Security Attack Mitigation Framework for the Cloud

Esha Datta, , Indian Institute of Technology
Neeraj Goyal,, Indian Institute of Technology

Key Words: Cloud computing, Security, Attack Graphs, Markov Chain, Security Administration.

SUMMARY & CONCLUSIONS

Cloud computing brings in a lot of advantages for enterprise IT infrastructure; virtualization technology, which is the backbone of cloud, provides easy consolidation of resources, reduction of cost, space and management efforts. However, security of critical and private data is a major concern which still keeps back a lot of customers from switching over from their traditional in-house IT infrastructure to a cloud service. Existence of techniques to physically locate a virtual machine in the cloud, proliferation of software vulnerability exploits and cross-channel attacks in-between virtual machines, all of these together increases the risk of business data leaks and privacy losses. This work proposes a framework to mitigate such risks and engineer customer trust towards enterprise cloud computing.

Everyday new vulnerabilities are being discovered even in well-engineered software products and the hacking techniques are getting sophisticated over time. In this scenario, absolute guarantee of security in enterprise wide information processing system seems a remote possibility; software systems in the cloud are vulnerable to security attacks. Practical solution for the security problems lies in well-engineered attack mitigation plan. At the positive side, cloud computing has a collective infrastructure which can be effectively used to mitigate the attacks if an appropriate defense framework is in place. We propose such an attack mitigation framework for the cloud.

Software vulnerabilities in the cloud have different severities and different impacts on the security parameters (confidentiality, integrity, and availability). By using Markov model, we continuously monitor and quantify the risk of compromise in different security parameters (e.g.: change in the potential to compromise the data confidentiality). Whenever, there is a significant change in risk, our framework would facilitate the tenants to calculate the Mean Time to Security Failure (MTTSF) cloud and allow them to adopt a dynamic mitigation plan. This framework is an add-on security layer in the cloud resource manager and it could improve the customer trust on enterprise cloud solutions.

1 INTRODUCTION

Cloud computing is a new paradigm of economical IT infrastructure which significantly reduces the companies' IT spending by renting the required infra-structure from third party (cloud) service providers. Technically, the cloud service providers share their IT resources among their client companies using appropriate control software products. Most of the cloud service providers like Amazon EC2, Rackspace, Microsoft azure etc. charge their customers' as per the utilization of the resources. Further, the customers' can also dynamically add resources to their IT facilities based on the real time demands. For example, a health care provider can gradually add extra disk space to store patients' health records as their service extends to more patients. A web search company can add extra servers and increase bandwidth of their service as they get more and more search requests. An online stock brokering company may require extra fast computing facilities as they expand their portfolios into variety of sectors. Startups in such business domains would leverage their expertise to many customers with less IT spending and in a short period. On the other hand large organizations may focus on their core competency by outsourcing maintenance of IT infrastructure. The elastic nature of cloud resources and structured maintainability of cloud systems would reduce the IT spending of all organizations in many ways.

The flip side of the technology is that the companies' data and processes would remain outside of their direct administrative purview. The public clouds are equally open to all customers; there are possibilities that the competitors and the adversaries of a company may also host their services in the same cloud service provider and access the same IT infrastructure. Even though the cloud systems have placed software based controls to protect each customer's data and services from other customers and attackers, the software products are often shipped with undiscovered vulnerabilities. Well engineered software products in the production systems also expose severe vulnerabilities to the hackers.

For example, many buffer overflow vulnerabilities are reported in Windows based operating systems, in different variants of Linux operating systems and in the popular DBMS systems. Hackers often write exploits for these vulnerabilities and use them along with the side-channel attacks to gain illegal access. Cloud technology is not an exception; within the cloud too it may also be possible to identify and locate the physical locations of software instances. Such techniques when combined with the vulnerability exploits and cross-channel attacks would create harmful security breach in the cloud. Such security concerns, despite of the overwhelming benefits of cloud technology, keeps away a large number of

978-1-4799-2848-4/14/$31.00©2014 IEEE

customers from moving their business into the cloud.

