Abstract
Organizations migrate to the cloud computing environment to achieve cost reduction, business agility, and elasticity. However, cloud migration brings another set of challenges such as data security & privacy, resource management, and compliance. Recently, cloud vendors have started offering services to their customers to collect detailed logs of events and resource utilization. However, the cloud computing ecosystem lacks frameworks, models, and IT artifacts to analyze such logs to draw business insights and address the above challenges. Following Design science research paradigm, we present a Gaussian Bayesian Network based approach for learning the underlying dependencies among the events in cloud services to determine the antecedents and consequences of security related events. Moreover, using our model, we predict the security related events with average mean error of 0.13 events for one day ahead forecast. We further discuss our research implications for software development, security, and IT audit in cloud computing environment.

Keywords
Cloud computing, security, Gaussian Bayesian Network, AWS

Introduction
Cloud computing is new model for IT provisioning for organizations (Labes, Hanner, & Zarnekow, 2017; Riedl, Leimeister, Böhm, Yetton, & Krcmar, 2010). It reduces fixed IT cost, increases organizational flexibility, and decreases time to market new products and solutions (Müller, Holm, & Søndergaard, 2015). Extant research in Information System has focused on studying various topics in cloud computing including pricing, such as pay-per use model and spot market (Cheng, Li, & Naranjo, 2016; Singh & Dutta, 2015), adoption (Bhattacherjee & Park, 2014), etc. However, there is paucity of research addressing the security challenges in cloud computing.

There has been numerous cloud security data breaches reported in the mainstream news media recently (Larson, 2017; O’Sullivan, 2017a). In one of the events, the data of over 198 million voters were left exposed by a firm working on behalf of Republican National Committee (RNC) due to improper cloud security deployment (O’Sullivan, 2017c). In another event, personal data of around 6 million customers were leaked from a telecommunication company due to inappropriate cloud storage security settings.1 As per Upguard.com, a leading cloud security provider, one of the world’s largest corporate consulting and management firm left its cloud storage unsecured and publicly downloadable, containing sensitive business information (O’Sullivan, 2017b). One of the emerging patterns in these cloud security risks is that these are result of human error involving inappropriate cloud security deployment. Moreover, the intensity and frequency of these risks are expected to grow with increasing cloud adoption and growing complexity of cloud services.

The organizations use numerous cloud services to run their business. The leading cloud vendors such as Amazon AWS provide around forty different services. Organizations combine these different services to build their application. These services are consumed using Application Programming Interfaces or APIs and each API use is an event or an independent unit of action. With growing number of services and

---

underlying complexity, organization face challenges to assess and monitor the security of their applications. To assist customers in this regard, cloud providers have recently introduced another group of services which monitor and log the events or API calls made by organizations in different services. These logs are valuable for security and compliance research in cloud computing.

The three important challenges for organizations securing their applications in cloud environment are (1) discover and quantify security threats (2) identify potential sources (associational or causal) of those security threats, (3) predicting security threats so that corrective measures may be applied to mitigate or reduce them prior to their occurrence. We follow Design science methodology (Gregor & Hevner, 2013; Hevner, March, Park, & Ram, 2004) to present an IT artifact which applies Gaussian Bayesian Network to learn probabilistic dependency structure among the cloud events from log files. Our key focus is to determine the association between the events related to security threats to other events. This approach provides two advantages (a) determine a smaller set of event types related to security threat events so that organizations can further narrow their focus on these events to find the probable cause. (b) This technique also helps in feature selection to select events which are associated with security threat events.

Our approach of modelling security-related events in a cloud environment follows a black-box approach with respect to applications which are generating these events. Our approach does not require information related to applications deployed over the cloud infrastructure. This is a pragmatic approach because it is challenging in the cloud environment to exactly determine the relationship among the cloud events and applications generating these events. Moreover, multiple applications within an organization may share the cloud resources and services and it is challenging to associate events to corresponding applications.

