Journal of Southeast University (English Edition) Vol. 34, No. 1, pp. 21 - 27 Mar. 2018 ISSN 1003—7985

Modeling and analysis of cloud computing system survivability based on Bio-PEPA

Zhao Guosheng[1] Ren Mengqi[1] Wang Jian[2] Liao Yiwei[1]

([1]College of Computer Science and Information Engineering, Harbin Normal University, Harbin 150025, China)

([2]School of Computer Science and Technology, Harbin University of Science and Technology, Harbin 150080, China)

Abstract: For the cloud computing system, combined with the memory function and incomplete matching of the biological immune system, a formal modeling and analysis method of the cloud computing system survivability is proposed by analyzing the survival situation of critical cloud services. First, on the basis of the SAIR (susceptible, active, infected, recovered) model, the SEIRS (susceptible, exposed, infected, recovered, susceptible) model and the vulnerability diffusion model of the distributed virtual system, the evolution state of the virus is divided into six types, and then the diffusion rules of the virus in the service domain of the cloud computing system and the propagation rules between service domains are analyzed. Finally, on the basis of Bio-PEPA (biological-performance evaluation process algebra), the formalized modeling of the survivability evolution of critical cloud services is made, and the SLIRAS (susceptible, latent, infected, recovered, antidotal, susceptible) model is obtained. Based on the stochastic simulation and the ODEs (ordinary differential equations) simulation of the Bio-PEPA model, the sensitivity parameters of the model are analyzed from three aspects, namely, the virus propagation speed of inter-domain, recovery ability and memory ability. The results show that the proposed model has high approximate fitting degree to the actual cloud computing system, and it can well reflect the survivable change of the system.

Key words: cloud computing system; Bio-PEPA (biological-performance evaluation process algebra); survivability; stochastic simulation

DOI: 10.3969/j.issn.1003 - 7985.2018.01.004

Survivability refers to the capability of a system to fulfill its critical services in a timely manner when the system is subjected to external attacks, such as viruses or internal failures[1]. Cloud computing is a new type of distributed network service computing model[2]. A cloud computing system can provide a variety of trusted critical services for users. Therefore, it is extremely important to ensure that the critical services in the cloud computing system one executed continuously when the system is subjected to external attacks or internal failures. Recently, domestic and overseas scholars have conducted large-scale research on survivability, and the main research areas include the formal modeling, survivability enhancement technology, survivability analysis, test and evaluation, etc.

Chang et al.[3] proposed a solution for the virtual system survivability model, in which the continuous time Markov chain is used to analyze the service survivability after its failure. Using the proposed solution, the viability of the system can be quantitatively assessed. Jin et al.[4] considered the survivability of each node, the deletion of links and nodes determined by the survivability of the nodes, a survivable topology evolution model based on the wireless sensor network was proposed. Through this model, the survivability of the node can be analyzed. Zhou et al.[5] proposed a method to predict survivability according to a large amount of log audit data of the system, and used a quantitative method to analyze the survival ability of the system over a certain period of time. Alobaidi et al.[6] also proposed a quantitative analysis method based on the smart grid, which described the changes of the system state and the decline conditions in its service performance, in order to maintain the highest system survivability during the recovery process.

The above studies analyzed the survivability of the system or critical service by models which can quantitatively analyze the survivability, and then clearly depict the changing process of the survivability. However, their studies did not consider the dynamic impact conditions of the system survivability, such as external attacks or internal failures in the actual system operation.

In summary, based on the features of the biological immune system[7-8], on the basis of the SAIR model[9], the SEIRS model[10] and the vulnerability propagation model of distributed virtualization[11], the survivability evolution model of critical cloud services in the cloud computing system is obtained.

1 Bio-PEPA Syntax

Bio-PEPA (biological-performance evaluation process algebra) combines some features of biological network and is well suited to describing the spread of the virus

Received 2017-09-16, **Revised** 2017-12-19.

Biography: Zhao Guosheng (1977—), male, doctor, professor, zgswj@163.com.

Foundation items: The National Natural Science Foundation of China (No. 61202458, 61403109), the Natural Science Foundation of Heilongjiang Province of China (No. F2017021), Harbin Science and Technology Innovation Research Funds (No. 2016RAQXJ036).

Citation: Zhao Guosheng, Ren Mengqi, Wang Jian, et al. Modeling and analysis of cloud computing system survivability based on Bio-PEPA [J]. Journal of Southeast University (English Edition), 2018, 34(1): 21 - 27. DOI: 10.3969/j.issn.1003 - 7985.2018.01.004.

within a cloud computing system and the survivability evolution of critical cloud services. The basic semantic expressions are as follows[12-13]:

$$S::=(\alpha,\gamma)\ op\ S \mid S+S \mid \text{Constant} \mid S@L \quad (1)$$

$$op::=\downarrow \mid \uparrow \mid \odot \quad (2)$$

$$P::=P \underset{G}{\rhd\lhd} P \mid S(x) \quad (3)$$

The meanings of the basic expressions are shown in Tab. 1.

