralized framework where several firewall run in parallel. As shown in Figure 1, hosting servers are grouped into several clusters and a VM instance is launched to host an individual firewall for each cluster. By distributing the packet arrival rate into several parallel firewalls and launching suitable VM instance for each firewall, response time through each firewall can satisfy the QoS requirement.

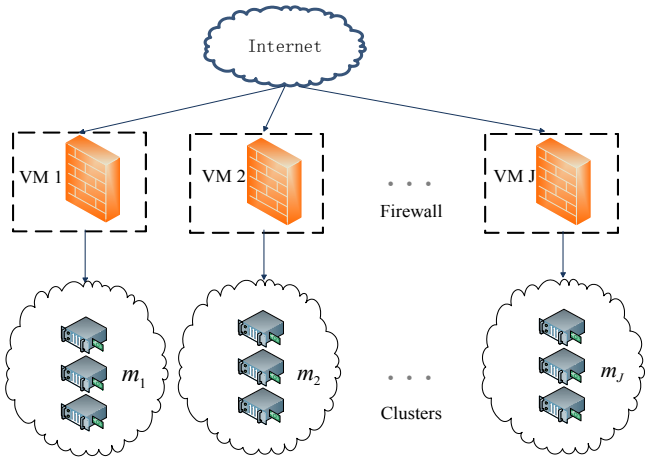


Fig. 1: A decentralized cloud firewall framework.

Suppose that there are M servers in the cloud data center hosting applications of an individual cloud firewall customer. χ^n denotes packet arrival rate to these applications in non-attack period (superscript n stands for normal or non-attack). The M servers are grouped into J clusters $(m_1^n, \ldots, m_j^n, \ldots, m_J^n)$ for processing legitimate packets. $(V_1^n, ..., V_j^n, ..., V_J^n)$ denote the set of VM instances which host the parallel cloud firewall for each cluster. $(\lambda_1^n, \ldots, \lambda_j^n, \ldots, \lambda_J^n)$ denote packet arrival rates to each firewall, then

$$\begin{cases} \sum_{j=1}^{J} m_j^n = M \\ \sum_{j=1}^{J} \lambda_j^n = \chi^n. \end{cases} \tag{1}$$

We define the corresponding variables in attack period as follows: χ^a denotes packet arrival rate to the hosting servers (superscript a stands for attack), which are grouped into K clusters $(m_1^a, \ldots, m_k^a, \ldots, m_K^a)$ for processing attack packets. $(V_1^a, ..., V_j^a, ..., V_J^a)$ denote VM instances space, and $(\lambda_1^a, \ldots, \lambda_j^a, \ldots, \lambda_J^a)$ denote packet arrival rates to each firewall. Similarly we have,

$$\begin{cases} \sum_{k=1}^{K} m_k^a = M \\ \sum_{k=1}^{K} \lambda_k^a = \chi^a. \end{cases} \tag{2}$$

3 Resources Provisioning Cost Optimization

In this section, we first formulate resources provisioning cost. As firewall service rate modeling is critical to resources provisioning cost optimization, we establish a mathematical model according to cloud firewall rule matching discipline and derive that system service times follow geometric distribution.

3.1 Resources provisioning cost

Let T^n denote the unit time interval that CSPs charge VM instances. T^a denotes average attack duration in T^n. For simplicity, the scenario that various types of attacks occur with unequal attack rate and attack duration is not covered in this paper. In fact, our model can be easily extended to this general case.

Our primary goal is to optimize resources provisioning cost, while satisfying QoS requirement at the same time. It is intuitive that resources provisioning cost for our proposed cloud firewall depends on packet arrival rate. Given χ^a and χ^b, it further relies on how many clusters (J and K) are formed. Moreover, it is determined by VM instance configuration for the parallel firewalls. In order to cover the vast cloud firewall related parameter space, the resources provisioning cost is formulated as follows,

Minimize

$$T^n \sum_{j=1}^{J} p_j^n + T^a \sum_{k=1}^{K} p_k^a \tag{3}$$

Subject to

$$\forall j \in [1, J], \begin{cases} \lambda_j^n \le \mu_j^n \\ \overline{r_j^n} \le \Delta T. \end{cases} \tag{4}$$

$$\forall k \in [1, K], \begin{cases} \lambda_k^a \le \mu_k^a \\ \overline{r_k^a} \le \Delta T. \end{cases} \tag{5}$$

Here p_j^n and p_k^a denote unit price of VM instance V_j^n in non-attack period and V_k^a in attack period, respectively (If the two VM instances are of the same type, then $p_j^n = p_k^a$). μ_j^n and μ_k^a denote service rate of the two VM instances when running the cloud firewall, which are in terms of packets per second and will be given later. $\overline{r_j^n}$ and $\overline{r_k^a}$ are response time through firewall for cluster m_j^n and m_k^a in non-attack and attack period respectively, and they also will be given later. ΔT is an acceptable response time threshold specified in firewall customers QoS requirement.

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TPDS.2014.2314672, IEEE Transactions on Parallel and Distributed Systems

The objective function (3) is to minimize resources provisioning cost for our proposed cloud firewall. Equations (4) and (5) are the conditions that have to be met when configuring VM instances for each firewall in non-attack and attack period, respectively. Concretely, QoS requirement constraint has to be met, and arrival rate to each firewall should be less than its service rate to keep the system in a stable state.

In general, CSPs specify a limitation of concurrent VM instances that are available to an account [4]. For example, this threshold is 20 in Amazon EC2. In other words, J and K are usually small. As a result, we can simply iterate J and K to find the optimal solution to equation (3). In each iteration, a greedy algorithm is applied to get an optimal cost for each J and K. Concretely, we rank the VM instances in ascending order according to service rate μ (which is determined by equation (13)), and then choose VM instance which satisfies QoS requirement ΔT with least μ. These obtained VM instances are just the VM configuration that leads to the optimal cost for each given J and K. Finally, by minimizing these optimal costs over J and K, we are able to find an optimal solution to equation (3).