It must be noted that in-house IT infrastructures are also vulnerable to similar security threats. There are several instances of security breaches in many in-house enterprise information systems. They had often led to financial losses. Software engineering processes are yet to mature to produce 100% bug free software products at industry scale. The difficulty of building provable correct software products stems from the fact that the software products are much versatile and huge in volume. Thus, there is a remote possibility that there will be any immediate fix for all security attacks. One practical solution could be to have a well-engineered attack mitigation strategy. In this regard, cloud computing has a collective infrastructure of many companies who may be the targets of security attacks. Within the cloud, enough intelligence can be gathered from the attempts made to compromise the security of different companies. This intelligence can be used to secure each tenant's resources. Such intelligence may not be obtainable from in-house IT infrastructure of an individual company.

In this paper, we propose a security attack mitigation framework for the cloud which could facilitate the collection and utilization of the security intelligence gathered from the cloud environment to secure the tenants resources from potential attacks. Simplicity of the framework would illustrate feasibility as well as would be helpful in building the trust of the cloud customers that they can get a better protection for their data and services while availing the benefits of cloud technology.

2 MOTIVATION

One interesting observation about the cloud is that many customers run different instances of same set of software products and such instances would have their own data and custom business logic. However, a vulnerability found in a software product can be exploited across different customer instances. Such security exploits may leave different impact on their business values; hence, it is reasonable to have different security policies, for an attack, specific to each tenant's specific requirements.

The nature of the cloud technology provides a lot of scope for the cloud administrators to gather intelligence about possible attacks on cloud infrastructure and such intelligence includes new vulnerabilities on the software products, presence of potential intruders who may exploit vulnerabilities and the possibilities of establishing(illegal) communication among various instances cutting across different customers. Further, it is possible for the cloud admin to accurately collect properties of all software instances (SI) such as the physical server which host the SI, the list of SIs which shares the same physical server and the established communication channels between the SIs. Such information can be systematically combined as an attack graph and potential vulnerabilities of each SI in the context of the cloud can be assessed.

Cloud administrators have the privilege for patching vulnerabilities; some vulnerability can be patched by the cloud admin themselves and most of the vulnerabilities require patches from the corresponding software vendors. For example, vulnerabilities in default router configuration can be patched by the cloud admin themselves whereas buffer overflow vulnerabilities require a vendor specific patch. Meanwhile, the cloud administrator may reconfigure the cloud network by reorganizing the virtual machines. However, they would not have the privilege for changing the internal configurations of SIs since they are owned by the tenants.