Tab. 1 The meanings of Bio-PEPA basic expressions

Character	Representative meaning
S	Species component
P	Model component
α	Action
γ	Transition rate
+	Choice between species S
Constant	Constant
$S@L$	Component L in position S
op	The role of S in the reaction
$\downarrow$	S as a reactant
$\uparrow$	S as a generator
$\odot$	General modification
$\rhd\lhd$	Cooperative operator
G	A set of actions that must be synchronized during cooperation
x	The number of initial components

Approximate steady-state probability is the ratio of the number of units to that of components, after the model reaches a certain stable state. Hypothesis $X=\{x_1, x_2, \ldots, x_n\}$, x_i is the number of components in the system, N_{total} is the number of components. For any positive integer $i(0<i<n)$, the approximate steady-state probability of type i component is

$$\pi_i=\frac{x_i}{N_{\text{total}}} \quad (4)$$

Kinetic law vector V_{KL} is composed of the universal set of reaction rate f_k, and

$$f_k=k^*\prod_i N_{\text{reaction}_i} \quad (5)$$

where N_{reaction_i} represents the number of reactants. As previously mentioned, X is the number of components and it satisfies the ordinary differential equation:

$$\left.\frac{\mathrm{d}X}{\mathrm{d}t}\right|_{X=X_0}=EV_{KL} \quad (6)$$

Due to the length of the paper, we will not elaborate more than is needed. Refs. [12-13] introduced the solution and derivation process of Bio-PEPA in detail. Ref. [14] introduced Bio-PEPA Eclipse plug-in syntax and modeling terminology.

2 Evolution Model of Survivability Situation

The survivability of critical cloud services in the cloud computing system is inevitably affected by the propagation of the virus. We first classify the survivability of key cloud services, and then study the impact of virus propagation in intra-domain or inter-domain. Finally, we obtain the evolution model of survivability.

2.1 Classification of survivability situation

Referring to the model of SAIR and SEIRS virus propagation, and the diffusion model in the distributed virtual system of vulnerability, the survivability of key cloud services is abstracted into 6 states: The susceptible state S, the latent state L_e, the latent state L_c, the infection state I, the recovery state R and the immune state A. By default, the latent virus is not activated or executed, so it is not infectious and can be obviously distinguished from the virus in the infected state. The specific states are as follows:

1) S represents that the nodes are not infected with the virus, but they have the possibility of infection.

2) L_e represents that the nodes contain the latent virus and have been detected.

3) L_c represents that the nodes contain the latent virus but have not been found.

4) I represents that the nodes are infected with the virus and the virus has performed its part or all of predefined functions.

5) R represents that the nodes are infected, but the virus has been cleared. The nodes may be transformed into S or A.

6) A represents that the nodes are infected and the virus has been cleared, but the nodes have immune function to the virus or similar virus.

The state set of all nodes is $W=\{S, L_e, L_c, I, R, A\}$, and the virus in the latent state and the active state forms the state set of virus as $L_{\text{set}}=\{L_e, L_c, I\}$. In order to further facilitate the description, we define the nodes in the state $Z\in W$ as component Z.

2.2 Intra-domain propagation rules

Compared with inter-domains, viruses are more likely to be propagated in intra-domains, and therefore, we first consider the simplest case, which is the impact of viruses propagation in intra-domains. Intra-domain propagation rules of virus are as follows:

〈Propagation1〉:

1) 〈link1〉$S+I\rightarrow 2I$: The component I spreads the virus to component S through the connection, and the virus is active.

2) 〈link2〉$S+I\rightarrow L_e+I$: The component I spreads the virus to the component S though the connection. Now the virus is in the latent state and has been found.

3) 〈link3〉$S+I\rightarrow L_c+I$: The component I spreads the virus to the component S though the connection. Now the virus is in the latent state and has not been found.

4) 〈activation1〉$L_e\rightarrow I$: The latent virus in the component L_e is activated.

5) 〈activation2〉$L_c\rightarrow I$: The latent virus in the compo-

nent L_c is activated.

6) $\langle \text{recovery1} \rangle L_e \rightarrow R$: The latent virus in the component L_e is cleared.

7) $\langle \text{recovery2} \rangle I \rightarrow R$: The latent virus in the component I is cleared.

8) $\langle \text{memory} \rangle R \rightarrow A$: Component R is transformed into component A by the memory function after virus clearance.

9) $\langle \text{insecure1} \rangle A \rightarrow S$: Component A is transformed into component S.

10) $\langle \text{insecure2} \rangle A \rightarrow L_e$: Component A is transformed into component L_e.