In order to simplify the calculation, we assume that packet arrival rate to each firewall is proportional to number of servers included in the cluster. Then the mean packet arrival rate to firewall for cluster m_j^n and m_k^a are given by,

$$\lambda_j^n = \frac{\chi^n m_j^n}{M}, \tag{6}$$

$$\lambda_k^a = \frac{\chi^a m_k^a}{M}. \tag{7}$$

3.2 Service rate modeling

As widely applied in cloud performance modeling [5], [16], service times are generally assumed to follow exponential distribution. However, rule match discipline has to be taken into account in the context of cloud firewall. For a rule-based firewall, suppose that there are N rules $R_i (i = 1, 2, ..., N)$ in the rule database. As shown in Figure 2, all incoming requests will be examined sequentially rule by rule until a match is found. Further actions will be taken, e.g., dropping or passing the incoming requests.

We denote the probability of matching succeeds at a particular rule R_i as p_i. In the case that we have no prior knowledge about rule matching probability distribution, we suppose that the N rules share the same probability of matching, i.e., $p_1 = \cdots = p_i = \cdots = p_N = p > 0$. Let random variable Y denote the number of trials for a first match, then the probability that we obtain it at rule R_i follows a geometric distribution, which is expressed as

$$Pr[Y = i] = \begin{cases} (1-p)^{i-1} p & 0 < i < N \\ (1-p)^{N-1} & i = N. \end{cases} \tag{8}$$

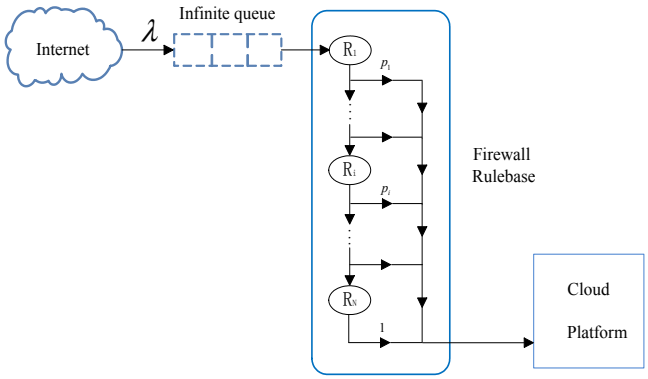


Fig. 2: Flow chart for firewall rule matching.

The mean of the geometric distribution is

$$E[Y] = \frac{1 - q^N}{p}, \tag{9}$$

where $q = 1 - p$. From a system point of view, we care about the average of system service time, which is denoted as $\bar{t}$,

$$\bar{t} = \sum_{i=1}^{N} \left(Pr[Y = i] \sum_{j=1}^{i} T_j \right), \tag{10}$$

where $T_j (j = 1, 2, \ldots, N)$ denotes match time for rule j. For VM instance V with m $(m \geq 1)$ computing cores, its service rate μ is given as,

$$\mu = m/\bar{t}. \tag{11}$$

Similarly, we assume that the N rules have the same matching time T, i.e., $T_1 = T_2 = \cdots = T_N = T$, in case we have no priori knowledge about them. Then

$$\bar{t} = \frac{T(1 - q^N)}{p}, \tag{12}$$

$$\mu = \frac{mp}{(1 - q^N)T}. \tag{13}$$

As can be seen from the last equation, service rate for each VM instance when running cloud firewall only depends on number of computing cores included and parameters related to firewall settings, i.e., p, N and T. Therefore, we only have to discriminate number of computing cores when deriving service rate μ_j^n and μ_k^a in equations (4) and (5).

4 Analytical Model

In this section, we establish an executable mathematical model for the proposed cloud firewall. Following this model, we also provide a thorough analysis of the system.

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TPDS.2014.2314672, IEEE Transactions on Parallel and Distributed Systems

4.1 An embedded Markov chain

We first define a series of variables for the following calculation. Define $\{a_n\}$ as a set of random variables at discrete time point n ($n > 0$). Then we denote random variable $\widetilde{a}$ as limit (if exists) of a_n, i.e., $\widetilde{a} = \lim_{n\to\infty} a_n$. $\overline{a}$ denotes mean value of $\widetilde{a}$, that is, $\overline{a} = E[\widetilde{a}]$.

According to performance analysis of cloud data center [17], [16], [18], it is generally accepted that service request arrival rate to the cloud server follows Poisson distribution,

$$Pr[X = k] = \frac{\lambda^k e^{-\lambda}}{k!}, k = 0, 1, ... \tag{14}$$

which describes the probability of k arrivals for a given time interval, given the mean arrival rate is λ.

In order to make our modeling, analysis feasible and practical, we make one reasonable assumption as widely applied in cloud performance analysis [5], [19]. For service times of requests, we assume they are independent and identically distributed random variables that follow a geometric distribution as described in the last subsection with mean service time $\overline{t}$. Service requests are served according to first-come-first-served (FCFS) queue discipline. According to number of computing cores, VM instances launched for hosting cloud firewall service are modeled as an M/Geo/1 or M/Geo/m queuing system.

An M/Geo/1 queuing system may be considered as a semi-Markov process [20], which can be analyzed by the embedded Markov chain technique. The fundamental idea behind this technique is that we have to select Markov points in which the state of the system is observed. In this paper, we choose the set of departure instants from service as the moments at which we model the number of tasks in system.

Let q_n and q_{n+1} represent the number of requests left behind by departure of the nth and $(n+1)$th requests from service, respectively, while v_{n+1} denotes the number of requests arriving during the service of the $(n+1)$th request. Then we have

$$q_{n+1} = \begin{cases} q_n - 1 + v_{n+1} & q_n > 0 \\ v_{n+1} & q_n = 0. \end{cases} \tag{15}$$

In order to derive equilibrium balance behavior of the system, we need to calculate the one-step transition probabilities with the embedded Markov chain, which are defined as

$$p_{ij} \triangleq P[q_{n+1} = j | q_n = i]. \tag{16}$$

Since we select the departure instants as the Markov points, it's clear that $q_{n+1} < q_n - 1$ is impossible; on the other hand, $q_{n+1} \geq q_n - 1$ is possible for all values due to the arrivals v_{n+1}. We also have to consider two cases with regard to q_n. Whether the nth request leaves behind an empty system (i.e., $q_n = 0$) or not leads to a slight difference in v_{n+1}. Therefore, we calculate p_{ij} as follows,

$$p_{ij} = \begin{cases} 0 & j < i - 1 \\ P(\widetilde{v} = j - i) & j \geq i - 1 \& i = 0 \\ P(\widetilde{v} = j - i + 1) & j \geq i - 1 \& i > 0, \end{cases} \tag{17}$$

where $\widetilde{v} = \lim_{n\to\infty} v_{n+1}$ denotes the limiting arrivals during service time for a request. Actually it should be v_{n+1} instead of $\widetilde{v}$. However, it's easy to find v_{n+1} depends only on the duration of x_{n+1} and not on n at all.