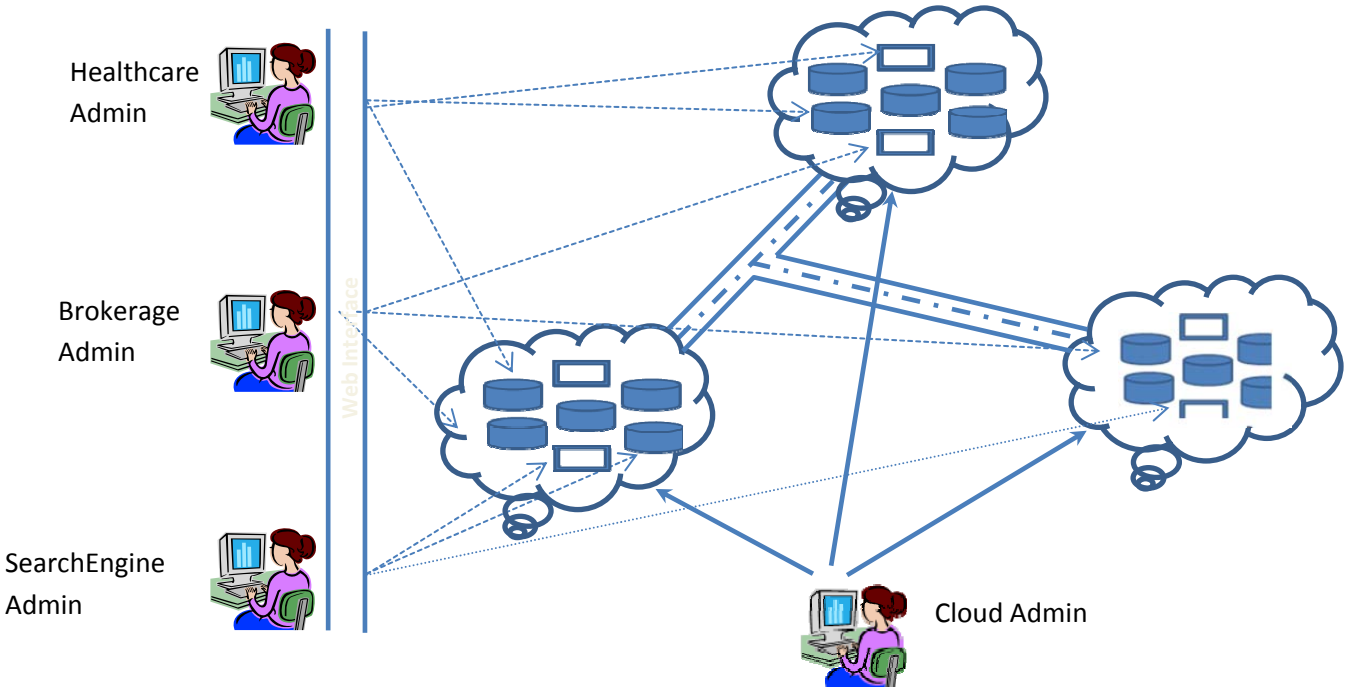

Figure 1: Administrators in Cloud Environment

Further, each tenant may have a different set of business values to protect. Cloud admin may not able to design and apply a uniform security policy which could protect all tenants' business interest. Such policies are best understood and controlled by the corresponding business administers. For example, in the figure 1, if there is potential attack on confidentiality of data, the health care administrator may adopt a security policy to allow only the doctors to access patients' medical records and deny insurance companies, pharmaceutical organizations, health surveys etc. from accessing to the records. The administrators of search engines and the administrators of stock brokering companies would like to adopt security policies different from the health care administrators. Hence, it would be appropriate for the cloud admin to convey the potential threats to the tenants' business admins and facilitate them to make changes in their own security settings such that their business values are protected and rearrange VMs places, if required.

In this work, we present a high level framework to facilitate the cloud admins and the tenants' business admins to collaborate and mitigate the potential security attacks. Realization of full advantages offered by cloud technology requires significant trust and reliability over the cloud services. In particular, the cloud platform should provide a level of administrative policy enforcement over their resources if there is a potential security threat present in the cloud environment. Giving the administrative flexibility and sharing the security intelligence about the potential security threats in the cloud environment would help the cloud service providers to gain trust of their customers as well as thwart the security threats.

3 BACKGROUND

In this work, we propose a graph based security attack mitigation framework to improve the security protection in the cloud. This section presents a brief summary of related works in cloud security.

Chunxiao Li *et al* [1] proposed a secure run-time environment which includes storage, network interface and computing for the VM instances within the public cloud. A small hypervisor layer with high assurance of integrity is separately managed and it uses trusted computing techniques. The management virtual machine is managed by the cloud administrator. The clients are ensured better confidentiality and integrity of security critical virtual machines, which they want to run within the cloud, even under an untrusted management of virtual machines. However, the security attacks based on vulnerabilities in the software instances like Word processors would remain open to the attackers.

Min Li *et al* [3] proposed techniques to place virtual machines in a way that minimizes the security risks by considering connections among virtual machines. It has been assumed that the virtual machine with the highest number of connections is the most vulnerable one, and probability of it being attacked has been calculated by a linear mapping function. They proposed an algorithm which sorts the virtual machines in descending order of attack possibility. Main idea of the algorithm is as follows: Each node (each server in the data center) in cloud is assigned a virtual machine from the VM set, then the node with minimal attack possibility is chosen to hold the rest of VMs such that they get to place the virtual machines with the highest vulnerability in the safest nodes.

Soren Bleikertz *et al*[4] proposed an approach to construct vulnerability based attack graph from the configurations of virtual machines in the Amazon EC2 cloud. They developed a policy language for specification of security requirement and an algorithm to perform the reachability analysis. The reachability analysis checks whether the given configurations of VMs have any potential security threat.

Miika Komu *et al* [5] show the application of Host Identity Protocol to establish a secure communication between virtual machines in the cloud. The HIP protocol (proposed by Internet Engineering Task Force) is an end-to-end protocol, that is, works to establish a communication between two end user applications using public key infrastructure. The idea of [5] is to separate the communication between VMs with the cloud and the communication between the customer and the VM. Developers and administrators can access cloud services directly over HIP, whereas consumers can access the cloud by using reverse HTTP proxy (without HIP). The proposed scheme mitigates some of the privacy issues related to multi-tenancy within a single data center. The HIP is deployed in an end to middle manner to tackle the security issues related to multi-tenancy and hybrid clouds.