The virus propagation set is Propagation1 = {link1, link2, link3, activation1, activation2, recovery1, recovery2, memory, insecure1, insecure2}. The rate of change for each reaction (propagation rule) is recorded as r_α, where $\alpha \in$ Propagation1. Assuming that the number and type of components are fixed within a certain period of time, recording the total number of components as N, the number of components S, L_e, L_c, I, R, A as n_s, n_e, n_c, n_i, n_r, n_a. The reaction rate f_k of each reaction satisfies

$$f_{\text{link1}} = r_{\text{link1}} n_s n_i,\ f_{\text{link2}} = r_{\text{link2}} n_s n_e,\ f_{\text{link3}} = r_{\text{link3}} n_s n_c$$

$$f_{\text{activation1}} = r_{\text{activation1}} n_e n_i,\ f_{\text{activation2}} = r_{\text{activation2}} n_c n_i$$

$$f_{\text{recovery1}} = r_{\text{recovery1}} n_e n_r,\ f_{\text{recovery2}} = r_{\text{recovery2}} n_i n_r$$

$$f_{\text{memory}} = r_{\text{memory}} n_a n_r,\ f_{\text{insecure1}} = r_{\text{insecure1}} n_s n_a$$

$$f_{\text{insecure2}} = r_{\text{insecure2}} n_e n_a$$

2.3 Inter-domain propagation rules

Assuming that the system is divided into n service domains, let the domain set be $K = \{\text{location}_1, \text{location}_2, \cdots, \text{location}_n\}$, $|K| = n$, location_i, $\text{location}_j \in K$, $\text{location}_i \neq \text{location}_j$, which represent different service domains. The inter-domain propagation rules of virus are as follows:

$\langle$Propagation2$\rangle$:

11) $\langle \text{link1}_{ij} \rangle S@\text{location}_i + I@\text{location}_j \rightarrow I@\text{location}_i + I@\text{location}_j$: The component I in service domain location_j propagates the virus to the component S in service domain location_i by connecting and the virus is activated.

12) $\langle \text{link2}_{ij} \rangle S@\text{location}_i + I@\text{location}_j \rightarrow L_e@\text{location}_i + I@\text{location}_j$: The component I in service domain location_j propagates the virus to the component S in service domain location_i by connecting, and then the virus is inactivated, and it can be recognized by the system.

13) $\langle \text{link3}_{ij} \rangle S@\text{location}_i + I@\text{location}_j \rightarrow L_c@\text{location}_i + I@\text{location}_j$: The component I in service domain location_j propagates the virus to the component S in service domain location_i by connecting, and then the virus is inactivated, but it cannot be recognized by the system.

The rate of virus transition between location_i and location_j is r_{link1_ij}, r_{link2_ij} and r_{link3_ij}. The number of components S in location_i is $n_{S@\text{location}\_i}$. The number of components I in location_j is $n_{I@\text{location}\_j}$. The reaction rate of f_k fulfils the following rules:

$$f_{\text{link1}\_ij} = r_{\text{link1}\_ij} n_{S@\text{location}\_i} n_{I@\text{location}\_j}$$

$$f_{\text{link2}\_ij} = r_{\text{link2}\_ij} n_{S@\text{location}\_i} n_{I@\text{location}\_j}$$

$$f_{\text{link3}\_ij} = r_{\text{link3}\_ij} n_{S@\text{location}\_i} n_{I@\text{location}\_j}$$

Especially, if there is no virus propagation relationship between two service domains, $r_{\text{link1}\_ij} = r_{\text{link2}\_ij} = r_{\text{link3}\_ij} = 0$.

2.4 Formal description of Bio-PEPA model

The impact of virus propagation on the survivability of critical cloud services can be represented by the state transition of cloud service nodes. The SLIRAS model is shown in Fig. 1.

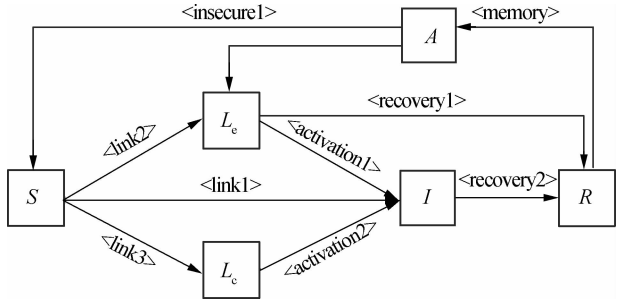

Fig. 1 SLIRAS model

Based on the SLIRAS model, the survivability evolution process of critical cloud service can be formally described by Bio-PEPA as follows:

$$S@\text{location}_i \overset{\text{def}}{=} (\text{link1}_i, 1) \downarrow S@\text{location}_i + (\text{link2}_i, 1) \downarrow S@\text{location}_i + (\text{link3}_i, 1) \downarrow S@\text{location}_i + (\text{insecure1}_i, 1) \uparrow S@\text{location}_i + \sum_j (\text{link1}_{ij}, 1) \downarrow S@\text{location}_i + \sum_j (\text{link2}_{ij}, 1) \downarrow S@\text{location}_i + \sum_j (\text{link3}_{ij}, 1) \downarrow S@\text{location}_i$$