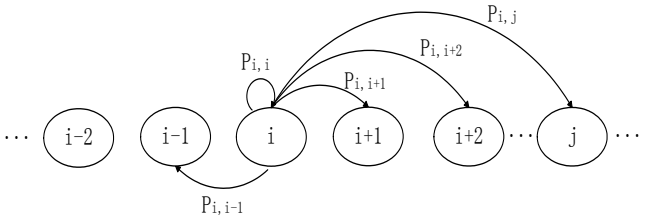


Fig. 3: State-transition-probability diagram for the M/Geo/1 embedded Markov chain.

The state-transition-probability diagram is shown in Figure 3, where states are numbered as the current number of requests in the system. For clarity, some states and transitions are omitted and state i is highlighted for illustration. Please note that though this state-transition-probability diagram is similar to that of an M/G/1 queue, state transition only happens at discrete time point in our M/Geo/1 queue due to that the service time follows a geometric distribution.

To completely specify the transition matrix $\mathbf{P}$, $P[\widetilde{v} = k]$ is calculated as follows [21],

$$P[\widetilde{v} = k] = \sum_{i=1}^{N} \frac{(\lambda i T)^k}{k!} e^{-\lambda i T} Pr[Y = i]. \tag{18}$$

After finding the transition probability matrix $\mathbf{P}$, the stationary probabilities may be obtained from the equation $\pi = \pi\mathbf{P}$, augmented by the normalization equation

$$\sum_{i=0}^{\infty} \pi_i = 1, \tag{19}$$

where π_k is the limiting probability that a departing request will leave behind k requests. Once we have obtained the stationary probabilities, we are able to derive the equilibrium system performance metrics.

Besides, in order to keep the system in a stable state, we have to make sure that

$$\rho = \lambda \overline{t} < 1, \tag{20}$$

which is always true due to that we choose VM instances which can sustain the arrival rate, as described in equations (4) and (5).

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TPDS.2014.2314672, IEEE Transactions on Parallel and Distributed Systems

4.2 System analysis of the M/Geo/1 queue

Here we use the Z-transform to derive a closed-form expression of mean packet response time. Consider a function of discrete time f_k, which takes on nonzero values only for the nonnegative integers, that is, for $k = 0, 1, 2, ...$ (i.e., $f_k = 0$ for $k < 0$). The Z-transform of f_k is defined as follows,

$$F(z) \triangleq \sum_{k=0}^{\infty} f_k z^k. \tag{21}$$

Firstly, we introduce a discrete step function Δ_k, which takes on values 1 only for positive integers,

$$\Delta_k = \begin{cases} 1 & k = 1, 2, ... \\ \\ 0 & k \leq 0. \end{cases} \tag{22}$$

Equation (15) now can be rewrote as

$$q_{n+1} = q_n - \Delta_{q_n} + v_{n+1}. \tag{23}$$

Then by first squaring equation (23) and taking expectations on both sides [21], we have an intermediate result for $\overline{q}$,

$$E[\widetilde{q}] = \rho + \frac{E[\widetilde{v}^2] - E[\widetilde{v}]}{2(1-\rho)}. \tag{24}$$

The unknown part of equation (24) is $E[\widetilde{v}^2]$ and $E[\widetilde{v}]$, which are the first and second moments of $\widetilde{v}$.

Due to that various derivatives of Z transforms (probability generating function) evaluated on $z = 1$ give the various moments of the random variable under consideration, we have to calculate $V(z)$, which is denoted as Z-transform for distribution of arrivals during a service time. Based on the probability distribution function described in equation (18), we have

$$\begin{aligned} V(z) &= \sum_{k=0}^{\infty} P[\widetilde{v} = k] z^k \\ &= \sum_{k=0}^{\infty} \left(\sum_{i=1}^{N} \frac{(\lambda i T)^k}{k!} e^{-\lambda i T} Pr[Y = i] \right) z^k \\ &= \sum_{i=1}^{N} \sum_{k=0}^{\infty} \left(\frac{(\lambda i T)^k}{k!} e^{-\lambda i T} Pr[Y = i] \right) z^k \\ &= \sum_{i=1}^{N} Pr[Y = i] \left(\sum_{k=0}^{\infty} \frac{(\lambda i T)^k}{k!} z^k \right) e^{-\lambda i T} \\ &= \sum_{i=1}^{N} Pr[Y = i] e^{\left(\lambda z T - \lambda T \right) i}. \end{aligned} \tag{25}$$

Let $Y(z')$ denote Z-transform for distribution of random variable Y, which is defined as follows,

$$Y(z') = \sum_{i=0}^{\infty} Pr[Y = i](z')^i. \tag{26}$$

We note the last two equations are of the same form, with the variable z' replaced by $e^{(\lambda z T - \lambda T)}$. So we derive the result that

$$V(z) = Y(e^{\lambda z T - \lambda T}), \tag{27}$$

which is similar to $V(z) = B^*(\lambda - \lambda z)$ for continuous service time, i.e., the M/G/1 model. Forming the second derivative of equation (27) and take value on $z = 1$, we have

$$\begin{aligned} \overline{v^2} - \overline{v} &= V^{(2)}|_{z=1} \\ &= \lambda^2 T^2 E[Y^2]. \end{aligned} \tag{28}$$

The last equation is just the unknown part in equation (24). By Substituting it into equation (24), we finally get the mean number of requests in system (in waiting or in execution) is

$$\begin{aligned} \overline{q} &= E[\widetilde{q}] \\ &= \rho + \frac{\lambda^2 T^2 E[Y^2]}{2(1-\rho)}. \end{aligned} \tag{29}$$

For any Poisson arrivals system, the distribution of number of requests in system at the time of a service departure is identical to the distribution of number of requests at an arbitrary time [21]; thus the mean number of tasks at an arbitrary time is equal to $\overline{q}$.