Eric Keller *et al* [6] proposed to remove the virtualization layer while retaining the key features enabled by virtualization. They proposed a new architecture called "NoHype" to perform the key roles of the virtualization layer such as arbitrating access to CPU, memory and I/O devices and acting as a network device and managing the initiating and terminating guest virtual machines. Proposed architecture makes use of hardware virtualization extensions available in newer version of CPUs and it removes the need for a separate virtualization layer to run the virtual machines. They also use the CPU extensions to flexibly partition the resources and isolate guest virtual machines from each other.

Our work is based on attach graphs. Attack graphs represents the relation between vulnerabilities in the given network configuration [7]. We use an annotated attack graphs to encode security vulnerability of cloud environment. We propose a framework to share the information about vulnerabilities present in the cloud environment with the tenants so that they can adopt their own security protection policies as per their business needs.

4 NOTATION

Symbols	Meaning
SE	Set of Security Events.
CIA	Confidentiality Integrity Availability.
PSE	Set of potential security events.
SI	Software Instance in the cloud.
SSP_{SI}	Security state space of SI.

AAG	Annotated Attack Graph of the cloud
$INCT_{seq}$	Incident sequence, a path in the AAG
SEC_{int}	Security intelligence collection function which takes a set of $INCT_{seq}$ as input and return PSE set.
SPP	Security Protection Policy.
B_{spp}	Business specific security protection policy.
SAM_{spp}	Security Adjacency Matrix for SPP.
$LSS_{\alpha i}$	Local Security State of the SI calculated using the security warning and intelligence received from the cloud admin.
$LSS_{\alpha t}$	Local Security State of the SI after$_{\alpha t}$ steps of enforcing $B_{spp.}$
US_i	Probability of attack on Integrity of Data and Computing Processes of the SI.
US_c	Probability of attack on Confidentiality of the SI's Data.
US_a	Probability of attack on Availability of Data and Services of SI to the cloud users.

5 ATTACK MITIGATION FRAMEWORK

Cloud computing technology is used to rent resources under three types of models, namely, Infrastructure as a Service (IaaS), Platform as a Service (PaaS) and Software as a Service (SaaS). All these three types of resources are accessible through a web based control panel using a standard web browser. The control panel provides an interface to the cloud resources. At the data center side, the virtualization technology based technologies are used to create soft instances of the resources as per the demands of the customers. For example, in the IaaS model virtual machines with specific computing power and storage capacity are created as software instances to share the infrastructure. Similarly other models also create respective software instances as per the customer request. We refer the IaaS instances, the PaaS instances and the SaaS instances using a generic term Software Instances (SIs).

5.1 Annotated Attack Graphs

Most of the security attacks in networked environment are multi-hop in nature; first the attacker has to establish a communication between his machine and one of the SI, then he may connect through other SIs to reach the target SI. In between, he may have to use hacking techniques like running exploits, executing side channel attacks etc. to gain appropriate access to the intermediate SIs before reaching the target SI.

We assume that the cloud administrator gathers knowledge about the present states of the entire SIs and their interconnection, vulnerabilities in the software products used in the cloud. Such vulnerabilities can be found through different sources like vulnerability reports of the product vendors, CVSS, failure data from the cloud etc. The cloud admin uses such information to form an annotated attack graph comprising of all (vulnerable)SIs within the cloud.

Definition 1: Security Event SE is defined as a hacking incident in an SI. Each security event SE has three Boolean attributes namely, C, I and A denoting possible compromises in the target SI's Confidentiality, Integrity and Availability.

For example, if the 'I' attribute of a security event is true then the hacking is aimed at compromising the integrity of the data available at the target SI. A security event may be aimed to compromise a subset of {C, I, A} in the target SI. To capture the relations among the vulnerable SIs, an Annotated Attack graph can be constructed using a security metric presented in [2]. In this work, we are not exploring the methods to construct an attack graph, rather our focus is on building a framework to use such attack graphs to mitigate the security attacks. We use a variant of attack graph called annotated attack graph which is defined as follows

Definition2: An annotated attack graph AAG(V, E) is a directed acyclic graph where the vertices set V is the set of vulnerable SIs. The edge set E is collection of binary relations over the vertices of the form (V_{src}, V_{dest}) and it represents a requirement that the attacker has to get access to V_{src} before accessing $V_{dest..}$ Additionally, each vertex in the set V is annotated with a set of possible security events $SE_{v.}$

In this attack graph, each vertex has the probability value P_v representing the degree of possibility that the attacker could succeed to get an access to the SI using all possible SEs. The terminal vertices in the AAG denote the possible target SIs. The paths leading to the SIs carries the valuable security intelligence from the cloud environment. The proposed framework passes this information to the business admins of the target SIs to adopt the security policies specific to their business assets and values.