$$I@\text{location}_i \overset{\text{def}}{=} (\text{link1}_i, (1,2)) \odot I@\text{location}_i + (\text{link2}_i, (1,1)) \odot I@\text{location}_i + (\text{link3}_i, (1,1)) \odot I@\text{location}_i + (\text{activation1}_i, 1) \uparrow I@\text{location}_i + (\text{activation2}_i, 1) \uparrow I@\text{location}_i + (\text{recovery2}_i, 1) \downarrow I@\text{location}_i + \sum_j (\text{link1}_{ji}, 1) \uparrow I@\text{location}_i$$

$$L_e@\text{location}_i \overset{\text{def}}{=} (\text{link2}_i, 1) \uparrow L_e@\text{location}_i + (\text{activation1}_i, 1) \downarrow L_e@\text{location}_i + (\text{insecure2}_i, 1) \uparrow S@\text{location}_i + \sum_j (\text{link2}_{ji}, 1) \uparrow L_e@\text{location}_i$$

$$L_c@\text{location}_i \overset{\text{def}}{=} (\text{link3}_i, 1) \uparrow L_c@\text{location}_i + (\text{activation2}_i, 1) \downarrow L_e@\text{location}_i + \sum_j (\text{link3}_{ji}, 1) \uparrow L_c@\text{location}_i$$

$$R@\text{location}_i \overset{\text{def}}{=} (\text{recovery1}_i, 1) \uparrow R@\text{location}_i + (\text{recovery2}_i, 1) \uparrow R@\text{location} + (\text{memory}_i, 1) \downarrow R@\text{location}_i$$

$$A@\text{location}_i \overset{\text{def}}{=} (\text{memory}_i, 1) \uparrow A@\text{location}_i + (\text{insecure1}_i, 1) \downarrow A@\text{location}_i + (\text{insecure2}_i, 1) \downarrow A@\text{location}_i$$

If there is no virus propagation between two service domains in the model, the corresponding transition rate is zero. Based on this model, we can analyze the impact of virus diffusion on the survivability of critical cloud services.

3 Simulation Analysis

Since the model contains many parameters, these parameters have a certain influence on the stability and rationality of the model. Therefore, the section will select some quantitative indicators, and briefly analyze the influence of virus propagation in intra-domains and inter-domains on the viability of critical cloud services, then compare them with the simulation test results.

3.1 Survivability assessment index

The section refers to the existing research results in the field of survivability situation assessment[15-16]. Two indicators are proposed to assess the survivability situation of key cloud services: peak service quality index P_v and steady-state service quality index π_v. Let $p \in W$ represent the type of component, location represents the service domain.

Definition 1 (peak service quality index P_v) The index is the maximum ratio of the number of components in the L_{set} collection to the number of all components in the service domain, at the time $0 < t < +\infty$,

$$P_v = \max\left\{\frac{1}{N_{total}} \sum_{location} \sum_{p \in L_{set}} n_{p@location}(t)\right\} \quad (7)$$

where $n_{p@location}$ is the number of components P in the service domain location; N_{total} is the number of all components in the domain.

Definition 2 (steady-state service quality index π_v) The index is the sum of approximate steady state probabilities of various types of components in L_{set} collection when the number of viruses in the service domain reaches a certain amount, in other words, it will no longer increase or decrease, and achieve a certain steady state:

$$\pi_v = \sum_{location} \sum_{p \in L_{set}} \pi_{p@location} \quad (8)$$

where $\pi_{p@location}$ is the approximate steady-state probability of component P.

P_v mainly measures the maximum range of virus spread in the system, while π_v characterizes the long-term potential impact of viruses on the system survivability[17]. Next, we will analyze the impact on the survivability of key cloud services on the basis of the above two indices from the scope and trend of virus propagation.

3.2 Example selection

In a cloud computing system, survivability may be changed by virus propagation in a service domain or any two service domains. In order to highlight the core of the problem, reduce the complexity caused by the interference of other uncertainties and the differences between different service domains, we only select one of the simplest examples, as shown in Fig. 2.

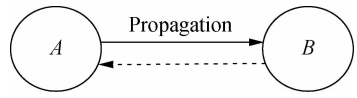

Fig. 2 A simple example of survivability evolution

Let $location_A$ and $location_B$ represent service domains that contain several critical cloud services, respectively. The initial state of $location_A$ contains all kinds of components, and all components in $location_B$ are susceptible to infection at the initial time. Virus diffusion may be present in the internal components of $location_A$; meanwhile, $location_A$ propagates the virus to $location_B$ through the connection. The example also includes survivability situation change caused by virus propagation in intra-domains or inter-domains.