Applying Little's law, the mean response time (waiting plus execution) is

$$\begin{aligned} \overline{r} &= \overline{q}/\lambda \\ &= \overline{t} + \frac{\lambda T^2 E[Y^2]}{2(1-\rho)}. \end{aligned} \tag{30}$$

4.3 System analysis of the M/Geo/m queue

For VM instance with several computing cores $m > 1$, we model it as an M/Geo/m queue. However, key performance metrics for an M/G/m queue cannot be obtained in closed form, which is also true for the M/Geo/m queue. To get a closed-form expression for the mean packet response time, we make some approximations based on deduction in [22], [23].

First, the probability that there is no request waiting in the queue is

$$P_0 = \left[\sum_{k=0}^{m-1} \frac{(m\rho)^k}{k!} + \frac{(m\rho)^m}{m!(1-\rho)} \right]^{-1}, \tag{31}$$

here $\rho = \lambda \overline{t}/m$. The probability that a request has to wait (approximately) is given as

$$P_Q = P_{Queuing} = \frac{(m\rho)^m}{m!(1-\rho)} P_0. \tag{32}$$

The expected number of request waiting in queue (not in service) is given by

$$N_Q = \frac{\overline{t^2}\rho}{2\overline{t}^2(1-\rho)} P_Q. \tag{33}$$

Applying Little's Theorem, we get the mean waiting time as

$$\overline{w} = \frac{N_Q}{\lambda}, \tag{34}$$

from which the mean response time (approximately) is given as

$$\overline{r} = \overline{t} + \frac{N_Q}{\lambda}. \tag{35}$$

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TPDS.2014.2314672, IEEE Transactions on Parallel and Distributed Systems

Please note that though these performance metrics are approximately given, they are exactly true for $m = 1$. In other words, equation (30) is a special case of equation (35).

5 Performance Evaluation

In this section, we first validate our analytical model and investigate basic parameter settings of the proposed cloud firewall. Then the tradeoff among resources provisioning cost, QoS requirement and packet arrival rate is thoroughly studied.

5.1 Analytical model validation

In the following experiments, we take VM instances offered by Amazon EC2 for calculation. Two pricing options for VM instances are offered by Amazon EC2: on-demand and reservation [24]. To capture the dynamic resources provisioning and allow for request of VM instances at any time, we employ the on demand pricing option.

We first have to give a sensible estimation of service rate of each VM instance, which is determined by N, T, m and p according to equation (13). Here N is set 1000 and $p = 1/N$. As cloud firewall should be transparent to users, we assume response time through each cloud firewall is in granularity of millisecond (which is reasonable according to analysis results in [5]). As a result, rule matching time T should be in granularity of microsecond. In this paper, T is set as $27\mu s$. Service rate of VM instances are listed in Table I.

TABLE 1: Service rate of VM instances

Instance Type	Instance Configuration	Instance Price	Service Rate
Small	1 ECU, 1.7GB RAM, 160GB disk	$0.080	58
Medium	2 ECU, 3.75GB RAM, 410GB disk	$0.160	117
Large	4 ECU, 7.5GB RAM, 850GB disk	$0.320	234
Extra Large	8 ECU, 15GB RAM, 1690GB disk	$0.640	468

As discussed previously, we use average packet response time through the cloud firewall as a key metric for our performance evaluation. First, we are interested in the comparison between our M/Geo/1 model and the general M/M/1 model. The experimental results are shown in Figure 4. It is easy to find that our M/Geo/1 model outperforms the M/M/1 model as it matches the simulation results much better. In other words, it's more reasonable to assume the firewall service rate follows a discrete geometric distribution than a continuous exponential distribution. Here λ is set at most 50 packets per second (pps) as service rate of the small VM instance is 58.

For an M/Geo/m queue, its closed-form response time is approximately given, which is a decision variable to resources provisioning cost optimization. Therefore, we have to check whether this approximation is reasonable. We simulate the relationship between average request response time and attack rate, and compare simulation results with analytical results derived for M/Geo/m. The simulation is conducted for an extra large VM instance, i.e., $m = 8$.

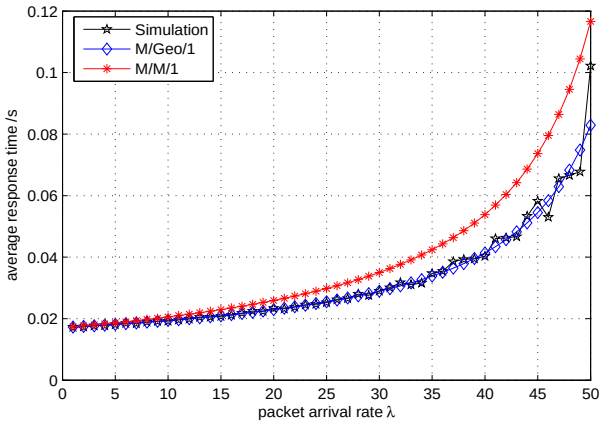


Fig. 4: Comparison of M/Geo/1 and M/M/1.

Both analytical and experimental results are illustrated in Figure 5. Our M/Geo/m model is confirmed by the simulation results that the mean experimental response time fluctuates around the expectation obtained from equation (35). As can be seen from Figure 5, the average response time smoothly increase when attack rate grows. However, as offered load $\rho \to 1$, the response time increase sharply. The reason for this sharp increase is that the arrival packets to the extra large VM instance reaches its maximum processing rate of $1/\bar{t}$, which is approximately 468 packet per second (pps).

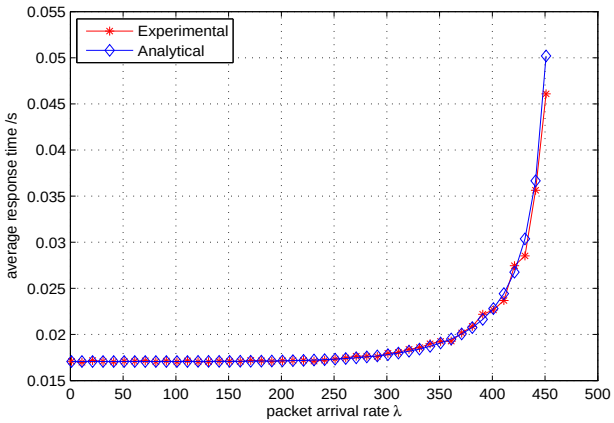


Fig. 5: Validation of approximate closed-from mean packet response time for M/Geo/m.