We use a Gaussian Bayesian Network to determine the relationship among the events without application knowledge (Geiger & Heckerman, 1994). Gaussian Bayesian Network is a probabilistic graphical model (PGM) which learns and identifies the relationships (probabilistic dependence) among cloud events in the form of a directed acyclic graph (DAG) (D. Koller and N. Friedman, 2009). It is an emerging technique which is different from traditional Bayesian Network over categorical variables (Pearl, 1988). In Gaussian Bayesian Network, we assume that the variables are normally distributed and the probabilistic dependence relationships among the variables are represented using a linear model involving means of the independent and dependent events. We use hill-climbing algorithm (Scutari, 2009) to learn the Gaussian Bayesian Network from the data.

We use one year of longitudinal data from a multinational organization using Amazon AWS platform. The data is in form of JSON files from AWS CloudTrail service. The organization is some analytics firm which consumes variety of different cloud services from AWS to support its different applications.

Our results demonstrate that the antecedents of security related events can be probabilistically identified with our approach. Moreover, with the knowledge of dependency network, organizations can build IT controls to review the actions of their employees which lead to security related events. Furthermore, using our prediction model build on features selected from Gaussian Bayesian Network, we predict the security related events one day in advance. The average (over 15 days) mean error for one day ahead prediction is 0.13 events. The rest of the paper is organized as follows. In section 2, we present our related work followed by our approach in section 3. In section 4, we present our results. We present discussion and implications of our study in section 5, followed by conclusion in section 6.

**Related Work**

Bayesian Networks are graphical models which abstract and model the real-world using a directed acyclic graph (DAG), wherein the direction of the relationships or links among the nodes of the graph represents conditional dependence (Margaritis, 2003; Pearl, 1988). One of the key advantages of Bayesian Network over other graphical models such as Markov models (D. Koller and N. Friedman, 2009) is that unlike Markov models which lack directionality in their graphical structure, Bayesian Networks are directional. Bayesian Networks have been used in multiple domains in extant literature such as Healthcare (Weng-Keen Wong and Andrew Moore, 2003), databases (Dey & Sarkar, 2000), banking (Sarkar & Sriram, 2001), marketing (Cui, Wong, & Lui, 2006), financial networks (Gandy & Veraart, 2016) and structural modelling (Zheng & Pavlou, 2010). Please see (Buntine, 1996) for recent literature on learning probabilistic networks from data.
Bayesian Networks have been used in predictive modelling as well (Kita, Harada, & Mizuno, 2012). Kita, Harada, & Mizuno (2012) used the Bayesian Network for stock price prediction. They compared their approach with the existing time series models such as autoregressive model (AR), moving average model (MV), autoregressive and moving average model (ARIMA), and autoregressively conditionally heteroscedastic model. The Bayesian Network based approach proposed by Kita et al. (2012) outperformed the traditional time series models mentioned above.

In the cloud computing research in Information Systems, some of the recent literature has investigated the factors which inhibit cloud adoption (August, Niculesc, & Shin, 2014; Chen & Wu, 2013; Cheng et al., 2016). However, very few studies have focused on the security issues in cloud environment (August et al., 2014). August et al. (2014) focused on the difference in impact of security risk in on-premises software and SaaS offering. The authors suggest that the impact of security risk is higher in the case of cloud offerings compared to on-premises offering because, in the former, the attackers can impact multiple organizations at once. Yang & Tate (2012) provides a comprehensive literature review of cloud computing business research highlighting the paucity of research in cloud security. Our research contributes towards filling the above literature gap. For comprehensive literature review of cloud computing, please see Weinhardt et al. (2009) and Müller et al. (2015).

**Research Context**

One of the factors inhibiting cloud security research is lack of the empirical data. Recently, public cloud providers such as Amazon Web Services (AWS) have started providing logging services to their customers so that they can track their activities and resource utilization within the cloud environment (Amazon Web Services, 2015; AWS, 2017). Organizations may consume several cloud services and the activities within those services are tracked using logging services. Moreover, AWS delivers these logs in semi-structured NoSQL (JSON) file data format to the organizations. This logging service in AWS is called CloudTrail (AWS, 2017) and a similar service provided by Microsoft Azure called Log Analytics (Microsoft, 2017). The logs consist of details of events generated by consumption of various cloud services.