Assuming that there are N nodes, the probability that a node which has been infected by the virus propagates the virus to another node is $1/N$, that is $r_{link1} = 1/N$. At the same time, due to the existence of latent viruses in node L_e and L_c, the probability of the two nodes being infected by the virus will be very high, that is $r_{link2} = r_{link3} = 10 \times 1/N$. The virus propagation probability of inter-domains may be smaller than that in intra-domains. We assume that the virus propagation rate of inter-domains is 1/10 that of intra-domains. In order to reduce the contrast between different service domains, highlight the difference between intra-domain diffusion and inter-domain diffusion, assume that the rate of virus propagation in $location_A$ and $location_B$ are the same. The parameters are shown in Tab. 2.

Tab. 2 The values of each parameter in the model

Parameter	Value	Parameter	Value
$r_{activation1}$	0.10	$r_{insecure1}$	0.10
$r_{activation2}$	0.50	$r_{insecure2}$	0.05
$r_{recovery1}$	0.10	r_{link1_12}	$1/10 r_{link1}$
$r_{recovery2}$	1.00	r_{link2_12}	$1/10 r_{link2}$
r_{memory}	1.00	r_{link3_12}	$1/10 r_{link3}$

This paper uses the Bio-PEPA Workbench and the Bio-PEPA Eclipse plug-in to solve the model. The test environment is Windows 7, 64 bit processor, dual-core CPU, 2.4 GHz and 8 GB RAM.

It needs to be explained here that because our model adopts a formal description method, it is suitable for any given system or instance, so that the selection of instances is not the only one.

3.3 Simulation experiments

In order to test whether the proposed model is reasonable, we used the stochastic algorithm to simulate the real system and compare it with the ODEs simulation. Assuming that the number of all components in the two service domains is 200, and the initial values of N_s, N_e, N_c, N_i, N_r, N_a are, respectively, (50, 10, 10, 10, 10, 10) and (100, 0, 0, 0, 0, 0). The Gillespie random algo-

rithm[18] was used to select 10 000 groups of random data, and the error is set to be 1×10^{-5}. The final comparison maps are shown in Fig. 3 and Fig. 4.

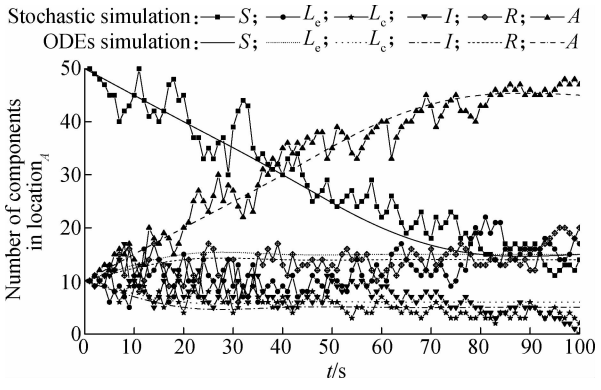

Fig. 3 The comparison of stochastic simulation and ODEs simulation in location$_A$

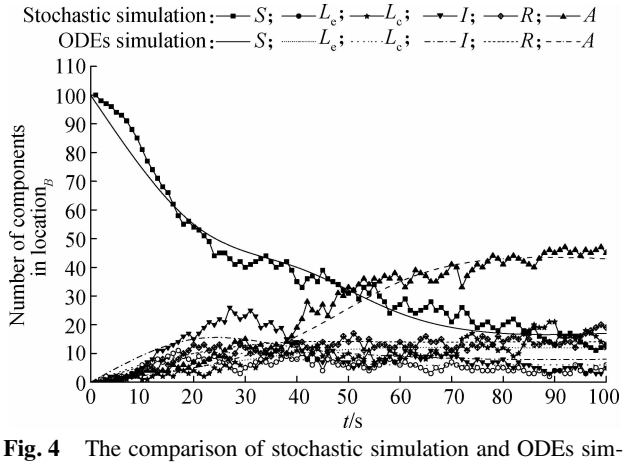

Fig. 4 The comparison of stochastic simulation and ODEs simulation in location$_B$

As shown in Fig. 3, at the beginning, there are some infection state nodes in location$_A$, and many nodes are susceptible to being infected. Virus propagation in intra-domains will first occur, and therefore, the number of components in L_{set} generally rises and fluctuates greatly. Later, due to the increase of memory state nodes and incomplete matching, latent state nodes L_e also increase relatively. However, they are much faster to repair than susceptible state nodes. Finally, the recovery state nodes and the latent state nodes L_e fluctuate within a certain range, when infection state nodes decrease and approach 0; other components gradually show a steady trend.