The results also confirm our earlier claim that a centralized firewall for a whole cloud platform is impractical. It is rather easy for packet arrival rate in attack period to exceed service rate of VM instances. A much

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TPDS.2014.2314672, IEEE Transactions on Parallel and Distributed Systems

larger N (e.g., 10000 rules) makes it even worse. Based on these results, we claim that the proposed decentralized cloud firewall is necessary and feasible.

5.2 Firewall parameter settings

Cloud customers usually have personalized requirements for firewall, which is mainly due to that different applications are of varying degrees vulnerable to various attacks. For example, an e-business web site is highly likely more vulnerable to phishing attack compared to a news site due to that cloud customers earn much more money from the former. Therefore, rule set in cloud firewalls differs for cloud customers. In this section, we aim to find the relationship between N, p and mean packet response time.

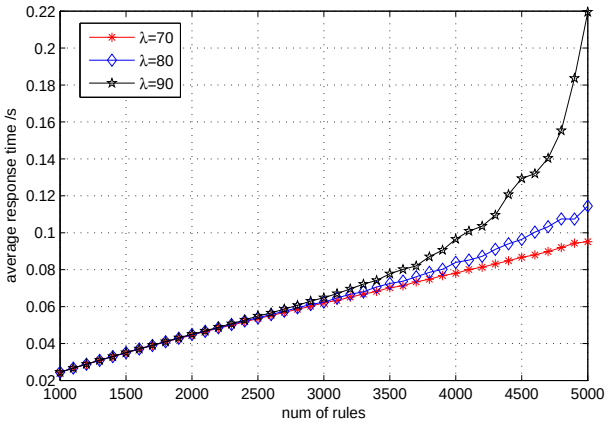


Fig. 6: The relationship between number of rules and mean packet response time.

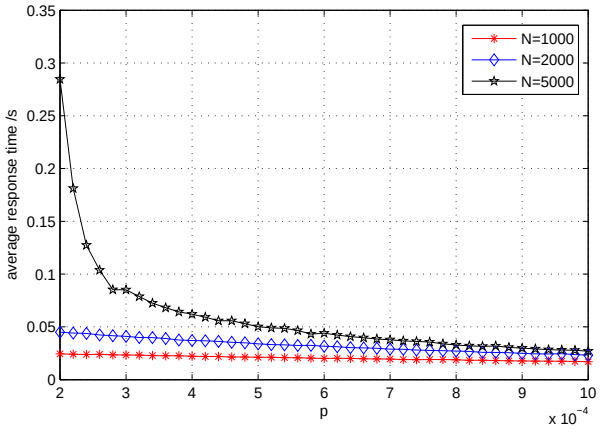


Fig. 7: The relationship between rule matching probability and mean packet response time.

In Figure 6, we show the relationship between number of rules N and average response time of a cloud firewall. Here $p = 1/5000$ and λ is set at most 90 pps due to that service capacity of the extra large VM instance is now approximately 93 pps according to equation (13). From our analytical model, it's expected that more rules decrease service capacity and result in more time to process arrival packets of a given attack rate, which is confirmed by the simulation results.

Figure 7 exhibits the impact of rule matching probability p against average response time. λ is set as 90. It's easy to find that a larger matching probability p leads to less response time, which means firewall service providers are encouraged to put rules easier to match on top of the rule list in cloud firewall to satisfy firewall service customers QoS requirement.

5.3 Cost and performance tradeoff

In this part, we try to show how performance metric "QoS requirement ΔT" affects optimal cost according to equation (3), i.e., whether there exists a trade-off. We are also interested in how packet arrival rate can affect this trade-off. As can be seen from equation (3), χ^a and χ^n have an independent yet similar impact on the optimal resources provisioning cost. So we take χ^a for example and set $\chi^n = 200$ pps as a constant. In the following experiments, $N = 1000$, $M = 20$, $p = 1/N$, $T^n = 1$ hour (unit time interval charging VM instances in Amazon EC2), $T^a = 5$ min (here we take average DDoS attack duration for example).

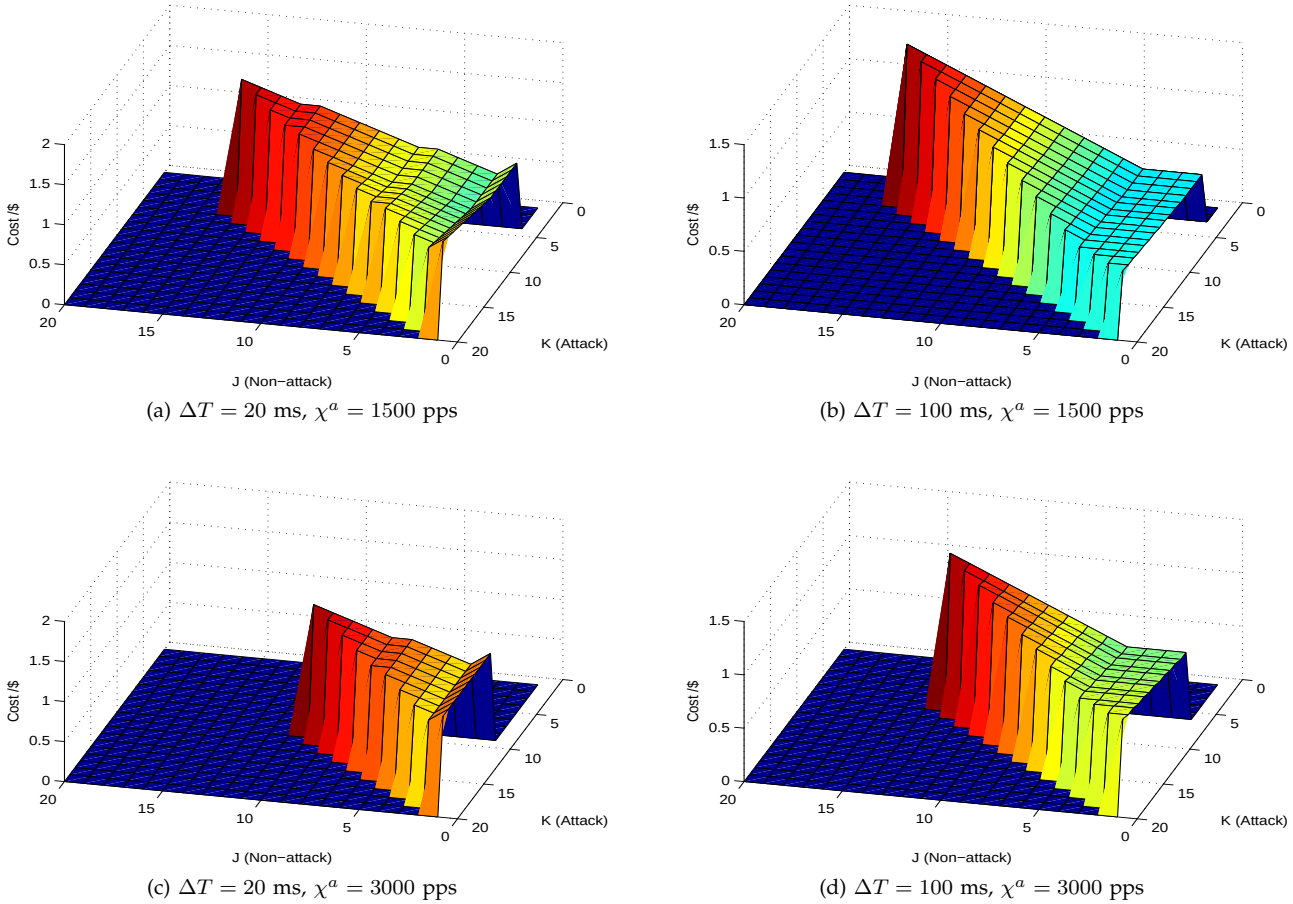
The *input* parameters are χ^a, χ^n and ΔT, and the *output* are the optimal resources provisioning cost, the corresponding number of clusters J and K for non-attack and attack period respectively. Besides, J and K are set less than 20 according to that at most 20 concurrent VM instances are available to an account in Amazon EC2 [4].