Organization using cloud computing from AWS may request for CloudTrail logging services. The organizations must provide the location for storing the log files which in case of AWS is S3 storage service. Once the organization has provided the storage folder name on the S3 service and enabled the CloudTrail service, the data of cloud events are collected in real-time in the S3 folders. Organizations may either take the archive of past data for analysis or may also build solutions to consume this data in real-time. In this paper, we use CloudTrail data to learn the probabilistic dependence among the cloud events especially security related events and further use it to predict these events. We use Gaussian Bayesian Network model for this purpose.

**Approach**

Bayesian Networks have been applied in the past to study various scenarios wherein the variables are categorical (Pearl, 1988). In this paper, we model the relationship among the cloud service events which are continuous variables using Gaussian Bayesian Network. Unlike categorical variables, the key challenge working with continuous variables in Bayesian Network is that it is difficult to conceptualize conditional probability distribution. The conditional probability distribution between a node and its antecedents represents the strength of probabilistic dependency (Pearl, 1988). To address this limitation of Bayesian Network with continuous variables, we assume the distribution of the variables as multivariate normal (Gaussian) and use Gaussian Bayesian Networks to learn the relationship among the events or variables. Figure 1 presents an example Gaussian Bayesian Network for three cloud events A, B, and C.

In Figure 1, \( E \) represents an event set, which is set of cloud events. We formally represent, \( E = \{ A, B, C \} \). The event A conditionally depends on the events B and C as shown in Figure 1. This dependence is formally represented as \( \{ B, C \} \rightarrow A \). The independent events such as B and C are represented as a normal distribution using their means and standard deviations. As shown in Figure 1, the event B has mean \( \mu_B \) and standard deviation \( \sigma_B \). Similarly, event C is represented by mean \( \mu_C \) and standard deviation \( \sigma_C \). The event A conditionally depend on events B and C such that the mean of the event A is a linear function of means of events B and C, and the distribution of A is represented as \( \sim N(\mu_A, \sigma_A^2) \) where \( w_B \) and \( w_C \) are weights and \( \alpha \) is a constant. \( \sigma_A \) represents standard deviation of event A. Here, we assume that the standard deviation of event A does not depend on the events B and C.
There are two ways to learn the structure of a Bayesian Network. First, the structure of the Bayesian Network is specified using domain knowledge or by the expert (Pearl, 1988). However, in case of cloud environment, it is challenging to map the application level knowledge of resource usage to the cloud infrastructure events recorded in the log files (Dutta & VanderMeer, 2014). Second, the data driven approach to learn the Gaussian Bayesian Network is dynamic and can be retrained to adapt with changing application development environment in organizations.

We learn the Gaussian Bayesian Network from the data using Hill climbing algorithm (Nagarajan, Scutari, & Lèbre, 2013). Hill climbing algorithm is a score-based structure learning algorithm for Bayesian learning. The score-based learning finds a Gaussian Bayesian network that optimizes (maximizes) the overall score of the network. The score is a measure of how well the Gaussian Bayesian Network describes the data (Margaritis, 2003). We define the score for a Gaussian Bayesian Network in Equation (1). Here, 'E' represents the Gaussian Bayesian Network of cloud events which can be represented as the posterior distribution of cloud events E given data.

\[
Score(E, Data) = P(E|Data) \ldots \ (1)
\]

Following the Bayes' rule, we can write the posterior probability distribution in terms of the likelihood function and prior distribution as presented in Equation (2).

\[
P(E|Data) = \frac{P(Data|E) \times P(E)}{P(Data)} \ldots (2)
\]