In Fig. 4, the initial state of location$_B$ is susceptible to being infected and there are no infection state nodes. After being affected by the virus propagation in location$_A$, the susceptible state nodes in location$_B$ will rapidly decrease and will be transformed into latent state nodes L_c and infected state nodes. In the process of repairing infected state nodes, memory state nodes will increase slowly. Meanwhile, the number of components in L_{set} rises and gradually decreases after reaching their peak value. Finally, all components will achieve a relatively stable state.

The comparisons of the two graphs show that the initial conditions of the two service domains are different, and the evolution process of the survivability state and the final results are also different. In location$_A$, $P_v = 0.37$, $\pi_v = 0.22$; in location$_B$, $P_v = 0.39$, $\pi_v = 0.24$. As a result, compared with location$_A$, location$_B$ shows that the range of viruses propagation is wider and the impact time on service domains is longer.

3.4 Model analysis

3.4.1 The effect of inter-domain virus propagation

One of the main factors that affect survivability is the virus propagation within service domains. In this paper, the virus transmission rate of inter-domains is mainly controlled by r_{link1_12}, r_{link2_12} and r_{link3_12}. Among them, r_{linki_12} contains r_{link1_12}, r_{link2_12} and r_{link3_12}. In order to analyze the influence of virus propagation on survivability, we use P_v and π_v as two indices, and consider the scope and trend of the virus propagation under a survivability situation for the whole cloud computing system.

Fig. 5 shows the changes in the number of components L_{set} in location$_B$, when the connection rate r_{link1_12}, r_{link2_12} and r_{link3_12} are expanded 5 times, 10 times and reduced 5 times, 10 times, respectively. As we can see from the figure, with the increase of r_{linki_12}, the number of components in L_{set} grows gradually, and the peak time is shortened and the peak value becomes larger. When r_{linki_12} is reduced, the number of components in L_{set} decreases gradually, and the peak time increases, but the peak value decreases relatively.

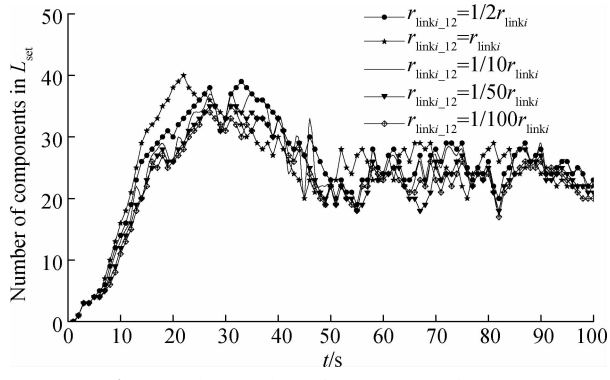

Fig. 5 The number of components in L_{set}

According to the data in Fig. 5, using Eqs. (7) and (8), we obtain the values of P_v and π_v when r_{linki_12} undergoes different changes, respectively, as shown in Fig. 6.

As can be seen from Fig. 6, with the increase of r_{linki_12}, the corresponding values of P_v are 0.40, 0.41, 0.39, 0.38, and 0.37, respectively. It means that the effect of the virus spreads with the increase of the inter-domain propagation rate. In the same way, the corresponding values of π_v are 0.25, 0.26, 0.24, 0.23, 0.22, respectively, which implies that the virus has a longer period impact on the system. This phenomenon is

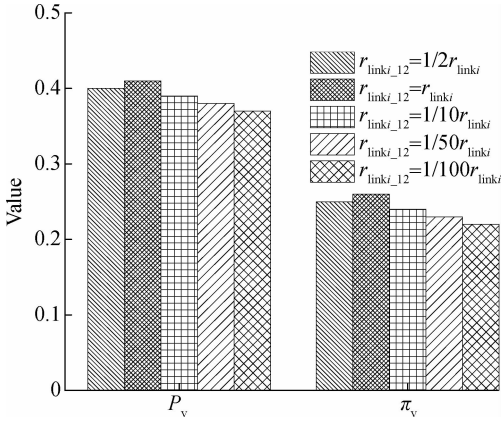

Fig. 6 The effect of inter-domain propagation rate

mainly due to the increased probability of virus propagation and the shorter propagation time. However, the overall recovery ability of the system remains unchanged, and the number of infected status nodes is increased, that is, it causes a longer recovery time of the system.

3.4.2 The effect of recovery ability

In the model, the recovery ability can make the critical services of the infected state nodes return back to normal, and it can reduce the proportion of infected nodes effectively, which is of great benefit to enhancing the survivability of the system. Next, we take P_v and π_v to analyze the influence of the recovery ability on survivability through the change between $r_{recovery1}$ and $r_{recovery2}$ in $[0.1, 1.0]$.