The results are shown in Figure 8 (a)-(d). The results indicate that: 1) for a given QoS requirement and packet arrival rate, there exists an optimal resources provisioning cost; 2) for a given packet arrival rate, the longer QoS requirement is, the cheaper the optimal resources provisioning cost (compare Figure 8 (a) and (b)); 3) for a given QoS requirement ΔT, the smaller packet arrival rate, the cheaper the optimal resources provisioning cost (compare Figure 8 (a) and (c)). The results also illustrate that there are generally several VM configurations (J and K) that can achieve an optimal resources provisioning cost.

In Figure 9, we compare the optimal resources provisioning cost for attack rate 1500 pps, 3000 pps and 4500 pps. QoS requirement ΔT is set to be 500 ms, 100 ms and 20 ms. It's easy to find that there is a trade-off among the three parameters, just as we have concluded from Figure 8. Something new is also illustrated: 1) the optimal resources provisioning cost keeps a constant when ΔT grows from 100 ms to 500 ms, which is due to that ΔT now is no longer a constraint in equation (4) and (5); 2) when $\Delta T = 20$ ms, the optimal resources provisioning cost is 0, which means the constraint is so strict that there is no solution to equation (3).

For readers' reference, we list the numerical results in Table 2. By comparing the optimal cost with price of

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TPDS.2014.2314672, IEEE Transactions on Parallel and Distributed Systems



(a) $\Delta T = 20$ ms, $\chi^a = 1500$ pps

(b) $\Delta T = 100$ ms, $\chi^a = 1500$ pps

(c) $\Delta T = 20$ ms, $\chi^a = 3000$ pps

(d) $\Delta T = 100$ ms, $\chi^a = 3000$ pps

Fig. 8: The impact of QoS requirement ΔT and packet arrival rate in attack period against optimal resources provisioning cost.

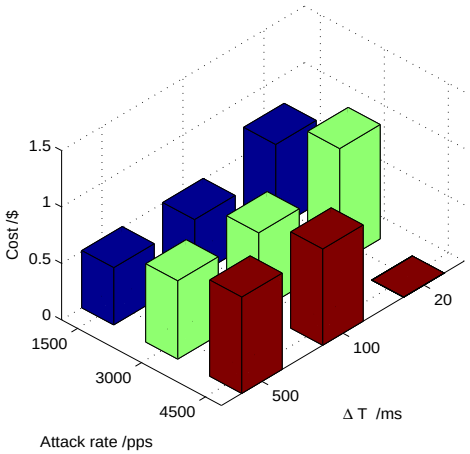


Fig. 9: Trade-off among resource provisioning cost, QoS requirement and attack rate.

TABLE 2: Optimal resources provisioning cost and corresponding VM instance configuration

QoS requirement ΔT Attack rate λ^a	Optimal cost θ	VM configuration	
		non-attack	attack
20 ms 1500 pps	\$0.7467	1 medium 1 large	5 extra large
100 ms 1500 pps	\$0.5067	1 large	1 large 5 extra large
20 ms 3000 pps	\$1.0133	1 medium 1 large	10 extra large
100 ms 3000 pps	\$0.6933	1 large	2 large 6 extra large
20 ms 4500 pps	\$0.0000	*infeasible*	*infeasible*
100 ms 4500 pps	\$0.8533	1 large	10 extra large

Amazon VM instances (refer to Table 1), we claim that its practical and feasible to set up an cloud firewall for individual cloud customers.

Table 2 also offers firewall service providers a tool for pricing. Assume that parameters N, p, χ^a and χ^b are given. Let $\theta(\Delta T)$ denote an optimal resources provisioning cost function with respect to a given ΔT. θ_{min} denotes the cheapest optimal resources provisioning cost when ΔT increases. We propose that θ_{min} can be published as price for the cloud firewall under the given parameters set. Of course, QoS requirement interval sustained by this price should also be published. For example, firewall

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TPDS.2014.2314672, IEEE Transactions on Parallel and Distributed Systems

service providers just offer $\$0.5467/hr$ as price for cloud firewall with packet arrival rate $\chi^n = 200$ pps, $\chi^a = 1500$ pps, $N = 1000$ and $p = 1/N$. The corresponding QoS requirement interval is $[49ms, \infty]$.