To maximize the score shown in Equation (2), we vary the structure E, treating P(Data) which is independent of structure E, as a constant in the score maximization process. Hence, the score is proportional to \( P(Data|E) \times P(E) \). We assume prior probability of events P(E) as an uninformed prior (Heckerman, 1995). With large sample size, the past research has shown that the \( P(Data|E) \times P(E) \) can be approximated to multivariate Gaussian (Kass & Raftery, 2005). Therefore, the score for event structure E given ‘Data’ can be represented as BICscore which is adapted from the literature (Margaritis, 2003; Schwarz, 1978) as shown in Equation (3). Here, \( \hat{p} \) represents the set of maximum likelihood estimates and 'd' represents the number of parameters in the multivariate gaussian distribution.

\[
\text{score}(E, Data) = \text{BICscore}(E, Data) = \log(Pr(Data|\hat{p}, E)) - \frac{d}{2}\log N \ldots (3)
\]

The hill-climbing search maximizes the BICscore to determine the Gaussian Bayesian Network with the best fit for a given data. One of the key features of the using BICscore is that it does not depend on the prior distribution of structure, i.e., the score does not require the prior information of cloud events P(E).

**Data**

We use a unique data set of CloudTrail logs from a medium size multinational analytics firm which uses AWS services extensively. The data is collected over a period of approximately one year from January 1, 2015, to December 15, 2015. AWS offers CloudTrail service to record customers’ API calls and delivers the log files to a cloud-based storage (Amazon Web Services, 2015; AWS, 2017). These files are stored in a hierarchical structure using a JavaScript Object Notation (JSON) file format. CloudTrail records provide details of the identity of the individual/employee using the service, name of API call and its

---

**Figure 1. Directed Acyclic Graph (DAG) for Gaussian Bayesian Network of cloud events**

There are two ways to learn the structure of a Bayesian Network. First, the structure of the Bayesian Network is specified using domain knowledge or by the expert (Pearl, 1988). However, in case of cloud environment, it is challenging to map the application level knowledge of resource usage to the cloud infrastructure events recorded in the log files (Dutta & VanderMeer, 2014). Second, the data driven approach to learn the Gaussian Bayesian Network is dynamic and can be retrained to adapt with changing application development environment in organizations.

We learn the Gaussian Bayesian Network from the data using Hill climbing algorithm (Nagarajan, Scutari, & Lèbre, 2013). Hill climbing algorithm is a score-based structure learning algorithm for Bayesian learning. The score-based learning finds a Gaussian Bayesian network that optimizes (maximizes) the overall score of the network. The score is a measure of how well the Gaussian Bayesian Network describes the data (Margaritis, 2003). We define the score for a Gaussian Bayesian Network in Equation (1). Here, 'E' represents the Gaussian Bayesian Network of cloud events which can be represented as the posterior distribution of cloud events E given data.

\[
Score(E, Data) = P(E|Data) \ldots \ (1)
\]

Following the Bayes’ rule, we can write the posterior probability distribution in terms of the likelihood function and prior distribution as presented in Equation (2).

\[
P(E|Data) = \frac{P(Data|E) \times P(E)}{P(Data)} \ldots (2)
\]

To maximize the score shown in Equation (2), we vary the structure E, treating P(Data) which is independent of structure E, as a constant in the score maximization process. Hence, the score is proportional to \( P(Data|E) \times P(E) \). We assume prior probability of events P(E) as an uninformed prior (Heckerman, 1995). With large sample size, the past research has shown that the \( P(Data|E) \times P(E) \) can be approximated to multivariate Gaussian (Kass & Raftery, 2005). Therefore, the score for event structure E given ‘Data’ can be represented as BICscore which is adapted from the literature (Margaritis, 2003; Schwarz, 1978) as shown in Equation (3). Here, \( \hat{p} \) represents the set of maximum likelihood estimates and 'd' represents the number of parameters in the multivariate gaussian distribution.

\[
\text{score}(E, Data) = \text{BICscore}(E, Data) = \log(Pr(Data|\hat{p}, E)) - \frac{d}{2}\log N \ldots (3)
\]

The hill-climbing search maximizes the BICscore to determine the Gaussian Bayesian Network with the best fit for a given data. One of the key features of the using BICscore is that it does not depend on the prior distribution of structure, i.e., the score does not require the prior information of cloud events P(E).
service, request parameters, service response along with timestamp. The data set contains 344,460 events stored in 31,062 files.