As we can see from Fig. 7 and Fig. 8, with the increase of $r_{recovery2}$, P_v and π_v show a decreasing trend, but the effect of $r_{recovery1}$ on P_v and π_v is not as obvious as $r_{recovery2}$. With the increase of $r_{recovery1}$, the values of P_v and π_v fluctuate within a certain range. Although there is a gradual decline in the trend, the magnitude is much smaller. This is mainly because components S and L_e will eventually be transformed into component I, and component L_e may also be converted to component I. Compared with the transformation of component L_e into component R, the probability of component I to be transformed into component R is much greater. Therefore, by enhancing the repair ability, component I can suppress the virus propagation more effectively.

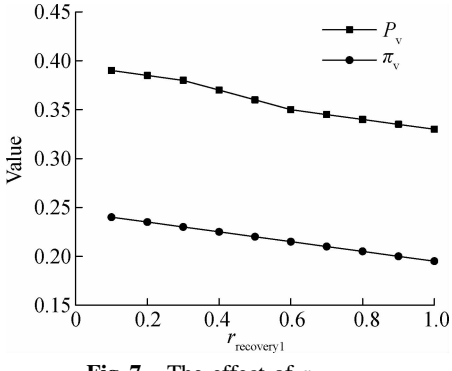

Fig. 7 The effect of $r_{recovery1}$

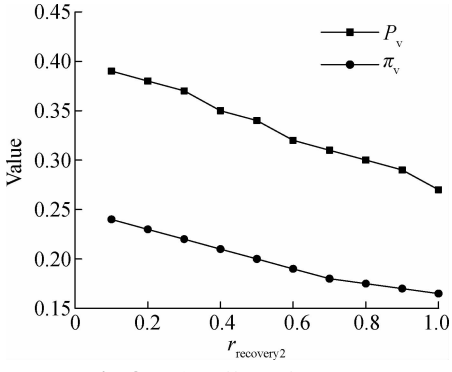

Fig. 8 The effect of $r_{recovery2}$

3.4.3 The effect of memory ability

Memory ability is a very important part of the proposed model. Due to the immune memory function and incomplete matching, the component of the recovery state can be transformed into the component of the immune state, and the possibility of reinfection is reduced. Even if infected again, it can be recovered in a relatively short time.

It can be seen from Fig. 9 that the values of P_v and π_v show a linear decline trend with the improvement of memory ability, especially π_v, and that P_v is relatively flat. The main reason is that when the memory ability of the system is enhanced, the recovery state nodes will be transformed into immune state nodes faster. As the number of nodes in the immune state increases, those nodes have a better resistance to the subsequent virus attacks. Moreover, the recovery ability of the system will be relatively enhanced, which will exert a certain inhibition effect on the virus propagation.

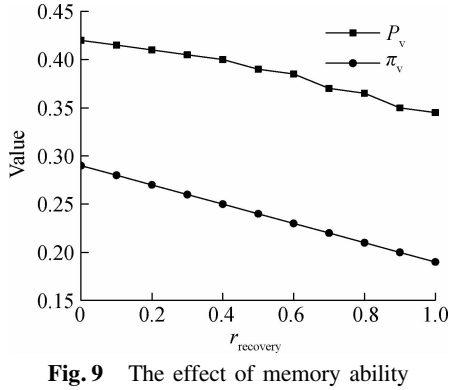

Fig. 9 The effect of memory ability

4 Conclusions

1) Reducing the propagation rate of inter-domains can effectively control virus propagation in inter-domains, and can delay the decline trend of the system survivability.

2) Strengthening the recovery ability of a system can make the system return back to normal work as soon as possible.

3) Enhancing the memory ability of a system can greatly improve the system survivability.

In the future, we plan to study the classification of each typical virus propagation, virus variation due to the changes in the external environment, and the survival resistance mechanism within a system and so on. The model may be adjusted according to the real situation, and the impact of the other parameters in the model will also be further analyzed.

References

[1] Westmark V R. A definition for information system survivability [C]//*Proceedings of the 37th Annual Hawaii International Conference on System Science*. Washington, DC, USA: IEEE Computer Society, 2004: 2086 - 2096. DOI:10.1109/HICSS.2004.1265710.

[2] Mell P, Grance T. The NIST definition of cloud computing[J]. *Communications of the ACM*, 2011, **53**(6): 50 - 50. DOI: 10.6028/NIST.SP.800-145.

[3] Chang X L, Zhang Z J, Li X D, et al. Model-based survivability analysis of a virtualized system[C]//*IEEE 41st Conference on Local Computer Networks (LCN)*. Dubai, United Arab Emirates, 2016: 611 - 614. DOI:10.1109/LCN.2016.104.

[4] Jin Y L, Zhou X Q, Bai Z S, et al. Survivability-aware topology evolution model with link and node deletion in wireless sensor networks [J]. *International Journal of Distributed Sensor Networks*, 2014, **10**(4): 278629. DOI:10.1155/2014/278629.

[5] Zhou J A, Miao H K, Kai J Y, et al. Survivability prediction of web system based on log statistics[C]//*IEEE ACIS 16th International Conference on Software Engineering, Artificial Intelligence, Networking and Parallel/Distributed Computing (SNPD)*. Takamatsu, Japan, 2015: 15359578. DOI:10.1109/SNPD.2015.7176170.