6 Further discussion

To the best of our knowledge, this paper is an early work to discuss resource provisioning cost optimization in the context of cloud firewall. As a new research field, there are many other mathematical tools available to address this optimization problem, such as game theory [25], integer linear programming [24] and stochastic programming [26]. Due to the limitations of knowledge, time and space, we have only employed queuing theory in this paper.

Our analytical model assumes that packet arrivals to the cloud server follow Poisson distribution, and the service times follow Geometric distribution. For certain types of network traffic, assuming Poisson arrivals is feasible [27]. However, for general traffic like Ethernet, their arrivals do not always follow a Poisson distribution but are rather bursty or heavy-tailed [28], [29]. Also, the assumption that all rules share the same matching probability is hard to meet in reality. Considering different matching probability and non-Poisson distribution will make a closed-form analytical solution intractable. To address these limitations, DES (Discrete Event Simulation) can be employed [30].

7 Conclusion and future work

In this paper, we point out that it's impractical to establish a firewall for a whole cloud data center. However, cloud service providers possess a potential to provide cloud firewalls for individual cloud customers. In view of this challenge, we propose a decentralized cloud firewall framework, where several firewall running in parallel to guarantee QoS requirement. As resources are dynamically allocated in cloud firewall, we investigate how to optimize the resources provisioning cost.

We establish novel queuing theory based model for performance analysis of the proposed cloud firewall, where firewall service times are modeled to follow geometric distribution. Extensive simulations confirm that M/Geo/1 reflects the cloud firewall real system better than traditional M/M/1. Besides, it is feasible to set up firewall for individual cloud hosted services with an affordable cost to cloud customers.

As future work, we firstly plan to improve the decentralized framework to capture more personalized details in application level. Secondly, we would like to propose a pricing model for the cloud firewall, which helps to achieve a financial balance between provider and customer. Real cloud environment experiments for the proposed cloud firewall are also expected in the near future.

References

[1] M. Armbrust, A. Fox, R. Griffith, A. D. Joseph, R. Katz, A. Konwinski, G. Lee, D. Patterson, A. Rabkin, I. Stoica *et al.*, "A view of cloud computing," *Communications of the ACM*, vol. 53, no. 4, pp. 50–58, 2010.

[2] Z. Xiao and Y. Xiao, "Security and privacy in cloud computing," *IEEE Communications Surveys and Tutorials*, no. in press, 2013.

[3] C. Hoff, "Cloud computing security: From ddos attack (distributed denial of servie) to edos (economic denial of sustainability)," in *http://www.rationalsurvivability.com/blog/?p=66.*, 2008.

[4] T. Ristenpart, E. Tromer, H. Shacham, and S. Savage, "Hey, you, get off of my cloud: exploring information leakage in third-party compute clouds," in *Proceedings of the 16th ACM conference on Computer and communications security*. ACM, 2009, pp. 199–212.

[5] K. Salah, K. Elbadawi, and R. Boutaba, "Performance modeling and analysis of network firewalls," *Network and Service Management, IEEE Transactions on*, vol. 9, no. 1, pp. 12–21, 2012.

[6] D. Rovniagin and A. Wool, "The geometric efficient matching algorithm for firewalls," *Dependable and Secure Computing, IEEE Transactions on*, vol. 8, no. 1, pp. 147–159, 2011.

[7] A. X. Liu and F. Chen, "Privacy preserving collaborative enforcement of firewall policies in virtual private networks," *Parallel and Distributed Systems, IEEE Transactions on*, vol. 22, no. 5, pp. 887–895, 2011.

[8] A. X. Liu, "Firewall policy change-impact analysis," *ACM Transactions on Internet Technology (TOIT)*, vol. 11, no. 4, p. 15, 2012.

[9] A. R. Khakpour and A. X. Liu, "First step toward cloud-based firewalling," in *Reliable Distributed Systems (SRDS), 2012 IEEE 31st Symposium on*. IEEE, 2012, pp. 41–50.

[10] S. Yu, R. Doss, W. Zhou, and S. Guo, "A general cloud firewall framework with dynamic resource allocation," in *IEEE International Conference on Communications*, no. In press, 2013.

[11] S. Yu, Y. Tian, S. Guo, and D. Wu, "Can we beat ddos attacks in clouds?" *Parallel and Distributed Systems, IEEE Transactions on*, no. in press, 2013.

[12] D. Moore, C. Shannon, D. J. Brown, G. M. Voelker, and S. Savage, "Inferring internet denial-of-service activity," *ACM Transactions on Computer Systems (TOCS)*, vol. 24, no. 2, pp. 115–139, 2006.

[13] U. Sharma, P. Shenoy, S. Sahu, and A. Shaikh, "A cost-aware elasticity provisioning system for the cloud," in *Distributed Computing Systems (ICDCS), 2011 31st International Conference on*. IEEE, 2011, pp. 559–570.

[14] R. Wartel, T. Cass, B. Moreira, E. Roche, M. Guijarro, S. Goasguen, and U. Schwickerath, "Image distribution mechanisms in large scale cloud providers," in *Cloud Computing Technology and Science (CloudCom), 2010 IEEE Second International Conference on*. IEEE, 2010, pp. 112–117.

[15] J. Zhu, Z. Jiang, and Z. Xiao, "Twinkle: A fast resource provisioning mechanism for internet services," in *INFOCOM, 2011 Proceedings IEEE*. IEEE, 2011, pp. 802–810.

[16] J. Cao, K. Hwang, K. Li, and A. Zomaya, "Optimal multiserver configuration for profit maximization in cloud computing," *Parallel and Distributed Systems, IEEE Transactions on*, no. in press, 2012.

[17] H. Khazaei, J. Misic, and V. B. Misic, "Performance analysis of cloud computing centers using m/g/m/m+ r queuing systems," *Parallel and Distributed Systems, IEEE Transactions on*, vol. 23, no. 5, pp. 936–943, 2012.

[18] H. Khazaei, J. Misic, V. Misic, and S. Rashwand, "Analysis of a pool management scheme for cloud computing centers," *Parallel and Distributed Systems, IEEE Transactions on*, no. in press, 2012.

[19] H. Khazaei *et al.*, "Performance of cloud centers with high degree of virtualization under batch task arrivals," *Parallel and Distributed Systems, IEEE Transactions on*, no. in press, 2013.