5.2 Tenant Specific Security Intelligence

Annotated attack graph of the cloud represents vulnerabilities of entire SIs and the potential sequence of exploits which would lead to the target SIs. However, the annotated attack graph cannot be shared with all the tenants of the target SIs since it may contain sensitive information about the other tenants' SIs. For example, a potential attacker may run exploits on SIs of other co-tenants before making an access path towards the target SI. If we share the potential access path with the target SIs tenant, he may get to know the vulnerabilities present in his co-tenant's SIs. Further, the business admin of the target tenant may not be able to reconfigure the other tenants' SIs. Hence, it would not be appropriate to share the AAGs globally with each tenant. However, information about the severity of the attacks and their impact on the confidentiality, integrity and availability properties of their own data and services would be much valuable security intelligence.

Definition 3: The security intelligence collection function SEC_{int} is defined as a mapping from a potential attack path in the annotated attack graph to the target SI. The SEC_{int} function returns a set PSE, the set of potential security events which could lead an attacker to gain an illegal access over the target SI.

Each security event in the set PSE has three attributes mentioning the possibilities of the security compromise (in terms of CIA) in the tenant's resources. As explained in the

motivation section, the business admins of each target tenant may enforce their own security policies. In the framework, each cloud tenant can access the SEC_{int} to get the PSE related to their SIs. The cloud admin could construct the AAG and link it with the SEC_{int}.

5.3 Business Specific Security Protection Policy Administration

The AAG of the cloud identifies a set of target SIs who may be the victims of potential security attacks. Each terminal vertex in the AAG carries a probability value $\underline{P_v}$which denotes the probability that the target SI is in an insecure state.

Definition 4: The security state space of the SI consist of four states, namely, S_{safe}, US_c, US_i and US_a representing the safe state, the unsafe states with respect to confidentiality, integrity and availability. The security state space of SI is denoted by the tuple SSP_{SI}<S_{safe}, US_c, US_i,US_a>.

Initial state probabilities ($LSS_{\alpha i}$) of the SSP_{SI} can be calculated from the P_v and *PSE* as follows

$$S_{safe} = 1\text{-}P_v$$

$$US_c = [\ \sum pse^k \text{ in } PSE(pse^k.c)\ \ /\ (\sum pse^k \text{ in } PSE(pse^k.c + pse^k.i + pse^k.a))\ ] * P_v$$

$$US_i = [\ \sum pse^k \text{ in } PSE(pse^k.i)\ \ /\ (\sum pse^k \text{ in } PSE(pse^k.c + pse^k.i + pse^k.a))] * P_v$$

$$US_a = [\ \sum pse^k \text{ in } PSE(pse^k.a)\ \ /\ (\sum pse^k \text{ in } PSE(pse^k.c + pse^k.i + pse^k.a))] * P_v$$

The initial state probabilities would indicate the possible compromises in CIA components of security in the SI. The business administrator the SI's tenant can devise a business specific security protection policy B_{spp} and gradually enforce it until the required security state is achieved. Each step of protection policy enforcement would make changes in the values of $LSS_{\alpha i.}$ The nature of the policy and the kind of state transition it may make would vary from tenant to tenant. For example, if the tenant has more than one SIs as target in the AAG then he may reconfigure the communication setting between the SIs. He may also express his intend of reconfiguring the physical placement of SIs within the cloud to the cloud admin, possibly by paying a nominal charge. He may simply change the authorization settings of business users.

In general, it would be possible for the business administrators of each tenant to predict the changes it can make in the security state space using their business logic and history of business data. The possible security impact of the policy B_{spp} can be represented using a Security Adjacency Matrix $SAM_{spp.}$ Markov chain can be used to predict the possible impact of security states in the SI after αt steps of the B_{spp} enforcement using the following formula

$$LSS_{\alpha t} = LSS_{\alpha i.*}\ [SAM_{spp}]^{\alpha t}$$

Alternatively, this formula can also be used to determine the number of enforcement steps required to achieve the desired protection for the given potential security threat. If the $LSS_{\alpha t}$ is not having the expected security properties then the business admin can calculate the MTTSF using the standard Markov procedures and request the cloud admin to reconfigure the SI placements within the cloud, possibly by paying appropriate charges. However, we are not elaborating the technicalities of Markov procedure since the aim of this work is to present a security attack mitigation framework which can make best use of security intelligence available in the multi-tenant cloud environment. Our approach would help the cloud service provider to gain customers trust on security of their data and hence, cloud services.