There are various types of events generated from different cloud services. We categorized these events in two groups (1) descriptive events (2) action events. Descriptive events are generated from API calls which queries the state of a cloud service. For example, an individual may use an API call to determine the network policy used in their cloud services. On the other hand, action events alter and modify the state of the system. For example, an individual uses an API call to start a new virtual machine in cloud environment is an action event.

We selected the events which represent customers' actions in the cloud environment shown in Table 1. The action events attempt to change the state of any system or service. For example, ‘Reboot instance’ is an action event which reboots any virtual machine. We excluded events which are used to determine the status of the system without performing any action from our analysis because these events are not likely antecedents of security events. Moreover, we represent the security events as Errors as shown in Table 1. The security related events include access denied, validation exception, and unauthorized operations. Table 1 presents the list of action events included in our paper.

<table>
<thead>
<tr>
<th>Event name</th>
<th>Cloud service</th>
<th>Details</th>
</tr>
</thead>
<tbody>
<tr>
<td>Reboot instance</td>
<td>Amazon EC2</td>
<td>It is used to reboot an instance or machine in the cloud environment.</td>
</tr>
<tr>
<td>Request spot</td>
<td>Amazon EC2</td>
<td>It is used to request a group of spot instances which are auction-based</td>
</tr>
<tr>
<td>instance</td>
<td></td>
<td>computer resources.</td>
</tr>
<tr>
<td>Terminate instance</td>
<td>Amazon EC2</td>
<td>It is used to terminate any instance in the Amazon EC2 service.</td>
</tr>
<tr>
<td>Run instance</td>
<td>Amazon EC2</td>
<td>It is used to launch a specific number of instances.</td>
</tr>
<tr>
<td>Create snapshot</td>
<td>Amazon EC2</td>
<td>It creates a snapshot of an EBS volume and stores it in Amazon S3.</td>
</tr>
<tr>
<td>Run job flow</td>
<td>Amazon Elastic MapReduce</td>
<td>It creates and starts a new job cluster or job flow in elastic MapReduce.</td>
</tr>
<tr>
<td>Purge queue</td>
<td>Simple queue service(SQS)</td>
<td>This API deletes messages from a queue and finally delete the queue as well.</td>
</tr>
<tr>
<td>Error (Security event)</td>
<td>Multiple services</td>
<td>We aggregated the number of errors from different services for every day. These errors have different types including access denied, validation exception, and unauthorized operation. We aggregated the errors daily level.</td>
</tr>
<tr>
<td>Delete snapshot</td>
<td>Amazon EC2</td>
<td>It deletes a specific snapshot. Snapshots are the backup images of virtual machine or instances.</td>
</tr>
<tr>
<td>Create network interface</td>
<td>Amazon EC2</td>
<td>This API call creates a network interface with a specified subnet for a compute instance.</td>
</tr>
<tr>
<td>Authorize security group ingress</td>
<td>Amazon EC2</td>
<td>The API call adds one or more rules to a security group. A security group is defined as a logical boundary wherein customer implement security controls and policies.</td>
</tr>
<tr>
<td>Delete network interface</td>
<td>Amazon EC2</td>
<td>The API call deletes a network interface from a virtual machine.</td>
</tr>
<tr>
<td>Create Tags</td>
<td>Amazon EC2</td>
<td>It adds or overwrites one or more tags for Amazon resources.</td>
</tr>
<tr>
<td>Add job flow steps</td>
<td>Amazon Elastic MapReduce</td>
<td>It adds new instances to an existing map-reduce cluster.</td>
</tr>
</tbody>
</table>

Table 1. Description of variables (events) used in modelling.