[20] D. P. Heyman and M. J. Sobel, *Stochastic Models in Operations Research: Stochastic Optimizations*. DoverPublications. com, 2003, vol. 2.

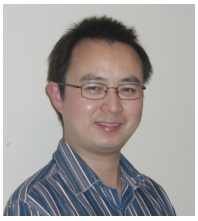
[21] L. Kleinrock, *Theory, volume 1, Queueing systems*. Wiley-interscience, 1975.

[22] Z. Zeng and B. Veeravalli, "On the design of distributed object placement and load balancing strategies in large-scale networked multimedia storage systems," *Knowledge and Data Engineering, IEEE Transactions on*, vol. 20, no. 3, pp. 369–382, 2008.

[23] P. Hokstad, "Approximations for the m/g/m queue," *Operations Research*, vol. 26, no. 3, pp. 510–523, 1978.

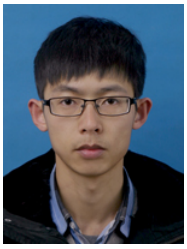
This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TPDS.2014.2314672, IEEE Transactions on Parallel and Distributed Systems

[24] R.-H. Hwang, D.-J. Zhang-Jian, C.-N. Lee, and Y.-R. Chen, "Cost optimization of elasticity cloud resource subscription policy," *IEEE Transactions on Services Computing*, p. 1, 2013.
[25] H. Wang, F. Wang, J. Liu, and J. Groen, "Measurement and utilization of customer-provided resources for cloud computing," in *INFOCOM, 2012 Proceedings IEEE*. IEEE, 2012, pp. 442–450.
[26] S. Chaisiri, B.-S. Lee, and D. Niyato, "Optimization of resource provisioning cost in cloud computing," *Services Computing, IEEE Transactions on*, vol. 5, no. 2, pp. 164–177, 2012.
[27] M. J. Karam and F. A. Tobagi, "Analysis of delay and delay jitter of voice traffic in the internet," *Computer Networks*, vol. 40, no. 6, pp. 711–726, 2002.
[28] V. Paxson and S. Floyd, "Wide area traffic: the failure of poisson modeling," *IEEE/ACM Transactions on Networking (ToN)*, vol. 3, no. 3, pp. 226–244, 1995.
[29] K. Jagannathan, M. Markakis, E. Modiano, and J. N. Tsitsiklis, "Queue-length asymptotics for generalized max-weight scheduling in the presence of heavy-tailed traffic," *IEEE/ACM Transactions on Networking (TON)*, vol. 20, no. 4, pp. 1096–1111, 2012.
[30] R. Jain, "The art of computer system performance analysis: techniques for experimental design, measurement, simulation and modeling," *New York: John Willey*, 1991.

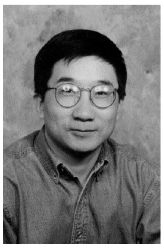


Shui Yu received the B.Eng. and M.Eng. degrees from University of Electronic Science and Technology of China, Chengdu, P. R. China, in 1993 and 1999, respectively, and the Ph.D. degree from Deakin University, Victoria, Australia, in 2004. He is currently a Senior Lecturer with the School of Information Technology, Deakin University, Victoria, Australia. He has published nearly 100 peer review papers, including top journals and top conferences, such as IEEE TPDS, IEEE TIFS, IEEE TFS, IEEE TMC, and IEEE INFOCOM. His research interests include networking theory, network security, and mathematical modeling.

Dr. Yu actively servers his research communities in various roles, which include the editorial boards of IEEE Transactions on Parallel and Distributed Systems, and three other International journals, IEEE INFOCOM TPC members, symposium co-chairs of IEEE ICC 2014, IEEE ICNC 2013 and 2104, and many different roles of international conference organizing committees. He is a senior member of IEEE, and a member of AAAS.



Meng Liu is currently working towards the PhD degree at the Department of Computer Science and Technology, Nanjing University, China. He has received his Master's and Bachelor's degree in Software Engineering from Xidian University and Dep. of Computer Science and Technology from Nanjing University of China, respectively. His research interests include cloud computing, privacy and security.



Zhensheng Zhang received his Ph.D. in electrical engineering from the University of California, Los Angeles in 1989. Dr. Zhang has over twenty years experience in design and analysis of network architecture, protocols and control algorithms, with very strong backgrounds in performance analysis, modeling and simulation of the communication networks. He is currently with Argon ST (formerly SDRC), Principal Scientist, Networking Research, serving as Principal Investigator for several DOD projects. Before joining SDRC, he visited Microsoft Research in the summer of 2002 and worked at Sorrento Networks, Department of System Architecture, responsible for designing the next-generation optical metro networks using the GMPLS control framework. Prior to Sorrento Networks he was with Bell Laboratories, Lucent Technologies, focusing on research and development in wireless networks. Dr. Zhang severed as Editor of IEEE Transaction on Wireless Communications from 2002 to 2006. He served the General Chair of Broadband Wireless Networking Symposium, October 2004. He has served as Guest Editor for the IEEE JSAC special issue on Overlay Networks, 2003 and the Journal of Wireless Networks issue on multimedia wireless networks, August 1996. Dr. Zhang served as Member at Large of the IEEE San Diego section 2004 and as Chair of IEEE Communication Society, San Diego section, 2004-2009. His research interests include wireless ad hoc networks, wireless sensor networks. He has given many invited talks and tutorials on wireless ad hoc networks at various conferences. He has published more than 100 papers in ACM/IEEE Transactions on Networking, IEEE JSAC, IEEE Transactions on Communications, and key ACM/IEEE conferences.

Wanchun Dou received his PhD degree in Mechanical and Electronic Engineering from Nanjing University of Science and Technolo-gy, China, in 2001. From Apr. 2001 to Dec. 2002, he did his postdoctoral research in the Department of Computer Science and Technology, Nanjing University, China. Now, he is a full professor of the State Key Laboratory for Novel Software Technology, Nanjing University, China. From Apr. 2005 to Jun. 2005 and from Nov. 2008 to Feb. 2009, he respectively visited the Department of Computer Science and Engineering, Hong Kong University of Science and Technology, as a visiting scholar. Up to now, he has chaired three NSFC projects and published more than 60 research papers in international journals and international conferences. His research interests include workflow, cloud computing and service computing.