ACKNOWLEDGEMENTS

This work is supported by the INSPIRE fellowship, Department of Science and Technology, Government of India.

REFERENCES

1. Chunxiao Li, Anand Raghunathan and Niraj K. Jha: "Secure Virtual Machine Execution under an Untrusted Management OS", *Proceedings of the IEEE 3rd International Conference on Cloud Computing (CLOUD),* Miami, Florida, USA, July5-10, 2010, Pages: 172-179.
2. Lingyu Wang, Tania Islam, Tao Long, Anoop Singhal, Sushil Jajodia: "An Attack Graph-Based Probabilistic Security Metric",*Proceedings of the 22nd Annual IFIP WG 11.3 Working Conference on Data and Applications Security*, London, UK, July 13-16, 2008, Pages: 283-296.
3. Min Li, Yulong Zhang, Kun Bai, WanyuZang, Meng Yu and Xubin He: "Improving Cloud Survivability through Dependency based Virtual Machine Placement", *Proceedings of the International Conference on Security and Cryptography*, Rome, Italy, July24-27, 2012, Pages: 321-326.
4. SörenBleikertz, Matthias Schunter, Christian W. Probst, DimitriosPendarakis and Konrad Eriksson: "Security audits of multi-tier virtual infrastructures in public infrastructure clouds", *Proceedings of the 2nd ACM Cloud Computing Security Workshop*, Chicago, IL, USA, October 8, 2010, Pages: 93-102.
5. MiikaKomu, MohitSethi, RamasivakarthikMallavarapu, HeikkiOirola, Rasib Khan and SasuTarkoma: "Secure Networking for Virtual Machines in the Cloud", *Proceedings of the IEEE International Conference on Cluster Computing Workshops*, Beijing, China, September 24-28, 2012, Pages: 88-96.
6. Eric Keller, JakubSzefer, Jennifer Rexford and Ruby B. Lee: "NoHype: virtualized cloud infrastructure without the virtualization", *Proceedings of the 37th International Symposium on Computer Architecture (ISCA 2010),* Saint-Malo, France, June 19-23, 2010, Pages: 350-361.
7. Paul Ammann, DumindaWijesekera and SaketKaushik, "Scalable, graph-based network vulnerability analysis", *Proceedings of the 9th ACM conference on Computer and Communications Security*, Washington, DC, USA November 17-21, 2002, Pages: 217 - 224.

BIOGRAPHIES

Ms Esha Datta,
Research Fellow,
Reliability Engineering Centre,
Indian Institute of Technology Kharagpur,
India, WB 721 302.

Email:eshacs07@gmail.com

Ms Esha Datta has received a bachelor and a master degree in computer science. At present, she is a Research Fellow in Reliability Engineering Centre, Indian Institute of Technology (IIT)Kharagpur, India. Her broad research area is Security and Reliability of Information Systems. Since last couple of years she is working on Enterprise Cloud. Her research is supported by the Department of Science and Technology, Government of India through the INSPIRE fellowship. She is also a member of the IEEE student chapter at IIT Kharagpur.

Dr Neeraj Goyal, Ph.D.
Assistant Professor,
Reliability Engineering Centre,
Indian Institute of Technology Kharagpur,
India, WB 721 302.

Email: ngoyal@hijli.iitkgp.ernet.in

Dr. Neeraj Kumar Goyal is currently an Assistant Professor in Reliability Engineering Centre, Indian Institute of Technology Kharagpur, India. He has received his Ph.D. from IIT Kharagpur in Reliability Engineering in 2006. His Ph.D. thesis was 'On Some Aspects of Reliability Analysis and Design of Communication Networks'. He received his Bachelor of Engineering degree in Electronics and Communications Engineering from MREC Jaipur, Rajasthan, India in 2000. His areas of research and teaching are network reliability, software reliability, electronic system reliability etc. He has undertaken consultancy projects from various organizations like NPCIL, Vodafone, ECIL etc. He has also contributed several research articles to international journals and conference proceedings.