There are 110 unique events captured in the logs. We observe 8115 events for different error codes consisting of 27 unique error codes. The error code represents a specific form of error which includes security related errors as well. We identified two security related errors – (1) access denied, and (2) unauthorized operation. The access denied events constitute 28% of total security events. Moreover, unauthorized operation accounts for 9% of total security events. In this paper, we focus on access denied and unauthorized access accounts which together constitutes ~37% of the security related events in CloudTrail.
Results

Gaussian Bayesian Network Modelling

We model the events shown in Table 1 using Gaussian Bayesian Network and present the complete probabilistic dependency network in Figure 2. The model fit of the dependency network is presented in Table 3. There are 28 edges in the network with average Markov blanket of size 5.07. The Markov blanket determines the direct dependence of a node in Bayesian Network which includes its parents, children, and parent of children (Aliferis, Tsamardinos, & Statnikov, 2003). The structural dependency of errors (security events) is shown in Figure 2. The error depends on four action events - 'Console login', 'Create tags', 'Reboot instance', and 'Terminate instance' events. The distribution of the security error event can be represented as a normal distribution \( N(5.878 + 6.294 \times \text{Create tags} + 8.394 \times \text{Console login} + 1.25 \times \text{Reboot instance} - 10.172 \times \text{Terminate instance}, 47.9022) \). The two of four action events which are responsible for security events – Reboot instance and Terminate instance are important from domain knowledge perspective because both these events make resources unavailable and therefore users received access denied error. Moreover, other two action events, failed attempts to console login and tags creation may lead to authorization related security error.

Prediction of security events using Gaussian Bayesian Network

We use Gaussian Bayesian Network to select the events which are associated with security related events and further use to predict the security events or errors using the probabilistic dependence relationship learned from the data in described the previous section. The predictive model for our security related error is shown in Equation (4) below. \( \text{Error}_{t+1} \) represents the number of security events or error events observed on the day \((t+1)\). \( \text{Console Login}_t, \text{Create tags}_t, \text{Terminate instance}_t, \text{and Reboot instance}_t \) represents the number of events which impact security related error. Unlike approaches discussed in literature (Aliferis et al., 2003) which use Markov blanket for the feature selection, we include only parents of security events and did not include its children and children’s parents due to time precedence of parent events before children event.

\[
\text{Error}_{t+1} = w_0 + w_1\text{Console Login}_t + w_2\text{Create tags}_t + w_3\text{Terminate instance}_t + w_4\text{Reboot instance}_t + \epsilon_t \ldots \ldots \ldots (4)
\]

We can observe from the Equation (4) that for Gaussian Bayesian Network based prediction, we do not need to include all variables. Therefore, this technique not only provides probabilistic relationship among events but also used as a variable selection/reduction technique. Using the Gaussian Bayesian fitting, we learned the estimates of the weights. We evaluated our prediction with one day ahead forecasting. We split the dataset into training and test set in 80:20 ratio. The training set consists of 280 samples and test set consists of 70 samples. We build our model with \([1, (280 + t)]\) samples and test the security error prediction for \((280 + t + 1)\) samples.

<table>
<thead>
<tr>
<th>Duration</th>
<th>15 days average</th>
<th>30 days average</th>
<th>45 days average</th>
</tr>
</thead>
<tbody>
<tr>
<td>Average (ME)</td>
<td>0.13</td>
<td>3.76</td>
<td>4.62</td>
</tr>
<tr>
<td>Average (RMSE)</td>
<td>12.57</td>
<td>15.58</td>
<td>16.89</td>
</tr>
<tr>
<td>Average (MAE)</td>
<td>12.57</td>
<td>15.58</td>
<td>16.89</td>
</tr>
</tbody>
</table>

ME: Mean Error, RMSE: Root Mean Square Error, MAE: Mean Absolute Error

Table 2. Prediction of security error using Gaussian Bayesian network (one day ahead forecasting)
Figure 2. (Color Online) Trained Gaussian Bayesian network