[6] Alobaidi I A, Sarvestani S S, Hurson A R. Survivability analysis and recovery support for smart grids[C]// 2016 *Resilience Week (RWS)*. Chicago, IL, USA, 2016:33 - 39. DOI:10.1109/rweek.2016.7573303.

[7] Moldovan R D, Todoran E N. Immune system modeling and analysis using Bio-PEPA [C]//*IEEE International Conference on Intelligent Computer Communication and Processing*. Cluj-Napoca, Romania, 2015: 475 - 482. DOI:10.1109/iccp.2015.7312706.

[8] Tan Y, Zhang P. Immune based computer virus detection approaches[J]. *CAAI Transactions on Intelligent System*, 2013, **8**(1): 80 - 94. DOI:10.3969/j.issn.1673-4785.201209059.

[9] Piqueira J R C, de Vasconcelos A A, Gabriel C E C J, et al. Dynamic models for computer viruses[J]. *Computers & Security*, 2008, **27**(7): 355 - 359. DOI:10.1016/j.cose.2008.07.006.

[10] Li J, Yang Y, Zhou Y. Global stability of an epidemic model with latent stage and vaccination[J]. *Nonlinear Analysis: Real World Applications*, 2011, **12**(4): 2163 - 2173. DOI:10.1016/j.nonrwa.2010.12.030.

[11] Lü H W, Wang H Q, Lin J Y, et al. A vulnerability propagation model of distributed virtualized systems based on Bio-PEPA[J]. *Chinese Journal of Computers*, 2016, **39**(2): 391 - 404. DOI:10.11897/SP.J.1016.2016.00391. (in Chinese)

[12] Ciocchetta F, Hillston J. Bio-PEPA: A framework for the modelling and analysis of biological systems[J]. *Theoretical Computer Science*, 2009, **410**(33): 3065 - 3084. DOI:10.1016/j.tcs.2009.02.037.

[13] Galpin V. Hybrid semantics for Bio-PEPA[J]. *Information and Computation*, 2014, **236**: 122 - 145. DOI:10.1016/j.ic.2014.01.016.

[14] Duguid A. An overview of the Bio-PEPA eclipse plug-in [C]//*Eighth Workshop on Process Algebra and Stochastically Time Activities*. Edinburgh, UK, 2009: 121 - 132.

[15] Zhao J, Zhou Y, Shuo L. A situation awareness model of system survivability based on variable fuzzy set[J]. *Indonesian Journal of Electrical Engineering and Computer Science*, 2012, **10**(8): 2239 - 2246. DOI:10.11591/telkomnika.v10i8.1691.

[16] Chen T P, Cui W Y, Meng X R, et al. A method of IP network survivability evaluation method under performance monitoring [J]. *Journal of Beijing University of Posts and Telecommunications*, 2015, **38**(6): 20 - 23. DOI:10.13190/j.jbupt.2015.06.005. (in Chinese)

[17] Van Mieghem P, Omic J, Kooij R. Virus spread in networks [J]. *IEEE/ACM Transactions on Networking*, 2009, **17**(1): 1 - 14. DOI:10.1109/tnet.2008.925623.

[18] Gillespie D T. Stochastic simulation of chemical kinetics [J]. *Annual Review of Physical Chemistry*, 2007, **58**(1): 35 - 55. DOI: 10.1146/annurev.physchem.58.032806.104637.

基于 Bio-PEPA 的云计算系统可生存性建模和分析

赵国生[1] 任孟其[1] 王 健[2] 廖祎玮[1]

([1] 哈尔滨师范大学计算机科学与信息工程学院, 哈尔滨 150025)

([2] 哈尔滨理工大学计算机科学与技术学院,哈尔滨 150080)

摘要:面向云计算系统,结合生物免疫系统的记忆功能以及不完全匹配性,通过对关键云服务可生存态势的分析,提出了一种云计算系统可生存性的形式化建模与分析方法. 首先,在 SAIR 模型、SEIRS 模型和分布式虚拟化系统脆弱性扩散模型的基础上,将病毒演化状态分为 6 种类型,然后分析了病毒在云计算系统服务域内的扩散规则和服务域间的传播规则. 最后,基于 Bio-PEPA 对关键云服务可生存性态势演化进行形式化建模,得到 SLIRAS 模型. 基于随机模拟和 Bio-PEPA 模型的 ODEs 模拟,从病毒的域间传播速率、修复能力、记忆能力 3 个方面对模型敏感参数进行了试验分析. 结果表明,所建立的模型与实际云计算系统的可生存性态势近似拟合度高,能够很好地反映系统可生存性的变化.

关键词:云计算系统;Bio-PEPA;可生存性;随机模拟

中图分类号:TP309