<table>
<thead>
<tr>
<th>Description</th>
<th>Value</th>
<th>Description</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Number of nodes</td>
<td>15</td>
<td>Score</td>
<td>BIC (Gaussian)</td>
</tr>
<tr>
<td>Number of arcs</td>
<td>28</td>
<td>Penalization coefficient</td>
<td>2.94</td>
</tr>
<tr>
<td>Number of undirected arcs</td>
<td>0</td>
<td>Tests used in the learning procedure</td>
<td>497</td>
</tr>
<tr>
<td>Number of directed arcs</td>
<td>28</td>
<td>Optimization</td>
<td>True</td>
</tr>
<tr>
<td>Average Markov blanket size</td>
<td>5.07</td>
<td>Average branching factor</td>
<td>1.87</td>
</tr>
<tr>
<td>Average neighborhood size</td>
<td>3.73</td>
<td>Learning algorithm</td>
<td>Hill climbing</td>
</tr>
</tbody>
</table>

Table 3. Summary result of Gaussian Bayesian network
The results show that the model can reliably predict the number of security related error events for next day with 0.13 mean error averaged over 15 days. With the increase in the number of days of prediction, we observe a reduction in the accuracy.

**Discussion**

The security of applications in a cloud environment is of utmost importance today. The cloud security and data privacy are two of the key factors that inhibit cloud migration. In this paper, we present a Gaussian Bayesian Network approach to model the events in the cloud environment and present the probabilistic dependency among the cloud events. This dependency graph represents the antecedents and consequence of cloud events.

We use this graphical network to determine probabilistic dependence of security related errors, and further use the conditional relationship among the events to forecast security related error events for one day ahead prediction. The proposed approach has multiple practical implications for the software developers, security analysts and information systems auditors. The application development in the cloud environment involves utilizing various services offered by the cloud providers. The reliability of these services is guaranteed by the service level agreement (SLA) of the cloud provider. However, the cloud users face challenges to resolve the issues related to their applications. An error in an application may originate from any of the cloud services. Our Gaussian Bayesian Network can help developers to determine antecedents of the errors and assists them in defect discovery and fixing process.

Security analysts in the cloud computing environment must proactively determine the antecedents of security related errors (Jaatun, Rong, & Nguyen, 2012). Moreover, they also need to predict security threats in advance so that the management could take appropriate measures to avoid them. Such mechanisms are termed as preventive security measures. Our Gaussian Bayesian Network approach to security error analysis in cloud environment not only provides antecedents and consequences of security events but also reliably predicts them in future and supports preventive security measures.

Information systems auditors in organizations have difficulties in assessing the security of the IT infrastructure in the cloud environment (Wang, Wang, Ren, & Lou, 2010). The auditors are dependent on the data log files provided by cloud providers to assess the security risk in an organization (Amazon Web Services, 2015). However, to the best of our knowledge, ours is the first research which provides a Gaussian Bayesian Network based framework which IT auditors may use to audit cloud security and compliance in an organization.

Finally, our paper contributes to the literature on probabilistic graphical model (D. Koller and N. Friedman, 2009). There has been increasing adoption of Bayesian Network and Bayesian statistics in the Information systems research (Dey & Sarkar, 2000; Sarkar & Sriram, 2001; Zheng & Pavlou, 2010). Our paper contributes to the Bayesian Network literature by demonstrating its application in determining antecedents and consequences of security related threats and its prediction in an emerging field of cloud computing.

**Conclusion**

The cloud computing infrastructure logs provide rich information regarding IT operations, security, and auditing. However, there is lack of framework, models, and IT artifacts to analyze such information and derive business insights. In this paper, we present a Gaussian Bayesian Network to model the interdependencies among the cloud events. We conduct an exploratory study to examine the antecedents and consequences of security related events in applications using cloud services. To the best of our knowledge, ours is the first paper which attempts to model the service interdependency in such environment. Furthermore, we predict the number of security related events in one-day advance with mean error of 0.13 events. One of the key limitations of our approach is that the computing resources required to build Bayesian network increases rapidly with increase in the number events. One of the solutions to this limitation is to prune the set events and only select critical events before building Gaussian Bayesian Networks. Moreover, the solution presented in this paper is generalizable across organizations and industries due to standardization of cloud services data format. We plan to build a SaaS application to generate actionable insights from these logs without requiring organizations to share data with third party companies as part of our future research. Moreover, we also plan to robustly test our proposed model for different categories of malicious and non-malicious events.
References


